Emulation: Right or Wrong?
aka "The EmuFAQ"

FINAL EDITION

copyright (c) 1999 Sam Pettus (aka "the Scribe"), all rights reserved

Permission is hereby granted to reproduce this document subject to the following terms:

All copies must not be altered in any way, including but not limited to reformatting and conversion to alternate document formats, without the express consent of the author. The sole exception is for necessary formatting changes that may be required to adapt this document to suit your particular needs; however, the complete original text must be retained in as close a layout to the original as possible. For any questions in this regard, please contact the author.

No copy may be reproduced in whole or in part within a for-profit commercial publication or Internet site without the express consent of the author. The author recognizes the right of said vendors to reproduce limited portions of his work under the "fair use" clause of the appropriate sections of the U.S. Copyright Act and the Berne Convention for the Protection of Literary and Artistic Works.

Any trademark or other such indica to be found within this document is the exclusive property of its respective owner, and is reproduced here merely for the sake of reference.

Foreward

We build our computers like we build our cities - over time, without a plan, on top of ruins.

- Ellen Ullman, "The dumbing-down of programming," *Salon*, 8 May 1998

This document was originally conceived in late April of 1999 in an effort to find for myself the answer as to whether or not emulation was legal; hence, the title *Emulation: Right or Wrong*? Let me state for the record that I am not an attorney, nor am I an emulation expert. I'm a former computer consultant (onetime youthful hacker) turned writer and researcher who has used his God-given talents in an effort to put this issue at rest once and for all. I ask forgiveness from those who may know more on these topics than do I, as I have done my best to keep things simple and straightforward. I also beg the indulgence of those outside the United States who may be reading this for their own edification, as it relies primarily upon that country's legal system for its examples.

About the Author

Sam Pettus, better known to the emulation community as "the Scribe" for all of the writing and research that he has done on emulation, is a middle-aged bachelor with a wide and wild history despite his youth. Reading his job resume is almost like reading an instruction book on computing, becasue he has been involved in almost every major area of personal computer usage from one kind to another. Whether it be hacking copy-protection schemes on his old Commodore 64 as a teenager or enjoying the life of a freelance personal computer consultant as an adult, he has done just about everything from Apple IIs to the Newtek Video Toaster. It was in 1983 that he got his literal "baptism in fire" with computers, when his Baptist minister father bought him a Commodore C64 and Datasette on the condition that he write a program to help the neighborhood church manage its membership records. Since then he has worked with computers in one form or another at every job he has held since his youth, from simple clerical work to vice-president of a computer training company.

Sam's love of computers almost put him in jail, though, when he and the hacker group that he led narrowly escaped arrest as part of the Operation Sun Devil sweep against all forms of unauthorized computer hacking during the spring and summer of 1990. From that moment on he made a point of doing his best to stay on the straight and narrow and avoid any further involvement in the burgeoning bootleg videogam software market. He spent the next seven years involved primarily in the applications and training side of the computer industry, including one full year of unemployment during the industry shakeout of 1991 - 1993, and refused to involve himself with videogames again until 1997. That was when a friend gave him a shareware copy of *DOOM* early in the year, and that excellent shooter reawakened his long-dormant interest in a sorely missed pasttime. He now uses his knowledge of software and systems past and present to help the emulation community, and his sorrow at its current troubles (remembering his own with Sun Devil) were what led him to write the *EmuFAQ*.

Index

Module Three: The Implications

Y2K Addendum: The Effects

Here is a list of mirror sites that maintain copies of the *EmuFAQ*.

Eidolon's Inn --> http://eidolon.psp.net

Everything you wanted to know about Sega home videogame consoles can be found here, including the *Genesis Game Guide* - the Scribe's very first emulation-related FAQ and the definitive catalog to the Genesis/MegaDrive videogame ligrary.

The Vintage Gaming Network - http://www.vintagegaming.com

Formerly known as Dave's Video Game Classics, this is by far the most popular emusite on the World Wide Web. Although for the most part dedicated to MAME and other arcade game emulators, it does provide decent coverage of other forms of emulation, and is highly recommended as a good starting point for emunewbies.

Zophar's Domain *-* http://www.zophar.net

One of the oldest and best general-purpose Internet emulation sites still in operation, Zophar's is home to the extended Emulation Timeline, *A History of Emulation*.

Emulation HQ (EmuHQ) - http://www.emuhq.com

One of the very first NextGen emulation sites, this remains an excellent place to monitor continuing developments in emulation's newest phase of development.

OverClocked (OC) - http://www.overclocked.org

Home to the emulation comic strip "OverClocked", as well as game reviews, remixed videogame music, and random miscellaneous flotsam and jetsam

1999: The Emuyear in Review

This is a reprint of an article that originally appeared in the January 2000 issue of the *Zophar's Domain* newsletter

> *And we're going to party like it's nineteen-- .... Hold up! It is!*
>
> - Will Smith, "Will-enium"

INTRODUCTION

Back in the early 1970s, Gerry and Sylvia Anderson created what many people regard as their finest live-action television series. *SPACE:1999* was set in that ominous-sounding year, and those of us who remember it also remember its premise. On 13 September 1999, a freak nuclear accident on the dark side of the moon blew the Earth's lone natural satellite out of orbit, causing mass chaos and disaster on the mother planet and condemning the 311 inhabitants of Moonbase Alpha to a solitary existence wandering the stars. While time has finally caught up with the series and its projected concept of mankind's future, the year itself managed to retain its portentous aura. 1999 would prove indeed to be a momentous year for many, but it would prove especially significant for the rapidly growing and often controversial Internet emuscene. It was a year that was a real roller-coaster ride for them, and it was shaken to its very core just like those thermonuclear explosions shook the moon in *SPACE:1999*. In just a little while we will begin sifting through the dim memories of the past twelve months, but one significant accomplishment stands out above all the rest of that year.

1999 will be chiefly remembered by the emuscene as the year that videogame emulation was finally legitimized.

It would take more space than is available for this article to explain the how and why this came about, and in fact the build-up to this year has been addressed in other places. Most of you who are reading this are already members of the emuscene, so there is little need to repeat what many of you already know. Like me, you have followed the rapid-fire developments both on and off the emuscene, on the Internet and in the physical world, discussed in the message boards and chatrooms or debated in the courtroom. You know how the emuscene came about, and how it has been treated by both friend and foe alike up to this year. This article will therefore limit itself to 1999, and review for you those significant events of that year that helped make this possible. This was a long time building, but 1999 would prove the year that the emuscene was finally accepted by many for what it was - not what a highly vocal few claimed it to be.

THE FIRESTORM ERUPTS

The year's first notable event was the unearthing of a beta copy of Sega's *Sonic the Hedgehog 2* for the Sega Genesis/MegaDrive (G/MD) videogame console. It represented an interim game between the retail releases of the first and second G/MD Sonic games, with lots of unique stuff that never made it into the final game. Of course, it was soon dumped, and the resultant ROM made the obligatory rounds. With the IDSA's "great sweep" of 1998 finally dead and buried, it seemed that things were finally getting back to normal.

Oh, how naive the hope!

The first major topic of the year was the legitimacy of videogame emulation itself. Connectix, the well-known Macintosh support vendor, had demonstrated its ***Virtual Game Station*** (*VGS*) emulator at the MacWorld expo the previous year. This commercial product permitted the playing of **Sony PlayStation** (PSX) games on a PowerMac personal computer. While PSX emulation was not new to the emuscene (remember *PSEmu*, *Psyke*, and ***PSEmu Pro***?), this was the first retail product for any computer system to openly embrace the concept of videogame emulation. *VGS* hit retail shelves at the first of the year, and many Mac vendors were hard pressed to keep it in stock. This did not sit well with Sony Corporation, makers of the PSX, who was already waging an unsuccessful fight against the emuscene's earlier freeware PSX emulators. They announced their intention to file an intellectual property violation lawsuit against the pioneering vendor, resulting in a sudden artificial boost in *VGS* sales. Those who supported PSX emulation on the Mac were going to make sure that they had their copies before the emulator was removed from store shelves.

Two other events happened almost simultaneously with the release of *VGS* that boded ill for the emuscene. Tratrax, one of the chief members of the triad that composed the core of the *PSEmu Pro* devteam, announced that he was leaving the emuscene for a job with a game developer. Also, a leaked beta of ***bleem!*** by Randy Linden, another commercial PSX emulator but this time for Win9x systems, found its way to the Internet and "made the rounds." Both incidents were major news items at the time, and along with the *VGS* release raised the public profile of the emuscene to new (and perhaps unwanted) heights.

It was about to blast out of orbit altogether.

On 28 January 1999, a new videogame emulator came out of nowhere that rocked both the emuscene and vendor community like nothing had done before. Epsilon and RealityMan's ***UltraHLE*** was the world's first working **N64** emulator, achieving rock-solid compatibility with several popular N64 titles from day one. These included the console's flagship game, *Super Mario 64*, the much-beloved *Mario Kart 64*, and the mother of all Nintendo RPGs - *Zelda: The Ocarina of Time*. *UltraHLE* was a phenomenal achievement in that many within the emuscene, including a number of its icons, believed that N64 emulation would not be achieved until the next millenium. These two bright young emuhackers proved everybody wrong. Nintendo's response was predictable, of course - overblown, overhyped, overlegated, and perhaps pressed over-hard. They promptly declared *UltraHLE* to be an infringing product, declared emulation itself illegal, and began waging a massive campaign against the emuscene, forcing the shutdown of any website that carried *UltraHLE* or its required ROMs and seemingly threatening swift legal action against anybody that so much as breathed *UltraHLE*'s name. *Only the IDSA assault on <u>MAME</u> back in early 1998 came close to the furor that was aroused over <u>UltraHLE</u>, and the legitimacy of emulation itself was once again called into serious question by many in the mainstream press.* It would not be until later in the year that the issues raised by the release of *UltraHLE* would be finally laid to rest.

<u>DARK PORTENTS</u>

The month of February opened with enough dark portents to keep even the most depressed emufan satisfied. On the very first day of the month, Duddie became the second member of the *PSEmu Pro* team to depart, leaving only Kazzuya behind. A war of words erupted between ClassicGaming.com and Emu News Service over domain name rights that would eventually result in one of the emuscene's most respected sources of information closing its doors. The death-blow for *PSEmu Pro* came in mid-month, when Kazzuya announced that he too was leaving for greener pastures; thus, formal development of this ground-breaking emulator (the very first PSX emulator) came to an end. Finally, at the end of the month, the Federal Communications Commission (FCC) declared the Internet to be a federally recognized forum for interstate commerce, thus opening the door to all sorts of potential regulation and taxation by the government sometime in the not-too-distant future.

As if that weren't enough, Nintendo continued to fan the fires of the *UltraHLE* controversy to white-hot intensity. *They publicly announced their intentions to sue both Epsilon and RealityMan over UltraHLE, continuing to maintain that it was "an illegal emulator solely designed to play infringing copies" of Nintendo software.* They also decried the appearance of several supposed pieces of *UltraHLE* source that began to appear on the backwater sites, although all of these proved to be attempts at decompiling the executable and thus ultimately worthless. The response of the emuscene was to publicly condemn Nintendo's actions at every opportunity, and Emulation Reality attempted to organize a boycott against the arrogant videogame giant. It ultimately failed, as do most similar efforts, but the emuscene was not happy with the way things were looking for emulation. The only bright spot on the legal front came from the *Sony v. Connectix* lawsuit, when Sony was denied its initial request for an injunction against continued *VGS* sales. They promptly re-filed, however, and their action coupled with Nintendo's efforts continued to maintain a dark legal haze over what many contended to be an innocent and non-infringing pastime.

It was about this time, however, that the first cracks started to appear in the seemingly solid wall that the vendors had thrown up against videogame emulation, and it was two vendors who were responsible for this move. *Crystal Dynamics, perhaps best known for their Gex franchise, publicly broke with the official IDSA stance on videogame emulation.* It was the first good news that the emuscene had received from the vendors in quite a while. *The real vendor jewel came at the end of the month, however, when Sega of America released the*

Sega Smash Pack** to retail sales.* Enterprising hackers quickly discovered that it was in fact a slightly modified version of the popular ***KGen G/MD emulator by Steve Snake, which he later confirmed to be the case. As the Snake eventually revealed, he had been in negotiations with Sega for the better part of 1998 over a commercial release of KGen, and it was both the support and compatibility reports from the emuscene that finally convinced Sega of the possibilities. *Thus, Sega became the first videogame vendor to openly embrace videogame emulation, and KGen became the first public domain videogame emulator to be re-released as a commercial program.* It was a heady accomplishment for the cause of emulation, and it gave the beleaguered emuscene a badly needed shot in the arm. At least some of the vendors were listening now, and that was a start.

THE GATHERING STORM

The storm surrounding videogame emulation continued to build towards gale force as the month of March came and went. Several background events only added to the gloom of emufans worldwide. It was at this time that word began to spread about an embedded serial number inside of Intel's new Pentium-III CPUs - a practice which had actually been in place since the release of the first Pentium-II some years earlier - thus drawing the wrath and ire of privacy advocates everywhere. The last beta build of *Psyke*, the second working PSX emulator, was released to the public, and an attempt to restart Emu News Service wound up going nowhere. Also, to the surprise of many on the emuscene, Lazarus, the oldest and most respected **Amiga** emusite on the Internet, was forced to shut down for a wide variety of internal and legal issues.

In fact, there was precious little good news during this time. Sony's fight against Connectix over *VGS* was not going well for them, and they were twice refused an injunction by the courts against continued *VGS* sales. That was some small solace for the battered emuscene, as were the twin announcements that both of the new up-and-coming 128-bit videogame consoles, the **Sega Dreamcast** (DC) and the **Sony PlayStation 2** (PS2), would use firmware emulation to support the software bases of their parent company's older consoles - in this case, the Sega Saturn (which never happened) and the original Sony PSX (which did). RPGe released its long-awaited 100% translation patch for the Super Nintendo (SNES/SFC) game *Magical Knight RayEarth*, and that was well received by fans of this particular anime TV series. Good news for the Nintendo emuscene came in the person of the Scribe, responsible for the well-respected *SNES Chart*, who announced that he was resuming work on an SNES/SFC games FAQ along the same lines as his popular *Genesis Game Guide* (G3). Finally, word about the impending releases of both bleem! and a new Sega 8-bit emulator named ***MEKA*** set many tongues a-waggin.'

The publicity campaign Nintendo was waging against emulation apparently came to its head in March, with a number of articles highly critical of the emuscene appearing in various mainstream publications. Even so lauded a public vehicle as *Time* magazine got into the act in a well-publicized article echoing the mainstream sentiments of the past three months, in that videogame emulation was little more than a transparent cover for blatant software piracy. Tired of the raging debate and sick at the bashing at the hands of both ill-informed critics and anti-emulation activists, RealityMan of *UltraHLE* fame announced that he was quitting the emuscene. N64 emufans mourned his apparent loss for days. N64 emulation continued without him, with the most promising project being *NEMU64* by Lemmy, but even he began to hearing threats of legal action from "the big N." As if that wasn't bad enough, *Sony had the long-running and highly regarded emusite Dave's Video Game Classics shut down under threat of a lawsuit for posting a link to an illegal dump of the PlayStation BIOS* - a rather stupid move for such a well-respected emusite, in retrospect, but nevertheless an action that reminded the emuscene yet again of the IDSA's "great sweep" of 1998 - a past action that Dave's had somehow survived.

The days were growing dark for the beleaguered emuscene. It would not be long before it would suffer its worst disaster yet.

ICEBERG DEAD AHEAD!

When looking back at April, two events tend to overshadow all others. It is not that anything else did not happen during this time. Zoop's eagerly anticipated *MEKA* was finally released, quickly establishing itself as the new standard in Sega 8-bit emulation. 3Dfx began a legal crackdown against makers and distributors of so-called "Glide wrappers," which allowed non-3Dfx video cards to support 3Dfx-keyed software (and thereby run UltraHLE, nudge, nudge). Lemmy's *NEMU 64* achieved the notable distinction of becoming the second N64 emulator to actually work (i.e. both load ROMs and make them playable) - which prompted a swift yet predictable legal notice from Nintendo about possible intellectual property infringement. Dave's Video Game Classics found a new home over on the Warzone Network and reopened its doors, and thus one of the oldest sites on the emuscene was able to resume operations. The latest news on *bleem!* was especially good, with everything pointing to an official retail release at the beginning of May and Sony refused two successive injunction requests in their efforts to block its impending market release. While these were notable events in themselves, they were not nearly as big as two others. These two dominated the month of April, and their shockwaves continue to echo around us. The storm surrounding the emuscene had been building all year, and it was time for the hurricane to hit shore.

***The Columbine High School massacre** in Littleton, Colorado (USA) ripped open a long-running debate on violent content in videogames that had its roots all the way back in the coin-op days of the late 1970s.* The fact that the two teenagers responsible for the slaughter were avid aficionados of such first-person shooters as *DOOM* and *Quake* served as potent ammo for the blue-nose crowd. Liberal and conservatives alike crossed ideological lines and joined forces to demand new restraints on videogame violence, and a brand-new round of Congressional hearings took place - a mere six years after the *Night*

Trap/Mortal Kombat debate Congress held in 1993. Those of us who paid attention felt like we were watching events through a time machine, except that it was the IDSA's Douglas Lowenstein and not Nintendo CEO Howard Lincoln prostituting his industry before the politicians. Even the emuscene felt the ground shake beneath its feet, as those championing the sanitization of videogames began churning up old titles to support (or refute) the many claims being bandied about. Once again, the emuscene was in the crosshairs of the mainstream press - except that this time it was sharing the field of fire with its bigger brother, the videogame industry itself. Both found themselves in the unwelcome position of having to defend a beloved pastime yet again. The same arguments both for and against violence in videogames were emotionally tossed back and forth, and in time the public debate died down, but the issue remained like a slow-burning coal in the backs of many minds for the rest of the year. Indeed, many vendors pulled violent content from their videogame demos at such trade shows as E3 and the Summer CES, not willing to stick their necks out and risk the public wrath. A lone few held firm, holding that it was the boys themselves and not the games that were responsible for the resulting violence, but the press wasn't listening. So, the videogame industry as a whole kept quiet about the issue, and a few months later it blew over - as it always does. Perhaps talk show host Montel Williams best summed matters up when he noted the following month, "That wasn't hate that was learned from a videogame. That hate was [already] inside of them."

On 22 April 1999, in the latest round of *<u>Sony v. Connectix</u>*, *the emuscene was handed what many at the time considered to be its legal coup de grace.* ***The U.S. 9th District Federal Court***, *in reviewing a third request by Sony for an injunction against sales of Connectix's VGS, determined that* *<u>VGS</u>* *was indeed a infringing product as Sony had maintained all along.* They agreed with Sony in that Connectix had made use of unlicensed proprietary Sony BIOS code in developing *VGS*, which constituted a violation of Sony's copyright on the BIOS code. The court granted a full injunction against continued sales of *VGS* in all venues until a formal trial could be scheduled, which was later set for early 2000. One could almost sense Nintendo executives breaking out of their plush offices and dancing in the streets when the news broke. It seemed that their three-month campaign against videogame emulation had finally struck paydirt - and best of all, they didn't have to pay a penny for the results. As for the emuscene, it felt that it had been carpet-bombed by the videogame vendors, with many emufans expressing shock, disgust, and somber reflection at this sudden turn of events. ***Videogame vendors had been using emulation in their products for almost a decade. It was a proven and legal technology. What was <u>so wrong</u> about <u>somebody else</u> doing it? - or to put it in plain English, "Why should the vendors be allowed to maintain an monopoly on the right to emulate?"*** Anyway, that's how almost everybody in the emuscene felt, and many of us were at a loss as to what we should do about it.

It was the emuscene's darkest hour.

<u>THE HURRICANE HITS SHORE</u>

The month of May is seen by many as the worst month for the emuscene all year. Blown out of the water by the *VGS* injunction, blasted all month long (and for some time thereafter) by a freshly invigorated Nintendo, and buffeted by the actions of its more insistent and irresponsible members, the battered emuscene staggered on somehow, alternating wildly between events good and bad like a drunken sailor wandering home from too much liberty. It was as if we were watching someone trying to walk across an open field in the midst of a hurricane, tossed this way and that, occasionally finding one's feet, and then just as quickly knocked off again. Indeed, *the period covering 20-24 May 1999 has been termed by Eidolon as "... one of the worst weeks in emulation history,"* and those of us who were active on the emuscene at the time tend to agree with him.

First, the good news. ***DGen***, the emuscene's first decent open-source G/MD emulator, was released to the public. It would eventually prove to be for the Sega emuscene what ***SNES9X*** was to the Nintendo emuscene, and would in time displace Sardu's ***Genecyst*** as the second-best G/MD emulator available (just a hair behind Steve Snake's *KGen98*).

Now, the bad news. *The public outcry against violent videogame content continued unabated, with the Congressional hearings remaining in session and U.S. Vice President Albert Gore proposing nothing less than outright censorship of Internet content*. The politicians were planning to capitalize on the nation's mood in the wake of the Columbine massacre, and it seemed certain that new legislation would soon be proposed that would violate the U.S. Constitution's First Amendment. Both videogamers and emufans alike watched with mounting concern.

Again, the pendulum swung. *The Scribe openly challenged the validity of Nintendo's assertion that emulation was illegal in a carefully worded response to Nintendo's official emulation FAQ.* While it was by his own admission incomplete and in need of further revision, it was widely hailed by the emuscene as the first intelligent countermove to Nintendo's claims. Thus emboldened, the Scribe continued his research into the legality of emulation - a move that would have profound implications for the emuscene.

Now the pendulum swung back. *Nintendo's Mike Chandler publicly asserted that the practice of dumping the ROMs of a videogame cartridge was blatantly illegal.* His statement was based on a number of court cases that Nintendo had won in recent years concerning the piracy of videogame cartridges for all of its systems to date, as well as attempting to prosecute those companies that manufactured and sold cart dumping equipment. Interestingly enough, his statement excluded ROMs of a disk-image nature - a comment that was largely overlooked at the time.

Once again, the sun shone bright. The long-anticipated PSX emulator *bleem!*was finally released after production delays, and advance mail orders for the program skyrocketed. Dan Boris programmed the very first Atari 7800 emulator. Glide wrappers gained commercial status once Creative started shipping one with its RivaTNT-based Graphics Blaster. Even Sony's highly vocal protests over the *bleem!* booth at the year's E3 trade show and their subsequent maneuvering to block sales in retail outlets failed to daunt the emuscene's spirit. Nevertheless, all of that was about to change. The sequence of events that took place from 20-24 May 1999 would rock the emuscene almost as badly as the *VGS* ruling itself.

Early on the morning of 20 May 1999, the Scribe posted the latest version of his widely respected *SNES Chart*. The 1.91 release would prove to be the last public edition of that unique document. About the same time, a resurgent Nintendo, apparently encouraged by Sony's legal success and confident of its own might (and right), renewed its assault on the emuscene with a newfound vigor and zeal not seen since the release of UltraHLE. They had the *SNES9X* home page shut down, threatening its ISP with a lawsuit if they didn't comply with its wishes. Their move prompted *NEMU64* author Lemmy to temporarily halt development and go into virtual hiding, since his N64 emulator had achieved working status and was now a prime target for possible legal action. The next day, Nintendo took action against Spirit in the Contraption, one of the more prominent ROM sites to spring up in the wake of the IDSA's "great sweep," and managed to force it to close its doors for a time. Sony chimed in as well, publicly supporting Nintendo's actions and adding to it their assertions that any attempt to dump the PSX BIOS (for use with a PSX emulator) constituted intellectual property infringement.

Finally, on the evening of 21 May, as he was digesting the events of the past two days and continuing his legal research, the Scribe made a shocking discovery that would prove to have profound implications on the emuscene and his contributions toward it. His research into emulation was not covered under ***the fair use clause of copyright law****, thus making him potentially eligible for prosecution as well.* The following morning, fearing possible legal action from Nintendo over his *SNES Chart*. as well as action from certain other vendors over his other emulation-related efforts, the Scribe pulled every single one of his emulation FAQs from the Internet for his own protection and announced a new project. The title was *Emulation: Right or Wrong?* You know it today as the *EmuFAQ*. It was a timely effort, as many in the emuscene were by now crying for answers to the verbal and legal assault of the past few months. Taking his own situation by the horns, the Scribe was determined to settle the emulation controversy once and for all regardless of whatever it took ... or cost.

The last notable event of May also deserves mention, because it hit especially hard for Sega emufans. The world's first working Sega CD emulator (that could actually play games), ***SuiCiDE***, was killed by its author before it was ever publicly released. Leila, as she was known by the emuscene, had been constantly hammered by impatient emufans who wanted the emulator "right now" regardless of its performance. They were quite used to dealing with imperfect emulation, and so naturally expected that Leila would come up with the goods. Instead, she chose to publicly break with the emuscene, and the behavior of these impudent little cretins was cited as the prime reason for her actions. SuiCiDE won Leila a job with an unnamed commercial developer, but the program itself would never appear on the emuscene. Again, *the actions of a* *belligerent few had ruined a promising development for the responsible rest*, and many and long were the debates that raged within the emuscene over the affair.

THE TURNING POINT

At the beginning of June, *Salon* magazine ran a series of article by Howard Wen under the banner, "Why emulators make videogame makers quake." It was one of the first mainstream pieces to probe the motivations of the vendor community and their often virulent attacks on the emuscene, and it nailed the issue squarely on the head. *The reason why certain vendors were so afraid of third-party emulation was not because of software piracy, as was so frequently quoted. Rather, the vendors were forced to face the unpleasant possibility of losing their respective monopolies over their artificially generated markets.* Why buy a PSX for US$100 or so when one could buy bleem! for US$40? For that matter, why buy the faltering N64 for US$150 when *UltraHLE* cost nothing and N64 ROMs were available on the Internet for anyone with the patience and diligence to smoke them out? Also, one no longer needed to worry about cross-country market locks or fancy modchips - emulators pretty much ignored these things and allowed videogamers worldwide to enjoy titles that they had never been intended to see. *The videogame vending community was losing its stranglehold over its customer base, and emulation was making that possible.* No wonder proprietary-minded vendors like Sony and Nintendo were so upset.

Good news for the emuscene came early in the month, when Eidolon's Inn reported that the Scribe would be resuming both G3 and the *Genesis Chart*. All legal questions regarding Sega-related emulation that had plagued the Scribe since his ill-fated discovery were resolved with the assistance of Steve Snake, *KGen* author and longtime videogame programmer. With that, the Scribe officially resumed his work on the Sega emuscene, much to the delight of its fans. No such aid was to be had from Nintendo, however, who officially denied him permission to continue using emulation in researching the software base of its systems. It resulted in the immediate (and apparently permanent) suspension of the *SNES Chart*, and threatened the SNES/SFC software FAQ that the Scribe had wanted to write for so long.

The next target on the radar turned out to be the all-too-brief revival of *PSEmu Pro* - one that would quickly die. Again, the ruffians and hooligans at the fringes of the emuscene were to blame. A promising beta of ***PSEmu Pro 2*** was leaked by an inside source and spread like wildfire, with the emusite Emulation Ireland going so far as to actually post it. Infuriated, the new *PSEmu* devteam abruptly abandoned the project and publicly condemned the emuscene for its behavior. Coming on the heels of the *SuiCiDE* debacle as it did, and reviving unwelcome memories of the *NeoRAGE* hack of 1998, the sudden and swift demise of *PSEmu Pro 2* gave the emuscene new cause to ponder its public image - along with the unwelcome behavior of its less savory adherents.

The legal issues surrounding emulation continued as the battle between Sony and the two commercial PSX emulators raged on. Sony began to subpoena *bleem!* betatesters for their dispositions in their efforts to prove that the bl*eem!*

code infringed upon Sony's copyrights. One of the side effects of Sony's move was the temporary shutdown of Matthew Elzer's EliteGamer website, as he was one of *bleem!*'s more prominent betatesters. 3Dfx began what is now perceived by many to be the start of its public downfall when it filed suit against Creative Labs over the inclusion of a so-called "Glide wrapper" for its non-3Dfx based Graphics Blaster video card. *Perhaps the biggest legal news of the month was the bill proposed to the U.S. Congress by Republican representative Henry Hyde, which sought to censor possibly violent, pornographic, or any other kind of content that could be perceived as being offensive.* It was shot down by the House of Representatives in a 282-145 vote against, and rightly so, with both Democrats and fellow Republicans alike joining ranks to kill the Hyde bill. The stated reason was also the proper one, as the proposed bill would have been in violation of the First Amendments rights of free speech and freedom of expression. With that, the public outcry over the Columbine affair pretty much disappeared from the radarscope, aside from the occasional mention, for the rest of the year, and both the videogame industry and its fans could breathe easier. Yet another attempt to censor videogame content had been thwarted ... for now.

The rest of the month's activities on the emuscene are brief, yet notable. Tristar64, the world's first Nintendo 8-bit (NES/FC) and 16-bit (SNES/SFC) cart adapter for the 64-bit N64, was announced. ***NeoCD***, the very first working NeoGeo CD emulator, made its debut. Two promising GameBoy emulators, ***DBoy*** and ***HelloGB***, were suspended due to the possibility of legal action by Nintendo against their authors. Finally, in a move long awaited not only by emufans but the videogame community at large, word began to spread that Square would release *Chrono Cross*, a PSX sequel to the legendary SNES/SFC RPG *Chrono Trigger*.

The last significant event of June is perhaps the most important one of the month in terms of legitimizing the emuscene. *On 29 June 1999, Nintendo quietly posted a new, revised version of its policy statement regarding videogame emulation. To the open shock and surprise of the emuscene, Nintendo's new FAQ now acknowledged the existence of unlicensed yet legal ROMs for Nintendo-based emulators. It also implicitly acknowledged the legal possibility of unlicensed videogame emulation as a whole.* Credit for this breathtaking change in public policy can be rightly credited to the Scribe, as early drafts of his *EmuFAQ* were by now beginning to circulate on the Internet. It was as if he had assumed the role of the news anchor so brilliantly portrayed by actor Albert Finney in the movie *Network*, who proudly proclaimed to his audience, "I'm mad as hell and I'm not going to take it anymore!" The Scribe had caught Nintendo in several factual errors regarding their assertions about the emuscene, and the videogame giant had no choice but to grudgingly acknowledge his research. It was a stunning achievement in the Scribe's one-man war over the legality of emulation, and many began to pay more attention than before to his work on the *EmuFAQ*. Thus began what the Scribe calls "a curious relationship" between him and Nintendo as work on the *EmuFAQ* continued, and both the vendor community and the emuscene began to take his efforts more seriously.

The mouse had roared.

QUIET TRIUMPHS

July was a rather mild month on the legal front of the emuscene. *Bleem!* finally hit store shelves, having been delayed by Sony's pressuring tactics on the retail industry. Nintendo's activity on the emuscene subsided somewhat, with its only major move being against nodeX over possible infringement of its highly profitable Pokemon franchise. Over at The Dump, Harry Tuttle was forced to abandon his cherished *Dracula X*
Project - the posting of the CD image of one of the best games ever written for the ***Turbo Graph/X 16*** (PC Engine for you Japanese fans, TG16/PCE for short), - after failing in negotiating distribution terms with Konami, the game's original vendor.

Perhaps the biggest news of all legalwise was the <u>*RIAA v. Diamond*</u> *decision, i.e.* ***the "Rio case,"*** *in which the courts ruled that converting a CD-audio recording to MP3 format for personal use on Diamond's Rio MP3 player was in fact legal. There was open speculation that this decision could be applied to ROMs at some point in the future, and some posited that it was a de facto justification of the current practice of ROM dumping.* While such was (and is) not the case, as the Rio case dealt specifically with audio recordings and not with multimedia efforts as a whole, it nevertheless demonstrated the growing acceptance of such format conversions - provided that they are of a purely non-commercial nature. Hope continues to abound that a similar decision will apply to ROMs someday.

In fact, most of the month was preoccupied with emuscene-specific news. ***ZSNES***, by far the most popular SNES/SFC emulator, resumed updates with the v0.9xx series after a half-year lull and also became the first such emulator to support the games *StarFox* and *Vortex*. Factor 5, the noted German videogame programming powerhouse, joined the growing number of vendors supporting the emuscene by posting ROMs of five of its most popular Amiga games. Borland released its venerable *Turbo Pascal* and *Turbo C* DOS-based compilers to the public domain, and this was notable in that they were among the most powerful such compilers of their day. All of this was fine and good, but one event that happened at the end of the month would dominate the news for days to come, and give the battered emuscene yet another needed shot-in-the-arm in terms of good press.

On 28 July 1999, a development team headed by Neill Cortlett released an English translation patch for the ROM of the SNES/SFC RPG <u>*Seiken Densetsu 3*</u>*.* Known in the U.S. as *Secret of Mana 2*, the game had never been released in this country by its vendor Square for a number of reasons - a decision that had caused much grief and anger among RPG fans. The patch was an instant hit, but its popularity went way beyond the emuscene. News that *Secret of Mana 2* had finally "made it" spread like wildfire among the online Nintendo gaming community, and *Seiken Densetsu 3* had climbed to the #6 spot in GameFaqs' top ten list by the end of the month. Pretty good for a four-year old RPG, eh? "Either this is a mix-up with the *Legend of Mana*, or the translation has struck a chord throughout gamers everywhere," noted taichou of Zophar's Domain in a news posting. *In retrospect, the* *Seiken Densetsu 3* *patch probably did more to increase public acceptance of the emuscene in the eyes of "straight" videogamers than anything else since the unspoken acceptance of the emuscene's "900-pound gorilla," MAME.*

The tide was turning ... but the best was yet to come.

THE SLEEPER AWAKENS

By August of 1999, a number of mainstream Internet videogame sites had small sections devoted to emulation, including one or more emulators for various classic consoles and arcade hardware. That was a far cry from the beginning of the year, when some of these same sites were belittling or even condemning it outright. Public perceptions on emulation were in the process of changing for the better, and the claims of "infringement" and "software piracy" were beginning to fade away as new fans discovered and old hands
relearned the benefits that emulation could bring. Thanks to the *Emulation Timeline*, a side product of the Scribe's *EmuFAQ* project, the emuscene was now aware of its heritage, while the developing *EmuFAQ* reminded them of their responsibilities. Attitudes towards emulation were changing both without and within the emuscene.

Mention must be made of an emulation hack that was used to solve a problem with the existing betas of *Windows 2000* (*Win2K* RC1) that were circulating about this time. A number of betatesters had reported problems with support for certain models of SoundBlaster and SoundBlaster compatible audio cards. The solution was found in the *SoundFX-NT* program by Software Solutions, designed originally for WinNT but found to work just as well with Win2K. Once again, emulation was employed to resolve a hardware problem - just as it had so many times before in its storied history.

August also market the v3.0 release of ***Amiga Forever***, the commercially vended version (by Cloanto) of the popular ***WinUAE*** cross-platform emulator. *To date, it is the emuscene's only commercially vended BIOS-dump dependent emulator to include licensed copies of the required BIOS and system software along with the emulator itself - in this case, a Kickstart image and copies of the AmigaDOS system software.*

It was also at this time that some enterprising emuhackers caught Sony with their pants down. They determined that Capcom's PSX re-release of the NES classic *MegaMan* (*Rockman* in Japan) was little more than a modified version of the original ROM running under emulation. This was in stark contrast to Sony's public assertions about the supposed illegality of third-party videogame emulation, and Sony wisely chose not to comment on the matter. Capcom was merely continuing in its unspoken support of videogame emulation, as it had permitted the licensing of several CPS-1 arcade ROMs for inclusion with the new HotRod arcade-style joystick back in June. What was good for personal computers apparently worked just as well with the PSX - and best of all, Capcom was making money off of emulation (unlike somebody else). Sony might not approve of videogame emulation, but it couldn't stop one of its best-known licensees from using a scaled-down Nintendo NES/FC emulator to make a fast buck or two.

An interesting bit of news regarding the emuscene's past came around mid-month, when MSNBC ran an article about software piracy on their website. The article referenced a *GameWeek* interview conducted with Sega's **Yuji Naka**, creator of *Sonic the Hedgehog* and *NiGHTS: Into Dreams*. In the interview, he revealed that he had written an NES emulator back during the heyday of the Sega Genesis, and that it gave him "great satisfaction" to see Nintendo software running on Sega hardware. *This pushed the date for the first known "true" videogame emulator back to 1991, and Yuji Naka's NES/FC emulator thus supplants Argonaut Software's Gameboy 68000 as the first such program in existence.* The fact that such a respected programmer had dabbled in emulation even before the emuscene existed was heady news to emufans. The news of Naka-san's accomplishment only added more fuel to the fire that was stoking the growing acceptance of emulation as a legitimate practice.

There were several other events that occurred in the emuscene proper during August. It was at this time that the Scribe finally abandoned his SNES/SFC research, having reached a point where he simply could not continue without the use of emulation (an avenue for which Nintendo continued to refuse permission). Hong Kong videogame bootleggers finally "broke" the Sega Dreamcast's unique 1 GB custom CD-ROM format (called GD-ROM by Sega) after ten months of its use on the market. Also, the devteam for ***1964*** (a new N64 emulator) quit posting open source code after another N64 emulator devteam plagiarized it and claimed it as their own.

A sharply worded article regarding videogame emulation by T. Liam MacDonald titled "You will be emulated" appeared in the September issue of MaximumPC, released in mid-month in order to hit the racks in time. It was a honest assessment of why the emuscene existed and why videogame emulation should be accepted. One of its oft-quoted passages was this: "The statement that the only purpose of emulators is 'to play illegally copied games from the Internet' is just a lie. Once you buy a game, you can play it wherever and whenever you want. It's yours. If you want to patch together a lawnmower, a Watchman, and a pair of ice tongs to play *Sonic Cruises for Hookers*, or anything else that [the vendors are] publishing, that's your right." MacDonald's opinions were shared by many, not just in the emuscene proper, but its adherents would be provided backing for their beliefs faster than anybody expected.

24 August 1999 is a date that will be forever remembered by emufans. That was the day on which videogame emulation was legitimized. *On that day, Judge Charles Legge of the 9th Federal District Court threw out Sony's third request for an injunction against bleem! retail sales, noting Sony had failed to prove intellectual property infringement on the part of Bleem LLC.* Nobody blamed Bleem LLC for celebrating their victory, and company CEO David Herpolsheimer wasted no time in voicing his opinion. "Sony had all the weapons on their side this time ... and they still couldn't make a case against *bleem!*" he said in an interview with GameSpot. The official *bleem!* website posted a message of thanks from Bleem LLC to its many customers and supporters later that day, saying, "This is a great step for *bleem!* and we want to thank all of you, because without you, we would have never made it this far.... Thanks again!" While cooler heads reminded the emuscene that the actual trial was still almost a year away, it did little to dampen everybody's enthusiasm. Judge Legge's decision had effectively negated the *VGS* ruling back at the end of April, and even those in the mainstream press acknowledged the emuscene's sudden turn of good fortune. From the depths of legal limbo to the shining light of public and legal acceptance, the emuscene had not only survived, but triumphed.

24 August 1999 ... a day that will live in emulation history.

NEW LIFE, NEW DEVELOPMENTS

Thankfully, the month of September did not turn out to be the disaster that had been so dramatically predicted in *SPACE:1999*. How could anything be a disaster after emulation's victory the previous month? As it was, September turned out to be a rather mundane month for the emuscene, with only the occasional negative byline or two to keep everybody's perceptions down-to-earth and the by-now requisite "big event" at month's end.

Good news first came to the Nintendo emuscene in the form of resumed development of both *DBoy* and *HelloGB* - both emulators having been suspended back in May for legal reasons. At the same time, and to the delight of emufans worldwide, RealityMan announced that he would resume work on a new and enhanced version of *UltraHLE*, the first and by far still the most powerful N64 emulator of the year. Complete English translation patches for the RPGs *Sailor Moon: Another Story* (SNES/SFC) and *Radia* (NES/FC) were released. Finally, RealityMan posted a long and rather pointed editorial on the *UltraHLE* website in which he noted that the same miscreants responsible for killing both *SuiCiDE* and *PSEmu Pro 2* were now apparently after him and his emulator. He noted that they were seemingly "intent on smearing and destroying the *UltraHLE* name and project as well as other emulation projects (e.g. *PSEmu Pro*, probably one of the most promising PSX emulators there was) and I WILL NOT let this happen." His posting garnered widespread support across the emuscene, as might be expected in the wake of the aforementioned *PSEmu Pro 2* debacle.

Some interesting events were occurring in the mainstream videogame community as well that also captured the fascination of the emuscene. The Sega Dreamcast, the industry's first 128-bit videogame console, made its North American debut on 9 September - shattering all launch records set by the PSX some four years earlier and more than making up for its lackluster Japanese launch back in November 1998. Sega was back, in a big way, and everybody was rooting for them - including their many supporters within the emuscene. Nintendo announced that the venerable GameBoy would be getting a serious overhaul, which would make it the industry's first 32-bit handheld portable. Beam International released some two dozen of their classic Amstrad CPC, C64, and Spectrum titles to the public domain, thus joining the steadily growing ranks of those vendors either openly supporting or willing to work with the emuscene.

The emuscene finally became aware of its true origins near the end of the month, when the Scribe revealed to all concerned how the concept of emulation had actually arisen. He credits the information to the Professor, one of his *EmuFAQ* contributors, who in turn credits it to computer historian Emerson Pugh. It seems that emulation's origins lie back in 1964 during the development phase of the IBM System/360 mainframe computer family, and was devised by systems engineer **Larry H. Moss** as a means of allowing IBM's new product line to run programs originally designed to run on the older IBM 7070. ***Today, Larry Moss is considered to be the "father of emulation," and one of the oldest and most respected U.S. technology firms, International Business Machines (IBM), deserves credit for having been the vendor that made the concept of emulation a reality.*** The emuscene thanks the Professor for alerting its members to the real story behind the birth of their favorite hobby, and you can find the full account in the Articles section of Zophar's Domain.

The only two blots on the record for September for the emuscene were the Amiga announcement and the resumption of the *Elite* dispute. Long-suffering Amiga fans, who had been eagerly expecting a fourth-generation version of the ubiquitous "chameleon machine," were heartbroken when the announcement came that it simply wasn't going to happen. Gateway's newly acquired Amiga subsidiary would concentrate on software instead, retooling their programs for industry-standard hardware. Somewhat later in the month, in an event that literally came out of the blue, co-authors Ian Bell and David Braben resumed their long-standing legal squabble over the rights to *Elite*, the legendary sci-fi strategy videogame and widely acknowledged as one of the true classics of the genre. The legal status of *Elite* remains in doubt as of this date.

End-of-month events were by now almost de rigeur to the emuscene, and September proved no exception. *25 September 1999 saw the release of the complete edition of the EmuFAQ - the Scribe's commentary on the state of emulation and the first in-depth scholarly analysis of its kind regarding the emuscene. To say it was controversial would be an understatement.* It was one of those documents where people either praised or denounced it, with nary a fence-straddler to be seen. Many in the emuscene felt betrayed that the Scribe had reversed his position on *UltraHLE*, declaring it to be an nothing less than an infringing emulator. Nintendo, who had been one of the few vendors to aid the Scribe in his research, was of course pleased by this development, but remained unsatisfied in that he left the door open as to the possibility of a legal unlicensed emulator derived from Nintendo hardware. Friend and foe alike praised his insights into the background of emulation and its unwelcome association with software piracy, yet at the same time were quick to challenge his assertions regarding the limits of legal emulation without researching his sources for themselves. The *EmuFAQ* was a no-holds-barred, pull-no-punches document that cast a critical eye on almost every aspect of the emuscene, and it ended by the Scribe blasting both vendors and emufans alike - the vendors for their monopolistic attitudes and the emufans for their irresponsible actions. In short, it ticked off almost everybody - which mean that the Scribe had probably achieved his goal of writing a balanced document. Its impact and influence were not immediately obvious, and the controversy within the emuscene over the *EmuFAQ* would rage well into October.

A TIME FOR REFLECTION

October was probably the most quiet month for the emuscene all year. There was no major event all month in the courtroom, and no big event at the end of the month to grab everybody's attention. This brief lull was welcomed by all sides in the emulation debate, with the vendors attending to other matters and the emuscene keeping largely to itself.

Not much happened this month on the legal front. IDSA president Douglas Lowenstein, seemingly ignorant that the Columbine massacre was now old news, voiced plans to restrict sales of videogames with mature or offensive content to minors and to limit such content on the Internet - a move that was for the most part ignored by the industry he was supposed to be representing. *Bleem!* v1.5 was released, significant in that it was the first version to use an INI (initialization) file for storing its game compatibility database. It was similar to the technique that *UltraHLE* used, and PSX fans quickly discovered that it could be used for the same purposes as the one used by the popular N64 emulator - game cheats and user hacks. Finally, Sony began taking yet another round of depositions by anybody even remotely connected with bleem!'s development as it prepared for the up-and-coming formal legal trial in early 2000.

All was rather quiet on the emuscene, too, save for the often emotional debates that raged on the message boards and in private emails over the *EmuFAQ*. The commercial release of *Final Fantasy V* in the U.S. for the PSX was noted, as fans of the series acknowledged that the groundbreaking English translation patch by RPGe for the original SNES/SFC version was in many ways superior to Square's own for the PSX release. An old friend put in a new appearance, as the most unique port yet of the venerable multi-system arcade game emulator *MAME* appeared - this time for the Kodak Digita D265 digital camera. There were also new translation patches (*Super Gajin 2, Fire Emblem Gaiden*) and new hacking utilities (Eidolon's *SCDConv*), but by and large the real action was taking place in the popular forums of scathing denouncement and emotional flame.

By mid-month, the row over the *EmuFAQ* had become so great that the Scribe felt the need to comment. In an extensive interview conducted with MDMaster of Genesis Temple, he explained how the final draft of the *EmuFAQ* came about and why he had changed his position on *UltraHLE*. In a seeming nod to his many detractors, he made the following observation

> I wanted the *EmuFAQ* to be a statement of the facts and the law, not illegal desire and emotional ranting. I'm not going to get caught up in silly word games about the technical difference between an adaptation and a derivative work, because the law treats them as one and the same. I'm not about to devolve into the 'personal use only' debate, because that's not what the *EmuFAQ* was about. It

> was about defining concepts and finding the true limits, not blurring the lines between legitimacy and piracy. For that matter, I'm not about to let others pervert the *EmuFAQ* into justifying illegal activities. The emuscene needs to know for its own good where the line lies that divides black from white instead of obscuring it in multiple shades of grey, because that's the only way we're ever going to get our reputation back.

The controversy died down after the interview, and even those who had seemingly had their feathers ruffled the most now acknowledged his intentions. Today, the *EmuFAQ*'s influence both within the emuscene and upon mainstream perceptions continues to flourish. It is recommended to others by both emufans and vendors alike, and its impact on the public can be measured in part by the tone and direction of the many articles that have since been about emulation written in the trades. *The average man on the street now understood what emulation was all about, and he was more than willing to give it the chance that it deserved.*

THE VENDORS CHIME IN

November opened with what has to be the most important development in the personal computer industry since its inception. *An initial "finding of fact" was issued as part of the process towards rendering a final verdict in the long-running U.S. v. Microsoft legal battle. It essentially stated that Microsoft had routinely engaged in unfavorable and often monopolistic practices against any would-be competitors.* The court may as well have painted the word **GUILTY** in bold block letters several hundred feet tall around the sides of Devil's Tower. To anyone with even the slightest knowledge about the lawsuit and how it had come about, there was but one conclusion to be reached - "Microsoft holds a monopoly on the software industry." While the real reasons behind the lawsuit and the finding of fact continue to be a subject of debate, and the possible implications of a recognized Microsoft monopoly remain to be seen, Microsoft itself remained strangely quiet on the matter. It was a severe blow in their legal dispute with the federal government, but Microsoft chairman Bill Gates was known for his uncanny ability to deal with sudden changes in fortune, and the industry waited to see what would happen next. Rumors of a possible settlement flourished until the end of the year, but nothing definite materialized.

Meanwhile, on the emuscene, November marked the official return of Bloodlust Software (minus Sardu) and news that Steve Snake was going to resume work on *KGen* in early 2000. The very last text-only version of the *Genesis Game Guide* was posted, having outgrown its format, and the Scribe chose to release G3 to public domain as a gift to the emuscene before transferring his efforts to its heir, *G3O*. The first edition of *Zophar's Newsletter* (i.e. this newsletter) was issued as part of the third anniversary celebration at Zophar's Domain, and the first issue of David Lloyd's popular *OverClocked* comic put in a well-received appearance later that month. The first working NeoGeo

Pocket emulator made its debut, and two more complete ROM translations (*Sanrio Smashball* for SNES/SFC and *SD Splatterhouse* for NES/FC) made their way to the boards. Things appeared to be pretty much back to normal for emufans, but the news that arrived at the end of the month (surprise!) was especially sweet.

In a joint venture with longtime partner NEC, Sega of Japan announced on 30 November 1999 that they would make available over 300 vintage G/MD and TG16/PCE ROMs on its website in Japan for Dreamcast owners. The ROMs would be usable under appropriate emulation software for Sega's 128-bit console, and could be downloaded for the modest sum of US$1-3 apiece. This bit of news delighted the emuscene to no end, and it further cemented Sega's positive reputation among them. Sega had started off the year by openly endorsing commercial emulation with the Sega Smash Pack, and it was ending the year by openly endorsing the Internet emuscene - at a profit, of course, but one that did not attempt to "rip a new one" with its many customers worldwide. *North American and European Sega fans began to cry for a similar service in their markets, and the emuscene smiled in knowing understanding. It was the first such move by any vendor, and emufans were confident that others would follow before long.* If there had been any doubt about the legitimacy of emulation support via the Internet before this, then Sega's announcement had neatly done away with it.

The emuscene had finally been vindicated.

THERE AND BACK AGAIN

The end of the year found emulation a long way from the position in which it had been when the year started. It had gone from being the onus of vendor ire to the darling of an increasingly tolerant videogame market. It had seen its public reputation shift as well, with its perceived shroud of possible illegality replaced by an air of public indifference. It had gained the mainstream's attention, and they were now beginning to give it the respect that it was due. Like it or not, emulation was here to stay.

There were a number of major events on the legal front, each being unique in its own way. On 1 December 1999, law enforcement officials in Hokkaido, Japan filed formal charges against a college student for maintaining an illegal ROM site that was distributing unauthorized NES/FC and GameBoy games. *Nintendo had neither forgotten nor forgiven "free emulation," but for now wisely concentrated on prosecuting cases it knew it could win - such as chasing down unauthorized Nintendo ROM sites.* 3Dfx finally threw in the towel on the "Glide wrapper" front, and posted both the actual Glide APIs and the source code on their own web site. It was another blow for what had once been the most respected 3D-accellerated graphics chipset manufacturer in the industry, marking yet another waypoint in their fall from grace. Back in the videogame market proper, though, and in a move that should not have surprised anyone, Bleem LLC countersued Sony for engaging in anti-competitive trade practices in the latter's underhanded efforts at trying to prevent bleem! from making it to market. Herpolsheimer and company were only following in the footsteps of other past unfair trade victims, such as Lewis Galoob Toys (the NES Game Genie) and Advanced Micro Devices (the Am386DX CPU), and past precedent seemed to favor an eventual out-of-court settlement sometime in late 2000. Finally, the "violence in videogames" controversy reared its ugly head once again, but this time in a new venue - Brazil, of all places. A well-publicized shooting caused by a drug-crazed videogame fan resulted in total ban of such shooters as *DOOM* and *Duke Nukem 3D* by Brazilian authorities, and more titles were slated to be added to the ban by year's end. It was Columbine all over again, except that this time the players involved were speaking Portuguese.

*In other related news, Nintendo finally won a clear-cut victory against **Bung Enterprises** in their long court battle over unlicensed N64 and GameBoy cart dumpers.* The court ruled that Bung had committed multiple violations of Nintendo's patents, copyrights, and trademarks with its devices, and the court granted Nintendo a permanent injunction against the sale of such devices by Bung in North, Central, and South America. Anybody who had read that section of the *EmuFAQ* which discussed the legalities of ROM dumping could have predicted the outcome. As it was, it was pretty evident to all parties concerned that Bung had willfully and egregiously violated Nintendo's intellectual property on numerous occasions and in various ways. They had publically defied Nintendo, and now was the time to pay the price. Oh well, so much for ROM underground's forlorn hopes that this would turn out to be the case that would somehow justify unauthorized ROM dumping. Nintendo was sticking to cases it could win, and this time it paid off.

There were also a number of events that kept the emuscene rather happy throughout the month. Emuhackers discovered yet another PSX title employing emulation, and it turned out to be none other than Square's re-release of *Chrono Trigger*. The actual game code proved to be nothing more than a modified SNES/SFC ROM dump, and was found to work under some of the popular SNES/SFC third-party emulators. Paul Robson and David Winter released the world's first emulator for the very first videogame console - the **Magnavox Odyssey** - which was quite an accomplishment due to the console's use of pre-chip (ancient!) discrete logic circuits. Rumors also began to circulate of a commercial vendor for the *HiVE* arcade videogame emulator, which, if true, would mark it yet another emuscene product that would join the likes of *KGen98* and *WinUAE* and "go commercial." Dave Spicer's ***Sparcade***, one of the oldest programs and the first multi-platform videogame emulator on the emuscene, received a long-awaited and much-needed update.

On a lighter note, it seems that Sega and NEC's move to profit from the Internet emuscene struck a chord with at least one of its competitors. Nintendo, longtime rival of both, announced plans for a NES ROM downloading service for N64DD owners (N64s with CD-ROMs) in Japan. It was to be a joint venture with the Randnet service, and Randnet president Masanori Tanaka confirmed that **the games would be played under emulation**. This move was seen by many in the emuscene as a complete about-face for Nintendo. Truth be told, they had actually began contemplating the move back in the days of the UltraHLE affair and had said as much in a public statement from that time. It had taken Sega's bold initiative to finally smoke them out from their earths, it seems. *The one company that had for long been the bitterest foe of the emuscene now found it necessary to change tune and join their ranks - at a profit, of course.* While details were still sketchy at press time, Nintendo's suggested download fee was reported as US$1 per NES ROM - comparable to Sega and NEC's initial offering. *The affair makes for an interesting epilogue to Nintendo's battles with the emuscene in 1999.*

The big news on the emuscene for December, however, was the return of David and Cedric Michel's ***Magic Engine****.* This all-new edition of the well-respected Turbo Graph/X 16 and PC Engine (TG16/PCE) emulator, termed by [E]mulation [N]ever [D]ies as "the only [videogame] emulator worth buying," boasted the usual assortment of new features. Two stand out above all others, however - support for CD games and the inclusion of a custom "System Card" for CD support. *CD game support had long been anticipated by fans of this emulator, but what made it especially sweet was the inclusion of the fully reverse-engineered custom System Card ROMs. This eliminated the need for any possibly infringing dumps of actual System Cards, thus permitting Magic Engine users to be able to play their CD games for the first time in a completely legal manner. To my knowledge, this was the first time that a freeware videogame emulator for a CD-based videogame console (or accessory) was developed that could be used without a ROM dump from the original console.Bleem!* had paved the way, and now the emuscene proper was beginning to take its first faltering steps down the trail it had blazed. This was a big move forward for the emuscene, and it is hoped by some that this will establish a new trend to be followed by those striving to emulate other CD-based consoles - such as the Sega CD, Sega Saturn, NeoGeo CD, and even the Sony PlayStation.

1999 .... what a long, strange trip it's been.

INTROSPECTION

It is simply amazing to look back and see what has happened in the past twelve months. The emuscene that came out of 1999 is quite different than the one that started the year. We have gone through one of the darkest moments in our shared experience ... and survived. We have experienced what may be the emuscene's highest point yet ... and rejoiced. We have shared and witnessed events that brought us both joy and grief, and many of us are now the wiser for our journey. Yes, 1999 was quite a year for the emuscene.

I would like to thank Zophar, Lycia, and the many kind patrons of Zophar's Domain for allowing me to share the year with you, and once again - if only for a few days - relive that "long, strange trip" that was 1999. I beg your indulgence if it seems to some of you that perhaps I may have stressed my part overhard, for I was one of the prime movers and shakers on the emuscene this year. It is a lifelong belief of mine that history is best written by those who have actually lived it, and no one can argue the fact of what I went through this year along with the rest of you. If there remains any doubt as to the part that I played, then name somebody else who was able to make Nintendo admit that they were wrong on anything regarding emulation, and then I will gladly step aside.

While mine may have been one of the more prominent roles that were played, mine was not the only one. There are many others whose parts were more important than mine, whose deeds I have hopefully recounted with proper justice.

> Connectix ... Randy Linden and David Herpolsheimer (bleem!) ... Sony ... Epsilon and RealityMan ... Nintendo ... Sega ... Steve Snake ... Lazarus (R.I.P., you will be missed) ... RPGe ... Lemmy ... Dave's Video Game Classics (now the Vintage Gaming Network) ... Howard Wen ... Diamond Multimedia ... Duddie, Tratrax, and Kazzuya (*PSEmu Pro*, R.I.P.) ... Neill Cortlett and the entire *Seiken Densetsu 3* translation team ... Yuji Naka ... T. Liam MacDonald ... the honorable Charles Legge ... Larry M. Moss ... the IDSA's Douglas Lowenstein ... Paul Robson and David Winter (the Odyssey emulator) ... the boys at Bloodlust ... David Lloyd ... David and Cedric Michel ... and more.

There are also many others that, while their deeds were less notable, were no less important and though unsung helped share the many burdens that the emuscene bore. We weathered the field of fire together, and we survived. I in my own small way helped make that possible, and I am proud of our shared accomplishments.

You should be, too.

There is an oft-repeated quote by George Santayana that is now a cliché of sorts. "Those who fail to learn from the past are doomed to repeat it." In retrospect, it seems that parts of this year echoed the Apple-Readysoft debate of 1988-1989, in which the legitimacy of emulation was challenged for the very first time. One would think that the vendor community would have learned its lesson then, but it didn't. One would think that today's emuscene could have learned from past precedent, but few today were aware of what happened during that time. While neither side is completely innocent of the confusion and cross-contentions that were the chief cause of concern this year, I hope that both are now better aware of their heritage and responsibilities with regards to the emuscene. One does the right thing not because one **must**, but because one **wants**. Perhaps, now that 1999 is behind us, both vendor and emufan alike can work together to allow the emuscene to flourish into what it could be, instead of what it has been.

The emuscene has now been legitimized in the eyes of many. The only thing lacking is official confirmation of that inescapable fact. The year 2000 marks the beginning of the court phase of the legal battle over PSX emulation, which should settle the matter in the eyes of what Alexander Hamilton once called "the great unwashed" once and for all. *While I rarely gamble, I would almost be willing to bet money on its outcome - and I'm not known for being willing to gamble. Sony is going to lose, big-time, provided they aren't wise* *enough to settle first, and that will result in a clear-cut victory for the emuscene.* Everything that I know or have learned up until now points to that inevitable outcome. Those holdouts in the mainstream who continue to refuse acceptance of us before this will have no choice after that. The emuscene is here to stay, and it's high time that the recalcitrant ones "get a grip."

See you there.

REVIEW QUESTIONS

1. What is the significance of the year 1999 with regards to videogame emulation?

2. What was the very first major event of the year for the emuscene? The first lawsuit? The first major new emulator?

3. What one emulator more than any other provoked the emuscene's legal woes in 1999? Who coded it? What did it emulate? Why was it so controversial?

4. Describe in your own words Nintendo's position regarding emulation as the year began. How did its position change by year's end? What events brought about this change?

5. Describe in your own words Sony's position regarding emulation as the year began. How did its position change by year's end? What events brought about this change?

6. What original vendor is probably responsible more than any other for the eventual public acceptance of videogame emulation? How did it deal with the emuscene? What were the products it released that aided in resolving the debate?

7. What effect did the Columbine massacre have on the videogame industry in general and the emuscene in particular? How did the emuscene get dragged into the debate? How did the public react? How did the politicians react? What was the eventual outcome?

8. What was "the emuscene's darkest hour?"

9. Explain in your own words why May 1999 is considered to be "the worst month for the emuscene all year." What significant development for the emuscene had its roots during this time? What brought it about?

10. Which two promising emulators for the year were never released? Why? What do these failed releases have to do with the great emulation debate?

11. What bearing does the Rio case have on the legitimacy of videogame emulation?

12. Explain how a translation patch for an unlicensed videogame ROM dump played an important role in the great emulation debate. What was the

game involved? Who did the patch? What impact did it have on the videogaming public?

13. Which date "will forever be remembered by emufans" and "go down in emulation history?" Why?

14. How did the emuscene respond to the Scribe's *EmuFAQ?* What was his response? What was the significance of the *EmuFAQ* for both the emuscene and the general public?

15. Which three vendors declared limited support for the Internet emuscene by the end of the year? Of the three, which one had originally been reluctant to put the idea into practice? Why do you think they resisted the notion for so long?

16. What is the significance of the TG16/PCE emulator *Magic Engine* with regards to the future of videogame emulation?

THOUGHTS TO PONDER

1. Explain in your own words why the Scribe felt that Sony was going lose the legal fight over videogame emulation. How close did he come to the way events actually unfolded?

2. How has Nintendo contributed towards setting the legal bounds of videogame emulation? What is your opinion regarding their efforts?

3. Will the controversy over violence in videogames ever be completely resolved? Why or why not?

Module One: The Emulator

Part 1 - The Basis for Emulation

And so it begins ....

Ambassador Kosh, *Babylon 5*

INTRODUCTION

I've been actively involved with the emulation community in one form or another since the early days of 1998. I first became aware of the concept back in September of 1988 when I bought my beloved Amiga 2000 personal computer. You see, there was this nifty accessory you could get from Commodore called the PC Bridgeboard - a combination of hardware and software that would allow you to run IBM PC software on a suitably equipped Amiga. "Cool!" I thought, but I couldn't afford the price tag for either the A2088 XT or A2286 AT model at the time.

I was bemoaning this fact to fellow Amiga user Ken Gillum one day, and he merely shook his head. "You know," he said, "Commodore used to sell a piece of software for the old A1000 that would let you run IBM software on an Amiga without the hardware."

"No way!?" I said, hardly believing my ears.

"Way," he replied, "and I've got it. Come to my house this weekend and I'll give you a copy."

That was my first step into the wonderful world of emulation. The program that my friend gave me was called *Transformer*, and it did indeed let you run IBM software on an Amiga without special hardware.   True, it was buggy as all out, only supported text-based software, and had no support at all for hard drives, but who wanted to play "Beemer" games on an Amiga anyway? The business apps, such as *WordPerfect 4.2* and *Lotus 1-2-3*, were what mattered, and those were handled ably by that creaky old software emulator.

It was several months later, sometime in mid-1989, now having established myself as one of the Amiga gurus in my local area, that I paid a visit to a nearby Commodore dealer to get a copy of the demo for the brand-new and highly acclaimed Psygnosis game *Shadow of the Beast*. The late Dick Conrad, owner of the Commodore Connection and a close friend of ours, was quite excited. "You've got to check this out!" he said, steering me and my ol' Amiga hacking buddy Steve Smith, aka "Skerran," to his display A2000. "Have you heard of *A-Max*?"

"No," I said, feigning ignorance. Of course I had, but I didn't know of anyone who had gone to the trouble of buying it.

"Then watch this," he said as he turned on his A2000, activated *A-Max*, and then inserted a MacOS System 6 disk for bootup. Skerran almost fell over when he saw this, but I held my tongue and watched. A short time later, we were looking at a flickering blue-and-white MacOS screen displayed on the store's interlaced A1084S monitor.

"Now it's a Macintosh," the store owner stated proudly.

For those of you who might not remember, *A-Max* was a combination hardware and software product along the same lines as the PC Bridgeboard, except that it gave the Amiga the capability for near-perfect Macintosh emulation. All you needed was the *A-Max* emulation software, the *A-Max* adapter, a MacOS boot disk, and a set of Mac BIOS ROMs. Apple was at that time dead set against any type of cloning of their proprietary hardware, so the only way *A-Max* could remain legal was to force its buyers to go to a licensed Apple vendor and order a set of real Mac BIOS ROMs. These plugged into the adapter, and the *A-Max* software worked in conjunction with them to reconfigure an Amiga into a pretty decent Mac clone. Yes, it was a bit of a specialty product, it was pricey, and it really wasn't worth the trouble if you were stuck with an interlaced monitor - but oh, was it ever so neat!

Two months later I was in another state at a meeting of several software pirate groups to shop the latest warez offerings from the Amiga underground. I was on good terms with one of the better groups in the region, and one of its members (who shall remain anonymous) tossed me a white 3.5" disk. "Have you heard of *A-Max*?" he said.

"Yeah," I replied, "but it's no good without the Apple BIOS."

"Here's a cracked copy," he said, "and you don't even need that f--kin' adapter to run it."

He was right. After testing the disk with the latest release of Steve Tibbet's *VirusX* (we had to be careful even back then), I booted the thing up. Sure enough, there was *A-Max*, but running without the Mac BIOS! "How did they do it?" I asked.

"Easy," the hacker replied, "they just dumped the BIOS and hard-wired it into the emulator. Now it can run all by itself."

I thanked him for providing the copy, and a week later was back at the Commodore Connection. I pointed to Dick's dusty display A500 off to one side - he rarely used it anymore since his A2000s had arrived. "I can make *A-Max* run on that," I said.

"Not without the adapter, and I'm not taking it off the A2000," he replied.

"I don't need it," I answered. "Watch and see." And with that, out came the little white disk.

About five minutes later, a glassy-eyed and slack-jawed Dick Conrad was staring at a MacOS screen on his A500. "How in the world did you do that?" he asked.

"This version doesn't need the adapter," I answered.

"Where did you get it?" he asked.

"Same place your techs get their stuff," I shot back, winking as I did so.

"I see," he replied with a grin. "What do I need to copy it?"

"Nothing special," I answered. "*Project D* will work just fine."

"Here's a disk and my copy of *Project D*," he said, shoving them into my hands. "Go for it."

Emulation has come a long way since those initial heady days on the Amiga (and the Atari ST and Mac - let's not forget their contributions, either), but the basic premises and questions remain the same. The recent attempt by Nintendo to formulate an official company policy on emulation may have sparked a lot of ridicule, but was actually a good thing to happen at this time. As emulation is now suffering from something of a bad reputation in the public eye, perhaps it is time to take a look back at where we've been, where we're going, and the legal ramifications that are involved. The two stories above, which actually happened to me in my younger, wilder days, serve to illustrate all of the main questions concerning the so-called "legitimacy of emulation" and the various questions that it poses.

I shall attempt in the ensuing series of discussions to try and lay to rest some of the myths and half-truths that shroud the emulation community within a grey legal haze. Rather than just tossing out highly emotional statements as most users prefer to do (and of which I am guilty at times), I will be citing specific examples from the history of the emulation community as well as related issues within the computer industry as a whole. I will also cite specific provisions of various laws, statutes, and regulations where applicable; and I will also refer to a wide range of court cases (i.e. "case law") that have a direct impact on the emulation issue.

THE PURPOSE OF EMULATION

The first question one might ask is the most obvious one: "Are emulators legal?" It depends on how you ask the question - are you dealing with the concept of emulation as a whole, or are you referring to the creation of an emulator? In order to resolve this dilemma, let's take a look at the purpose of an emulator. We will save the legal question concerning the creation of an emulator for later.

What is an emulator? *An emulator is a product designed to imitate one system with another so that both accept the same data, execute the same programs, and achieve the same results* (American Heritage Dictionary, Second College Edition, s.v. "emulate"). Emulators can be pure hardware, pure software, or a combination of both. In the two stories that I relate from my past, the Amiga *Transformer* program would be considered a pure software or "true" emulator, while both the PC Bridgeboard and the unadulterated *A-Max* would be considered "combo" emulators because of the combined hardware and software approach in their design. An example of a pure hardware or "firmware" emulator would be the Sega Genesis/MegaDrive's VDP mode 4, which converts the console into a functional Sega Master System emulator running entirely on its internal hardware, and all of which is activated via a special code received from the console's cartridge port.

For those of you who are still new to the emulation scene and don't have any idea what I'm talking about, let me describe emulation another way. Many of you are using IBM PC compatible computers running *Microsoft Windows* as its primary operating system. Take a moment to look inside the Accessories folder. You'll see about a dozen or so programs and directory entries, but the one I want you to look at is named Calculator. Run it, and see what happens. You should now have what looks like a little calculator on your screen - not all that different from the little credit card sized ones they sell on the J-hooks at your local grocery store. If you've got one handy, go get it and set it beside your keyboard. Now, I want you to work a few math problems on Calculator; if you've got the real thing handy, then double-check your results with it. Neat, eh? That's because Calculator "emulates" a real calculator - it allows you to conveniently work math problems without having to have a real calculator handy, and it serves just as well as the real thing anytime you're sitting at your computer. Since Calculator is a computer program that does not require any kind of extra hardware (other than the computer itself), it would be considered a pure software-based, i.e. "true" emulator.

This brings us to the underlying idea behind an emulator. It is something designed to perform a task or series of tasks that you couldn't otherwise do because they weren't intended for your hardware. That's the core concept. You're not supposed to be able to play Sega Master System games on a Sega Genesis, but you can with the use of the PowerBase Converter - a unique cartridge port accessory that includes a special boot ROM which reconfigures the console so it can support the older game cartridges. You're not supposed to be able to run *Windows* software on a Macintosh, but *SoftWindows* makes it possible. Speaking of my beloved old Amiga 2000, you're not supposed to be able to run Amiga software on a high-end IBM PC compatible, but Cloanto's *Amiga Forever* (the commercially licensed version of *UAE*) makes that possible. The concept is sound and has been employed by both vendors and users in various ways over the years. Emulation is a valid use for computer hardware regardless of who you are.

THE REASONS FOR EMULATION

So why even emulate at all? Four reasons. First, **convenience**. Remember the example of Calculator? Beats digging in the ol' office desk for the one that your secretary borrowed last week and never returned. Second, **back-compatablity**. Assuming you have a need to keep using an older piece of software, it sure is nice to have some kind of setup that let you use the old stuff with newer hardware - that way you can junk the old hardware and free up some space. Third, **software development**. It's nice being able to develop for a custom system using one that you already know how to use. Fourth, the old standby of **cross-platform support**. It's a lot cheaper to buy just one multipurpose computer system and different kinds of software to use with it than buy different systems for different purposes and the software to match. Emulation allows you to save your money by running software on the system you have that may have been intended for use with a system that you either do not have or cannot afford.

Need some examples of emulation in action? Well, we've already got Calculator from *Microsoft Windows* as an example of convenience. There are plenty of examples for back-compatability, but I'll limit myself to three: the PowerBase Converter for the Sega Genesis/MegaDrive; the "real 8086" mode of all Intel and Intel-derived 80286 processors and their descendants; and the famous "triple mode" operating system feature of the venerable Commodore 128. As for software development, the best example I know is Sega's TeraDrive, which was a specially modified 386SX PC-AT compatible computer with on-board Sega videogame hardware, produced and distributed almost exclusively to game developers and the like. That also serves as a good example of cross-platform support, since it allowed the use of both IBM PC and Sega Genesis/MegaDrive software within the same box.

With the various back-compatability and SDK issues whizzing about in the minds of computer programmers, it was inevitable that some bright young programmer would say to himself, "Screw the US$1000 or so for the SDK - I'm going to write my own." On the other side of the fence, you have a passionate home console owner with a computer background and no spares to be found in local stores to replace his worn-out game carts due to the obsolescence of the console. "Forget that," he says, "I'll just write an emulator and dump the cart ROMs instead." Bounce over to the frustrated personal computer user who has just learned that the game he bought will not work on his new system because it was designed to run on an older, now-obsolete model that is no longer in production. "No way," he says, "If the store won't give me my money back, then I'll find some way to run it anyway." Along those lines, dare we forget the retrotechs - who just flat out love old computer stuff and will do almost anything to keep it alive? "We're tired of hunting for out-of-production parts," they say. "This neato program will let me run this software on my newer systems." Or what about the big corporate vendor, who needs to find a way to make its old software work with its soon-to-be-released new system in order not to lose a good-sized chunk of its customer base? "What old hardware?" they say. "We'll just make an emulator for our new system." Many faces, but the same set of dice. Thus was the modern emulation community born.

THE BIRTH OF EMULATION

The concept of emulation was invented by systems engineer Larry Moss of IBM back in 1964 as part of the development effort behind the **IBM System/360** family of mainframe computers. They were quite revolutionary for their day, being the first mainframes to use integrated circuits all the way throughout their design, which meant that programs hard-coded for older, transistorized IBM computers would not work with the new product line. Three approaches were tried by IBM - first, a series of computer simulations running entirely in software (sound familiar?), then next came the Moss proposal, and finally a straightforward firmware approach that resulted in the IBM 1410S (that sounds familiar, too). Neither of the two extremes would work with the System/360 product line, though - the "software simulations" were just too slow and the firmware approach used in the IBM 1410S was way too machine specific. There had to be a comfortable middle ground in there somewhere, and that is where Larry Moss stepped into the picture.

Moss used the word *emulator* to set his proposed project apart from the pure software and firmware approaches by others. His design "strove to be like" the real McCoy without actually being it (per the dictionary definition of the word), running the same programs at a comparable speed but on a completely different system. The actual emulator was a mix of hardware and software components, which worked in conjunction with each other to allow higher-end System/360 computers to behave as if they were an IBM 7070 mainframe, one of the top computers of its day. Moss believed then, as the passing years have shown, that *the closer the hardware in the target system was to the system being emulated, then the better the emulator would function.* This is why he chose the "combo" approach, as it offered the best compromise between portability and cost. It had only those hardware components that were really required for effective emulation at a decent speed, with the remaining issues being handled by the emulation software. Stuart Tucker, who was in charge of System/360 development issues, liked the idea and gave Moss the go-ahead. The rest, as they say, was history. The System/360 went on to become IBM's best-selling product line of that era, and the *7070 Emulator* by Larry Moss was so successful that many customers were running their old software under emulation well into the early 1970s without a hitch. Thus it was that the technology known as emulation was born, as well as the three basic approaches to designing an emulator, and we owe it all to Larry Moss and his coworkers at IBM. *For his achievement, Larry Moss is today considered to be the father of computer system emulation.*

SCALING THE TECHNOLOGY DOWN

Credit for the birth of personal computer emulation is given to three of the true pioneers in the personal computer industry - Tim Patterson, Don Burdis, and Bill Gates. In 1980, Microsoft released the **Z80 SoftCard**, its first hardware offering and its first such accessory for the **Apple II** personal computer. This US$350 add-on was comprised of a plug-in card and accompanying software that would allow Apple II users to run **CP/M** software on their machines. Tim Patterson designed the actual hardware side of the emulator, with Don Burdis managing the product (and the requisite software) in its final stages and Bill Gates making sure that everything stayed on track. Remember, these were the days before the arrival of the IBM PC and MS-DOS. You couldn't do everything with software alone, because the available technology was quite limited. CP/M, invented by Gary Kildall, was the dominant operating system for business computers prior to IBM's arrival on the scene, but it was incompatible with the Apple II's MC6502 CPU. Microsoft's SoftCard had an on-board Zilog Z80 CPU, which was one of the CPUs that CP/M was designed to work with. *The Microsoft Z80 SoftCard was the very first computer system emulator for a personal computer*, allowing the Apple II to greatly expand its software base and contributing a great deal towards the success of that classic computer. CP/M itself would remain a viable operating system well into the mid-1980s, and products similar to the SoftCard subsequently appeared for other personal computer systems of the decade. The presence of these various CP/M emulators meant that users didn't have to ditch their old CP/M software while learning how to use their new systems, and personal computer users grew to appreciate the concept just as much as mainframe users had before them.

In 1982, computer chipmaker Intel announced the release of the **80286 CPU**. This was one of the company's first true 16-bit processors intended for mass production and represented a quantum leap forward in personal computer processing capability. All that was well and good, except that Intel had a problem - almost all of the software of the day was 8-bit in origin. Fortunately, these were the early heady days for IBM, as its IBM 5150 Personal Computer (aka the **IBM PC**) had taken the business computer market by storm and thus set the standard for all machines that would follow in its wake. The IBM PC used the 16-bit 8086 as its CPU, but its system bus and software base were 8-bit in nature. Compaq, who was in the process of winning the legal battle over the right to use the Phoenix reverse-engineered BIOS in its PC clones, favored the compatible 8-bit 8088, which Intel had designed as a cheap alternative to the 8086. With this in mind, Intel devised a unique solution to ensure that existing 8086/8088 software would work with their new CPU. The Intel 80286 was designed to operate in either one of two modes: standard mode, using a 16-bit data path and 16-bit memory addressing; and real mode, using an 8-bit data path and 8-bit memory addressing. *In doing this, Intel had unintentionally created the first firmware emulator for a personal computer*, since the 80286's real mode perfectly emulated that of an actual 8086 CPU. It should be noted that 8086 "real mode" remains to this day part of every Intel CPU released for the personal computer market.

Videogame emulation had its beginnings around this time, during the heady days of the **Atari Video Computer System** (VCS, later redubbed the Atari 2600), as manufacturers rushed to tap the new and rapidly expanding home videogame market. The Atari VCS was the undisputed top dog until 1982, when rival manufacturer Coleco released its **ColecoVision** console. It was a superior product, in fact a somewhat retooled Japanese **MSX** personal computer in disguise, and its more powerful capabilities provided Coleco the means by which they could tap into the existing market for Atari VCS games. Coleco offered a special adapter for the ColecoVision, the Expansion Module #1, that allowed their console to use videogame cartridges designed to work with the Atari VCS. Atari promptly sued for patent infringement, but lost in the courts when it was ruled that the product did not violate any Atari patents. Coleco later "thanked" Atari for their generosity in 1983 by releasing the Gemini, a 100% clone of the Atari VCS. *The ColecoVision Expansion Module #1 is nowadays considered by most to be the first videogame system emulator, but the first "true" videogame emulator would not appear until some eight years later (Yuji Naka's unreleased NES emulator for the Sega MegaDrive).* It should be noted that so-called "cart adapters" were made by many different manufacturers for various systems, each supporting different cartridge formats, and continued to be manufactured right up until the days of the N64 in the late 1990s, which was the very last videogame console to use the cartridge format.

The threshold of so-called "true" emulation with regards to today's systems was crossed in 1985 thanks largely to the arrival of one particular personal computer - one that would forever change the industry in many ways. We're still dealing with Commodore, but with an entirely different system - the legendary **Amiga** personal computer, generally regarded as the most powerful personal computer of its day, the world's first multimedia computer, and the first computer system to openly support emulation technology. The Amiga was so powerful that Commodore really didn't know what to do with it, but at least they had the good sense to realize that it needed to be IBM compatible in order to stay alive in a market dominated by PC clones. The Amiga design team had already foreseen this need, and had a working prototype of a combo IBM PC emulator on hand to demonstrate to Commodore executives. The literature of the day describes two parts, with the software side called the Transformer and the hardware side called the TrumpCard. It was well received, but Commodore insisted on a number of changes (such as ISA card support for the TrumpCard) that delayed the emulator's arrival on the market by a full year. In the meantime, this gave rival computer maker Atari the chance to hurry development of its own prototype IBM PC emulator, which first demonstrated in a "smoke-and-mirrors" demo at the rollout for the **Atari ST** in mid-1985. Their IBM PC emulator development was accelerated and the product was rushed to market as soon as possible, finally making it to store shelves in early 1986. *Thus it was that **PCDitto** by Avant-Garde Software for the Atari ST can lay claim to the title of being the first IBM PC emulator released to the general public.* It was a pure software product (i.e. a "true" emulator) and notoriously slow, but the fact that it was available at all spurred Commodore's own efforts in this regard. The software for the Amiga prototype emulator was quickly recompiled to remove the need for hardware support and released shortly thereafter by Insignia Solutions as the ***A1025 Transformer***, Commodore's own "true" IBM PC emulator program. It was not much better than *PCDitto*, but at least Amiga users would have an IBM PC emulator to tide them over for a few months, and with that the original Transformer/TrumpCard concept was abandoned for good. The rest of the year was spent with the retooled TrumpCard concept, and it was finished in time for the 1986 winter trade shows under the name of the **PC Sidecar**. It was universally reviled at the time for various and sundry reasons, but would eventually serve as the basis for IBM hardware support in the next generation desktop Amiga that Commodore then had under development. That machine that was eventually released in 1987 as the Amiga 2000, and it would prove to be the workhorse of the Amiga product line - emulation support and all.

Atari may have been first to market and can still claim credit for theirs being the first "true" IBM PC emulator, but it is Commodore and the Amiga design team that must be given credit for having the first <u>working</u> IBM PC emulator. In fact, so many emulators for various systems, ranging from the well-known to the obscure, were eventually developed for the Amiga that it earned the unofficial nickname of "the chameleon machine." No other machine of its day had the capabilities for emulation that the Amiga had, and only recent advances in personal computer technology have brought these abilities to other platforms. The Amiga's reputation for emulation is one that no other personal computer system has managed to supersede as of this date.

CROSSING THE PROPRIETARY LINE

The concept of emulation was turned on its head in 1988 by a young man named Simon Douglas. He was the driving force behind the development of *A-Max*, a Macintosh emulator for the Amiga and the first Mac emulator for that platform. Readysoft demonstrated their Mac emulator at the World of Commodore show in November of that year, and it immediately drew the attention of everybody at the show. It received rave reviews and a number of write-ups in the major personal computing magazines of the day. Like the PC Bridgeboards by Commodore that were introduced for the Amiga 2000 at the same show, *A-Max* was a combo emulator, incorporating both hardware (an external adapter) and software components (the *A-Max* program) in order to provide the widest possible range of performance and capability. Its external hardware adapter (and its later internal carded incarnation) required an actual piece of a Macintosh computer in order for *A-Max* to work, and that "missing part" was a genuine Macintosh BIOS 128K chipset (two chips, as I recall). There was a **big** difference between *A-Max* and the other Mac emulator that preceded it, however. *A-Max represented the first time that a fully functional emulator had been commercially released for a given vendor's proprietary system.* There had been an earlier product for the Atari ST (*Aladdin*), but its capabilities were rather limited and it was quickly relegated to well-deserved obscurity. *A-Max* transformed a suitably equipped Amiga into a true Mac clone in every sense of the word, and Apple wasted no time in quickly voicing their objections to the product.

There was no question that in-house emulation of a vendor's own products was perfectly acceptable. There was no debate that neither licensed nor developmental emulation of proprietary platforms was legal, and a number of such products had already appeared. There had never been a problem with the older CP/M emulators, due to their widespread acceptance and the lenient usage terms by which CP/M had been made available. There was no problem with emulating the IBM PC ever since IBM had lost the court battle over the Phoenix BIOS. But a commercially vended emulator for the Mac?! Everybody knew how jealous Apple was of its little black-and-white computer, going so far as to use every trick in the book (and then some) to ensure that the Mac would not be cloned. That is what caused all the buzz in the personal computer industry at the time, and many predicted that Readysoft was going to be in for a long and nasty lawsuit at the hands of the proprietary-minded Apple Computer Corporation. Does this sound familiar, N64 and PSX emulator users? It was the same thing that you are going through now, except that it happened a decade ago with an unauthorized emulator for a personal computer system.

What happened next is a matter that remains shrouded in mystery. Apple continued to strenuously object to *A-Max*, decreeing that it was infringing use of proprietary Macintosh technology. Many Amiga users of the day, myself included, remember the widespread fears and discussions about a possible Apple lawsuit against Readysoft. Strangely enough, though, object was just about all Apple ever did. Nothing happened, and that was the biggest surprise of all. Well, almost nothing. The only public move Apple made over *A-Max* was to try and dry up all sources of loose Mac BIOS chipsets, but it did little good. Apple distributors and repair centers with dead or disabled Macs quickly pulled the BIOS out of those systems and were quite happy to sell them to cash-toting *A-Max* users. Eventually, Apple gave up on this tactic, and by the following year *A-Max* had become an established part of the Amiga and Mac communities, who now had something in common - even if it was only an emulator. Not only did *A-Max* remain a viable product on the Amiga market for several years afterward, it also became one of the most touted emulators in the Amiga arsenal, spawning its own line of copycat products (*Shapeshifter, Emplant, Fusion*). Though it was eventually superseded by its competitors, *A-Max* was one of the products that helped build the Amiga legacy. Macintosh emulation is quite commonplace nowadays, and it was *A-Max* that blazed the trail for later efforts in this regard.

So why didn't Apple sue Readysoft?

Because they couldn't. There was no way for them to win, and they knew it.

Wow! You mean ... even back then ... and they couldn't? How did that come about?

The following explanation is part reason and part conjecture, based on what facts are known and on careful research of the legal issues involved. Apple has refused the opportunity to comment on the affair even at this late date, so perhaps this may be the only answer that we will ever have. While it is incomplete and given to surmise, it is a reasonable explanation given the data at hand.

The legal department at Apple knew full well that unauthorized duplication of copyrighted computer code, especially a computer's operating system, both in BIOS and on software, is illegal; in fact, they had helped establish the precedent some five years earlier over the Franklin ACE 1000, an illegally produced Apple II clone computer (Apple v. Franklin, 1983). Simon Douglas and his team were also aware of this, as well as the dispute over the reverse-engineering of the IBM PC BIOS (IBM v. Compaq, 1982), so they had deliberately designed *A-Max* in such a fashion that it would not infringe upon Apple's intellectual property rights. The Amiga and Mac used similar processing hardware and similar components in a similar fashion, and the only real difference between the two computers was their respective operating systems and communications ports. This was the reason why the combo approach was chosen for *A-Max*. Its hardware adapter contained the minimum amount of hardware required for full Mac computability, and that amounted to no more than the custom Appletalk port and Mac external floppy connector, as well as the appropriate sockets for a genuine Mac BIOS chipset. The emulation software handled the rest, reconfiguring the Amiga internals for proper Mac operation. The fact that almost identical hardware was involved explained why *A-Max* was such a solid product, and also reinforced the original supposition by Larry Moss that similar systems would make for a faster emulator.

In retrospect, reconfiguring the Amiga floppy disk drives to read native Mac floppy disks was probably one of the most brilliant pieces of work in the original version of *A-Max.* It gave Amiga users who did not have access to external Mac floppy drives the ability to use Mac software without having to convert the disk to another format; thus, you could use a real MacOS system disk to boot up the emulator. Not only was this extremely convenient, but it was also legally prudent, as we will later learn. *A-Max* could handle original Mac software in its native format; therefore, software-related issues in any possible dispute were neatly sidestepped.

The requirement of a genuine Mac BIOS chipset was another gutsy yet well-reasoned move, since Simon Douglas and his team did not have either the resources or desire to reverse-engineer it. Forcing the use of the actual chipset neatly killed two birds with one stone. It saved them from the long and arduous reverse engineering process and the inevitable lawsuit that would have resulted, and it provided for some small compensation back to Apple in the form of profits from Mac BIOS sales. It was not as much as Apple would have gotten from selling an entire Mac, but it was money in their pocket nonetheless; thus, it could be said that Apple was making money every time someone bought a copy of *A-Max*. In addition to the initial purchase of *A-Max* by a potential user, the buyer would also have to cough up extra money to Apple for the proprietary Mac BIOS that was required to make *A-Max* work. It was a brilliant idea and one that was subsequently borrowed in later products, although the concept of requiring an outside BIOS would be perverted sooner than anyone would realize (remember the *A-Max* bootleg?).

As an aside, it should be noted that Apple's other legal dispute at the time may have played a part in their eventual muted response to the rise of Macintosh emulation. Remember, the *A-Max* affair arose about the same time that Apple was considering suing Microsoft over the release of *Microsoft Windows*, which it felt was an unlawful infringement of MacOS on multiple counts. Put yourself in Apple's shoes for a moment. You have before you two different potential lawsuits against a possibly infringing vendor - one which you may not win based on past case law precedent (*A-Max*), and one which you may very well win for the same reason (*Microsoft Windows*). Which are you going to pursue?

To sum it all up, *there would have been no way that Apple could have proven willful intellectual property infringement on the part of either Readysoft or Simon Douglas and his team*. They had been very careful in the way they had designed *A-Max*, and their foresight proved itself in the long run. The question of Macintosh emulation remained the center of much debate during the better part of 1989, even after *A-Max* finally made it to market that summer, and everybody within the Amiga and Mac communities held their collective breath over the affair. When the year finally ended, *A-Max* had emerged unscathed, and all of us breathed a collective sigh of relief. Apple had decided not to challenge the legality of Macintosh emulation, and in doing so unknowingly set a legal precedent for emulation technology that remains with us to this day. I call it the

***A-Max* precedent**, named after the emulator at the center of the dispute, and it is as follows:

The *A-Max* precedent
It is lawful for a third party to develop an unauthorized emulator for a proprietary system, provided that said emulator cannot be shown to violate the intellectual property rights of the original system vendor in any way.

And with that, the word began to spread - excitedly discussed among the Amiga user community, whispered around the PC clones in many an office, and commented upon in front of the Macs at the local schools. **Emulation of proprietary systems was now permissable**. There was no more debating the matter. And as we Amiga users of that day saw a steady stream of emulators begin to appear for various systems and wondered to ourselves what system would be emulated next, and as an indignant Apple flailed away at Microsoft in the courts over the release of *Microsoft Windows*, we knew something special had happened. Though we could not put a label on it at the time, the resolution of the *A-Max* affair had kicked off what is now called the **Golden Age of Emulation** (1989 - 1998). The technology known as emulation was no longer bound by proprietary desire, and we owe it all to Simon Douglas and the Commodore Amiga.

<u>INTROSPECTION</u>

So now we're back where we started. "Are emulators legal?" The answer is a resounding "YES!" The concept of emulation is perfectly legal regardless of the nature of the system involved and has been in use for many years now. So much for that. The next question that comes to mind is how one goes about creating a legal emulator. Before we can do that, we first need to try and come to an understanding of the basic concepts behind intellectual property protection and the various forms that it takes. It is best to know how the legal system may affect your intended emulation project, and it is that forum that we shall discuss next.

REVIEW QUESTIONS

1. What is an emulator?

2. What are three different kinds of emulator?

3. What are four reasons for creating an emulator, and what are some examples of each in action?

4. Why would one might want to develop or obtain an emulator?

5. What desire of the computer industry gave rise to the concept of emulation?

6. Who is considered to be the father of emulation technology? Can you tell in brief the story behind the very first emulator?

7. How did emulation get started on personal computers? Who were the people behind it, and what was their product? What did it emulate, and why?

8. How did the development of the Intel 80286 CPU contribute to the rise of emulation?

9. What are the origins of videogame emulation? Why was that product not a real emulator?

10. What legal battle can be said to have "primed the pump" for videogame emulation? How was this important? Why was such a thing possible with a mere videogame console?

11. Why is the Commodore Amiga remembered as "the chameleon machine?"

12. Who was responsible for the first "true" IBM PC emulator? What was the product? How did it come about?

13. What product changed the application of emulation technology forever? Why was it so significant? How did it reinforce one of the guiding principles of emulation?

14. What is one explanation as to why Apple failed to file a lawsuit against this "significant product?" Be sure to give specifics that back up the main points of your answer.

15. What significant precedent did this "significant product" set? What does that precedent say? What does it mean?

QUESTIONS TO PONDER

1. Can you describe some of the emulators that arose from the legitimization of proprietary system emulation? Who made them? What systems do they emulate? Are each of these considered legitimate products to the best of your knowledge? Why

2. How did IBM's failure to stop the PC clone vendors contribute to the eventual rise of the emulation community?

3. How did videogame emulation actually get started?

4. Is emulation legal, at least from a vendor's perspective?

5. What are your thoughts as to the legality of a noncommercial (i.e. unlicensed or freeware) emulator?

Module One: The Emulator
Part 2 - Developing an Emulator

When crime is finally outlawed, then only criminals will live in America.

From an anonymous Internet posting

UNLEASHED

The time is the wee hours of the morning. The setting is inside the confines of a small, darkened room. Crumpled printouts and soft drink cans are overflowing out of a small wastepaper basket, while a medium-sized contingent of insects and various rodents are feasting upon the remains inside an ever-growing pile of take-out food containers in a far corner. The human occupant of the room appears unconcerned; in fact, he has been working for days, and has taken to humoring himself by making sure that some morsels are left behind for his small guests. The erstwhile hacker takes the last swig from his drink, crushes the can, and then throws it at the basket. It bounces off of the top and into a nearby pile of junked computer hardware, but by this time he has turned back to his console and is concentrating intently on the computer code on which he is working. Beside him are two monitors - one showing various bits of screen garbage and the other showing a videogame display. The latter is connected to a strange-looking device sitting on a nearby table. In another life it might have been an older model computer, videogame console, or some other piece of computer hardware. Today it has been stripped to the bare boards, data lines, and power leads, and has enough logic probes and other such monitoring devices attached to make it look like some weird Kafkaesque mutation of its former self.

Suddenly, there is flickering on the monitor screen that was displaying the garbage but a few moments ago. The hacker looks up with furrowed brow, preparing to swear profusely. The screen goes black for a few seconds, and then a display pops up. It is an almost perfect match for the one next to it - the one coming from the Rube Goldberg box on the table. The young man is overjoyed - he kicks back in his chair away from his console and into another pile of parts behind him. The resultant crash startles his little guests and they scamper off, but he is no longer paying attention to them; the young man has become excited by the almost-matching displays. He grins, begins laughing, and then raises his clenched fist into the air, exclaiming, "That's it! I did it!"

Sound familiar? This scene, or something vaguely similar to it but probably less dramatic, is familiar to anyone who has worked on a computer project for an extended period of time and has been played out by many a emulator developer around the world. It doesn't matter if the place is a converted garage and the young man is an American teenager, or if the place is an apartment inside a Tokyo high rise and the young Japanese man is shouting "Yatta! Yatta!" over and over again to the irritation of his neighbors, or if the place is a college dorm somewhere in Mannheim and the awed German college student at the console can do nothing but whisper "Mein Gott!" in reverent awe. It really doesn't matter either if the unusual device on the table used to be an Amiga computer, or a Sega Master System, or even an Hewlett-Packard scientific calculator in a former life. Another triumphant milestone in the emulation community has been achieved. Somewhere out there in the murky shadows and backwaters of the computer underground, another privately developed emulator has been born.

We've already seen that emulation is a legitimate use for a computer. We have established that emulation in one form or another has been with us for many years. That now leads us to the question that all developers who are not affiliated with original product vendors must ask themselves: "Is it legal for me to develop an emulator?"

THE CONCEPT OF INTELLECTUAL PROPERTY

"The introduction of videogame emulators represents the greatest threat to date to the intellectual property rights of videogame developers." So says Nintendo concerning emulation in their industry, but the same can be said for other computer industries as well; the operating system industry (*Microsoft Windows* and *SoftWindows*), the computer hardware industry (Gateway 2000 and *WinUAE*), and so on. Nintendo should know - they are one of the biggest vendors in the videogame industry and have a widespread reputation for having a hair-trigger legal attitude in this regard. Other companies within this and other related industries have also expressed concern over the growing popularity of emulation and taken legal action on their own, and the process will no doubt continue. Why? Do they just like picking on people who like "old stuff" or "cool cross-platform tools," or is there something else involved? There are a number of concerns, but almost all of them can be grouped together under the legal term that Nintendo employs in its emulation policy statement - **intellectual property rights**.

So *what is intellectual property*? At the risk of oversimplifying the legal definition, let's reduce the concept to its fundamentals. *Intellectual property is one or more unique creations that belong to you and nobody else.* They can be anything, so long as you can demonstrate that said items are unique. These items embody an particular set of ideas or productions to which you own all legal rights. They do not necessarily have to be created by you; they may have been produced by someone in your employ or under contract for you, known in legal circles as a "work for hire," or you may have bought the rights to them from another party, which is commonly referred to as a "technology transfer." Regardless of how you obtained the ownership of these items, your intellectual property rights imply that these creations are yours and yours alone to do with as you please, and nobody else has the right to do anything with them without your express consent. Governments around the world recognize the concept of intellectual property rights, and it is usually embodied in one of three forms: the **patent**, the **copyright**, and the **trademark**.

WHAT ARE PATENTS?

Patents are awarded to creations of the hand. Again, at the risk of oversimplification, *patents are used to provide legal protection to any type of device that can be created or any process of creation in and of itself*. Just take a moment to look around you. Many of the physical objects you see are covered by some kind of patent, or may have been produced using some kind of patented process. Not everything is covered, though, because *patents do not cover generic concepts or manufacturing methods*. For example, just about all of the parts inside your computer are patented and many of the components on the individual boards also have patents, whereas the chair upon which you are seated is probably not patented. *There is no such thing as an "implied" patent*; you must apply for a patent before you can enjoy the rights and benefits of patent protection. *There is no such thing as "fair use" of patented material*, as this runs counter to the exclusive nature of patent protection. It should also be noted that governments tend to be quite stringent when it comes to awarding patents; you must prove that your device or process is either a unique creation or a creation that uses existing resources in a new and unique way.

In the computer industry, *patents used to be limited to hardware, but recent changes in federal law allow patents to cover software as well*. These are awarded under certain limited conditions; for example, a patent can be used to protect a piece of software if it contains a unique process like the mathematical algorithms used in compression software, or aids in the working of a unique process within the hardware itself. The main reason behind this change is that patent protection gives software developers, owners, and vendors more legal leverage than does copyright protection - copyrights only protect the expression of the concept (the code as a whole) and not the many processes it might enable.

In the United States, patents are governed under the terms of the U.S. Patent Act (35 USC). The United States is also a signatory to the internationally recognized Patent Cooperation Treaty, which means that it honors the patents of other countries and they honor U.S. patents in return. United States law was recently changed with regards to the lifetime of a patent; as a result, patents now last twenty years from the date of original application (not the date the patent was granted, as was previously the case). Extensions are sometimes granted under certain circumstances, but the maximum length of extension cannot by law exceed five years from the end of the original patent. Patents are further governed in the United States under the terms of the Intellectual Property Rights Act (37 CFR). For more information, you can drop by the official web site of the U.S. Patent and Trademark Office (http://www.uspto.gov/).

The University of California at Berkeley has a course on intellectual property rights which I highly recommend to anybody who might be attending that institution. I have borrowed from their course outline to present the major points with regards to patent protection and the computer industry:

What is patentable?

1. *Machines* (any system or other apparatus that can be programmed).

2. *Manufactured devices* (internal hardware and external accessories used in conjunction with a system).
3. *Processes* (a series of operational steps to be performed on or with the aid of a machine).

What is not patentable?

1. *"A compilation or arrangement of data independent of any physical element."* (something stored inside a system, such as a database or other similar compendium of data)
2. *"A known machine-readable storage medium encoded with data representing creative or artistic expression."* (these are considered literary creations, and program source code is the prime example)
3. *"A data structure independent of any physical element."* (data tables, linked lists, etc.)
4. *"A process that does nothing more than manipulate abstract ideas or concepts; e.g. a process consisting solely of the steps one would follow in solving a mathematical problem."* (generic mathematical equations, as opposed to a mathematical process created to serve a unique purpose)

What are the advantages of patent protection?

1. *Coverage is quite broad*, protecting against almost all forms of independent development.
2. Patents *prohibits other individuals or organizations from laying claim to the same invention*.
3. *Deterrent factor* of patents is their chief value - it's harder to fight or circumvent a patent than any other form of intellectual property protection.

What are the disadvantages of patent protection?

1. Application and processing *costs are high*.
2. *Duration* of patent protection *is relatively short*.
3. *Totality of invention is disclosed once application is made* (this is a requirement under current patent law).

WHAT ARE COPYRIGHTS?

Copyrights are awarded to creations of the mind. These include but are not limited to such forms of expression as printed matter, music, and all forms of audiovisual production. Examples include books, poems, essays, epics, songs, works of art, musicals, movies, television shows, plays, animation, and so on. With regards to the computer industry and at the risk of oversimplifying things again, *copyright protection is usually limited to any and all types of computer code and digital audiovisual works: programs, programming languages, program source code, specialized databases, expert systems, unique sound and/or* *graphic creations, specialized computer software (such as but not limited to games, applications, and operating systems), and any digital reproduction of the old-style media mentioned earlier*.

There are two forms of copyright. **Implied copyrights** cost nothing and are assumed at the moment of creation, but are difficult to prove and protect. On the other hand, **statutory copyrights** are granted by a government copyright office (usually for a small fee) and ensure full legal protection for said work under applicable laws. In the United States, copyrights are awarded and protected under the terms of the U.S. Copyright Act (17 USC), which itself is in sync with the internationally recognized Berne Convention for the Protection of Literary and Artistic Works (828 UNTS 221). *It is important to note that the provisos of Berne do not require the inclusion of a copyright notice within a work to qualify for copyright protection*, as was previously the case under the old 1976 U.S. Copyright Act.

As a general rule, *copyrights for all commercial works, which are regarded as "works for hire" by copyright law, that were created on or after 1 January 1978 last for the lifetime of the copyright holder plus an additional seventy-five years*. As an aside, *this includes all personal computer programs and practically all videogame releases*, and only excludes a mere handful of of pre-1978 concepts for mainframes and early arcade videogames that are still in use today. Extensions are no longer granted for copyrighted works except for items that were copyrighted before 1978, but these can only be extended for a maximum of seventy-five years before they expire. Some of the exceptions that I named are protected by copyright, but not all may have been extended (you need to check with the original vendor or copyright owner for more details). Just like patents, copyrights are further governed in the United States under the Intellectual Property Rights Act (37 CFR). For more information, you might want to drop by the official web site of the U.S. Copyright Office (http://lcweb.loc.gov/copyright).

Again, as with the patent, I have borrowed from the Berkeley course outline and adopted their format to note the distinctions with regards to copyright protection and the computer industry:

What is copyrightable?

1. *Videogames, computer software, sound recordings, works of art (graphics), and other audiovisual forms of expression.*
2. *Any old-style media that can be digitally reproduced* (in such cases, the original copyright protections apply to the digital reproduction as well).

What is not copyrightable?

1. *Everything that is a work of authorship*, regardless of whether or not the author is known, *is copyrightable* (this tends to cover the ground that patent protection does not).
2. Copyrights *protect the means and manner of expression, not the idea or thought itself* (this is the fundamental difference between copyrights and patents).

What are the advantages of copyright protection?

1. *Easy and quick to obtain*, as opposed to a patent.
2. *No substantive examination required before granting*. In contrast, the patent process is quite rigorous and requires exhaustive proof of uniqueness.
3. *Minimal cost* to obtain and maintain.
4. *Long-lived* - about three times longer than patent protection.

What are the disadvantages of copyright protection?

1. *Independent development is permitted*, thus allowing others to create and release similar products.
2. *Reverse engineering is permitted*, thus allowing others to learn the ideas and concepts that underlie your work.
3. *Burden of proof to show infringement lies with the copyright holder*.

WHAT ARE TRADEMARKS?

Trademarks are awarded to creations of the marketplace. These include but are not limited to names, terms, corporate "mascots," and advertising gimmicks such as corporate logos and other such graphics designs. *The core idea behind a trademark is that it creates a unique image or association within the mind of a consumer that is linked to a particular company or company product.*

A trademark has the longest life of any form of intellectual property protection. They are initially granted for a term of ten years, but they become incontestable after the fifth continuous year of use. Some countries require periodic proof-of-use or maintenance fees, but the general rule-of-thumb is that *trademarks last forever* unless they fall into disuse. It is up to the owner to show proof-of-use and continue using it in a proper manner, otherwise it passes back into the public domain. In such cases, the trademark is said to have been abandoned, and anybody can then use it or register it themselves. Even so, *limited forms of public domain use of a registered trademark are acceptable (i.e. no license from the owner) so long as the owner deems the use to be noninfringing* - that is, it doesn't harm the image that the trademark is supposed to represent. In these cases, the use must somehow acknowledge its owner, and there are specific ways of doing so that are laid out under law.

Patents and trademarks usually go hand-in-hand - you patent a product or process, then create a trademark to advertise it. The Landham Act (15 USC 1051-1127) is the key law within the United States that governs trademarks; they are also covered to a large extent by the same laws that govern patents (35 USC, 37 USC), and the same international treaties also apply.

My copy of the Berkeley course outline does not cover trademarks; nevertheless, I have adopted their format to note the distinctions with regards to trademark protection and the computer industry:

What can be trademarked?

1. *Unique or invented words or terms*.
2. *Unique number and letter combinations* that would rarely appear in normal or mathematical use.
3. *Unique graphic images* (logos, characters, mascots, etc.).
4. *Unique objects* (subject to size and other restrictions).
5. *Unique uses for a generic word or term* (e.g., *Microsoft Windows*™ is a piece of computer software, as opposed to a generic architectural construct).

What cannot be trademarked?

1. *Generic numbers and mathematical equations.*
2. *Generic words and terms* (provided that the intended use is clearly generic in nature).
3. *Generic graphics and geometric symbols.*

What are the advantages of trademark protection?

1. Longest life of any intellectual property protection - *trademarks last forever*.
2. *Can provide protection to any product that includes the trademark in some form.*
3. *Can be as stringent as a patent* with regards to use and proper licensing.

What are the disadvantages of trademark protection?

1. *Must be constantly maintained*; burden of proof-of-use lies with the owner.
2. *Weakest form of intellectual property protection*; covers only marketing concepts and not always the product itself.
3. *Subject to certain forms of unlicensed public domain usage.*

For further inquiries regarding the appropriate federal statutes of U.S. law that govern copyrights, patents, and trademarks, I recommend consulting the Internet version of the United States Code (USC). This is the codex that contains all federal laws that govern the United States of America, and it serves as the basis for the Code of Federal Regulations (CFR). There are several different sites and mirrors available, but the website of the Cornell University School of Law (http://www4.law.cornell.edu/uscode/) is a good place to start. A somewhat more user-friendly site with great background data (but limited only to intellectual property issues and associated sidebars; it does not contain the entire USC) is BitLaw: A Resource on Technology Law (http://www.bitlaw.com/).

THE LEGAL DANCE OF EMULATION DEVELOPMENT

"So why should I care about all this legal claptrap?" the more impatient among you are probably screaming by now. "I just wanna program a emulator, not go to law school!" *If you're going to write an emulator, then you're going to be walking all over one or more of these forms of legal protection for the original vendor and software developers.* "How?" you may ask? Let's see just how that can happen with videogame emulators, since these are the cause of emulation's current woes.

A videogame console, be it the standalone coin-op type found in the arcades or the ubiquitous book-sized boxes found in people's homes, is an extremely complicated piece of engineering. You have to have the code for the game(s) that it will run for starters. Next, you have to have the hardware that will make it possible for users to play the game(s) the way you intended. You have to have a way to store the game inside the console; likewise, you need some kind of processing hardware to interpret and execute the many lines of code within the game(s). You have to have some kind of method to generate a visual display. Said output must properly project the imagery that the game is creating in such a fashion so that it will not cause the user to suffer eye strain or possible seizures. Sound is not a necessary requirement but is extremely desirable, since it adds to the impact of the gaming experience on the user. You must have some kind of control interface so the user can interact with the game; this can vary wildly depending on the hardware and/or intended experience(s) involved. Finally, if the console is designed to handle more than one kind of game, you will need what the legal profession terms a "delivery system." This is a means whereby your game(s) can be stored outside of the console until such time as gameplay is desired, at which time they can be connected to or inserted within the console in order to deliver the game code into the console's hardware for execution.

Wow, that's quite a tall order, isn't it? That involves a lot of specialized knowledge in order to make such a piece of computer hardware a reality. This is where intellectual property rights become involved. *It really doesn't matter what kind of emulator you are designing for whatever system you have in mind to emulate - you have to do it in such a way as to avoid infringing upon the intellectual property rights of the original vendor and its licensees.* Remember our discussion of patents, copyrights, and trademarks? Here is where all of that "legal claptrap" comes into play.

- The name of the console, its outward appearance, any corporate "mascot" or logo that may have been created to promote the console, any "mascot" or other graphics belonging to the original program vendors, and images of all of the above are protected by trademark law. In certain cases, they may also be protected by copyright law.
- The commercial program base for the original console is protected by copyright law. If it is housed inside a custom delivery system or contains vendor-specific code that enables at least one vendor-specific process within the hardware, then it may also be protected by patent law.
- The original console and its hardware are almost always protected by patent law. If the inner workings contain any vendor-specific parts such as a BIOS, then those parts are indeed protected by patent law. Any internal

microcode to those parts is covered by copyright law and may also be covered by patent law.
- Graphics and sound processors are covered by patent law. The internal microcode to same is protected by copyright law and may also be covered by patent law. This is especially true if you are dealing with vendor-specific processors.

*The key word in all of this is **process**.* Remember, patents are used to protect a vendor-specific product or process. Patents are more protective than are copyrights; therefore, vendors will almost always try to use patent violation as the chief means of prosecution whenever they accuse an emulator developer of stealing their intellectual property. Software developers for vendor products are limited to claims of copyright violation unless they can establish that their code either contains a unique process or can be linked to a vendor-specific process within the hardware, in which case they also qualify for patent protection. Filings for trademark violation are problematic at best, due to the nature of the beast and the concept of "fair use" (which we shall discuss at a later time). *To wit, patents provide the heaviest legal protection for a vendor product, followed next by copyrights, and last and least by trademarks.*

You ever wondered why vendors chase after unlicensed developers of emulators regardless of their design? You ever wonder why software companies tend to frown on any modification of their titles? You ever wonder why certain companies are quick to claim violation of the computer code inside their proprietary hardware? Patents and copyrights are the chief reasons why. Patents are now the main assault weapons in the original vendor's arsenal, and patent protection packs some pretty heavy caliber. In comparison, copyrights have the status of a rapid-firing pistol with a large magazine - copyright protection packs a sustained punch, although not nearly that which patents provide. Trademarks, by their very nature, are naught but a derringer in a vendor's potential legal assault on an emulator developer - only one shot, but sometimes that shot can kill you. Their greatest role is played in validating claims of software piracy, but there is one specific case in which trademark infringement can be used against an emulator developer, and we will discuss it at the appropriate time.

The primary problem for an emulator author is finding a means to duplicate the vendor's patent-protected custom processes in such a way as to not violate those patents. *The only legal recourse for doing so without a vendor license is a technique known as **reverse engineering**.* This is the creation of a device or piece of code that duplicates the function of a given process while at the same time not being an exact copy of the original. The most common interpretation of this within the computer industry is that your process can work like the original vendor's, but it cannot contain any of the vendor's proprietary hardware or software. It can be put inside similar packaging with a somewhat similar-sounding name and interface in a similar way as the original, but it can't be a perfect clone. In other words, it might look like a duck, walk like a duck, swim like a duck, and quack like a duck, but it's not a real duck under those feathers. Bear in mind, though, that if a reverse engineered process successfully duplicates the function of a patented process but is not or cannot be proven to be an independent creation, then the original vendor will consider it illegal and respond in kind.

Frequently the situation will arise where a developer cannot surmount a certain technical problem, or decides not to develop their own reverse engineered code for any number of reasons. Instead, they opt to obtain a legal copy of a vendor creation for use in their own product. This is where **vendor licensing** comes into play. In this situation, an emulator author opts not to duplicate a certain process. The developer may elect to license the process from the vendor; they pay for the right to use that vendor-developed process in their product. They can also elect to require the user to purchase these materials directly from the vendor or one of its licensees in order to get the developer's product up and running. Regardless of the form it takes, vendor licensing is a universally accepted practice within the computer industry and one to which most vendors do not object. *In effect, it is a recognized legal technology transfer between the original patent holder and the licensor or user who wishes to employ a patented process to suit their own needs.*

INTROSPECTION

So let's assume that an independent developer has found some means of surmounting the many legal and technical obstacles in order to create a working emulator. How can the original vendor deal with its release? How can the developer combat the almost certain lawsuit that will be thrown his or her way? Does it matter whether or not the emulator is commercially released, or just handed out free in the public domain? Can the original vendor find some way to take advantage of the situation to its profit, or do we end up with nothing but a shouting match and lots of flame email? In short, just what is at stake with the release of an independently developed emulator? That's a good question, and that is what we shall look at next.

REVIEW QUESTIONS

1. According to Nintendo, what represents the greatest single threat to the videogame industry? Why?

2. What is "intellectual property?" How is one entitled to intellectual property rights?

3. What is the main hurdle one must overcome to have his or her creation qualify for intellectual property protection?

4. How do governments protect intellectual property rights? To what areas do these protections apply?

5. What are two different forms of protection for a work considered to be an expression? Which is preferable? Why? Give examples of each.

6. How does each form of intellectual property protection relate to the computer industry?

7. What two forms of intellectual property protection can be used to cover computer software? How can it qualify for both? Why would you want to use one as opposed to the other?

8. What form of intellectual property protection is said to last forever? Is this true, or not? Explain why.

9. Which two forms of intellectual property protection usually go hand-in-hand? Why?

10. What are two methods developers of so-called "clone products" can use to avoid intellectual property disputes?

11. In your own words, describe the best way developers have found to produce a clone product while at the same time avoiding legal hassles from the original vendor. How has this technique been justified in the courts?

12. If the developer of a clone product cannot surmount certain technical obstacles, then what can they do? Describe several methods of employing this alternate technique.

13. Is it legal for an end user to develop an emulator "for private use only?" How about "as a backup for the original console?"

14. Apply the concept of intellectual property to the two Amiga-based examples given in the introduction. What are the various ways in which intellectual property rights were honored, abused, or violated?

QUESTIONS TO PONDER

1. In your own words, describe how you as a vendor could pursue somebody who has violated your intellectual property rights by releasing an emulator. Give specific examples as to how you would achieve this.

2. In your own words, describe how you as a developer could create an emulator that would not infringe upon the intellectual property rights of a vendor. Give specific examples as to how you would achieve this.

Module One: The Emulator
Part 3 - Releasing an Emulator

> *I still feel that most emulators ... are perfectly legal. It kinda falls into that "legal to own, illegal to use" category that seem to cover game copying devices and cable descramblers.*
>
> Jeff Gerstmann, review editor, *VideoGames.com*

CONSPIRACY AND RETALIATION

28 January 1999 marked a significant event in the world of videogame emulation. It was on that day that a team of two hackers, known only by the aliases of Epsilion and RealityMan, released ***UltraHLE*** - the world's first working emulator for the Nintendo 64 videogame console. Many had tried over the previous year to devise an emulator for that system and failed. Nintendo was well aware of the rising interest in emulating its popular videogame systems, what with working NES and SNES emulators already available, and they had strongly supported the IDSA sweep against illegal Internet ROM sites during the spring and summer of 1998. They kept abreast of the growing possibility of N64 emulation and were the ones behind the shutdown of *Project Unreality*, the most promising N64 emulation project of 1998. They threatened legal action if that project produced a working N64 emulator, so the authors of *Project Unreality* abandoned their efforts at that time. Both Epsilion and RealityMan were already quietly conducting a parallel effort to develop a working N64 emulator, but they decided to try a different approach than that of the *Project Unreality* team. Instead of starting with the console's base functions and working upwards, they started with its high level functions and worked down (hence the name *UltraHLE* - Ultra High Level Emulator). They also decided, in light of Nintendo's public attitude with regards to emulation, that they would not tell anyone about their project until they felt it was ready, and then make it available to as many people as they could in the shortest amount of time.

The sudden and unexpected release of a working N64 emulator caught the emulation and videogame communities completely by surprise. What was even more insulting was that *UltraHLE* promised full compatibility with *The Legend of Zelda: The Ocarina of Time* (aka *Zelda 64*), the first N64 entry in the long-running and popular RPG series by Nintendo and its biggest seller of the 1998 Christmas shopping season. The release of a working N64 emulator promising full compatibility with their newest hit title was just too much to take, so it should not have surprised the emuscene when Nintendo reacted like it did. The blindsided videogame giant quickly rushed to shut down the *UltraHLE* website and threatened prosecution against the authors and anyone carrying or supporting *UltraHLE*, but it was too late. Epsilion and RealityMan's release tactics had ensured that *UltraHLE* would be spread far and wide given the nature of the Internet, and a helpless Nintendo could do little but watch as *UltraHLE* popped up on one site after another in quick and rapid succession. N64 cart dumps had already been available on the backwater sites for over a year, and now patrons of the "warez scene" had something with which to play their bootleg videogames. *Zelda 64* was the most sought-after "ROM" of the lot, and it took little effort for dedicated Internet users to eventually track it down. *UltraHLE* was by no means perfect, and only worked with about one-third of the "ROMs" in existence at the time, but it worked with *Zelda 64* and *Super Mario 64* (the console's flagship title), and that was enough for most folks.

Nintendo's next reaction was what those familiar with its history might expect. It continued to threaten legal action against anyone who supported *UltraHLE*, and threatened to file suit against its authors claiming that the emulator was an infringing work that promoted software piracy. The threats were considered to be pure bluff by the emulation community and many sites called them on it - those that weren't carrying any bootleg N64 "ROMz," that is. As for the lawsuit threat against the *UltraHLE* authors, it is still looming as of this date, and the resultant legal pressures coupled with the instant popularity of *UltraHLE* among the software pirates forced the temporary "retirement" of RealityMan for a period of several months. In the meantime, Nintendo continued to flail away at the illegal N64 "ROM" sites whenever it found them, but it was of little use. At least two more sites would spring up for every one Nintendo managed to shut down, often hosted by unscrupulous ISPs and frequently in countries where Nintendo's famous legal resources were of little avail. Nintendo continues to maintain to this day that *UltraHLE* violated the patents and copyrights that it held on the N64 internals and microcode, including the system's anti-piracy security system. While the construction of *UltraHLE* remains a closely guarded secret even to this day, both Epsilion and RealityMan contend that it was an 100% reverse engineered software-based emulator; furthermore, the anti-piracy feature was not an issue because they did not emulate it (which would have been illegal). In the meantime, though, Nintendo's legendary intimidation tactics continue to be employed against anybody who carries *UltraHLE* and the requisite "ROMs," but the sad truth is that *UltraHLE* continues to be freely available and can be downloaded by anybody with the will to find it.

This incident caused some serious soul searching among the emulation community, as well as many a comment from the videogame industry. Most were of the opinion that *UltraHLE* should not be supported because it was designed to emulate a product that was very much alive and kicking. The release of *UltraHLE* implied that the end-days of the N64 were near, and nobody really believed that. On the other hand, there were those who argued that they should not discriminate against a supposedly legal emulator regardless of the status of the original system, and these were the folks who continued to carry *UltraHLE* on their web sites while the rest refused. The videogame community was equally split, with many expressing admiration for the programmers and their accomplishment while others called for their prompt prosecution. The debate seems to have died down after RealityMan announced his retirement from the emulation scene, and not long after *UltraHLE* reappeared on just about every single major emulation site. Those who had fought its support before now felt justified in carrying it after Nintendo's failed attempts to stop it; furthermore, they noted that there were several public domain "ROMs" available for it just as there were for other videogame emulators. The availability of a legally downloadable software base for the emulator, however limited and incompatible that might be, seemed enough justification enough to many minds for supporting this remarkable program. As an afterthought, it is ironic to note that Nintendo announced plans to release a new videogame console (the N2000, later renamed as the **Dolphin**) not long after the release of *UltraHLE*, so perhaps those who feared the N64's impending obsolescence may have been right after all.

In previous discussions, we have seen that emulation is a valid use for a computer system; furthermore, that an unlicensed emulator is possible once the intellectual property concerns are properly addressed. The time has now come for the next important step: the release of an unlicensed emulator.

IMPERFECT EMULATION

In this and future discussions, I will refrain from discussing pure firmware emulators, as well as the more common varieties of hardware-software combinations. Firmware emulators are almost always produced by original vendors or other companies under license to the original vendor. Remember, the ability of the Sega Genesis/MegaDrive to emulate a Sega Master System via VDP mode 4 was due to Sega itself and installed as a desire to support its customer base for the older platform. Sony is taking the same position with regards to the announced release of its newer **PlayStation 2** console; they have included a PlayStation emulation mode that permits playing of many (but not all) older titles on the newer system. This is a practice that is universally accepted within the industry; furthermore, firmware emulators are quite profitable to the original vendor due to the proprietary nature of such. It would be almost impossible to build a firmware emulator without the consent of the original vendor, aside from the ubiqutous cart adapters, and the laws are quite clear with regards to the illegality of unlicensed clone products. Just ask all of the folks who tried to manufacture the many unlicensed clones of the NES console during its lifetime and see where it landed them.

A combination emulator is a different animal altogether, and their very nature tends to blur the legal lines somewhat. For example, Commodore's cross-licensing agreements with various PC clone manufacturers gave them solid legal ground upon which to base their PC Bridgeboard product for their Amiga computer systems. The same could be said of *A-Max*, as its only vendor-produced requirement was the use of the original Macintosh BIOS in its special hardware adapter - the Mac BIOS was the only truly proprietary part of the system - and this practice was eventually deemed acceptable. Jump forward in time to the present, where combo emulators have taken a new twist. The modern variety frequently require the use of a **BIOS dump**, which is nothing more than a software image of the computer code contained within the BIOS of the original system. Remember the *A-Max* bootleg? It's the same thing, but almost a decade later. I will reserve the discussion of the legality of this practice for another time, but will note in passing that such emulators have become widespread. For example, **Cloanto** of Italy sells a package called *Amiga Forever* designed expressly for users of the *WinUAE* Amiga emulator. Along with the emulator and other extras, it provides a legal copy of the AmigaDOS software, along with a legally licensed copy of the Amiga Kickstart ROM (the system BIOS, in other words). True hardware-software combos are becoming more rare with each passing day due to the incredible advances in computer processing capability, and the time will no doubt come when their existence will be limited to providing emulation for highly specialized niche systems that have a minimal impact at best upon the computer industry.

EMULATION AND REVERSE ENGINEERING

It goes without saying that *a pure software-based emulator should be comprised entirely of reverse engineered code before you release it to the world.* This neatly avoids any legal claim that the original vendor can mount against you in this regard, such as what happened between Intel and AMD over the Am386 (Intel vs. AMD, 1991). The courts have ruled time and again that reverse engineering does not void copyright protection, as it does not replicate the original's "defining algorithms." A program written entirely in 100% reverse engineered code is therefore legal, so it stands to reason that any emulator that is 100% reverse engineered should also be legal. The best way to do this is what is known as the "clean room" technique, in which reverse engineered code is developed on the basis of observing the actual operation of the original system and then coming up with an independently developed means of duplicating its various processes. The legality of the "clean room" technique was validated in the famous legal dispute over the cloning of the IBM PC BIOS (IBM vs. Compaq, 1982). The "clean room" technique is also the chief reason why Sony was unable to prevent the ***bleem!*** PlayStation emulator from going to market (Sony v. Bleem LLC, 1999), as the company had ample documentation to prove that "clean room" techniques were used in every aspect of *bleem!*'s design. This has established beyond the shadow of a doubt that the legality of the "clean room" technique also applies to emulator development. Another example you might consider is Steve Snake's **KGen** emulator for the Sega Genesis/MegaDrive, which was also a "clean room" product. It proved to be such a good program that Sega eventually licensed the source code from the author for use in its commercially released *Sega Smash Pack*. Talk about coming full circle!

As a final nod to the issue of of reverse engineering, let me share something with you from a leading major industrial organization with whom I have dealt in an indirect way for many years. One of the chief trade organizations in the electromechanical industry is the **Institute of Electrical and Electronics Engineers** (IEEE). You normally hear about them whenever there is a dispute about specifications for electrical or mechanical components; they are the industry's means of regulating itself. Here are some significant excerpts from their official policy statement regarding the concept and practice of reverse engineering which you might find interesting:

> ...We also believe that the high intellectual content of a computer product and competition are enhanced when *computer products developed by one vendor are capable of operating with computer products developed by another vendor*. This compatibility promotes the development of interoperable products by independent and competing vendors and, therefore, promotes enhanced value to the vendee at a competitive price....
>
> We further believe that *lawful reverse engineering of computer programs is fundamental to the development of programs and software-related technology*.... We further believe that lawful reading, analysis, or disassembly of machine language is a reverse engineering technique by which an engineer can reconstruct the ideas of a computer program.
>
> Accordingly, an engineer having the right to use a copy of a computer program should be entitled, without the authorization of the author, to observe, study, and test the functioning of the program, in order to determine the ideas that underlie the program, if it is accomplished while performing any of the acts of reading, displaying, running, transmitting, receiving, or storing the program or other lawful acts involving the program that the engineer is entitled to do.
>
> We support the fair use rulings in the recent Appellate Court decisions in the Ninth and Federal Circuit, in Sega Enterprises vs. Accolade, 977F.2d 1510 (9th Cir. 1992) and Nintendo vs. Atari, 975F.2d 832 (Fed Cir. 1992) pertaining to disassembly of computer code.

Pretty strong stuff, isn't it? And from one of the "big guns," too. These are the people who tell companies like Nintendo exactly how they can build their systems in such a way as to be considered safe for its users. The IEEE would not issue a policy statement with regards to the legality of reverse engineering and its underlying requirements unless they were absolutely sure of their facts. These facts were recently codified into law by the **Digital Millenium Copyright Act**, which I shall hereafter often abbreviate as the DMCA. Emulator developers should take this to heart whenever an original vendor begins making broad claims of copyright violation against them for the use of reverse engineered code. Such claims are frequently unfounded, and are almost always proven so when the issue is pressed in the courts.

THE PITFALLS OF EMULATION RELEASE

Finally, the day arrives when you plan to unveil your emulator. Whether you are a commercial company with considerable time and resources invested in your product, or you're just an extremely gifted hacker like the one described earlier, the time has come to make your creation known to world. What happens next? Well, that depends on two things: *who originally vended the system* you are emulating, and *how old that technology might be*.

There has been a growing acceptance of emulation within the computer and videogame industry; however, it is not yet universal. Nintendo's attitude is well known and is embodied in their official emulation FAQ:

> The *UltraHLE* [sic] is illegal. The N64 emulator infringes Nintendo's intellectual property rights, including copyrights, and circumvents Nintendo's anti-piracy security system.

While you may not agree with their stance and not all of their contentions may be provable, it nevertheless highlights one extreme of the scale that original vendors use to measure emulation. A completely different approach is taken by Hasbro, now the owner of Nintendo's longtime competitor Atari, who recently released all rights for the now-defunct **Atari Jaguar** videogame system into the public domain:

> We realize there is a passionate audience of diehard Atari fans who want to keep the Jaguar system alive, and we don't want to prevent them from doing that. We will not interfere with the efforts of software developers to create software for the Jaguar system.

These are the two extremes you will have to face when deciding how the vendor will react to the release of your unlicensed emulator. Each company chooses to respond in a different way for different systems at any given time, so attempting to predict their behavior is not always successful.

The other item to consider is the age of the technology you are emulating. Has the system been dead and gone for several years, or its it just on the verge of expiring? Or perhaps, as was the case with *UltraHLE*, the system you wish to emulate is still economically viable. What then? Before you answer that question, try looking at the issue from the vendor's perspective, and Nintendo's emulation policy statement makes an excellent point in this regard.

> Copyrights and trademarks of games are corporate assets. If these vintage titles are available far and wide, it undermines the value of this intellectual property and adversely affects the right owner. In addition, the assumption that the games involved are vintage or nostalgia games is incorrect. In fact, there are now more and more programs available that emulate current game systems such as Game Boy and the Nintendo 64.

If you release an emulator for a product that is still on the market, you are presenting yourself as a direct threat to that product's market share. That fact, in retrospect, seems to be why Nintendo reacted as violently as it did to the appearance of *UltraHLE.* Epsilion and RealityMan boldly bragged about their emulator's ability to handle *Zelda 64*, along with other hit N64 cartridges at the time. *UltraHLE* directly threatened Nintendo's revenue stream on those titles, and that of the newly released *Zelda 64* most of all.

Original vendors, like most companies, try to maximize sales in order to achieve maximum profit at minimum cost. They only have so much time to sell that product before something new comes along or a competitor releases a better or cheaper product. On top of that, most computer-related products have an fairly short shelf life due to rapidly advancing technology. As Moore's Law puts it, processing power doubles every eighteen months. Your emulator could cut in on the original system vendor's best market window - not very much, perhaps, but enough to make a difference on the balance sheets or come tax time. Do you think they relish this thought? Not at all. Vendors of systems that have long since gone off the market or are on the verge of dying anyway usually don't complain about emulation, but it is the "live" systems that pose the biggest risk to emulate. An excellent example of this comes from the personal computer world of the 1980s, during which time Apple Computer went after anybody and everybody who tried to market clones of their computers in any form, shape or fashion. Products such as the Franklin ACE 1000 clone of the Apple II and Readysoft's *A-Max* emulator for the Amiga threatened those projected profits. The same holds true today, with Apple back up to its old tricks again - not to mention Nintendo running to ground the makers of every off-brand unlicensed derivative based on one or more of its proprietary product lines anytime one pops up. The corporate mindset that lies behind prosecuting those who make and distribute unlicensed products works equally well against those who make and distribute unlicensed emulators.

So why do these vendors get so upset about emulation? Well, look at it from their point-of-view. Somebody wrote the code to that game you're playing on your emulator. A mid-sized company paid them for it or the license to it, and then a big company bought the rights to vend it. On top of that, system vendors have their own financial arrangements whenever it comes to developing and/or porting software to and from their proprietary hardware. That software is designed to be specifically used with the original vendor's system. With regards to that system, somebody spent a lot of money researching and designing it. They may have contracted others to do it for them, or they may have made it themselves. It costs money to conduct research and development, arrange intellectual property protection, design promotion and advertising, tool up for manufacturing, and so on. The only way they are to make that money back is to sell their product at a profit - which means they have to sell lots of units and lots of software for that system in order to get a return on their investment. Paid ... bought ... financial arrangements ... money ... contracted ... profit ... sell ... investment - do you see the common theme? Now do you understand why a firm like Nintendo screams bloody murder whenever somebody comes along and emulates a system that has yet to even show signs of finishing its market run? Do they get upset? You bet! *In their eyes, emulation is an infringement not only of their intellectual property rights but also a direct threat to the profitability of their corporate assets*

- however small in truth that may be. As long as said assets remain viable as far as the markets are concerned, as long as such threats exist, and as long as there are legal means to eliminate them, then you can expect them to defend their sources of revenue by any means necessary.

After reviewing what few cases have been brought to bear in the courts concerning computer system emulation, it seems that there are three primary attacks used by an original vendor against these so-called "infringing products." It is not surprising that each falls under the three major categories of intellectual property protection:

- **Patent violation**: Emulation *permits the bypassing of any anti-piracy protection measures, as well as unlawfully replicating other proprietary vendor processes* internal to the hardware and original software of the system. This represents a direct threat to the intellectual property rights of the original vendor.
- **Copyright infringement**: Emulation promotes software piracy because *users are no longer required to buy the original system*, and as such represents a clear and present threat to the original vendor's profit margin. It is also a given that *users may no longer be required to obtain the system's software in its original commercial format*. Since vendors traditionally reap the most profit from software sales, this further impacts potential revenue.
- **Trademark concerns**. It is a fact that *emulators are never as good as the actual hardware*. This could contribute to a detrimental mindset against the original vendor, as an emulator does not accurately replicate the experience of using the original system. As such, emulation poses a direct threat to both the corporate image and quality control of the original vendor and its affiliated licensees.

It should be noted that these are all valid charges, and vendors with this mindset can bring an amazing array of facts and case law precedent to back up their claims. Here are some examples for you to ponder.

- Nintendo views any N64 emulator that either somehow invokes or includes emulation/bypass code for its proprietary anti-piracy security system (APSS) to be both patent and copyright infringement. It violates the patents on the hardware by either emulating or invoking the internal APSS code, and it violates the copyrights on the code because any tampering with a computer security system (as Nintendo regards its APSS) is in direct violation of the DMCA. The N64 emulator developers have contended these many months that the APSS is not an issue, since it is not required for actual program execution; however, Nintendo often adds additional copy protection and internal hardware checks to its program cartridges that are not dependent on the APSS. Nintendo views these as an additional "security system;" therefore any emulator that can run so-called "cracked" N64 dumps is still guilty of violating the DMCA.

- Sony views any PlayStation (PSX) emulator out-of-hand as an infringing product. It is their contention that any successful PSX emulation would unlawfully duplicate the proprietary code within the Papillon chip, which is its name for the PSX BIOS. They were able to provide reasonable proof of this against Connectix, makers of the *Virtual Game Station* for the Mac, and were thus able to halt further marketing until the matter came to trial (Sony v. Connectix, 2000). They were unable to provide such proof against Bleem LLC, makers of *bleem!* for IBM PC compatibles, because the company had more than enough evidence to prove "clean room" reverse engineering of the PSX BIOS (IBM v. Compaq, 1982). The U.S. 9th Circuit Court of Appeals eventually ruled that Sony's original copyright and trademark claims were without grounds (Sony v. Connectix, 2000), so Sony has resorted to the attack it should have taken in the first place - multiple counts of patent infringement of proprietary Sony hardware. They are also reportedly considering filing a second copyright lawsuit contending PSX security scheme violation under the terms of the DMCA as this article goes to press.
- It should also be noted that Sony initially declined an offer from Bleem LLC to market their emulator, judging it too slow and incompatible to truly replicate the PSX "product experience." To go back in time for a moment, the problem that Apple originally had with *A-Max* was that it too accurately replicated the Mac "product experience" on an Amiga. This should have come as no surprise, since the only real difference between the hardware requirements for the two systems was the Mac BIOS itself! The Amiga had comparable hardware to the Mac (and more), so emulating the Mac was relatively easy. The same holds true when comparing an actual Sony PSX to a 3D-accellerated, Pentium-powered IBM PC computer, although it has been a long-proven principle of emulation that a pure software approach is usually slower than working with hardware (as is the case with *bleem!*).

So, with the litigious nature of certain proprietary-minded vendors in mind, there is an obvious question that needs to be asked. *Is it possible to develop and then release an unauthorized, noninfringing emulator of a proprietary computer system? The answer is **YES**, but there are some qualifications that must be met first in order for such an emulator to be considered a legal product.*

THE *A-MAX* TEST FOR EMULATOR LEGITIMACY

It has taken me a lot of time, careful research, and a good deal of deliberation in order to come up with a generalized seven-point test that both the vendors and the emuscene can use to determine the legitimacy of any emulator that comes down the pike. I call it the ***A-Max* test**, so named after the emulator that established the possibility of unauthorized legitimacy in the first place. It works equally well for any product being emulated, from hand-held electronics to sophisticated computer and videogame systems. I will first present the test itself, then I will explain it, and then finish up with some examples of how it can be used to determine emulator legitimacy.

1. **The original system's microcode is not duplicated.**
2. **The original system's firmware is preserved.**
3. **The original system's operating system is not violated.**
4. **The original system's security measures are not circumvented.**
5. **The original system's delivery format is not discarded.**
6. **The original system's trademarks are not compromised.**
7. **The original system's economic viability is maintained.**

Let's take a moment to see just what each of these points involve.

> *The original system's microcode is not duplicated.*
>
> In other words, the emulator was created completely from scratch using lawful reverse engineering techniques (IBM v. Compaq, 1982; AMD v. Intel, 1991; Sony v. Bleem LLC, 1999; Sony v. Connectix, 2000). It only emulates a select range of system functions that are absolutely necessary to achieve desired results **and nothing more** (Nintendo v. Atari, 1992). This is almost always the first infringement that an offended original system vendor will claim.
>
> *The original system's firmware is preserved.*
>
> In other words, the emulator does not employ any proprietary microcode of any kind from within any part of the system hardware unless said code is licensed from the original vendor (Apple v. Franklin, 1983; Intel v. AMD, 1991; Sony v. Connectix, 2000). If it requires some portion of proprietary microcode in order to achieve desired results, then said system code must be either licensed or retained within its original "package." If the "package" approach is employed, then provision must be made for the emulator to interface with that "package" in its original, unaltered form (the *A-Max* affair of 1989; Nintendo v. Atari, 1992).
>
> *The original system's operating system is not violated.*
>
> In other words, the emulator may not be distributed with any portion of unlicensed vendor operating system code, whether that be in software form external to the system or embedded within its hardware (Apple v. Franklin, 1983; Sony v. Connectix, 2000). This includes the original system BIOS, any patentable custom chipset unique to the original system, and any support programs designed to facilitate the proper operation of the original system, more commonly referred to as a software-based operating system. For example, all of the early CP/M emulators included licensed copies of the actual CP/M operating system by Digital Research. In another example, Cloanto's *Amiga Forever* is a commercial Amiga emulation package designed around the public domain emulator *WinUAE* that includes licensed copies of both the Amiga Kickstart BIOS and the AmigaDOS operating system. Any emulator that is

distributed with unauthorized copies of operating system code regardless of its origin (hardware or software) can be considered an infringing product.

The original system's security measures are not circumvented.

In other words, the emulator does not bypass, alter, or otherwise modify any technological measures incorporated into the original system, its supporting accessories, or its designed delivery format in such a manner that affects controlled access to the original program base. The wording is taken almost directly from the DMCA (17 USC 1201) and applies to any embedded security systems contained either within the original system itself or as part of the delivery format for its program base. Some authorities have contended that the DMCA effectively overrides the duplication of copy protected software for backup purposes as previously deemed permissible by case law (Vault v. Quaid, 1987; referring to 17 USC 117). It is the opinion of most authorities that the DMCA complements the Vault decision rather than compromising it; to wit, you have the right to back up copy-protected software *so long as you back up the copy protection scheme with it*, and that is entirely in keeping with the Vault decision. All the DMCA would do would be to make it illegal to "crack" such a scheme, as well as then removing that copy protection scheme from the program in question, thereby facilitating easier duplication and distribution. This item will eventually pose a real problem for future emulator authors as vendors strive to design future systems in such a way so that emulating an embedded security system may become a necessity, which would place the legality of such an emulator at risk under the terms of the DMCA.

The original system's delivery format is not discarded.

In other words, the emulator must make some provision for using original copies of system software contained within their original delivery format unless its development team has obtained the proper authorization to do otherwise. Many emulators are designed from the onset to handle one or more special program formats normally employed by developers; however, *any publicly released version of that emulator (commercial or public domain)* ***must*** *make provision for the original program delivery format*. It does not matter one whit whether or not such a delivery format is still in use, because the courts have interpreted the ultimate right of adaptation (i.e. derivation) under copyright law, which belongs to the copyright owner, to include the intended or original delivery format (Mirage Editions v. Alberquerque ART, 1988). This is the big problem with most "classic" videogame emulators and some "classic" computer emulators in that they do not include any provision for loading original system programs from their original delivery format. So-called "transfer programs" or specialized hardware that converts original software to a special format are not permissible unless so authorized, and this is another problem with these kinds of emulators (Sega v. MAPHIA, 1994; Sega v. Sabella, 1995; Nintendo v. Computer & Entertainment, 1996; Nintendo v. Bung, 1999). I note in passing that a potential "personal use" argument for derivative formats has been raised under the terms of the Betamax decision (Sony v. Universal, 1984); however, it has yet to be acknowledged by the courts.

The original system's trademarks are not compromised.

In other words, the emulator does not contain any code or data to generate any kind of registered trademark in any form in and of itself without proper authorization, nor does it contain any code or data to effect any unauthorized alteration or modification of any kind of a permanent nature to any registered trademark that might be generated by a program running on that emulator. The practical upshot of this is that your emulator cannot generate any trademark displays that were generated by the original system hardware in and of itself (Sega v. Accolade, 1992); and while it is legal to generate the display of any embedded trademark contained within a program designed for use on that system, you may not do anything that would change or possibly block the permanence of said trademark (Playboy v. Frena, 1993). Generating an embedded trademark is considered innocent infringement (Sega v. Accolade, 1992), but doing anything beyond displaying that trademark is in direct violation of the Landham Act, as the Sega and Frena cases clearly demonstrated. Temporary alteration of a generated trademarked image is permissable under certain circumstances; however, permanent alteration is not (Galoob v. Nintendo, 1992)

The original system's economic viability is maintained.

In other words, the emulator does not constitute a significant threat to the current or potential market impact of the original system (Sony v. Connectix, 2000). The concept of *economic viability* is a rather hazy one and is usually the second thing that an offended vendor claims with regards to a possibly infringing emulator. It involves the notion that a given product with one or more unique selling points has only so much time on a free market to exercise its capability to bring its vendor a profit. Once that capability vanishes, then the product is no longer profitable (Harper & Row v. Nation Enterprises, 1985). There is no clearly defined point at which a given vendor's product economic viability disappears, and it would be foolish to try and come up with a "one-size-fits-all" approach. This part of the *A-Max* test must be determined on a case-by-case basis as necessity dictates and in regards to both the particular system being emulated and the amount of time it has already spent on the market. It can be and has been successfully argued that the older and less technologically advanced a given system has become, the more likely it is to be emulated since sufficient time has passed for any willing third party to duplicate its functions (Kewanee Oil v. Bicron, 1974). This is probably the best guide by which to judge economic viability.

Now that we understand what the *A-Max* test entails, let's see how it can be applied. I shall use four different emulators as examples: Readysoft's *A-Max* (as it defines the test), the original release of Epsilon and RealityMan's *UltraHLE*, Cloanto's *Amiga Forever* (the commercial release of Bernard Schmidt's

WinUAE), and finally Vision Thing's improved PSX emulator *PSEmu Pro*.

#	***A-Max* test item**	***A-Max (all vers.)***	***UltraHLE 1.0.0***	***Amiga Forever***	***PSEmu Pro***
1	The original system's microcode is not duplicated	X	X	X	X
2	The original system's firmware is preserved	X[1]	X	X[2]	-
3	The original system's operating system is not violated	X[1]	X	X[2]	-
4	The original system's security measures are not circumvented	X	?	X	X
5	The original system's delivery format is not discarded	X	-	X[2]	X
6	The original system's trademarks are not compromised	X	X	X	X
7	The original system's economic viability is maintained	X	-	X	X

(1)Uses original system BIOS ROMs in a legal fashion (actual chips in special adapter)
(2) Licensing involved to clear legal hurdles

I know that this analysis is going to have a lot of *UltraHLE* and *PSEmu Pro* users up in arms, to say the least, so let's see just why they failed the *A-Max* test.

- *UltraHLE 1.0.0* - There are no real issues insofare as the N64's trademarks, operating system, firmware, and internal microcode are concerned. Almost everybody who has analyzed *UltraHLE* (save Nintendo, of course) is in agreement on this much. It's the other three items where the problems arise. The subject of whether or not this N64 emulator actually violates Nintendo's APSS is an open matter of debate that will not be resolved until the source code is made available for analysis - hence the question mark in point 4. It doesn't matter anyway, because *UltraHLE* goes on to fail the two remaining parts of the test. It **will not work** with N64 titles in their original delivery format, which are videogame cartridges, nor is there any provision for such; thus if fails point 5 (Mirage Editions v. Alberquerque ART, 1988). It clearly impacted sales of N64 consoles and cartridges at the time of its release by whatever reasonable yardstick you want to use, therefore it fails point 7 (Harper & Row v. Nation Enterprises, 1985).

- *PSEmu Pro* - Here the situation is a little different. This PSX emulator supports actual PSX discs, which effectively negates all of the arguments that bring down *UltraHLE*. Videogame software is the true money train of any such system, therefore economic viability is not an issue. The main problem here is its requirement of an unauthorized copy of the PSX BIOS for operational use. Using an unauthorized BIOS dump instead of the actual BIOS compromises the original system; the item in question is a derived adaptation, not an authorized part, therefore it fails point 2 (Nintendo v. Atari, 1992; Sony v. Connectix, 2000). The use of copyrighted BIOS code in an unauthorized manner is a violation of copyright law, therefore it fails point 3 (Apple v. Franklin, 1983).

I challenge you to apply the *A-Max* test to your emulation project under development, or try it on your favorite emulator by another party for a different system. You may be surprised by the results.

One final reminder before we leave this subject. *Even if a given emulator fails the <u>A-Max</u> test, it is still the the responsibility of the vendor to actively claim perceived intellectual property infringement.* They are required to do so by law within a set period of time from the date of the perceived infringement, depending on the manner and type of infringement involved. If they do not claim infringement during this time, then they lose the right to do so in the future (35 USC 286, 17 USC 507, 15 USC 1065). This right lasts as long as their intellectual property protections are in place, and they have been known to wait until what seems to them like an appropriate time to strike. For example, Sony knew about the two commercial PSX emulators under development, but they waited until just before they were to hit the market to file their lawsuits. This is something you need to keep in the back of your mind should a vendor initially fail to object to the release of your emulator.

<u>INTROSPECTION</u>

Up to this point, we've talked about the basis for emulation. We've talked about developing an emulator. We've just finished talking about releasing an emulator. So what conclusion can we draw about emulation in general? Apply Occam's Razor and draw the obvious one. As *KGen* author Steve Snake recently noted in an Internet message board posting, "Emulation is legal. It's that simple and not open to debate."

Commercial vendors can no more stop emulation than King Canute could order back the tide. Vendors with closed minds have little chance of stemming the production and subsequent release of emulators for their systems for very long, if ever at all. Other companies have tried to do so in the past with their systems and failed. It is legal for original vendors to develop their own emulators. It is legal for an independent developer to make their own emulator, provided it does not violate the original vendor's intellectual property rights. It is legal for an independent developer to release such an emulator without the original vendor's approval, provided that it can negotiate the legal hurdles in its path. Once those obstacles are overcome, then an independently developed unauthorized emulator has every right to be released to the general public.

We have now come to the end of the various discussions with regards to the legality of emulation. You should now know what is involved in developing, releasing, and possibly defending an independently developed unlicensed emulator. Now comes one of the biggest issues of all, and it represents (as the British would say) a "sticky wicket" indeed: "How do I support an emulator?" That will comprise our next big area of discussion.

REVIEW QUESTIONS

1. Why is hardware-based emulation not considered to be a problem?

2. What particular issues do developers of emulators involving hardware-software combinations have to consider?

3. What is the best means of development in order to produce a legal unlicensed software-based emulator? Name two examples of such a product from the text or provide valid ones of your own. What does the IEEE have to say about this development practice?

4. What are two items of concern that independent developers must consider before releasing an unlicensed emulator?

5. What are the three primary arguments that vendors use whenever they sue the developers of an unlicensed emulator? What charges can you derive from those arguments?

6. How can an emulator developer defend themselves against charges of intellectual property infringement by the original vendor?

7. What is the truth behind "antipiracy security systems" in their current form? What are some ways to deal with this issue? Why do some emulator developers choose to ignore the problem in their products?

8. What does the release of an emulator usually indicate with regards to the original system? Why would this upset the original vendor?

9. Which is easier for a emulator developer to work with - program code within the original delivery format or an alternate format? Why?

10. How does software piracy affect the choice of delivery systems for a given vendor? How does this impact upon the release of an emulator?

11. What was the chief mistake committed by Epsilon and RealityMan concerning the release of UltraHLE? When is the right time for a developer to release an unlicensed emulator? Why?

12. What causes certain vendors to become concerned whenever an emulator is released for one or more of their systems?

13. Describe the hurdles that an emulator must jump in order to be considered a legal, noninfringing product. Why do some succeed and others do not?

14. Do you think that it is possible to release a legal third-party emulator? Why or why not?

THOUGHTS TO PONDER

1. If it is legal to design and then release an emulator, then is it legal to provide copies of the commercial software originally designed for use with the system being emulated?

2. Is it legal to dump a BIOS image in order to avoid having to create an emulator that would otherwise be a combination in design?

3. What is a "ROM?" Is there such a thing as a "legal ROM?"

4. Is it legal to backup computer software that was originally vended in some form of permanent storage format, such as a game cartridge or CD-ROM?

5. Can you define the concept of fair use? How does the concept of fair use apply to emulation?

6. How can the Internet legally support the emulation community?

Module Two: The Software
Part 1 - Establishing the Software Base

Thou shalt not steal.

The Eighth Commandment,
Exodus 20:15 (KJV)

THE EMULATION WAR

There is a war raging between the original system vendor community and the emulator development community. A steady barrage of emulators of all kinds thunders down upon the original developers. Undaunted, they hold their positions as the rain falls. Some shots go awry, some only cause minor damage. A few, though, strike their mark, and these are the ones that their generals worry about. They zero in on the source of their affliction, and then respond to their tormentors with an overwhelming barrage of threats, intimidation, angry emails, and attorneys. They relentlessly pummel the ones who dared strike so close to them in the hope that it will silence the offenders. Their hope is in vain. From farther across the slope, or some way down the valley, another group opens up and lobs a shot a little closer. The vendor now has two threats to worry about. They respond in kind, but not as strong as before, since they are having to now having split their resources between two threats. Yet another round is lobbed by the emulation community. The harried vendor is beginning to feel the strain. And so it goes, until the gunfire falls silent for a time - not because anybody is winning, but because both sides have ran out of ammunition.

As with any war, there are casualties. Vendors bow to the pressure of increased competition from their more ruthless fellows, as well as the any emulator developers who might be sniping at them from across the ridge line. More often than not, one or more emulator developers take a direct hit or are wounded severely enough to take them out of action for some time. The war in the marketplace is matched by a war of words that is just as fierce, and those who have the courage and the will to speak are often ruthlessly ground down by their implacable foe. It doesn't have to be this way, but that's the way it is. Capitalism is the rule book by which the computer industry plays in the free world, and its maxims are both stern and unforgiving.

There is, however, one group in the equation who tends to be largely ignored, just as they are in the real world. They are the civilians - or in this case, the users. You see them roaming through the stores, greedily eying the fancy displays with the expensive products, and then digging frantically through the bargain bins for that one product that they saw last week but didn't have the money to buy at the time. You see them in the main office or at the cubicle next to yours, ostensibly working on the quarterly report while keeping a muted videogame hidden under that spreadsheet window. You see them in your living room - you and your friends, or possibly your children and their friends, eagerly examining a new piece of computer software or checking out the latest cool videogame for your home console. You also see these masses as they make their presence felt on the Internet just by sheer volume, spinning up the counters at the sites they like to frequent. Users are to the software industry what civilians are to a nation state - they provide the resources and manpower for its economic viability. Without a civilian population, a nation state does not have the necessary labor pool to wage war. Without a user base, an original vendor or emulator developer does not have the necessary support base to continue a given product. They may sit on the sidelines and watch while the war rages on, but users are the key ingredient when it comes to winning the emulation war, and they always find some way to take advantage of the resultant fallout.

The one tool with which users can influence the battle at hand is with their pocketbooks. Users buy the products they support and avoid the ones they don't; however, anytime they can get something for nothing, some will. While a complete discussion of the economics of the computer industry is beyond the scope of this document, suffice it to say that users tend to trend towards cheaper, better products and shy away from pricey, monolithic ones. The recent rise of Linux in an operating system world dominated by *Microsoft Windows* illustrates this principle. It also stands to reason that if a user can get something for free, many will do just that. That is the prime reason why Internet game sites dedicated towards playable demos, such as Happy Puppy, continue to be extremely popular. It's the old "getting something for nothing" principle in action. You get a useable product, albeit "crippleware," and you pay nothing for it. Beats paying US$10 for the same demo at your neighborhood CompUSA, doesn't it? Do you blame them when a lucky few come across a means of getting full-version software for free? Perhaps you might, but try and look at it from their point of view. Why pay for something when you know a way to get the same product for nothing? After all, they are bills to pay and mouths to feed. Anytime that these folks can cut corners, they will, and to hell with the original vendors.

On the other hand, vendors are looking to maximize profits while minimizing costs. It is to their advantage to force users to pay as much as conceivably possible for their product - that way, profit margins skyrocket and they get the funds they need to both stay in business and maintain research and development into other projects. Nintendo's "inventory management" is an oft-cited example of the ruthless but legal methods that it employs to ensure a constant and sizable chunk of the videogame market. They are not the only guilty party in this regard, but they are the best at the game. Anything that represents a threat to that market share is an enemy to be destroyed, and videogame emulation represents a clear and present danger. They stand to lose money from console sales due to the presence of a working emulator, and they stand to lose royalties from bootleg software manufactured and distributed to support that emulator.

If you as a user plan to use an emulator on your computer, then you will need software to use with that emulator. That point is an obvious one; however, it is one that is either glossed over by the emulator authors or overemphasized by the original vendors and their allies. The time has come to deal with the issues regarding emulation support, and the issue of software availability is the biggest one of all. Before we get into the details, however, we need to deal with the concept of software piracy, take a brief look at how intellectual property laws protect against it, and then come to grips with what is perhaps the most common justification for it - the concept of fair use.

THE ECONOMICS OF SOFTWARE PIRACY

Software piracy *can be defined as the unauthorized duplication, modification, or distribution of any computer program in direct violation of copyright law.* As long as there is software, there will always be software piracy. This is a given within the industry, which is why organizations such as the Software Industry Information Association (or SIIA; formerly the Software Publishing Association, or SPA) and the Interactive Digital Software Association (IDSA) were formed. While the industry admits that the impact of software piracy as a whole is negligible, nevertheless specific cases of software piracy can have a profound impact on its manufacturers. The IDSA notes that as of 1999 it took as much as US$1.5 million to develop a commercial program; also, that in 1998 the entertainment industry lost over US$3 billion to software piracy. The potential economic impact of software piracy on a given piece of computer software is therefore quite significant.

Examples of software piracy abound in the general computer industry, where bootleg copies of computer games and applications software have run rampant for many years. Indeed, industry experts estimated back in 1984 that a full one-half of the then-existing software base for computer-based systems had already been pirated. Why is this practice so widespread? Many explanations have been offered, but I will stick to the simple one - it's the old "something for nothing" principle we discussed earlier. What kind of effect can it have on the software industry? Meheroo Jussawalla and Hajime Oniki, in "Changing Technologies and Intellectual Property: The Economic Perspective" present what I shall call *the vicious cycle of copying*:

- copying permits acquisition of knowledge at reduced cost
- copying reduces the number of originals sold, thereby inflicting economic harm to the copyright owner
- copying may force the copyright owner to increase costs
- copying increases as the price of originals are raised

Based on my own years of experience in both the retail and computer support industries, I find this model to be too simplistic and not reflective of "real world" practices. I would like to present an alternate at this time, divided into two phases - *setup* and *cycle.* It is somewhat different from the Jussawalla and Oniki model.

Setup

- a vendor spends a considerable amount of time and resources in developing a new and original piece of computer software for eventual release
- a bootleg version of that software is also developed, based on or duplicated directly from the original, without the approval of the vendor
- the original is released to market by its vendor with the highest possible price, given the buying power of its intended customer base along with the twin threats of possible competition and software piracy, to ensure maximum profit at minimum cost in the shortest amount of time
- the bootleg is also released to market - this could be before, at the same time, or after the original, depending upon the perceived popularity of the title and/or the availability of a master copy for pirating purposes

and now the cycle begins....

Cycle

- profits for sales of the original do not meet projected expectations due to widespread availability of the bootleg

- the vendor is forced to maintain the price of the original for a longer period of time than intended in an effort recapture lost revenue

- the resultant artificially induced maintenance of the current price increases the popularity of the lower-cost (or no-cost) bootleg, resulting in its further distribution

- the vendor is forced to either adopt new advertising techniques or lower the retail price for the original, thereby reducing net profit

- more originals are sold at a reduced profit

- the change in marketing stance by the vendor creates the perception that it is losing interest in the original, thus further increasing the general acceptance of the bootleg

- the cycle repeats ad infinitum until the vendor is forced to either sell off the title to a third party or

> cease its manufacture altogether, and it may never
> realize the full extent of its projected net profit

The only major difference between my expanded model and the simpler one is that my perception has been that prices for computer software fall over time, rather than increase. Regardless of which model you prefer, the only real out from the vendor's perspective is to change their marketing strategy (new advertising and/or change in price) in order to sell more originals, thereby hoping to offset the twin impacts of reduced revenue and unlicensed duplication. While this will not stop software piracy, it does help limit its impact. This solution is not an ideal one, though, because vendors can do this for only so long until it is no longer worth their while to maintain the marketability of the software in question. *It is often quicker and more economical for vendors to prosecute software piracy, and that is exactly what most chose to do.*

Unfortunately, in the real world, the unauthorized duplication of computer software is such a universal practice that it would be both impractical and uneconomical for copyright owners to chase down every single recipient of a bootleg. Instead, they focus on the unlicensed distributors, as their activities are the more obvious and harmful and thus easier to identify and prosecute. They are the main link in the software piracy chain - break the link, and unauthorized distribution is for the most part eliminated. What does that mean to the average user? It seems to be the operational rule-of-thumb that what you as a user do with the software that you have obtained for use while you are in the privacy of your own home is generally regarded as your business and nobody else's. The legal justification for this is **the famous "Betamax" case** (Sony v. Universal, 1984), in which the U.S. Supreme Court ruled that was not illegal for a homeowner to use a VCR to record a copyrighted TV broadcast to watch at a later time, so long as that recording remained with the homeowner. The recording was for personal use, without the intent to distribute, and therefore deemed to be noninfringing. On the other hand, it is what you do that directly affects the public (those outside your home) that is of concern to software vendors. It may take place outside your home (the "copy parties" and "warez meets" of old) or inside your home (digital transmission via a modem or other telecommunications device), so long it can be show to directly affect the public in some way (Sega v. MAPHIA, 1994; CompuServe v. Patterson, 1996; Panavision v. Toeppen, 1996). In short, *personal use ends where public use begins*. Making available unauthorized copies of computer software, regardless of where you are and how you do it, automatically makes you a unlicensed distributor, i.e. "software pirate," and therefore guilty of software piracy. *It is in your best interest not to be identified as a software pirate by any offended vendors seeking retribution*, and they can and will come after you if possible with every force of the law that they can muster. *It is easier and cheaper for them to prove their case against a handful of software pirates than the pirates' many customers* (unlicensed recipients), with the result being that the latter are largely left to their own devices.

AN UNWANTED ASSOCIATION

Both Nintendo and the IDSA are fond of saying that emulation promotes software piracy, but in truth they have it backwards. *Software piracy promotes emulation, not vice versa.* It is an unfortunate result of its evolution that the emulation industry is so closely associated with software piracy, especially with regards to the videogame industry, because this is a case where the egg came before the chicken. For example, software pirates were dumping console videogame cartridges and coin-op arcade games long before emulators for these products even existed. Zoop, the webmaster of EmuCamp and author of the popular ***MEKA*** Sega Master System emulator, notes that the bulk of the dumps that he worked with during MEKA's development were actually made almost a decade ago by the Image group of computer hackers operating out of Helsinki, Finland. The Orient is an oft-quoted example of the stereotypical software pirate's den; most of the Asian SNES cart dumps that first appeared on the scene originated with Chinese and Taiwanese bootleg game cart companies. At the same time, software pirates and private users alike amassed vast collections of bootleg computer software during the lifetime of the systems in question. Those of us who were on the Commodore and Amiga scene remember "warez" groups like EagleSoft Incorporated, FBR (F--ked Beyond Repair), Fairlight, 2001 Crew, Quartex, and so on. I'm sure there were plenty others for other systems, and I invite you to dig through the memory banks of your local "old-timer" hackers for even more examples. As these older computer systems were discontinued or became obsolete, these software libraries were rendered useless with the passage of time. A lot of people jettisoned theirs and moved on to other systems, but some did not or accepted what others were giving away. The practice has never stopped - today's generation of software pirates are busily duping and dumping away newer forms of computer software, along with arcade and console videogames. *Couple those old software collections with today's base of "warez," and you have a massive ready-to-use software archive for just about any emulation project you care to take on.* How do you get this software to work on your particular system when the technology for which it was designed is dead and gone, or simply tempting? Write an emulator, of course.

This brings us to the current state of affairs so excellently shown by the standoff between Nintendo and the N64 freeware emulator developers, or Sony and the commercial PSX emulator developers. Just as with the computer industry as a whole, the pace of emulation is no longer hobbled by limited resources. *It is now possible to write an emulator for a system that is still economically viable given today's rapid advances in computer technology.* This happened first with *PSEmu* and the Sony PlayStation, and most recently with *UltraHLE* and the Nintendo 64. The computer industry started going through this same situation almost a decade ago, and it can be argued that the same ramifications apply. The average user is not concerned with all of these head games, however. They just want a product that will let them use the software they want to run on the system with which they feel the most comfortable, and many have no qualms as to how and from where they get an emulator and the software to support it. I have yet to see a user that has any real objections to playing Sega Genesis games on a personal computer once they realize that such a feat is possible. I have yet to see Unixoids object to running Amiga software on their systems once it dawns on them that they can. Going back to the Apple world for a moment, *SoftWindows* continues to be a strong niche product for the Macintosh, and there's little that either Microsoft or Apple can or will do about it.

There are four basic ways in which computer software can be protected from intellectual property infringement. One of these will be new to our discussion, while the other three will sound awfully familiar. Each protects the software in different ways, and different means are used to enforce them. *The four shields of software protection (as I term them) are the* ***end user license agreement****, the* ***software patent****, the* ***copyright****, and the* ***corporate trademark****.*

THE END USER LICENSE AGREEMENT

The **end user license agreement**, or **EULA** (pronounced "yew-lah") for short, is a form of legal contract that defines the rights granted by the authors or vendors of a computer program to a user of that program. It takes various forms and can be presented in various ways. For example, one might find a EULA on the package (known as "box-top license" in legal circles), within the user manual, as a separate document inside the package, or as an on-screen display during the setup or bootstrap processes for the program in question. It is usually presented in such a fashion that you have to take notice of it before you use the program in question. Whether or not you actually read it is entirely up to you, but it never hurts and is often to your advantage.

A EULA spells out in exact legal terms what the vendor says you can or cannot do with that copy of the program covered that is in your possession. The language is precise because of its contract nature, and there is a set format and predefined phrases that a EULA must use in order to be legally valid. This is because of the danger presented by certain overzealous vendors, who try to use their EULA to impose restrictions on the user that are unnecessary and/or prohibitive. In these cases, when the terms of a EULA are deemed to be overreaching, then the broader bounds of federal law override the terms of the EULA. Since a EULA is, in a sense, a one-sided contract, in which the user has no say as to the terms that it includes, then federal law dictates both the form and the limits that a EULA can take. It used to be the vendor's responsibility to ensure that the terms of a EULA were not overreaching; however, this is no longer the case.

There has been a recent retreat of sorts with regards to the federal government's ability to override the terms of a EULA; however in the now infamous **Ziedenberg case** (ProCD v. Ziedenberg, 1996), many legal protections with regards to software EULAs were restored. I will not go into the details of the case, as they are publicly available for anybody who wants to look them up. Suffice it to say that the federal government has once again recognized the protective need for EULAs and allowed the pendulum to swing back the other way. *The U.S. Court of Appeals ruled that EULAs are considered valid contracts* *under sections 2-204 and 2-606 of the Uniform Commercial Code (UCC).* This is one reason why all EULAs since then have had specific wording straight along UCC and industry guidelines regarding rights and privileges that a user may be granted with regards to the purchase (or obtaining) and subsequent use of a copyrighted piece of computer software. This language is worded in such a manner as to recognize any and all rights that a user may have under the U.S. Copyright Act or any others that the vendor deigns to grant them, and rarely do they grant any leeway beyond those rights. *A EULA's terms are not considered overreaching once the user accepts any product that the EULA covers, aside from certain clear-cut exceptions spelled out in federal statutory and case law.* There still remains considerable debate in legal circles about the implications of the Ziedenberg case, and there are several excellent works on the subject that you can find on the Internet through the use of your favorite search engine.

THE SOFTWARE PATENT

The issue of **software patents** software will not be dealt with in this document, since they are at present used mainly to protect highly specialized pieces of code that generate unique processes. *The average piece of computer software or your run-of-the-mill videogame does not usually qualify for software patent protection* because of the lack of such processes; however, this may change as patent laws are rewritten to provide extended protection to software owners and vendors whose products are in danger of being pirated. Suffice it to say, based on our earlier discussion of intellectual property law, that patent protect represents the greatest form of intellectual property protection available to computer software, provided that it qualifies for such. Those wishing to pursue this topic should consult those portions of patent law that deal with the new protections afforded to qualifying computer software (see Diamond v. Diehr, 1981 and Stac v. Microsoft, 1995).

THE SOFTWARE COPYRIGHT

Anybody who has had any dealings with a computer-based system knows the value of making a backup of your computer programs. These for the most part take the form of software, of diverse media and formats, which can and is in most cases fairly easy to duplicate. Backing up computer software is a practice is universally recognized and accepted by all parties concerned, and the principle is also incorporated into federal copyright law. *Computer software, regardless of media or format,* ***is protected by copyright law****.* It embodies ideas and concepts, and as such qualifies as a form of expression and is therefore eligible for intellectual property protection. 17 USC 117 is an oft-quoted passage of United States law with regards to the duplication and/or distribution of computer software, and I reproduce it here for convenience.

The form of 17 USC 117 that I give is the current one as amended by the **Digital Millenium Copyright Act** (DMCA) of 1998. Here is how it reads, following the directions of that part of the DMCA known as the **Online Copyright**

Infringement Liability Limitation Act (Title III, Computer Maintenance or Repair Copyright Exemption):

a) Making of additional copy or adaptation by owner of copy

Notwithstanding the provisions of section 106, it is not an infringement for the owner of a copy of a computer program to make or authorize the making of another copy or adaptation of that computer program provided:

- that such a new copy or adaptation is created as an essential step in the utilization of the computer program in conjunction with a machine and that it is used in no other manner, or

- that such new copy or adaptation is for archival purposes only and that all archival copies are destroyed in the event that continued possession of the computer program should cease to be rightful.

b) Lease, sale, or other transfer of additional copy or adaptation

Any exact copies prepared in accordance with the provisions of this section may be leased, sold, or otherwise transferred, along with the copy from which such copies were prepared, only as part of the lease, sale, or other transfer of all rights in the program. Adaptations so prepared may be transferred only with the authorization of the copyright owner.

c) Machine maintenance or repair

Notwithstanding the provisions of section 106, it is not an infringement for the owner or lessee of a machine to make or authorize the making of a copy of a computer program if such copy is made solely by virtue of the activation of a machine that lawfully contains an authorized copy of the computer program, for purposes of maintenance and repair of that machine, if--

- such new copy is used in no other manner and is destroyed immediately after maintenance or repair is completed.

- with respect to any computer program or part thereof that is not necessary for the machine to be activated, such program or part thereof is not accessed or used other than to make such new copy by virtue of the activation of the machine.

d) Definitions.

For purposes of this section--

- the "maintenance" of a machine is the servicing of the machine in order to make it work in accordance with its original specifications and any changes to those specifications authorized for that machine.

- the "repair" of a machine is the restoring of the machine to the state of working in accordance with its original specifications and any changes to those specifications authorized for that machine.

The reason behind the changes and expansion to 17 USC 117 is to address the issues raised by the MAI v. Peak court case of 1993, in which the courts effectively ruled that merely booting up a computer created derivative copies of any software contained within its storage systems that were subsequently copied to system memory. These changes allow those who service computers to work on them and with any of the system's software as necessary without infringing vendor copyrights.

Computer users often get so caught up in the legality of making a backup of their software that they loudly proclaim this right to the exclusion of all else - including the rest the U.S. Copyright Act, the Intellectual Property Rights Act, and the many examples of case law that deal with the copying and archiving of computer software. Let's see what they're missing.

Copyright protection extends to any and all versions of a copyrighted work, including inclusion in compilations or collections, along with derivative works. Copyright owners enjoy the following exclusive rights, which they may exercise at will or license out as they see fit: reproduction, derivation, distribution, and audiovisual performance. They also enjoy the rights of exclusive or hidden authorship, association with or distancing from the work in question, and prevention of damage (physical, moral, social, etc.). It is up to the copyright owner to prosecute any perceived copyright infringement. They must bring charges against the violator within three years of the perceived violation or the case will not be allowed to to go to trial. In certain circumstances that involve willful criminal activity, the limit to prosecute is lengthened to five years.

So what does copyright law permit you as a user to do with the software you intend to use with your emulator? Both federal statutes and applicable case law permit the recipient of a copyrighted piece of software to engage in the following behavior regarding their software. I call them **the ten rights of a software user**.

1. *You may use that software in conjunction with the terms of its EULA*. By accepting the EULA, which you did when you purchased or otherwise obtained that software, you implicitly agreed to all terms that it contains. If you feel that any terms of that EULA may be unnecessarily prohibitive, then it is up to you to prove your case. If any term of the EULA clearly contradicts rights granted under federal law, then they are automatically voided (17 USC 117.a.1; Lasercomb v. Reynolds, 1990; ProCD v. Ziedenberg, 1997; UCC 2-204 and 2-606).
2. *You may make one archival copy of that software in the event that the original fails.* You may use any legally available means at your disposal to make that copy. This includes archiving of software stored in protected formats (17 USC 117.a; Vault v. Quaid, 1988).
3. *You may make an archival copy of the original in a fashion that the copyright owner may not have intended.* Such copying, even if in direct violation of the terms of a EULA, produces what are considered to be noninfringing derivative works and is therefore permissible. This permits the lawful archiving of proprietary or copy-protected software (17 USC 117.a.1; Vault v. Quaid, 1988). This includes both media and format conversion, so long as the copies in question are either limited to noncommercial use (Sony v. Universal, 1984) or made for developmental purposes (Sega v. Accolade, 1992; Sony v. Connectix, 2000). Any form of technology required for lawful archiving is exempt from DMCA restrictions (17 USC 1201.a.1.C, 17 USC 1201.c). Derivative works involving media or format changes that made available for public use or distribution do not qualify as noninfringing works (17 USC 106, Mirage v. Alberquerque Art, 1988; Nintendo v. Computer & Entertainment, 1996).
4. *You may resell the original if you so choose.* Any archival copy you have in your possession must either be included with the sale or promptly destroyed unless the copyright holder permits you to do otherwise (17 USC 117.b specifically mentions transfer by sale).
5. *You may rent or lease the original to a third party.* Any archival copy you have must go with it unless otherwise permitted by the copyright holder (17 USC 117.b, specifically mentions leasing).

6. *You may decompile and disassemble the software in an effort to understand its underlying concepts, which is a necessary and legal step in the process of reverse engineering, provided you own the original and the resultant product is noninfringing.* Such an effort may be undertaken in direct violation of any EULA placed on the software in question (17 USC 1201.f; Vault v. Quaid, 1988; Sega v. Accolade, 1992; Nintendo v. Atari, 1992; Sony v. Connectix, 2000).
7. *You may make limited modifications to a copy of that software in order to get it to work on you system, provided you own the original.* Such a copy may be derivative, but is not considered to be infringing (Vault v. Quaid, 1988; Nintendo v. Galoob, 1990; also, 17 USC 117.1, specifically mentions the existence of adaptations). This also includes media and format changes, so long as access to the original is maintained and the resultant copy is either for noncommercial use (Sony v. Universal, 1984) or for developmental purposes (Sega v. Accolade, 1992; Sony v. Connectix, 2000). Since both of these forms of copy are deemed justifiable fair use under case law, both the production and usage of technology required for these specific situations are exempt from DMCA restrictions (17 USC 1201.c)
8. *You may use that software in a fashion which the copyright owner may not have intended, provided you own the original and such use is considered reasonable* (Narrell v. Freeman, 1989; Nintendo v. Galoob, 1990). 17 USC 117 only says that a program is to be utilized in conjunction with a machine; it does not say how such use is to be defined.
9. *You may make a copy of a piece of computer software on a system that you are servicing as need requires, so long as such copies are destroyed once your service work is finished* (17 USC 117.c-d). This arose from a long-standing legal quandary regarding one form of unintentional derivative work (MAI v. Peak, 1993.
10. *You may run that software on an emulator* (the *A-Max* affair, 1989; Sony v. Bleem LLC, 1999). This practice is deemed justifiable fair use under 17 USC 107. Neither the original system vendor nor the vendor of the software in question has the legal right to prevent you from using that software on an emulator, unless said emulator or software is of an infringing nature (Sony v. Connectix, 2000).

The key to all of these rights is that *you must have a legitimate copy of the software in question (i.e. an original) before you are entitled to these rights.* If anything you want to do with the software you intend to use with your emulator is not covered by the above, then it is illegal under current copyright and case law. If you do not have legitimately obtained originals in your possession, then you are not entitled to any of these rights and any activity you may undertake with any unauthorized copies you may have is illegal. Period.

<u>THE CORPORATE TRADEMARK</u>

Any piece of computer software that contains a trademark or generates a trademark on its host system is further protected under the Landham Act, which governs the illegal usage and duplication of trademarks. The key parts of the Landham Act most often quoted to combat software piracy in this regard are 15 USC 1114-1116, which deals with the concept of **trademark infringement**.

If you will recall from our earlier discussions, trademarks are a form of intellectual property protection. Their primary purpose is to protect the "image" of a product or business. They can take any number of forms, ranging from specialized words to custom graphics, so long as the trademark in question is undeniably unique. Once a trademark is registered, then its owner can do anything it wants with it. Use of a registered trademark is denied to everybody but the owner, except in cases of fair use when use of the trademark cannot be avoided. This is what the Landham Act means by *innocent infringer* - for example, if you are writing a news article on the Sony PlayStation and need to show a picture of the PSX logo, then you can provided you acknowledge the logo belongs to Sony. That's why you see the "circle R" or the letters TM after a trademarked word or graphic, or see a notice of trademark ("X and Y are trademarks of Z corporation" is a common form).

It is up to the owner to prove trademark infringement; likewise, it is up to the owner to maintain rights to that trademarks or it will be allowed to pass back into the public domain. For example, almost everyone in the videogame industry agrees that Nintendo tends to be overly protective of its trademarks, but they would rather be overly protective that have them subject to charges of abandonment. Can you imagine what their first top-selling game for the Nintendo 64 would have been like if the character of Mario was not trademarked? Super Luigi 64? Super Peach 64? Super Bowser 64? Somehow, it just wouldn't be the same.

Case law has determined that trademarks contained within a piece of computer code are not protected under the law; however, any trademark that the code might generate on its own or cause to be generated upon the display of a system on which it is running are protected (Playboy v. Frena, 1993). There are **two forms that trademark infringement can take** with regards to computer software: **unauthorized display** of a trademark, and **unauthorized alteration and/or duplication** of a trademark. There are many pieces of software that do either one or the other, or both.

The reason why Accolade eventually settled with Sega in their unlicensed software dispute (Sega v. Accolade, 1992) was that Accolade's unlicensed games clearly activated Sega's TradeMark Security System (TMSS), causing a TMSS display to be generated on a Genesis/MegaDrive display whenever one of Accolade's licensed games booted up on later model consoles. Accolade was forced to concede the point and retool their products in such a manner as to avoid tripping the TMSS; otherwise, they would have been in direct violation of trademark law. This is why many Accolade titles for the Genesis/MegaDrive will start with an Accolade logo instead of the TMSS display. While this may seem like a small matter to a user, trademark infringement is often one of the most overlooked issues with regards to emulator development and support, and one that will catch a developer every time unless they are careful. Remember, it is lawful to display an unmodified trademark contained within a program running under emulation (Sega v. Accolade, 1992); however, the emulator itself may neither generate a trademark from the original console's internal code of its own volition nor alter any trademarks contained in any programs that it runs. For example, it should be noted that every single Sega console emulator to date does not generate a TMSS display, whereas they pass through without modification any displays contained within the programs that they run. Why? Because the emulator authors have wisely chosen not to include the TMSS as part of their emulation. It is an unnecessary function to proper emulation of the console, therefore it would be an infringing act to include TMSS emulation. That is the same reason why developers of some of the newer Nintendo console emulators are going to such pains to avoid displaying the Nintendo logo that is generated by the console itself whenever a game is booted. Such would be a prosecutable infringement of a Nintendo trademark (in this case, the Nintendo logo).

So does that mean that you can patch a copy of a piece of computer software in order to prevent a trademark from being displayed? Absolutely not. *Remember, trademarks that appear within the body of the actual program code are not protected by trademark law, but any trademark that is generated on a system display by whatever means for whatever purpose is protected* (Playboy v. Frena, 1993). It would be illegal for somebody to develop a patch for a piece of computer software that would alter any displayed trademark in any fashion. Your only option is to talk the emulator developer into finding a way not to display the trademark, but even that might be successfully challenged in court, since the trademark was intended to be displayed in the first place. This is one of the many reasons why certain types of software patches for commercial software titles are illegal - they patch a character or object in the game which is trademarked. Need an example? How about all of those hack patches for the various Mario "ROMs" that do all sorts of things to him - change size, mutations, new look, or even replace him with a new set of graphics? Every one of those unauthorized patches violates Nintendo's trademark on the Mario character, and their history is to prosecute any and all such cases of trademark infringement. This might give pause to anybody wanting to hack their favorite Mario "ROM."

THE NOTION OF FAIR USE

There is one facet of intellectual property law that deserves mention at this point. It can be used to both defend intellectual property rights or to void them - most commonly the latter. It is a concept that has been bandied about so much as a defense by the emulation community that I often wonder if some of them really know what it means. "So what?" you hear them say. "None of these legalisms matter. I don't care about all that. I'm not hurting anybody. I'm just exercising my **fair use** rights. Fair use protects anything I might want to or could do with MY software." Does it? Just what exactly is this notion of fair use? Before you can justify a claim of fair use, you need to understand how the law defines this particular concept, and exactly how that applies to you as the owner of a piece of computer software regardless of its origins.

I should note at this time that the concept of fair use also covers trademarks to a certain extent, but I shall deal specifically for now with copyrights. The fair use claims for copyrighted material tend to be the more prevalent and onerous with regards to computer software. I shall deal with the fair use of trademarks in a later discussion.

The concept of fair use was created over time as the courts realized that there are certain cases where limited reproduction of a copyrighted work often has greater social value than absolute ironclad control by the copyright owner. This is true in such areas as the advancement of the arts and sciences (artwork "prints," scientific collaboration on projects), news reporting (journalistic privilege), criticism (the use of quotations), and "other endeavors of educational or social usefulness" such as teaching, scholarship, and research. The fair use clause first made its its appearance in the 1978 revision of the U.S. Copyright Act, and the current form (17 USC 107) reads as follows:

> Notwithstanding the provisions of sections 106 and 106A, the fair use of a copyrighted work, including such use by reproduction in copies or phonorecords or by any other means specified by that section, for purposes such as criticism, comment, news reporting, teaching (including multiple copies for classroom use), scholarship, or research, is not an infringement of copyright.
>
> In determining whether the use made of a work in any particular case is a fair use, the factors to be considered shall include --
>
> 1. the purpose and character of the use, including whether such use is of a commercial nature or is for non-profit educational purposes.
> 2. the nature of a copyrighted work.
> 3. the amount and substantiality of the portion(s) used in relation to the copyrighted work as a whole.
> 4. the effect of the use upon the potential market for or value of the copyrighted work.
>
> The fact that a work is unpublished shall not itself bar a finding of fair use if such finding is made upon consideration of all the above factors.

What this means is that the concept of fair use can be applied to anything that is copyrighted, including unreleased works - since they are protected by implied copyright. *In the case of the emulation community and the computer software industry, the principle of fair use also covers* ***unreleased games, alpha and beta copies, prototype games,*** *and* **playable demos** (Franklin v. Franklin, 1971).

Fair use disputes are decided on a case-by-case basis, and any court hearing such a case must weigh the following four factors as defined in 17 USC 107:

1. the purpose and character of the use, including whether such use is of a commercial nature or is for non-profit educational purposes.
2. the nature of a copyrighted work.
3. the amount and substantiality of the portion(s) used in relation to the copyrighted work as a whole.
4. the effect of the use upon the potential market for or value of the copyrighted work.

All four factors must be weighed equally before one can make a claim of fair use. Likewise, the courts must weigh all four factors equally before ruling on the legality of such a claim.

The only exception to the fair use proviso of copyright law is the "**safe harbor**" concept, which is almost exclusively reserved for non-commercial libraries and other similar archival organizations needing to procure or reproduce single copies of a copyrighted work. Those who want to claim "safe harbor" protection must prove the following three things:

- any reproductions of a copyrighted work are not made for direct or indirect commercial advantage.
- any archival service making the claim is either open to the public or open to all researchers in a specialized field.
- any reproductions must include a copyright notice (17 USC 108) at the point of distribution and with any copies made.

The safe harbor concept was recently enacted into law in 1998 as part of the DMCA (17 USC 1201.d).

So how does fair use relate to computer software? *The first thing you need to realize is that those programs that you may have obtained by various means **are not your software**.* You didn't get the intellectual property rights along with the program; you only got a copy of the program to use on your computer. Every commercially vended piece of software comes with some kind of EULA limiting the uses to which its owner can put it. EULAs tend to vary somewhat from program to program and across platforms, but the general gist is that you have a license from the program vendor to use that piece of software on your system and to do anything else with permissible under the appropriate federal laws and whatever additional rights that the vendor may grant you. You have to agree to the EULA before you can legally use that software on your system. Merely accepting the program implies EULA agreement, which also independently confirms recognition of federal law and regulations regarding the proper uses to which computer software can be put. Any use to which you put that piece of software that is not defined in the EULA or acceptable under federal law (which may supersede the EULA in certain clear-cut cases) does not qualify as a fair use.

Second, the concept of fair use regarding computer software is quite restrictive. The law firm of George, Donaldson, and Ford have come up with an effective rule-of-thumb concerning fair use in this regard:

> "If I depended on this work to feed my family and pay my taxes, [then] would I feel like I ought to be paid for [the] use that I propose to make of this work?" If the answer to that question is "yes," then you may question whether [or not] your use is fair.

Fair use concerns are strictly limited to matters of the public interest. That is why reporters and journalists are allowed to employ questionable sources and use questionable materials in their reporting; both the law and the courts have recognized the need, if not necessarily the desirability, of "journalistic privilege." Remember this the next time you read one of those sleazy London tabloids or flip the TV over to the *Drudge Report* on Fox News. One might argue that the wanton duplication of commercial computer software serves the public interest, but that is not quite true - it only serves the interest of that portion of the public who is willing to obtain those products without paying for them. This does not comprise the majority of software users, as most industry experts will readily agree. *The courts have ruled time and again that* ***you have to own an authorized copy of the software in question*** *before you can claim the privilege of fair use to excuse any action beyond that permitted by a EULA or copyright law.* Therefore, software piracy cannot and does not qualify for fair use regardless of how the argument is presented.

Let's see how each of the four points of the fair use test apply to someone who is making more than one copy of a piece of computer software with intent to distribute. This automatically puts such action in the realm of commercial use (Sony v. Universal, 1984). This could be the kid down the street duping copies of his favorite videogames for his friends, or the team of software pirates running cart dumpers 'round the clock, or an Internet site where "ROMz" and other such "warez" are posted for download by any and all who just happen to drop by and find them. I will reserve discussion of the "safe harbor" exception until such time as we deal with Internet related issues.

> *The purpose and character of the use, including whether such use is of a commercial nature or is for non-profit educational purposes.*
>
> The first portion of the fair use test is considered to be the most important with regards to the public interest. The case most frequently cited as setting the precedent is Williams and Wilkins v. United States, 1973 (aka "the National Library of Medicine" case). In this case, the purpose and character of use is to provide a copy of a commercial piece of software to someone who never intended to pay for the original. This can in no way be construed as being for "educational purposes." This is a clear-cut case of copyright infringement both on the part of the distributor and the recipient.

The distributor must have a license of some kind from the copyright holder to distribute multiple copies of that software regardless of intent (AGU v. Texaco, 1994). In addition, the recipient is bound by copyright law only to use those copies of software that were legally obtained (Nintendo v. Atari, 1992). As such, this situation qualifies as a unquestionable and deliberate violation of copyright law. The failure of this point becomes even more obvious if some kind of financial transaction is involved for the copy itself. Many software pirates charge for their "warez," and this represents a blatant violation of federal copyright law (17 USC 106).

The nature of a copyrighted work.

The second portion of the fair use test is considered to be the most important with regards to the original intent of the author. Computer software is eligible for copyright because it is a work of expression (Harper & Row v. Nation Enterprises, 1985). In particular, videogames are afforded copyright protection because they are escapist fare and are considered by the courts in the same light as other escapist fare such as books, magazines, and movies (Playboy v. Frena, 1993). Since escapist fare by definition is protected by law from duplication, then duplication of any commercial videogame is a clear violation of copyright law. To broaden the scope again, any duplication of a work of expression without proper license is also a violation of copyright law, and making multiple copies of a piece of computer software qualifies as such a violation.

The amount and substantiality of the portion(s) used in relation to the copyrighted work as a whole.

The third portion of the fair use test is the most important when considering the question of derivative works or inclusions within compilations and collections. Copyright law grants you a limited license to make one and one copy of a piece of legitimately obtained software for archival purposes. If you want to make more than one copy, then you have to get a license for multiple copies from the copyright holder. Wanton duplication of a piece of computer software involves the entire piece of software, and as such cannot conceivably qualify as any kind of limited "amount" or "substantial portion." As such, it represents a clear violation of copyright law (AGU v. Texaco, 1994).

The effect of the use upon the potential market for or value of the copyrighted work.

> This last portion of the fair use test is considered by legal experts to be the most important one of all (*Nimmer on Copyright)*, and has been used to defeat many a contention of fair use. We briefly touched upon this back in our discussion regarding the legality of creating and then releasing an independently developed software-based emulator. Hardware cannot be copyrighted, but software can be and frequently is copyrighted. The presence of multiple copies of a program for which the owner did not pay diminishes the revenue that the copyright holder would receive. As such, unlicensed duplication represents a viable threat to the potential market for such a product, and could also force the owner to change the price in order to increase sales and thereby make up for the loss in potential revenue. (Harper & Row v. Nation Enterprises, 1985; Lotus v. Borland, 1990; Playboy v. Frena, 1993). Thus, unlicensed duplication of computer software is not justified by the fair use exception.

So does making multiple unauthorized copies of a piece of computer software for any Tom, Dick, or Harry who wants one qualify as fair use? No, it does not. It fails the fair use test on every single point. *Unlicensed duplication of copyrighted computer software for commercial purposes is no more protected by fair use than someone who chooses to photocopy an entire book at the local print shop instead of paying for it* (Atari v. JS&A Group, 1983). Remember this the next time you hear someone loudly proclaiming fair use on their "ROM"-laden Internet emulation site.

For those who would like to learn more about copyrights, fair use and their ramifications in the computer age, and are willing to delve deep into legal philosophy, I would recommend reading the works of **Raymond Nimmer**, the noted legal scholar. His many dissertations on the subject are required reading in the legal field, and more than one federal court (including the U.S. Supreme Court) has relied heavily on his insights. I have briefly alluded to his landmark tome *Nimmer on Copyright*, which is considered the definitive work on the field, but he has authored or assisted in many other. References to his works can be found with your favorite search engine all over the Internet, and you can currently find his home page at the Internet site of the University of Houston Law Center (http://www.law.uh.edu/faculty/RNimmer/RNimmer.html).

INTROSPECTION

So what does all of this mean? **It means that the only kind of computer software that you can use with an emulator is that which is legitimate in nature - just as if you were using it with the actual system.** This means using either the originals or some other version of that software either permitted under the strict limitations of federal law or specifically authorized by the copyright owners involved. You have the right to archive that software, so long as you produce one and only one unadulterated copy unless otherwise approved by the copyright holder or permitted within the constraints of federal law. You do have a limited right to alter or modify the program so that it will function in exactly the same fashion and generate the same on-screen displays as the original, provided you own the original beforehand and such modifications do not excessively infringe upon the rights of the copyright owner or trademark holder. Given all of this, it is perfectly legal for you to build and maintain a software base for use with your emulator.

REVIEW QUESTIONS

1. Who are the three parties involved in the "emulation war?" What roles do they play? How can each affect the outcome?

2. What two broad principles of vendor-user economics are briefly mentioned? Give examples of these principles in action. Do you believe these to be valid? Why or why not?

3. Why is the emulation community so inexorably linked to the bootleg software community? Why would this concern an original system vendor or licensed developer? How would this affect an unscrupulous user?

4. What is software piracy?

5. How does the law determine a legitimate claim of software piracy?

6. What organizations exist within the computer industry to battle software piracy?

7. How does software piracy impact the market for computer software? Describe the simple "vicious cycle" model of copying. Do you agree with the contention that it is unrealistic? Why?

8. Who will the copyright owner be more likely to prosecute with regards to software piracy, the distributor or the recipient? Why?

9. What are you buying when you obtain a piece of commercial software? What stipulates the terms by which your purchase is governed?

10. When can federal law override a vendor's desire for intellectual property exclusivity?

11. To what rights is the copyright owner entitled regarding a piece of commercial computer software? In what ways could the owner use these rights to prosecute someone infringing on the copyright?

12. To what rights is the user entitled regarding a piece of commercial computer software? Explain how these rights are justified by federal statute and/or case law.

13. How does trademark infringement become involved in software piracy disputes? Describe the two key areas that may be involved in such a dispute.

14. What is the origin of the concept of fair use? What two tenets govern the fair use of computer software?

15. What is the sole exception one can claim to abuse of the fair use principle? What are the hurdles one must overcome to make such a claim?

16. What are the four parts to the fair use test?

17. As far as the public interest is concerned, which is the most important part of the fair use test? Why?

18. As far as the copyright holder is concerned, which is the most important part of the fair use test? Why?

19. Give an example of how you would use the fair use test with regards to a specific situation concerning computer software. Do not use the example cited in the text; instead, come up with one of your own. Be sure to justify your claims.

20. Describe some real world attitudes and practices with regards to software piracy, the legality of these beliefs, and whether or not these could be successfully prosecuted.

<u>THOUGHTS TO PONDER</u>

1. If you knew somebody that was violating the copyright on a piece of computer software, no matter how small the violation, would you do something about it? Should you do something about it? Why or why not?

2. If you were an emulator developer or software author, how would you devise a means of bypassing any code that might trigger or alter a trademark display? Would your method(s) be considered legal? Why or why not?

3. Assuming that you cannot get access to the original system's software base, what kinds of software are legal to use with an emulator?

4. Can you pirate an emulator?

Module Two: The Software

Part 2 - Altering the Software Base

Changing the format ... from the original cartridge to a disk format also violates Nintendo's rights to prepare derivative works under 17 USC 106.2.

Nintendo v. Computer & Entertainment, No. 96-0187 (W.D. CA, 1996)

A BLACK MARKET EXPERIENCE

Inside the confines of a warehouse located somewhere within the tightly packed mass of buildings that make up the waterfront district of Hong Kong, sometime around the early 1990s, an extraordinary event is occurring within an otherwise ordinary place. It looks just just like what one might expect of such a place, both inside and out - boxes and crates, pallets stacked full of goods being moved around by intent-looking drivers on forklifts, the occasional foreman shouting out directions (albeit in Chinese), and the floor manager hiding out within the comfortable confines of his air-conditioned office. Not far from him, though, in another air-conditioned room inside the warehouse, a quite different set of activities from those on the warehouse floor are taking place. The two, to a casual eye, might seem unconnected at first, unless one were able to stay around long enough to see the process all the way through. Let's sneak inside for a peak, shall we?

The room is wall-to-wall electronics, with some dozen or so people doing several different things at once. In one corner, we see a tall pile of oddly shaped black boxes, each about the size of a pack of cigarettes. There are several different kinds, and each has different labels applied to them. They are game cartridges for use with many of the popular videogame consoles of the day. The gentlemen closest to them are pulling them from the piles, one at a time, and slapping them inside an odd-looking contraption attached to each the dilapidated personal computers sitting on their desks. There is a flash of activity on the screen for about half-a-minute or so, then the cart gets yanked and tossed into a steadily growing pile building inside an old lidless wooden crate. Meanwhile, across the room, the same process is happening in reverse, but in a somewhat expanded form. There are many more people here, and many more computers with different-looking little devices attached to them. This time, the pile of game cartridges have no labels, but they are going through the same process as the others. One at a time, they are taken from their stacks and shoved into the weird-looking machines. The appropriate computer hums for a bit, then spits a message back out at its operator. Immediately, the cartridge is pulled off and handed to another person, who hurriedly but neatly slaps on a label and then packs it inside a box along with its brothers. Intrigued, we manage to get a closer look at the boxes that are being so carefully filled. The markings are the same as those on the large crates outside on the workroom floor.

Carefully, so as not to be noticed, we manage to ease our way back out into the main work area of the warehouse, grabbing a hard hat and a pull bar in the process. Walking nonchalantly so as not to attract attention, we eventually make our way to the far side of the warehouse, behind a large stack of crates that prevent us from being seen by the others. Scanning down a row of smaller crates, we find one with those unusual markings that we noted earlier. Using the pull bar as quietly as we can, we manage to prize the lid off of the crate and peer inside. It is full of hundreds of videogame cartridges - ones that must have been made in that electronics-filled room we were inside earlier. Carefully, without attracting attention, we manage to re-secure the lid and replace the crate in its original location, then make a quick yet deliberate exit though the loading doors. As we walk down the loading dock and away from the place, we note several truckloads of these crates lined up alongside, each almost ready to roll. Each manifest has a different destination printed on it ... Taiwan ... Japan ... Korea ... Malaysia ... Singapore ... Guam ... Hawaii ... the continental United States ... Mexico ... Brazil ... England ... Germany ... France ... and so on. And as we slip back into the shadows and make good our escape, the implication hits us - what we saw is going on every day, around the clock, and affecting every part of the world. Thousands, tens of thousands, perhaps even millions of illegally produced videogame cartridges. The wind blows cold, and we shiver. How many other buildings that we pass by have similar operations inside them?

As you may recall, the bulk of the software base available to videogame emulator developers and users arose during the actual lifetime of the machines in question, during which unlicensed cart manufacturers turned out counterfeit products by the truckload. It hasn't stopped, as anybody in the software community will tell you - it still goes on. Every copy these folks make is illegal in both origin and purpose, resulting in what the law terms as **counterfeit copies**, but this matters little to them. Not having to pay license fees to the real owners means more money in an unlicensed distributor's pockets, and that is why software piracy on a commercial level, which is often referred to by the quaint term of **bootlegging**, is such a tempting proposition. Both the technologies involved and the resultant software remain with us, and such activities continue in many parts of the world even as we speak. China is notorious as the bootleg capital of the world due to its lax interpretation of already weak local intellectual property laws, but they are not the only ones that are guilty. Russia is another well known alternate source, as are other former Soviet states and any other country with a strong underground "black market." The practices and equipment that videogame bootleggers have at their disposal are often the same as or based upon those used by bona fide developers. They remain as popular and available as they ever were, and thanks to the Internet more and more users are becoming aware of their existence and potential. Both the resultant copies of the software made from any original and copies of the bootleg forms, both of which were produced with this technology, are in many cases available to anyone with the determination and resolve to find them, no matter how long and where they must search. In my youth, we called it *going for the "warez."* Today's generation of unscrupulous gamers call it something else entirely.

With this in mind, we shall now take a look at certain specific issues that deal with the emulation software base. We shall deal with the concept of a "ROM," whether or not a "ROM" is legal, and what can be legally done with a "ROM." We shall also finally address the issue of BIOS dumping, and we shall see whether or not there is such a thing as public domain software for an emulator. At the end, I shall sum up the specific legal issues that are in doubt with regards to the emulation software base.

WHAT IS A "ROM?"

If you start surfing the multitude of emulation and "warez" site on the Internet, you will run into a term familiar to most computer users but employed in a new and rather unique way. That term is **"ROM,"** written with quotes here to distinguish it from the traditional definition of Read-Only Memory. "ROM" is emulation slang for a piece of software that is used with an emulator, but it is now actually a bit more involved than that - thanks to the activities of the software pirates, who tend to use the term "ROMz" rather loosely.

To formalize the slang use of the word in its widest sense, *a "ROM" is a piece of computer software stored within a unique archival format not found in everyday personal computer usage that is intended to work with special program development tools, or some form of an emulator, or both.* There are to date two distinct types of "ROM" - the **binary image file** and the **disk image file**. The common form of the binary image file as of today is the cart dump; this term arises from the use of special hardware to literally "dump" the computer code stored within a ROM-based delivery system like those used in your typical videogame cartridge. This is where the slang use of "ROM" first originated. The disk image file arose from the need to support software storage formats that are for obsolete or niche systems. The main reasons for doing so involves either the slowness or obsolescence of the original storage device (being able to use the data on more modern devices is faster and more convenient), or the copy protection techniques for certain forms of software (which are commonly disk-based). The disk image file is a one-pass "dump" of the entire contents of an original disk stored as a single continuous file, which appropriate routines inside an emulator can then treat as if it were the actual disk. Disk image files are usually of older floppy disk formats, but it is conceivable that this practice will be applied to other forms of removable storage media. It is almost certain that the image files generated by today's CD copiers and recorders could very well become the "ROMz" of tomorrow.

We are not here to debate the legality of using a "ROM" with an emulator. We have already established in our prior discussion that it is perfectly legal to run legitimately obtained software under emulation. The question that immediately comes to mind is whether or not "ROMs" are legitimate in themselves. This brings up two related issues that have a direct bearing on the question, and we shall deal with each in turn before answering that question: **the practice of ROM dumping**, and **the theory of ROM backups**.

The following discussions will deal primarily with those "ROMs" that originated on some form of **permanent storage media**. Practically all forms of regular computer software are stored on recordable storage media, with the most popular forms at this time being the ubiquitous floppy disk and the ever-increasing usage of CD-R (i.e. the write-once form of the CD-ROM).   As for "ROMs" originating in permanent storage media, videogame cartridges for handheld and tabletop consoles come immediately to mind; however, the same principles apply to dumps of the integrated circuits and other forms of hardware-based storage media used in coin-op arcade videogames, since "ROMs" for these standalone systems are also quite widespread. It should also be noted that the videogame industry is currently attempting to redefine the meaning of the term *permanent storage media* so that it will apply to recordable storage media intended for long-term storage, such as CD-ROMs and DVD-ROMs (see the **Digital Media Recording Act** of 1995). There is an ongoing fight in the courts between proprietary minded vendors and freedom of media advocates over this issue, and it will doubtless continue right along with the industries that require both permanent and recordable storage media. Insofar as computer based systems are concerned, original system vendors will continue to develop custom storage formats and new forms of copy protection in a concerted effort to prevent any kind of duplication regardless of intent. As a result, all of what we are about to discuss here will also apply to these new iterationss of "ROM."

THE PRACTICE OF ROM DUMPING

ROM dumping first originated with program developers who wanted to examine the computer microcode stored within integrated circuits. You see, actual ROMs come in two flavors - programmable read-only memory (PROM) and erasable-programmable read-only memory (EPROM). PROMs are the most common - you "burn" in the code once and then forget about it. EPROMs, on the other hand, can be erased and "re-burned" with new code (for example, most BIOS chips are EPROMs). Regardless of which form you use, there is no way to get at the machine data code stored within the ROM once it is burned into those little black rectangles of silica and metal unless you come up with a way to download it from the chip itself. Downloading in this case refers to some kind of special interface linked to the ROM itself whereby its internal code can then be "dumped." The result is object code in software instead of hardware, which is easier by far for a developer to work with. This may sound a little convoluted to you non-technical types out there, but there is a common example. A **cart dumper** is a special kind of ROM dumper designed to work with the ROMs found in videogame cartridges. You plug in the videogame cart, and the cart dumper spits out the game's object code in the form of a binary dump file - which is how the emulation slang use of "ROM" came about. Cart dumpers are but one form of ROM dumping, but there are many more. How do you think all of those arcade "ROMs" are being produced? ROM dumping, of course. The resultant "ROM" is almost always a perfect copy of the game's complete object code. The only problems coming from this procedure are either due to incomplete dumps or special hardware installed by the original vendor to thwart ROM dumping, and the latter is usually the result of antipiracy systems for console and arcade videogames developed for exactly this contingency.

There is a legitimate basis for ROM dumping, though, and it is best to address this now. Videogame vendors often provided pre-upload copies of their software to licensed potential developers for their own use, but there were more than a few who did not want to be subject to the whims of a dictatorial vendor. They wanted to know how certain games worked to produce the effects that they did on-screen, not necessarily with the original vendor's consent, so they could either duplicate or adapt those ideas into their own titles. The only way to do this was without the vendor's permission, and the only way to find out these ideas was to dump the game(s) in question and analyze the actual machine code, which would often help once they began the actual process of reverse engineering the concepts for use with their products. Having a dump of the original game served a multitude of purposes, chief of which was to ensure that the object code form of the reverse engineered code bore as little resemblance as possible to the original object code. This practice is almost as old as the computer industry itself (IBM v. Compaq, 1982) but is not just limited to old-fashioned console videogames. Everybody who has dipped into the arcade and personal computer videogame markets is aware of what happens when a new and successful title is released - it gets cloned beyond belief. It is safe to say that those vendors who produce clone products that do not use licensed code have spent at least part of their time reverse engineering the original game.

Prior to 28 October 1998, ROM dumping hardware in and of itself was not illegal for the average user to purchase and own. If it had been, then any piece of equipment that could make a copy of anything would have been illegal. Why? The original argument went something like this. Most folks own an audiocassette deck or one of those fancy audio CD recorders. Just because they can use them to make copies of their favorite albums doesn't necessarily mean that they will. Likewise, just because you have floppy disk drives or a CD recorder in your computer doesn't necessarily make you a software pirate. To use another example, this time from the auto industry, owning a police radar detector is not in itself illegal. It is the use to which you put it and in what part of the country that you use it that makes it illegal. The ultimate responsibility for the use of these and other such kinds of devices rests upon the user, and not with the manufacturer or vendor. To repeat an oft-quoted expression that is commonly heard from the gun rights lobby, "Guns don't kill people - people kill people." It is not the fault of the product or its design; it is the use to which that product is put. *Owning a cart dumper or other such piece of equipment does not make someone a software pirate. It is what they do with it that makes them one.* A lot of average users picked up on this defense for ROM dumping hardware and began purchasing the equipment on their own, provided they had deep enough pockets and could find somebody who sold it . As for the videogame bootleggers, they cared little about justifying themselves. You already know how they put both their ROM dumpers and ROM burners to use.

Original software vendors have never approved of the practice of ROM dumping due to its bootlegging implications. They treated copies of their software produced by ROM dumping just as they would any illegal duplication of a copyrighted work. Any unauthorized copies of their computer software were considered to be **counterfeit copies** of those products, and the illegal production and distribution of such copies was thus punishable under federal computer fraud statutes (15 USC 1127, 18 USC 1030). These are the same statutes used to deal with counterfeit copies of software for regular computer systems, and the best known and most notorious example of the invoking of these statutes was the FBI/Secret Service joint exercise Operation Sun Devil in 1990. Unfortunately, despite their best efforts, which at least managed to make the practice of videogame cartridge dumping unjustifiable by the average user under the backup clause of copyright law, (Atari v. JS&A Group, 1983), there was little decisive legal action in this regard until 1992, when two important court cases happened within the same year. *These two cases (Sega v. Accolade and Nintendo v. Atari) legalized the practice of ROM dumping by recognizing it as a legitimate practice of videogame developers for the process of developing their products.* In other words, any developers who saw an unusual or intriguing aspect or feature of a videogame that they had not developed had the legal right to go out and buy a copy of that game, dump or decompile the game's object code (regardless of original storage format or media), and then attempt to reverse engineer those functions that attracted their attention in the first place. Insofar as the home videogame consoles of the day were concerned, this meant obtaining cart dumpers or other similar devices so they could download the game's object code from the cartridge ROMs. The resultant copy was ruled by the courts to be an **intermediate copy**, since a change in format from the original delivery system had taken place. Since the developers had obtained an original copy of the title in question by legal means, and the change in format was necessary for the efforts of the developers in question, any intermediate copies that they produced as part of the reverse engineering process were therefore deemed to be noninfringing. The practice of reverse engineering computer programs, regardless of original storage media, by this and other such techniques has recently become embodied into federal law with the passage of the Digital Millenium Copyright Act in 1998 (17 USC 1201.f).

It is important to note two things in regard to the legalization of ROM dumping. First, *an original has to be involved at all times*. If a group of developers dump one or more ROMs associated with a particular program, then the original must be maintained so long as the resultant intermediate copy remains in their possession - just as if a legitimate backup copy had been produced. Second, even though a change in format from the original delivery system has taken place, *intermediate copies are still protected under copyright law as if they are the actual originals.* The latter means that intermediate copies cannot be freely distributed, since such an act would void a developer's special protections under case law and make them eligible for prosecution by the usual means. They may distribute copies within their immediate circle of associates, but not beyond - not even for review purposes by outside parties - without the consent of the original vendor.

Needless to say, the fact that ROM dumping had now been legitimized, even within the supposedly strict constraints of case law, did not sit well with system vendors. In particular, Nintendo began doing anything it could to discourage the practice. It still held that any kind of ROM dump was an unauthorized infringing copy of its proprietary software (MAI v. Peak, 1993) and therefore still illegal under copyright law. While they could no longer touch software developers and their associates in this regard, they were going to make absolutely sure that the camel's nose didn't get any farther under the edge of the tent with regards to the user base. *Many different approaches were tried, with varying degrees of success, but the one on which they and other software developers eventually settled after five years of effort was the EULA.* The Ziedenberg decision (ProCD v. Ziedenberg, 1996) had restored the vendor's rights to include certain forms of restrictive language within their software EULAs under the terms of the Uniform Commercial Code (UCC 2-204 and 2-606), and computer software produced in the post-Ziedenberg era soon included EULA clauses similar to that reproduced below. This is the form of Nintendo's standard EULA included with my nephew's *STAR WARS: Rogue Squadron* N64 videogame cartridge, and I reprint it in its entirety straight from page 36 of the owner's manual. Even though it was printed in 1998, it is typical of most post-Ziedenberg EULAs imposed upon the purchasers of a videogame cartridge:

> **IMPORTANT (REV B)**
>
> WARNING: Copying of any Nintendo game (including this game) is illegal and is strictly prohibited by domestic and international copyright laws. "Back-up" or "archival" copies are not authorized and are not necessary to protect your software. Violators will be prosecuted.
>
> This game is not designed for use with any unauthorized copying device. Use of any such device will invalidate your product warranty. Nintendo, LucasArts (and/or any Nintendo licensee or distributor) are not responsible for any damage or loss caused by the use of any such device. If use of such device causes your game to stop operating, disconnect the device carefully to avoid damage and resume normal game play. If you game ceases to operate and you have no device attached to it, please contact your local authorized Nintendo retailer.
>
> The contents of this notice do not interfere with your statutory rights.
>
> This manual and other printed matter accompanying this game are protected by domestic and international copyright laws.
>
> The rental of this game without permission of Nintendo or its licensees is strictly prohibited.

Such EULA language did not sit well with many console videogame users, who were by now quite active on the Internet. They argued that archiving their videogame software for use with the new crop of videogame emulators was perfectly legal under copyright law (17 USC 117), since there was no other way to make them work with the emulator. Such protections were claimed either for developmental or operational adaptive purposes, even though it was users and not developers making these claims. Since the price of cart dumping hardware was now within the reach of many users, they bought cart dumpers and began dumping their collections in order to use their games with the now widely available videogame console emulators. In addition, there was a mad rush to dump the ROMs of arcade games and newer videogame consoles, even though emulators for many of these systems did not yet exist. The bulk of videogamers, though, not having deep pockets, used the same argument to justify their downloading of "ROMs" for games that they owned which could be found on various backwater sites on the Internet. These "ROMs," for the most part, had been put on-line as a so-called "service" by other users who had them in their possession - either obtained from bootleg channels or dumped themselves with their own ROM dumpers. Users would also download the requisite emulators for the same reason - it was more convenient to use the dumped copies and an emulator than it was to mess with the actual hardware. This brings us to one of the most novel arguments in existence for the continued practice of ROM dumping and subsequent distribution of the resultant dumps - *the theory of ROM backups*.

THE THEORY OF ROM BACKUPS

A common argument you will hear from the emulation community goes something like this: "It's okay for me to back up my game carts. Copyright law says so." You will also hear this variation: "I download 'ROMz' because I want backups for my real carts. It's okay, because backups are legal." Another one you might here is this: "Second copies of 'ROMs' are legal because they are backups of legal developer copies." Are 'ROMs' legal? Is it legal for you to backup a piece of software that is contained within some form of permanent storage media? Its it legal for you to backup a 'ROM?' While the actual reference is to the "ROMs" generated from a videogame cartridge, the same argument is also used to justify ROM dumps of arcade videogames, so they also have to be dealt with in this regard. Let's see what the law and the courts have to say about the subject.

It is legal for you to make a backup copy of a piece of computer software that you have obtained through appropriate means, and emulation sites are often fond of quoting paragraphs (a) and (b) of 17 USC 117 on that subject. Let's refresh our memory and see what the law as we know it says:

a) Making of additional copy or adaptation by owner of copy

> Notwithstanding the provisions of section 106, it is not an infringement for the owner of a copy of a computer program to make or authorize the making of another copy or adaptation of that computer program provided:
>
> 1. that such a new copy or adaptation is created as an essential step in the utilization of the computer program in conjunction with a machine and that it is used in no other manner, or
> 2. that such new copy or adaptation is for archival purposes only and that all archival copies are destroyed in the event that continued possession of the computer program should cease to be rightful.

b) Lease, sale, or other transfer of additional copy or adaptation

> Any exact copies prepared in accordance with the provisions of this section may be leased, sold, or otherwise transferred, along with the copy from which such copies were prepared, only as part of the lease, sale, or other transfer of all rights in the program. Adaptations so prepared may be transferred only with the authorization of the copyright owner.

Given the fact that there is a change in storage format involved (from the hardware-based format of a set of ROMs to a software-based format such as a computer's recordable storage media), *we cannot lay any claim that our new copy is an exact copy of the original*. The actual object code of the software has not changed; what has changed is its delivery system. This is considered a *format change* by the courts (Mirage Editions v. Albuquerque ART, 1988). This fact was also recognized by the courts who had to deal with the subject in regards to computer software (Sega v. Accolade and Nintendo v. Atari, 1992). *What we are now dealing with is an adaptation of the original* - i.e. a **derivative work**, as the legal profession terms it - and there can be no debating the matter. This is from the same line of reasoning that the legal system used to come up with the concept of an intermediate copy, and they did this in order to better deal with the legitimate efforts of videogame developers.

Why am I drawing the distinction between an exact copy and a derivative work? Look at paragraph (b) of 17 USC 117, which many cart backup enthusiasts conveniently ignore. The presence of an exact copy, i.e. our so-called "backup," requires the presence of an original vendor-approved copy. To quote the law, "any exact copies ... along with the copy from which such copies were prepared." "A-ha!" the theorists will say, "I've got the original cart!" That's fine and good, but your so-called "backup" is not stored on the exact same media as your original copy. The change in format invalidated any claim you could make with regards to the exactness of your copy. The actual object code may be the same, but the media is not. *What you now have is an original copy and a derivative work, not an original copy and an exact copy.*

"Well," they stammer, "that's the same as saying it's an adaptation. Copyright law says that adaptations are legal. You said so yourself." That is true. A user is permitted to make an adaptation of a computer program in order to get it to work with their system, and this was one of the ten rights of the software user that we discussed last time (point 7 on the user rights list). Don't forget, though, that this is the one and only right of adaptation that a user has. Any other rights of adaptation rest with the copyright holder (17 USC 106, 106A). For example, it is a well-known practice of Sega Genesis videogame console owners to modify their systems so they work with any MegaDrive carts in their possession. The Genesis and the MegaDrive are the same console sold under different names in different markets, but the MegaDrive carts have a slightly different case housing than do Genesis carts. Owners adapt them by breaking off a pair of tabs on the cartridge port, thus permitting the larger MegaDrive carts to fit in the smaller port of a Genesis console. The same logic also applies to different system adapters (the G/MD PowerBase converter) and adapters for another vendor's console - all of which are legally recognized forms of videogame cartridge adaptation. "You see?!" comes the immediate response. "that makes my cart dumps legal. Those last two you mentioned are adaptations that involve emulation. Running software under an emulator is legal." Yes, running software under an emulator is perfectly legal (Sony v. Connectix, 2000). "Okay then," comes the reply, "what's the problem?"

I will ignore these pundits brushing over the fact that the last two adaptations I mentioned did not involve a change in delivery system format and directly address their contention. *An emulator by its very nature is an after-the-fact product*. Remember, emulators are designed to replicate the functions of systems that are either already in existence or no longer being produced. *You cannot make an emulator unless you have two things - knowledge of the system you are emulating and access to that system's software base.* It goes without saying, as any emulator developer worth his or her salt will tell you, that there has to be some form of software base already in existence from which they can work. They have to have something with which they can test their emulator in order to ensure that it works as well as the original hardware. Which came first, the chicken or the egg? Answer - the egg, as I explained last time. The original system always comes first, followed by its software base (the egg), and then any emulators that might crop up (the chicken). *If you are going to defend the cart dump backup theory, then you have to do it on the basis of the software alone. An emulator has nothing to do with it.*

I can imagine the face of many an average user turning red in frustration right about now, and some might even be spluttering in rage. "But ... but ... you haven't told us anything! Backups are legal! The law says so! So what if the emulator has nothing to do with it? I HAVE THE LEGAL RIGHT TO BACK UP

MY GAME CARTS!"

Wrong. *Users do not have the right to back up any kind of object code stored in ROM for any videogame system.*

"WHAT?!"

Listen, and listen carefully. Any unauthorized copy of a computer program, regardless of the original or resultant media, can be considered a counterfeit copy under federal law. (15 USC 1127, 18 USC 1030). The practice of dumping the videogame cartridges of a home videogame system by the average user is not justified under the backup proviso of copyright law (Atari v. JS&A Group, 1983), and this restriction also covers arcade videogames and any other ROM-derived formats as well (Tandy v. Personal Micro Computer, 1981). The one exception for ROM dumping is granted solely to bona fide developers and their associates (Sega v. Accolade and Nintendo v. Atari, 1992), with any resultant "intermediate copies" having exactly the same protections and restrictions as if they were the originals themselves. You, as a user, do not have the right to dump a piece of computer code stored in ROM format for use with an emulator, since you are not a developer (Sony v. Connectix, 2000). Also, this is not considered to be justifiable as an operational adaptation due to the necessary format change involved (Mirage v. Alberquerque ART, 1988). What you have produced is a derivative work, and all such works must be authorized by the copyright owner in order to be legal (17 USC 106). You cannot use fair use to justify ROM dumps, since the courts have denied this venue to the average user due to the illegal nature of the resultant copies (Sega v. MAPHIA, 1994). You may not obtain intermediate copies from a developer for your personal use, as this voids the developer's protections under case law (Sega v. Accolade, 1992). As an additional note, current EULA language by practically all computer program developers utilizing some form of permanent storage media specifically forbids the practice of dumping their code from its original media, and they are justified in doing so under federal contract law (ProCD v. Ziedenberg, 1996). Any unauthorized "ROM" in the possession of the average user is considered to be at best an infringing copy of the program in question (MAI v. Peak, 1993) and at worst a counterfeit copy (15 USC 1127, 18 USC 1030). *In short, the average user may not produce, obtain, own, use, or distribute any kind of infringing "ROM" without the authorization of its copyright owner.*

I can now hear a number of counterpoints quickly being raised by angry users who believe otherwise. "But what if I'm a developer?" No problem - your derivative work qualifies as an intermediate copy, and such are legally permitted; however, you had better be able to prove your claim of being a developer in the event that the copyright owner or original system vendor just happens to drop by (Sega v. Accolade, 1992). "But what if I just want to review the program? Can't I claim the fair use exception?" No, and we went through that discussion earlier, remember? I won't rehash it here, but suffice it to say that you have to have bonafide journalistic credentials (Bill of Rights, First Amendment) or possess the original before you can make such a claim (Nintendo v. Atari, 1992). "Okay, can't I just use it to test my emulator?" No - the very act of possession of an unauthorized "ROM" by the average user violates copyright law (17 USC 501).

"But what about the Betamax case? I can do anything I please in the privacy of my own home!" Maybe you can, but how did you get possession of that "ROM" in the first place? What about all those other "ROMs" that you have? More importantly, where are the originals from which they were produced? You can't backup copies of software that aren't in your possession (17 USC 117). *Face it, folks - no matter how you twist it, bend it, invert it, turn it around or even inside out, it is illegal to produce or possess a backup of a "ROM" that originated from some form of permanent storage media unless specifically authorized by the copyright owner of the computer code involved.*

So much for the vaunted ROM backup theory. It was fun while it lasted, but it ultimately proved bogus. The only way you can make a valid copy of the computer code contained within some form of permanent storage media, without being a bonafide developer or working in association with one, is to first obtain authorization from the copyright holder before you make your dump. If the program in question has been released into the public domain (as some have), then there is no problem. If it hasn't, then you've got a problem. What does this mean? *If you want to backup your favorite game cart, then go get another cart. If you want to backup your favorite arcade game, then go get another console.* Where do you get a spare cart? Contact the vendor - almost all of them are on record as stating that they will repair or replace a damaged or defective cart at little or no cost. Where do you get a spare arcade console? Try checking with your local coin-op vending company - they're fairly reasonable in their terms - especially with older, well-used systems, which you can then cannibalize to get your original back in working condition (as one of my friends recently did with his highly prized original *Star Wars* arcade videogame). I know this isn't what the emulation community wanted to hear, but that's what the law says.

> [EDITOR'S NOTE - *EmuFAQ* contributor Chuck Cochems has recently advanced a novel theory concerning unauthorized "ROMs." According to Chuck's theory, the average user could legally make and use "ROMs" of games stored in permanent media that they legitimately own under the concept of **personal use**, which is a protected form of noncommercial use under the concept of fair use as outlined by the Supreme Court's Betamax decision (Sony v. Universal, 1984). While I have certain problems with some of Chuck's arguments, I believe that the basis of his theory is sound enough to warrant closer inspection. You can read his complete thesis in the *EmuFAQ*'s Y2K supplemental article, "The Question of ROMs."]

BIOS DUMPING

This brings up another related issue that I have been putting off until now, since it goes along the same lines as the cart dump. This is the **BIOS dump**, which is *an image of the original BIOS required by a computer system or* *videogame console for its initialization and proper operation*. The practice of BIOS dumping goes as far back as 1982 and is still quite common in both the computer industry and the emulation scene. Reverse engineering a BIOS is rather tricky, and continues to be a considerable time and resource consuming process as the complexity of newer systems increases. Many emulator developers would rather not involve themselves in this process; therefore, they find it more convenient to dump a copy of the actual system BIOS and build their emulator around that. This is where the modern variation of the combined hardware/software emulator, the *BIOS dump dependent emulator*, first emerged, and the best known current example of this is the now-discontinued *PSEmu Pro* emulator for the Sony PlayStation.

So is it legal for a user to dump a BIOS? Absolutely not, for the same reason that cart dumping is illegal - it is a violation of the copyrights on the computer code contained within the BIOS. Why? To find the answer, we need to look back at the early days of personal computing.

The year is 1982. The place is the "clean room" of Phoenix Technologies, which is in a flurry of activity. Why? Because the company is under contract to Compaq Computer and is currently in the midst of a concerted effort at reverse engineering the IBM PC BIOS. Compaq wants a reverse engineered BIOS so it can build the world's first legal **PC clone** computer. If they can accomplish that one task, Compaq reasons, then they can build the rest of the computer from off-the-shelf parts. The suggestion comes up that Phoenix should make an exact copy of the IBM PC BIOS. "No, no!" someone shouts, "that's copyrighted computer code!" The decision is eventually made to reverse engineer as many functions as possible in order to get it to work exactly like the real thing, while dropping ROM BASIC along the way. The finished product is subsequently released to the market, and IBM sues for copyright infringement. The courts are unable to find any proprietary IBM microcode within the Phoenix BIOS. Phoenix is cleared of all charges, and the "clean room" reverse engineering technique becomes a legitimate bulletproof means of software development.

About the same time, in the same year, a deal is quietly being completed behind closed doors that would prove to have profound implications for the CPU industry. The senior partner in the deal is Intel Corporation, and the junior partner in the deal is AMD Technologies. Intel needs some help in producing the x86 series processors of the day - especially with regards to its new 80286 processor. The two companies enter into a technology transfer agreement, in which AMD receives copies of proprietary Intel microcode to use inside its clone CPUs so long as they do not share it with anybody else. AMD agrees, and shortly thereafter begins producing legal clones of Intel's 80286 CPU.

Move ahead in time to the year 1983. The place is federal district court, where a copyright infringement lawsuit is underway. The plaintiff is Apple Computer, maker of the immensely popular Apple II personal computer. The defendant is Franklin Computer, makers of the Franklin ACE 1000 clone computer. Franklin had decided to cash in on the success of the Apple II by producing its own clone. In order to cut corners and reduce development time, they dumped the Apple II BIOS and used parts of its microcode inside the

Franklin ACE 100 BIOS. Apple sues, and the court finds Franklin guilty of copyright infringement under federal law. The resultant fines and levies effectively end Franklin's involvement in the personal computer industry.

Jump ahead in time six years to 1988, where a young man named Simon Douglas is working on a product called *A-Max* that will provide Macintosh emulation for the Amiga. Fully aware of the IBM v. Compaq and Apple v. Franklin lawsuits, he builds his product so that it requires the use of a genuine Macintosh BIOS in order to function. He also designs an external adapter (and later a plug-in card) that will allow users to use the Macintosh BIOS with *A-Max* without violating Apple's copyrights on the internal BIOS microcode. His foresight is confirmed a year later in the courts, who rule that this use of the Macintosh BIOS in its original format is perfectly legal.

Now we shall take a rather large a jump to 1994, in which an appeals court ruling was handed down that shook the CPU industry. The plaintiff is Intel Corporation. The defendant is AMD Technologies. The issue is the Am386 CPU, which uses proprietary Intel 80286 microcode that the company had originally licensed from Intel as part of their secret pact back in 1982. The court rules that AMD's use of proprietary microcode, although originally permitted under their secret pact with Intel, will have to cease. AMD promptly settles with Intel and pays them for the right to continue using that code within their current product line.

Are you beginning to get the picture? *It is illegal for users to dump, duplicate, use, or distribute any copy or portion thereof of a computer system BIOS without authorization from its copyright owner.* The computer code stored within a system BIOS is specifically covered by case law, as the examples involving IBM and Apple demonstrate. If it has not been placed into the public domain, then licensing that code is the only legal way to use it, as the Intel v. AMD case shows. If you can come up with a way to use an original system BIOS without dumping it, as was the case with the original *A-Max*, then that is also legal. What does this mean? *It is illegal to duplicate an actual system BIOS in any way without the authorization of its vendor.*

I know some of you are immediately going to start claiming that you are only making or obtaining a "**BIOS backup**," but you will not find any refuge in that outlandish theory, either - no more than you did with the preposterous theory of cart backups. Like a game cart, a system BIOS is a piece of computer hardware. In its original form, it is not intended for any use other than proper operation of the computer system in which it is installed. You do not backup hardware components of your system in the same manner as you would a piece of system software - you obtain spares for the part(s) in question. A BIOS dump is not a valid spare due to the change in delivery system, which is the same case as with cart dumps. If you need to obtain a replacement part for your system, like the integrated circuits used to store a computer BIOS, then you have to obtain the part from the original vendor or a licensed vendor. If a legal clone exists, regardless of format, you may use that instead. Even if the system in question is equipped with a so-called flash BIOS, you do not have the right to dump it except for the sole purpose of upgrading - an action of which most system vendors approve, provided you destroy the resultant dump of the original BIOS once the upgrade is complete. *The only legal ways in which you can use a BIOS image with an emulator is if either the original vendor authorizes the use of an BIOS dump or you can find a reverse engineered BIOS image whose author(s) have placed it into the public domain* (which is something of a rarity). The novel concept of a "BIOS backup" of proprietary original system vendor computer code is both ridiculous and unsupported by either the industry or the courts except in certain carefully defined vendor-approved cases, such as the flash BIOS upgrade that I briefly touched upon a moment ago. *The only way you can legally dump a BIOS image for any purpose other than reverse engineering is with the approval of the original system vendor* (Apple v. Franklin, 1983). Creating an image for the purpose of a "BIOS backup" is as illegal as it is to use an unauthorized BIOS image with an emulator.

"A-ha!" you respond. "Reverse engineering! That makes it legal for me to use a dumped BIOS with an emulator, right?" No, not if you're a user. It's the same situation that we had with the argument over "ROM" backups - developers have this right, but users don't, and even there developers only have limited rights to use that dumped code. They can use a dumped BIOS to <u>develop</u> an emulator, but the final product can't require that dump for proper operation. They've got to either reverse engineer the BIOS as well or license the code. As a matter of fact, and this may surprise you, their dumping rights don't just stop with "ROMs" and the system BIOS. The U.S. 9th Circuit Court of Appeals has recently ruled that it is perfectly legal for an emulator developer to dump any code from the original system or its accessories that they need, <u>including the BIOS</u>, so long as that code is not protected by patent and the end result is a noninfringing product (Sony v. Connectix, 2000). In plain English, developers can do just about whatever it takes to make an emulator, so long as it doesn't require its users to do illegal things in order to make it work. *If you're not a developer, then you can't claim reverse engineering to justify a BIOS dump.*

<u>PUBLIC DOMAIN "ROMS"</u>

This leads us to the next obvious question, and the one which the frustrated majority is by now quietly asking. "Is there such a thing as a **public domain 'ROM'**?" You may be surprised to know that the answer to that question is "Yes." *Public domain "ROMs" do in fact exist*, even though a lot of original system vendors and their allies seemingly do not want you know about this - with some even going so far as to flatly deny their existence. Are you intrigued? Read on!

The idea of **public domain** software is widely credited to the invention of the LOGO programming language in 1980, which was generally available to anybody who wanted to use and modify it free of charge. The concept was given the modern twist of **shareware** in 1982 by Andrew Fluegelman with his *PC-Talk* communications software; the initial version was free, but continued support and updates were only available at a price. Today, both concepts are firmly established parts of the computer software community, with thousands of such pieces of software produced each year.

Computer software operating under emulation is no different that computer software operating under real hardware. It is regulated by the same laws and principles as any other kind of software. This means that there are a group of individuals either directly or indirectly involved with the emulation community who, like the "real" software community, do not mind or object to their software running under emulation. Therefore, they do what regular software authors and vendors do who feel the same way - they place their software in the public domain. Emulator authors do the same thing, with a handful making available their programs as either shareware or commercial products and the vast majority as public domain offerings. Back to the subject, though - there are a steadily growing number of public domain "ROMs" available for use with your favorite emulator. *Public domain "ROMs" are perfectly legal to both possess and distribute, since they are not under any kind of copyright restriction in this regard.*

Unfortunately, there is a body of opinion within the emulation community that feels that there are a number of "ROMs" that were commercial in origin but are now public domain. Let us take a brief look at some of these "ROMs" and whether or not they are indeed public domain.

"ROMs" without a copyright notice

Prior to the United States joining the Berne Convention, all copyrighted works (with certain exceptions such as paintings and so on) were required to contain a copyright notice. Berne waives this requirement, and grants automatic copyright protection to a given work the moment that it is created. *This means that any "ROM" you may run across without a copyright notice is still protected by copyright.* The only way to know for sure whether or not these "ROMs" are public domain is to check with either the original author or vendor (preferred), or with a person or organization that is an established authority on the subject. That way, you can find out whether or not the author has indeed placed that "ROM" into the public domain.

Prototype "ROMs"

These go by a number of names, depending on the development stage they had reached prior to their conversion into "ROM" format. **Alphas** are early copies of a computer program that are almost always non-functional to a large degree, and represent an early stab at actually implementing the intended concept. **Betas** are late-stage copies that are at least functional to some extent, though they tend to be rough around the edges and rather buggy. **Test copies**, which are more commonly called **protos**, are usually identical or almost identical to the finished product, intended for late-stage performance testing or "sneak peak" distribution, and can usually be identified by the crudeness or lack of title screens or legal disclaimers. Regardless of which of these examples you encounter, *a prototype "ROM" is protected by the same copyrights as if it were an actual finished product.* Just because the "ROM" in question is a prototype does not automatically mean that it is a public domain "ROM." It must be released into the public domain by either its author or vendor before it can be freely distributed, just as if it were a real commercial title.

Unreleased "ROMs"

These are programs that for one reason or another were never commercially vended. *Unreleased "ROMs" are protected by copyright law*. The fact that the program was never released does not void its copyright protection. Remember, distribution is one of the rights of the copyright owner. If they chose not to distribute it, then that does not automatically put that program into the public domain. You must first get permission from the copyright holder before you can legally distribute an unreleased "ROM."

This is all that we shall discuss on the subject of public domain "ROMs" for now. We will deal with it again when it comes time for our discussion of emulation's relationship with the Internet.

THE URGE TO PATCH

Computer software, being a human creation, is never perfect. It may have one or more internal coding problems that prevent it from working as designed. It may have been designed in such a way as to not work with your computer system. There are even cases where users develop a personal pique against certain aspects of a computer program, even though that program may work just fine. This is where **software patches** enter the picture, which are designed to fix a program in one or more ways so that it will better serve the needs of the user. So how does this apply to the concept of "ROMs?"

Insofar as I have been able to determine, there are at least three different kinds of software patch for a "ROM." The **hack patch** alters the graphics, sound, or performance of the "ROM" in some noticeable manner. These are most commonly made for videogame "ROMs," as they tend to enhance or otherwise change the game involved in some fashion. The **runtime patch** permits an otherwise difficult "ROM" to work with a given emulator. It removes or alters certain parts of the internal "ROM" microcode that would otherwise cause difficulties for the emulator. Finally, and most importantly, there is the ever-popular **translation patch**. This can be installed into a given "ROM" so that it is readable in the user's native language.

This begs the question - *is "ROM" patching legal? Yes and no, depending on the kind of patch and the "ROM" involved.* Yes, it is legal if you are dealing with a public domain "ROM." No, it is not legal if you are dealing with a "ROM" that is still protected by copyright. Why? The right of alteration with regards to a copyrighted piece of computer code is exclusive to the copyright owner (17 USC 106). Users have a limited right to adapt a piece of copyrighted computer code to work on a given system (17 USC 117). The right of adaptation deals exclusively with the actual operation of that piece of code on the system in question. Users have the right to adapt a program to work with their system, so if a legal "ROM" is involved then runtime patches are perfectly legal (point 7 on the user rights list); however, the same is not true of either hack or translation patches. The right of adaptation does not extend to the on-screen presentation (Playboy v. Frena, 1993). Limited copyright protection is afforded to on-screen presentations (Apple v. Microsoft, 1992); furthermore, any textual presentations displayed on a computer screen can be protected under copyright law (Digital v. Softklone, 1987). Furthermore, it is illegal to alter the contents of a program originally stored in ROM without consent of the copyright owner (Kramer v. Andrews, 1986). On top of that, *the United States recognizes copyright protection for computer software manufactured in foreign markets* (US 17 USC 104A). This means that foreign market "ROMs" are protected by copyright just the same as domestic "ROMs." *It is illegal to patch a "ROM" so as to alter the on-screen presentation of any protected material, regardless of country of origin, without the consent of the copyright holder.* This provision of U.S. copyright law that includes foreign-produced "ROMs" was added as a direct result of the United States becoming a member of the Berne Convention in 1995.

This problem was recently illustrated by an incident on 13 April 1999 involving the KanjiHack "ROM" translation service and ASCII Software of Japan. KanjiHack had produced an English translation patch for the "ROM" of the ASCII release *RPG Tukuru 2* for the Super Famicom videogame console. This title is better known to English-speaking Super Nintendo fans as *RPG Tool: Super Dante 2*. They offered this patch via their Internet site, which was apparently hosted by a service operating out of the United Kingdom (UK). ASCII had never authorized any third party to develop an English patch for the program, which meant that KanjiHack's translation patch was a direct violation of ASCII's copyright on *RPG Tukuru 2*. As such, it was within ASCII's legal rights under UK copyright law to demand that KanjiHack pull the patch, and they did so following the procedures laid out by that law. KanjiHack complied, with the result that ASCII did not carry through with its threatened legal action against them. The reason why this should be noted is that both Japan and the UK are WIPO Treaty signatories, and they have modified their copyright laws in this regard. Since the United States also recognizes the WIPO Treaty, and has also modified its laws along the same lines as Japan and the UK, then anybody who distributes the KanjiHack translation patch for the *RPG Tukuru 2* "ROM" within the United States without ASCII's consent is violating federal copyright law.

PATCHING IN PART

I can now hear the inevitable objection. "So does that mean all translation patches are illegal? What about a partial patch?"

The answer is, "No, not all translation patches are illegal." There is an exception to every rule, as the old saying goes, and so it is with translation patches. *There is at least one kind of partial translation patch that is perfectly legal, and that is the* ***menu patch.***

"So what's a menu patch?"

A menu patch is one that translates only the command structure of a given program. For example, a given program will start up and present you with a list of options - i.e. a menu of selections from which to choose. You pick one, and it either causes the program to perform some action or takes you to a sub-menu, from which you can make additional choices, and so on, and so on. You might also have additional options screens that allow you to change various program settings and other such minor items, and these too are considered part of the program's command structure. All a menu patch would do is render the program's command structure so it can be read in your native tongue. It does not translate any other part of the program - no help screens, no prompts, no conversations with program or player characters (in videogames), and so on. As such, the menu patch is rather limited in comparison to a full translation patch, but it does allow someone who is not familiar with the original language used in program presentations to actually use the program to some degree.

I can now hear the immediate question that comes from the various "ROM" translation groups our there. "Why is a menu patch legal, whereas a full translation patch is not?"

You will recall that copyright protections were established early on for interactive texts (Digital v. Softklone, 1987). You will recall that the courts have determined that user menus are generic items, analogous to the controls of a VCR, and therefore specific menu layouts do not qualify for copyright protection (Lotus v. Borland, 1990). You will also recall that only specific portions of an on-screen presentation can be copyrighted (Apple v. Microsoft & Hewlett-Packard, 1992). There is now a fourth case to consider that cements the legality of a menu patch, and that is the MiTek case (MiTek v. Arce, 1996). *The gist of the MiTek case is that ANY command structure for ANY program is a generic concept and therefore cannot be protected by copyright.* That includes menus, options screens, and the like - in short, a program's entire command structure. *Since a program command structure cannot be copyrighted, then it is perfectly legal to make a menu patch for it.*

So does this mean that all menu patches are now legal? ***Yes****. They always have been, and always will be.* "Cool - now I can patch all of my 'ROMs' and not worry about it anymore, right?!" Wrong. You still have to deal with the questionable legality of the actual 'ROM.' While the menu patch in and of itself may be legal, applying it to an illegal 'ROM' is not, nor does it make that illegal 'ROM' suddenly become legal. Remember, the ultimate right of adaptation rests with the copyright owner (17 USC 106). Copyright protections are afforded to foreign software within the bounds of various recognized international treaties (17 USC 104A). It may be perfectly legal for you to write a menu patch for your favorite 'ROM,' but it is not legal to apply that patch to an illegally generated 'ROM' without permission from the copyright owner. This is considered to be the same as an unauthorized modification of the original object code stored within the "ROM" (Kramer v. Andrews, 1986) and therefore illegal under copyright law (17 USC 501). For example, the MiTek case would not have helped KanjiHack one bit in their dispute with ASCII, as the real issue was the questionable legality of the *RPG Tukuru 2* "ROM" that needed patching in the first place.

MELDING A UNIFIED "ROM"

A new form of "ROM" alteration has been brought to my attention since the first drafts of the *EmuFAQ* were released, and that is *"ROM" merging*. The idea seems to have originated with the *M.A.M.E.* team and the various *Street Fighter 2* arcade ROM variants for that program's Capcom CPS-1 arcade hardware emulator, but has since been applied to other similar sets of "ROMs." The premise seems simple enough - take the different variant "ROM" sets for a given title, weed out the duplicate "ROM" dumps that are common to all sets (leaving you with only unique dumps), and then keying the emulator so can load up the appropriate configuration of "ROM" for each title variant. It's actually a rather neat idea, and it's a surprise that no one had thought of it earlier.

I've got bad news for the M.A.M.E. team, though, and for anybody else involved in "ROM" merging and the use of merged "ROM" sets. What you are doing is illegal.

"Aw, come on! Enough already! Leave us alone!"

No can do. I've been painfully honest so far, and I'm not about to stop now. Here's the lowdown.

It is illegal to alter or otherwise modify the contents of a computer program stored in ROM without the consent of its copyright holder (Kramer v. Andrews, 1986). It is also illegal to take parts of different versions or generations of a computer program and patch them together into a workable program (Allen-Myland v. IBM, 1994). This is classified as **rainbow code**, and it's easy enough to understand the allusion - a rainbow is a visible material object composed of different colors of the spectrum. Likewise, rainbow code is comprised of material that came from different sources and then melded together into a single, unified whole. Remember, the ultimate right of modification rests with the copyright holder (17 USC 106). You do not have the legal right to create rainbow code without the authorization of the copyright holder. That means that *it is illegal to* *merge "ROM" sets, let alone distribute them.* This is another additional avenue of attack that a vendor who feels that its intellectual property rights have been infringed by emulation might choose to use should it ever decide to take the matter to court. At least the emuscene now know about this possibility, and that way you can better prepare yourselves should the unthinkable happen to you someday.

THE RULES HAVE CHANGED

Almost everything that we have discussed up to this point changed when President Bill Clinton signed the **Digital Millenium Copyright Act** (DMCA) into law on 28 October 1998. Among other things, it modified several sections of the current U.S. Copyright Act and added a whole new section - *Chapter 12, Copyright Protection and Management Systems* (17 USC 1201-1205). I will deal in greater detail with the DMCA in our discussion of Internet-related issues, since that is where it has the greater impact on the emulation scene. There is a provision that deals specifically with the concepts that underlie the creation of "ROMs," though, and the relevant section is 17 USC 1201.a.1-3:

> 1. No person shall circumvent a technological measure that effectively controls access to a work protected under this title.
>
> > [editor's note - I have omitted the rest, as it deals primarily with regulatory oversight, implementation provisions, and a survey to be conducted on a regular basis by the Library of Congress in conjunction with the Department of Commerce with regards to the legalization of selective technologies covered by this law.]
>
> 2. No person shall manufacture, import, offer to the public, provide or otherwise traffic in any technology, product, service, device, component, or part thereof, that --
>
> > A) is primarily designed or produced for the purpose of circumventing a technological measure that effectively controls access to a work protected under this title.
> >
> > B) has only limited commercially significant purpose or use other than to circumvent a technological measure that effectively controls access to a work protected under this title.
> >
> > C) is marketed by that person or another acting in concert with that person, with that person's knowledge, for use in circumventing a technological measure that effectively controls access to a work protected under this title.

> 3. As used in this subsection,
>
> > A) to "circumvent a technological measure" means to descramble a scrambled work, to decrypt an encrypted work, or otherwise to avoid, bypass, remove, deactivate, or impair a technological measure without the authority of the copyright owner.
> >
> > B) a technological measure "effectively controls access to a work" if the measure, in the ordinary course of its operation, requires the application of information, or a process or a treatment, with the authority of the copyright owner, to gain access to that work.

Did you like that? Or are you (like most of the rest of us) getting a queasy feeling in the pit of your stomach? *In other words, any product that provides unauthorized access by the average user to original vendor code is illegal.* With regards to emulation and computer software, this includes any measures that vendors may include as part of their products to prevent unauthorized duplication, such as copy protection schemes and those tiresome antipiracy systems that certain vendors (a-hem!) like to include as part of their videogames. *Both the technologies and practices of ROM dumping, along with anything resembling it (including BIOS dumping), according to this and other provisions of the DMCA, are now illegal to manufacture, distribute, or use*. Face it, guys - the vendors have won this round in the emulation war. If you are a passionate collector of original videogame hardware and haven't yet dumped those beloved titles in your collection, then that's just too bad. Hardware dumping and anything having to do with it, including the technology, for any system that employs even a rudimentary form of "controlled access" is now effectively banned in the United States and its territorial possessions to all but a select few.

Remember that business about the translation patch? ASCII made its claim against KanjiHack's unauthorized translation patch on 13 April 1999 - well after the DMCA revisions to the U.S. Copyright Act were in place. *If this had happened within the United States, then their claim would have been based on the new form of the U.S. Copyright Act*, *which restores copyright protections to any and all works of foreign origin currently protected by copyright*. The additional modifications made by the DMCA with respect to the WIPO Copyright Treaty (WCT) and the WIPO Performances and Phonograms Treaty (WPPT) are rather extensive and I will not duplicate them here; rather, I will quote from the U.S. Copyright Office's official summary of the DMCA in this regard. If you have the Adobe PDF version of the DMCA summary, this is the last paragraph on page 2. The emphasis is mine throughout the quotation.

> Restoration of Copyright Protection:

> Both treaties require parties to *protect preexisting works from other member countries that have not fallen into the public domain in the country of origin* through the expiry of the term of protection. A similar obligation is contained in both the Berne Convention and the TRIPS Agreement. In 1995 this obligation was mentioned in the Uruguay Round Agreements Act, *creating a new section 104A in the Copyright Act to restore protection* to works from Berne or WTO member countries that are still protected in the country of origin, but fell into the public domain in the United States in the past because of a failure to comply with formalities that then existed in U.S. law, or due to a lack of treaty relations. *Section 102(c) of the DMCA amends section 104A to* ***restore copyright protection*** *in the same circumstances to works from WCT and WPPT member countries.*

There is language in the DMCA that briefly assures that such traditional exemptions as fair use will continue to go on their merry way, but its wording is such that it effectively returns almost all control with regards to any unauthorized use or modification of copyrighted computer software back into the hands of the copyright owner. Reverse engineering is still permitted, but under tightly controlled conditions - no freeware hacking as in the days of old. Field testing of new or soon-to-be-released products and performance testing of existing products (with or without the consent of the copyright holder) is also permitted - again, under rigidly established guidelines. Encryption scheme hacking is all but banned. Anything that infringes upon or alters in any way the original intended display or performance of a copyrighted work is likewise tightly controlled. The U.S. Copyright Act, which used to be fairly neutral with regards to technology issues, is now decidedly slanted in favor of the vendors.

> [EDITOR'S NOTE - see Chuck Cochem's *EmuFAQ* article "The Question of ROMs" for a different opinion on this issue.]

INTROSPECTION

So what have we learned? We can now agree on a common definition for the emulation community's use of the term "ROM." *A ROM is piece of computer software stored within a unique archival format not found in everyday personal computer usage that is intended to work with special program development tools, or some form of an emulator, or both.* As far as the first part of "the great emulation debate" goes, we have established the following:

*It is legal to produce a "ROM" from a piece of computer software stored within any form of permanent storage media **if and only** if you are a bonafide developer or legitimately affiliated with same.*

> Such "ROMs" are defined under case law as intermediate copies of said software, and they are to be treated as if they are the actual originals from which they were made. You must maintain the original hardware from which the dump was made so long as you maintain the intermediate copy, and you may not share that dump with anybody who is not directly involved with your development efforts. Free distribution of intermediate copies voids the special protections that developers enjoy in this regard and makes them prosecutable for intellectual property violation as if they were no different than the average user.

It is illegal for the average user to produce or obtain a "ROM" of a piece of computer software stored within any form of permanent storage media unless such practices are authorized by its copyright owner(s) or otherwise specified within its EULA.

> With regards to the specific practice of ROM dumping, this does not void the user's archival rights under copyright law, since one cannot restore a "ROM" copy to a read-only device. Any so-called "backup copy" of computer code dumped from ROM is in fact a derivative work, and as such is unquestionably illegal without consent of the copyright owner. In cases where a user needs a second copy of a program stored in ROM, the copyright owner must make reasonable provisions to replace any damaged or degraded copy of that software. If such provisions are not made available to the user, or are not available by other means (such as buying used or aftermarket copies), then federal law takes precedent and the user may archive that software by any means necessary as permitted under said law. [EDITOR'S NOTE - see Chuck Cochem's "The Question of ROMs" for a different opinion on this issue.]

It is illegal for the average user to manufacture, distribute, or use any kind of product that bypasses or violates the security systems of any computer-based system.

> Among other things, this effectively makes the manufacture, sale, distribution, and use of videogame cartridge dumpers by anybody who cannot demonstrate bonafide development concerns illegal under federal copyright law. [EDITOR'S NOTE - see Chuck Cochem's "The Question of ROMs" for a different opinion on this issue.]

It is illegal to dump, upload, download, distribute, modify, or otherwise infringe upon the copyrights for "ROMs" of computer software from foreign markets that are signatories to the Berne Convention, as well as those who are signatories to the TRIPS Agreement, the WCT, or the WPPT treaties of the World Trade Organization (WTO).

> Protections for qualifying foreign software were restored in full as of 1995 and further strengthened and expanded as of 1998. Qualifying foreign software is accorded the same rights and protections as domestically produced software under U.S. copyright law.

It is illegal to produce any kind of patch for a "ROM" that alters any protected elements of copyrighted on-screen presentations without the consent or authorization of the copyright holder.

> Case law has determined that specific elements of a given program's on-screen presentation may be protected by copyright, including any displayed texts outside of its command structure that are an integral and necessary part of the program's operation. The only documented exception to date is for patches that only affect the program's command structure, since that and the ideas that underlie it are generic concepts and therefore uncopyrightable. Any patch that goes beyond the subject program's command structure (menu, list of options, etc.) does not qualify for this exception. This includes translation patches (except for limited forms such as the menu patch), as the original displayed language of the program's copyrighted texts has nothing to do with the actual internal execution of its object code.

Any computer program without a copyright notice, any prototype, or any title that was unreleased for any reason is still protected by *copyright law to the same extent as is any other piece of computer software.*

> To lapse into emulation slang for a moment, it is illegal to distribute alphas, betas, protos, tests, or any other form of unfinished or unreleased piece of code without the consent of the copyright holder.

It is perfectly legal to produce a piece of software intended for public domain release that you then convert into some form of "ROM" format for use with a given emulator.

> You, as the author of that program, have the right to develop it for use with any computer system you choose, and you have the right to place both the original and the resultant "ROM" into the public domain regardless of whether or not you are a licensee to the original vendor for the system(s) or emulator(s) in question. Remember, the right of distribution, along with the ultimate right of alteration, rests with you - the copyright owner. If you choose to release your program as a public domain "ROM" for use under emulation, then that is your business and not the original system vendor's.

It is illegal for the average user to produce, distribute, or use an unauthorized dump of the BIOS of a computer system.

> Users cannot fool around with the original system BIOS or its internal microcode in any way without the authorization of the original system vendor. Emulator developers have a limited right to do as they please not only with the system BIOS, but any other system code stored within system firmware, so long as that code is not protected by patent and their final product is noninfringing. Developers may use a dump of an original system BIOS in developing an emulator, but the final or public release version of that emulator <u>cannot require its use</u> in order to function. [EDITOR'S NOTE - see Chuck Cochem's "The Question of ROMs" for a different opinion on this issue.]

This ends our discussion of common emulation community practices with regards to the software base. Now you know the legality of the issues concerning "ROMs," the BIOS dump, and "ROM" patches. You may continue to argue these issues, and I have no doubt that some of you will (or will deliberately choose to look the other way), but you can no longer claim ignorance of the law in this regard. You now know what you can and cannot do. It is sad that many will choose to do otherwise, because this will encourage the vendors to sponsor even more concerted attacks on the emulation scene, such as was recently leveled with the DMCA. Next time, we shall take a look at Internet-related emulation issues and how this controversial technology can successfully exist within this new digital frontier.

REVIEW QUESTIONS

1. What is software bootlegging? What illicit goods are marketed by this activity? How does it relate to the concept of software piracy?

2. What is a "ROM?" What are the two different types of "ROM?" How did each come about?

3. What is ROM dumping? How did this practice come about? What is the most common form of ROM dumping, and why does it go under that name?

4. Why was it not illegal for the average user in the United States to own a ROM dumper prior to 28 October 1998?

5. What was the court case that defined the relationship of videogame archiving with copyright law? What did it say?

6. Which two court cases established the legality of ROM dumping? What were the limits they placed on its use? What are two important points to remember regarding the ruling in these two cases?

7. What approach did videogame cartridge vendors eventually adopt to combat the perceived "broad legality" of cart dumping? How were they able to justify this approach?

8. Can you explain the ROM backup theory? Why does it apply to other forms of "ROM," such as arcade ROM dumps and disk image files?

9. Is a ROM dump an exact copy or a derivative work of the original program? How does this square with the rulings by the courts with regards to intermediate copies made by developers?

10. Do emulators play a valid role in the ROM backup theory? Why or why not?

11. Why are EULAs so important with regards to the cart dump backup theory?

12. What is the proper way to obtain a "backup" to a computer program stored within some form of permanent storage media? Why must this approach be used as opposed to certain other techniques?

13. Are BIOS dumps legal? Why or why not? Are there any exceptions? Why or why not?

14. Explain if you can why certain forms of "ROM" patching are legal, whereas others are not. Give examples when appropriate to illustrate your contentions.

15. What is rainbow code? How does this apply to the creation and distribution of "ROM" sets?

16. Is there such a thing as a public domain "ROM?" Why or why not?

17. Name three different types of "ROM" that the emulation community frequently assumes to be in the public domain, and explain why each is not.

18. How did passage of the Digital Millenium Copyright Act affect "ROMs?"

THOUGHTS TO PONDER

1. How would you define the term "permanent storage media" with regards to computer software? Is it legal to archive forms of permanent storage media that are not hardware-based? Why or why not?

2. Can a EULA ever be used to ban all forms of software archiving? Why or why not? Be sure to justify your answer with federal statutes, examples of case law, or both.

3. Is the mere claim of "developer intent" sufficient to justify either production or possession of copyrighted "ROMs?" Why or why not?

4. Will there ever come a time in which an emulator is created before its software base? Why or why not?

5. If unauthorized alteration of software is illegal under copyright law, regardless of country of origin, then why are there so many software patches available in the public domain? Why do some vendors tolerate this situation and others do not?

Module Two: The Software
Part 3 - Supporting the Software Base

Today, almost every piece of hardware ... has been emulated, or is about to be; and an expansive scene for emulation has emerged on the [Inter]net.

Howard Wen, "Why emulation makes video-game makers quake," *Salon*, 4 June 1998

THE END OF AN ERA

One of the most popular Internet sites on the emulation scene during 1997 and the early days of 1998 was a place known simply as **The Dump**. The brainchild of Harry Tuttle, it was at one time the premiere site for Sega Genesis/MegaDrive (G/MD) emulation on the World Wide Web. It had almost everything you would expect in a first class emulation site - a top-notch webmaster, an excellent site layout, a comprehensive database of information and reviews (for its day), and the largest collection of G/MD "ROMs" to be had above the board. The only real gripe of its users was its host, ClassicGaming.com, which was notoriously slow and erratic when it came to connections and downloads, but most of its users were willing to forgive this issue for the wealth of other treasures that The Dump offered. As the site grew, it expanded into two sections, the first comprising the original Genesis site and the second adding support for the Turbo Graf/X 16 - one of Harry's all-time favorite videogame consoles. Still, it was that part of the site now known as The Dump: Genesis that continued in its flagship role as *the* G/MD emulation site on the Internet, a role that Harry took quite seriously. It was a well-known fact on the emulation scene that Harry had or was in the process of getting and posting every single G/MD "ROM" known to be in existence, and there were many who supported him in this endeavor. As a result, The Dump: Genesis soon became the focal point of a concerted effort by the G/MD emulation community to account for every G/MD title ever made. Indeed, it was this project that soon made me a regular patron of The Dump in late 1997 and early 1998, and it was as part of this endeavor that I began work on what would eventually evolve into *The Genesis Game Guide* - the first, but not the last, FAQ that I would write for the classic videogame scene.

It was inevitable that the high profile of The Dump would cause it trouble with those who felt that videogame emulation was not an innocent hobby. The day of reckoning arrived on 20 March 1998, and it hit The Dump like a planeload of cement blocks dropped en masse from the nether regions of the stratosphere. On that day, webmaster Harry Tuttle found an official "cease and desist" order in his e-mail box, delivered courtesy of his ISP and sent to him by the IDSA. With his permission, I reproduce that infamous e-mail below:

> I am writing to you on behalf of the members of the Interactive Digital Software Association ("IDSA"), a trade association whose members include the leading publishers of interactive entertainment software. Our 45 members publish video and computer games for video game consoles, personal computers, and the Internet. A list of our member companies is attached as exhibit 1.
>
> It has come to our attention that you are distributing unauthorized copies of our members' copyrighted entertainment software titles - which you characterize as "emulators" - without their permission. Your statements in the disclaimer posted in your Website, that owning a pirate "backup" to a legitimate copy is false. The fact that a person owns a legitimate copy does not authorize any party to copy and distribute it without permission. The owner of the trademark and copyright in that product has the exclusive right to authorize its reproduction and distribution. As such, your unauthorized reproduction and distribution of our members' entertainment software titles is a violation of their exclusive rights under the Copyright Act, [as well as] federal and state trademark law and the law prohibiting unfair competition. Your actions are illegal and can result in criminal prosecution and/or civil liability.
>
> On behalf of your members, I demand that you immediately cease reproducing and distributing any entertainment software titles published by an IDSA member and that you delete any copies from your hard drive. We will be monitoring your Website for compliance with the demands contained in this letter and reserve all right to take any steps necessary to protect our member's rights, title, and interest in their property. In addition, in the event that you do not comply with these terms within fifteen days, we will pursue other avenues of relief.
>
> Sincerely,
> Gail Markels
>
> cc: classicgaming.com

And with that, The Dump: Genesis died.

There was an immediate firestorm of protest by regular patrons of The Dump. Angry emails poured into the IDSA, decrying their action against the forced shutdown of this and other similar sites around that time. Harry Tuttle himself received his own share of angry emails protesting his perceived cowardice, and one lamer even attempted to send him an email bomb as a sign of protest. None of it mattered. The Dump was dead, as far as G/MD emulation was concerned, and the emulation community would just have to get used to it. There was talk and even some plans of bringing The Dump: Genesis back as an info-only site, but nothing ever really came of it. Nobody's heart, not even Harry's, was really in the effort. What was The Dump without its "ROMs?" That was what had built its reputation in the first place. To many classic gaming fans, it just wouldn't be the same without them. The Dump staggered, reeled, lurched on, and eventually recovered almost two years later, but it never was the same. Its vast G/MD "ROM" archive was forever lost to the community. As it turned out, it was but one of many casualties from the IDSA's "great sweep" of 1998, which started on that black day and would continue through the spring and summer before finally tapering off in the fall. The shutdown of The Dump: Genesis marked the end of the Golden Age of Emulation, and terminated what old hands in the emuscene call its "salad days" - a time of unbridled restraint that is now forever lost to future emufans.

THE IMPACT OF THE INTERNET

As it has with every other aspect of the computer industry, the Internet has greatly expanded both the possibilities and the reach of the emulation community. It allows potential developers to trade ideas and test programs, as well as provide a common media through which they can converse with their testers. It allows fans to establish a support network, however haphazard that may be, for whatever emulation products they prefer to pitch. It allows vendors to monitor the progress of attempts to efforts to replicate the functions and support of their original products, or perhaps another vendor's product, and possibly consider the commercial implications of such. Finally, lest we forget the common complain of the software industry, it also provides a free and largely unchecked medium whereby both old and new computer software and videogames can be rapidly distributed to and from worldwide locales an in a manner the likes of which has never been seen before.

Those of us who were around back in the old days, when emulation was first legitimized, did not have the luxury of the Internet at our disposal. We had to do things the old fashioned ways: "sneakernet," word-of-mouth, friendly visits to a friend, pinching pennies for neato yet expensive products, and finally the good old-fashioned "copy party." All of that has changed with the arrival of the Internet. While us old-timers like to romanticize about those days and what they meant to us, I wouldn't want them back. The Internet is where the action is at, and that is where the bulk of system support now takes place.

As with anything related to computers, though, the Internet is a double-edged sword. What works for the honest vendor works equally well for the software pirate. What provides convenience for software support also provides opportunity for software bootlegging. The infamous copy parties of old are now about as dated as a horse-drawn cart on a freeway, what with all of the various "warez" sites out there one can visit.. The twin edges of the Internet's blade continue to enter more and more homes around the globe every year, and almost all home users gets a taste of both sides sooner or later. Overt and covert ... aboveground and underground ... legal and illegal ... these are the twin sides of the computer support coin, and they will be with us as long as computer systems remain a viable commodity.

So how does the Internet support the emulation community? How should the Internet support it? How would some like to see it supported, and how is it actually supported? These and other issues are what we shall now discuss.

THE TYPICAL EMULATION SITE

If you bounce around the various Internet sites dedicated to the emulation community, you'll see that they aren't all that different from sites devoted to other subjects. They have their news, message boards, editorials, "rants," and so on. They can have simple text-oriented layouts, or they can be flashy Java-driven affairs. They may or may not have "banners" and "pop-ups" (read "Internet advertising," i.e. "commercials" - ugh!), and they may or may not use "frames" to better compartmentalize the layout of the site. Many put up graphics content related to their favorite software, along with screenshots, box art scans, graphics "grabbed" from their favorite programs, and so on. Many will post "links" which you can use to jump to other related sites. Also, like many computer-oriented sites, almost all will have a download section where you can get what emulation-related software that the site offers. Even with all these commonalities, though, there are several things that set an emulation site apart from its Internet brethren, and the chief places to look will be in the links and downloads sections. *In most cases, your average emufan is looking for four things: **emulators**, **"ROMs,"** **patches**, and **links**.* I have taken it upon myself to nickname them "the four basic food groups of emulation."

- **Emulators**: Whether you are an emulation newbie or a veteran fan, users almost always want the latest and greatest versions of their favorite emulators. This is due to the natural assumption that the newest version works better and is more compatible with the emulated software base than the last release. Regardless of whether or not a site is dedicated to just one kind of emulated system or covers a range of systems, you should always see at least one emulation program of package offered for download. It is also here that you will usually find BIOS dumps for those emulators that require them.
- **"ROMs:"** If you offer the emulator, then it makes sense to offer the software, too - right? Almost all emulation sites will offer one or more "ROMs" for download to use with the emulator(s) also available on the site. Unfortunately, almost all of the "ROMs" offered in these cases are illegal due to their commercial origins, although many of the more reputable emulation sites are making a concerted effort to offer only legal "ROMs." On certain sites, you may also find BIOS upgrades for certain types of commercially vended cart dumpers, and I consider these to be a form of "ROM" as well. More on this subject later.
- **Patches**: Patches are important for two reasons: first and foremost, to alter a given "ROM" in some way; and second, to fix a problem or provide additional capability to an emulator. Almost all of the really good

emulation sites have download sections dedicated primarily to "ROM" patches. Emulator patches are rather rare, especially with regards to freeware emulators, as the author(s) usually prefer to recompile and then re-release their product instead.

- **Links**: So what if you can't find the emulator, "ROM," or patch for which you are seeking? The obvious alternative is to visit the site's links section and "surf" on over to one of the other emulation sites in the list. Chances are that the original site's webmaster(s) put that link there for a good reason....

In addition to your "four basic food groups," more sophisticated emulation sites offer several "side dishes" for those with advanced knowledge and skills. Here are two prime examples, but there may be others to consider as well.

- **Documentation**: The more you know about your emulator, the better you will be able to use it. The more you know about the program you are trying to run under emulation, the better you will be able to make it function. The more you know about the original hardware, the better equipped you will be for any emulation project you may wish to undertake on your own. The best emulation sites on the Internet have extensive documentation on almost every conceivable subject related to emulation: hardware and software user manual, OEM documentation, programming FAQs, source code archives, and so on. *Documentation is the key to making emulation work properly, and it is an area that the bulk of the community frequently overlooks.* Do you want to know just what hardware is contained inside you favorite emulated computer or videogame console? Do you want to know exactly how that one obscure memory map keeps your favorite emulator from working? Do you want to know how to build your own cart dumper for your favorite videogame console? These and many more topics can be perused on truly sophisticated emulation sites.
- **Utilities**: These little programs allow both emulator programmers and emulation power users to do all sorts of neat and nifty things. They run the gamut of subject matter, from "ROM" format conversion utilities and save game editors to sophisticated "ROM" hacking programs and alternate cart dumping hardware managers. You can also find such things as "ROM" and BIOS patching/upgrade tools, music players for any music that might have been dumped from a "ROM," "ROM" graphics editors, archive management software, and so on.
- **Save states**: These are usually files used in conjunction with a videogame to restore a player's saved game and therefore continue from the point where gameplay was suspended.   These are quite popular for obvious reasons, and rarely will you find a videogame vendor that objects to their existence.
- **Audiovisual rips**: These consist of one or more elements of a given program's audiovisual presentation that have been extricated for some

other undefined use. The term "rip" figuratively describes the act of "ripping out" the desired part of the program for use in your own project, which immediately raises all sorts of intellectual property infringement questions. The two major forms of audiovisual rips are program graphics (any on-screen elements up to and including a full-screen snapshot or graphic data embedded within the program itself) and extracted audio (such as digital samples or background music tracks)

WHAT IS LEGAL AND WHAT IS NOT

All of this wealth of material available for use by Internet emulation sites immediately begs the questions: *"Is this stuff legal for me to post?"* The question isn't as simple to answer as one might think. In some cases, the answer is "Yes," but in some cases the answer is "No." In addition, a lot of the perceived legality has to do with the objections of the vendor. A surprisingly large number of them choose to remain silent on the subject, which most emufans take to mean than their actions are perfectly legal. This is a mistaken assumption, but a quite common one that many make.

So why don't most vendors object to the emulation scene? It is difficult to say, since about the only ones who will even discuss the subject are the ones who are loudly objecting to the use of any of their products that are in some way affected by emulation. At the risk of speaking for the silent majority, though, let me offer a common-sense theory - and mind you, this is just a theory and nothing more. *The bulk of vendors do not object to emulation because it is too small a matter for them to really care.* In the majority of cases, the vendor product affected is either off-the-market or about to fall victim to that horrid economic calamity. There's no money in it anymore. It no longer has a perceptible impact on profit margins. That seems to be why they don't care. Now, before you start celebrating and jumping for joy, fellow emufans, that doesn't mean that they've totally lost interest in their product, nor does it mean that they've given up their intellectual property rights. *Anytime they want to object, they can do so as long as their product's intellectual property protection remains in place.* So what if that game hasn't been on the market for almost a quarter of a century? If they perceive a new market for their product, in this case growing popularity among emufans, then they may just decide to "reclaim" their product. They may be planning to possibly remarket it at some future date, or they just don't like the idea of not receiving any royalties from its widespread distribution by the emulation community. Whatever the real reasons may be, and they have no obligation whatsoever to reveal them to emufans, they have the full legal right to do whatever they want with their products. *If they want to deny the use of their products to the emulation community without proper compensation, then that is their legal right.*

Above all else, what you must understand is that legal restrictions on the use of material covered by one or more forms of intellectual property protection do not end once that material appears on the Internet. This concept of the law stopping at "the water's edge," as it were, also known as the "free flow of information" principle, was a common mistaken assumption early on in the frontier days of the Internet and actually succeeded in winning one solitary court battle in this regard (United States v. David LaMacchia, 1994), but that misguided notion has long since been laid to rest in subsequent legal action (RTC v. Netcom, 1995, is but one example among many). The U.S. Supreme Court has ruled in the past that "certain minimum contacts with [a forum]" are legally valid so long as they do not "offend 'traditional notions of fair play and substantial justice'" (International Shoe v. Washington, 1945). The immediate effect of their ruling was the passage of the various "long-arm" statutes by both federal and state governments that allow for the prosecution of criminal or civil offenders across geographical or political boundaries. The Internet is considered to be such a forum - it crosses all boundaries, such as local, state, provincial, regional, federal, international, and so on - and as such is eligible to be covered by any "long-arm" statute that may be brought to bear in an intellectual property dispute.

So how do you determine potential jurisdiction in an Internet-related dispute? The U.S. Supreme Court has addressed this issue by developing a three-part test to determine possible jurisdiction in such disputes (Asahi Metal Industry v. Superior Court, 1987), and the third part lists five fundamental factors to determine the fairness of such prosecution in a cross-border forum. I shall hereafter refer to this as the **Asahi test** due to the case that set the precedent.

1. The defendant must purposefully avail itself of the privilege of conducting business in the forum.
2. The cause of action must arise out of the defendant's activities in that forum.
3. The exercise of jurisdiction must be fundamentally fair under the following five factors:
    1. the burden on the defendant
    2. the forum state's interest in adjucating the dispute
    3. the plaintiff's interest in obtaining convenient and effective relief
    4. the interstate judicial system's interest in obtaining the most efficient resolution of controversies
    5. the shared interests the several states have in furthering fundamental substantive social policies

Again, as with all other forms of intellectual property infringement, it is up to the owner of that property to establish that a valid infringement has taken place. *"Long-arm" statutes are governed under rules 4e and 4f of the Federal Rules of Civil Procedure, and the courts have generally held that intellectual property infringement is prosecutable under same.*

Now that we up to speed with regard to original vendor concerns and how they could prosecute possible infringements via the Internet, let us take a look at the "four basic food groups of emulation" on the Internet and see how vendor concern plays into their true legality.

EMULATOR DISTRIBUTION REVISITED

Remember our discussion not that long ago about releasing an independently developed software-based emulator? That's a bit of a mouthful, so let's start using a comparable term - a **freeware emulator**. *A freeware emulator is an emulator that has been placed into the public domain by its authors without restrictions of any kind.* Almost all independently developed software-based emulators are freely available in the public domain, with only a rare handful offered as shareware. In addition, some vendors of commercial emulators will offer test copies or crippleware versions of their products as freeware emulators, with **bleem!** being the most recent and well-known example. As you may also recall, there are only two major areas of concern with regards to emulator release: *who originally vended the system* being emulated and *how old that technology might be*. To put it more formally, webmasters who would like to post a freeware emulator have to worry about both the emulator's **legal status** and its **economic impact** with regards to both the original system vendor and its licensees.

The legal status of a freeware emulator comes into play whenever intellectual property disputes arise. As you will recall, it is a common practice for an original vendor who objects to an emulator that is not their property - an "in-house emulator" as opposed to a freeware emulator - to claim patent, copyright, or trademark infringement in one or more ways. To borrow the example of a commercially vended emulator, Sony's claims against the *Virtual Game Station* videogame console emulator by Connectix were based upon the copyrights of the PlayStation BIOS code (violation of the U.S. Copyright Act) and the patents on the PlayStation hardware (violation of the U.S. Patent Act). To cite another example, this time from the burgeoning crop of freeware emulators, Nintendo took action against the developers of several different GameBoy emulators due to their unauthorized display of the Nintendo corporate logo upon startup, which was a clear-cut case of trademark infringement (violation of the Landham Act). As a result, posting a freeware emulator for a commercially vended system, especially one that has not been emulated before, is always a risky venture. Finally, never forget that posting an unauthorized BIOS dump is *always* illegal.

The economic impact of a freeware emulator is perhaps the most important concern to an original system vendor. As we mentioned before, it is not generally acceptable to emulate a system that is still economically viable. Yes, it has been done and will continue to be done, but the legal risk to the emulator development team decreases exponentially with the age of the system in question. The newer the system, the greater the risk to emulate. Systems which are still on the market pose the greater risk, and the newest systems pose the greatest risk of all. Epsilon and RealityMan's *UltraHLE*, a freeware N64 released to the public when the actual videogame console itself had only been on the market for just under three years, is perhaps the most oft-quoted example to date, but it is not the only one. Christian Bauer's *Shapeshifter,* a shareware Mac emulator for the Amiga first released in 1993, is an older example of this principle in action - although Apple's experience with *A-Max* meant that *Shapeshifter* was left for the most part left alone. In general, the greater the potential economic impact of a freeware emulator, the more likely the original system vendor is to object.

It is at this time that we should also discuss in brief a related topic that has reared its ugly head in recent months - **the pirating of a commercial emulator**. Yes, not even emulators are immune from the activities of the software pirates. It is now an open secret that a bootleg version of the commercial PSX emulator *bleem!* has been available on the "warez sitez" ever since its commercial debut. I could name names and give appropriate URLs, but I won't. This is not the first time that this has happened - remember the *A-Max* bootleg? - but bootlegging a commercially vended emulator is just as illegal now as it was then. *It is illegal to distribute a commercially vended emulator without the authorization of the vendor*. If you do, then you are just as liable for charges of software piracy as if you were posting any other kind of bootleg software.

What does this mean for webmasters of emulation sites? It means that you had better be reasonably certain that the emulator(s) you are making available for download are valid public domain products. If there is the least bit of doubt with regards to a given emulator, then don't post it. Is it really worth being the first with a new emulator if it means getting your site shut down? Let someone else take the legal hit, and that way you don't have to worry about any flak from the vendors and their allies.

THE LEGALITY OF "ROM" POSTING

There is only one court case that I will mention concerning the posting of "ROMs," but it is key to anyone who is planning on establishing an emulation site with "ROMs" being an integral part of their download section. Although it dealt with an old-fashioned electronic BBS, *its impact on the Internet is obvious*. That case is **Sega v. MAPHIA, 1994.**

Chad Scherman, aka "Brujjo Digital" (among other aliases) was a computer hacker living in San Francisco, California who was both operating and networking with a series of computerized bulletin board systems (BBS) in and around the city. These fell under the moniker of *pirate BBS* due to a number of illegal activities that were both promoted and conducted on these boards - the chief of which were Psychosis and MAPHIA, all of which were linked together under a loose alliance named PARSEC. Among the many activities that Scherman conducted on MAPHIA was the posting of unauthorized copies of Sega Genesis videogames in cart dump format. These were uploaded by PARSEC members or Scherman himself and collected primarily on the MAPHIA BBS, where they were made freely available to anybody who wanted them. This was done by and with the full cooperation of Scherman and his buddies, with numerous message posts and documentation files available on the MAPHIA BBS describing how to dump the originals via a cart dumper and how to patch or fix the resultant dumps in order to make them work properly. Most importantly, Scherman charged fees for his copies and his services, as well as for the information available on and material sold through PARSEC and its allies.

On 17 December 1993, Sega filed for court action under federal and state law against Scherman's activities, resulting in the complete shutdown of the PARSEC network and confiscation of all of its properties. Sega filed the following charges against Scherman's activities under the then-current forms of U.S. intellectual property law:

- federal copyright infringement (17 USC 101, et. seq.)
- federal trademark infringement (15 USC 1051, et. seq.)
- federal unfair competition and false designation of origin (15 USC 1125, paragraph a)
- trade name infringement under California state law (California Business and Professions Code, section 14400)
- unfair competition under California state law (California Business and Professions Code, sections 14210 and 17200-17203)

On 28 March 1994, the U.S. District Court, Northern District of California, ruled in favor of Sega on ***<u>all</u>*** counts.

Sega v. MAPHIA is considered to be the guiding case when it comes to the illegality of "ROMz" and "warez" sites on the Internet. I have listed below some of the chief points from Judge Wilken's ruling on the matter that are of prime concern to those of us within the emulation community.

- *Federal copyright law covers the on-line storage, uploading, and downloading of "ROMs." Doing so without the consent of the program's author(s) or vendor(s) constitutes a clear-cut violation of the law.*

  "Sega has established a prima facie case of direct copyright infringement under 17 USC 501. Sega has established that unauthorized copies of its games are made when such games are uploaded to the MAPHIA bulletin board, here with the knowledge of Defendant Scherman. These copied games are thereby placed on the storage media of the electronic bulletin board by unknown users.... Sega [also] has established that unauthorized copies of these games are also made when they are downloaded to make additional copies by users, which copying is facilitated and encouraged by the MAPHIA bulletin board." (see MAI v. Peak, 1993)

- *Any and all members of the site staff can be held culpable for any form of intellectual property infringement that takes place in connection with the illegal storage and distribution of "ROMs."*

  "[O]ne who, with knowledge of the infringing activity, induces, causes or materially contributes to the infringing conduct of another," may be held liable as a contributory infringer." Judge Wilken cited an example from case law that went back to a legal dispute between the estate of noted music composer George Gershwin and Columbia Music regarding the unauthorized reproduction of some of Gershwin's music.

- *Ignorance of the law is no excuse.*

  "Even if Defendants do not know exactly when games will be uploaded to or downloaded from the MAPHIA bulletin board, their role in the copying, including provision of facilities, direction, knowledge and encouragement, amounts to contributory copyright infringement." (see Playboy v. Frena, 1993)

- *Any site that contains unauthorized copies of copyrighted software, whether or not it was actually commercially released, is in direct violation of copyright law. "Fair use" may not be cited to excuse such activities, as they are not in the best interest of the original vendor and therefore fail point 4 of the "fair use" test.*

  "Because users of the MAPHIA bulletin board are likely and encouraged to download Sega games therefrom to avoid having to buy video game cartridges from Sega, by which avoidance such users and Defendants both profit, the commercial purpose and character of the unauthorized copying weighs against a finding of fair use." (see Atari v. JS&A, 1983; Nintendo v. Galoob, 1990)

- *Any site that contains unauthorized copies of software that contain one or more unauthorized trademarks, whether or not it was actually commercially released, is in direct violation of trademark law.*"

  "It is clear that Defendants' activities subject Sega to the possibility of irreparable harm. Each illegal copy of a Sega game which Defendants distribute deprives Sega of revenue. Moreover, distribution of altered, inferior copies of Sega games and of confidential, pre-release unperfected games subjects Sega to damage to its business and reputation." (see Franklin v. Franklin, 1971)

In short, *there is no legal basis for the existence of "ROMz sitez,"* nor is there legal cover for other emulation sites who wish to post unauthorized "ROMs."

So which "ROMs" are legal to post on the Internet and which are not? The answer is quite clear - *only those products whose authors have approved their widespread distribution without recompense or restriction are safe for posting*. As far as the emulation community is concerned, there are three broad categories of "ROMs:"

- **Public domain** - those "ROMs" that were created for the express purpose of free distribution at no user cost. There are usually no restrictions placed on them, although some authors have been known to charge a small fee for continued maintenance and support. *Any "ROM" that has been declared to be in the public domain by its author or copyright holder is safe to post - so long as it does not contain any infringing trademarks.* If a "ROM" intended for public domain contains an infringing trademark, then it should not be considered public domain. It is up to the owner of the trademark to decide whether or not the use of that trademark in that piece of software is infringing - not the programmer, users, or distributor.

- **Inactive commercial** - This is a term that I have coined to denoted those "ROMs" originally produced for commercial release - i.e. "works for hire" per copyright law - whose copyright owners have decided to waive their distribution rights as protected under applicable copyright laws. In such cases, the owner involved will make a public statement saying that these commercial products are now being placed into the public domain (or something along these lines), and a copy of this statement can and should be made available to anyone concerned. This is where I get the term "inactive commercial" - the copyright protections on these "ROMs" are no longer active. *Any "ROM" whose owner has waived their copyright protection with regards to restrictions on distribution as described above is safe to post.* One example of such a "ROM" is *Zero Tolerance* by Zyrinx Software for the Sega Genesis/MegaDrive videogame console, but there are many others.
- **Active commercial** - as opposed to "inactive commercial," these are "ROMs" to which their owners have *not* relinquished their copyright protection. Said protections are considered to be still active; thus their distribution is still restricted under applicable copyright law. *It is illegal to post any active commercial "ROM."* For example, Nintendo has made it clear that it has not authorized the free distribution of <u>*any*</u> of its videogame software for <u>*any*</u> of its videogame consoles, both past and present, at this time. This puts all existing Nintendo "ROMs" into the "active commercial" category, and therefore illegal to post. I note in passing that Nintendo has confirmed the validity of this definition in both their updated emulation FAQ and in several private emails sent between us as part of my research. The only way to legally obtain "ROMs" of Nintendo titles over the Internet is through Nintendo itself, and it has recently started an Internet service open to any willing to patronize it as paying customers.

<u>THE POSTING OF PATCHES, DOCS, AND UTILITIES</u>

So if it posting emulators is for the most part okay, and if is okay to provide public domain "ROMs" for them, then what about other kinds of emulation-related materials? Well, it depends on the material in question, and I shall deal with it in three broad categories - patches, documentation, and utilities.

- **Patches** - If you will recall, it is illegal to make any kind of patch that modifies a piece of software in any way, with the exception of producing an adaptation that enables it to work properly with your system, without the consent of the copyright owner. *This means that just about the only kind of patch that are unquestionably legal to post are runtime patches.* As to other kinds of patches, it all depends on the copyright holder -

whether or not they object to a given patch. You will recall the example of KanjiHack's English translation patch for ASCII's Japanese release of *RPG Tsukuru 2* for the SNES/SFC - an excellent example of the illegality of unauthorized translation patches for computer software originating from a foreign country (17 USC 104A). The same principle also applies to the various kinds of unauthorized "hacks" - not only for "ROMs," but emulators as well - does everybody recall the unauthorized *NeoRage* hack from 1998? Setting aside certain other legal issues for the moment, the authors of that emulator could have brought legal action against those who hacked their program. Most vendors don't object to patches for their products, but the operational rule of thumb is this: if a vendor objects to a given patch, then accede to their wishes and pull it immediately. If you want to play the patch game, then play it straight and safe.

- **Documentation** - Most of the better emulation sites carry extensive documentation archives covering a variety of topics. Unfortunately, not all of this material is legal. *It is illegal to reproduce any form of copyrighted document in extended form within any kind of media without the consent of the copyright owner*. While it is legal to reference such material, and even quote it to some extent under the principle of "fair use" (three midsize paragraphs is the generally accepted rule-of-thumb for the top limit), anything else must be specifically authorized. Remember, the exclusive right of distribution rests with the copyright holder (17 USC 106). Many on-line documents that emulation sites make available are complete reproductions, and as such are illegal in nature unless the webmaster has been granted the right of reproduction from the copyright holder. Examples of commonly found on-line documentation that can be considered illegal include the following: original system technical documents, original system owner's manuals, reproductions of advertising materials (such as box art and published ads), original software user's manuals, and extended listings of source code for copyrighted computer software. In addition, any of this and other such material that contains one or more trademarks further compounds their illegal nature. I will deal more with trademarks and the Internet later on within this discussion.
- **Utilities** - The urge to "hack" or "rip" necessitates the creation of tools to facilitate these activities. With this in mind, a number of ingenious programmers have and continue to develop a growing number of emulation-related utilities. It is not illegal to post an emulation-related utility so long as the intended use for that utility is not illegal in nature. Yes, I know that this may sound a bit confusing, but remember the big discussion we had about the legality of cart dumping? N64 cart dumpers are now illegal under the DMCA because one of their designed functions is to bypass the antipiracy security system built into every single licensed N64 cartridge, so any software-based utility program that also bypasses the N64 antipiracy system in any way is illegal. (17 USC 1201.a). This immediately brings into question the legality of a number of "ROM" hacking utilities that are out there, but they are not the only emulation-

related utilities one can post on a site. What about emulator "front-ends?," or graphics editors, or music utilities, and so on? Remember, the ultimate right of modification, except in cases of adaptability for proper operation, rests with the copyright holder (17 USC 106). The operational rule-of-thumb seems to be that utilities are okay to post unless one or more vendors perceive possible intellectual property infringement. *If a vendor object to one or more unauthorized utilities on your site that can be used to modify one or more of their commercial products, then it is in your best interests to remove the offending utilities at once.*

THE LEGALITY OF LINKING

Perhaps the most misunderstood form of intellectual property abuse on the Internet is the hyperlink, or "link" for short. This innocent-sounding item is actually a HTML tag that permits a user to immediately access data on another site from the site they are on. It can be a direct link, in which the user is immediately transported to the other site; or it can be an indirect link, in which the item is merely made available for immediate reference or download by the user. The practice of linking has been around almost as long as the Internet itself; however, its legality is frequently misconstrued.

Is it legal to provide links on your site? Not always. Okay then, is it legal to provide links to other sites? Same answer - not always.

"Huh?" you might ask.

Your right to put anything you want on your Internet site ends as soon as intellectual property protection enters the picture. It is now a common practice to make both copyrighted and trademarked material available on the Internet. Patented software's impact is minuscule in comparison, but its day is coming, too. Remember, the owner of a piece of intellectual property has the exclusive rights of distribution and usage of that work, which they can grant on as wide or narrow a basis to whomever they want on pretty much whatever terms they want. In addition, there is a growing body of case law that deals specifically with Internet linking and what you can and cannot do or use with regards to copyright material. While a full discussion of these cases is beyond the scope of this document, I do want to take a moment to deal with the one that appears to have started it all.

In Shetland Times v. Dr. Jonathan Wills and Zetnews Ltd, 1996, the principle was established that the practice of linking can be governed under intellectual property law with regards to perceived infringement or violation. The actual case itself involved trademark infringement within the United Kingdom and was judged by Lord Hamilton at the Court of Session, Edinburgh, Scotland, but has served as the yardstick by which Internet intellectual property disputes around the world are judged (for an example within the U.S. based on the Shetland case, see Futuredontics v. Applied Anagramics, 1998). Also, in Intermatic v. Toeppen, 1996, it was established that infringement can occur regardless of the place of residence of the offender. To wit, *any kind of link that involves the unauthorized* *use of commercial materials or directs a users to or from a commercial site in an unauthorized manner is illegal under copyright law.* The *Futuredontics* case extended this to include "framed" material as well.

If you plan on operating an emulation-themed site on the Internet, then you need to be careful not to include any kind of infringing links on that site. Below is a list of the most common pitfalls to avoid in this regard.

> *Providing a link to infringing material which can be accessed on your site or an entirely different site*.
>
> It doesn't matter whether or not the infringing matter is on your site or somebody else's, nor does it matter if either of the sites in question are in entirely different physical locales. "Long-arm" statutes allow for the enforcement of of intellectual property laws with regards to violation in a valid forum such as the Internet regardless of where, when, and how the infringement takes place (CompuServe v. Patterson, 1996). As far as the law is concerned, the only issue at stake is who gets jurisdiction - hence the Asahi test.
>
> *Constructing a link so that the actual link itself contains infringing material*.
>
> Use of a registered trademark (a name, phrase, or graphic) as a link is illegal outside of three specific exceptions. The first is in making a link that takes you directly to the home page of the trademark owner's Internet site (Playboy v. Universal Tel-a-Talk, 1998). The second is in making a link that is obviously critical of the trademark owner and does not come from internal or former (read disgruntled) sources, as open criticism is protected under the First Amendment right of free speech (Bally v. Faber, 1998). The third, and perhaps the one most important to a site dedicated to emulation, is the use of a trademark when such is absolute necessary to describe the subject matter at hand (Playboy v. Welles, 1998).
>
> *Utilizing infringing material to support a link or linking system*.
>
> While I am not yet aware that this has actually happened, it is worth noting due to the rise of software patents. If a link uses patented code or a patented process without the authorization of the patent holder, then such use would be illegal. This does not apply to search engines, as the search engine is a generic concept (In re Compton, 1994). It is still theoretically possible, however, to construct a link that might illegally employ a program that contains unauthorized patented code, and there are similar examples to be

> found in the regular computer software industry (Stac v. Microsoft, 1994).
>
> *Bypassing the intended operation of another site in an unauthorized manner by means of a link*.
>
> The case that set the precedent for intellectual property infringement on the Internet (Shetland v. Wills and Zetnews, 1996) established this principle from the beginning. The *Shetland Times* had established a set procedure on its copyrighted Internet site whereby users could access news stories. Zetnews bypassed that procedure, and the Edinburgh Court of Sessions found their actions to be illegal with regards to Scottish copyright law. The United States and other Berne, WCT, and WPPT signatory countries all acknowledge the Shetland precedent. In other words, if you want to provide a direct link to something on somebody else's site, then you need to get their permission first. If you don't, then you could be found guilty copyright violation under the Shetland precedent.

FAIR USE AND THE INTERNET

So, if intellectual property protections also extend to material transmitted via the Internet, then the principle of fair use should also extend, right? Yes, it does, along with its inherent limitations. They are the same ones that we talked about earlier when discussing the unlawful duplication of computer software. It is because of the free-wheeling nature of the Internet that "fair use" issues tend to get a little murky, and this principle is seemingly used at will by both webmasters and users to justify almost any practice under the sun. While a complete discussion of fair use as it exists on the Internet is beyond the scope of this document, I would like to take a moment to see how it applies to the emulation community. Legal precedents exist that an emulation site must follow with regards to perceived intellectual property infringement on the Internet.

Remember our discussion of "fair use" and copyrights? This begs the question: *"Is it legal for me to use copyrighted material on my Internet site?"* The answer is, *"**NO** - not without the consent of the copyright holder.*" If you will recall, copyrights can be used to protect any form of expression. In the digital age, this includes any and all types of computer code and digital audiovisual works: programs, programming languages, program source code, specialized databases, expert systems, unique sound and/or graphic creations, specialized computer software (such as but not limited to games, applications, and operating systems), and any digital reproduction of the old-style media. You may also recall that only non-commercial ventures qualify for the safe harbor provision of fair use under copyright law; in other words, you have to be a bonafide educational institution or non-profit archival service in order to be exempt from prosecution for duplication of copyrighted material. The average website does not qualify for the safe harbor exception unless it is owned and operated by one of the aforementioned organizations. You can't even use original, noninfringing material you find on other web pages without the express permission of that site's operators, since your use of their original material constitutes copyright infringement on your part. Therefore, it is illegal for you to use any form of copyrighted material in the design and layout of your site without the copyright holder's permission (RTC v. Netcom, 1995). As a caveat, I remind readers that it is the responsibility of the vendor to make the charge once they learn of such an act; otherwise, they lose their right to do so after a set period of time.

One of the most annoying assaults by a vendor upon a Internet site made by the fans of one or more of its products is to claim *trademark infringement*. In almost every case, the vendor will claim that the site in question is making improper use of one or more of its trademarks, and that such usage has not been authorized by the vendor in any way (Maritz v. Cybergold, 1996). It may interest you to know that emulation sites are not the only ones who suffer from this practice. The computer industry as a whole is affected by charges and counter-charges of trademark infringement, with Nintendo being one notable example in the videogame industry. In fact, there are many fan-based Internet sites for popular products, not all of which deal with computer software, who have been taken to task by the corporate big boys for perceived trademark infringement. Do the names Paramount, LucasFilm, and Disney ring a bell? All of these corporations have at one time or another gone after Internet sites for their unlicensed use of their registered trademarks. If you are the webmaster of an Internet site, then you by now probably asking yourself this question: *"Is it legal for me to use a trademark on my site without authorization from its owner?"* The answer is *"**Yes**, but only under certain conditions."* What are they? They are the same as the three exceptions for linking - a direct reference or credit to the trademark owner, obvious criticism by a non-employee or recent former employee of the trademark owner, and absolute essential use with regards to site content. These exceptions are interpreted rather strictly by the courts, so you need to keep that in mind before you start using trademarks on your site.

THE DMCA AND THE INTERNET

The passage of the Digital Millenium Copyright Act in late 1998 codified many of the legal issues that had been raised in recent years with regards to intellectual property protection in the computer age. While it was not the law that both the Clinton administration and the vendors originally wanted (that was the National Information Infrastructure Copyright Protection Act of 1996, which failed to clear Congress), nevertheless it gave vendors a new set of cudgels with which to beat unsuspecting infringers. We have already seen how it has had a major (and in certain cases detrimental) effect on programmers, developers, and users of computer software. We shall now see where it has its greatest effect, and that is the Internet.

Think about the term ISP for a moment. It is an acronym for *Internet service provider*, and in its simplest sense means one who provides access to the

Internet. The DMCA definition is somewhat more elaborate, in which an *online service provider*, or OSP, embodies the dual concepts of communications (connection, transmission, routing, etc.) and online services (information, storage, caching, directories, linking, et. al.). Such a broad definition covers every conceivable activity that can take place on the Internet, which means than anybody who qualifies as an OSP under federal law is also legally liable for ensuring the sanctity of intellectual property protection. Since this discussion is limited to just Internet-related issues, I will to stick to using the more familiar term of ISP; however, keep in mind that ISPs are covered under the DMCA's broad definition of an OSP.

The DMCA recognizes four major areas in which an ISP could be held liable for intellectual property infringement. These are as follows:

- ***Storage*** - this covers any and all forms of storage, including transient, intermediate, and permanent, of any kind of material by users for the purposes of connection, transmission, or routing. Examples of such storage include *web pages*, *chat rooms*, *message boards*, and *downloadable files*.
- ***Cache*** - this covers the temporary storage of material by the ISP's system while en route to another ISP, or the online storage of material used by the ISP itself or another ISP for access by its users. The *e-mail attachment* is an excellent example of caching on the part of an ISP - it "holds" the attachment, whatever it may be, until such time as the e-mail can be downloaded to its intended recipient. This also applies to the practice of *mirroring*, in which popular material is copied to more than one site for easier access.
- ***Service*** - this covers any act on the part of an ISP that would make any kind of information available to its users or at the direction of certain users for other users. For example, you might find a way to obtain additional storage space from your ISP to provide more elbow room for your ever-expanding Internet site. This is a service provided to you by the ISP, and they can be held legally liable for whatever you do with that extra storage space.
- ***Hyperlinks*** - this covers any means or method whereby a user could be directed to a specific file or location either within the ISP or available through another ISP. The common practice of *linking* immediately comes to mind, but this also covers on-line storage *directory listings* and *site indexes*as well.

ISPs are by their very nature passive organizations. They provide a means and the methods by which their users operate; they are not nor should be a babysitting service. In theory, all they should have to do is provide the connections, provide the storage space, and then sit back and collect the user fees. Right? Well, that's how it should be, but that's not the real world. The activities of many an unscrupulous user (or employee, in certain infrequent cases) coupled with possible legal liabilities should one of their users break one or more laws can cause major headaches for an ISP's management team. Like any other business, they don't want to mess with the courts. Lawyers are expensive, and court cases take a long time to resolve. That's dollars lost in potential profit, and enough losses over a long enough time will put them out of business. If there is any way they can legally get off the hook at minimum cost (preferably none), they will. Thanks to the DMCA, their "out" has now arrived in the form of the Online Service Provider Liability Limitation Act, which adds yet another new section to the U.S. Copyright Act dealing specifically with the legal liabilities of ISPs (17 USC 512).

So how does an ISP avoid legal liability for the activities of their users? The DMCA provides the following broad guidelines for any ISP who wants to protect itself in this regard:

1. The ISP must maintain a written policy with regards to repeat offenders of site hosting guidelines, up to and including termination of service.
2. The ISP must have a full-time representative who can both interface with individuals and organizations claiming intellectual property infringement. This representative must be clearly identified and full contact information provided by the ISP to both it users as well as the federal government.
3. The ISP must provide any and all information as necessary to interested parties under the terms of federal law with regards to any intellectual property violation(s) on the part of one or more users. It is the responsibility of the ISP's official representative to act as liaison in this regard.
4. The ISP must promptly respond to any and all perceptions of intellectual property infringement.

The main point is that every single ISP who wants to avoid any legal liability must develop a legal intellectual property protection policy, make sure that this policy is clearly understood by all parties involved, and strictly enforce that policy when it comes to dealing with possibly infringing material. In addition, the DMCA has some specific things to say concerning certain areas of possible intellectual property infringement that might affect any claim that the ISP may lay to the "safe harbor" provision laid down for OSPs under the DMCA:

Storage and linking

1. The ISP must not know that the material or link in question is infringing.
2. The ISP must not have any information that would make them aware of the infringing nature of said material or link in question.
3. The ISP must act expeditiously to block or remove access to said material or link as soon as it is made aware of its infringing nature.
4. The ISP must not receive any kind of financial compensation with regards to the infringing material or link in question.

Caching

1. The storage of cached infringing material must not have been provided as an active service on the part of the ISP (the DMCA mentions the use of an *automatic technical process* that does not involve any direct ISP participation, and such processes are commonly understood to include routine automated server operations).
2. The cached material is passed on without comment or modification by the ISP.
3. The caching process used conforms to standard industry guidelines.
4. The caching process used does not interfere with any information about the material that might be passed back to its originator (such as *cookies, bots*, and other such standardized data gathering mechanisms).
5. The ISP must conform to any conditions or requirements placed on the material by its originator (such as limited access or payment-for-use provisions)
6. The ISP must promptly remove any cached copy should the original be deemed infringing.

Transmission and routing

1. The transmission or routing of infringing material must have been originated by someone other than the ISP (such as one of its users).
2. The ISP must not be actively involved in the transmission or routing process (again, the term *automatic technical process* pops back up, and this provides cover for such routine user activities as web site maintenance, posting, uploading/downloading of files, or anything else in which the ISP does not have a direct influence).
3. The ISP must not be actively involved in selecting the recipients of any infringing material (i.e. an *automatic technical process* selects the recipients instead of direct action on the part of the ISP).
4. The ISP does not copy the infringing material for anyone other than the user(s) involved, nor do they maintain any copy of said material for any period longer than is reasonable for normal communications processes.
5. The ISP does not modifiy any material that passes through its system in any way.

Once the ISP is in adherence to the guidelines laid down in the DMCA, then they qualify for "safe harbor" protection as defined under the appropriate sections of copyright law (17 USC 512.n).

A MATTER OF PERCEPTION

Does this mean that your ISP is now required to have the "software police" monitor any and all activities in order to ferret out any perceived offenders? Of course not. Remember, an ISP is by its nature a passive business. They are not under any obligation to monitor everything you do with their service, nor would it be cost-effective to do so. Most assume good faith practices on the part of their users, and such is quite common across the Internet. In other words, they won't mess with you unless you do something that they don't like. Their passiveness ends if and when they determine that you are doing something not in accordance with their policies. Any form of perceived intellectual property infringement is almost always a violation of those policies (or should be - *wink*). *Once they know that* ***you might*** *be breaking the law, then your ISP* ***is required by law*** *to do something about it then and there, else* ***they will*** *become legally liable for your actions*. It is often impossible for an offended party to track down the source of the offense if it was performed on the Internet; however, the ISP is almost always known and therefore a legitimate legal target. If they can't prosecute the offender, then they can and will prosecute the ISP, unless the ISP can show that they have taken action against the offense in accordance with federal law. This is called *vicarious infringement* in legal circles, or more commonly *contributory infringement*. In plain language, it means that they can hold the ISP as an accomplice to the offense. The last thing that an ISP wants is one or more intellectual property violation charges laid at its doorstep, so you can bet that they will act as soon as they hear about any such acts on their service. That's why you might log on one day to find that your favorite emulation site has suddenly gone down without warning. The ISP took them down in response to an official complaint about something that was on the site. It doesn't matter what it was, or whether or not the complaint was based in fact. If an ISP perceives that one of their hosted sites might be violating official policy, regardless of whether or not that perception is based in fact, then they have every legal right to shut the site down.

So how do you, the emusite webmaster, find out that you have offended your ISP? How do you then deal with it? The following procedure will sound familiar to anybody whose ISP has shut down their emulation-related site since the passage of the DMCA, and is in fact the procedure that has been established by that law:

1. The ISP must immediately remove any and all access to the infringing material in question.
2. The ISP must immediately serve notice to the infringed party that they have taken appropriate action in their regard.
3. The ISP must also promptly serve notice to the infringing user(s) that action has been taken against them. They must describe the nature of the/each infringement, what action was taken in regards to the/each infringement, and finally what additional actions will be taken in

accordance with ISP policy should the user(s) persist in infringing behavior.

4. The user(s) in question has/have the right to file a counter notice, ***under penalty of perjury***, to prove that the material or activities in question are not infringing with regards to the intellectual property rights of the infringed party, unless the matter is referred to the courts. If the user(s) can prove their case to the satisfaction of the ISP, then the ISP must immediately inform the infringed party that the infringing material is being restored, state why it is being restored, and then restore access to said material within fourteen (14) business days of notifying the infringed party. If the infringed party elects to file a court action over the material in question, then the ISP has no obligation to restore said material.

While this procedure does not keep the courts from becoming involved should any infringed party press the issue, it does limit any liability to which the ISP in question can be held. After all, they are acting only in accord with the strict letter of the law (17 USC 512), and as such any action that can be taken against them by an infringed party is limited by that law. *If the ISP complies with the DMCA with regards to any infringing material on their system, then they cannot be held liable for that material. It is the infringing user(s) that must held liable for said material, as the user(s) are the one(s) truly at fault.*

INTROSPECTION

Needless to say, the operation of an Internet site devoted to emulation is a bit tricky due to the various legal complications involved. I know that people are going to do it anyway, and most vendors won't object so long as you keep your nose clean. So what have we learned in our look at emulation and the Internet?

> *It is within the rights of the owner of any patented, copyrighted, or trademarked material to object to its use on an emulation site without prior approval*.
>
> This can take many forms, but almost all of these fall within the "four basic food groups" of Internet emulation - emulators, "ROMs," patches, links - or one of the "side dishes," such as documentation and utilities. Case law and subsequent legislation have decreed that the Internet is a legally governable forum, and perceived intellectual property violation can be taken against any Internet-based infringement.
>
> *It is not illegal to distribute a freeware emulator that is legal in nature*.
>
> The two things that determine the right of free distribution are the legal status and the economic impact of the emulator in question.

Once those two issues are resolved to the satisfaction of all parties involved, then the emulator may be freely distributed without fear of prosecution. It is illegal to distribute a current or former commercial emulator without authorization, however, as the distribution of commercial products regardless of age is protected by copyright law.

"ROMz sitez" and any other kind of site making copyrighted "ROMs" available without authorization are clearly illegal.

Case law (Sega v. Maphia, 1994) has clearly determined the factors by which such sites are legally culpable for intellectual property infringement. Such sites are prosecutable under appropriate federal, state, and local statutes, as well as any "long-arm" statutes that may come into play, for actions that are clear and deliberate violations of various intellectual property laws. The only kinds of "ROMs" that are safe to post are those of either public domain or inactive commercial in nature.

Freeware support for emulation is governed by the same intellectual property laws that govern the emulator and its software base.

The principle of emulation is no excuse for voiding the intellectual property rights of a program author or vendor by producing an infringing patch, infringing copies of documentation, or by coding an infringing utility. Vendors have as much right to object to these forms of emulation support as they do to emulators and "ROMs."

The practice of using links that involve the improper use of or connection to unauthorized material, regardless of the site or origin or usage, can be considered a form of intellectual property infringement.

A link serves as the gateway to further broaden one's base of knowledge. Because of its nature, it is subject to the same restrictions as any other format that interacts with various forms of intellectual property. There are four ways to create an infringing link: accessing unauthorized material on any site (the traditional definition), including infringing material within the link itself (such as a trademark), using an unauthorized process to enable the link (such as patented computer code), and using the link to enable unauthorized access to portions of a copyrighted site regardless of the legitimacy of the link's actual destination.

> *The same restrictions that govern claims of "fair use" in the physical world also apply to the Internet*.
>
> In general, any activity that would not past the "fair use" test with regards to possible infringement would also be considered an infringement on the Internet. The "water's edge" theory about unhindered distribution of material on the Internet was never valid in the strictest legal sense and has long since been laid to rest. Claims of activity in the public good must also be balanced against the other three parts of the "fair use" test before one can even consider qualification for the "safe harbor" provision of copyright law. The principle of "fair use" of copyrighted or trademarked material on the Internet is recognized, but this does not extend to wanton unauthorized use.
>
> *Internet service providers (ISPs) have the legal right to block access to parts of or to shut down entirely any site once they are informed that it might contain some form of possible intellectual property infringement*.
>
> The ISP is required by law to act once they are aware of such a claim, otherwise they also become legally liable for any and all infringement taking place on the offending site that they are hosting. There is a set procedure under law whereby an infringed party must first notify the ISP of possible infringement before the ISP can take action. Any party operating a site whose access is blocked either in part or in full by the ISP for possible intellectual property infringement has the right under law and under penalty of perjury to counter the infringed party's claim. Should the counter-claim prove valid and the infringed party elects not to pursue the issue in court, then the ISP is required by law to restore full access to any blocked material.

This ends our formal discussion legality of the issues of emulation. With this, we also end our overview of the current state of emulation, as it has developed from its beginnings and gradually moved to the point where it sits today. You now know that emulation is legal, provided that the appropriate intellectual property and economic impact issues are properly addressed. You now know that obtaining, using, and providing software for an emulator is also legal, provided this is done under strict guidelines - more restrictive in certain cases, such as console and arcade videogames, than for regular computer software. Now that we know exactly where the law and the courts stand in this regard, where do we go from here? Where does emulation stand with respect to the rest of the personal computer industry? Given our past experiences with personal computer emulation and current technological trends, how should dedicated system vendors, such as those for console and arcade videogames, respond to emulation within their venues? These topics and more are what we shall address in our next series of discussions. The so-called "great emulation debate" will kick off with the emufans getting their say first, followed next by the vendors, and then I will close with a final editorial relating my insights into the present and future of "this strange realm" known as emulation.

REVIEW QUESTIONS

1. How has the Internet made an impact on the emulation community? In what ways is this impact good or bad?

2. What are the "four basic food groups" of emulation? What are the additional "side dishes" that may be offered?

3. Why do many vendors seemingly choose not to object to the effect on their products by the emulation scene? Does their reaction void their intellectual property rights? Why or why not?

4. How does the law deal with possible intellectual property infringement on the Internet? Can you name and describe the legal test used to determine proper jurisdiction for prosecution of such cases?

5. How can the legal status of a freeware emulator come into question? Which of these factors is the most important insofar as the original system vendor is concerned? Why?

6. Is it possible to pirate an emulator? Why or why not?

7. Which court case serves as the basis for the legal contention that "ROMz sitez" are illegal? Can you describe some of the five areas in which it has a direct impact on emulation-related activities within the Internet?

8. As far as the Internet emulation community is concerned, what are the three different kinds of "ROM" and what is the legal status of each?

9. What are the legal concerns with regards to the availability of emulation-related material other than emulators and "ROMs" on the Internet?

10. Is it legal to provide links on an Internet emulation site? What are the four ways in which such a link might turn out to be an infringing link?

11. How does one deal with "fair use" of copyrighted material on the Internet?

12. Can you describe the three specific exceptions for the "fair use" of trademarks on the Internet?

13. Under the terms of the DMCA, what are the four areas in which an ISP can be held legally liable for intellectual property infringement? Can you describe how an ISP can avoid such a charge with regards to at least one of these areas?

14. What is the proper procedure for an ISP to follow in dealing with a possibly infringing Internet site that it hosts?

THOUGHTS TO PONDER

1. Why was The Dump: Genesis shut down? How can this example be applied to other kinds of emulation-themed Internet sites?

2. Assuming that the author's theory about vendor indifference to emulation is incorrect, what are some other reasons certain vendors might not object to the emulation scene?

3. How might an original vendor use the example of Sega v. MAPHIA, or other examples of law and case precedent, to build a reasonable claim of intellectual property infringement against different kinds of emulation-themed sites?

4. With regards to the fair use Internet exceptions for trademarks, how could these exceptions be properly applied to an emulation site?

5. Will software patent protection ever become a valid issue with regards to the Internet? Why or why not?

6. Is there such a thing as "fair use" of materials or processes protected by patent law? Why or why not?

Module 3: The Implications

Part 1 - In Defense of Programming Freedom

The problem isn't the rise of emulators. It's that there are too few of them.

T. Liam McDonald, "You Will Be Emulated," *MaximumPC*, September 1999

NEXTGEN VINDICATION

On 24 August 1999, Judge Charles Legge of the 9th District Federal Court in San Francisco, CA, United States handed down what would prove to be the third and final ruling in the preliminary round of **Sony v. bleem LLC**, one of two landmark cases initiated that year with regards to videogame emulation. Sony Corporation, the well-known international multimedia conglomerate originally founded in Japan, had sought for a third restraining order against bleem LLC, the makers of the *bleem!* PlayStation emulator for IBM PC compatible personal computers. They accused the company of numerous infringements and violations of Sony's intellectual property with regards to the origins and operational functions of *bleem!* Chief among their contentions was the same that they had successfully used against Connectix (the makers of *Virtual Game Station* for the Apple Macintosh) earlier that month, which was that *bleem!* unlawfully duplicated portions of proprietary Sony object code stored within its Papillion ROM (the PlayStation BIOS). In order to counter Sony's claim, bleem LLC company president David Herpolsheimer made available to the court both the source and object code forms of *bleem!*, as well as every single document in his company's possession with regards to its origins, development, and intended distribution. After considering the matter for some time, Judge Legge ruled against Sony's request for a third injunction over the emulator, citing bleem LLC's overwhelming evidence in favor of original development, plus the fact that Sony had been unable to successfully prove any of its claims. *bleem!* had finally cleared all legal hurdles for unrestricted retail distribution and continued development up to the date of the actual civil action, which is scheduled to take place in April of 2000.

Herpolsheimer wasted no time in making his opinions known to the public on the affair. "This was a major hurdle for us," he said in an interview with *GameSpot News*. "Sony had all the weapons on their side this time ... and they still couldn't make a case against *bleem!*" In another, somewhat earlier interview with *EGM*, he vented his feelings on Sony's continuing efforts at stopping his emulator's market penetration. "We're real confident. It's almost silly at this point - the attorneys feel like there's just no point in going on. It's like, 'we've won.' They feel really good about it ... [but Sony] could drag this thing out forever. I hope at some point they'll stop." And in an open letter to users of *bleem!* on the company's Web site, the following statement was posted. "This is a great step for *bleem!* and we want to thank all of you, because without you, we would have never made it this far.... Thanks again!"

Emulation has now come full circle. **Sony v. bleem LLC** and **Sony v. Connectix** are to the videogame console industry what the *A-Max* affair was to the personal computer industry a decade ago. Not only do they uphold the legality of emulation as a non-infringing technology, they also help to reinforce the guidelines whereby unauthorized emulation of a proprietary vendor product can be lawfully conducted. While proprietary-minded vendors may not be able to stop emulation, they certainly have the right to object to what they perceive as overreaching infringements on the part of the emuscene - or do they? There are those both within and without the emuscene who would contend otherwise.

It is time to revisit the controversy over emulation in the form of a formal "us-vs.-then" debate. A lot of this material will sound familiar, as we have touched upon it at times in our earlier discussions to varying degrees. This time, though, I will stand back for the most part and act in the role of moderator instead of author. I will let the opposing sides speak for themselves, and let you be the judge as to the reason and logic of their arguments. First we will hear from the emuscene, as their opinions are often ignored or distorted by mainstream reporting. Next will come the vendors, particularly from those who feel that unauthorized emulation represents a direct threat to their bottom line. Finally, in the third and last part of this module, I will resume my role as author and share with you my thoughts and experiences with regards to the topic of emulation.

MARKET PENETRATION

The one claim that offended vendors make against emulation time and again is perhaps their strongest one - that emulation adversely affects their intended market share. The emuscene sees things differently than does the vendor community, as one might expect, so let's hear what they have to say.

> *Emulation is at best a minimal threat to vendor profits.*
>
> In fact, a very strong case can be made that emulators have the potential to actually boost a vendor's profits. Consider the words of T. Liam McDonald in his recent editorial for *MaximumPC*, "You Will Be Emulated."
>
> > How does [emulation] hurt the developers, who now have an entirely new market (PC owners) to sell to? Not one iota. It's the best thing that could happen to publishers. Or course, it may cut into PlayStation *hardware* sales, but how serious is that? Everybody knows that the profit is in the software, not the hardware. When Sony introduced the PlayStation, the machine was sold for close to cost in order to establish a user base for the true money train: the

> games. Thus their arguments for the damage emulators cause are hollow.

Sony's sales approach is actually quite common and is not restricted to the videogame industry alone. Don Levine, the creator of G.I.Joe, calls it the razor-and-blades approach: "If you sell someone a razor, then they're going to need the blades." Thus it was that Hasbro first sold its "razor" back in 1964 - in this case one of four different basic G.I.Joe action figures, and then sold many different "blades" - multiple different accessory sets - to go with them. Those blades can be quite profitable indeed, with net profits in the multi-millions of dollars range. It is an open secret that most videogame console vendors, like Sony, sell the actual console at or just above cost (the "razor"), because they expect to make up the difference in program sales (the "blades"). A recent *GameWEEK* editorial on the impending arrival of the Nintendo Dolphin points out that Sony is claiming just over one-third of the end-retail cost of PlayStation games in royalties and other related indirect charges. If you think that's high, consider Nintendo, who claim back in royalties and other fees almost half the retail price of N64 games.

It is clear that the vendors are making money hand-over-fist in software sales, and third-party emulators designed to work with original system software (such as *bleem!* and *Virtual Game Station*) can actually boost those sales by opening up new markets hereforeto denied to the original system vendor. IBM PC compatible and Mac owners who wouldn't have given PlayStation games a second thought now have the option to do so - and many are, judging from initial sales figures for *bleem!* and *Virtual Game Station*. The same holds true for older systems as well, since many an emulator user goes on afterwards to seek out the original console and the actual original program base. Need proof? Just visit eBay's Computer Games section sometime and see the prices that both original systems and original copies of its programs are fetching. It will take only a few minutes browsing eBay's site to realize that the market for classic computing is rather sizable. A threat to profits? Not likely.

Emulation permits experiences outside of a proprietary system.

McDonald's comments also bring up another point that the vendors don't care to discuss - that emulation actually allows users to enjoy programs outside of their intended platform. Mark Asher, writing for *C|NET GameCenter*, puts it this way: "What this denial of injunction [for Sony] means for gamers is simple: freedom of choice." Howard Wen, writing for *Salon*, calls it "tearing down the barriers for code among proprietary formats," and has a lot more to say about it in his article "Why Emulators Make Video-Game Makers Quake."

> Emulators are finally bringing into question the need for [vendors'] specialized gaming platforms and dedicated gaming boxes in the first place. If every video game console is easily emulated on a personal computer, then what's the point in even having specialized formats in the first place?
>
> Emulators herald the end of the era of the proprietary video game console because they render such dedicated gaming boxes technically superfluous. Emulation progress improves, PC hardware technology advances relentlessly - and the notion that games must be played on the console hardware for which they were developed is becoming as antiquated as an old Atari game system. Just as information wants to be free, code doesn't want to be restricted to running on a single format. That, not simple piracy, is what emulation is all about - and it's why the videogame companies want to squash it.

Sales figures for both of the commercial PlayStation emulators far exceeded initial projections, with bleem LLC reporting over 50,000 units of *bleem!* sold in the first month and Connectix reporting over US$1 million in *Virtual Game Station* sales every week for the first three weeks it was on the market. To quote Marc Saltzman, a regular columnist for *C|NET GameCenter,* "Clearly there is a market for commercial emulation software." L. Bank, author of the *C.A.G.E.* arcade emulator, words his thoughts on the matter rather bluntly.

> Game manufacturers are only interested in the bottom line. They don't think releasing their old games in new forms is worth their time; thus, the only way to play these games is through emulation.

There is also another factor to consider, and that is greater freedom in developing products for specialized systems. Again, Howard Wen.

> Additionally, computer and [video]game hardware companies might want to informally support the efforts of emu programmers [for development purposes]. That's what U.S. Robotics did earlier this year when it included the Palm computing device emulator *Copilot*

> in its development tools packet. The work of an independent programmer, *Copilot* benefits U.S. Robotics by encouraging software development for its *PalmPilot* line of personal digital assistants [PDAs]. While the emuscene is [busy] tearing down the barriers for code among proprietary formats, it can also support the intrinsic value of a piece of hardware by opening up development for it and extending its market life.

This is exactly what certain proprietary-minded vendors such as Sony and Nintendo fear, because they have a nice little captive niche market for specialized development consoles and software development kits (SDKs) for which they can earn hundreds or even thousands of dollars apiece. Introduce an inexpensive yet highly effective emulator, and you eliminate those captive profits, as bleem LLC's David Herpolsheimer indirectly asserted when he was recently interviewed for *GameSpot News*. The report is by Sam Kennedy for Ziff-Davis Publishing.

> Several game developers have taken a strong liking to *bleem!*, according to Herpolsheimer. He's had several discussions with developers who have informed him that they have been using his product to develop or test their titles due to their blue PlayStations (development units used to play non-final titles) constantly failing to operate. He mentioned that developers designing titles to work on *bleem!* will automatically be fully functional on the PlayStation hardware with little or no tweaking. Herpolsheimer also pointed out that one of the largest developers of PlayStation software - one whose name he couldn't reveal - was developing with *bleem!* in mind and had just ordered several copies of *bleem!* to help test their products.

Perhaps it is as Jean Guillaume Paradis said in a recent *RPGamer* posting. "Maybe the reason Sony won't [release their own emulator] is because they simply can't ... maybe it's because they're too stupid."

Emulation does not promote software piracy.

The point that Howard Wen raised earlier about specialized systems is an excellent one and should not be ignored. The reason that proprietary-minded vendors are so afraid of emulation is that it is the harbinger of their doom. People will not spend money on their pricey proprietary systems when they can get the appropriate emulators for a lot less. Naturally, these vendors do not plan on giving in so easily, and the quickest and easiest attack is to issue blanket charges of software piracy. Again, we hear from the succinct wisdom of T. Liam McDonald, who briefly addresses this in "You Will Be Emulated."

> The statement that the only purpose of emulators is "to play illegally copied games from the Internet" is just a lie. Once you buy a game, you can play it wherever and whenever you want. It's yours. If you want to patch together a lawnmower, a Watchman, and a pair of ice tongs to play *Sonic Cruises for Hookers*, or anything else that [the vendors are] publishing, that's your right.

Howard Wen examined the piracy angle a different way in his *Salon* article, "New Life for Old Games."

> ... for the copyright holders of pirated game ROMs, going after the emu programmers may be understandable but it misses the point. Extending the pirate analogy, emu programming at worst is akin to building ships favored by pirates. You can't stop the shipbuilders, because there are perfectly legitimate reasons for using ships - just as there are for emulators. Emulators enable code to be use across various platforms; they allow archived personal and business information - such as data that can only be read on an antiquated spreadsheet program that, in turn, ran on a long-defunct computer operating system - to be accessed on modern PCs.

One can argue (and Nintendo resolutely contends) that the sole purpose of emulation is software piracy, but it's a blatant lie - just as McDonald asserts. Emulation has served a wide variety of functions and fulfilled a variety of needs throughout its long history. For example, if (as Nintendo contends) the sole purpose of emulation is to promote software piracy, then any product that contains one or more emulators must also promote software piracy. Microsoft will be the first to tell you that its *Windows* operating system contains several emulators, as do MacOS, Unix/Linux, BeOS, NeXTStep, and so on, all to allow the use of various functions that may not be supported by the platforms on which they may be installed. Do these products promote software

> piracy? Certainly not. Let's try another angle. There are certain videogame consoles that are "guilty" of including hardware emulators for back-compatibility issues. The Sega Genesis/MegaDrive may come immediately to mind, but others also suffer from this "affliction" and Sony's new PlayStation 2 will be the latest. If these consoles include emulation hardware, then they must promote software piracy, right? That is as ridiculous as it sounds. The emuscene responds to Nintendo's charge by quoting the space probe Nomad from the *Star Trek* episode, "The Changeling."
>
> > Non sequitur - your facts are uncoordinated.

INTELLECTUAL PROPERTY

The next general charge that the vendors bring against the emuscene usually falls in the area of intellectual property infringement - that an emulator somehow or in some way infringes upon a vendor's patents, copyrights, or trademarks. The emuscene responds that such an act is possible with any program if its authors will it, but that is not their desire. They are looking to honor the vendors, not to walk all over them.

> *Emulation does not infringe upon a vendor's intellectual property rights.*
>
> It may shock the vendors to know that the vast majority of the emuscene is actually concerned about honoring the rights of the vendors whenever possible. Kevin Brisley, author of the *Replay* arcade emulator, puts it this way.
>
> > The bottom line is that companies have a right to defend their copyrights. If they want to force a ROM archive site to remove their images, [then] they're perfectly within their rights. However, my personal feeling is that they shouldn't alienate their past and current customers by quibbling over programs that haven't generated any income for them over the past 10 years."
>
> Along these lines, it should be noted that the last thing on the mind of many emucoders out there is the notion of "being exclusive," which is to the emuscene what a proprietary mindset is to the vendor community. Epsilon and RealityMan, the authors of the N64 emulator *UltraHLE*, took a lot of heat from the emuscene for not releasing their source. Six months later, there are several

different N64 emulators, and one or two of these even have open source to boot. A proprietary mindset invites attack from others, and the emuscene's relationship with the vendors is no exception. As Howard Wen notes in *Salon*,

> ...the programming skills of the emu developers will continue to undermine the incentives to make a new hardware format proprietary.

The same is true whether you are a proprietary-minded vendor or an exclusive-minded emucoder. Cameron MacMillian is an Irish computer consultant who often does beta testing for the emuscene, and was recently quoted in *Salon* on the subject of exclusivity within the emuscene itself.

> Realistically, there's no point in being competitive. If one person withholds information, chances are a legion of [other] coders will eventually figure it out for themselves. Besides, what *really* is the prize for being first? Having your name in cathode ahead of everybody else's? Ultimately, it just delays the project.

Emulation gives more respect to the original authors and programmers than do the vendors of their products.

If you ever get time, then you need to drop by Emulators Unlimited (http://www.emuunlim.com) and go to the part of their site labeled, "The Dot Eaters." What you will find is a concise, well-written and illustrated history of the early days of computer videogames - from their beginning back in the early 1950s all the way to the "great crash" of 1983. It spares nothing on detail - dates, places, names, events, and so on. Many a programmer is listed there, along with the programs that they did, and they get far more credit than the vendors for whom they worked ever gave them during that time of old.

All of this ... and on an emulation site, too ....

It's not just limited to the mainstream emuscene, either. You can find Internet sites dedicated to all different kinds of software, or to just a single program, and many give ample credit to the authors - along with a copy of the program and the requisite emulator to run it.

Some of these programs haven't been available in decades ... imagine that ....

Odd, that this could be so, because one gets the impression from the vendors that these "pirates" never credit those who made

> the programs in the first place. Then again, those vendors are the same group who insist on exercising their "work for hire" rights whenever possible, rarely pay good programmers what they're really worth, and often find ways of not giving them credit for their work. Find this impossible to believe? Then go ask the programmers. They'll tell you that finding programmer-friendly employers is almost impossible, and they end up spending a lot of their time doing "hack work" for low pay and little, if any, credit. Rare comes the opportunity when they can tap the time and resources to be truly creative and produce the kind of product that users really want. Go ask them. They'll tell you the truth - not the vendor line.
>
> Good programmers never fade away. They just earn newfound appreciation from a whole new generation of fans, almost always without asking for it, and this gives them the impetus they need for new creations. Emulation is good for programming, and not the other way around.

OPERATIONAL CONCERNS

Another issue often raised by emulation critics is one of operational concerns. They fault the emulator for not working with this or that program, for running too slow or too fast, for failing to display subpixel cluster 2145 in high resolution mode on level 600 while engaging multiplayer gameplay, and so on. The emuscene is not blind to this criticism, and responds with two strong assertions of their own.

> *Emulation allows things that were not possible with the vendor's original product(s).*
>
> It can be claimed by the emuscene that classic computing has not only been successfully resurrected, but lives on in the form of emulation. In fact, as Howard Wen notes,
>
> > ... emulators for most of the old videogame hardware - including the Atari VCS, ColecoVision, and Nintendo Entertainment System - were perfected long ago. All of the major brand home computers from the late 1970s and early 1980s - systems from Commodore, Apple, Atari, Tandy, and Texas Instruments - have also been resurrected in software form. These days, most emu programmers work on tweaking and enhancing their emulators.

Tweaking, indeed! Many of these emulators are quite remarkable for the simple fact that you can do things with them that you could never do with the original hardware. Some of these were long desired by users at the time, and others have been added as a result of recent developments within the personal computer industry.

When Sardu of Bloodlust Software first announced that his team was going to add network play to *Callus*, their Capcom arcade emulator, everybody monitoring the emuscene thought he had lost his mind. His team pulled it off, the network play feature actually worked, and now network play is something that is increasingly becoming a standard feature among the more sophisticated videogame emulators. It should be noted in retrospect that none of the Capcom arcade games that *Callus* emulated supported network play. Now they do, but *only* under emulation. This is just one example of a new feature unavailable on the original system, but there are many, many more. Code the uncodeable, do the undoable, run the unrunable - in short, anticipate the impossible. This seems to be the operational maxim of the emuscene, and it is proven more and more with each passing day.

Emulation permits the freedom to use older, smaller, more efficiently coded, and often better working versions of a given computer program.

Brad Oliver, a student at Arizona State University who has worked with both the *M.A.M.E.* and *M.E.S.S.* emucoding teams, makes the following comment about the emuscene.

> I think hardware is still king, but emulation is proving [that] code is quite powerful in today's age of quantity over quality.

Interesting observation! It is an established fact that many user prefer older versions of a program in favor of newer ones. A variety of reasons are offered, with size, familiarity, and reliability being the three most commonly cited. Older incarnations are smaller, are better known to the user, and don't have any "new or improved features" to jeopardize the program's stability. For example, many IBM PC users still prefer *Microsoft Windows 95* or *Windows NT* to *Windows 98* for precisely those three reasons. *Win98* is massive, obtuse, and buggy compared to its older, smaller, better understood, and more reliable ancestors. Many of us who are in this camp will insist on continuing our use of *Win95* or *WinNT* until something better comes along or we jump operating systems entirely to, say, *Red Hat Linux*.

Working Designs recently ported and released in English a special version of *Lunar: Silver Star Story* for the Sony PlayStation. It is an impressive upgrade of the original Sega CD release and well worth the price, yet you can still find the original release selling for original retail (or often higher) in eBay's Computer Games section. Why? Was it that good of an RPG? Certainly! Then why not buy the PSX version? Two reasons are often cited by current (and would-be) *Lunar* aficionados: the Sega CD original is not copy-protected (the Sony PSX version is) which makes it easy to back up, and Sega CD emulation seems to be an imminent reality. That last bit is important to defenders of emulation, as original Sega CD consoles are quite rare nowadays. Having a Sega CD emulator would save wear and tear on the actual console or permit playing of the game should an original console prove to be unavailable. Furthermore, the emulator would be working with an original Sega CD version of *Lunar: Silver Star Story* in its original format, which would nicely dispense with any claim of software piracy.

It should also be noted that the so-called "improvements" are sometimes little more than slightly retooled versions of the original running on the same basic program engine and executing in exactly the same fashion. What changes exist are slight, usually consisting of altered databases and altered graphics. Another increasingly irritating vendor tactic is to release a working, albeit buggy version of a program so everybody will be forced to contact customer support for the fix. T. Manelli recently vented his ire on the topic when interviewed by Marc Saltzman for *C|NET GameCenter*.

> I'm sick and tired of companies like Sierra and E.A. Sports releasing the same old crap year after year, save maybe for a few games like *Half-Life* or the *NHL Hockey* series. They expect gamers to shell out US$40 or US$50 for a computer game when it's not even finished, or there will be another version released less than a year later (in Sierra's case, six months later). I hope the rise [of emulators] gives all the suits at these fat companies a wake-up call, so they'll realize we've had it with being ripped off with their incomplete and rehashed games.

PRODUCT EXPERIENCE

The question of product experience was first raised by Sony in the lawsuit against Connectix over the *Virtual Game Station*, but it has since been echoed by both other vendors and original programmers alike. They seem to be of the notion that if a program is not being executed on the original hardware using the original interfaces (joysticks, arcade console, etc.,) then the ultimate result is an inferior product experience. The emuscene does not agree with this contention, so their response is understandable.

Emulation enhances the product experience.

The rise of emulation technology permits users to experience a vendor's products in ways not possible before - a point that we have previously discussed. There are those who would say that making these new options available actually enhance the product experience, because it allows the user to become involved with the product in new and unique ways beyond those conceived by the vendor in the first place. Howard Wen puts it this way in his *Salon* editorial.

> Despite [claims of emulator-enabled piracy], something other than copyright and intellectual property violation is bugging Sony and Nintendo. These companies are fighting to reinforce the boundary separating the console and computer gaming worlds.

Naturally, loss of that boundary is something that like-minded vendors do not want to happen. If users become accustomed to the emuscene and both the promise and versatility that it offers to the product experience, then they will begin demanding it of future products. This will drive up product costs, thereby affecting the bottom line and reducing profits. Better to destroy emulation, thereby removing it as a threat and keeping the user base enslaved to a proprietary system irrespective of what the desired product experience might be. It is the *vendor's* product, not the user's - therefore the *vendor* should dictate the product experience instead of being dictated to by the user. That is the impression that the emuscene is getting from the vendors in this regard.

Emulation allows a user to experience the "missing parts" that were delivered to other markets within a given product line.

Many who regularly monitor the English language videogame scene were surprised to see a four-year-old Japanese RPG for the Super Famicom (aka Super Nintendo) break into the American "top ten" lists for a time. The program was *Seiken Densetsu 3* by the noted RPG maker SquareSoft, but is better known to English-speaking fans as *Secret of Mana 2*. According to SNES historians, *SD3* (as it will hereafter be abbreviated) was one of many excellent RPGs that never made it to American shores due largely to the bungling of SquareSoft, its vendor. SquareSoft cancelled the translation of *SD3* when it was only two-thirds complete, opting instead to release the inferior *Secret of Evermore*. The market demand was certainly there for *SD3*, but the vendor did not see fit to port the product, with the continued cost of translation cited as the chief factor.

Enter emulation into the picture some three years later. The reason that *SD3* jumped into the charts is that a team of dedicated SNES fans founded their own group, translated the *entire game* into English, created a translation patch for the "ROM" version of the game, and then put it out on the Internet. For the first time, fans of the first *Secret of Mana* who had been clamoring for years to SquareSoft to finish the English port and release it could now finally enjoy the product that they had wanted and pleaded for to SquareSoft for so long. Was the wait worth it? "**YES!"** they almost unanimously shout. Dan Duett, writing for *RPGamer*, had this to say about the whole *SD3* affair.

> Now anyone who is involved in the emulation world, such as myself, and particularly the translation world, knows that the translation of [*SD3*] is a massive achievement. This was a project which took over a year in itself and its history included many failed attempts. Now someone finally comes along and translates the game - not to make profit, [nor] gain attention, but to simply bring a game over to the States that should've been here 4 years ago. To top it off they overcome [sic] several complicated problems in the translation project, and even perform a better translation than many SquareSoft games of today. How can one say they're at fault? They should be congratulated for doing something [that needed] to be done ....
>
> ... do I believe that anyone who plays [the translated *SD3* ROM] is breaking the law? No. They're simply playing a game that should have been played a long, long time ago.

Brad Carsten, also writing for *RPGamer*, put the matter more succinctly.

> ...I believe emulation is a great way to play games that you couldn't play otherwise. [The vendors aren't] losing any money because I wouldn't be able to buy the games anyway.

Dementer of the translation group DemiForce agrees.

> Without emulators, I never would have found this niche. It enabled me to take games that were far too rare to ever dream of getting my hands on in the physical world, play them, and translate them into English ... thanks to people like me, many games that were rejected for professional translation and release in English-speaking countries now can be understood and played by the entire emulation community (and those who own console copier devices).

NOSTALGIC YEARNING

Finally, there is an issue that a lot of us older computer and videogame users must raise: emulation satisfies and preserves our shared past for future generations to experience on their own terms. "Today's games suck," says L. Bank of *C.A.G.E.* fame, and many of us older users believe he makes a valid point.

Emulation lets older computer users enjoy the products that they know and love.

For those of us who grew up as part of the personal computer revolution back during the 1970s and on through to the mid-1990s, there is something to be said for reliving the excitement of those days. Brad Oliver puts it this way.

> [What this is] really all about [is] playing the games and reliving good memories. You can emulate the games, but you can't emulate the experiences. That sounds ridiculously cheesy, but hey - so was Dig Dug.

Chris Baker, writing for the *Austin Chronicle*, has a slightly different take on the matter:

> Maybe like all other retrotrends - clothing, music, film - the rise in the interest of retrogames is just another example for the need for the familiar comfort of childhood. Or, maybe when you get down to it, the games were just better then.

There is an ingrained human instinct to yearn for what many a psychoanalyst has termed "the simplicity of the past." This applies to computer systems just the same as it does for anything else. If I may break in and indulge for a moment in my own memories, I like emulation because I *know* those systems and I *know* that software like the back of my hand. Obsolete? Certainly. The games of yesteryear are still fun because of their deceptive simplicity. They may look old, grainy or blocky, painfully 2D, and ugly as sin compared to today's new, razor-sharp, texture-mapped and antialiased polygonal 3D "gaming experiences," but more than one has subtleties to its gameplay that escape even the slickest-looking modern offering.

You know, as for myself, I get the biggest kick sharing the past with the next generation of young gamers as they discover for themselves the old software that made possible the offerings from which they can choose nowadays. And you know what? They grow to love 'em just as much as they do the new stuff. Some even prefer the old stuff over new offerings. Case to point - my nephew must have every single rollerblade or skateboarding title ever made for the PlayStation or N64, but do you know what is his favorite game of this genre? *Skitchin'*, by Electronic Arts for the Sega Genesis. He used to rent it from the local Hastings on a regular basis until he accidentally broke the RF modulator on his console. His parents bought him a PlayStation that Christmas, and a N64 a year later, but he never forgot that game. He rediscovered his love for the game after I became involved in the Sega emulation scene and obtained the cart dump, and it was all I could do to keep him off of my computer on the weekends he would drop by to visit. This year, he made it clear that I *absolutely had* to buy him an original cart for his Genesis after I had repaired it and got it working again. While I now have his Genesis (his parents recently gave it to me), he still plays *Skitchin'* for hours on end every time he drops by to visit. Case closed.

Emulation preserves the past.

In the real world, hardware decays with the passage of time. There will eventually come a point where original systems will not work anymore because they are too old. What then? We open the discussion on this final topic with the words of Chris Baker, who makes an interesting observation in this regard:

> Retrogame fans, like all fan bases, combine a childlike sense of enthusiasm with the grownup's desire to understand, articulate, and pass on that enthusiasm. Ultimately, it's the intensity of the fans' enthusiasm that saves retrogames from simply becoming a piece of history.

Again, the words of T. Liam McDonald.

> Gaming, like any form of entertainment, has a heritage that is completely out of reach for today's consumers. A few issues ago [in *MaximumPC*], I lamented the industry's practice of making great games instantly disposable by rushing to the "next big things." The early days of Atari, Commodore, Sega, Intellivision, Apple II, Jaguar, Lynx, and ColecoVision are lost to current and future generations without emulation. N64 and PlayStation will one day suffer the same fate as NES and Sega Genesis.

And now a word from the legendary creator of *M.A.M.E.,* Nicola Salmora.

> Most importantly, our work is culturally significant: we are trying to preserve all these games for future generations. Emulation is the only viable option, because it's too hard to keep the original boards in working order .... If we don't do it now, and do it quickly, many games [are at] risk [of] disappearing forever.

Add to that the sentiments of the mysterious "Dave," webmaster of the premier emulation website Dave's Video Game Classics.

> I feel [emulation] benefits developers, especially of older games, simply because it gets people interested in these old games that would otherwise be forgotten.

Let us also add to that the sentiments of emu programmer Kevin Brisley.

> I would love to see companies offer their ROM images for a nominal fee.

So would many of us, Kevin! In fact some vendors and original program authors have even taken your idea and done it one better - they offer their old programs in "ROM" format at no charge! Some do it for the sake of charity, while some do it to promote new and enhanced versions of the product for newer systems. These in turn are copied by the emusites, mainly because not every emufan has the habit of dropping by vendor websites. As a result, the emuscene gains legitimate commercial ROMs (which I term "inactive commercial") to use with the appropriate emulators.

Howard Wen touches upon this idea in his *Salon* article.

> ... What better way to plug your company's new games than by giving old titles away that are no longer bringing in significant, if any, revenue? ... The company could provide users with access to the ROMs only after seeing online promotions for the new game, filling out a demographic survey, and/or agreeing to be put on an email list announcing new products.

Let's be fair to the vendors, now. Many were re-releasing classics from their archives long before the current emulation boom, and many have jumped on board since. Atari ... Williams ... Midway ... Namco ... Konami ... Capcom ... Sega ... all of these have re-released classic videogame ports or packs within the past decade for newer systems, and it surprises no one that an increasing number of them are actually employing emulation in some form to run the original (or sometimes slightly tailored) videogame program code. At least one of these, the *Sega Smash Pack*, was derived directly from an emulator (Steve Snake's *KGen*) that was available to the emuscene long before the commercial product arrived. Even so, obstinate holdouts like Sony and Nintendo remain, and they do little to champion their own cause by unfairly targeting those who prefer their older products over their latest offerings.

INTROSPECTION

It is clear in the minds of everyone involved in the emuscene that emulation is a necessary and vital part of the computer industry. It is a technology that should not be shackled by the chains of vendor desire, nor should it be hobbled by the designs of vendor strategy. The vendors have yet to raise a valid charge against the emuscene, and it is because they cannot. They know the truth, but would deny it not only to themselves but to the rest of the computer industry as well. Like stubborn children, they refuse to accept that which they hold unacceptable.

Face it, vendors.

YOU WILL BE EMULATED.

REVIEW QUESTIONS

1. Why is the "razor-and-blades" approach to product marketing so effective? Which makes more money, the "razor" or the "blades?" Why?

2. Why is there such a sizable market for classic computing?

3. What is the biggest threat that emulation poses to proprietary-minded vendors, and why?

4. Explain why the emuscene feels that emulation does not promote piracy.

5. How does the age of a given product factor into its treatment by the emuscene?

6. Does emulation truly affect the perceived "product experience?" How? Is this necessarily a good thing?

7. How does emulation enable users to escape the bounds of a vendor's fixed market for a given product?

8. Why do some computer users prefer old products over new ones? Give examples to justify your answer.

9. Why is the emuscene so important to the preservation of computer history?

10. Why should the vendors embrace emulation?

THOUGHTS TO PONDER

1. Why are vendor royalties for their products set so high?

2. Is there any truth to the charge that emulation promotes software piracy? Why or why not?

3. Is recognition of original authorship a valid part of the great emulation debate? Why or why not?

4. Why do perfectly good products get so-called "upgrades" that are little more than fancy make-overs?

5. Who should dictate the "product experience" - the vendor or the user? Why?

6. Do you see the rise vendor-sponsored ROM sites as a good thing? Why or why not?

Module Three: The Implications

Part 2 - In Defense of Corporate Assets

> *Our experience shows that many computer hacker suspects are no longer misguided teenagers mischievously playing games with their computers in their bedrooms. Some are now high tech computer operators using computers to engage in unlawful conduct. The losses to the American public in this case are expected to be significant. The Secret Service takes computer crime very seriously, and we will continue to investigate aggressively those crimes which threaten our nation's businesses and government services.*
>
> Gary Jenkins, U.S. Secret Service - Assistant Director, *Operation Sun Devil*

NEXTGEN ELIMINATION

On 25 March 1999, Sony Corporation filed a intellectual property infringement lawsuit under the terms of the Online Service Providers Liability Limitation Act (17 USC 512) against Action World, Inc., a well-known Internet service provider (ISP). ActionWorld was the parent company of UGO Networks, Inc. (formerly United Gamers Online), which is an Internet gaming community comprised of some 100 or so Web sites dedicated to computer gaming in just about every form conceivable. Unfortunately for ActionWorld, one of UGO's affiliates at that time was a site known as Dave's Video Game Classics - a place devoted to illegal activitiy in the form of videogame emulation. Dave's had willfully uploaded to its site and provided a working link to an illegally dumped copy of the Sony PlayStation BIOS, which was required in order to make the infringing PlayStation emulator *PSEmu Pro* work properly.

It is a well established fact that unauthorized, infringing copies of a computer system BIOS is a direct violation of copyright law, and that linking to infringing material is also a direct violation of copyright law. Sony had no choice but to take action in order to protect its intellectual property. Dave's, one of the oldest and most "revered" emulation sites on the web, was subseqently shut down as the law dictated until the matter could be resolved. The end result was that Sony and Acton World eventually settled the matter, and UGO Networks expelled Dave's from its system to prevent further such incidents. Nevertheless, the offending site reappeared a few weeks later, except that this time it was hosted by a new and "more understanding" ISP. Curiously enough, the new incarnation of Dave's did not contain neither an illegally dumped PlayStation BIOS nor links to one anywhere on the Internet. The software pirates in the emulation community had been taught a lesson, and Sony's intellectual property rights were reaffirmed.

The rise of the so-called "emuscene" has its parallels in the rapid growth of software piracy in all forms and on a worldwide basis over the past two decades. With the rise of the Internet and the capabilities that it offers, the concept of intellectual property protections for original works is being abused and degraded on a scale that would have been unimaginable. Both hardware and software vendors alike are forced to commit considerable time and resources battling this new breed of pirates, who are quite adept at their trade and quick to seize upon any technology that will further their illegal aims. One of these technologies is computer system emulation, which has advanced in the past decade to such a degree that it is now possible to easily exploit a vendor's efforts without proper compensation almost at whim. It is all too easy nowdays to violate a vendor's rights over popular products, and many a software pirate is doing just that.

"Now wait just a minute!" the so-called "emuscene" protests. "Who are you to come off with that kind of attitude?"

We are the vendors. We are the ones that made the computer industry possible in the first place. We are the ones who bear the cost of continuing development, of researching new technologies, and the many other minutae that are involved in bringing a successful product to market. We are the ones who suffer the most from those not willing to pay a just and equitable price for the fruit of our labors. You who complain the most would not be able to enjoy your favorite systems and releases without *our* efforts, not yours. We do not cater to those who would unlawfully profit from the sweat of our brows by taking an otherwise legal technology and exploiting it for illegal gain.

We are not here to debate the legality of emulation itself. What we are here to debate are the illegal purposes for which emulation is currently being used. The way we shall do this is to take the arguments that the emufans use to promote their unlawful behavor and expose them point-by-point in the harsh spotlight of the law for the support of unbridled piracy that they are. Some of our fellow vendors have been accused of of making broadly worded statements, of issuing generalized proclamations, and of making assertions that are not based in fact. You shall now learn the basis for our contentions and how the facts and the law both prove that emulation, as it is now practiced by the so-called "emuscene," is nothing more than open support for that illegal activity which is commonly known as ***software piracy***.

MARKET PENETRATION

> *Emulation is a major threat to vendor profits.*
>
> The danger that emulation poses to the vendor is quite real. British emucode Gordon Hollingworth admitted as much to the readers of *Salon*.

> The emu development process is getting quicker.... It won't be long before a company releases a piece of hardware, and an emulator [for it comes out] two months later. Emulation could advance to such a point where people would ask, 'Why develop [new hardware] when we can emulate the process quicker in *Windows*, MacOS, [and so on]?

It is a fact that emulation is now inexorably linked to the growth of software piracy due to the well-documented rise of illegal "ROMs" and their widespread availability on so-called "ROM sitez." Such actvities can put a significant dent into vendor profits. Nintendo, long the victim of "ROM piracy," has this to say on the subject.

> As is the case with any business or industry, when its products become available for free, [then] the revenue stream supporting that industry is threatened. [Emulation has] the potential to significantly damage a worldwide entertainment industry which generates over US$15 billion annually, [not to mention] tens of thousands of jobs.

This attitude is echoed by market analyst Nanako Sakaguchi of Dresdner Kleinwort Benson Asia.

> If Sony lets [emulation] go unchallenged, [then] consumers may end up buying video games the way that they buy music CDs - without regard to the hardware.

Advocates of "free software" are often fond of pointing out that vendors make tremendous profits from their products, thus significantly reducing the impact that any illegal activites might make. The truth is a different tale altogether. The Interactive Digital Software Association (IDSA) estimates that sales of counterfeit videogames came to at least US$3.2 billion during 1998, which is quite a hefty figure by any yardstick. Compare this with an estimated US$16 billion in legitimate worldwide videogame sales, and it amounts to a 20% dent in total vendor profits. In plain English, software piracy is taking away at least one-fifth of vendor profits, which is a far cry above the supposedly negligible amount that supporters of "free software" claim. Pat Becker, director of corporate communications for videogame vendor Electronic Arts, makes the following observations regarding the impact that piracy has had on that company.

> We figure that we lost approximately US$400 million last year [alone]. Publishers with hot titles tend to be the ones that get targeted. Bad games don't get copied much in the marketplace.

Also consider the words of Nintendo general counsel Rick Flamm.

> Our definition of 'piracy' consists of counterfeit software - something that purports to be an authentic cartridge but is not. There is a lot of counterfeit GameBoy software, and there is a little bit of counterfeit N64 software. We estimate that Nintendo lost approximately US$725 million to piracy in 1998, and that [figure] does not include Internet piracy.

Nintendo also estimated that they had lost an estimated US$866 million in 1997. Nintendo's legal offensive is not just limited to American shores, either. In October 1998, they sucessfully put a halt to a massive videogame counterfeiting operation based in Taiwan, whose operations spread from mainland China to Paraguay. Clearly, Nintendo's legal strategy is working in Nintendo's favor, but there is a long way to go before the scourge of software piracy is eliminated.

Another weak argument raised in defense of emulation is that it lets a user see a videogame for a particular platform before they decide on its purchase. The old "try-before-you-buy" argument is valid for demos that are relased by the vendor, but not for illegally pirated full-version programs irrespective of the platform involved. This argument was laid to rest long ago for the reason that Peter Baruk of the Software Publishers Association (SPA, now the SIIA) mentioned when questioned by *TechWeb* on the subject. It is a common-sense argument that anybody of minimal intelligence should understand.

> Why would [prospective users] buy it when they've already got it? There are people who do [eventually buy pirated software], but I think there's a greater number of people who [just say], "Screw it, I got it."

It is perhaps best to let the IDSA have the final word on this particular topic, as they boil the matter down to its basics in their online FAQ.

> Software publishers must generate a meaningful return on their investments in this intellectual property if they are to continue to meet the demand for

> technologically advanced products. The suggestion that some piracy is benign undermines respect for the intellectual property rights on which companies are built. Piracy of any kind on any scale erodes this foundation.

Emuation violates a vendor's intended market for its products.

The software pirates who make up the so-called "emuscene" like to brag that emulation makes it possible for them to use any piece of software for any system and from any market whenever they feel like it. Howard Wen, writing for *Salon* magazine, puts it this way

> Emu programming is seriously blurring the lines between the proprietary formats that have always balkanized elecronic gaming.

While we disagree with his quaint emotionalism about "balkanizing," nevertheless the point he makes is sound enough regardless of whether or not you are pirating videogames or applications software. Software piracy in the form of emulation is breaking down the established market barriers that allow vendors to customize their products for different audiences. It is a fact based on long experience by the vendor community that not all markets are alike, hence the need for such barriers in order to ensure that the optimum product gets to the most receptive user base.

Let us take a moment to hear the words of Beth Llewelyn, speaking on behalf of Nintendo against piracy of Nintendo videogames.

> Emulators [such as *UltraHLE*] are illegal They obviously had to circumvent our security chip. [Emulation] promotes continued piracy.

The vendor community acknowledges that there exists a diverse market for computer products. What may sell in one country may not in another. What may be popular in one part of the world may be highly offensive in another. What may be permissible in one region may not be in another. Market locks are a way to enforce these barriers in order to ensure that the right product goes to the right market. Vendors are not about to remove them for the one or two software pirates living in their mother's basement who want "the full uncensored version" of their favorite videogame. Conversely, the software pirates do not and never had a right to remove these protections in the first place, and this fact is now law.

The Digital Millenium Copyright Act (17 USC 1201) makes it a crime to violate the security systems of any product. Since market code enabling is part of the security measures place on certain computer programs *and videogames in particular*, then bypassing the market locks is in fact a crime.

It is already a matter of public record that so-called "cracked" versions of both infringing PlayStation emulators, *Video Game Station* and *bleem!*, are now available and can be located by those with the diligience to find them. The hackers involved have overcome the rather limited precautions built into these products by their respective companies, permitting pirated PlayStation discs to be use *irregardless of market origin*. *PSEmu Pro*, the illegal PlayStation emulator, could do this from the beginning. Bypassing marketing codes is a common illegal feature of other such infringing emulators regardless of the platform emulated. The authors of these programs seem to care not one whit that the original vendors of both the hardware and software involved may not have wanted a particular product released in a particular market. If it's available, they want it, and they want to make it "freely available" to others that follow in their illicit wake.

Emulation promotes software piracy.

"Piracy in all of its forms represents the biggest threat to the continued growth of this industry," according to IDSA president Douglas Lowenstein. It is a fact that emulation, despite its legal roots, has fast become one of the most favorite tools of software pirates. While it may not be illegal to write an emulator, it is illegal to use it to perform an illegal act. Software piracy in all forms is illegal, so if an emulator is used to run a pirated piece of software, then that makes emulation illegal.

We concede that emulation technology is both legal and permissible under current intellectual property laws. It is our contention, however, supported by the facts, that emulation is being used for an illegal purpose - namely software piracy. A real-world comparison would be the police radar detector. It is not illegal to own a radar detector; however, it is illegal in many places to use one in order to commit the misdemeanor crime of speeding. If you use the radar detector that you own in one of these places, then you are committing a crime. So it is with emulation. While it may be legal to code an emulator for a given vendor's proprietary system, and while it may be possible to do so in a fashion that does not void that vendor's intellectual property rights, it is clearly illegal to run pirated copies of that system's programs on that emulator regardless of how they are procured or in what format they may reside. The IDSA makes its position on the subject quite clear in regard to videogame emulators; and we take the liberty of adding the appropriate legal reference to their statement.

> While some emulators are made by hobbyist programmers, that does not mean that they are legal. If the sole purpose of an emulator is to allow the playing of a console game on a PC, and the owner of the copyrights in that console game has not authorized the copying, performance, display, or derivative work created when a console game is played on a PC [17 USC 106, 106A], then the creation and use of that emulator constitutes an infringement of the copyrights in that console game.

Perhaps the favorite target of pirates, emulation or otherwise, is the Nintendo Corporation, the world-famous videogame company. Their position on emulation is crystal-clear, made with all the subtlety of a sledgehammer, and was voiced earlier by company spokesperson Beth Llewelyn.

> Emulators [such as *UltraHLE*] are illegal .... [Emulation] promotes continued piracy.

Even some of the emulator authors themselves concede this point. Marat Fayzullin, a computer science graduate student at the University of Maryland in College Park, is something of a god to the so-called "emuscene." Author of the popular Virtual GameBoy and MasterGear videogame emulators, he nonetheless remains rather critical of it.

> I am not a "scene" person and would very much dislike to be [regarded] as one. I have been here [long] before these pirates and do not want to have anything to do with them.

It is clear that Fayzullin, who has been on the so-called "emuscene" longer that almost anybody there, understands the stakes of the emulation game. Another perspective on the matter is given by one "Dethblade," writing for *PSX Underground.*

> Emulation is piracy, and it's only the IDSA's job to combat it

Lest you forget the point that we have made, we shall reiterate the cold facts once again. Emulation as a practice is technically legal within strict bounds laid out by both statutory and case law.

Software piracy as a practice is illegal. An unauthorized emulator that does not violate our intellectual property rights is perfectly legal, but running pirated copies of our software on that emulator is in fact illegal. *Emulation provides indirect support for illegal activities; therefore,* ***emulation promotes software piracy****.*

INTELLECTUAL PROPERTY

Emulation always infringes upon a vendor's intellectual property rights.

If emulation promotes software piracy, then it must by definition infringe the intellectual property rights of the various vendors behind the original system and its programs. The IDSA makes it clear in its online FAQ.

> In fact, most emulators that are freely available today are merely software [based] emulators that have no role in the creation of properly licensed video games; these emulators have the exclusive purpose of infringing copyrights and are [therefore] illegal.

Nintendo echoes this sentiment in their own intellectual property rights statement when it contends that "...the only purpose of video game emulators are to play illegal[ly] copied games from the Internet." They are well-known for going after any Internet site that supports such illegal activity, whether it be the distribution of an illegal emulator for a Nintendo console or the prosecution of Internet sits that make illegal copies of Nintendo "ROMs" available to the public. Nintendo chairman Howard Lincoln made the following public statement regarding the Taiwanese counterfeit software bust of 1997, but it applies equally well to Nintendo's attitude regarding emulation.

> This is just the latest example of Nintendo's resolve to do everything within our power to stamp out illegal distribution of our intellectual property.... The collective threat of copyright thievery is far greater than any competetive challenge within the software industry.

The vendors are not alone in their worries. There are many among the programming community who are also concerned over the unbridled software piracy that emulation is causing. Ed Logg, legendary Atari videogame programmer, who either authored or contributed to many of that company's classic titles, has this to say on the subject.

> If someone copied a CD or downloaded a game [ROM] off the Internet, then I have a problem with it.... Since manufacturers sell the game [for use on] one platform, it may not be 'right' to run that version on another platform. Manufacturers often get the rights to, say, sell a PSX version, but another company may have the rights for the Mac or PC version. I can see why [emulation] would be a problem.

Edge magazine states the ugly truth about emulation and its rampant spread on the Internet in a recent article.

> "[Illegal] websites are easily located and closed by industry bodies such as the IDSA and ELSPA, but for every homepage erased another replaces it. The existence of N64 'backup' units such as the Z64 and Doctor V64, used by many for piracy, means that virtually every game released is posted on the Internet literally hours after its release. One source [of ours] even went so far as to claim that over three terabytes (3 million MB) of [illegal] N64 ROM images changed hands on Sunday, January 31 [right after *UltraHLE's* release]. This figure may well be exaggerated, but the fact remains: Nintendo is unfairly losing out.

Many vendors are understandably concerned by the rampant pirating of their products. Nintendo makes this clear in their intellectual property policy statement.

> All ROMs published for play on Nintendo hardware and available on the Internet are unauthorized and infringing copies of copyrighted works.

Vendors do not like to spend valuable company time and resources battling software piracy in all its various forms, including emulation, but they have no choice in the matter. Failure to file immediate protests eventually results in the loss of that right per federal law, so vendors are forced *by law* to devote precious resources to the fight - resources that could better be used elsewhere. Nintendo legal counsel Rick Framm agrees.

> When[ever] we launch a new system, we have to relaunch our counteroffensive against copier devices and computer software.

It should also be noted that Nintendo and other like-minded vendors hold that emulators in themselves are counterfeit products. Nintendo makes it clear in their official company policy statement regarding intellectual property protection of their products.

> A counterfeit Nintendo product is [defined as] an illegal copy of an authentic Nintendo product.

Their reasoning can be explained in one of two different ways. A counterfeit is anything that purports to be something that it is not. An unauthorized program that claims to be a Nintendo videogame console emulator can be considered an unauthorized working duplicate of that console, which would make it a counterfeit product in Nintendo's eyes. Another way to explain their reasoning is to examine 18 USC 1030, the Federal Computer Fraud and Abuse Act. This makes it illegal to produce a counterfeit copy of a computer program. Videogame systems contain many programs embedded within the firmware of their ROMs in order for them to function properly. Duplicating these programs and then claiming that they work just like proprietary vendor code is not only copyright violation, it also fits the dictionary definition of counterfeiting and would therefore be in violation of 18 USC 1030. Whichever approach you take, emulating a Nintendo videogame console invites one or more charges of counterfeiting by Nintendo. It is a novel approach in the fight against software piracy, but one that other videogame vendors are now seriously considering.

Perhaps it is best to let the so-called "emuscene" have the final say in this regard, speaking in the voice of *PSX Underground*'s "Dethblade."

> Emulation is as illegal as most things are illegal, yet [it is] still being done.

No wonder that the vendors must continue their campaign against this and all other forms of software piracy support.

Emulation does not credit the original authors or vendors for their work.

It doesn't matter how old your product may be - programmers are still proud of their work. Rob Fulop, 1983's Game Designer of the Year, made the following observation to a friend shortly after the recent World of Atari convention in Las Vegas, which celebrated the heyday of the original Atari 2600 video console.

> "I don't think most people understand why walking down those aisles made me sad. I was seeing the boulevard of broken dreams. [The people in the dealers' room] are looking for bargains, while I'm seeing a game that a friend of mine spent two years programming."

While Fulop was speaking abou the uneven pricing structure for the resale of original carts (good games going cheap while bad, rare games fetch high prices), the sentiment is often the same with regards to the so-called "emuscene." Actual Entertainment's Franz Lanzinger, who himself programmed many of the classic games that are being pirated by the so-called "emuscene," has this to say on the subject of proper recognition.

> It is an outrage when emulated versions of games are released with credit given to the people who wrote the emulator, but no credit - or merely a special thanks credit - [is given] to the original game designers.

The so-called "emuscene" is for the most part populated by immature people who are completely ammoral with regards to the intellectual property of others. They consider themselves "electronic artists" who have the freedom to do what they want, go where they will, take what they need, and don't give a second thought about from where it comes or to whom it might actually belong. If it is usuable, they use it and claim credit for themselves. If not, they discard it and seek for something else. They are so debased as to quarrel with or even steal from each other, proving just how much of a threat they are to honest programming. The only positive side to this is that the true talents are usally driven out of these questionable activites and into something else better resembling honest work. Consider the words of our friend "Wook" from one of his EmuCamp editorials.

> *PSEmu* version 2 [an illegal Playstation emulator] got killed because of people who posted the leaked beta publically.... Kiddies and idiots who've mail-bombed and ragged several authors about various games have caused a few authors to throw up the surrender flag and proclaim the effort isn't worth the cost ... [and] there are [emulation] websites that make a habit out

> of stealing others' words and claiming them as their own work.

Even the so-called "emuscene" itself seems to suffer from an identity crisis. Any claim to reputability among its denizens is marred by the frequent "hacks," "rips," and other such infringing nonsense. For example, the author of the illegal Super Nintendo emulator *VSMC*, one Chris George, wound up withdrawing support for his infringing program after a team of his fellow hackers "broke" the internal protections of the public distribution release, thus rendering it fully useable. The *NeoRage* hackers suffered from this, as did the vendors of such infringing products as *A-Max* and *bleem!*. These hackers did not care one whit who made the product or why, even though they were from among their own ilk. All they wanted were "the warez," and they put their own names on the "cracks" that they released to the public. It seems the old adage is true that thieves will often steal from their fellow thieves whenever the mood strikes them.

<u>OPERATIONAL CONCERNS</u>

Emulation violates a vendor's intended usage for its products

Nintendo's policy statement regarding the intended usage of their own videogame consoles is quite clear on the subject of emulation.

> [An emulator] infringes Nintendo intellectual property rights, including copyrights, and circumvents Nintendo's anti-piracy security system[s].

One might wonder how Nintendo arrived at such a statement, but the answer is obvious to anyone who consults the law. There are multiple ways to violate the various forms of intellectual property law, whether they be the patents that Nintendo has on certain proprietary devices contained within its products (the Patent Act, 35 USC), the copyrights on any programs either burned into the firmware of those devices or stored within the various formats of its proprietary delivery systems (the Copyright Act, 17 USC), the trademarks that they incorporate into their products (the Landham Act, 15 USC 1051-1127), or in a broader sense with regards to the more general forms of intellectual property violation (the Intellectual Property Rights Act, 37 CFR).

So does the intent of usage stop with the emulator? No, it applies to the program base as well. Torter Miltenberger scores an excellent point in this regard in his *RPGamer* editorial "The Devil's Advocate."

> Most [Playstation] emulators can't tell the difference between an original copy and a CD-R "backup." [Suppose somebody] obtains a CD burner and starts selling 'backups' of games for US$10 apiece. This would be damaging to video game corporations and if it continues on a great enough scale, [then] it will drive up the price of legitmate video games.

This ties in neatly with the marketing model presented several discussions ago, in which vendors are forced to artificially inflate the price of first-release titles in order to counter the perceived threat of software piracy. It is a fact that the *bleem!* Playstation emulator for IBM PC compatables works just as Miltenberger describes - it does not care one whit whether or not you are using a legitimate Playstation disc or a CD-R copy, as its emulation did not include any functions from Sony's antipiracy system. It is a fact that a real Playstation will not do this without performing an illegal hardware hack to install a special "mod chip" which can be easily purchased off of the Internet black market, or by obtaining an unauthorized "mod card" for its I/O port that essentially does the same thing. Sony never intended for pirated copies of Playstation games to work on its consoles, but this can be done on an emulator with no trouble. As a result, any emulator that can do this is performing an act of *contributory copyright infringement* and is therefore illegal. It is more proof that emulation as it is practiced today is definitely illegal.

What is true for systems utilizing recordable storage media for program delivery (floppy discs, CD-ROM, DVD-ROM, etc.) also holds true for systems utilizing permanent storage media (videogame cartridges, plug-in boards, "flash" cards, user-insertable ROMs, etc.). The altering of a copyrighted work's intended format or its transfer into another format is part of the ultimate right of adaptation which by law is exclusive to the copyright owner (17 USC 106; Mirage Editions v. Alberquerque ART, 1989). Again, we quote Nintendo on the matter, but we take the liberty of inserting the legal references that their printed statement omits.

> There is a good deal of misinformation on the Internet regarding the backup/archival copy exception [of copyright law, 17 USC 117]. It is not a "second copy" rule and is often mistakenly sited for the proposition that if you have one lawful copy of a

> copyrighted work, [then] you are entitled to have a second copy even if that second copy is an infringing copy. That is utterly ridiculous and legally unsupported [MAI v. Peak, 1993; Allen-Myland v. IBM, 1994; ProCD v. Ziedenberg, 1996]. Therefore, whether you have the authentic game or not, or whether [or not] you have possession of the ROM for a limited amount of time (24 hours [is commonly cited by the so-called "emuscene"]), it is illegal and infringing of a copyright to download and play a ROM from the Internet.
>
> The backup/archival copy exception [of copyright law] is a very narrow limitation .... It is well established by judicial decisions in the United States that this limited exception does not apply to game data contained in ROM ... [Tandy v. Personal Micro Computers, 1981; Apple v. Franklin, 1983; Atari v. JS&A Group, 1983; Sega v. MAPHIA, 1994].

It doesn't matter how or from where you got your "ROMs" for your emulator. They were intended to be used in a predefined way and in a predefined manner by their vendor. Anything outside of that constitutes intellectual property infringement.

One other point. The so-called "emuscene" is making a lot of noise about computer programs (videogames in particular) not requiring the anti-piracy measures that were originally included with them in order to work with their emulator. That is beside the point. Any tampering with a computer security system, including its removal from the computer program(s) it is protecting, is a direct violation of federal law (17 USC 1201). It is just as illegal to distribute and download a so-called "cracked ROM" as it is for one that still has its anti-piracy measures intact. This is just another example of the verbal contortions that these software pirates employ in a vain effort to justify their actions.

Emulation limits users by limiting the operations one could perform with newer, enhanced, and often superior versions of a given program.

Advocates of emulation often like to brag about how they are "preserving the past," or how they are "saving users from new and buggy products." This is simply not true. New release of old products offer many advantages over old ones in almost every category. The common fallacy that every new update of a current or older product is bug-laden to the point of unreliability is a devious lie used in a pathetic attempt to justify the continued spread of software piracy. It is just as illegal to pirate *Windows 1.0* as it is to pirate *Windows98*. It is just as illegal to pirate the original *Donkey Kong* arcade game as it is to pirate *Diddy Kong Racing* for the N64. Piracy is piracy, regardless of the version of the program(s) involved.

PRODUCT EXPERIENCE

Emulation distorts the product experience.

It is a fact that an emulator, however well it may be designed, always falls short of the actual product in some respect. This is the very nature of the beast involved, as software can never be a perfect substitute for hardware. This is especially true if the product was designed to perfrom with certain hardware and in a certain fashion, as are almost all videogames. This concept is what is known as the "product experience," and Franz Lanzinger provides an original programmer's perspective on the matter.

> [Videogame emulators] are seriously flawed to the point where the game experience is quite different from the original.

For example, anyone who is a fan of classic arcade games will remember the first *TRON* arcade game. The infringing arcade emulator *M.A.M.E.* supports *TRON* via illegal "ROMs," but it cannot properly mimic the special controls of the original console, thus resulting in a distorted product experience. For another example, Sony rejected *bleem!* in the first place because it was (and still is) a sub-standard product that fails to accurately convey the entirety of the PlayStation product expierince with each and every title in the PSX library. There are many more examples we could cite, but these two should be sufficient to convey the fact that emulation is never as good as the original hardware.

Emulation violates the experience we intend to permit within a given market or within a given product line.

In an earlier discussion, the problem of cross-market violation of a given vendor's product intentions was briefly addressed. This problem not only affects the intended market, but it also affects the intended product experience for that market as well. It is a fact that different markets have different tastes, as well as different desires and expectations for the products that it prefers. What may be permissible in one culture may be an abomination in another. What may be desired by one market may be deemed frivolous in another. We do not see the necessity to neither port certain titles nor refrain from editing certain titles for a given market. The answer is both simple and obvious - ignoring the united desire of an entire market for the strident demands of an intransigent few would ultimately prove detrimental to our corporate image and put at risk other titles more worthy of the market in question. Emulation destroys our ability to maintain this control; therefore, emulation violates our ability to control the product experiences that we feel are best suited for a give market

There are places in the world today, such as Malaysia, many former Soviet bloc states, and mainland China, where software piracy is given a wink and a nod. Drop by any store vending computer software and you will find bootleg upon bootleg, frequently with counterfeit packing, and more often than not pirated from another market to vend in the local one. The same is especially true of the Internet, where the most blatant acts of software piracy are carried out by offenders whose sites are hosted by ISPs located within these so-called "safe" countries. We lose millions and perhaps billons of dollars in lost revenue each year to software piracy in all forms, and part of these lost profits could have been used to both port and tailor additional products for our various markets. The ready availability of pirated copies of our programs forever distorts the product experience that we intended for any given market, and it does not matter whether or not the bootleg in question is a "full version" from another market or a third-rate hack job by local pirates utilizing illegally "ripped" material from a different port. Piracy is piracy, plain and simple, and dealing in unauthorized copies of our products that were never released in your market is and always will be piracy no matter what you may claim in your defense.

NOSTALGIC YEARNING

Emulation effectively discourages us from re-releasing classic products.

The existence of pirated ROMs makes vendors think twice about re-releasing classic games. Torter Miltenberger, in his editorial "The Devil's Advocate" for *RPGamer*, makes the point from the perspective of the average classic gamer.

> There is always the possibility (however slight) that the game may be re-released. A perfect example of this is *Final Fantasy V*. If emulation were non-existent, then the upcoming re-release of *Final Fantasy V* [for the Sony Playstation] would sell better.

> instead, people will look at it and say, "I already beat the ROM." If that doesn't harm SquareSoft then I don't know what does."

We quote Nintendo to provide the perspective of the vendor community.

> Distribution of an emulator trades off of Nintendo's good will and the millions of dollars invested in research & development and marketing by Nintendo and its licenses. Substantial damages are caused to Nintendo and its licenses. It is irrelevant whether or not someone profits from the distribution of an emulator. The emulator promotes the play of illegal ROMs, ***not*** authentic games. It has the opposite effect and purpose.... If these vintage titles are available far and wide, [then] it undermines the value of this intellectual property and adversely affects the right owner.

In other words, continued activity by software pirates concerning Nintendo products is what keeps that particular vendor from re-releasing its classic titles or porting them to other platforms. What is the point of their doing so when illegally pirated copies are available far and wide? They stand to lose money on such a venture, and that thought has kept many a vendor from re-releasing their older yet still popular products.

Emulation corrupts the present.

So what is the big problem with emulation? It is a matter of principle. The availability of emulation effectively stymies vendor support for the user base of its past, present, and future products. Kazuo Hirai, president of Sony Corporation (makers of the PlayStation), puts it this way.

> I don't think I would want to be ... in a position where I am profiting from sales of software to run on somtheing that is based on copyrights and intellectual properties that are, by the way, being stepped all over! I would say, "Okay, we don't need that extra unit sale," if it means I get to protect my copyrights.

That is the opinion of Sony, one of the major players in the videogame market, as viewed from the financial and intellectual property angles. Nintendo, another major player, provides a slightly different slant along the marketing angle in its public policy statment.

> Emulator/ROM piracy is competing head-on with Nintendo's current systems and software.

To that we add the words of SquareSoft's Akira Kaneko, spokesman for the well-known software house and makers of the popular *Final Fantasy* series of RPGs for both Nintendo and Sony platforms.

> Fundamentally, our stance is the same as Sony's. If companies like us aren't able to sell as many games (because of a proliferation of pirating), it'll be that much harder to develop future titles. The software industry may actually shrink.

Jeff Gerstmann of *VideoGames.com* agrees.

> ... like it or not, all of these ROMs are copyrighted, and spreading them around is simply piracy.... There's also a bit of a moral issue with the dumping and distributing of ROMs. You could argue that all those Atari 2600 ROMs being out there don't hurt anybody (though I'm sure Activision would disagree, since they released that Atari 2600 pack for the PlayStation), but let's face it. Games like King of Fighters 99 and well, any N64 game just shouldn't be out there for the taking.

Finally we quote the so-called "emuscene" itself, which openly admits its involvement in software piracy. Again, the words of "Wook" from EmuCamp.

> "We stand together or we fall alone. If you won't take the time to get involved with emulation beyond downloading roms [sic], then your [sic] not part of emulation. You're a user, a taker, a pirate, and unwelcomed.... If you don't like what emulation has become then leave or contribute. There are no other options. Whining accomplishes nothing.... If a site-op is getting harrassed by the IDSA, try to support them rather than thinking 'with XYZ out of the way more people will visit my site.' We're all in this together. If you make a path of standing alone, you

> will be alone when you need help. Be a giver, not a taker."

One of his fellow hackers, the mysterious "ManBeast," had this to say on the subject.

> "It just [so] happened that N64 emulation has sparked a huge interest in the piracy factor of emulation (which exists anyway). [Posting the] Mario Bros. arcade ROM is just as much a crime as [posting] a Super Mario 64 N64 ROM no matter how you justify the latter. So [why don't] those [of you] criticizing N64 ROMs go look at your hard drives and CDs and tell me that you are doing something less legal with those Atari 2600 and classic arcade ROMs, or those C64 and Amiga games. We are all guilty. Get over it - emulation has an illegal aspect to it no matter how you justify it to yourself(the emphasis is ours)."

"Dethblade," one of the writers for *PSX Underworld*, puts the matter rather bluntly.

> ***Emulation is really nothing more than software piracy*** (emphasis added).

Truer words could not have been spoken.

It is a fact that millions of dollars are lost to the vendor community each year as a direct result of software piracy. When you corrupt an otherwise legal and viable technology to further this illegal activity, then the effect of those losses are further magnified. There is no conceivable reason for the average user to want or need an emulator for a dedeicated videogame system, be that arcade or home console, if there is not present an intent to bypass the use of the original console. Additionally, if the security features of the original console are bypassed, this enables the use of pirated copies of its program base that would not otherwise work on the actual hardware. There is clearly no need for videogame emulation other than to enable piracy. It is a corrupted technology that is thwarting the present desires of the vendor community.

INTROSPECTION

On 8 September 1999, the infamous computer hacker known as RealityMan publically announced that he would continue work on his infringing N64 emulator, *UltraHLE*. This is a slap in the face of a vendor that has given so much to the videogame community, and we support any action that Nintendo takes in this regard. *UltraHLE* was an infringing emulator when it first appeared this past January, and it remains so to this day. Products such as his do nothing more than promote software piracy. The theory behind it may be lawful, but the acts that it enables are certainly unlawful.

There is a word used in legal circles to describe the commission of an unlawful act that in other circumstances would be considered lawful and proper. That word is *malfesance*. Such could aptly describe the situation that emulation poses to the vendor community today. We do not have a problem with the concept of emulation. We use the technology ourselves on occasion whenever it stands to profit us. We do have a problem with the unbridled use of emulation by software pirates to perform acts that are clearly illegal. So what should be done about these illegal activities? The so-called "emuscene" gives its own telling answer, and again we quote "Wook" of EmuCamp.

> Think of this this way. If you were throwing a party for all [of] your friends at your house, would it be OK with your neighbors if you just walked into their house, without asking your neighbors, and raided the neighbor's refrigerator for all the food your party needed?"

We could not have said it better ourselves. We cannot stand idly by and allow others to benefit from our efforts without proper compensation.

Emulation is being used to assist the rampant spread of software piracy. *This is a situation that we will not tolerate*.

It is time to face the facts.

Unauthorized, infringing emulation ***is*** illegal. Period.

REVIEW QUESTIONS

1. Why was Dave's Video Game Classics shut down by its ISP? How was the situation resolved?

2. Does emulation pose a significant threat to vendor profits? How so?

3. How does the vendor community feel about tboth the concept and the practice of emulation?

4. Explain how programming an emulator may be legal, whereas using one may not be legal.

5. What is the real reason for videogame emulation [according to the vendor community - ed.]?

6. Why will a vendor object whenever it deems that its intellectual property is being violated? What is the driving reason behind such swift action?

7. Can you explain one of two ways in which an emulator or a "ROM" could be considered counterfeit?

8. What is the standard practice used by software pirates with regards to assigning credit for their "releases?" Is this limited to just commercially vended products? Why or why not?

9. Why do software pirates prefer infringing emulators over non-infringing ones? Can you describe some infringing emulators?

10. Why are "ROMs" illegal regardless of their format?

11. Why is it illegal for an emulator to bypass the security measures built into computer systems and videogame consoles?

12. How can an infringing emulator violate a given vendor's market intentions?

13. How does emulation discourage the reissue of older or "classic" products?

14. What does the legal concept of malfeasance have to do with the so-called "emuscene?"

THOUGHTS TO PONDER

1. How has the Digital Millenium Copyright Act limited the possible infringements that unlawful emulation could pose to the vendor community?

2. What are the legal bounds restricting emulation technology as they exist today?

3. How can current law be changed to better protect vendor rights? For that matter, how could the be changed to better promote the cause of "free software?"

4. How does the anarchistic behavior of the so-called "emuscene" influence its impression on outside observers? Will this behavior ever change? Why or why not?

[... and now the editor inserts some questions of his own ...]

5. Do you think that the vendors are responding rationally to emulation? Why or why not?

6. Is it possible to separate emulation from charges of software piracy? How?

7. What is the real truth behind any counterfeiting charge by an offended vendor?

Module Three: The Implications
Part 3 - Reflections

Questions are a burden to others, answers a privilege to oneself.

common Village saying from *The Prisoner*

COMING FULL CIRCLE

In late May of 1999, I decided that it was necessary to empty out the basement of my home. It was suffering from a water seepage problem due to a crack in one of its concrete walls, and I had noted with some alarm that the spring rains had already taken their toll on certain items that I had stored down there. It was with great reluctance that I arranged to set an entire weekend aside from my many projects for the unenviable task. When the time came, I spent it removing all of the contents from my basement, checking them for water damage, discarding the things that were too badly damaged, setting aside those with only minor damage for future repair, and repacking the rest in a nearby storage building. Things did not go as fast as I had hoped due to the sheer volume of the material to be moved, and eventually three weekends had to be spent in the effort. There was also another factor involved, quite unexpected at the time, and that is the one I shall share with you.

It was on the second weekend, which was largely concerned with shuttling undamaged items from the basement to other storage, that I rediscovered an old friend. It was necessary for me to shift and restack part of the boxes in the storage building in order to make additional room. One of these, an old cardboard "carry box" with handle that used to house a Reveal 2X CD-ROM retail kit, proved to be surprisingly heavy and off-balance with regards to its contents. I set it aside, curious at what it might contain, and finished restacking the boxes. Once I had finished all that I could before daylight began to fade, I took a deep swig from my nearby glass of tea and then focused my attention on the mysterious carry box. I took it inside my shop, set it on a nearby table, and opened it.

Inside was a stock Amiga 1000 computer with all cables, the keyboard, the mouse, and around hundred-odd floppy disks.

All thoughts of relaxation that evening vanished from my mind and I smiled to myself with delight, running my forefinger over its beloved "rainbow checkmark" on the case's right front corner. It was not one of my original Amigas, of course - those had been sold along with all of my accessories and software many years before. It was a gift from one of my co-workers, someone who had also been a fellow Amiga user and had just put it away, forgetting about it with the passing years. He had stumbled across it while cleaning his apartment last year and given it to me, software and all, knowing my love for the machine and the fact that it would be impossible to resell in local markets. I had stuck the box in storage and subsequently forgotten about it until today. It sat before me now, my old friend, a beautiful but tempestuous lady who I had grown to know and love over the space of five years from 1988 to 1992. Does it surprise you, readers, that I simply could not resist the invitation?

I spent the rest of the evening with that old Amiga. I had a stack of old Commodore 1702 and 1802 monitors that I had picked up rather cheaply at a local school auction, and one of these was quickly brought over and connected to the Amiga's color composite video port. I then spent about a half-hour reconstructing a good set of Kickstart and Workbench disks from the multiple copies my friend had thoughtfully included, all of which had suffered from digital degradation over time. Once that was accomplished, I spent the next several hours going through the entire set of remaining disks, seeing which ones could be salvaged and which were irreparably lost. Many fond memories were stirred that evening during my figurative trip back through time - the Amiga sitting before me, its floppy drive softly clicking (oh no, *NoClick*'s corrupted!), while I cycled through each set of program disks, checking them with *DiskSalv* and then testing them for operability. You videogame fans of the era will recognize many of the games that I relived that evening ... *TV Sports Football and Basketball ... Xenon ... Onslaught ... The King of Chicago ... Defender of the Crown* (why has nobody remade this excellent game?!) *... The Great Giana Sisters ... It Came From the Desert* (the original, not that Genesis piece of crap!) ... many, many Psygnosis titles (they made their name on the Amiga, remember?) ... and lets not forget the requisite apps, either ... *X-Copy Pro* (yes!) ... *Project D* ... *Marauder* (pan pipes and all!) ... *AmiPro* ... *SuperBase* ... *Deluxe Paint II* ... *CygnusEd Professional* ... *DiskMaster 1.3* ... and so on. There were many, many more that I missed from those days which had not been included with the disks that came with the system. I was half hoping to run across a copy of *Transformer* for you folks, but that was not to be. I eventually wound up with seventy-odd floppy disks that I had either salvaged or repaired and then recopied, and then discarded the rest. These were neatly packed up so they wouldn't get scattered, and then I took one last, long, loving look at the familiar blue Workbench screen running on the actual hardware. Click ... click ... click ... click .... click .... I shook my head, saddened at what I now must do. "They just don't make 'em like that anymore," I sighed aloud, and with that I turned it off. I unscrewed the case and removed the lid, flipping it over, and took one last look at the many names embossed in the plastic of those who had helped to create this technological marvel for its day. After that, I repacked it back in the box with its disks, put the monitor away, and put the Amiga back into storage with my other momentos from my past.

Will I ever look at it again? Unlikely. Amiga emulators for newer systems have gotten so good that there is really no need to mess with the original hardware anymore. My old friend will stay there unless I choose to dig it out again or my heirs discover it. So why did I keep it for so long? Perhaps it was there to remind me of possibilities realized, of obstacles overcome, and of potentials yet to be tapped. The Amiga was a legend in its own time, and rightly so, for its reputation was justly deserved. Even today you can feel the effects of the Amiga legacy in the computer industry, and it has only been within the past two or three years that technology in the IBM PC compatible market has finally surpassed all of the features and options that the Amiga had to offer its users. It showed the world that computer users wanted more from their systems than "running today's software," and the industry eventually responded. The Amiga was a window into today's high-tech, high-performance world, where multitasking and multimedia are part of the expected norm, not the exception to the rule. Emulation was one of the things that helped to build the legend that the Amiga became, and its legacy is still with us. The Amiga was the one personal computer more than any other that made the concept of emulation into an everyday reality for the common folk, and that legacy will live on long after the last surviving Amiga crumbles into dust.

LOOKING BEHIND

It is fitting that we should come full circle as our discussion of emulation draws to its close, and find ourselves back where we started with the legendary Commodore Amiga. The legal battle for the heart and soul of emulation has come back to its roots, and today's contest between Sony Corporation and bleem LLC is over a matter that was first addressed by the industry a decade ago, during the heyday and as a result of the Amiga. The very same issues that the lawyers of Sony, Bleem LLC, and Connectix throw at each other have eerie parallels to the *A-Max* affair. This is why I and many others hold that the fight over PlayStation emulation directly challenges the legality of third-party emulation that was first established by the *A-Max* precedent. Once again, history repeats itself.

I cannot help but be struck by the fact that classic computing never dies. Like vintage wine, it just gets better with age. The actual hardware may deteriorate, the original disks may degrade to the point of uselessness, but ideas and concepts embodied within remain with us. They are resurrected time and again in new forms, with new window dressing and new features, but underneath all of those high-tech trappings can often be found a notion that in some cases may be decades old.

It was around mid-September of 1999 that I paid a visit to my local videogame arcade. I hadn't been in one for about a year, having more important things to do. I was only going now because some kids had tipped me off that the arcade versions of certain Dreamcast titles were to be found there. I was planning on buying one of Sega's new 128-bit videogame consoles after the initial rush, giving the market time to detect any undiscovered flaws as it had with other computer products (does the name Microsoft come to mind, anybody?). I was in effect window shopping, and spent about twenty minutes or so wandering around the place and checking out the games. Yes, two or three of Sega's new offerings were there, but they were not what had the attention of the folks in the arcade. Nor did the slick-looking offerings from other vendors hold them, either. Almost everybody in the place was clustered around a lone cabinet near the front of the store, where a balding, grey-haired fellow not much older than I was busy racking up the high score into the stratosphere. All of the kids, including the teenagers, were cheering him on. I noted with fond nostalgia the line of quarters laid across the top of the machine, awaiting their turn in the coin slots in exchange for extended gameplay. What was the game being played that was gathering so much interest? The original *Galaga.*

It is a shame that it is almost physically impossible to maintain the classic computers and videogames from the past in good working order. The cost and care for obsolete parts can get rather high, and there comes a point (like many do with public monuments) where you just have to seal the thing up and paint it over, or even replace it with something else. The computer industry is blessed by the presence of emulation in this regard. If we are fortunate, and we have been in many cases, thoughtful users and developers (and even bootleggers) have saved copies of our shared past long enough for the emulation scene to rescue them. In this manner future generations will be able to share in the things that we developed or enjoyed. Earlier, I proposed four reasons why emulation is still with us. I would now like to add a fifth - **preserving the past**. The Second Law of Thermodynamics states that everything deteriorates over time - a principle that scientists refer to as *entropy*. This was recognized by the British Film Institute back in 1997, when they began setting aside room in order to save copies of videogames and other forms of computer entertainment software in an effort to preserve some portion of this particular creative heritage for the enjoyment of future generations. Emulation represents the last, best hope for preserving working copies of our computer past, because *entropy dictates that the original hardware must someday die*. If that hardware has been converted to workable software, though, then it can be copied and preserved along with its software base, archived in multiple locations and stored in multiple copies. Thus, future generations will one day get to see just what all the hubbub was about back in our time.

Not too long ago, I wrote a little editorial about the TNT docu-drama *Pirates of Silicon Valley*, which melodramatically recreated the battle between Steve Jobs and Bill Gates for market dominance in the early days of the personal computer industry. I was already familiar with the story, and the only reason I watched the movie was to see all of that classic hardware again. I said as much in my editorial, and half-jokingly asked if anybody had an Altair 8800 emulator. It was less than 24 hours later when the first of what would wind up to be seven copies of such a program wound up in my email courtesy of my readers. It was the *Altair 8800 Simulator* by Claus Gioli of Microsoft, and it was a dead ringer for that classic personal computer. True, it was just a software-driven graphic on my monitor screen with program code behind it, but it acted and executed just like the real thing. I challenge you to find anybody who's still using a real Altair 8800 today. You may find the system, but its users have long since moved on to better things.

With this in mind, I would like, if you do not mind, to use the final installment of this series of discussions as a soapbox to express my own concerns about the emuscene. I have tried to refrain from any personal interjections up to this point, preferring instead to lay out the background material in an effort to ensure that all of us are reading the same pages from the same book, as it were. Mine is not the only opinion you will hear about emulation, but I hope that my background and research coupled together with this particular forum might add some weight to my words. I have words of criticism for both sides in the great emulation debate, and it is my hope that at least some of them will consider what I have to say.

A WORD TO THE VENDORS

It is often easy for a large company or even larger corporate entity to forget that its products are made, sold, and bought for the most part by average, everyday people of the ordinary sort who do indeed have powers of reason, observation, and discretion. Gene Roddenberry, the creator of the *Star Trek* science fiction television franchise, was often fond of remarking that his creation was an effort to show "the network" (NBC) that "...there is intelligent life on the other side of the picture tube." It only takes a brief glance to see the impact that his creation's popularity has had on worldwide culture since its inception back in 1965. So it is with any market. If you make a product that pleases the people, then your success is eventually assured. If, however, you disgruntle your customer base, then you place your product (and your bottom line) at severe risk. This is one of the oldest maxims in a free market economy; however, it is one of which the big boys often loose sight.

Consider the case of Atari, one of the true pioneers in the videogame industry and the first major player in the home videogame console market. The day that they started putting profits ahead of people (the hiring of Ray "the Czar" Kassar) was the day that Atari's inevitable fall was all but assured. Most of us are by now quite familiar with **the great shakeout of 1983** in the home videogame industry, which most attribute to Atari's heavy-handed operations the previous year that eventually resulted a glut of bad, unsaleable Atari 2600 videogames on the market. The people refused to be force-fed products that they didn't want, and the market wound up being oversaturated. The crash was caused by the overloaded market, and it was a blow from which Atari has never really recovered. Indeed, it was a full year before the home videogame market found its feet again, and then only by the sheer willpower and marketing savvy of a little-known Japanese company named Nintendo.

Consider the case of Apple Computer, the company that helped shape personal computing as we know it today. They ruled the personal computer industry during the first half of the 1980s with their Apple II and Macintosh PCs, both innovative and widely regarded in their day. Like Atari, though, Apple finally reached a point where it thought it knew what was better for its customers than the customers did themselves. The captive Apple user base paid a steep price for Apple's dispensations from on high, and many a disgruntled user chafed over what they perceived as Apple's costly yoke of inadequate user support. It should have come as no surprise that many bailed once an alternative, cheaper, GUI-based platform in the form of *Microsoft Windows* and 386-powered IBM PC clones began to make their mark. Bill Gates and Microsoft were only doing the same thing to Steve Jobs and Apple what the latter had done to the folks at

Xerox - appropriating a good idea and adapting it to suit their own tastes - as the subsequent and lengthy lawsuit showed. *Windows* did pretty much the same thing as MacOS but at a lower price. "Our stuff is better!" Steve Jobs reportedly screamed at Bill Gates once he realized the full extent of Microsoft's development efforts. "It doesn't matter," Bill Gates is reported to have calmly replied, and he was right. Today's personal computing world is a *Windows*-driven environment, with Apple firmly relegated to second place in its notable niche market.

Two different corporations. Two different product lines. Both shared the same problem - failure to listen to the demands of their respective user bases. Both paid the price for their desire to maintain high profits from a proprietary market.

There's a lesson to be learned here, vendors, and I can sum it up in two words.

GET REAL.

The inevitable reality of emulation is something about which you can do a whole lot of nothing. Yes, you can huff and puff and churn up the water with position papers and court injunctions, but the sad truth of the matter was stated quite succinctly by T. Liam McDonald in his *MaximumPC* editorial - "You will be emulated." It is part of the same competitive trend that affects all unique products, whether they be computer hardware or used spatulas. Your user bases have already indicated their acceptance of emulation and the promise that it holds. Is it too much to ask you to come to grips with a true market force and accept that which until now you have held to be unacceptable?

I'm sure that those of us who know our American history will recall that Eli Whitney invented and patented the cotton gin over two centuries ago, but what most people don't know is that he spent the next two decades in court fighting people who were unlawfully copying and selling his product without proper compensation. He never got to enjoy the fruits of his labor because he was too busy spending the money suing his competitors. The same pattern has been repeated time and again - every time a new and innovative product hits the market, somebody eventually figures out a way to take advantage of it without compensating its inventor. The American legal and market systems favor competition regardless of how it comes about because it ultimately results in even more innovation. Chief Justice Warren Burger, writing the majority opinion on behalf of the U.S. Supreme Court in the landmark case of *Kewanee Oil v. Bicron*, puts it this way:

> Novelty ... is not required for a trade secret .... The holder of a trade secret would not likely share [that] secret with [another] manufacturer who cannot be placed under binding legal obligation to pay a license fee or to protect that secret. The result would be [to] hoard rather then disseminate knowledge .... If something is to be discovered at all, very likely it will be discovered by more than one person. Even if an inventor were to keep that discovery to himself, something that neither the patent nor trade secret laws

> forbid, there is high probability that it will soon be independently developed. If the invention, though still a trade secret, is put into public use, the competition is alerted to the existence of the inventor's solution to the problem and may be encouraged to make an extra effort to independently find the solution thus known to be possible....

Remember, U.S. intellectual property law was conceived from day one to favor the dissemination of knowledge over the proprietary concerns of the vendor. They can have those rights for a time in order to profit from them, but after that they must give them up to the public interest whether they like it or not. Face it, vendors - the law is slanted against a proprietary mindset. Sooner or later, whether you like it or not, your product will have to deal with a similar and possibly competing product. Someday, whether you wish it or not, you will have to deal with the emuscene.

So, provided that emulation will somehow impact your product sooner or later, what's the best thing to do about it? Easy - **take advantage of it**. Don't try to kill the messenger just because you don't like his message. He's more likely to sing your tune if you're providing his paycheck. The same holds true for the presumed threat that emulation poses to proprietary-minded vendors. Why waste time, money, and effort in legal battles over emulation when you can make money from it instead? True, it won't be as much as you may have received in original product sales, but it will be a whole lot more than the *nothing* you're getting now. Why close your mind to what emulation offers?

A closed mind rejects alternate possibilities, forestalls the necessity of change, and in the end brings about the kind of disastrous downfalls that both Atari and Apple suffered. A closed mind is no safety from the attacks of more open-minded competitors, both above and below the board, who are perfectly willing to take the kind of risks that are necessary to move ahead in the marketplace. A closed mind accepts no assistance, castigates the critic, and endeavors to eliminate everything that so much as deviates one degree from the accepted norm. Emulation is outside the norm, therefore it is a deviant - inadvisable, inexcusable, and in the end unacceptable. It allows one to go beyond the boundaries, therefore it must infringe on the necessity of those boundaries. It permits experiences once thought impossible, therefore it violates the intent of the original experience. It is perceived to be an evil technology that cannot be controlled, therefore it needs to be destroyed in order to restore that control. Emulation, to the closed mind, offers nothing in the way of possibility save piracy.

There is no point in being so close-minded about emulation. Look around and see how your fellows are taking advantage of it and the promise that it holds. See how they are accepting the inevitable, grasping the attainable, and then profiting from the profitable? They are finding ways to make emulation work for them, and they are lining their wallets with their results. It may not be along the lines that certain advocates in the emuscene would prefer, but at least they're taking steps in what they hold to be "the right direction." Your fellows are realizing a profit from emulation, while you are not. You are letting others succeed in a market that you deliberately ignore, and there may not be any fertile ground left for you to plow should you ever come to your senses.

Above all else, please try to keep in mind that your user base does not consist of a herd of mindless lemmings waiting for you to direct their path through the hoops and hazards of your intended market. They do not live their lives waiting from day to day for the little driblets of vendor beneficence doled out from corporate headquarters on high. They have a mind and a will of their own, which you have either forgotten or blithely ignore, and heaven help you if enough of them rebel against the "company plan." They will only stand being spoon fed on the same bland pabulum for so long, and then they will revolt and go elsewhere for satisfaction. It has put bigger and better companies than yours out of business. Others it has forced to the brink of bankruptcy. You are not immune to the same fate, no matter who you are. Perhaps its time to show some respect for the wishes of your user base - after all, they're the ones who are putting the paychecks in your pocket every week. If they want an emulator, then fine - give them an emulator. Don't drive them away to somebody else's product - not when you can sell them your own product and recapture what would have been lost revenue. That way, everybody stays happy. You keep your user base intact, they keep buying your products, you keep pocketing the profit, and that way you can maintain your dominant share of the marketplace.

Emulation can benefit the vendors, but only if they let it.

A WORD TO THE EMUFANS

It is rather disturbing to sit back and watch the various activities taking place within today's emulation community and compare them with my own back in the Commodore warez scene just over a full decade ago. Nothing has changed. The situations and antics remain the same. Only the names and faces are different. You would think that today's users, ranging from the idealists down through the freeware and shareware types and finally ending up with the lamers at the low end of the scale, would have learned something by now. Instead, we still have the same battles over "first releases," the same titanic struggles among egocentric hackers and their adherents, the same shouting matches and shotgun-style accusations that seemingly fly back and forth at will without thought or comprehension (I think most folks call it "flaming" nowadays), the same rush to collect everything regardless of availability or obscurity without bothering to worry over little details like legalities, the same clash between freewheeling gamer and uncompromising vendor, the same drive to duplicate the seemingly unduplicatable, and so on. Yes, even the lamers are still with us (although I think most folks in the emuscene call them beggars and moochers), doing their best to leech anything and everything whenever and wherever they can without so much as a "thank you" or "by your leave," and on the whole behaving so badly that they end up lousing up the scene for everybody else. Sometimes, as I surf the Net and check out the various emusites and the message boards, I almost feel like I'm back at one of the legendary copy parties of old - with I and my fellow hackers along with the real power users on one side of the room, staring in disgust across the tables at the lamers and maggots rooting in their filth on the other side. We would smile with restrained benevolence and scarcely disguised contempt as they came begging to us for "the new stuff," and then dole out what few offerings we felt like they could handle - all the while secretly despising them and wishing that they would drop dead on the spot. It is as the saying goes. "The more things change, the more they stay the same."

At least one thing has changed, though. I have. I'm no longer the devil-may-care computer hacker and bootlegger extraordinare that I was back in those interesting and colorful years of my computer youth. I'm older, wiser, and more aware of the stakes than I was back then. I'd be lying if I didn't admit that my close brush with the Sun Devil folks in 1990 played a big part in that, but that was only one factor among many. I deliberately stayed out of the computer underground for seven years - a place that I loved dearly and still do to this day - and spent most of that time on the corporate side of things. There's nothing like seeing how the grass grows on the other side of the fence to widen your perspective on the industry, and I've done everything from clerk to company vice president. It's one thing to be the self-styled power gamer walking up to the front desk with that brand-new product in one hand and your money in the other, angling for a good deal so you can pay as little as possible and get that puppy home and copied for your buddies as fast as you can. It's quite another thing to be the fellow in the pressed shirt and carefully arranged tie on the other side of the desk looking across your glasses at the long-haired, scruffy-looking dweeb on the other side, who probably still lives in the basement of his parents' home, suspiciously wondering why this little bum just asked you to mark down new merchandise "because it's the only one left, and it's shelf-worn." Having the vendor's perspective gives you an entirely different outlook on the situation, and calls into question some of the silly things I used to do before I knew better.

I have two words for the emuscene.

GROW UP.

If you ever want to be treated with the respect that you feel the emuscene is due, then you need to start behaving more like the responsible adults you claim to be instead of the unruly pack of preschoolers you appear to be. What that means, in plain English, is that you need to stop trying to walk all over the rights of the vendors any way and any time you can. So they've got something new, and you want it? Just because it's available doesn't give you the right to pirate it. Don't mess with stuff they don't want you to mess with. Wait until the time is right, not until the coast is clear. If you really do need that "ROM" to test your planned emulator, then better be able to document why, and go out and get access to the original *as the law requires* while you're at it. If you have a recalcitrant vendor on your hands who hasn't got the message about emulation yet, then the emuscene needs to wall them off and move on. They'll get the message sooner or later, even the most stubborn of the lot. Messing around with vendors who don't want to "play the game" is what has gotten us in this fix in the first place, so perhaps it's time to leave them be. The emuscene is just as guilty of making mistakes as the vendors, and it's time for it to start taking responsibility for its actions.

As I review the various forms that the emulation community takes, the causes that it tends to champion, and the behavior it tends to decry, one thing in particular tends to strike this former hacker as a little odd. Why isn't the emulation community policing itself more effectively? Anarchy invites outside intervention, and that's exactly what's been happening these past two years. A first-time visitor to the emuscene is most likely to think that we weren't policing ourselves at all, and that we tend to pretty much do what we want when we want and how we want no matter who it hurts, just like our bastard cousins over in warezland. Take it from me - I know, because I was a pretty big software pirate back in my day, schooled in all the tricks and familiar with all its forms. You want to know a secret? We don't look all that different than the warez scene to most eyes, and it's because you guys aren't acting any differently than the people you decry so much. ManBeast had it right in his editorial - you're pirates, folks, and there's no other way to define it. You have nobody to blame for that but yourselves. You, the emuscene, are to blame for all the bad stuff that's happened to you these past two years, and it's all because some of you don't know when to stop and leave well enough alone.

There are a few of us who are doing our dead level best to try and restore some sanity and legality to the emuscene, but it's like trying to fight our way upstream against a swift down-river current. We're trying to help the emuscene back on its feet, so it can be what it once was, but many of you seem content merely to reminisce about the "good ol' days" rather than restore the conditions that could bring them about again. I for one realized what I was doing wrong several months ago, and have been making a concerted, public effort to get my act together, but most of you don't seem to give a damn. As long as you can get your favorite "proggys" and all the "ROMs" you want, then everything's just fine and folks like me need to "shut yer hole" because there's no problem. "Vendors?" you sneer. "What vendors? Who needs 'em? They're all rich fat cats, anyway. They don't need our money. Besides, we're not hurting anyone, so shaddup and gimme some good ROMz."

It's time for a reality check, Mr. "I'm-too-good-to-pay-so-bugger-off" Towhead Emufan. You wouldn't have all those beloved "ROMs" past and present if it weren't for the vendors in the first place. They put up all of the money and put in all of the work that made them possible, and it's a lot more than you could ever do for them. Most of you don't even have the decency to go out and buy a used copy of that special favorite to halfway justify its "ROM," and that really gets the vendors wound up. *You are **stealing** the fruits of their labor* - there's no other way to put it - and they don't appreciate it one bit. That's why they call it piracy, and they will continue to beat up the emuscene until they either knock some sense into your collective heads or shut you down for good. It doesn't have to be this way, but that's where the emuscene is headed unless you guys come to your senses and grow up.

Emulation is legal - but software piracy is a crime. Think about it.

THEN AND NOW

I would like to beg your indulgence for a while in order to take you on a trip back into time, to a beautiful spring weekend in the month of May. The year is 1990. The place is an undisclosed location somewhere in the central United States. The reason for our journey is a much-anticipated event - the monthly "warez meet" of the Commodore aficionados from almost a half-dozen states within the region. It is a paid event with many regular attendees, ranging from the lamest of the lame lamers to the brightest and best of the pirates, hackers, phreakers, karders, and other purveyors of questionable and downright illegal activities of the day. Among the crowd who set up in the hall early that morning are yours truly and his two companions - Tinman (real name withheld by request), a "retired" CIA operative turned sometime hacker and unofficial keeper of the C64/128 flame in our area; and Steve "Skerran" Smith, whom you met briefly back at the beginning - a young Amiga graphics wizard I had befriended in the months following my return from military service. We had our own little pirate group, just like most of the folks who were in attendance with us, and the primary mission of the day was "to get the warez." We spent the better part of the morning getting to the prearranged location, whereupon we quickly paid our dues, unloaded our gear, and then cranked up our computer systems to maximum duplicating capacity.

It is difficult now, looking back, to put into words the sensations I used to get whenever I attended one of these meets. One can never forget the sound of a dozen or so C64/128 computers busily whirring away, merrily chirping the copy music from the *1541 Superkit* in unison (though rarely in sync). To say I miss those days would be something of an understatement, for I have not been to anything like them since, and only my recent experience with the emulation underground even comes close to matching it. They were part carnival and part market, pure adrenaline and no limits.

On one hand you might have a couple of Amiga gamers busily duking it out with each other on Epyx's *Barbarian* (off with his head!); on the other, you might have a half-dozen lamers fighting over the newest box of warez as if they were a pack of blue-haired biddies all trying to claim the last in-stock Cabbage Patch doll for themselves. Soft drinks and finger foods, the two food groups of the hacker, were in plentiful abundance. The hall would be solid Commodore from one end to the other, except perhaps for two or three "Beemer" users at the far end who largely stayed to themselves. I usually paid them sympathy visits and always tried to have something to share with them, since mine was often the only Bridgeboard-equipped A2000 in the place. The lamers mentioned earlier would be engaged in their mini-riot at the center of the hall, where the warez groups would stack box upon box of bootleg warez like free tracts at a tent meeting. No taking home, unless they gave them to you, but you could copy anything you wanted as long as you wanted, provided you paid your dues and had enough blanks to last until you finished or quitting time came. We never came home with empty disks, and I and Tinman made sure that we always got the latest stuff first. The older crap could wait - our users were counting on us to come back with the new stuff, and we could fix any bad copies at our convenience on return trips.

Approximately one-and-a-half hours after the event had officially started, an undercover operative working on behalf of the U.S. Secret Service entered the building and began to set up an Amiga system not far from the door. His identity was known to me, because Tinman had tipped me off through his law enforcement contacts that he might show up. Both he and I knew that Operation Sun Devil was already in full swing, but so far Tinman had kept us one step ahead of its reach. It will come as no surprise to you who are reading this when, acting on Tinman's advice, we immediately packed up and left. Only Tinman and I knew what was really happening; we did not share our knowledge with Skerran despite his objections at being rushed out before the meet was over. I did not tell the meet about the informant per Tinman's instructions; I was more concerned that the place was about to be raided and did not want to be caught inside when it happened. Federal agents stormed the building a mere half-hour after we had departed, by which time we were already a hundred miles away and headed to the presumed safe shelter of our respective homes. We had escaped the federal dragnet that snared so many others on that day - or so it seemed.

You will not find my name nor those of my companions anywhere within the Sun Devil records, both public and private. All trace of our involvement has been neatly expunged, and I suspect that Tinman had a lot to do with it. It turned out that he too was keeping the Secret Service apprised of the activities of this particular series of warez meets, due to the presence of the phreakers and karders, and we were fortunate enough to have been the group picked to provide his cover. It would not have been good for the press coverage if the Sun Devil dragnet had netted a "spook" in the process, so we were given just enough time to clear out before the raid came down. I was understandably upset at my unintended role in these events. I told Tinman as much about a month later while visiting him at his home, when he revealed as much of the whole story as he ever would.   He died of old age this past May, about the time that the very first version of the *EmuFAQ* was released, but we often reminisced about that fateful day when my life underwent a sudden and dramatic change.

"You knew it was going to happen, didn't you?" I demanded of Tinman.

"Yes, I knew," Tinman replied. "My friends inside warned me. That's why they let us go - my presence there might have raised some questions that they didn't want to answer. Besides, it was to their advantage to have one of their boys scoping out the earlier meets."

"Why?"

"Guess."

I thought about it for a moment, then shook my head angrily. "I don't believe this. You used us. You used me. I don't appreciate being used - not one bit."

Tinman reached up and took the cigarette from his mouth, then snuffed it on the porch rail. "Beats being arrested, doesn't it?" he said, looking straight at me. To that I could not reply.

It's things like this that make you think twice about being a software pirate.

I spent the next seven years keeping as far away from videogames as possible. It took me about a year to break up my widely acclaimed and massive collection of Amiga warez (many of which I have yet to see resurface again), and two years later I had sold my beloved souped-up A2000 and everything with it. I spent the time heavily involved on the business side of the counter - consulting, training, sales, management, and so on. I got to see and know the computer industry through the eyes of the vendors, and it was an interesting education indeed. As for my spare time, most of it was spent reacquainting myself with Japanese animation (anime) - in particular the so-called "third wave" that was just hitting the U.S. markets. I had been a fan of the genre ever since *Star Blazers* hit the U.S. television market, and it was a knowledge base that would one day prove useful in later research. Eventually, though, some seven years after Sun Devil, I felt safe enough and confident in my knowledge and experiences to brave the tempestuous waters of the Internet and enjoy the challenge of this bold new frontier in computing.

I first became aware of the Internet emuscene near the close of 1997, while I was surfing links for *DOOM* WADs and MP3 decompression utilities. I noticed a banner for a site called Amiga Emulators Central, and you can imagine the nostalgia that was stirred at the reference to my favorite PC and the best PC of its day (damn straight!). Several links later I found myself at Jan van Hertog's original Atmospherical Heights, scrolling down the *M.A.M.E.* list, and I was hooked - hard and fast. After a month or so, though, I felt rather guilty at not being able to help these youngblood hackers with their emulators, and that's when I came up with the idea of my FAQs. Harry Tuttle over at The Dump was the first to support me by posting the very first version of the *Genesis Game Guide* (aka *G3*), and things eventually developed from there. I had no idea that it would eventually wind up with the writing of the *EmuFAQ*, but I'm glad it happened - and so is the rest of the emulation community, too, judging from all of the positive email I'm getting.

So how did the *EmuFAQ* come about?

Back up a step to April of 1999. Nintendo had just released the first version of its policy statement regarding emulation of its proprietary videogame consoles, and it was met with a great deal of open derision by the emuscene. "They're just trying to intimidate us," was the common refrain, usually followed by, "It's a load of crap. They can't prove anything."

Or could they?

There had been several efforts to refute Nintendo's "emufaq," as it was fast becoming known, but I was quickly dissatisfied with all of them. They were written from the heart and amounted to nothing more than singing to the choir, as it were, not reasoned from the head and written for the masses. They attempted to refute Nintendo's rather broad and unsupported contentions with emotion and rhetoric instead of logic and reason. I was already appalled by the black eye the emuscene had received over the *UltraHLE* affair, so I decided to do something about it. I wrote my own refutal based on my understanding of the facts coupled with some initial research into the legal backing for Nintendo's claims, and Zophar's Domain was kind enough to post it for me. Needless to say, it caught everybody's attention and I received as many positive responses for it as I had over the *Genesis Game Guide*. Thus encouraged, I decided to press on with more thorough research on a revised refutal.

There's an old saying that is sometimes attributed to Oscar Wilde. "Be careful what you ask for - you just may get it."

It was during this period of expanded research into the legality of *UltraHLE* that I first stumbled across the *Sega v. MAPHIA* court case. Everything seemed fine until I began reading its logic regarding fair use, videogame cart dumps, and electronic transmission of same. I clearly recall getting a queasy feeling in the pit of my stomach and then reading it again, just to be sure. The fair use defense, which I had previously believed covered my emulation research, was not and never had been available to me as an average program user. I was potentially liable for multiple counts intellectual property infringement. I read it again, one more time. The injunction was too detailed, too precise, and I could not ignore its implications, being the thorough researcher that I am. I could sense the specter of Operation Sun Devil rising up from my past to haunt me once again as I shook my head in shock.

"No," I whispered, "please God, no. Not again! No, no, not again ...."

The emuscene remembers well that dark day when I pulled every single emulation-related document that I had either written or contributed towards from the Internet. Many were of the opinion that I was overreacting, but I knew better. Now that I knew the law, I was responsible for it. Now that I was liable for that knowledge, I could no longer claim innocent infringement. The attorneys for the vendors were high-priced, well-versed in the law, and very good at their jobs. The odds were that I would not get the same kind of break that I got during the Sun Devil raid. There would be no second chance.

I had to know. *Was emulation legal?*

Now I know - and so do you.

What will *<u>you</u>* do with this knowledge?

<u>LOOKING AHEAD</u>

I have always wondered, in a metaphysical sense, why I was spared the fate of my fellows on that dark day back in 1990. The stated reasons are obvious enough, but I've always sensed there was more behind it. Call it what you will - the hand of God, the mercy of Allah, divine providence, chance of fate, blind luck - but I remain of the firm conviction that there was more at work that day on a higher plain than I realized at the time. Looking back at those days, and what has happened in my life between then and now, it seems in a very real sense that I was being prepped for this day, when I would be needed to use my knowledge, highlighted by my many experiences and augmented by my talent for research, to address the burning questions about the legality of emulation. I am one of the few people on the scene uniquely qualified to look at it from all perspectives, which is why I have tried so hard to be even-handed in my treatment of the subject. To be honest, I'm really no different that those of you who are reading this document. I'm human just like you, I live a life not all that different from yours, and I enjoy many of the same joys and endure the same mistakes as do you. The only difference between us is that I was not afraid to speak out in defense of a beloved pasttime, and it is now enriched as a result. I am naught but a voice from the past crying out to the present in an effort to save the future. It gives me great satisfaction to know that you have listened to me, and that I have indeed made a difference. Almost all that I have put forth in the *EmuFAQ* has been confirmed by Judge Legge's ruling in the *bleem!* case. Let us hope that I will ultimately be vindicated.

If there is an ultimate conclusion to be reached with the *EmuFAQ*, it is this. Like any other form of technology, emulation is at best a double-edged sword. It has its "light" side, and it has its "dark" side, just like the mythical Force so prominently featured in the *Star Wars* saga. The dark side is the quicker, faster, easier path to tread, but its ultimate destination is a bottomless void from which few return. The light side requires discipline, dedication, and above all patience, but its rewards are greater by far in the long run. Emulation represents great promise, but it also holds great pitfalls. It should be used to promote the common good, not abused to promote uncommon lawlessness. There are specific guidelines in place that the emuscene should follow in order to restore and maintain its once-favored reputation, and it needs to do so posthaste. We now have the momentum to carry this new age of emulation to untold heights; but that same momentum, if abused, could propel us to such depths that the technology could be ultimately withdrawn from us for the remainder of our natural lives. Do not repeat the mistakes of the past, emufans. We cannot afford another period such as the past two years have been. Emulate with your head - not your heart.

WINDS OF CHANGE

The light of a new dawn finds the old battlefields that we know so well lying still and silent. A chill mist hangs in the air, partly hiding the bodies scattered hither and yon across the plain. One can just make out the outlines of the medics and the litter bearers moving among them, tending to the injured or carrying the wounded away. Oftentimes they will stop and mutter a silent prayer over a still form and then move on, following the muffled moans to the next one down the line. Off in the distance beyond the mists one can hear the sounds of a great host slowly retreating, while from behind come the calls and rattling of new forces being drawn up, both fresh and veteran, readying themselves for the next confrontation. I sit to one side, pen and paper in hand as I have done these many months now, and do my best to try and put into words what I have just seen unfold before my eyes.

We have won.

For the first time in what seems like an eternity, the winds of change are beginning to blow across the battle lines of the emulation war. They catch the mournful mist, tearing it into ribbons and speeding the tatters away. It is a brisk breeze with the scent of hope in it, a fresh fragrance that fans the previously low embers of my fellows into a bright and shining light. They march past me now, their eyes ablaze and a swagger in their step that was not there before, while from somewhere ahead a cocky cadence begins to call. I know where they are going, because I will be going with them. Our front has moved forward, because our enemy has suffered a major defeat. The forces that would stop free emulation are now in slow retreat while they attempt to regroup themselves, and have left many casualties behind in their wake in their long-fought struggle to resist the inevitable.

We have won an important victory.

For myself, I am glad to see that the NextGen wave of emulation will be echoing the new systems coming down the line, forcing the emuscene to once again work with their favorite programs in their original format - just as it was back in my time. Today's systems share a lot more in common than the average person might think, and the adoption of a common delivery system (irrespective of custom vendor formats) will do much to make the emulation dream a practical reality for the second time around. The hardware of today and tomorrow will be so powerful that emulation could very well once again become the accepted norm as it once was, not the despised and maligned exception to the norm that it is today. We are inexorably returning to the days when original systems and their emulators can sit side by side, each running the intended software within its intended delivery system without having to resort to extremes. It will be good to see those days once again.

As for me, I still consider myself a part of the emuscene, even though I am not as active as I once was. There will come a time, as it does to us all, when I will have to step aside and let it go on without me, but that will not be the sad day that I once thought it might be. We know from where we have come, we know where we now are, and we know where we are going. That's more than can be said for emulation a year ago, and I am glad to have played my small part. Even so, I still have a project or two to finish, and I never like to leave things undone. My ears prick up as I hear the call from the columns marching by.

"Are you coming?"

"You bet!" I reply, "just give me a minute to get my stuff."

And as I grab my gear and hurriedly rush to join them, I cannot help but feel a sense of awe at what is taking place. Emulation is an inevitable reality, part of the natural evolution of computing technology. It will happen to all systems sooner or later, given enough interest in the platform. The original vendors as a whole will not care for this, and will do everything in their power to stop it, but they are fighting a losing battle and they know it. Our quest to win the emulation war will not come easy, for there are many battles still to fight. The taste of our recent victory is sweet and both inspires the will and moves the heart, but the head knows better than to rest easy. The tide may have turned in our favor for now, but the war for the heart and soul of the technology known as emulation rages on. And as I take my place in the marching columns and keep time with the count of the cadence, I cannot help but reflect on the famous quote by General George S. Patton, arguably one of the finest military field commanders who ever lived.

> For over a thousand years, Roman conquerors returning from the wars enjoyed the honor of a triumph - a tumultuous parade. In the procession came trumpeters and musicians, and strange animals

> from the conquered territories, together with treasure and captured armaments. The conqueror rode in a triumphal chariot, the dazed prisoners walking in chains before him. Sometimes his children, robed in white, stood with him in the chariot or rode the trace horses. A slave stood behind the conqueror, holding a golden crown, and whispering in his ear a warning - *that all victory is fleeting*.

We have won an important victory. The vendors will not forget it.

REVIEW QUESTIONS

1. For what particular "classic" computer system does the author express profound nostalgia? Why?

2. What is the truth behind the phrase, "classic computing never dies?"

3. What is the fifth reason for emulation? Why is it so important?

4. What message does the author share with the vendor community? Explain.

5. Describe in your own words the problems that a close-minded vendor will ultimately face.

6. What is the inevitable reality that faces all proprietary technology?

7. What are some of the parallels that the author draws between the warez scene of the late 1980s and the emuscene of the late 1990s?

8. What important role do vendors play in making the emuscene ultimately possible?

9. In the event that the author relates from his past, how did he manage to escape being arrested for software piracy? What was the real reason for his so-called escape? How did he react, and what effect did it have on him?

10. Describe in brief the chain of events that resulted in the writing of the *EmuFAQ*. What was the ultimate reason that provoked the author to undertake this task?

11. What is the "ultimate conclusion" that the author draws about emulation? Do you agree with him? Why or why not?

12. In your own opinion, why does the author close the *EmuFAQ* with the famous Patton quotation?

THOUGHTS TO PONDER

1. Knowing what you do now, what is your opinion with regards to the legality of emulation in general? What about specific areas, such as computer or videogame emulation?

2. Will it ever be possible to separate videogame emulation from software piracy? Why or why not?

3. What are some things that the vendor community could do to change its relationship with emulation?

4. What are some things that the emuscene could do to change its relationship with emulation?

5. What would you like to see happen to the emuscene?

Y2K Addendum: The Effects

Sony v. Connectix - Analysis and Implications

> *We believe that this landmark decision will have broad ramifications throughout the software and other media industries. This ruling supports the clear goal of U.S. copyright law to allow fair use of prior works to create new intellectual property which broadens consumer choice. This is a great victory for every American.*
>
> - Roy McDonald, CEO of Connectix, Inc.

THE PARADIGM SHIFT REALIZED

For over a year, the first legal battle to draw blood in the great emulation debate of the late 1990s had been working its way though the courts. The fight between Sony Corporation and Connectix, Inc. over the ***Virtual Game Station*** (a PlayStation emulator for the Macintosh personal computer) was the first of its kind to address in a straightfoward manner the legality of emulation as a whole and videogame system emulation in particular. Sony was under the impression that it was going to win the dispute when on 22 April 1999 it won a permanent injunction against Connectix's product. The U.S. 9th Circuit Federal District Court ruled that Connectix had made improper and infringing use of the actual microcode contained within the PlayStation BIOS, which would be a violation of the U.S. Copyright Act (17 USC 101, et. seq.). It also found that the Connectix product had the potential of "tarnishing the Sony PlayStation [trade]mark," thus constituting trademark infringement under the terms of the Landham Act (15 USC 1051, et. seq.). As a result, it ordered Connectix to pull *Virtual Game Station* from the market and halt all moves towards preparing future retail releases of same. Connectix disagreed with the court's findings and appealed the decision. The case was subsequently reviewed by the U.S. Court of Appeals some five months later, beginning on 14 September 1999. The 9th Circuit Court of Appeals was well-schooled in the application of intellectual property law in the Information Age, having already made a number of landmark decisions with regards to its role in the ever-changing field of computer technology. On 10 February 2000, Judge William C. Canby Jr. issued the following decision on behalf of the full Court of Appeals.

> ***The intermediate copies made and used by Connectix during the course of its reverse engineering of the Sony [PlayStation] BIOS were protected fair use, necessary to permit Connectix to make its non-infringing Virtual Game Station function with PlayStation games. Any other intermediate copies made by Connectix [of the BIOS] do not support injunctive relief, even if those copies were infringing.***

The court's decision hit Sony like a bombshell. To make matters worse, Connectix promptly filed a motion with the district court that, in light of the appellate court ruling, Sony's lawsuit against them should be thrown out in its entirety. Connectix employees were rejoicing in their sudden good fortune, from the lowest clerk all the way up to its corporate board of directors. "Everybody's really excited," noted Connectix CEO Roy McDonald, "but a whole huge load of work has just landed on everybody's desk." He was referring to the long-stalled Macintosh update and impending Windows port of *Virtual Game Station*, both of which had been languishing in limbo ever since Sony's restraining injunction had been granted. Now that the legal hurdles had been cleared, *Virtual Game Station* could rightfully resume its place next to *bleem!* on retail store shelves as a viable alternative for buyers interested in running PlayStation games on their personal computers.

Sony v. Connectix is the single most important court decision to date concerning the legality of emulation. It has established beyond a shadow of a doubt the legality of the technology within the bounds of intellectual property law. This was the point that I was trying to drive home in the original *EmuFAQ,* and the basis for my stance had been asserted by the emuscene long before my arrival. "Emulation is legal. It's that simple and not open to debate," noted longtime emucoder Steve Snake (*KGen 98, Sega Smash Pack*) when queried on the subject. His assertion has now been vindicated by the U.S. Court of Appeals. If these were Biblical times, some might consider him a prophet. In these modern times, though, he must content himself with a smile and a smug, "I told you so."

There has been a lot of argument since the release of the original *EmuFAQ* over the merits of case law as opposed to statutory law. I must remind you that both hold equal weight within the U.S. legal system. Each augments the other. Statutory law is established by Congress, then case law interprets those statutes for a given situation. If the statutory law is found to be wanting or in error, then the courts have the constitutional right to re-interpret or nullify it with case law. It is then the responsibility of Congress to issue new statutory law which takes that case law into account over the situation in question. That is how the system works, whether you like it or not, and it has worked pretty well for the United States ever since the Constitution became the basis for our laws. A lot of you have taken me to task for backing up my arguments in the *EmuFAQ* with case law, failing to understand how the system works. It would be interesting to see how these same individuals feel now that a major case law ruling in their favor has finally come down the pike.

Why I am making such a big deal about this ruling? Is it just because of the subject matter, videogame emulation? No, there's more to it than that. This decision was issued by the U.S. Court of Appeals - not a federal district court, nor a state court, nor a local court. There is only one federal court higher than the Court of Appeals, and that is the U.S. Supreme Court. Because of this, federal appellate court rulings carry a lot of weight in our nation's legal system at all levels, and they play a major role in shaping the development of statutory law. While *Sony v. Connectix* does not refute any current intellectual property laws, it permits the legal system to better apply those laws to the concept of computer system emulation. Only twice before (to the best of my knowledge) has our federal legal system dealt with computer system emulation in any form (*Penwalt v. Durand-Weyland*, 1987 and *Atari v. Nintendo*, 1992). Neither case boded well for the emuscene, as the first inferred that emulation could be an infringing technology and the second demonstrated an example of the possibility realized. *Sony v. Connectix was the first federal court case to deal directly with the development of a potentially infringing emulator. Coupled with the fact that its resolution came by the capable minds of the U.S. Court of Appeals, it becomes a case law ruling of utmost importance to everyone within the emuscene.*

THE IMPORTANCE OF PATENT PROTECTION

One of the most unusual aspects of this case is that Sony did not claim patent infringement on the part of Connectix. To quote from the docket's introductory paragraph's:

> Sony has asserted no patent rights in this proceeding.

One would think that this would have been their first line of attack. Back in 1982, Atari sued Coleco for patent violation over the ColecoVision Expansion Module #1, an add-on adapter that allowed a stock ColecoVision to handle and play Atari VCS (2600) games. Videogame consoles have almost always contained patentable parts since those early days. The Nintendo Entertainment System (NES) led the pack of Atari's successors back in 1984 with a patent for the first antipiracy security system (APSS) designed for a home videogame console, and others have patented other parts of their respective systems. For example, Atari successfully sued Sega back in the heyday of the Sega Genesis for patent infringement, since the Genesis gamepads and gamepad ports were essentially unlicensed modifications of the ones that Atari had first developed for the Atari VCS. I can only speculate that Sony's legal team initially felt that no patentable part of the PlayStation had been violated by the actions of Connectix; therefore, they did not have a case for patent infringement. It would have been quite a different affair if Sony had made such a claim. If you will recall from our discussion in the *EmuFAQ*, the standards for patent protection are considerably more stringent than any other form of intellectual property protection. Accordingly, the protection it affords is more complete. It is conceivable that Connectix might not have able to make their case of "fair use" had Sony claimed the higher standard of patent infringement instead of copyright infringement.

Sony's decision not to file patent infringement most likely arose from case precedent. There is the dismissal of the *Atari v. Coleco* case to consider. The courts threw out Atari's case on the basis that Coleco's product did not violate any Atari patents. Also, as a rule, violations of a computer system BIOS are addressed under copyright law, not patent law. The standards for patent protection require that the object, concept, or process being protected must be unique in some way from other comparable products (35 USC). The basic concepts that underlie the processes and operations of a computer system BIOS are generic and therefore unpatentable; however, the object code stored within that BIOS is eligible for copyright protection. This was first established in *Tandy v. Personal Micro Computer* and later affirmed in the famous *Apple v. Franklin* and *IBM v. Compaq* decisions. The precedent had already been established that a computer system BIOS was a protected form of expression under copyright law; therefore, Sony's decision to go with the copyright infringement charges probably seemed the most likely to succeed. Unfortunately, it also proved to be Sony's undoing, as we shall shortly see.

INTERMEDIATE COPIES REVISITED

It was during the final development stages of the *EmuFAQ* back in mid-1999 that I had the opportunity to contact Connectix regarding their legal dispute with Sony. While they could not give me a straight answer at the time as to the legality of *Virtual Game Station*, I learned that their approach would be through the *Sega v. Accolade* decision of 1992. This caused me to re-examine my own questions on the subject and eventually led toward the development of the **A-Max test** - the *EmuFAQ*'s seven-point test for determining the legality of any emulator. It also led me to the conclusion that there were certain things that emulator programmers could do that emulator users could not - like dumping the ROMs within videogame cartridges for testing the accuracy of any emulator they might be developing. Not only has *Sony v. Connectix* vindicated my assertions, it has taken them one step farther.

Sega v. Accolade is the case most frequently cited regarding the legality of **reverse engineering**. This is the technique by which computer programmers independently develop a non-infringing compatable copy of any given system or program through close and detailed examination of the original. This is the case upon which the reverse engineering clause of U.S. copyright law is based (US 17 CR 1201, paragraph f). It introduced the legal concept of the **intermediate copy**, which is a copy of copyrighted computer code generated from an original vendor product in order to develop a non-infringing product. It does not matter how the intermediate copy is produced so long as it is made, but it is significant to note that this dispute involved the dumping and disassembly of object code originally stored in ROM. Here are the exact quotes from the decision docket concerning the legality of Accolade's intermediate copies of Sega object code.

> ... [W]e conclude that Accolade copied Sega's code for a legitimate, essentially non-exploitative purpose, and that the commercial aspect of its use can best be described as of minimal significance ... We conclude that where disassembly is the only way to gain access to the ideas and functional elements embodied in a copyrighted computer program and where there is a legitimate reason for seeking such access, disassembly is a fair use of the copyrighted work, as a matter of law.

Since the intermediate copy is utilized in the production of a non-infringing product, and since the copy in question does not become part and parcel of the resultant product, it is held to be a non-infringing copy of original vendor code. In addition, the court found that Accolade's use of Sega-derived intermediate copies to develop its own non-infringing software passed the "fair use" test of copyright law (17 USC 107). With this in mind, let us now examine the *Sony v. Connectix* ruling and how the concept of intermediate copying was brought to bear on this dispute.

You will recall my assertion from the *EmuFAQ* that it was illegal to make any kind of copy of a system BIOS for any reason whatsoever. This was based on the *Franklin* precedent. *What Sony v. Connectix does is to take the concept of making intermediate copies from videogame code stored in ROM and apply it to intermediate copies made of code stored in any kind of ROM - in this specific case, the BIOS of a computer system.* This does away with my *EmuFAQ* assertions about BIOS dumping, as the fundamental concepts that supported my earlier stance have now changed significantly. I have no doubt that this turn of events will be good news to those of you who are hard at work developing for the emuscene.

For starters, the court reaffirmed the *Sega v. Accolade* precedent that reverse engineering is frequently the only means by which developers can gain access to the fundamental concepts that underlie computer code. It also affirmed the right of Connectix to seek protection under the "fair use" statute (17 USC 107), just as Accolade had done in its dispute with Sony some seven years earlier. In doing so, the court invoked that favored darling of the "free anything" crowd, *Sony v. Universal* (i.e. the "Betamax case."), saying, "The fair use doctrine preserves public access to the ideas and functional elements embedded in copyrighted computer software programs." Under the requirements of that statue, the Court of Appeals was then required to apply the "fair use" test to Connectix's claim.

> *Nature of work* - PASSED. Under the terms of the *Sega v. Accolade* precedent, the court noted that it would be wrong for Sony to erect an "artificial legal barrier" against anyone who wanted to examine the PlayStation BIOS object code. *The only way that it could erect such a barrier was* ***under patent law****, not copyright law*. Copyright law was not designed to grant the kind of protections that Sony was seeking for its BIOS code. *Since Sony had failed to assert patent protection for the PlayStation BIOS, they had no grounds for claiming violation of their intellectual property under copyright law.* Sony counter-claimed that there were other methods available to Connectix for developing their non-infringing code, and that making multiple copies of its dumped BIOS were unnecessary to their development effort. The court refuted Sony's argument, noting that "this construction stretches *Sega* too far .... ***If Sony wishes to obtain a monopoly on the functional concepts in its software, it must satisfy the more stringent requirements of the patent laws .... This Sony has not done.***"

> *Amount and substantiality of use* - DISMISSED. Connectix had not only dumped the entire Sony PlayStation BIOS, but they had also made multiple copies of it. Furthermore, they had used these copies in developmental copies of *Virtual Game Station* while debugging their own non-infringing code. One would think that this would have ended Connectix's case right there, as their actions failed this part of the fair use test. In an unusual move, however, the court threw out this part of the fair use test. Why did the court do that? The answer can be found in *Sega v. Accolade*, where the same thing happened with regards to Accolade's use of the fair use defense. That court had ruled that if a final product based on intermediate copying did not contain any infringing code, then this part of the fair use test "was of very little weight." Accordingly, the Court of Appeals followed case law precedent and dismissed this part of the fair use test as irrelevant.
>
> *Purpose and character of use* - PASSED. The court noted that *Virtual Game Station* had been designed to supplement the PlayStation, not to replace it. It was an entirely new product, despite the fact that it had remarkable similarities to the actual hardware. In addition, the final product used Connectix-developed code, not original Sony code. Accordingly, the court was "at a loss" to see just how Connectix's development efforts, which resulted in a non-infringing product, violated any Sony copyrights. *Even though Connectix's Virtual Game Station was running PlayStation games in an emulated environment on a personal computer instead of on actual PlayStation hardware, no harm came to Sony or its products as a result.*
>
> *Effect of use on potential market* - PASSED. The court acknowledged Sony's claim that it could lose "console sales and profits" in the PlayStation videogame market due to the presence of *Virtual Game Station*, but held that this made Connectix a "legitimate competitor" and not an infringing party. To quote the court, "***Sony understandably seeks control over the market for devices that play games Sony produces or licenses. The copyright law, however, does not confer such a monopoly.***" They based their finding on *Sega v. Accolade*, which said, "[Any] attempt to monopolize the market by making it impossible for others to compete runs counter to the statutory purpose of promoting creative expression and cannot constitue a strong equitable basis for resisting the invocation of the fair use doctrine."

Finally, the Court of Appeals noted the statutory requirement (17 USC 107) that all four factors of the fair use test had to be weighed equally before any "fair use" claim could be granted. As the only factor that could have killed Connectix's claim had been thrown out per case law precedent, the court granted their request. *"[W]e conclude," reads the docket, "that Connectix's intermediate copying of the Sony BIOS during the course of its reverse engineering was a fair use .... With respect to its claim of copyright infringement, Sony has not established either a likelihood of success on the merits or that the balance of hardships tips in its favor."* With that, Sony's claims of copyright infringement went down in flames at the hands of the Court of Appeals.

<u>TRADEMARK TARNISHMENT</u>

The final issue which the Court of Appeals had to address was Sony's claim of trademark infringement. Sony's "derringer shot" took the form of a tarnishment charge (15 USC 1125.c.1). The concept is akin to the oxidation of metal - the dark film that appears on polished silver - from which the legal concept was derived. Sony asserted that its world-famous PlayStation trademark was being tarnished by the arrival of an inferior competing product, *Virtual Game Station*. There is a four-part test for trademark tarnishment that is similar to the "fair use" test for copyright infringment. It is this test that the court applied to Sony's claim.

> *The PlayStation mark is "famous"* - Connectix did not dispute this obvious fact.
>
> *Connectix is "making commercial use of the mark"* - The Court of Appeals chose not to address this element of the test. Connectix was selling *Virtual Game Station*, not bootlegging PlayStations. Both were distinct and unique products - an observation upon which they elaborated in the fourth part of the test.
>
> *Connectix's "use began after the mark became famous"* - Again, Connectix did not dispute the obvious. I remind you of one of the many observations from the *EmuFAQ.*
>
> > An emulator by its very nature is an after-the-fact product. Remember, emulators are designed to replicate the functions of systems that are either already in existence or no longer being produced. You cannot make an emulator unless you have two things - knowledge of the system you are emulating and access to that system's software base.
>
> *Connectix's "use of the mark dilutes the quality of the mark by dimishing the capacity of the mark to identify and distigush goods and services."* - The legal concept of dilution is identical to the scientific one. For example, an alcoholic beverage diluted with water is not as strong as the original. The more you dilute the drink, the lower its "proof" rating and the lessening of the probability

> that it will make you drunk. In a similar fashion, Sony claimed that *Virtual Game Station* diluted the worldwide reputation of the original PlayStation. Since it was not as good as the actual hardware, unsuspecting users might associate its problems with the PlayStation itself. While Sony was able to convince the lower court on this basis, the Court of Appeals rejected this claim out-of-hand. They made two key observations in this regard. First, "the evidence on the record does not support such a finding of misattribution." The court noted that the only studies on the matter clearly distinguished the differences in quality between the two products but "shed no light in the question of misattribution." Second, "the evidence here fails to show or suggest that Sony's mark or product was regarded or was likely to be regarded negatively because of its performance on Connectix's *Virtual Game Station*." The court used Sony's own compatability studies to note that its testers were clearly able to distinguish the gaming experience delivered by both products running PlayStation games. While their opinions about *Virtual Game Station* ran the full range from "generally acceptable" to "nearly unplayable," all of the testers could make the distinction between the two. In fact, most of them "preferred the PlayStation experience over the Virtual Gaming Station experience."

On the basis of the evidence at hand, the Court of Appeals summarily rejected Sony's claim of trademark infringement by Connectix. With that, Sony had no case left to bear against Connectix.

THE VERDICT

The court's final conclusion in the case of *Sony v. Connectix* reads as follows:

> ***Connectix's reverse engineering of the Sony BIOS extracted from a Sony PlayStation console purchased by Connectix's engineers is protected as a fair use. Other intermediate copies of the Sony BIOS made by Connectix, if they infringed Sony's copyright, do not justify injunctive relief. For these reasons, the district court's injunction is dissolved and the case is remanded back***

IMPLICATIONS FOR THE EMUSCENE

"Cool!" I can hear many of you say, "but spare the legalisms, please. Just give me the straight dope. Just how exactly does *Sony v. Connectix* affect the emuscene? Put it in plain language, Scribe!" Okay, here goes.

It is perfectly legal for you to use a emulator to execute software written for the original system. There's no more questioning the legality of emulation. *Sony v. Connectix* finally settles that issue once and for all. Both developing and using an emulator are protected under the "fair use" statute of copyright law. Whew! That's a relief for every emufan out there, myself included. Remember, it was my investigation into the "fair use" defense for "ROM" dumping that got me started on the *EmuFAQ* in the first place. While you're celebrating the emuscene's victory, however, don't forget that the Court of Appeals never dealt with the issue of "ROMs." This means that the "original format" requirement for the software you plan to run under emulation still stands (17 USC 106.2, c.f. "derivative work;" see also *Nintendo v. Computer and Entertainment*, 1996). I'm sure Nintendo will be happy to remind anybody who might attempt to convienently forget that point.

It is perfectly legal for a emulator developer to dump, decompile, and examine any object code stored within any ROM that is part of the system being emulated if such access is necessary for the production of a non-infringing emulator. This will be the biggest legal boost that emulation development gets out of this case. If you're a emulator developer and working within the strict legal confines of the concept of reverse engineering, then you can dump just about anything you need that isn't patented - the BIOS, the chipset, the CPU, the sound processor, the VDP, and whatever else is soldered to the planar - in order to get your emulator up and running. *There's only two "gotchas" left to worry about. You can't mess with any code that is patented or contains patented code, and the public release of your emulator can neither contain any of the original object code you dumped nor require the use of same.* Ouch. *For that matter, you can neither post a dumped BIOS on the Internet nor download one that is already posted* (see the actual decision brief, footnote 11). That means that any emulator you release has to be 100% non-infringing. Sorry, *PSEmu* fans, no relief here - your requirement of a BIOS dump is addressed in this case. You can't release an emulator that requires a BIOS dump because users don't have the legal right to download or dump a BIOS - even if they *do* own an original console. Only developers have that right, and not everybody's a developer. Just claiming to be one doesn't make you one. You real developers out there should be on your knees praising the Court of Appeals for broadening the terms of *Sega v. Accolade*. You didn't have such a broad right before *Sony v. Connectix*, but you do now. Try not to abuse it, okay? Remember, what the courts grant, they can also take away.

> **Any claim of potential market impact made by an original system vendor against the developers of a non-infringing emulator is without merit.** If you will recall from the *EmuFAQ,* one of the front-line attacks made by vendors against emulation in the past is trademark infringement. Emulation impacts the market for a vendor's products, thus decreasing revenue and reducing potential profits. To paraphrase the judgement by the Court of Appeals in this case, this is a groundless argument that cannot be proven. In other words, it's pure bull$#!+. Nothing more, nothing less. Only an uneducated moron could confuse a software-based emulator with the actual original hardware. The public as a whole will trend towards distinguishing the two as separate entities, and will gauge potential purchase of either based on their own respective merits. "It is a fact that emulators are never as good as the actual hardware," to quote the *EmuFAQ* on this subject, and the Court of Appeals determined that the public as a whole would be able to distinguish between the two. *Emulation is not the monster threat to profits that certain vendors made it out to be. It never was, and never will be.*

I would now like to take this time to remind you of something that I brought up in the very first module of the *EmuFAQ*. It is the **A-Max precedent**, in which I first posited the legality of emulation technology.

The *A-Max* precedent
It is lawful for a third party to develop an unauthorized emulator for a proprietary system, provided that said emulator cannot be shown to violate the intellectual property rights of the original system vendor in any way.

The U.S. Court of Appeals has confirmed the legal validity of this precedent. "The *Virtual Game Station* itself infringes no copyright," reads their comment in one of the footnotes. In upholding this precedent, though, they have also augmented it with another one of equal importance. This I shall call the **Connectix precedent**, named for the remarkable company that made emulation a legal reality.

The Connectix precedent
It is unlawful for an original system vendor to in any way inhibit or impede the development, distribution, or usage of a non-infringing unlicensed third-party emulator for one or more of that vendor's proprietary systems.

Both of these precedents, taken together, can be said to set the legal "bookends" for emulation technology. The *A-Max* precedent addresses the emuscene's point-of-view, whereas the Connectix precedent takes care of things from the vendor's perspective. Between these boundaries lie a broad legal space wherein emulation can be lawfully pursued.

As the matter stands now, Sony has brought yet another lawsuit against Connectix. On 16 February 2000, Sony filed with the U.S. 9th District Court claiming 11 different counts of patent infringement against Connectix over the technology that *Virtual Game Station* emulates. Sigh. Some folks never learn. They should have done that in the first place. Sony is going to have to defend its self-imposed monopoly from a weaker position than before. The emuscene now has the legal right to emulate, and that will be an important factor in weighing the merits of Sony's claim. All Connectix has to do is to prove that no Sony patents were violated, and that is going to be easier that most might think. It is ironic that Sony's misinterpretation of the law has done the most in making the legalization of emulation possible. Sony has a rather big job ahead as it tries to dig itself out of its own hole. This time around, the emuscene holds the high ground.

REVIEW QUESTIONS

1. What two forms of intellectual property protection did Sony seek against Connectix? What was the ruling of the district court? What was the ruling of the appellate court?

2. Why is *Sony v. Connectix "*the single most important court decision to date concerning the legality of emulation?" How does it differ from previous court cases regarding emulation?

3. What was one of the most unusual aspects of Sony's claims against Connectix? How did this ultimately work against them?

4. What is the significance of *Sega v. Accolade* with regards to the resolution of this case?

5. Describe in your own words how *Virtual Game Station* passed the "fair use" test of copyright law. What was so unusual about the manner in which the court resolved the test? Was there legal precedent for their action? If so, what was it? If not, where did they err?

6. If Sony were to press its claim for a monopoly on the market for PlayStation games, under what form of intellectual property protection could it do so? What makes it different from other forms in this regard?

7. What is "trademark tarnishment?" Can you describe how it works?

8. What were the two reasons behind the court's decisionregarding possible PlayStation trademark tarnishment? Can you describe the key role that product testers played in that decision?

9. What are the three conclusions that the emuscene can draw from *Sony v. Connectix?*

10. What major legal precedent regarding the activities of the emuscene was established by this case? How does it complement any earlier precedents?

THOUGHTS TO PONDER

1. What is the legal basis for Sony's new suit against Connectix? How does this differ from their previous action?

2. In your opinion, does Sony have grounds for their new claims? Why or why not?

3. Do you think Sony will succeed in its new case? Why or why not?

4. Is it possible to code, release, or use an emulator that does not violate the most stringent form of intellectual property protection? Be sure to justify your answer.

Sources

- *Apple Computer, Inc. v. Franklin Computer Corporation.* (1984).
- *Atari Games Corp. v. Coleco.* (1983).
- *Atari Games Corp. v. Nintendo of America, Inc.* (1992).
- *Atari Games Corp. v. Sega Enterprises, Ltd.* (1991)
- Bowman, Lisa M. "Connectix bests Sony in legal dispute." *ZDNet News*, 10 February 2000.
- "Connectix Announces Victory in Sony Lawsuit." Connectix, Inc. press release, 10 February 2000.
- Lloyd, David. "A Polite Reaction." *OverClocked*, issue #86, 12 February 2000.
- Pettus, Sam. "The Basis for Emulation." *Emulation: Right or Wrong?* Internet document, 1 October 1999.
- ----------. "Playing With the Software Base." *Emulation: Right or Wrong?* Internet document, 1 October 1999.
- ----------. "Releasing an Emulator." *Emulation: Right or Wrong?* Internet document, 1 October 1999.
- *International Business Machines, Inc. (IBM) v. Compaq Computer Corp.* (1982).
- *Nintendo of America, Inc. v. Computer & Entertainment, Inc.* (1996).
- *Sega Enterprises, Ltd. v. Accolade, Inc.* (1992)
- "Sony Files New Suit Against Connectix." *Bloomberg News*, 16 February 2000.
- *Sony Computer Entertainment, Inc. v. Connectix Corporation*. (2000)
- *Sony Corp. of America v. Universal City Studios, Inc.* (1983).
- *Tandy Corp. v. Personal Micro Computers, Inc.* (1981).

Y2K Addendum: The Effects

The Question of ROMs

by Chuck Cochems (special guest writer)

Freedom and independence will survive capitalism.

- Rush Limbaugh, *The Rush Limbaugh Show*, 13 January 2000, 13:56 CST

SICK AND TIRED

Comedian Bill Cosby has a routine in which he talks about the way he and his little brother Russell would often aggrevate his exasperated mother . "I'm sick of this," she would roar at them, and "tired always followed sick," Bill Cosby would add with a grin. "Sick and tired. Sick and tired. I am sick and tired ...." You get the picture.

> You know, one day mother started into it. She said, "You make me sick." "And tired," I finished. [long pause while the audience laughs] I don't remember anything that happened that day.

You want to know something? I'm sick. I'm tired, too. I'm sick and tired. I'm so sick and tired of all this. No, I'm not having any trouble remembering what this is. I know quite well why I'm sick and tired. Do you want to know why I'm sick and tired? Well, I'll tell you.

I'm sick and tired of certain vendors who've made a nasty habit of screaming, "Software piracy!" every time they think something threatens their precious profits. I'm sick and tired of the way this crowd has used the notion of software piracy to bash the emuscene every chance it gets. I'm sick and tired of corporate fat cats who think emufans are always out to burn them, that we somehow have the means and the might to steal millions of dollars from them just because we want to play our games on somebody else's hardware. I'm sick and tired of pointy-headed company mouthpieces who flatulate position papers and policy statements without backing them up with hard facts, who often condemn a thing without really taking a close look at it, who claim that any little thing we do to make our life on the emuscene easier supposely steamrolls over their so-called "intellectual property rights." I'm sick and tired of these twisted toads for calling out the SWAT teams every time I try to play a game that I legitimately purchased under emulation on my personal computer instead on their precious little console. That's why I'm sick and tired.

Had enough? Too bad. I'm just getting started.

The emuscene went gonzo with excitement whenver Sony got their face rubbed in their own sludge by the U.S. Court of Appeals, and I don't blame 'em one bit. I was celebrating right along with 'em. Sony's screwup has made unlicensed emulation legal for all time. They'll never take that away from us, and they'll have a helluva fight on their hands if they so much as try. That was the first mountain we had to climb, and thank goodness it's behind us now. It's time to climb the second and even bigger mountain standing in our way. You know the one I'm talking about. It's that dark peak right in front of us which has loomed over the modern emuscene ever since the Careless Gamer released *MegaDrive* to the public. It's now my turn to lead the quest for truth up the slopes of this final obstacle, and I intend to surmout and conquer it once and for all.

What am I talking about? I'm talking about ROMs.

CODE IS CODE

We all know what ROMs are. They're the old programs that you run on an emulator. These are either binary dumps of code originally stored in ROM (hence the name) or image files of old disks in obsolete formats. A run-of-the-mill personal computer (PC for short) can't handle the originals directly without specialized adapters, and these cost a lot of time and money to cook up. It's a whole lot easier to convert that old package of programs into somethat that my PC can handle. Besides, there will come a time when that old system and that old program base is going to up and croak. You won't be able to use it anymore unless you've converted it into a ROM for use with an emulator That's not me talking, that's the physics of entropy. That is the chief reason why ROMs exist, and that's also the cause of the problem.

It's nice to know exactly where the vendors come up with their silly and cockamamie "exclusive rights" claims. I didn't know until last year that this falls under a specific provision of copyright law, 17 USC 106. Let's take a look at it, 'cause I want to make sure everybody's on the level before I go on.

> *Subject to sections 107 through 120*, the owner of copyright under this title has the exclusive rights to do and to authorize any of the following ... (emphasis added - ed.)

Now let's home in on section 2 of this provision, because here is where the problem lies.

> 2) to prepare derivative works based upon the copyrighted work

Now compare that with this extract from *Nintendo v. Computer & Entertainment:*

> Changing the format ... from the original cartridge to a disk format also violates Nintendo's rights to prepare derivative works under 17 USC 106.2.

That, in a nutshell, is the primary argument of Nintendo and the rest of their ilk. They claim the absolute right of derivative works, which just so happens to include ROMs. You can't so much as sneeze on their programs without orders signed in triplicate, sent in, queried, sent back, revamped, sent in, sat on for a decade or so, and then lost in a filing cabinet stuck in a forgotten warehouse somewhere in the frozen tundra just outside of Vladivostok with a big fat padlock on the door. Performing a format change to a program that you have legitimately obtained? God forbid!

One of the things that has always irritated me about this whole issue is the unequal treatment given the two different types of ROM. Nintendo, Sony, and their kind scream bloody murder every time the issue of ROMs is raised, and they've managed to convince the courts that converting code from one type of storage to another is unjustified due to the format change involved. Well, I got news for you. *Code is code, damnit, regardless of the format!* They bitched just as much whenever folks began making cart adapters that would let you run their carts on other vendor's consoles almost two decades ago, and now here they are again with the same old crap argument. Well, what about disk image ROMs? Very few vendors seem to object to them, and when they do, it's that tired old scratchy "software piracy" Victorola creaking away in their press releases. It seems that there are things you can do with software-based ROMs that you can't do with firmware-derived ROMs, or so we're being led to believe. I'm told that it has to do with the vendor's intent as to the format that they want their program presented, or something like that, but that seems awfully weak to me. As I said, *code is code*. It doesn't matter one whit if it's in hardware or software, if it's stored on permanent or non-permanent media, or whether or not it gets changed from one format to another. If you're going to preserve that vendor's precious program, then you have to preserve the code itself. The format doesn't mean a damn thing, so long as you can still maintain the integrity of that code. That's one of the consequences of the Digital Revolution - we can change formats at whim and still have a working program. Attempting to lock a program into only one specific format is a useless gesture, because you can still access and execute that code irrespective of its format. A difference that makes no difference is no difference. *Data is data. Code is code*.

While I'm at it, let's take a look at this business of derivative works, shall we? If you'll glance back up at the opening lines to 17 USC 106, you'll see that I've emphasized the opening line.

> *Subject to sections 107 through 120 ...*

What this means, in a nutshell, that *a copyright owner's right to a derivative work is not absolute*. Not duplication, not derivation, not adaptation, not performance - ***<u>none</u>*** of them are absolute. That's why the vendors have had to cook up this EULA nonsense - they're trying to force restrictions on you which, in most cases, they have only a limited right to do and in some cases (reverse engineering, etc.) absolutely ***<u>no</u>*** legal right to do. So you see, these so-called "exclusive rights" that Nintendo and other proprietary-minded vendors aren't so exclusive after all. In fact there are some well-known exceptions to these rights.

It is a fact that certain forms of derivative work are not only permissible, but in some cases absolutely essential for proper or intended use of the program in question. This is possible because of that opening line to 17 USC 106, and is acknowledged in other parts of copyright law such as 17 USC 117's "operational adaption" clause". Need a good example? How about that overgrown behemoth of an operating system, *Microsloth Winblowze*? In the old days, you used to get it on a set of floppy disks. Nowadays, you get it on a CD-ROM. In order to make it work, in order to run it, in order to use it in the way the vendor intended, you have to install the thing to your computer's hard drive. As part of this process, it unpacks itself and sets itself up to work with your particular system. The end result is a set of code residing on your hard drive that bears little physical resemblance to that code sitting inside your disks or CD-ROM. You now have a derivative work that has gone through one helluva format change in more ways than one. Yes, that's an authorized derivative work - Microsoft gives you permission to do this in their EULA - but there are other similar cases in which such works are unauthorized. What about servicing a computer? Copyright law says you can make such derivative works all day long in order to service that machine, and that's not infringing (17 USC 117.c). What about the Game Genie case (*Nintendo v. Galoob*)? The Court of Appeals said that the Game Genie produced derivative works of NES videogames, but they also said that such works were ultimately non-infringing due to their temporary nature - they vanished once you turned off the console. In fact, *there are several different forms of derivative work that are deemed to be noninfringing* - vendor-approved copies, servicing copies, archival copies, intermediate copies, and "gameplay enhanced" copies. I wouldn't be surprised if there are still more out there that I've missed. Yes, this whole business of derivative works isn't as cut and dried as certain vendors would lead you to believe.

A NEW APPROACH TO AN OLD PROBLEM

So how do we get around this quandary? How can emufans find a legitimate means of coming up with ROMs of their legitimately purchased games for use with an emulator? That's the US$64,000 question, isn't it?

You know, I must have spent at least two months or more arguing with the Scribe on this very issue, and believe me, I tried all the regular arguments. Backup theory ... public fair use ... development purposes ... operational adaption ... and so on. He shot me down every time because of that silly, way-too-narrow definition of "derivative work" that is embodied in copyright law (17 USC 106.2). Yes, I was getting sick and tired of this, too. Why was he doing this? Wasn't he supposed to be the emuscene's best friend? Our original discussion ended when he cut me off, claiming that I was positing dogma instead of arguing logic. I wasn't being dogmatic, I was just trying to prove my point. After a while, though, once things had cooled down, I began to re-read our arguments, and something that he said caught my eye.

> If the emuscene is ever going to find justification for ROMs, cart dumps in particular, then they're going to have to find a new approach. None of the old arguments will work. Nintendo, Sony, the IDSA, and their allies have made sure of that. I think you're going about this the wrong way. If there is an answer, then I believe you'll find it in the Betamax case. That's the key to solving the problem.

He's right, you know. We can't keep on defending ROMs the old way. Those defenses won't fly in court. We, the humble emufans, plain and ordinary average users, don't enjoy the same legal rights as do the emucoders. They're developers. They get to enjoy developer rights. We're users. We don't. I guess it's time we stopped bitching about it and get used to that sad fact, 'cause its not changing anytime soon.

"So what can be done about it?" you ask.

Well, I think something new is needed, and I think I've figured out what the Scribe was driving at. It's going to require us to change our whole way of thinking about how we users can justify our ROMs. "You must unlearn what you have learned," Yoda tells Luke in *The Empire Strikes Back.* We're going to have to forget about the old defenses, and go with something new. You want to know something? I know how to do it.

"What is it?!" you eagerly ask.

The personal use defense.

THE BETAMAX CASE

In 1983, the U.S. Supreme Court issued a landmark ruling in the case of *Sony v. Universal* whose implications have dogged the monpolistic intentions of many a proprietary-minded vendor ever since. The so-called **Betamax case** is of particular interest to those looking for a legal means to make and use their own ROMs, because it is here where the concept of personal use was first defined. The case was argued on the merits of the fair use defense (17 USC 107), so I reproduce the statute for your convienence.

> Notwithstanding the provisions of sections 106 and 106A, the fair use of a copyrighted work, including such use by reproduction in copies or phonorecords or by any other means specified by that section, for purposes such as criticism, comment, news reporting, teaching (including multiple copies for classroom use), scholarship, or research, is not an infringement of copyright.
>
> In determining whether the use made of a work in any particular case is a fair use, the factors to be considered shall include --
>
> 1. the purpose and character of the use, including whether such use is of a commercial nature or is for non-profit educational purposes.
> 2. the nature of a copyrighted work.
> 3. the amount and substantiality of the portion(s) used in relation to the copyrighted work as a whole.
> 4. the effect of the use upon the potential market for or value of the copyrighted work.

> The fact that a work is unpublished shall not itself bar a finding of fair use if such finding is made upon consideration of all the above factors.

For now, I want you to sit back and watch the playback as the Supreme Court finds legal justifcation for the concept and act of personal use under the fair use statue (17 USC 107) of copyright law. In this particular case, we are dealing with the <u>unauthorized</u> recording of a live television broadcast <u>by use of a potentially infringing technology</u>, the Sony Betamax videotape recorder, <u>in a manner other than that intended by the program's vendor</u> (playback at a later time). You're going to hear references to "time-shifting," which was the court's legal shorthand for the primary issue at stake, so pay attention.

> [The fair use doctrine] identifies various factors that enable a Court to apply an "equitable rule of reason" analysis to particular claims of infringement. Although not conclusive, the first factor requires that "the commercial or nonprofit character of an activity" be weighted in any fair use decision. If the Betamax were used to make copies for a commercial or profit-making purpose, such use would presumptively be unfair. The contrary presumption is appropriate here, however, because the District Court's findings plainly establish that time-shifting for private home use must be characterized as a noncommercial, nonprofit activity. Moreover, when one considers the nature of a televised copyrighted audiovisual work (17 USC 107.2), and that timeshifting merely enables a view to see such a work which he had been invited to witness in its entirety free of charge, the fact that the entire work is reproduced (17 USC 107.3) does not have its ordinary effect of use.

Let's be clear right now about what the Supreme Court said. *A noncommercial, nonprofit use qualifies for fair use. <u>All</u> personal use is by its very nature noncommercial and nonprofit*. If you're going to go commercial, then you have to make the appropriate interpersonal arrangements. If you're going to make a profit, then you've got to have some form of commercial enterprise, legal or not. *The concept of personal use is based on the principle that all activity involved rests solely with you.* No one else is involved - not you family, not your friends, not your co-workers, not your email recipients - no one except you and you alone. Since no one else is involved, personal use cannot be considered commercial because you are lacking the one thing that drives commercial activity - the existence of a forum for commerce. You can't have a forum when there's only one participant; hence, all personal use is by its nature both noncommercial and nonprofit.

> This is not, however, the end of the inquiry because Congress has also directed us to consider "the effect of the use upon the potential

> market for or value of the copyrighted work" (17 USC 107.4). The purpose of copyright is to create incentives for creative effort. Even noncommercial purposes may impair the copyright holder's ability to obtain the rewards that Congress intended him to have, but a use that has no demonstrable effect upon the potential market for or the value of the copyrighted work need not be prohibited in order to protect the author's incentive to create. The prohibition of such noncommercial use would merely inhibit access to ideas without any countervailing benefit.

Sounds just like what a lot of us emufans of legitimate purpose have been going through these past two years, doesn't. *A use that has no demonstrable effect upon the market ... or value ... need not be prohibited....* That's part four of the fair use test - the one that Raymond Nimmer said was the most important. We've already passed the rest of the test, and you know as well as I what that means.

It doesn't stop just there, though. The Supreme Court isn't finished with its discussion. Let's read on, shall we?

> Thus, although every commercial use of copyrighted material is presumptively an unfair exploitation of the monopoly privilege that belongs to the owner of the copyright, noncommercial users are a different matter. A challenge to a noncommercial use of a copyrighted work requires proof either that the particular use is harmful, or that it should become widespread, it would adversely affect the potential market for the copyrighted work. Actual present harm need not be shown; such a requirement would leave the copyright holder with no defense against predictable damage. Nor is it necessary to show with certainty that future harm will result. What is necessary is shown by a preponderance of the evidence that some meaningful likelihood of future harm exists. If the intended use is for commercial gain, that likelihood may be presumed; but if it is for a noncommercial purpose, the likelihood must be demonstrated.

The Supreme Court realized that personal use is an entirely different matter, one quite different from commercial use, when it comes to potential intellectual property infringement. They acknowledged that personal use has the potential use to infringe, but not nearly as much as commercial infringement. Noncommercial uses, such as personal use, only have two hurdles to jump - "the particular use is harmful," and widespread activity "would adversely affect the potential market." This is why Bung lost their court battle with Nintendo over the Doctor V64 (*Nintendo v. Bung*), since Nintendo was able to prove that the device, while advertised as one that had personal uses, could obviously be used for infringing commercial purposes. I'm going to come back to cart dumpers a little later, but just remember this. Commercial use will always fail to jump these twin hurdles. Personal use is an entirely different manner. *If a vendor wants to void* *the personal use defense, then* <u>*it*</u> *has the responsibility of proving potential harm*. Sound familiar? It ought to. It's not your responsibility to establish noninfringement. It's the vendor's. They've got to prove that your activity goes beyond mere personal use.

So what happens if the vendor can't prove potential harm from personal use? Well, that very thing happened in the Betamax case. Let's see what the Supreme Court had to say.

> In this case, respondents failed to carry their burden [of proof] with regard to home time-shifting. The District Court described respondent's evidence as follows: "Plantiffs' experts admitted at several points in the trial that the time-shifting without librarying would result in 'not a great deal of harm.' Plantiff's greatest concern about time-shifting is with 'a point of important philosophy that transcends even commercial judgement.' The fear [was] that with any Betamax usage, 'invisible boundaries are passed: 'the copyright holder has lost control over his program.'"

Universal and its allies were arguing that they were losing control over their rights to their works - specifically, the time that they chose to dictate when and how their television programs could be viewed. Fortunately, the high court didn't buy that argument.

> Most of the plantiff's predictions of harm hinge on speculation about audience viewing patterns and ratings, a measurement system which Sidney Sheinburg, MCA's president, calls 'a black art' because of the significant level of imprecision involved in the calculations.

Whoa! *Vendors cannot speculate about the potential harm of infringement!* Why? Becasue <u>speculative damages cannot be predicted with any accuracy!</u> They have to provide reasonable proof, not wild assertions. Remember, one of the reasons Sony lost its lawsuit against Connectix was that it failed to prove infringement (*Sony v. Connectix*). Their own presumptive evidence was both contradictory and inconclusive, and the Court of Appeals nailed 'em on it. That means that the broad policy statements about "potential market impact" for which certain vendors (a-hem!) are just that - statements. They don't establish or prove a damn thing unless they provide the facts to back them up.

The lesson to be learned from the Betamax case is that personal use is justifiable fair use. Remember, our justice system requires a preponderance of evidence in order to prove civil liablility. Simple and logical arguments concerning the perceived infringement will find in favor for or against insofar as the fair use test is concerned. When you apply the fair use test as the law requires, weighing all four parts equally, then the noninfringing nature of personal use becomes obvious. Only the first and last parts of the fair use test (purpose and effect) really need to be considered, because the other two (nature and amount) are pretty much "cut and dry." Even when these remaining two factors are considered, it can be shown that personal use meets these requirements. The establishment of personal use *does not constitute a preponderance of evidence* against a finding of fair use.

Alright, now that you hopefully understand the legal background, it's time to take the big leap. It's time to lay out the personal use defense for ROMs in easy-to-understand terms.

1. *Copyright law says that I have the legal right to archive computer software that I have legitimately obtained (17 USC 117). It says nothing whatsoever about preserving the original format of the software in question, nor does it require that I have to follow the vendor's intent in this matter.* I can make my backup on any media and in any format I want. So much for disk image ROMs. Just because it's a disk image doesn't mean it's not a backup
2. *Case law has demonstrated that an uncirculated, unauthorized derivative work based upon a program originally stored in ROM can be noninfringing (Nintendo v. Galoob).* As long as I paid for the original, and as long as that noninfringing derivative work remains with me, then no harm is done to the original vendor. So much for binary dump ROMs. They are not automatically infringing.
3. *Both copyright law and case law recognize a user's need to modify a given computer program in order that it may perform as designed irrespective of vendor desire (17 USC 117, Vault v. Quaid, Narrell v. Freeman). Such works are certainly derivative, but the courts have ruled them to be noninfringing.* So much for the vendor's infringement claims. Just because a derivative work is made doesn't automatically mean it's infringing.
4. *The U.S. Supreme Court has ruled that the personal use of copyrighted material, irrespective of how those copies are made, or in what format they are stored - even if it differs from the original, is noninfringing (Sony v. Universal).* Personal use is noncommercial use. The Supreme Court made the case that noncommerical use is noninfringing, so long as it does not cause harm to the vendor. Personal use does not cause harm to the vendor by its very nature - the vendor doesn't know about it and has no access to works derived under personal use. They can only speculate about its possibly infringing nature, and they are forbidden by the Supreme Court from engaging in speculation. If a ROM is made for the express purpose of personal use, then it qualifies for protection under the high court's Betamax decision.

Ergo,
ROMs made for personal use are noninfringing!
Q.E.D.

PRACTICAL APPLICATIONS

So now that we've established the legality of making ROMs from legitimately obtained originals under the concept of personal use, how do we go about actually doing it? Again, I quote from the Supreme Court's ruling.

> One may search the Copyright Act in vain for any sign that the elected representatives of the millions of people who watch television every day have made it unlawful to copy a program for later viewing at home, or have enacted a flat prohibition against the sale of machines that make such copying possible.

Well, guess what? The Copright Act has been amended to include the very "flat prohibition" that the court noted. It's not as simple as you might think to make your own ROMs due to the new law in town, the one that the Clinton Administration shoved down the users' throats with the gleeful cooperation of the vendors. What am I talking about? I'm talking about the **Digital Millenium Copyright Act** (DMCA), the 1998 amendment to the Copyright Act, and we have the Scribe to thank for informing us about an issue that has now become cold reality. It's the first part (17 USC 1201.a.1-3) that concerns us the most, and I reproduce it straight from the *EmuFAQ*.

> 1. No person shall circumvent a technological measure that effectively controls access to a work protected under this title... [rest omitted - ed].
> 2. No person shall manufacture, import, offer to the public, provide or otherwise traffic in any technology, product, service, device, component, or part thereof, that --
>
> > A) is primarily designed or produced for the purpose of circumventing a technological measure that effectively controls access to a work protected under this title.
> >
> > B) has only limited commercially significant purpose or use other than to circumvent a technological measure that effectively controls access to a work protected under this title.
> >
> > C) is marketed by that person or another acting in concert with that person, with that person's knowledge, for use in circumventing a technological measure that effectively controls access to a work protected under this title.
>
> 3. As used in this subsection,

> A) to "circumvent a technological measure" means to descramble a scrambled work, to decrypt an encrypted work, or otherwise to avoid, bypass, remove, deactivate, or impair a technological measure without the authority of the copyright owner.
>
> B) a technological measure "effectively controls access to a work" if the measure, in the ordinary course of its operation, requires the application of information, or a process or a treatment, with the authority of the copyright owner, to gain access to that work.

The law deals specifically with technologies that deal with audiovisual duplication and distribution, such as the old Macrovision scheme used to copy-protect commercial videotapes. Is it any wonder that the vendors have now started to apply this very same law to computer systems and their respective program bases?

Before Slick Willie signed the DMCA into law, it was perfectly legal for us users to go out and buy the same equipment that the developers were using - such as ROM analyzers, cart dumpers, and disk duplicators - and then burn away. That's not the case anymore. You set up a place that deals in this stuff, new or used, and rest assured that the vendors will have the SWAT teams breaking down the door in record time. Why? It's all because of the DCMA, that's why. We may have the legal right under personal use to make our own ROMs, but nobody has the legal right anymore to provide us with the required equipment. If you interfere with vendor control in any way, then you're breaking the law. Talk about a Catch-22! You know, the Supreme Court warned us about such laws, and I quote from the Betamax decision.

> Whatever the future percentage of legal versus illegal home-use recording might be, an injunction which seeks to deprive the public of the very tool or article of commerce capable of some noninfringing use would be an extremely harsh remedy, as well as one unprecedented in copyright law.

If I'm reading this right, what we have in the DMCA is an unjust law. It goes against the very nature of copyrights, and sets up a fundamental contradiction in United States intellectual property law. There's a whole discussion we can go into on that subject, but I'm not going there. I'm not a legal scholar, and that's not my purpose. My purpose is to provide you with a legal means of making your own ROMs, and that's what I'm about to do.

Never forget that you have a legal right to personal use of computer programs that you legitimately obtain for your own purposes. By the way, that also gives you the right to ignore the DMCA. **"WHAT?!"** I can hear the vendors screaming. "Yes," I quietly respond. The intent of the DMCA is to control potentially infringing technology of a "limited commercially significant purpose." That one word, *commercial*, is all the justification we need to bypass the DMCA. What we are engaged in is personal use, not commercial use. There's all kind of privacy rights arguments that come into play here, but the gist is that *if I'm engaged in personal use, which the courts have already deemed to be noninfringing, then the DMCA does not apply to my activities.* I can do whatever the hell I what, whenever the hell I want, and however the hell I want to do it because I'm engaged in personal use, not commercial use. The DMCA doesn't affect me in the slightest.

So how do we make our ROMs? We can't go out and buy a dumper or copier. The DMCA specifically forbids people from distributing that kind of stuff. There's nothing in there that says I can't obtain the plans for such devices and make them myself in order to enable my own personal use. That's the key - *personal use*. Personal use is noninfringing fair use, according to the Supreme Court, and would you like to know what the DMCA has to say about that? I quote directly from the DMCA (17 USC 1201.c.1):

> Nothing in this section shall affect rights, remedies, limitations, or defenses to copyright infringement, ***including fair use***, under this title [emphasis added - ed].

Guess what? *The DMCA does not apply to personal use!* I can go to the library and look up the data I need, or I can get it from friend, or I can just as easily download it from the Internet, and I'm not infringing on any vendor's precious rights. I can build all the homebrew ROM analyzers, cart dumpers, and disk duplicators that I want all day long, so long as I limit my operations strictly to personal use. You know, if anybody's liable under the DMCA, it's the people who distribute such stuff - not the recipients. That means if I so much as give one device away, or handoff my plans to someone else, then I've voided my right to the personal use defense and stand naked under the shadow of the DMCA. But, hey - I'm not going to do that, am I? Of course not. I'm not that stupid. If you're going to arrest somebody, then go arrest my local librarian. She has shelves full of books on computers and electronics that you might find violate the DMCA. Circumvention indeed!

Is it possible to build your own homebrew devices? Certainly. The information is out there in lots of places, just waiting to be tapped by you. For example, homebrew hardware hacking is what produced the IO-58, one of the best homemade cart dumpers ever invented. Of course, this device became the darling of many a software pirate group who couldn't afford hundreds of dollars for a commercial cart dumper, and the vendors will be quick with their "Software piracy!" mantra once they find somebody using one. I ask you, though, if you bought the IO-58 at your local computer store, would you be breaking the law? Absolutely, under the terms of the DMCA. If you downloaded the plans from the Internet, would you be breaking the law? No. If you built your own IO-58, would you be breaking the law? No. If you dumped your very own carts, ones that you bought and paid for yourself, for use on an emulator, would you be breaking the law? No. If you loaned your IO-58 to a friend, would you be breaking the law?

Yes. If you gave your friends copies of the IO-58 plans, not to mention a ROM or two in the process, would you be breaking the law? **Yes**. You voided your right to the personal use defense the minute distribution became involved, whether it was for the hardware or software side of things. Personal use ends where commercial use begins, and the last two instances are commercial use according to both the law and the courts. The same argument holds true for other such devices and applications, such as disk image creators, format transfer or conversion programs, media duplicators, and so on. If it weren't true, then no perosnal computer would have any kind of storage device at all. All in theory violate the DMCA's controlled access provisions, and to apply these to a product intended for personal use is as ludicrous as it sounds.

Now before I finish, let remind you of one thing. Personal use doesn't give you the right to download ROMs. You have to make them yourself. This is one area in which I agree with the Scribe, because neither statutory law nor case law permit anything else. I know a lot of you will hate this, as it's a lot more convienent to download the ROM you need instead of making it yourself, but there's just no other way around this fact. *The only legal way under the personal use defense to own a ROM for a program that you legitimately obtained is to make it yourself.* If you don't, then you're dealing in a distributed work, which means that it could be an infringing derivative copy under both 17 USC 106 and 17 USC 1201. The only other way around this is to get permission from the copyright holder. I'm sorry, but them's the facts. Just be glad that there ***<u>is</u>*** a legal way for you to come up with your own ROMs. It's a narrow and treacherous path, for sure, but it beats having no way at all.

As for using our newly made ROMs, what's the issue? None. That's what personal use is all about. These are <u>my</u> ROMs, made from <u>my</u> legitimately acquired games, using <u>my</u> equipment in the privacy of <u>my</u> home for <u>my</u> use only. I'm not going to sell them or give copies to friends. I'm defintely not going to upload them to the Internet. So I say again, what's the issue regarding the personal use of such ROMs? None! Period! That's the beauty of personal use! *The key word is **personal**. These ROMs are made for personal use, not commercial use*. No intent to distribute, no intent of profit - nothing. If I was going to do anything with them other than that, then I couldn't defend my actions under the concept of personal use. It's no longer personal. I'd have to use the concept of public use instead, and that's a road down which the courts have pretty much said we average users can't travel. We're not developers. We're not emucoders. We're **users** - acquiring, operating, and sometimes adapting these programs for our **personal use**. Again, quoting the Supreme Court,

> A challenge to a noncommercial use of a copyrighted work requires proof either that the particular use is harmful, or that if it should become widespread, it would adversely affect the potential market for the copyrighted work.

So tell me, does my personal use of ROMs derived from my legitimately purchased games qualify as "a particular use [that] is harmful" or one that "would adversely affect the pontental for that copyrighted work? The answer is ***NO***. I've paid for the game, made the ROM on my own equipment, and I'm not going to pirate it to everybody in sight. No infringement. None. *My personal use doesn't harm the vendor's potential market, and do you want to know why? Because the program's already bought and paid for, and all copies I make stay with me. That's why.*

You want to know something else? *Personal use of ROMs also fall under the ten rights of the software user in at least three different ways*. Here's what they are:

> **Running under emulation**: As we all know by now, emulation is perfectly legal. It is a fact that many emulators are literally works-in-progress and require ROM forms of programs in order to work. It is a fact that developers and emucoders are the only ones who currently have the legal right to make as many ROMs as they need of whatever programs they want for their various projects (*Sega v. Accolade*). Guess what? I'm not an emucoder. I'm not a software developer. Why can't I make a ROM from a game that I've already bought or obtained through other legitimate means? I have the original sitting right here, and I have the means to make the ROM. I'm not about to start up a software piracy operation. All I want to do is run my legitmately obtained program on an emulator, which is my legal right. As long as it's going to be for personal use, then I have the right to make a non-infringing ROM of that program (*Sony v. Universal*) and run it on my non-infringing emulator (*Sony v. Connectix*).
>
> **Operational adaption**: Copyright law says that I can adapt a program as need requires in order to get it to work properly on my system (17 USC 117.a). The vendors say I can't make ROMs of my own software because adaption is their exclusive right (17 USC 106.2, c.f *Nintendo v. Computer & Entertainment*). Tell me something, Mr. Vendor - how else am I going to be able to use my legitimately purchased software on an emulator, which is my legal right (*Sony v. Connectix*), if the emulator that I want to use won't work with anything except ROMs? Huh?! Tell me! You can't, can you? Well, I can. It's just like what Sony did when they invented the Betamax. They had to come up with a way to record a live broadcast of a television program so the average viewer could have a convienent means of watching it at a later time. The Supreme Court said that was perfectly legal - i.e., no infringement was involved (*Sony v. Universal*). That videotape recording was for personal use, not commercial use. TV shows are broadcast to anybody with the equipment to receive them, but you have to record them onto magnetic media, such as a videotape, in order to watch them later. Is it just me, or does that sound awfully like that

> dreaded "format change" that certain vendors are always bitching about as their "exclusive right?" *The Supreme Court said that the format change from live broadcast to videotape was <u>perfectly legal</u> for <u>noncommercial use</u>. Their ruling has been affirmed in other court cases involving personal use of other media, too, such as the Rio case (<u>RIAA v. Diamond</u>). It stands to reason that any format change involved in making my own ROMs from my legitimately obtained software for my own personal use is also perfectly legal.* It is a legitmate, noninfringing operational adaption in order to exercise my legal right as an <u>average user</u> to run that program under an emulator.
>
> **Freedom from vendor constraint**: One of the most curious rights on the list is being able to use a program "in a manner which the original vendor may not have intended" (*Narrell v. Freeman*). Let's face it - these guys never intended for their programs to be used in an emulated environment. It is a testament to the emucoders that many of these programs even work at all, given such conditions. While the vendors may have not intended to provide their programs to the emuscene, they can't stop the emuscene from enjoying them just as much as if emufans were using the intended platform instead. I have the legal right to operate my software irrespective of vendor desire, so long as I do not infringe upon their rights. My personal use of ROMs does not infringe upon their rights, because my actions are strictly personal. What I do in the privacy of my own home has no impact on their precious market share. I'm not affecting that market by my personal use, either directly or indirectly; therefore, my personal use does not infringe on their rights as a vendor.

"Hey now, wait just a doggone minute!" scream the offended vendors. "My EULA say you can't do thus-and-so with my product. That's a legal contract, bub, and you're bound to honor it!" May I humbly remind the vendors that any overreaching provision of a EULA, one that violates my rights as a user or potential developer, have no legal grounds (*Vault v. Quaid*, *Lasercomb v. Reynolds*) and may therefore be ignored? Such provisions are often put in for the sole purpose of raising a barrier to legitimate competition, like, say, using an emulator instead of the actual intended hardware. I'm sure you've seen phrases like, "You may not rent or lease ..." and "You may not reverse engineer ..." and "Copying is illegal and strictly prohibited ..." in many EULAs. Newsflash, vendors - all of these activities are specifically permitted by copyright law, so any EULA clause to the contrary isn't worth the effort you took to set it down. Federal case law has ascertained that overreaching EULA provisions erect an arficial barrier that is <u>against the spirit of copyright</u>, since the end result is an absolute monopoly over your product's market - one which copyright law does not and cannot confer. Federal case law trumps your precious EULA every time, sweetheart.

The same holds true for personal use. I have a legal right to personal use; therefore, I can do whatever the hell I want with your program, even if it violates that EULA, so long as I remain within the bounds of personal use. Your EULA says I can't change the format of your program, which may be required so I can run it on an emulator? Well, to hell with your EULA! I most certainly can and will, because I have the legal right to do so under personal use.

And if that's not enough, here's the haymaker. I remind you of what the Supreme Court had to say about the concept of personal use. I quote directly from the *Sony v. Universal* decision docket.

> **Any individual may reproduce a copyrighted work for a "fair use;" the copyright owner does not possess the exclusive right to such a use.**

Well, well, now, how 'bout that? I guess developers aren't the only ones who are entitled to the fair use defense after all. *Personal use qualifies as fair use, and is a protected act under the Betamax ruling!* Very interesting ... very interesting, indeed. I hope the Scribe is reading this, because he may end up having to rewrite all that stuff in the *EmuFAQ* about ROMs.

Of course, you know certain vendors are just going to hate this. They're going to say that I'm making it all up, that I'm crazy, that I'm wrong, or some such silly crap. Well, they'll be full of it, and you know it, because you know I'm right. Deep down, you know I'm right, even if some of **you** won't admit it.

IN CONCLUSION

With that, my diatribe is over. I've proven my point. *There exist a legitimate means for the average user to make their own ROMs from their own leigitmately obtained programs, and that is the protected freedom of personal use*.

The personal use defense for ROMs
It is not an infringement to make or use any kind of ROM with an emulator, provided that its source is of legitimate origin and such actions are strictly limited to personal use.

So stick that where the sun don't shine, Mr. Repressed Vendor! That is, assuming you can get it past the royal wedgie I've just given you. You can scream, "Software piracy!" all you want, but it doesn't make one damn bit of difference. Not one. As long as I'm making my ROMs for personal use, there's not a damn thing you can do about it. If you don't believe me, then go argue with the Supreme Court.

You want to know something? I own a Sony PlayStation. I own a couple of Nintendo consoles, the NES and Super NES. I own lots of original titles for these three consoles. Amost all emufans out there are in the same boat. Guess what? I can't take them with me everywhere I go. I can't enjoy your product experience whenever and wherever I want if I'm limited by your proprietary desires. Thanks to the Supreme Court, I am no longer shacked by your monopolistic constraints. Emulation has opened a whole new door whereby I can personally take pleasure in your products, and I'm not afraid anymore to assert my rights. *The societal gain of fair competition, in this case emulation, outweighs any potential loss that you might try to claim.* You want to keep bitching about it? Go right ahead! See if I care. In the meantime, though, I'm going to enjoy your products via my legal right to experience them through emulation. Good luck pressing your case! You're gonna need it.

REFLECTIONS

I will not dispute the claims of most vendors that they are often the victims of software piracy. I will not dispute the fact, proven by certain vendors, that so-called "console copiers," or "cartridge backup devices," or "high-speed disk duplicators" sold on the market as such are in fact devices that are frequently used and often implicitly advertised for the express purpose of copyright infringement. Most of the people who buy such devices do so with the intent to pirate. Such devices have no legitimate role in producing noninfringing archival copies, and such has proven itself in several notable court cases. It is an undisputable fact, however, that these and other legal devices that share similar design and operation can be used for entirely legitimate purposes. They can be used by developers for the express purposes of reverse engineering and program development, and that is a right protected by law. They can also be used to make noninfringing personal copies of legitimately obtained games, so long as these meet the criteria for personal use.

Hopefully, I have explained to one and all what personal use is all about, and all parties concerned now know that it can in no way infringe upon either the rights or marketability of a vendor's product. *ROMs made for personal use may be derivative, but they are not infringing.*

Anybody who is stupid enough to deal in illegitmate ROMs for any reason is a pirate and deserves to be treated as such. It should be obvious by now, though, that an emufan who follows the rules and generates a ROM for personal use neither breaks any law nor causes irrepairable harm to its original vendor. The U.S. Supreme Court long ago affirmed the principle of personal use as noninfringing (*Sony v. Universal*), and it works just as well for digital technology as it did for analog devices. The making and usage of a ROM can qualify for the Supreme Court's definition of personal use along the lines I've just finished discussing. This never has been nor never will be infringing, and you can take fact that all the way to the bank.

- Chuck Cochems

INTROSPECTION

Time and again, average everyday emufans has tried various ways of defending the ROMs that they want to use with their favorite emulators. All of their arguments have failed up to this point. While I must admit that I have certain problems with some of Chuck's assertions, my gut instinct tells me that he may be on to something here. It is a fact, as he asserts, that the U.S. Supreme Court has already laid the basic foundations for the theory of personal use in the now-infamous Betamax case (*Sony v. Universal*). It is a fact that the personal use defense has already been applied successfully in other venues, such as the audio recording industry (*RIAA v. Diamond*). Whether or not the courts will recognize its validity with regards to ROMs remains to be seen.

Let us assume for a moment that a personal use defense does exists for ROMs. If so, then what legal bounds would limit its potential use? Only one, insofar as I can see, but it is one that Chuck did not deal with in depth during his eloquent and impassioned arguments. I have discussed the matter with him, though, and it is one on which both he and I are in complete agreement. *Personal use cannot and will never provide justification for nor a viable legal defense against unauthorized distribution of protected works*. This fact is an inherent part of its makeup irrespective of the material involved, as the Supreme Court's dissenting opinion in the Betamax case made clear. Justice Blackmun notes,

> I am aware of no case in which the reproduction of a copyrighted work for the sole benefit of the user has been held to be fair use.

While this was written back in 1983-84, and such cases have indeed appeared for other forms of media, it still holds true today insofar as ROMs are concerned. You see, distribution implies that you are operating in a viable forum for commerce. According to current legal interpretation, a "viable forum for commerce" exists whenever you have two or more parties engaged in some form of transaction. It does not matter who (or what) the parties are, where each may be physically located, what medium of communication they employ to conduct their business, nor the material(s) involved in the transaction. If you have at least two individuals engaged in some kind of business, then a forum exists. The right of distribution for any copyrighted material is exclusive to the copyright owner (17 USC 106), and copyrights are the chief form of legal protection for computer programs such as ROMs. *If you engage in a forum, and the subject of your transaction is an unauthorized copy of a protected work, then you are dealing in an infringing derivative work and your right to the personal use defense is automatically voided.* Any actions you take in such a forum can only be conceivably defended by fair use, not personal use.

"But what about the Internet?" someone will inevitably ask. "I conduct personal business on there all the time." I've even seen this defense attempted by some of the ROM sites, e.g. "... these ROMs are made available for personal use only ...." While it may be true that you conduct personal business whenever you go online, let me remind you that the Internet was originally conceived as, set up to be, and operated as a public forum accessible by all for whatever business needed to be conducted. It has has remained the core operating principle of the Internet as it has grown and expanded beyond its original academic and government uses. This fact has been recognized by the courts (*Compuserve v. Patterson*) and affirmed by the federal government in a 1999 Federal Trade Commission ruling. People have been conducting personal business in public forums ever since the days of the open marketplace, but that doesn't change the nature of the forum. Face it - every time you sign onto the Internet, it is just like walking into the biggest shopping mall in this dimension, and practically all of the same rules and regulations apply. You are taking part in the most expansive forum for commerce and communication that humantiy has yet devised. What does that mean? *You can't deal with unauthorized commercial ROMs on the Internet in any form, shape, or fashion. No uploading, no downloading, no infringing links, no questionable email attachments, no "CD collection" offers or anything along those lines, no email linking or "burst" transfers, none of those curiously threaded multipart message board/news group/chat room postings (you know what I'm talking about) or the like, and definitely no "ROMz sitez." If you do, then you're engaging in software piracy, plain and simple. Personal use is not a valid legal excuse for defending Internet transactions of an infringing nature.*

It's also important to note the high court's reference to "librarying," which implied yet another possible voiding of the personal use defense. From the District Court all the way up to the Supreme Court, the plantiffs' contention remained that building up large archives of copyrighted works, even if it was for personal use, could be a potential infringement. I don't see a problem with this so long as you limit yourself to just one ROM for every original in your possession. That's perfectly reasonable, and any court rooted in both common sense and the law will understand such usage. That might even qualify for the backup clause (17 USC 117) if and when personal use of ROMs gains full legal sanction. It's when you make and maintain more than one copy of that ROM that the defense begins to fall away. The vendors could argue that you only need one ROM for that original, and having multiple copies isn't required for personal use. Also, having multiple copies of the same ROM might be shown to void usage and distribution rights (17 USC 106), which would fall under the pitfall of "potential harm" and thus become infringing activity.

So you see, personal use isn't the wide-ranging cure-all for ROM woes that many on the fringes of the emuscene might have hoped. Still, it's better than nothing. It's a start in the right direction, anyway. Better limited ROM rights for the average user than no rights at all. If personal use is the only legal justification available for users right now, limits and all, I say let's take the ball and run with it. It beats having no justification at all.

- Sam Pettus

REVIEW QUESTIONS

1. What is the single most common complaint by vendors against the production and usage of ROMs?

2. According to your understanding of copyright law, what is a derivative work? What do derivative works have to do with ROMs? Why?

3. Explain in your own words the "code is code" argument and how that applies to the creation of ROMs. How does this apply to disk image ROMs? To binary image ROMs?

4. What does the Digital Revolution have to do with changing the format of a program?

5. Is the right to make derivative works absolute? Be sure to justify your answer.

6. Why do developers have greater freedom to work with ROMs than do users?

7. Explain in your own words the implications of the Betamax case, in which noncommercial use of copyrighted materials qualifies as fair use. How does personal use qualify as a noncommercial use? How does the Betamax case protect personal use regardless of potential infringement?

8. Why is any argument posited by a vendor regarding perceived potential harm of an infringing use of its product without merit?

9. State in your own words the logical proof of the personal use theory. How is this justified under statutory and case law?

10. What single law stands in the way of a user obtaining the materials needed through commercial channels to produce ROMs for personal use? Why was this law put into place?

11. Can the abovementioned law ever be used to prosecute personal use? Why or why not?

12. In what ways does personal use qualify for acceptance under the legal rights of the software user?

13. Can a vendor EULA be used to inhibit or even prohibit personal use? Why or why not? Be sure to justify your answer.

14. Can you state the personal use defense for ROMs? What aspects of ROMs does it cover? Be sure to justify your answer.

15. Why is it that "ROMs made for personal use may be derivative, but they are not infringing?"

16. What potential pitfall would void a user's right to the personal use defense? What are some various ways in which this legal tripwire might be crossed by a user?

THOUGHTS TO PONDER

1. Is it legal for you to download a ROM, provided you own a legitimate copy of the program in question? Why or why not?

2. Is it legal for ROM sites to claim the enabling of personal use in order to justify their existence? Why or why not?

3. Is it legal for a program vendor to deny its purchasers the right to derive a ROM from that product? Why or why not?

4. Is it illegal for you to invite your friends over to enjoy a game that you intend to play under emulation? Why or why not?

5. Is it illegal for a vendor to prohibit the personal use of ROMs? Why or why not?

Sources

- 17 USC 101, et. seq., *The United States Copyright Act*.

- *Galoob v. Nintendo*, 1992.
- *Lasercomb v. Reynolds*, 1990.
- *Narrell v. Freeman*, 1989.
- *Sony v. Universal*, 1983.
- *Sony v. Connectix*, 2000.
- *Vault v. Quaid*, 1987.

- Cochems, Chuck. "Fair Use and ROMs." Internet article, Zophar's Doman.
- Cosby, Bill. *My Mother Confused Me. What Must I Do? What Must I Do?*
- *Microsoft Windows 98 Second Edition*, End User License Agreement.
- Pettus, Sam. "Establishing the Software Base." *Emulation: Right or Wrong?*
- "The Official Nintendo Emulation FAQ." Second edition.

Y2K Addendum: The Effects
Aftermath

> *Hi. It's me.*
>
> *I know you're out there. I know you're working as fast as you can to catch me. I thought I should call and let you know how things stand. I know you're real proud of this world you've built, the way it works, all the nice little rules and such, but I've got some bad news.*
>
> *I've decided to make a few changes .....*
>
> - Neo, *The Matrix* (original shooting script)

It was this past January that I sold my old Amiga.

You remember? The one of which I wrote so lovingly at the end of Module 3? The machine that I accidentally discovered in my basement, the same that helped revive my fond hacker memories from my past? The machine which, along with its millions of brethren, helped bring about the Golden Age of Emulation? Yes, that one. It is now residing in the home of a well mannered Japanese gentleman, who is undoubtedly putting it to more use than I would have these days. He wanted it, and I sold it to him. It's a "thing" that I have about old computer hardware. I'd rather have it put to good use than slowly molder away unused in a storage shed somewhere. Even old hardware has its uses, and that is a mantra that I have maintained ever since I cracked open the packing crate to my beloved Commodore 64 so many years ago.

I guess that my love of all things retrotech is what first attracted me to the emuscene in the first place. This was not mere nostalgia; indeed, it was something more. I <u>knew</u> these systems, inside and out. I <u>knew</u> the software, from both the user's and developer's perspectives. I <u>knew</u> how to use it, forwards and back. In an age were technology advances are measured by the second, it was refreshing to see something familiar, something constant, something tried and true, something proven and established amid the rapidly changing panoply of ever-improving computer technology. It may be old and in many cases obsolete, but that in no way diminished its value. I'm excited that so many people are rediscovering our shared digital past. There's a lot of good stuff back there, even if it isn't very pretty by today's' texture mapped and polymorphic standards. It was "good stuff" then, and it's still "good stuff" now, even after the passage of so many years.

Of course, my retrotech nostalgia also revived memories of my darker side, the days when I was an active participant in the hacking scene of the 1980 and early 1990s. Like today's space cowboys, we wanted and got the best in order to carry out our own anarchistic attacks against the digital establishment of our time. Back then, real hackers worshipped such vintage machines as the Commodore 64, and I was one of the lucky few who managed to scrape up the dough for the prized SX-64 portable version. Oh, the fun I used to have with that thing and my trusty copies of *Sgt. Pepper* and *Fast Hack'em*! While we're at it, let's not forget the warez. You wanted warez? I had 'em ... all of 'em ... anything you could ever possibly want for the C64, C128 (yes, I shelled out US$450 for the 128D), and my beloved, hacker-worn Amiga 2000. Those days are now but a distant memory, and you know by now how they eventually ended, but I still can't help but smile as I think back to that time. I was young, headstrong, irresponsible, and determined to conquer the world. You see, I had an advantage that most of the adults were sadly lacking. I knew computers, inside and out. I knew how you were supposed to use and maintain them, and I knew the other things you could do with them that the manuals didn't cover. Even after I was forced to go legit, it was a knowledge base that kept me employed for quite a while.

That's one of the biggest problems in getting too close to technology. Unless you can somehow manage the near-impossible effort of keeping up with all of the latest changes, it will eventually pass you by. I saw it coming, I knew that I couldn't stop it, and so I prepared for that inevitable day when I couldn't play the part of computer freebooter anymore. I sharpened my skills in other areas, so that when that day finally came and I was left behind, I was not entirely without resources. It was still a bit of a shock, and it took me about a year before I finally got back up into speed, but I can't count myself a true *hacker* anymore. I'm just a *user* now, same as the majority of you reading this. Some of you are in the same boat as I, technology having passed you by, whereas others are just getting started. Regardless of who you are, though, we share one thing in common - a love for the machine and a respectful reverence for the potential that yet remains to be unlocked. We stand on the shoulders of giants, and others will rise above us to raise the digital pyramid to new and lofty heights. It doesn't hurt to climb down once in a while and see where we've been. It helps us to see where next we should go.

There remains a tendency among the vendors and the mainstream press to slam emulation almost as hard as they do computer hacking. I frankly have never considered computer hacking to be the disrespectful profession for which it is often portrayed. If I recall my high school English classes correctly, a "hack" was a writer who got the knock-off or odd jobs that nobody else wanted. They had to be fast in order to meet deadline, yet their work had to be good enough to pass muster with the public. So it is today with the "hacker" - the digital reincarnation of the "hack" writer. There's nothing dishonorable about being a hacker. It's just that the nature of the profession requires you to think outside the box and beyond the accepted parameters almost all the time, and it takes considerable skill to be a good hacker.

Yes, it is true that there are a few renegades who give hacking a bad name, but not all hackers are evil teenagers or unprincipled college students whose sole purpose in life is to somehow break into the CIA's mainframe and uncover the truth behind the Roswell Incident. Hackers have their own ethics, their own morals (yes, they do, believe it or not), and their own code of honor. The emuscene owes hacking a debt of gratitude, because it would never have existed had not a now-forgotten IBM customer back in the early 1960s hacked his mainframe in order to run programs it wasn't designed to execute. That story is what supposedly gave Larry Moss the idea for emulation, and hacking has remained an integral part of the emuscene even since. In fact, the commercial emulators are by far outnumbered by hacker products, and the latter are often just as good as or in some cases even better than what the big boys produce. The same is true for the rest of the computer industry - almost all real innovations can be traced back to creative hacks cooked up on the spot or in a flash of inspiration by an inspired hacker (or a closet one). You want to know something else? Even the big boys, such as Steve Jobs and Bill Gates, have hacking in their blood. It's part of the industry. It's not going away. There has been hacking in the past, hacking continues in the present, and hacking will go on in the future. I just wish that for once we could get the mainstream to understand that, instead of focusing on the few bad apples that bounce out of the barrel.

I hope that in some future time, when computer historians look back and are forced to come to grips with the emuscene of our day, that they will give proper credit to the emucoders in our midst. These überhackers have managed to recreate complete vintage systems entirely in software (in many cases), allowing old programs to run on new hardware. That's no small feat, believe me. You emucoders know what I'm talking about. Those of you who have worked with them know just how much time and effort goes into their projects. Those of you who patronize your favorite emulators appreciate their efforts. It's such a shame that there remain so many who would rather snipe away at what they blithely label as "illegal emulation" rather than become involved and get up close and personal with the emuscene. I have had the good fortune to do that just these past two years, and I am enriched thereby. The emuscene has reinvigorated my love for the machine, has restored my delight in experience the joy of artful programming, and allowed me to once again relive the highlights of computer days gone by. It's our shared heritage that the emuscene is saving. No matter that certain vendors can't see beyond the buckles of their money belts. What they think doesn't matter in the long run. What they provide will outlast them, so long as there remains some means to preserve and present on tomorrow's technology.

You can't save live code in cold print. You can't capture the essence of the program presentation in static media. No, you have to recreate the entire experience, both the good and the bad, as best as programming skill and available resources will permit, so that current and future generations will better appreciate just how far we've come. There's really only one way to do that that remains constant, giving the rapid changes in computer technology, and that is the art of emulation. Time will pass, memories will fade, and machines will crumble away, but the code will survive. Long after we are all dead and gone, the code will still be there for some future computer user to rediscover and say, "Wow! They did *that* back *then?* Neat!" Think I'm crazy? Perhaps ... but just look at how today's users are rediscovering the vintage treasures of our past. The code will survive, for thanks to the emuscene, it will never die.

I find it rather ironic that the company who opened the Pandora's box on noninfringing technology back in the early 1980s is the same that has brought about the legalization of emulation. Sony, the inventors of the Betamax, who fought long and hard to justify its existence against a resolute foe, also fought equally hard some two decades later in order to stop the same thing from happening to them. They failed, in part to their own successful actions in the past, and emulation is now legal as a result. *Emulation is legal*. It still gives me chills whenever I say that. I knew it instinctively, as did we all, back in the dark time when I set out to prove that it was actually true. You know, in a way, it's sad that the one company who had originally championed a form of noninfringing technology didn't want the same yardstick applied once their turn came. "Not in my backyard," as the old saw goes. Well, so much for their efforts. They were doomed from the start, as I knew they would be, but rest assured that I impatiently waited, perched on the edge of my seat, along with the rest of you. Would I be proven right, or would the unrestrained use of emulation technology be forever hobbled by monopolistic vendor desire? It was fortunate that things turned out the way that they did, and a royal screw-up by one of our greatest foes has turned out to be our unlikely salvation. We have Sony to thank for the legalization of emulation, just as we have Sony to thank for the legalization of audiovisual broadcast recording. Who would have thought it?

You want to hear something almost equally strange? How about Nintendo taking part in the emuscene? Sega, yes - they've always been there for us - but Nintendo? Who would have thought that, too, back during the time when Nintendo was going after anybody and everybody who even so much as dared to discuss emulation of their systems? Yet here we are in the year 2000, and RandNet is getting ready to set up a commercial NES ROM site for paying Nintendo customers. I can only imagine how much it took for them to even consider the possibility in the first place, so my hat's off to them. I know it's not as big a step as some of you would like to see, but give them credit anyway. It took a lot for them to go even that far. Let's hope that this tentative step leads to even bolder moves in the near future, but they won't do it unless there's something in it for them. Only your patronage and support will encourage them to continue, otherwise they'll pop right back into their shell faster than you can say *Nintendo Entertainment System*. If that means going to their website and paying them for their ROMs, then so be it. Oh, and while you grit your teeth and mumble about having to pay for computer games that are long past their heyday, remember that it wasn't very long ago that Nintendo was busting one ROM site after another. It pays to have at least one legal Nintendo ROM site, even if you have to pay Nintendo to use it.

Speaking of which, as this is my final address to the emuscene, I would like to thank Sega for its unspoken support of my activities all this time. You were always there for me, even during those days in which I questioned whether or not I could continue at all, and it was largely due to people who work with you, who know and love your systems and your software, that I was able to continue. You never objected to my efforts, even though they involved unlicensed emulation, and you have given me a straight answer to every question I have ever asked. You have never been afraid of the promise that emulation holds, and you have been willing to stick out you neck time and again in unceasing quest of its potential. Such effort deserves to be rewarded, and I have done so in more than words alone. In the past two years, I have purchased over US$2000 worth of Sega products, and it is a trend that will no doubt continue into the foreseeable future. It's like the old adage says, "You scratch my back, and I'll scratch yours." Some of that money might have gone to your competitors, but they have as yet to show any evidence of your class. It's not big by your multimillion dollar standards, and is not as much as I would have liked to give, but I have done what I can. I wish I could do more, but other tasks draw me away. Perhaps, once all is said and done, I might be able to come back and pick up where I have left off. I had seven years of catch-up to do last time. Let's hope it's not nearly as much should next time come.

I now find myself viewing the emuscene from a distance, no longer playing the front line role that I once so thoroughly enjoyed. My part in the great emulation debate is now over. I set out to prove that emulation was a legitimate practice, and the U.S. Court of Appeals has affirmed my assertions as legal fact. Having fought and triumphed, it is now time for me to move on.

I have no doubts that I may pop back in now and again, watching the Second Golden Age of Emulation unfold, but not for long and never in the prominent role that I once played. This is not my time. I was part of the First Golden Age, from its glorious inception back in those heady Amiga days up to the recent dark times. I saw the paradigm shift taking place. I saw my fellow emufans suffering from a lack of knowledge and direction at the hands of those who would take full advantage of the situation. I drew a line in the sand and said, "Enough." I helped my fellows in their time of trial, and in so doing helped usher in the Second Golden Age. I guess that makes me a midwife or attendant of sorts, if anything, but I'm not alone. As I have stated time and again, mine was but one small part among many. It just happened to be more vocal and therefore more visible than others who deserve more recognition than I.

Yes, you are now witnessing the unfolding of the **Second Golden Age of Emulation**. The first began with *A-Max* and the nextgen machines of the 1990s - the Amiga, the Macintosh, the Atari ST. The second age comes as personal computers undergo yet another convulsive revolution in capabilities - the G4 PowerMacs, the *Windows 2000* and Linux boxes, the 128-bit videogame consoles, and so on. We have passed yet another threshold, a new virtual explosion in computer technology is upon us, and the trend will continue as surely as the one terabyte hard drive will soon inhabit our desktops. New technology. It is at the heart of each successive paradigm shift that catapults the computer industry to new and lofty heights every five to ten years or so, and it is carrying the emuscene right along with it. New emulators will be required for the new systems. New projects will be undertaken at emulating the old systems on the new ones. New efforts will be made to preserve old code and make it work with these new systems. It's going to be a wonderful time, just like it was back in the early 1990s. It's the dawn of a new age for the emuscene - the Second Golden Age of Emulation. My only regret is that I will not be here to enjoy the full fruits of our shared labors.

You see, change is at the heart of humanity. One of my all-time favorite quotes on the subject was uttered as an admonition by Duke Leto Atredies to his son Paul in the movie version of Frank Herbert's *Dune*. "Without change, something sleeps inside us and seldom awakens. The sleeper must awaken." My time in the emuscene has been long, eventful, and ultimately enjoyable, but the sleeper has awakened for another cause. New opportunities await, new fields of battle beckon, and others in desperate straits cry out for someone to aid them in their fight for justice. I know I can help them. I've done it before, and I'll do it again. As long as the cause is just and the means exist, I will always resist those who would repress others in any way. The Goliaths of this world will never last so long as there remains at least one David to take his sling, pick up five small stones, and walk unflinchingly towards his seemingly invincible foe. My sling is in hand, and my pouch is full. It's time for the next challenge. The road is unfamiliar, the foe untested, but my resolve is no less strong that it was here. Be that as it may, though, I leave the future of the emuscene in good hands. I know that my place will not remain empty if need requires. I was but one David among many, and others will surely rise to combat the challenge should the emuscene ever suffer such travails again. Your foes may be different, your battlefield digital instead of physical, but your determination and desire will be no less strong than those in the physical world. You will someday triumph, despite everything that can and will be thrown at you. You are survivors. You will survive.

Be seeing you.

The old battlefield which we know only too well lies quiet again, save for the sounds of movement on its outer edges. Sitting on the horizon just within the line of sight is the final prize - the last fortress of monopolistic might. Facing its walls, as far as the eye can see, is a vast host of diverse peoples, united in common cause against a common foe. From tender youths to grizzled veterans, from neophyte newbies to hardened hackers, they stand ready for battle. There can be no doubt about the outcome now, save for an act of divine providence or utter stupidity. They have the momentum now. It is their fight to lose ... or win.

The leaders of the emuscene survey their assembled might as they plan their final stroke. They are not alone, for three other armies stand ready to join the fray. There are the legions of the black army, the freebooters and mercenaries, who are only their to take advantage of the situation, but the emuscene does not scoff their aid. There is a host of renegade vendors, far larger and stronger than even the emuscene had initially hoped, who have seen the truth of the cause and joined their fates to it. Finally, there is a crack battalion of federal troops waiting in reserve, who entered the affray unlooked-for in the emuscene's greatest hour of need and helped it win the field. They will not join the fight unless absolutely necessary, but their aid is no less welcome. The forces of free emulation are thus far stronger than even their dwindling foes had ever dreamed. It is an uneven fight against a desperate and well-entrenched foe, but one from which the emuscene does not shirk. Even so, it will take courage and skill to manage such disparate hosts and unite them in common cause. The emuscene now has the means to win the emulation war. Victory can be theirs for the taking, provided that they have the will and resolve to seize every opportunity, whatever the cost may be.

Suddenly there rings forth a shout from the front of the host, and a lone warrior stands forth before the fray. A grizzled veteran, he is but one of the many who have led the charge in the emulation war, and his is the honor of launching this, the final assault in a long and vicious campaign. With one hand he slide-cocks his shotgun, and then slaps the other into place on its grip. He surveys his fellows as he grins with delight, chomping down on the half-burned cigar sticking out of one side of his mouth, then utters the final call to arms.

"**Let's rock!**"

> *I have, myself, full confidence that if all do their duty, if nothing is neglected, and the best arrangements are made, as they are being made, we shall prove ourselves once again able to defend our Island home, to ride out the storms of war, and to outlive the menace of tyranny, if necessary for years, if necessary alone. ... [W]e shall not flag nor fail. We shall go on to the end, we shall fight in France, we shall fight on the seas and the oceans, we shall fight with growing confidence and growing strength in the air, we shall defend our Island, whatever the cost may be, we shall fight on the beaches, we shall fight on the landing grounds, we shall fight in the fields and in the streets, we shall fight in the hills; we shall never surrender, and even if, which I do not for a moment believe, this Island or a large part of it were subjugated and starving, then our Empire beyond the seas, armed and guarded by the New World, with all its power and might, steps forth to the rescue and the liberation of the old.*
>
> - Sir Winston Churchill
> "Dunkirk," *Blood, Sweat, and Tears*

Acknowledgements

The author would like to thank the following individuals, groups, organizations, firms, companies, and corporations for their aid in providing additional information and insights concerning their knowledge of emulation and their relationship with the emulation community. This document is solely the work of the author, and should not be interpreted as the official stance on emulation as held by those named below. Their opinions and policies do not necessarily agree with those of the author; however, they have been kind enough to help the author in his research, and he remains grateful for their assistance.

Individual contributors: Mike Blair, Chuck Cochems, Omar Cornut, Brad Levicoff, Franklin Nixon, Christian Schiller, Steve Snake, Joseph Sullivan, Brad Taylor, Harry Tuttle, Andy Wolan

The emulation community: Connectix, Inc., Dave's Video Game Classics, Emulation HQ, Zophar's Domain

The vendor community: The Interactive Digital Software Association (IDSA), Nintendo of America, Inc.

Special thanks: David Lloyd, for granting me permission to reprint selected issues of his world famous cartoon strip *OverClocked* in the *EmuFAQ*.

Laws and statues (United States):

- The Computer Fraud and Abuse Act (18 USC 1030).
- The Copyright Act of 1976 (17 USC 101, et. seq., amended 1980, 1988, 1998).
- The Digital Millenium Copyright Act of 1998 (amends U.S. Copyright Act).
- The Electronic Communications Privacy Act (18 USC 2510, et. seq.).
- The Landham Act (15 USC 1051, et. seq.).
- The Patent Act (35 USC 1, et. seq.).
- The Semiconductor Chip Protection Act (17 USC 901, et. seq.).
- The Stored Wire and Electronic Communications Transactional Records Act (18 USC 2701, et. seq.).
- The Intellectual Property Rights Act (37 CFR)

Laws and statues (other):

- The Berne Convention for the Protection of Literary and Artistic Works. 828 UNTS 221 (1886, amended 1928, 1971).

Court cases:

- *Allen-Myland, Inc. v. International Business Machines, Inc (IBM).* 33 F.3d 194 (3rd Cir., 1994).
- *American Geophysical Union v. Texaco, Inc*. 37 F.3d 881, 894 (2nd Cir., 1994).
- *Apple Computer, Inc. v. Franklin Computer Corporation.* 714 F.2d 1240 (3rd Cir., 1983), 464 U.S. 1033 (1984).
- *Apple Computer, Inc. v. Microsoft Corp.* No. 93-16883 (9th Cir., 1994).
- *Asahi Metal Industry Co. v. Superior Court*. 480 U.S. 102, 112 (1987).
- *Atari Games Corp. v. Nintendo of America, Inc.* 975 F.2d 832 (1992), No. 91-1293 (9th. Cir., 1992).
- *Atari, Inc. v. JS&A Group, Inc.* 597 F.Supp. 5 (N.D. Ill, 1983).
- *Bally Total Fitness Holding Corp. v. Andrew S. Faber*. 29 F.Supp. 2d 1161 (C.D. Cal, 1998).
- *CompuServe, Inc. v. Richard S. Patterson and Flashpoint Development*. (9th Cir., 1996).
- *Diamond v. Diehr.* 450 U.S. 175 (1981).
- *Digital Communications Assoc. Inc. v. Softklone Distrib. Corp.* 449 F. Supp 449 (N.D. Ga., 1987).
- *Franklin Mint, Inc., v. Franklin Mint, Ltd.* 331 F.Supp 827, 830 (E.D. Pa., 1971).
- *Futuredontics, Inc. v. Applied Anagramics, Inc., et. al.* No. 97-6991 (C.D. Cal, 1998), also No. 97-56711 (9th. Cir., 1998).
- *Harper & Row Publishers, Inc. v. Nation Enterprises.* 471 U.S. 539, 592 (1985).
- *In re Compton's New Media.* USPTO No. 5241671, Control No. 90/003270 (1994).
- *Intel Corp. v. Advanced Micro Designs (AMD)*. 12 F.3d 908 (9th Cir., 1993)
- *Intermatic, Inc. v. Toeppen*. Magistrate's Report (N.D. Ill, 1996).
- *International Shoe Co. v. Washington*. 326 U.S. 310, 316 (1945).
- *Kewanee Oil v. Bicron Corp.* 416 U.S. 470 (1974).
- *Lasercomb America v. Reynolds*. 911 F.2d 970 (4th Cir., 1990).
- *Lewis Galoob Toys, Inc. v. Nintendo of America, Inc.* 964 F.2d 965 (9th Cir, 1992).
- *Lotus Dev. Corp. v. Borland Int'l, Inc.* 49 F.3d 807 (1st Cir., 1995), aff'd per curiam U.S., No. 94-2003 (1996).
- *M. Kramer Mfg. v. Andrews.* 783 F.2d 421 (4th Cir., 1986)
- *MAI Systems Corp. v. Peak Computer, Inc.* 991 F.2D 511, 518 (9th Cir., 1993).
- *Maritz, Inc. v. Cybergold, Inc.* 947 F.Supp 1328 (D. Mo., 1996).
- *MiTeK Holdings, Inc. v. ArcE Engineering Co.* No. 94-5262 (11th Cir., 1996).
- *Mirage Editions, Inc. v. Albuquerque A.R.T. Co.* 856 F.2d 1341, 1343 (9th Cir., 1988), 489 U.S. 1018 (1989).

- *Narrell v. Freeman*. 872 F.2d 907, 913 (9th Cir., 1989).
- *New West Corp. v. NYM Co. of California, Inc.* 595 F.2d 1194, 1201 (9th Cir., 1979).
- *NEC Corp. v. Intel Corp.* 10 U.S.P.Q.2d (BNA) 177 (N.D. Cal., 1989)
- *Nintendo of America, Inc. v. Computer & Entertainment, Inc.* No. 96-0187 (W.D. Cal., 1996).
- *Nintendo of America, Inc. v. Dragon Pacific Intern*. 40 F.3d 1007 (9th Cir., 1994).
- *Nintendo of America, Inc. v. NTDEC*. 51 F.3d 281 (9th Cir., 1995).
- *Panavision International v. Toeppen*. 938 F.Supp. 616 (C.D. Cal, 1996).
- *Penwalt v. Durand-Weyland, Inc.* 833 F.2d 931 *en banc*, 4 USPQ 2d 1737 (Fed. Cir., 1987).
- *Playboy Enterprises, Inc. v. George Frena*. 839 F.Supp. 1552 (M.D. Fla., 1993).
- *Playboy Enterprises, Inc. v. Terri Welles*. 7 F.Supp. 2d 1098 (S.D. Cal., 1998).
- *Playboy Enterprises, Inc. v. Universal Tel-A-Talk, Inc., et. al.* No. 96-6961 (E.D. Pa., 1998).
- *Princeton University Press v. Michigan Document Services.* 1996 U.S. App LEXIS 7474 (6th Cir., 1996).
- *ProCD, Inc. v. Ziedenberg*. 86 F.3d 1447 (7th Cir., 1996).
- *Recording Industry Association of America (RIAA) v. Diamond Multimedia Systems*. No. 98-5627 (9th Cir., 1999)
- *Religious Technology Center v. Netcom*. No. 96-20091 (N.D. Cal., 1995).
- *Sega Enterprises, Ltd. v. Accolade, Inc*. 977 F. 2d. 1510, 1517 (9th Cir., 1992).
- *Sega Enterprises, Ltd. v. MAPHIA*. 857 F.Supp. 679, 686 (N.D. Cal., 1994).
- *Shetland Times Ltd v. Dr. Jonathan Wills and Zetnews Ltd.* Court of Session, Edinburgh, Scotland (Lord Hamilton, 1996).
- *Sony Corp. of America v. Connectix, Inc.* No 99-15852 (9th Cir, 2000).
- *Sony Corp. of America v. Universal City Studios, Inc*. 464 U.S. 417 (1983).
- *Stac Electronics v. Microsoft Corp.* No. 93-0413 (C.D. Cal, 1994).
- *Subafilms, Ltd. v. MGM-Pathe Communications.* 24 F.3d 1088, 1093 (9th Cir., 1994).
- *Tandy Corp. v. Personal Micro Computers, Inc.* 524 F. Supp. 171 (N.D. Cal., 1981).
- *United States v. David LaMacchia.* 871 F.Supp. 535 (D. Mass., 1994).
- *United States v. Richard D. Kenadek*. No. 94-10221 (D. Mass., 1994).
- *Vault Corporation vs. Quaid Software Ltd.* 655 F. Supp. 750 (E.D. La., 1987).
- *Williams & Wilkins v. United States.* 487 F.2d 1345 (1973), 420 U.S. 376 (1975).

Other court cases (dockets not found)

- *Atari, Inc. v. Coleco*. (1983).
- *International Business Machines, Inc. (IBM) v. Compaq Computer Corp.* (1982).
- *Nintendo of America, Inc. v. Blockbuster Entertainment.* (1987)
- *Nintendo of America, Inc. v. Bung Enterprises and Carl Industries*. (C.D. Cal., 1997).
- *Nintendo of America, Inc. v. Prima Publishing*. (W.D. Wash., 1998).
- *Nintendo of America, Inc. v. Tengen*. (1989).
- *Sony Corporation v. Bleem LLC*. (N.D. Cal, 1999).

Other authorities:

- Abbot, Mark. "Copyright for an Electronic Age." *PowerPoint* slideshow. Cobalt Qube, 1997.
- Aiken, James H. "The Jurisdiction of Trademark and Copyright Infringement on the Internet." Mercer University School of Law, 1997.
- Ajami, Amer. "Sony Injunction against Bleem! Denied." *GameSpot News*, 24 August 1999.
- *The American Heritage Dictionary*. Second College Edition. Boston: Houghton Mifflin Company, 1985.
- "The Amiga Personal Computer." Promo handout. Commodore Business Machines, 1985.
- Asher, Mark. "Bleem Claims Another Victory." *C|NET Gamecenter*, 26 August 1999.
- Atkin, Denny. "Amiga's 10th Anniversary Year Scrapbook." *AmigaWorld*, January 1995.
- Baker, Chris. "Playing on the Past: Why Old Video Games Refuse to Die." *The Austin Chronicle* (online edition), 12 June 1999.
- Band, Jonathan. "The Digital Millenium Copyright Act." Morrison & Foerster, 1998.
- Borland, John. "Anti-Piracy Forces Target Arcade Classics." *TechWeb*, 22 April 1998.
- Brochstein, Martin. "Why Sega didn't want its day in court." *Computer Retail Week*, 24 October 1994.
- Broiles, Greg. "Teen Offers Way to Crack Blocking Software." Internet message board post, 1 May 1999.
- Burtenshaw, Jonathan (nee Harry Tuttle). "The end of an era...for now...." ClassicGaming.com, 1998.
- ----------. "The Fate of the Emulation Scene." ClassicGaming.com, 1998.
- "Case Summaries of Computer Copyright Cases." Schwegman, Lundbert, Woesnner, & Kluth P.A., 1998.
- Chapman, Gary. "Innovation." *The Los Angeles Times*, 20 May 1996.
- Cherry, Andrew. "Game Emulation." Updates.com, 1999.
- Christie, Ian. "Nintendo Sues Publisher over James Bond Maps." *Wired*, 20 November 1997.

- Cohen, Julie E. "Reverse Engineering and the Rise of Electronic Vigilantism: Intellectual Property Implications of 'Lock-Out' Programs." Ithaca: Cornell Law School Legal Information Institute, 1995.
- "Compton's awarded patent on multimedia." *Seybold Report on Desktop Publishing* (Volume 8, Number 4), 1996.
- "Computer Menu Not Protected by Copyright." Schwegman, Lundbert, Woesnner, & Kluth P.A., 1998.
- Crouse, Robert. "Are Shrink-wrap Prohibitions Against Reverse Engineering and Disassembly Preempted by Federal Copyright Law?" MBSS, 1997.
- Davidson, Stephen J. "Developing the Evidence for the Trial of a Trade Secret or Reverse Engineering Case." Minneapolis: Leonard, Street, and Deinard, 1990.
- Dethblade. "Emulation." *PSX Underworld*, February 1999.
- "The Digital Millenium Copyright Act." Endless Visions, 1999.
- Donaldson, David H. (ed.) "BBS Sysop Liability For Copyright Infringement: Let the Owner Beware!" and "Copyright Law: 'Fair Use of Copyrighted Materials by Schools & Non-Profit Organizations." *Legal Bytes* (Volume 2, Number 1). Austin: George, Donaldson, and Ford, 1994.
- *The Dot-Eaters - Classic Video Game History*. Emulation Unlimited, 1998.
- D®þ£þþ£®. "Is the Game Boy Color a criminal's best friend?" 1999.
- Elliot, Stephen. "Apple Computer, Inc. v. Franklin." *Copyright Law: Definitions*. Nolo Press, 1998
- "Emulate This!" Ziff-Davis Publishing, 1999.
- "Federal Appeals Court Issues Wide-Ranging Copyright Decision in *Vault v. Quaid.*" Lucash, Gesmer, and Updegrove, 1998.
- "On the Spot - Fixo Interviews Jeff Gerstmann." *EmuGaming.com*, 18 August 1999.
- Flinn, Patrick J. "Horseshoes, Hand Grenades, and Equivalents: Where 'Almost' Counts." Internet document: Alston and Bird LLP, 1997
- Gribble, Steve, with assistance from Fox, Armando, Katz, Randy, and Wang, Amy. "Intellectual Property - CS39C Spring 1998." Berkeley: University of California at Berkeley, 1998.
- Guillot, Gregory H. "How are 'marks' protected?" *All About Trademarks*. Gregory H. Guillot, Chartered, 1999.
- "Hasbro Interactive Releases Rights to the Atari Jaguar Videogame Platform." Atari Gaming Headquarters, 1999.
- Hassett, Rob. "Impact of Apple vs. Franklin decision." *Computer World*, 5 December 1983.
- Herel, Suzanne. "Bleem Blasts Onto Emulator Scene." *C|NET Gamecenter*, 24 March 1999.
- Hodor, Ken. "Adapting Methods to Protect Intellectual Property from Pirates." Silicon Strategies, 1998.
- "HR 2281 - Myths and Facts." CEMA Government & Legal Affairs Department, 1998.

- Institute of Electrical and Electronics Engineers. "Reverse Engineering: USAB Position Statement." IEEE-USA, 1993.
- *Instruction Booklet: STAR WARS Rogue Squadron*. Fremont: LucasArts Entertainment, 1998.
- Interactive Digital Software Association. "The IDSA FAQ." IDSA, 1998.
- Jacobson, Russ. "*A-Max* - It's In and It Works!" *The Status Register*, June 1989.
- Johnston, Chris. "Press Start - PlayStation Emulator Arrives in Stores." *Electronic Gaming Monthly* (San Francisco: Ziff-Davis Publishing, September 1999), p. 36.
- Jung, Robert. "The Atari Timeline." 1999.
- Jussawalla, Meheroo, and Oniki, Hajime. "Changing Technologies and Intellectual Property: The Economic Perspective." *Impacts of Information Technology on Management and Socioeconimics*. Tokyo: Maruzen Planet Co. Ltd., 1995, pp. 131-144.
- Kent, Steven. "Battling Video Game Pirates." *Technology*, MSNBC, 4 August 1999.
- ----------; Horwitz, Jer; and Fielder, Joe. "The History of Videogames." VideoGameSpot, 1998.
- Lawrence, Richard. "The Legalites of Emulators." 1999.
- Levicoff, Brad. "The Emulation Dream." Zophar's Domain, 1997
- Levine, Don, with Michlig, John. *G.I.JOE - The Story Behind the Legend*. San Francisco, Chronicle Books, 1996.
- Liberman, Michael. "Overreaching Provisions in Software License Agreements." *Richmond Journal of Law and Technology*. Richmond: Richmond University, 1995.
- Liedholm, Marcus and Matthias. "The History of Nintendo." Nintendo Land, 1998.
- ----------. "Inventory Management (Nintendo's Sales Tactics)." Nintendo Land, 1998.
- Lieske, Stephen C. "Copyright and the Internet." Schwegman, Lundbert, Woesnner, & Kluth P.A., 1998.
- Luskin, Frederic Michael. "COMPTON Patent Reversed (details)." Stanford University, 1994.
- Lutzker, Arnold P., Lutzker, Susan J., and Settlemyer, Carl H. III. "The Digital Millenium Copyright Act: Executive Summary." Lutzker & Lutzker LLP, 1998.
- Macavinta, Courtney. "House Clears Copyright Act." *C|NET News,* August 1998.
- Maclachlan, Malcom. "Game Makers Challenge Emulation Precedent." *Techweb News*, 3 February 1999.
- Mattison, Wendy J. "Connectix Wins 1st Round in Sony Suit." *MacWEEK*. Mac Publishing LLC, 1999.
- McDonald, T. Liam. "You Will Be Emulated." *MaximumPC*, September 1999.

- McGrath, William L., on behalf of Perkins Cole LLC. "Re: WWW.FREEROMS.COM: Infringement of Nintendo copyrights and trademarks." Copy of a hand-delivered letter by Nintendo of America, Inc. to SiteAmerica, Inc. and the webmaster of FreeROMs.com, declaring intent to file suit against the named parties regarding proveable intellectual property infringement (scanned and posted by FreeROMs on their site shortly before it went off-line on 5 May 1999, pp 1-2 of 3 pages).
- Mishra, Rohan. "Reverse Engineering in Japan and the Global Trend Towards Interoperability." *Murdoch University Electronic Journal of Law*, Vol. 4 No. 2, June 1997.
- Morgan, Charles. "Copyright and Private Use in the Digital Age: Concerns Regarding the Commodification and Control of Private Interaction." Canada: McGill University, 1997.
- Moryath and Zach. "Nintendo FAQ Refutal v0.02." Emulation HQ, 1999.
- "Nasty Nintendo email." Emulators Unlimited, 1998.
- Niccolai, James. "Clinton Signs Digital Copyright Bill." Press release. San Francisco: IDG News Service, 28 October 1998.
- "Nintendo Loses Legal Challenge to Prima." *TechMail*, 6 January 1998.
- Nintendo of America. "The Official Nintendo Emulation FAQ." Nintendo Corporation, 1999.
- "Nintendo Settles With Taiwanese Counterfeiter; Defendants Sentenced to Jail Terms." *Inventor News*, 15 October 1998.
- Obser, Jeffrey. "Nintendo Pirating Device Still for Sale." *Wired News*, 26 August 1999.
- Ocampo, Jason. "Bleem Restraining Order." *C|NET Gamecenter*, 12 April 1999.
- Palumbo, Paul. "Nintendo's Dolphin: Proposed DVD system would have major cost-of-goods-sold implications." *GameWEEK Magazine*, 25 August 1999.
- Polsson, Ken. *Chronology of Events in the History of Microcomputers*. 1999.
- Pugh, Emerson W., Johnson, Lyle R., and Palmer, John H. "Solving the Conversion Problem." *IBM's 360 and Early 370 Systems*. Cambridge: MIT Press, 1991.
- Pugh, Emerson W. "The Road to System/360." *Memories That Shaped an Industry*. Cambridge: MIT Press, 1984.
- "Put a PlayStation in your PC." *PC Gamer*, October 1999, pp. 50, 60.
- "Quartermann - Video Game Gossip and Speculation." *Electronic Gaming Monthly* (San Francisco: Ziff-Davis Publishing, September 1999), p. 50.
- "Recent Copyright Developments (Copyright Misuse and the Allen-Myland Case)." Lucash, Gesmer, and Updegrove, 1998.
- "Reverse Engineering Software to Achieve Compatability Is Fair Use." Boston: Lucash, Gesmer, and Updegrove, 1993.
- Russell, Michael. "Emulation... The Final Word (For Now)." Dn'A Enterprises, 1999.

- Saltzman, Mark. "Emulation Nation." *C|NET GameCenter*, 19 March 1999.
- ----------. "Sony Sues Dave." *C|NET GameCenter*, 2 April 1999.
- Samson, Martin. *The Phillips Nizer Internet Law Library*. Phillips, Nizer, Benjamin, Krim, and Balloon LLP, 1997.
- Seeman, Roderick. "Nintendo ordered to stop price fixing." *The Japan Lawletter*, May 1993.
- "Sega vs. Nintendo: War Games." *Reputation Management*. EMMI, 1996.
- Schaefer, Gary. "Sony sues to block PlayStation emulation software." *The Detroit News*, 1 February 1999.
- Shamanix. "The Good Old Days...." Zophar's Domain, 1998.
- ----------. "Myths and Misconceptions: The Buck Stops Here." Zophar's Domain, 1998.
- Smith, J.C. "Emulators vs. Consoles." Above and Beyond Technologies, 1999.
- Smith, Tony. "Dissecting Sony's Game." *MacWEEK*. Ziff-Davis Publishing, 1999.
- ----------. "N64 emulator vanishes after lawsuit threat." *The Register,* 3 February 1999.
- ----------. "Nintendo legal rumblings provoke boycott call." *The Register,* 8 February 1999.
- ----------. "Nintendo N64 emulator programmer quits." *The Register*, 5 March 1999.
- ----------. "Nintendo pursues emulator sites." *The Register,* 24 May 1999.
- ----------. "Nintendo WILL sue UltraHLE developers." *The Register*, 12 February 1999.
- ----------. "Sony to fight PlayStation emulator in court next month." *The Register*, 20 August 1999.
- Solomon, Christopher. "Nintendo 64 debut empties store shelves." *The Seattle Times*, 2 October 1996.
- Spregner, Polly. "Sony Sues Over PlayStation Clone." *Wired*, 29 January 1999.
- "Status of Connectix Virtual Game Station." *Mac Gamer's Ledge*, 20 August 1999.
- Syrowik, David R., and Cole, Roland J. "The Challenge of Software-Related Patents." *Michigan State Bar Journal*. Lansing: Michigan State Bar Association, 1994.
- The SYS2064 news archive - August 1997 to present.
- taichou. "ASCII begins to strike." Zophar's Domain, 1999.
- Tanner, Eileen. "Nintendo Sues Hong Kong Manufacturer and U.S. Distributor Over Game Copier Device." *TechMail* (online edition), November 1997.
- Tauber, Daniel A., and Kienan, Brenda. "Legal FAQs for Webmasters: Who Owns What." *webreview.com*. Songline Studios, 1998.

- Templeton, Brad. "10 Big Myths About Copyright Explained." Brad Templeton, 1995.
- Tysver, Daniel A. (of Beck and Tysver). "Important Copyright Decisions." BitLaw - A Resource on Technology Law, 1998.
- Redbox. "Video Doctor 64 Modification - The Truth About the Modification." Supplex, 1998.
- "Videogame Emulators." Ziff-Davis Publishing, 1999.
- Watson, Brett. "Philosophies of Free Software and Intellectual Property." Nutters.org, 1999.
- The Weasel. "The legal lowdown on game emulators." *CNN.com*, 2 June 1998
- Wen, Howard. "New life for old games." *Salon*, 23 June 1999.
- ----------. "Why emulators make video-game makers quake." *Salon*, 4 June 1999.
- "When Copying Is OK: The 'Fair Use' Rule." Nolo Press, 1998.
- Winebarger, Lynn. "Intellectual Property, Reverse Engineering, and Free Software." Bloomington: University of Indiana, 1999.
- Woods, Bob. "Nintendo Sues Over Alleged Game Copying Device." *Computer Currents Magazine*, 24 November 1997.
- Wook. "Segregating the Scene from the Community." Emulation Camp, 1999.
- xix. "Q&A with Dave of Dave's Video Classics emulation site." geeknews, 1999.

Appendix B
Important Court Decisions Concerning Emulation

The following are selected summaries and accompanying commentary of many of the major examples of case law that are referenced in the *EmuFAQ.* I have done what I can to explain their significance; however, I would appreciate someone with real legal credentials, such as a law student or professional attorney, to aid in revising them and filling in the gaps. The eventual goal is to include a summary for every court case listed in the *EmuFAQ* credits.

Allen-Myland, Inc. v. International Business Machines, Inc. 33 F.3d 194 (3rd Cir., 1994).

> Allen-Myland was a third-party service provider for IBM 3090 mainframe systems. They had the practice of routinely archiving 3090 system microcode on tape, and would then later combine selected portions of their vast tape archive in order to recreate copies of this code for their customers - all of whom were legitimate licensees for the programs involved. IBM sued for copyright infringement of its proprietary 3090 system software, and the matter eventually wound up in the U.S. Court of Appeals. The high court ruled that it was not legal for Allen-Myland to build its archive in the first place; also, the "adaptation" clause of the U.S. Copyright Act did not give Allen-Myland the right to create "rainbow copies" (the court's term) from multiple sources. It also ruled that Allen-Myland's intent was not purely archival; the tape archive had been created with the intent of recreating copies of that software at will for other systems and customers; therefore, their actions did not qualify for protection under the "archival" clause of the U.S. Copyright Act.
>
> *This case defined the concept of **rainbow code**, determining that such creates an infringing derivative work under the current form of U.S. copyright law.*

American Geophysical Union v. Texaco, Inc. 37 F.3d 881, 894 (2nd Cir., 1994).

> Texaco, the well-known international petroleum firm, was charged by the American Geophysical Union for copyright infringement by illegally duplicating copies of AGU-licensed periodicals for its large research staff. The courts ruled in favor of the AGU, noting that Texaco's operations were commercial in nature and therefore did not qualify the corporations' research libraries for the "safe harbor"

exception. As such, its duplication of the copied journals was not permissible under "fair use" and thus violated copyright law.

This case helped define the limits of the "safe harbor" exception with regards to unauthorized "fair use" of copyrighted materials by anyone making the claim that they are a qualifying archival institution. As a general rule, any organization that is hosted or directly sponsored by a commercial firm does not qualify for "safe harbor" protection. Educational institutions and non-profit organizations, such as libraries and public archiving services, are qualified for "safe harbor" exemption. Their dealings with commercial firms are at best indirect, usually via donations. The implications with regards to Internet distribution of copyrighted material are obvious.

Apple Computer, Inc. v. Franklin Computer Corp. 714 F.2d 1240 (3rd Cir., 1983), 464 U.S. 1033 (1984)

In 1982, Franklin Computer produced a working clone of the Apple II computer called the Franklin ACE 100 that incorporated Apple microcode into its system ROMs. Apple promptly took Franklin to court, claiming copyright infringement of its proprietary programs. The First District Court ruled that object code (machine language) is not readable by the average person and is therefore not copyrightable. This decision was subsequently reversed by the U.S. Court of Appeals, who ruled that operating system object code, whether it is embedded in ROM (i.e. a system BIOS) or stored on some other form oft media, is covered by copyright law. An appeal to the U.S. Supreme Court was later rejected out-of-hand. Franklin was forced stop production of its Apple II clone and pay hefty damages to Apple, and it eventually withdrew from the personal computer industry altogether.

One of the results of this case was the rewriting of the U.S. Copyright Act in order to be more explicit with regards to all forms of computer software - to wit, copyright protection is now afforded to both the source and object code forms of a program. This is key with regards to the legality of both making and using an unauthorized binary dump of computer code stored in some form of permanent media, such as a ROM chip, since that is a form of object code and is therefore protected by vendor copyright. This case has grave implications for the practice of BIOS dumping and the unauthorized distribution of a BIOS image.

Apple Computer, Inc. v. Microsoft Corporation. No. 93-16883 (9th Cir., 1994).

Sometime around 1983, Microsoft founder Bill Gates noted his admiration for what would eventually become MacOS and warned Apple executives that they should take out copyrights on its chief features. Apple decided not to do so, so it should have come as no surprise when Microsoft eventually released its own GUI environment, *Microsoft Windows*, in 1985. It went largely unnoticed until 1988, when it adopted a layout that bore a striking resemblance to MacOS. While it didn't quite duplicate the "look-and-feel" of MacOS, Apple felt it sufficiently close enough to file suit anyway the following year. They also took advantage of the opportunity to sue non-contender Hewlett-Packard over its *NewWave* GUI for the same reasons. In 1989, Xerox weighed in on the dispute and sued Apple, claiming that MacOS was little more than a direct ripoff of its *Xerox Star* GUI environment originally designed for its business workstations. The court threw out the Xerox lawsuit, but took note of their contentions with regards to the continuing dispute between Microsoft and Apple. MacOS was eventually broken down into more than 180 distinct elements in order to test the validity of Apple's claims. Notice was also taken of Apple's lack of legal protections for its GUI, as well as Xerox's argument about Apple's "borrowing" from *Xerox Star*. The court eventually ruled in 1992 that only specific screen elements of a computer program are copyrightable - not the entire on-screen presentation. As a result, Apple was only granted copyright protection to a very small number of GUI elements. Apple appealed the decision, but the appeal was subsequently denied. *Microsoft Windows* had now been legitimzed as a vaild GUI for personal computer systems.

This case is important to the emulation industry with regards to the successful duplication of original vendor product functions and user interfaces, as well as the overall "look-and-feel" of an similar product in comparison with the original. Ideas cannot copyrighted, therefore similar-looking products do not necessarily violate copyright law.

Apple Computer, Inc. v. Readysoft, Inc. (aka **the *A-Max* precedent**, 1989)

<u>While this court case never happened to the best of my knowledge</u>, the potential threat of it and the subseqent inaction on Apple's part to bring it about set a imprtant precedent for emulation technology. *A-Max* was the brainchild of Simon Douglas and the first Macintosh emulator for any platform. It allowed any Amiga computer to emulate a Mac via a combination of software and special hardware. Apple was unable to mount a case against *A-Max* due to the fact that it had been designed so as to not interfere with Apple's rights with regards to Mac technology and the Mac market. Therefore, by default, Readysoft was allowed to continue marketing and improving *A-Max* at will.

*The **A-Max precedent** is important to anyone who may be required to defend the development and subsequent release of an emulator. It established the legality of an unauthorized emulator for a proprietary system so long as it does not infringe upon the original vendor's intellectual property rights.*

Asahi Metal Industry v. Superior Court. 480 U.S. 102, 112 (1987).

Asahi Metai was a well-known Japanese industrial firm whose products were shipped worldwide. A number of these eventually made their way to California, where Asahi was subsequently sued under California law for violating certain portions of commerce law. Asahi appealed their conviction, and the case eventually wound up before the U.S. Supreme Court, which upheld the decision in a 5-4 ruling.

*The importance of **Asahi** is twofold, First, the Internet is considered be a viable "stream of commerce" as defined by the Supreme Court's ruling. Second, the case laid down the five point "Asahi test" for determining fairness of jurisdiction in such a dispute. The Asahi test is frequently used in conjuction with the three-fold test of jurisdicitional locale laid down in the **International Shoe** case in order to determine proper venue and scope of prosecution in cross-border disputes. See also **International Shoe v. Washington, 1945**.*

Atari, Inc. v. Coleco. (docket number unknown, 1983)

In 1982, Coleco produced a special adapter for its ColecoVision videogame console that would allow the system to both use and play videogame cartridges designed for the Atari 2600 videogame console. It was able to do so because Coleco's product was actually a scaled-down Japanese personal computer (built along the lines of what would later be known as the MSX standard), and its more powerful hardware could easily support the simpler Atari games. Atari filed suit in federal court charging Coleco with patent violation, but subsequently lost the case. Atari 's argument was that any product derived from a unique one already on the market was by default the intellectual property of the original producer. The courts determined that Atari's videogame technology was too generic to be considered unique, and the concept of a videogame was also generic and therefor could not be used to establish the uniqueness of Atari's product. The resultant setback for Atari caused it to seek an out-of-court settlement, and the end result was that Coleco continued to manufacture its Atari videogame adapter for the ColecoVision.

This was the very first court case involving cross-platform support for videogame systems, and as such is significant with regards to videogame emulation due to the unique origins of the ColecoVision console. It had almost been forgotten by everyone until the dispute over the legality of PlayStation emulation was first raised in 1998.

Atari, Inc. v. JS&A Group, Inc. 597 F.Supp. 5 (N.D. IL, 1983)

JS&A Group were the vendors of a cart dumper called the "PROM Blaster" for Atari 2600 videogames. It was a device designed to allow its operators to dump the object code from within the ROMs of an Atari 2600 videogame cartridge. Atari sued JS&A for contributory copyright infringement of its proprietary videogame software. JS&A argued that archival copying was permitted under copyright law (17 USC 117), so the purpose of a cart dumper constituted "substantial non-infringing use." The court found that a computer program embedded within a piece of hardware, such as the ROMs used within a typical videogame cartridge, cannot be reprogrammed or erased. As such, it was contained within a form of storage media designed to permanently preserve the program. JS&A's contention that cart dumps protected the actual cartridge against possible physical harm (and thus the program embedded inside) might also be applied to other forms of physical media, such as phonograph records and books. Since copyright law did not allow for this practice in regard to these and other such forms of physical media (photocopying a book, physically duplicating a phonograph record), they likewise did not apply to the practice of archiving computer programs embedded within a piece of hardware. The archival exception for computer software did not apply to programs stored within permanent storage media (in this case a videogame cartridge) because these forms of media are not subject to the sort of risks that the archival clause was designed to guard against. As a result, the court ruled that dumping a videogame cartridge for archival purposes is not covered by the archival clause of copyright law.

This case would prove to be the most important one in subsequent cases regarding the desire to duplicate program code contained within some form of permanent storage media in general, and the twin practices of videogame cartridge and arcade ROM dumping in particular. It has been modified in at least one respect over the years; see also ***Sega v. Accolade, 1992****.*

Atari Games Corp. v. Nintendo of America, Inc. No. 91-1293 (9th Cir., 1992)

Rival videogame company Atari sued Nintendo in U.S. Federal District Court under the claim that the Japanese videogame giant was trying to establish an unlawful monopoly within the American videogame industry through such techniques as price fixing, unethical business practices, and the shutting-out of independent software developers due to the use of proprietary hardware (i.e. Nintendo's antipiracy systems). The initial case dragged through the courts until 1993, when Nintendo was cleared of all charges - they were no longer the dominant force in the videogame market due to Sega's rise. It should be noted that both the New York State Attorney's Office and America Video Entertainment (AVE) took

Nintendo to court in 1991 in separate cases accusing Nintendo of an illegal monopoly on the videogame market and won, at which time Nintendo was forced to pay considerable sums in damages. On a parallel track, however, events were unfolding along different and indisputably infringing lines. Not willing to tolerate what they perceived as an unlawful market by Nintendo on the videogame market, Atari decided not to license NES ports of its software titles and to find a means of emulating the NES antipiracy system. They obtained a copy of the 10NES lockout program souce code via fraudlent means, and used it to creat their new *Rabbit* lockout emulation technology. Once they determined that *Rabbit* would work, they began releasing unlicensed NES titles in 1989 under the newly created Tengen banner. As Atari was still a licensed developer for Nintendo, the latter promptly filed suit for both copyright and patent infringement (the NES antipiracy system was protected by company patent). They also charged Atari with violation of the RICO Act, claiming Tengen had been created for the express purpose of fronting unlicensed videogames. Nintendo began systematically taking other measures against Atari and its supporters in the videogame market by various means - actively discouraging its vendors from carrying Tengen software, "freezing out" Atari and/or limiting their presence in industry trade shows, and so on. Atari won the initial round by gaining a preliminary injunction against Nintendo's behavior; however, that injuction was subsequently thrown out on appeal. The case wound up with the U.S. Court of Appeals, which eventually ruled that although it was not unlawful for Atari to produce and sell unlicensed videogames, it was within Nintendo's rights to discourage the purchase of said product by its customer base. They further ruled that Atari had violated Nintendo's copyrights on the 10NES lockout codes used in the NES antipiracy system because of the way in which had developed their *Rabbit* emulator; it "incorporated elements of the 10NES program unnecessary for the chip's performance." The following year, the court upheld Nintendo's claim of patent protection, and the two parties eventually settled out of court.

The practical upshot of this case was that Nintendo gained the court's blessing to continue its practice of intimidation against anyone who chose to market or distribute unlicensed videogames for Nintendo's proprietary home console technology, illustrating Nintendo's monopolistic and often overreaching attitude with regards to any forms of competition for its videogame products. Nintendo has since chosen to interpret this as giving them the full legal right to intimidate anyone who produces or distributes what they believe to be a product that infringes upon their intellectual property rights - including emulators, by the way, since emulation was one of the points brought up in the case. Interestingly enough, this case is often quoted as one of the prima facie examples of the practice of decompilation of computer code as acceptable under the concept of "fair use," since it is often the only means whereby the underlying ideas and concepts behind the code can be derived. The other is ***Sega v. Accolade, 1992****.*

Bally Total Fitness Holding Corp. v. Andrew S. Faber. 29 F.Supp. 2d 1161 (C.D. Cal, 1998).

> Andrew Faber was a young man who didn't particularly care for the way Bally Total Fitness ran its health clubs. In fact, he was so critical of them that he set up an Internet site titled "Bally Sucks" to air his opinion. The site included the word "bally" in its URL, and opened with the Bally logo followed by the word "sucks." It should have come as no surprise that Bally chose to sue him for trademark infringement in an effort to shut down his site. The courts sided with Faber, first dismssing Bally's claim and later dismissing the case altogether. In rendering its decision, the U.S. District Court ruled that users of Bally Sucks would have no illusions that they were using an anti-Bally Internet site. On the countrary, users would most likely assume that the site was in no way authorized by Bally. Since such action did not infringe upon Bally's reputation, neither did Faber infringe upon Bally's trademarks and Bally's claim was therefore without grounds. It also noted that Bally Sucks was not a commercial operation, and that it would be improper to deny Faber the right to use the word "bally" in his URL because this would effectively isolate him from the bulk of the Internet. Bally had no right to oppose criticism to its business practices in the matter that Faber was conducting, which was perfectly legal and the reason why the court eventually threw out the case.
>
> *This is as close as will you get to a "free speech" justification for the use of a registered trademark on the Internet. Open criticism is perfectly acceptable so long as it is not of a competitive nature, as was the case with Andrew Faber and the Bally Sucks website.*

CompuServe, Inc. v. Richard S. Patterson and Flashpoint Development. (9th Cir., 1996).

> CompuServe, the telecommunications networking giant, was taken to court by shareware developer Richard Patterson for illegally duplicating an Internet browser that he had developed. Patterson won in federal district court, but the decision was overturned on appeal. The Sixth Circuit Court of Appeals noted that the reach of the Internet meant that existing legal precedent for due process had to be reinterpreted in light of its capabilities. It used the three-point jurisdiction test laid down by the *International Shoe* and *Asahi* cases to redetermine culpability. First, it found that Patterson's contacts with CompuServe had indeed established a viable forum for commerce, or to quote the court, "a relationship intended to be ongoing in nature." Second, it found that "CompuServe's claims against [Patterson] arise out of his activities in Ohio if we are to find the exercise of jurisdiction proper" - which meant that the state of

Ohio was the proper venue of prosecution. Third, it found that, based on the *Asahi* precedent, that "there is a substantial enough connection between Patterson and Ohio to make it reasonable for an Ohio court to assert personal jurisdiction over him." In accordance with its findings, the Court of Appeals reversed the lower court's decision and remanded the case for re-hearing.

What this means for any party on the Internet is that one can be prosecuted for any activity that infringes upon any form of intellectual property protection, regardless of where the respective parties are located or how the infringement took place. This case is considered one of the landmark cases with regards to resolving jurisdictional disputes arising from infringmenet via the Internet; see also ***International Shoe v. Washington, 1945****, and* ***Asahi Metal Industry v. Superior Court, 1996****.*

Diamond v. Diehr. 450 U.S. 175 (1981)

In one of the true landmark cases for the computer software industry that made it all the way to the U.S. Supreme Court, the U.S. Patent and Trademark Office had granted a patent to a firm under the terms of the U.S. Patent Act (35 USC) who had developed a computer program to aid in a rubber refining process. The Court of Customs and Patents reversed the patent when it was contested, but the decision was subsequently overturned on appeal by the Supreme Court. Writing for the majority, Chief Justice William Rehnquist noted the following:

> "... when a claim containing a mathematical formula implements or applies that formula in a structure or process which, when considered as a whole, is performing a function which the patent laws were designed to protect (e.g., transforming or reducing an article to a different state or thing), then the claims satisifes the requirements of Section 101 [of the Patent Act]."

This case opened the door for software patents.

Digital Communications Associates, Inc. v. Softklone Distributing Corp.
449 F.Supp 449 (N.D. Ga., 1987)

One of the early "look-and-feel" lawsuits arose in a dispute over Softklone's *Mirror* telecommunications software which Digital felt violated its copyrights on its own proprietary *Crosstalk XVI* software. The courts noted that copyright infringement did not occur with regards to either the source or object code, but agreed with Digital that the near-identical screen displays constituted such an infringemnt. The court noted that copyrights could be applied to "the placement, arrangement, and design" of words on a computer display so long as such a display served to explain a given program function and was not a mere represntation of program function or status. Softklone's *Mirror* too closely followed the setup of Digital's *Crosstalk* in this regard, and the court ruled in Digital's favor. Digital did not succeed in stopping Softklone, however, becasue the latter redesigned its user interface so as not to duplicate Digital's and later re-released *Mirror* using the new display format.

This case had strong implications three years later in the ***Lotus v. Borland*** *dispute, and should be noted by those who plan on making unauthorized translation patches of foreign computer software.*

Franklin Mint, Co. v. Franklin Mint, Ltd. 331 F.Supp 827, 830 (E.D. Pa., 1971)

The Franklin Mint Company is a well-known firm located in Pennsylvania specalizing in cast and stamped metal collectables, such as coins, sculpture, and so on. When another company took a similar name and began to produce its own products, the original Franklin Mint sued in the Pennsylvania court system, accusing its competitor of irreperable harm to its business practices. The courts eventually ruled in favor of the original Franklin Mint.

This case is important to consider concerning the unauthorized distribution of active commercial "ROMs." Its importance is noted in ***Sega v. MAPHIA****, which states, "Each illegal copy of a Sega game which Defendants distribute deprives Sega of revenue. Moreover, distribution of altered, inferior copies of Sega games and of confidential, pre-release unperfected games subjects Sega to damage to its business and reputation." This case provides incontrovertible evidence that copyright protection covers any and all forms of a commercially released product, including any pre-release copies that may be in existence.*

Futuredontics, Inc., v. Applied Anagramics, Inc., et. al. No. 97-6991 (C.D. Cal, 1998), also No. 97-56711 (9th. Cir., 1998).

As part of its Internet site, Applied Anagramics created a link to the Futuredontics site within one of the frames of its web pages. This allowed copyrighted material to appear on the Applied Anagramics site without authorization. Furthermore, since the balance of the page was filled with Applied Anagramics content, an unsuspecting user might erronously believe that the material in question belonged to them. As a result, Futuredontics filed suit in U.S. District Court charging Applied Anagramics with copyright infringement. Initially, the court refused a preliminary injuction due to lack of evidence, but later granted it after Futuredontics was able to better establish its claim. The case is still winding its way through the legal system on appeal as of this date; however, the district court eventually found in favor of Futuredontics and granted the injunction. In its ruling, the court based its final decision on the possibly that the link may have allowed Applied Anagramics to create an unauthorized derivative work on its own web site.

*This case extended the principle of intellectual property infringement on the Internet to include "framed" content copied or displayed from another Internet site. It is a natural extension of the Shetland precedent; see also **Shetland Times v. Wills and Zetnews, 1996**.*

Harper & Row Publishers, Inc. v. Nation Enterprises. 471 U.S. 539, 592 (1985)

This particular case involved the newsmagazine *The Nation*, which had secured a preview copy of former U.S. President Gerald Ford's soon-to-be-published memoirs. They were particularly interested in his thoughts regarding the pardon of his predecessor, Richard Nixon, concerning the Watergate affair; so they published a small excerpt from the unreleased manuscript. The book's publisher, Harper & Row, promptly filed suit. The case went all the way to the U.S. Supreme Court, which ruled in favor of the manuscript's originally contracted publisher. In the court's opinion, the marketability of the manuscript had been compromised by the premature publication of the except, as it was the most important part of the book at the time with regards to potential sales, and that pressing concerns with regard to the public interest were insufficient to justify said usage. As such, the "fair use" claim of *The Nation* were without grounds.

This case established the marketability principle of fair use, which is point four in the fair use test (17 USC 107). Noted legal scholar Raymond Nimmer has called it the most important of all four points of the test (Nimmer on Coypright), even though the law states that all four points are to be given equal weight when deciding a fair use dispute.

In re Compton's New Media. USPTO No. 5241671, Control No. 90/003270 (1994).

While this is not an actual legal case per se, this action filed with the U.S. Patent and Trademark Office deserves mention. *Compton's Interactive Encyclopedia* was the first widely acknowledged CD-ROM based encyclopedia to hit the software market. It used a patented search engine that made for quick and easy research. The only problem with this was that the original software patent for that search engine was so broadly worded that it effectly blocked anybody else from developing or using any other kind of search engine, as well as making owners of previously existing search engines liable for back royalties. Compton made it clear that it expected and would be seeking concurrent and back royalties from anybody employing any kind of search engine within their software. The patent was successfully challenged by the IMA, and subsequently thrown out by the Patent Office as too expansive in its claims.

This case is important to remember whenever an original vendor or licensee begins making broadly worded claims against an unlicensed developer. It also has certain implications with regards to the patenting of computer software.

Intel Corporation v. Advanced Micro Designs (AMD). 12 F.3d 908 (9th Cir., 1993)

Intel found it difficult to keep up with production of its x86 family of CPUs during the heady days of sales for the early 8-bit personal computers. To offset the shortage, it subcontracted several firms to produce compatable CPUs, among them AMD. In 1982, the same year that Intel unveiled the16-bit 80286 CPU to the public, it entered into a special agreement with AMD that permitted AMD to build 100% working clones of Intel CPUs by using real Intel microcode within them. When it came time for Intel to roll out the 80386 (also known as the i386) in 1985, it decided that it had sufficient production capability and didn't need AMD's help anymore. AMD countered by releasing its own 80386 clone CPU, dubbed the Am386, which was a combination of Intel's internal 80286 microcode and certain reverse-engineered features of the 80386. Intel promptly took AMD to court claiming both copyright and trademark infringement, and the legal battle raged throughout the lifetime of the 80386 and well into that of its successor, the 80486 (which AMD also cloned in a similar fashion). The courts eventually ruled in 1994 that Intel's prior licensing agreement with Intel gave AMD the legal right to use 80286 microcode inside its clone processors, but it would have to discontinue the practice in future products. Also notable was the ruling that Intel had no legal claim to the number "386" as a trademark, since it was too generic. The following year, Intel and AMD settled their few remaining differences out of court, with AMD paying Intel for the right to continue using proprietary Intel microcode in its Am386 and Am486 product line.

This case is the one most frequently quoted with regards to the use of proprietary vendor code within an independent product, and is also important with regards to reverse-engineering and valid trademarking issues.

International Business Machines v. Compaq Computer Corp. (docket number unknown, 1982)

IBM sued the fledgling Compaq for using what they contended to be an counterfeit copy of the IBM PC BIOS in their Compaq Portable PC and therefore an unlawful duplication of copyrighted computer code. Compaq had foreseen the legal challenge and produced a reverse-engineered product known as the Phoenix BIOS (after its developer, Phoenix Technologies) that did not contain any proprietary IBM microcode. They also omitted duplicating the internal ROM BASIC (i.e. "Cassette BASIC") of the IBM BIOS within the Phoenix BIOS for a number of reasons (avoidance of counterfeiting charges, for starters), thus rendering it not entirely compatible with the then-existing base of IBM PC software. As a result, the courts eventually ruled in favor of Compaq and allowed the continued manufacture and use of the Phoenix BIOS. This case opened the floodgates on the PC compatible market, since the rest of the computer was comprised of standard hardware. An interesting sidebar to note is that Microsoft's Bill Gates quickly moved to include a software version of BASIC in his MS-DOS package for the new PC clones, thereby providing full compatability with the rest of that day's IBM PC software base. ROM BASIC is now largely forgotten, except for those of us who remember that time and the unlucky few who have stumbled across its absence. This is the reason behind the situation that occurs whenever anyone tries to boot up a PC compatible without an installed operating system and finds the message "NO ROM BASIC - SYSTEM HALTED" displayed on their computer monitor.

*The **IBM v. Compaq** case is considered the key case with regards to the legality of reverse-engineering computer code contained within firmware. It is also important from an emulation perspective, considering the fact that Compaq deliberately crippled the Phoenix BIOS in order to maintain legality and the market responded by working around the omission in a legal fashion. The case is important for emulator developers for two reasons. First, it established the bulletproof legality of "clean room" reverse engineered procedures. Second, it established that reverse-engineering should be limited to only those functions necessary to duplicate necessary functions of the original product, and this concept would be strengthened in **Nintendo v. Atari, 1992**.*

Intermatic v. Toeppen Magistrate's Report (N.D. Ill, 1996) and **Panavision v. Toeppen.** 938 F.Supp. 616 (C.D. Cal, 1996).

The defendant in this case was a resident of the state of Illinois and ran a web site featuring various aerial views of major cities within the state. In order to attract attention to his site, he registered a number of domain names that contained well-known trademarked terms within the photographic industry. Two of these were Intermatic and Panavision, and their owners subsequently sued for trademark infringement. In the case of the Intermatic violation, a federal district court ruled that Toeppen's use of the Intermatic trademark violated both state and federal anti-dilution statutes, since Intermatic did not want and had no intention of associating itself with Toeppen's activities, and so it found in favor of Intermatic. In the case of the Panavision violation, the plantiff filed suit in California under both state and federal anti-dilution laws, and a federal district court found that Toeppen's use of a registered trademark within an Internet domain name was sufficient grounds to establish trademark infringement. Even though Toeppen was not conducting any business on his site, the cross-border nature of the Internet permitted California residents to access his site, and the court ruled accordingly in favor of Panavision.

These cases serve to confirm that intellectual property disputes can cross borders. They are especially important with regards to trademark infringement within domain names and determination of proper jurisdiction in such cases.

International Shoe v. Washington. 326 US 310, 316 (1945)

In a case that goes back to the closing days of World War II, the International Shoe Company filed suit against the state of Washington concerning a dispute over due process under the state's unemployment compensation laws. The case was eventually appealed to the U.S. Supreme Court, which ruled that International Shoe was liable for prosecution under Washington state law per federal statute (26 USC 1606A), that the state had every right to sue it to recover payments not already rendered to the state unemployment compensation fund, and that the taxes levied by the state to support that fund was not a violation of the "due process" clause of the 14th Amendment to the U.S. Constitution.

*This does not appear to have anything to do with emulation until the Internet becomes involved. It is one of the precedent-setting cases for Internet litigation in that it helped to establish the legal concept of a viable forum, in which two parties can conduct business in a cross-border fashion. It also established the precedent that an offending party can be prosecuted under the laws of the place of residence of the offended party. See also **Asahi Metal Industry v. Superior Court** and **Compuserve v. Patterson**.*

Kewanee Oil v. Bicron. 416 U.S. 470 (1974)

This case dealt with the violation of confidental trade secrets that were protected under the terms of an exclusive contract. Kewanee Oil, developers of a specialized crystal-growing process, sued Bicron after the latter company obtained the process from former Kewanee employees and utilized it themselves. It was eventually heard in the U.S. Supreme Court, which wound up overturning a permanent injunction against Bicron's use of the process. The Supreme Court ruled that a state (in this case Ohio) could not protect a trade secret via statutory law, that the public as a whole does not benefit by such disclosure, that federal patent law does not conflict with trade secret issues, that a public domain process is not eligible for patent protection, and that a trade secret cannot be patented if serious questions arise as to its patentability. Below are some of the quotes from that decision that are frequently cited with regards to contracts for computer software.

> "If something is to be discovered at all, very likely it will be discovered by more than one person.... Even were an inventor to keep his discovery completely to himself, something that neither the patent nor trade secret laws forbid, there is a high probability that it will be soon independently developed. If the invention, though still a trade secret, is put into public use, the competition is alerted to the existence of the inventor's solution to the problem and may be encouraged to make an extra effort to independently find the solution thus known to be possible."
>
> "A contract cannot be used to go beyond the limits of the statutory right. Under our law, as it has developed, the statutes provide the limits to contract."
>
> "The necessity of good faith and honest, fair dealing, is the very life and spirit of the commercial world."

The implications with regards to reverse engineering and the presumed sanctity of the EULA are obvious, with the second quote about contracts and statutory rights often frequently noted in contract violation disputes. This was the defining case concerning EULA limits for over two decades, but its reach has been sharply limited due to recent developments. See also ***ProCD v. Ziedenberg.***

Lasercomb America v. Reynolds. 911 F.2d 970 (4th Cir., 1990).

This dispute centered around a computer program called *Interact* that had been developed for an industrial die-making process. Lasercomb, the author, had licensed the object code to a company named Holiday Steel under the terms of a highly restrictive EULA containing a number of non-competetive and non-intrusional clauses. Holiday duplicated the program in violation of the EULA, had one of their programmers reverse-engineer it, encrypt the output so as to disguise its origins, and then sold the resultant program under the name *PDS-100*. Lasercomb promptly charged them with two counts of intellectual property infringement, two counts of fraud, and one count of unfair competition. The district court found for Lasercomb on all counts, but Holiday Steel appealed the decision. One of the arguments in the appeal was that Lasercomb's EULA was anti-competetive and contrary to public policy. The Fourth Circuit Court of Appeals agreed, and subsequently threw out everything except for one count of fraud.

This decision is important with regards to the anti-competitive natures of certain types of EULAs that might be imposed on a user. See also ***ProCD v. Ziedenberg, 1996****.*

Lotus Development. Corporation v. Borland Internationall, Inc. 49 F.3d 807 (1st Cir., 1995), aff'd per curiam U.S., No. 94-2003 (1996).

Lotus had made its fortunes with its *Lotus 1-2-3* spreadsheet package, staving off many attempts by its competitors to cut into its market share. Borland, better known for its programming packages, had decided to widen its horizons by producing several pieces of application software. The most popular one of these was *Quattro*, a speedy spreadsheet package that included as an extra feature a perfect copy of *1-2-3*'s "slash-bar" menu and support for *1-2-3* macros. Lotus promptly took Borland to court claiming that Borland had unlawfully duplicated these feature from *1-2-3*. The courts agreed and Borland was forced to remove them from subsequent releases of *Quattro*, as well as pay back royalties to Lotus and a heavy fine. Borland eventually won on appeal several years later, with the court noting that an application program's menu structure was analagous to the controls of a VCR. As such, it was a mode of operation rather than a unique expression and did not qualify for copyright protection. It was a hollow victory, though - they had already sold the rights to *Quattro* to another company, and *Lotus 1-2-3* had by that time already lost its one-time dominance in the spreadsheet market to newcomer *Microsoft Excel* - but it did save them from having to pay damages to Lotus over the affair.

This is the key court case that is frequently quoted with regards to the "look-and-feel" debate. The main lesson to be learned from this dispute is the recognition of the need for similar interfaces for similarly functioning programs, be they created either by the first company on the scene or a consortium of similarly-minded interests. The debate over the copyrightability of menu structures was finally laid to rest by ***MiTek v. Arce, 1996****.*

Lewis Galoob Toys, Inc. v. Nintendo of America, Inc. 964 F.2d 965 (9th Cir, 1992).

At the 1990 Spring CES, Lewis Galoob Toys announced the release of the Game Genie, a device that allowed NES users to enter codes that would make it easier to play their videogames. Nintendo promptly sued, claiming that the Game Genie lessend the life and playability of the game, and was eventually successful in blocking sales of the Game Genie within the U.S. before its commercial release. Meanwhile, across the border in Canada the following year, the Canadian federal courts ruled that the Game Genie did not pose a viable threat to Nintendo's profits and found in favor of Galoob. The Game Genie was commercially released in Canada, and U.S. courts reversed their decision in October 1991, permitting the sale of the Game Genie within the U.S. In 1992, Galoob sued Nintendo for compensation resulting from lost sales of the Game Genie as a direct result of Nintendo's legal maneuvering, and Nintendo was forced to pay Galoob US$15 millon in damages. This resulted in the legalization of so-called "game enhancement technology" that allowed users to play videogames in a manner which the original designer did not intend to occur.

This case has significant implications for the videogame emulation community. It legitimized the generation of a videogame display by the use of a device not approved by the original vendor, it ruled that such use was indeed "fair use" provided that the users paid for the original software, and that a videogame hardware or software vendor could not redefine its interpretation of the market impact of its products at whim to prevent derivative works based on same. It is also one of the key cases to consider when attempting to to defeat a "market impact" charge by a vendor.

M. Kramer Manufacturing v. Andrews. 783 F.2d 421 (4th Cir., 1986)

This dispute arose over the unauthorized duplication of a video poker game. The defendant had one of its engineers remove the ROMs from the poker game, dump them, and then make minor changes to them, including changing the vending company's name and portions of the displayed game text. The U.S. Fourth Circuit Court of Appeals held that copyright protection covers both the program itself and the displays that it generates to the extent that such displays are necessary for the proper function and operation of the program by a user. The court specfically noted the "fixed" nature of ROMs, and commented that a audiovisual work so "fixed" could not be altered without violating copyright law. As a result, it found in favor of the plantiff.

And you wondered why certain original software vendors get so upset about "ROM" hacking and patching? This was the first such case of its kind on record, to the best of my knowledge, and subsequent cases followed its lead in confirming the "uncorruptable" intent of computer programs that are "fixed" in ROM.

MAI Systems Corp. v. Peak Computer, Inc. 991 F.2D 511, 518 (9th Cir., 1993).

MAI Systems was both an manufacturer and software developer of computer systems. It held a number of copyrights on software developed specifically for its computers. Peak Computer was a computer support firm that often had to perform maintenance on MAI products. In order to do so, it would frequently have to turn on the computer and load proprietary MAI program code into system memory. It did not copy the code, which was wiped whenever the system was shut down. The problem arose when Peak began offering its customers the temporary loan of a similar MAI system until repairs could be made to the customer's own. MAI filed suit against Peak with regards to this practice, charging it with copyright infringement - the customer had not paid for the right to use a loaned system and loaned software in place of that which it purchased. The courts ruled in favor of MAI, and the Ninth Circuit Court of Appeals upheld the lower court's ruling. They found that "...unauthorized copying, for purposes of copyright law, occurs when a computer program is transferred from a permanent storage device to the computer's RAM." Such an action, regardless of its results, produced a copy of the copyrighted work as defined by the U.S. Copyright Act (17 USC). The resultant code in system RAM was considered to be a "fixed work" and therefore an unauthorized copy of same. To further quote the court, "the unauthorized copying of copyrighted computer programs is prima facie an infringement of the copyright."

While nobody in the computer industry seriously accepts the argument that any program uploaded to system RAM is copyright infringement, it is interesting to note that all vendor EULAs made since this decision now specifically authorize this action on the part of the user. The court's ruling is broad enough that it can be interpreted any number of ways and often is, but the key quote from the verdict is the last one. To wit, any copy that you make of a piece of software, whether it is legal or not, that is not authorized by the copyright owner is by definintion an infringing copy of same. The problem posed in the original situation was recently resolved with the passage of the Digital Millenium Copyright Act of 1998, which amended 17 USC 117 in order to make the uploading of computer software into system memory by any third party trying to service that computer legal under copyright law.

Maritz, Inc. v. Cybergold, Inc. 947 F.Supp 1328 (D. Mo., 1996).

This dispute arose over two on-line vendors offering identical services. One was located in the state of California, and the other in the state of Missouri. Maritz was the first to file suit, claiming that Cybergold was unlawfuly infringing upon its business. Cybergold countersued on the grounds of trademark infringement, claiming that Maritz's "goldmail" service might be confused with its own service (named after itself). Called in to resolve the dispute, a Missouri district court ruled that Cybergold's business activities crossed Missouri borders, thanks to its presence on the Internet, and it therefore had jurisdiction in the matter. Surprisingly, though, it ruled in favor of Cybergold, since Maritz was unable to provide sufficient proof to support its contention of trademark infringement. The two parties eventually elected to settle out of court.

What this means is that the mere presence of a venture on the Internet is sufficient to open the backers of such a venture to the possibility of an intellectual property dispute in a cross-border dispute, as the very nature of the Internet allows it to cross all borders.

MiTek Holdings v. Arce Engineering. No. 94-5262 (11th Cir., 1996).

MiTek and Arce were two well-known wood truss manufacturers in the building industry. Each company created their own computer program for determining the proper placement of these trusses during the construction of a building's roof. MiTek sued Arce over their rival's program, claiming that the menu structure of Arce's *TrussPro* was so much like that used by MiTek's *ACES* that it constituted copyright infringement. Arce lost its case in district court, and the decision was affirmed by the Eleventh Circuit Court of Appeals. Basing its ruling on the *Lotus v. Borland* precedent, the court ruled in effect that the command structure of a computer program is a generic process and therefore not copyrightable.

This pretty much sets the seal on the uncopyrightability on any kind of menu, option, or other kind of command structure contained within a computer program, an issue first addressed in ***Lotus v. Borland, 1990****. It is now perfectly legal to duplicate the command structure of a competitor's program; however, it is still not legal to duplicate any unique texts (such as help and information screens) generated by those commands, as established by* ***Digital v. Softklone, 1987****.*

Narrell v. Freeman. 872 F.2d 907, 913 (9th Cir., 1989).

I don't have much on this case beyond the following two quotes - one from the Sega v. MAPHIA docket, and the other from ADA v. Delta Dental, 1996 (not included in this document)

> "The doctrine of fair use allows a holder of the privilege to use copyrighted material in a reasonable manner without the consent of the copyright owner."
>
> "Phrases and expressions conveying an idea typically expressed in a limited number of stereotyped fashions are not subject to copyright protection.... "

The implications are clear with regards to intent of the user versus intent of the developer or owner of a copyrighted work. For example, if you wanted to hit your friend in the back of the head with a rolled-up newspaper, then you could not be prosecuted for copyright infringement; however, you might be prosecuted for assault under criminal law. In a computer-related example, you might decide to replace the worn-out circuit board inside a videogame cartridge with a better one *from another copy of the same cartridge whose label is not in as good shape as the first one; such an action would be considered reasonable even though it was not intended by the original vendor.*

Nintendo of America, Inc. v. Blockbuster Entertainment. (1987)

Blockbuster Video built its business upon providing surprisingly lenient terms for the rental of commercially released videotapes. When home console videogames came along in the 1980s, they rented them out to their customers, too. Nintendo, the largest manufacturer and licensor of videogames at the time, sued Blockbuster for copyright infringement. The chief complaint was that the practice of unauthorized software rentals of Nintendo videogames on the part of Blockbuster was a violation of federal law. The two parties eventually settled out of court, with Blockbuster retaining the legal right to continue the practice, but one particular aspect of the case deserves special mention. Nintendo successfully prosecuted Blockbuster for providing photocopied instruction manuals along with its rental NES carts. Blockbuster unsuccessfully argued that it was only protecting its investment in the original manuals, as it would cost more to replace the originals than it would to photocopy replacements. In this particular part of the case, the courts found in favor of Nintendo, therefore Blockbuster was required to start providing the original manuals with its rental games. After the settlement was reached, Blockbuster and other videogame rental outlets all but discontinued the practice of providing the manual with a rental game except by request - an action which went unchallenged. The practice continues to this day.

The implications from this are obvious - any copy of the original instruction manual for a piece of computer software that has not been authorized by the original vendor or developer is illegal. This has also been extended in later cases to cover documentation for any and all computer-based systems.

Nintendo of America, Inc. v. Bung Enterprises and Carl Industries. (C.D. Cal, 1997)

Bung Enterprises was a Hong Kong technology firm who had invented a sophisticated accessory for the N64 videogame console. The Doctor V64 plugged into a standard N64 console and allowed its users to dump their N64 cartridges to recordable CD media, as well as play the dumped games straight off the CD. Carl Industries was the primary distributor of the Doctor V64 in the United States. Nintendo sued both in a joint action in federal district court, seeking significant financial compensation for and an immediate injunction against the sale of the device within the United States and its territorial possessions. Nintendo claimed that its publishers and developers had lost an estimated US$810 million in potential worldwide sales as a result of software piracy, and Bung's product was the chief cause of that piracy. Carl Industries claimed that the Doctor V64 served a legitimate development need as defined by several court cases on the subject under U.S. law (Sega v. Accolade and Nintendo v. Atari, 1992) and that Nintendo's sole purpose in the dispute was to prevent any non-licensed Nintendo developer from working on products designed for the N64 - despite ample public evidence to the contrary. As a result, Bung was forced to change the way that it manufactured the Doctor V64 so that it could no longer be used to directly dump N64 cartridge data. Nintendo eventually triumphed in the affair, and the resultant decision effectively banned the manufacture, distribution, and sale of unauthorized cart dumpers for commercial purposes.

This represents the last and greatest victory for Nintendo in the cart dumper wars. The only way around the legal instructions imposed on cart dumping technology by this case would be under the concept of noncommerical use. See the "Betamax case," i.e. ***Sony v. Universal, 1984****.*

Nintendo of America, Inc. v. Games City. (S.D. Ca, 1997)

Games City was a retailer based in Monterey Park, CA who sold a number of cart-dumping devices for several videogame systems. Among these were cart dumpers for the Super Nintendo and Nintendo 64 videogame consoles. Due to the widespread underground popularity of the N64 cart dumpers, Nintendo elected to sue Games City for intellectual property infringement. Among the charges that it leveled against Games City, Nintendo claimed that its promotion of cart dumpers was an infringment of Nintendo's copyrights, since they could be used to create illegal copies of Nintendo video games. Games City eventually elected to settle out-of-court and paid a $100,000 settlement to Nintendo in order to resolve the dispute.

This was one of Nintendo's biggest victories in its prosecution of the practice of cart dumping, and helped them to build their case against cart dump manufacturers. See also ***Nintendo v. Bung Enterprises and Carl Industries, 1997****.*

Nintendo of America, Inc. v. Prima Publishing. (W.D. Wa, 1997)

Nintendo contended that a reproduction of a screenshot map used in Prima's guidebook to the videogame *GoldenEye* was an illegal copy of the same used in the official Nintendo guidebook. The judge dismissed the case, noting that the screenshot of the map display from the game fell under the definition of "public knowledge." It was meant to be shared gameplayers; therefore, Nintendo had no grounds for their claim.

Among other things, this confirmed the legality of screenshots and their free distribution, which is a practice often employed by emulator developers and videogame players to illustrate their achievements.

Nintendo v. Samsung. (1995)

Nintendo alleged that the Korean electronics giant was promoting software piracy by making certain hardware and software widely available to interested third parties. These parties were mostly Oriental outfits known for producing bootleg Nintendo games, including one firm owned by the Chinese government (no surprise there). The dispute was eventally settled out of court, with both sides pledging to help each other in efforts to stop software piracy.

This case helps to highlight the rampant bootleg markets in the Orient, especially with regards to computer software and videogames.

Nintendo v. Tengen. (1989)

In what is perhaps one of the most famous cases of software distribution rights for a videogame, Nintendo sued Tengen over their distribution of a copy of the popular Russian videogame Tetris designed for use with the Nintendo Entertainment System (NES). Nintendo had previously released its own version of Tetris, and claimed that Tengen's version was an unlawful violation of its distribution agreement with the game's original author, Alexy Pajitnov. Tengen countered that it had obtained its rights for distribution from a European licensee, Mirrorsoft. The courts ruled that Mirrorsoft did not have the authority to grant Tengen the U.S. distribution rights to Tetris and found in favor of Nintendo. Both Tengen and Mirrorsoft were forced to pull their copies from U.S. distribution and pay heavy damages to Nintendo. The Tengen Tetris NES game cartridge has since become something of a collector's item.

The "Tetris case" is important to consider with regards to the proper authorization of software distribution rights in accordance with copyright laws.

Penwalt v. Durand-Weyland, Inc. 833 F.2d 931 *en banc,* 4 USPQ 2d 1737 (Fed. Cir., 1987)

In this dispute, which involved the unintentional and independently derived duplication of a patented process, "the court applied the 'all elements rule,' holding that 'the district court correctly relied on an element-by-element comparison to conclude that there was no infringement under the doctrine of equivalents,' holding that this was the proper method of analyzing equivalents. Judge Nies, in a concurring opinion, emphasized this feature of the opinion: 'If an accused device does not contain at least an equivalent for each limitation of the claim, there is no infringement because a required part of the claimed invention is missing.'"

The above quote is from *Horseshoes, Hand Grenades, and the Doctrine of Equivalents: Where 'Almost' Counts* *by Alston and Bird LLP. The conclusion was there must be either literal or equivalent infringement to justify a patent violation claim. This opens the possibility that substituting a piece of software for physical hardware (i.e. an emulator) could be construed as patent violation. The requirement that "an accused device [must] contain at least an equivalent for each limitation of the claim" is important, as this was indirectly referenced in* ***Atari v. Nintendo, 1992****. It leads one to the inescapable conclusion that an unlicensed emulator's design may not go beyond those functions absolutely required to duplicate necessary functions from the original system unless otherwise approved by the vendor.*

Playboy Enterprises, Inc. v. George Frena. 839 F.Supp. 1552 (M.D. Fla., 1993).

George Frena, the sysop of the Techs Warehouse BBS, had 170 digitized images from both *Playboy* and *Playgirl* magazine posted to his computerized bulletin board system. The two magazines were commercial adult publications protected under copyright law. Playboy Enterprises, owner and publisher of both magazines, sued Frena for copyright infringement. The Federal District Court acknowledged Frena's claims that the uploading had been done by his users without his approval; however, it still found him liable for intellectual property violation. It ruled that Frena's users had illegaly copied the pictures by digitizing them; furthermore, Frena had infringed on exclusive vendor distribution rights by making the pictures available for download by his users. It also found Frena in violation of trademark law, since the infringing material contained registered trademarks belonging to Playboy Enterprises (the *Playboy* and *Playgirl* logos).

This case established two things. First, courts can find against a defendant in an intellectual property dispute whether or not the defendant is aware of such activity. Second, intellectual property protection extends to all copies of a given work regardless of how they are made or the media on which they are presented.

Playboy Enterprises, Inc. v. Terri Welles. 7 F.Supp. 2d 1098 (S.D. Cal., 1998).

In yet another grand legal case of FUBAR for the famous men's magazine, Playboy sued a young woman named Terri Welles for trademark infringement. Ms. Welles had previously appeared in a number of issues of *Playboy* and other such adult magzines published by the company, even going so far as to receive the honor of being the 1981 Playmate of the Year. She created her own semierotic website, which among other things (including adult content) advertised her former relationship with Playboy and repeatedly used Playboy trademarks as metatags for the various Internet search engines. In deciding the matter, the court ruled that Ms. Welles' use of the Playboy trademark was not infringing and justified by the "fair use" princple, as her prior relationship with Playboy was a matter of public record. The court noted that the so-called honor of being awarded the title of Playmate of the Year "... becomes part of their identity and adds value to their name. Indisputably, these winners represent the awarding organization of sponsor, but the title becomes part of who they are to the public." The court ruled in favor of Ms. Welles because "... it references not only her identity as a 'Playboy Playmate of the Year 1981' but it may also reference the legitimate editorial uses of the term 'Playboy" contained in the text of defendant's website."

Well, well, how 'bout that? It seems that if you are given a form of special identity by a particular sponsor which incorporates one or more of their registered trademarks, then you have the right to use those trademarks when such is absolutely essential to prove your claim to that identity. This is one of only a handful of "fair use" exceptions insofar as unauthorized trademark usage is concerned.

Playboy Enterprises, Inc. v. Universal Tel-A-Talk, Inc., et. al. No. 96-6961 (E.D. Pa., 1998).

In one of the more amusing cases of trademark infringement, the world-famous adult magazine publisher sued Tel-A-Talk for using the Playboy logo as a link to Playboy's own web site. Playboy made two charges: first, that use of its trademarked logo without authorization was a violation of the Landham Act; second, that such usage also constituted counterfeiting under federal law (which would have tripled any damages and attorney compensation fees it might have received). The U.S. District Court ruled against Playboy on both counts. On the first charge, it noted that Tel-A-Talk was not claiming to provide the same "services" as Playboy, therefore its use of the trademark to direct users of its website to Playboy's in order to obtain those aforementioned "services" was deemed valid. It threw out the second charge on the basis that the Playboy trademark had been registered to cover its magazine, which Tel-A-Talk had not attempted to duplicate in any way. The court gave Playboy the option to renew its motion as soon as it came up with a better case.

While you may be laughing as hard as I did when I first read about this dispute, there is an important point worth considering here. It is perfectly legal for you to use a trademarked name, phrase, or graphic to represent a link on your Internet site if such a link takes a user directly to the Internet site of the trademark owner. Many folks have seen this practice in use on a multitude of Internet sites, and this is the case that firmly established the validity of same.

Princeton University Press v. Michigan Document Services. 1996 U.S. App LEXIS 7474 (6th Cir., 1996).

Michigan Document was a photocopying service that did many jobs for nearby Princeton University. One of these was the making of "coursepacks" for certain classes. A coursepack is a compilation of readings from various print works assigned by a professor for a given class that are collected together into one document. Copyright law grants limited exemptions for educational institutions and other non-profit organizations to use copyrighted material in certain ways not available to the average person, and coursepacks are one of these accepted methods. Over the years, a system has developed whereby the copyright owner of any material used within a coursepack receives some compensation for same. Michigan Document did not honor this system; they made the required coursepacks without recompense to the copyright owners. They were subsequently sued for copyright infringment, but won in district court arguing that the practice was an exercise of "fair use" for educational purposes. The U.S. Sixth Circuit Court of Appeals reversed the decision of the lower court in an *en banc* opinion after careful consideration of all "fair use" factors.

What this means to the emulation community is that the vast majority of those beloved "ROM" packs that are floating around are illegal, as they contain copyrighted material for which the copyright owner has not granted the right of free distribution without recompense.

ProCD, Inc. v. Ziedenberg. 86 F.3d 1447 (7th Cir., 1996).

ProCD was the author of the SelectPhone database, which incorporated phone number and address information from over 3,000 phone directories across the United States. As a database, it could not be copyrighted due to an earlier court decision (Feist v. Rural Telephone Service, 1991). ProCD sought protection for its program through two different EULAs - one for end-users, and one for businesses wishing to further disseminate its data. Matthew Ziedenberg, owner of Silken Mountain Web Services, violated the EULA on ProCD's SelectPhone package by purchasing the product as an end-user and then disseminating the data on his web site. The business EULA for SelectPhone permitted this activity; however, Ziedenberg bought the cheaper version of the package whose EULA did not include this clause. Ziedenberg initially won in district court by arguing that the EULA imposed on the average end-user was too restrictive; however ProCD appealed the decision. The Software Publishers Association (SPA) and many other industry orgainzations filed numerous "friend of the court" briefs on behalf of ProCD, essentially saying that the federal government was letting users get away with too much with regards to intellectual property infringement, and recommending that EULA protections be strengthened. In the end, the Seventh Circuit Court of Appeals noted the high cost and tremendous effort that ProCD had put into the SelectPhone project - a massive undertaking that any competitor could duplicate if it so chose. Ziedenberg was attempting to enjoy the fruits of ProCD's labor without proper recompense. Its ruling found in favor of ProCD, noting three things. First, the court found that a EULA was implicitly accepted by the user at the moment that software was obtained (in this case, through purchase). Second, the court found that a EULA was a legally binding contract under sections 2-204 and 2-606 of the Uniform Commercial Code (UCC). Third, the court found that a EULA was to be considered a contract between two private parties (the vendor and the end-user) and therefore enforceable under federal law (17 USC 301, 49 USC 1305).

This case effectively overturned protections for computer software formerly assumed under ***Kewanee Oil v. Bicron****, with the result that EULAs are once again the front line of defense for software vendors seeking to protect their intellectual property rights. In order to ensure that this protection is not taken away from them again, software vendors have adopted a standardized format and use of phrasing approved by the federal government for EULAs issued since this decision was rendered. EULAs are no longer as restrictive as they once were, but the resultant language is quite specific with regards to priviliges that a vendor may grant to a user regarding computer software. One can still challenge the terms of a EULA as being overly restrictive, but the burden of proof once again rests on the user, not the copyright holder.*

Recording Industry Association of America v. Diamond Multimedia Systems. No. 98-5627 (9th Cir., 1999)

Diamond Multimedia Systems, better known for their audiovisual products for personal computer systems, produced the Rio MP3 audio recorder/player for sale. The Recording Industry Association of America promptly took Diamond to court over the issue, noting that the MP3 format was an unlawful duplication of music originally recorded in "Red Book" format for audio CDs and thus violated the Audio Home Recording Act of 1992 (17 USC 1001, et. seq.). The district court found in favor of the RIAA, but their decision was reversed upon appeal. The U.S. Ninth Circuit Court of Appeals noted that maufacturers of audio recording and playing equipment pay a small per-unit royalty to the music industry to partially offset the threat of audio piracy; however, this same agreement did not cover computer equipment. The Rio was not a "digital audio recording device" as defined by the AHRA, but "... the Rio is a device that makes copies in order to render portable, or 'space-shift,' those files that already reside on a user's hard drive .... Such copying is paradigimatic non-commercial personal use entirely consistent with the purposes of the [AHRA]." As a result, Diamond won their case and was allowed to continue the sale and manufacture of the Rio.

This case is worth considering with regards to cross-format duplication of copyrighted material.

Religious Technology Center v. Netcom. No. 96-20091 (N.D. Cal., 1995).

This dispute arose between the Church of Scientology and a former disgruntled member (Dennis Elrich) who put certain confidential church documents on the Internet for public access. The Church of Scientology claimed that Netcom had no authority to permit the posting of these documents and that the Religious Technology Center (or RTC, the publishing branch of the Church of Scientology) had sole legal right to these documents. A federal district court granted a preliminary injunction in February 1995 permitting the search for and seizure of those documents; however, it was forced to amend its decision later that month when it became clear that the search had been conducted in an overzealous manner by law enforcement officials under the direction of the RTC. In its amended decision, the court that one of the documents in dispute had fallen into the public domain due to the plantiff's neglect in renewing the copyright; therefore, it was not an issue and could be legally posted. It next applied the four-point "fair use" test as defined by copyright law (17 USC 107). Netcom passed the first point in the test, since use for critical purposes is recognized by law. It failed the second point with regards to part of the documents (noted as Exhibit B in the official court documents) because of the claimed religious nature of the works in question - the undisputed portion (noted as Exhibit A) were of a purely informative nature and therefore non-infringing. It failed the third point on all counts, since either an entire document or the "heart" of the document was posted, per the Nation case (Harper and Row v. Nation, 1985), Finally, it passed the fourth and most important part of the test, the marketablility principle, since the defendant was not involved in "a systematic attempt ... at posting the complete works necessary for setting up a competing religious group." Subsequently the court determined that Netcom's usage of copyrighted Scientology material failed the "fair use" test. The court ruled that Netcom had no right to post any of the material in dispute save for third-party critical matter that was independent of the documents in dispute and the one document that had fallen into the public domain due to the lapse of its copyright. It did rule, however, that the subsequent search and seizure overreached its legal bounds, and RTC was directed to return or provide compensation for any and all material unlawfully seized that was not directly involved in the dispute within ten (10) days of the court's ruling. RTC had no choice but to comply with the court's order.

This is one of the defining cases with regards to copyright infringement on the Internet. The heart of the court's decision can be found in its concluding remarks, which state the following: "Unauthorized reproduction, transmission, or publication includes placement of a copyrighted work into a computer's hard drive or other storage device; 'browsing' the text of a copyrighted work resident on another computer through on-screen examination; scanning a copyrighted work into a digital file; 'uploading' a digital file containing a copyrighted work from the computer to a bulletin board system or other server; 'downloading' a digital file containing a copyrighted work from a bulletin board system or other server to the computer; and 'quoting' a copyrighted work that is cited in an on-line message in sending, responding to or forwarding that message...." It also affirmed that posted extracts from a copyrighted work "must comprise only a very small percentage of the copyrighted works both from a quantitative and a qualitative standpoint" and may not include "the heart of the work." It serves as an excellent example of the "fair use" test in action with regards to an Internet issue, and it should be noted that Netcom's failure to pass the test rested on the authorship and scope principles, which are frequently overlooked by both developer and vendor alike.

Sega Enterprises, Ltd. v. Accolade, Inc. 977 F. 2d. 1510, 1517 (9th Cir., 1992).

Accolade, a noted developer of videogame software, had produced six unlicensed games for the Sega Genesis home videogame console. Sega promptly sued Accolade, claiming copyright and trademark violation. The courts upheld Sega's contentions, but the decision was later partially reversed upon appeal. The Ninth Circuit Federal Court found that it was "fair use" for Accolade to dump and decompile the internal codes of the Sega Genesis and its games for developmental purposes, provided there was no other way to gain access to the concepts involved in their operation. This would have permitted Sega to establish a de facto monopoly over games for the console, which would have been unfair to Accolade. However, it was also ruled illegal for Accolade to activate the Sega Trademark Security System (TMSS) within these unlicensed games, however unintentional that may have been, since this gave the wrong impression that Sega had authorized Accolade's unlicensed titles when in fact they had not. Sega and Accolade eventually settled their differences out of court.

This case is notable because it established the legality of dumping a ROM-based storage device (like a game cartridge) under the strict rule of pure development purposes. Such dumps are considered "intermediate copies" under the law, and it is clearly understood that they are not meant for distribution without the consent of the copyright holder. The decision is frequently applied to any kind of practice involving an IC dump, but apparently does not apply to the practice of BIOS dumping; see ***IBM v. Compaq, 1982*** *and* ***Apple v. Franlklin, 1983****. It is is the second precedent-setting case regarding the limited "fair use" protection for the practice of "cart dumping," with the first being* ***Nintendo v. Atari, 1992****. It is also an important ruling for the emulation community with regards to unauthorized trademark usage, due to the fact that many systems or software titles make use of trademarks (of display of same) during the bootstrap process, as well as actual system operation or performance. Such usage has been deemed to be non-infringing as a result of this case, since the original vendor put those trademarks in their software for the express purpose of being displayed. The emulator vendor therefore has no choice but to display them, as messing with a trademark display is a violation of the Landham Act per* ***Playboy v. Frena, 1993****.*

Sega Enterprises, Ltd. v. MAPHIA. 857 F.Supp. 679, 686 (N.D. Cal., 1994).

MAPHIA was the self-appointed moniker of a group of computer software pirates who owned and operated a number of computerized bulletin boards operating in and around the San Francisco area. Part of the "warez" that were offered on their site were cart dumps (nowdays called "ROMs") of Sega Genesis games, along with instructions on how to dump the original game carts and how to both obtain and operate the requisite cart dumpers. Users of the bulletin boards uploaded and downloaded Sega software with the full knowledge and encouragement of MAPHIA. Subsequently, Sega filed an action in federal district court against MAPHIA, charging them with multiple counts of trademark and copyright infringement. The copyright infringement was obvious; Sega also contended that downloading and then running these unlawful copies also infringed upon its corporate image and thus constituted trademark violation as well. The court granted an injunction in Sega's behalf, and in accord with that ruling local law enforcement officials barred MAPHIA from carrying unauthorized copies of Sega software on its bulletin board for any reason whatsoever. In accord with the court's injunction, they also seized all infringing copies of Sega software, as well as all materials relating to their storage and distribution, to ensure that MAPHIA would be seriously handicapped in any effort to resume such unlawful practices.

This is the most important court case up to this point with regards to the Internet and the emulation community. It appears to have served as one of the guides for a new provision of U.S. copyright law (17 USC 512) dealing specifically with on-line service provider liability that was laid down with the passage of the Digital Millenium Copyright Act in late 1998. The conclusions one should draw from this case are obvious and need not be explained.

***Shetland Times* v. Dr. Jonathan Wills and Zetnews Ltd.** Court of Session, Edinburgh, Scotland (Lord Hamilton, 1996).

> Zetnews Ltd, an Internet information site based in Scotland, had developed the practice of duplicating headlines from the website of the *Shetland Times* newspaper, which served as links to the actual articles on the *Shetland* website. Once they learned of this practice, *Shetland* sought an injunction against Zetnews and its webmaster, Dr. Jonathan Wells, ordering them to cease and desist this practice. Their claim was based on perceived violations of the Scotland Copyright Design and Patents Act of 1988, sections 7 and 20. In granting the injunction, the Edinburgh Court of Session noted "access to the pursuer's items ... can be obtained by bypassing the pursuers' front page and accordingly missing any advertising material which may appear on it." The court ruled that *Shetland* had "... a *prima facie* case that the incorporation by the defenders in their Web site of the headlines provided at the pursuers' Web site consititutes and infringement ..." and granted the injunction, subject to further litigation on the subject.
>
> *Although decided under Scottish law, this is considered to be the landmark legal case with regards to the legality of linking. Any link that involves the unauthorized use of commercial materials or directs a users to or from a commercial site in an unauthorized manner is illegal under copyright law. The Shetland case has been the guiding factor in subsequent court decisions within the United States; see also* ***Futuredontics v. Applied Anagramics, 1998***.

Sony Corporation of America v. Bleem LLC. (9th Cir., 1999)

> Sony sued Bleem LLC over the release of *bleem!*, a commercially produced PlayStation (PSX) emulator for IBM compatible computer systems, using the usual arguments. Howerver, Judge Legge eventually refused to stop the release of *bleem!* after numerous requests by Sony for a restraining order, noting that Sony's claims in this case were without merit. Bleem LLC was able to successfully prove the use of "clean room" techniques during the development of their PSX emulator, with the result that their product did not contain any infringing code.
>
> *It seems that the videogame console market has now going through what the personal computer industry endured a decade ago during the dispute between Apple and Readysoft over Macintosh emulation. These two cases marked the first release of an videogame console emulator for a product that was still commercially viable, which was also another eerie parallel to the Apple-Readysoft dispute (as the Mac was also a viable system at the time). Once again, rapid strides in the emulation community have caught up with the technology being emulated. These two cases are still working its way through the courts and will go to trial in the first half of the year 2000. It is important with regards to the future of videogame emulators, in particular those that are*

*designed specifically for use with the actual console's software base within its original delivery system. See also **IBM v. Compaq, 1982**, **Intel v. AMD, 1991**, and **Sony v. Connectix, 2000**.*

Sony Corporation of America v. Connectix, Inc. No. 99-15852 (9th Cir., 2000).

Sony sued Connectix over the release of the Virtual Game Station (VGS), a commercially produced PlayStation (PSX) emulator for the Macintosh computer system, using the usual arguments. VGS was designed to use actual PlayStation videogames (as opposed to the cart dumps or disk image files used by many other videogame emulators), and these were manufactured by Sony and its licensees in the same CD-ROM format as that used by most personal computers - including the Macintosh. The U.S. District Court refused to grant a preliminary injunction against Connectix, noting that Sony's claims were without merit, but Sony appealed both cases. Judge Charles Legge of the 9th District Court of Appeals later agreed to block the sale of *Virtual Game Station* after agreeing with Sony that the Virtual Game Station possibly employed portions of PSX microcode as part of its operations, which was and still is a reverse-engineering "no-no" due to copyright violation. Connectix appealed the disctrict court's ruling, and the case was thrown out by the U.S. 9th Circuit Court of Appeals the following February. In its opinion, the high court deemed the development and release of an emulator to be noninfringing provided that no patents were violated and that the final product itself did not contain any infringing code; furthermore, it also ruled emulation itself to be protected fair use of computer software under 17 USC 107.

*This is the case that legalized emulation once and for all. It should also be noted that the high court's ruling also permitted emucoders to use whatever copyrighted code they pleased from the original system (such as the BIOS) while developing their emulator, so long as the final product did not require the use of such code in order to function. That is why it is still illegal to use BIOS dumps with an emulator - the developer can do that all day long while coding the emulator, but not a user of any public release version. It is yet another example of the "developer's exception" first granted under the concept of the intermediate copy; see **Sega v. Accolade, 1992**.*

Sony Corporation of America v. Universal Studios, Inc. 464 U.S. 417 (1984)

In the now-famous "Betamax case," Sony argued that court actions on the part of Universal (and Disney) was an unlawful infringement of legal technology. Universal argued that Sony's new Betamax videocasette recorder (VCR) permitted the unlawful duplication of their copyrighted television programs. The case worked its way through the courts for years, eventually going all the way to the U.S. Supreme Court. As part of the case, a videotaped copy of *The Mickey Mouse Club* was played for the high court justices. Not amused, the court eventually ruled that private, non-commercial copying of television broadcasts does indeed qualify as fair use. To quote from the Supreme Court's ruling:

> The sale of copying equipment, like the sale of other articles of commerce, does not constitute contributory infringement if the product is widely used for legitimate, unobjectionable purposes. Indeed, it need merely be capable of substantial noninfringing uses.... Whatever the future percentage of legal versus illegal home-use recording might be, an injunction which seeks to deprive the public of the very tool or article of commerce capable of some noninfringing use would be an extremely harsh remedy, as well as one unprecedented in copyright law.... [Even when an entire copyrighted work was recorded, such copying is deened fair use] because there is no accompanying reduction in the market for [the] plaintiff's original work.... A use that has no demonstrable effect upon the potential market for, or the value of, the copyrighted work need not be prohibited in order to protect the author's incentive to create."

This case is often cited by "free software" advocates concering personal duplication of computer software that is not intended for anything beyond private use, and has been successfully used on more than one occasion to defend the practice for various and sundry reasons. It has never succeeded as a defense for software piracy, though, since that activity goes beyond contributory infringment into direct infringement, and is therefore still illegal. EmuFAQ contributor Chuck Cochems argues that this case opens the door for personal use of noninfringing ROMs, and you can read all about it in his intriguing article, "The Question of ROMs."

Stac Electronics v. Microsoft Corp. No. 93-0413 (C.D. Cal, 1994).

Stac Electronics had developed a popular personal computer utility program called *Stacker*, which allowed system owners with small hard drives and not enough money to buy larger ones to effectively double their available storage space. It did this by using advanced compression algorithims to reduce the amount of storage space required for all material stored on a user's hard drive, uncompressing it as needed by the user, and Stac took out a number of software patents on its compression technology. Microsoft, always on the make for any technologies that might complement or compete with sales of its MS-DOS operating system, developed the *DoubleSpace* utility for MS-DOS 6.20 to do the same thing. Stac prompty sued under the terms of the U.S. Patent Act, claiming infringment of its proprietary patents on disk compression technology. The courts found in favor of Stac, and Microsoft was forced to remove *DoubleSpace* and immediately issue a new release without it (MS-DOS 6.21). As always, the resourceful folks at Microsoft found a way to deal with the situation. First, it developed its own reverse-engineered disk-compression algorithim that, while not as effective as Stac's, did the job well enough - this was included in the next release of the operating system (MS-DOS 6.22's *DriveSpace*). Second, it eventually bought out Stac, thereby gaining the rights to its proprietary software patents on *Stacker* technology. *Stacker* as a standalone product subsequently disappeared, with the technology being folded into *Windows 95* and all subsequent *Windows* releases.

This is perhaps the best-known example of software patent violation within the personal computer industry. Microsoft's reaction is typical of its behavior during its corporate growth - if you can't use the technology, then buy the company. Emulator developers should keep this example in mind whenever they have to deal with a piece of code protected by one or more software patents. As this case showed, reverse engineering is the only legal means available to bypass patent protection, but it does not necessarily protect you from subsequent court action on behalf of the patent owner.

Tandy Corporation v. Personal Micro Computers. 524 F. Supp. 171 (N.D. Cal., 1981)

To quote the Schwegman summary, "Tandy claimed that Personal Micro Computers took Tandy's I/O routine from a ROM chip and copied the chip for use in Personal Micro's computer. A computer program is subject to copyright protection. A silicon chip having the program imprinted on it is a "tangible medium of expression" of the program and so is also subject to copyright law. Personal Micro's motion to dismiss was denied."

This ruling would set the stage for all subsequent cases regarding the unauthorized duplication of computer code stored within a ROM chip. It has grave implications for anyone who wants to dump the contents of a ROM without the permission of the copyright holder to the object code stored within the ROM.

Vault Corporation v. Quaid Software. 655 F. Supp. 750 (E.D. La., 1987).

Vault had developed the PROLOK copy-protection scheme for software vendors in order to prevent unauthorized duplication of their products. Quaid Software designed and released a commercially available archiving program called *CopyWrite* that would bypass PROLOK, and allow any program protected by PROLOK to be successfully duplicated. Vault sued, claiming copyright violation via production of an infringing program that resulted in an unauthorized derivative work. The Fifth Circuit Court ruled that it was perfectly legal for *CopyWrite* to perform its archival function as designed, including its infringement by duplicating copyright Vault computer code, since it allowed a user to exercise legally acceptable archival rights as defined in the U.S. Copyright Act (17 USC 117). It also determined that the actual infringing code replicated from PROLOK was too small for the resultant copy to be considered an unauthorized derivative work. Another interesting ruling that came from this case was that Vault's EULA, which specifically prohibited decompilation or disassembly of its PROLOK program (which Quaid had done in order to understand its functions) was unenforceable under federal law.

This case helped establish the need to successfully duplicate copy-protection schemes in order to ensure successful function of a resultant archival copy. As such, it legitimized all forms of duplication software designed to replicate copy-protection schemes. It also opened the door to what is known as the "crack patch," a unique and popular offshoot that completely removed the copy-protection scheme altogether, provided that the use of such a patch was in accordance with copyright law. "Crack patches" were later outlawed by the Digital Millenium Copyright Act of 1998 (17 USC 1201), thus limiting lawful archiving of copy-protected software to only those technologies that duplicated the intended copy protection scheme along with the software being archived. This case is also important with regards to the concept of fair use, in that it recognized the legality of a software licensee's use or adaptation of a computer program without the authorization of its licensor if the resultant copy is "created as an essential step in the utilization of the program in conjunction with a machine."

Williams and Wilkins v. United States. 487 F.2d 1345 (1973), 420 U.S. 376 (1975)

This case involved the National Library of Medicine, a private library operated by the National Institutes of Health (a federal government agency). In 1970, the library made some 93,000 photocopies of commercial perodicals in order to keep up with the demands of researchers using the facilities. The action was subsequently challenged in a class-action lawsuit by the publishers of 37 different medical perodicals, claiming that such wanton photocopying was copyright violation. The courts ruled that it was "in the public interest" of the National Library of Medicine to make these copies available, and a divided court found in favor of the federal government. The decision was subsequently upheld on appeal. It was not long after that Congress amended the U.S. Copyright Act to include the "safe harbor" proviso for libraries.

This case is important becasue it established the public interest principle of "fair use" - in other words, use of copyrighted material in an unauthorized manner is legal so long as it can be conclusively proven that such usage serves an acceptable public service. The courts tend to be rather sticky on just what is an "acceptable public service," which effectively limits the practice to established non-profit and public service institutions such as churches, libraries, schools, etc.

I would also recommend checking out the Phillips Nizer Internet law library (http://www.phillipsnizer.com/), as they have a wealth of data concerning many of the issues related to the computer industry. Many of the summaries I have included are based on my research at the Phillips Nizer site.

Appendix C
Emulation Timeline

The following is a summary in brief of some of the more important events in the history of personal computer and videogame emulation. For a more detailed look at the subject, including any items that you feel were unjustly omitted, I suggest you read the original, *The History of Emulation*, which you can find in the articles section of Zophar's Domain (http://www.zophar.net/).

Birth Pains: Emulation Prehistory (1800 - 1961)

1822	British mathematician Charles Babbage invents the difference engine, the world's first computer. The steam and gear driven technology of the day is incapable of making his design a reality.
1842	Lady Ada Augusta Byron, the Countess of Lovelace and friend of Charles Babbage, writes the world's first computer program.
1889	Nintendo is founded in Japan by Fusajiro Yamauchi.
1896	IBM is founded in the United States by Herman Hollerith.
1931	German engineer Konrad Zuse invents the world's first calculator.
1939	American academians John Atanasoff and Clifford Berry build the world's first working computer.
1941	The British government uses Alan Turing's Colossus computer and its successors during World War II to simulate certain functions of the German Enigma coding machine. This is considered by some to be the first practical example of emulation, long before the concept itself was defined.
1945	American naval officer Grace Murray Hopper discovers the first computer "bug."
1958	American computer programmers Willian Higginbotham and David Potter devise *Tennis for Two*, the world's first videogame.
1956	Sega (SErvice GAmes) is founded in Japan by American entepreneur David Rosen.
1958	American engineer Jack Kilby invents the integrated circuit (IC).

Thus Spake Zarathustra: The Dawn of Emulation (1962 - 1988)

1962	American programmer Steve Russell devises *Space War*, the world's first arcade-style videogame. American engineer John Haanstra, an employee of IBM, devises the world's first cross-platform computer. American engineer Larry Moss, a member of IBM's System/360 development team, devises the concept of computer system emulation. For this, he is widely regarded as "the father of emulation."
1965	IBM releases *7070 Emulator* for the System/360 product line, permitting the use of IBM 7070 programs on high-end System/360 computers. It is the world's first emulator.
1968	Intel (Integrated Electronics) is founded in the United States by Gordon Moore and Robert Noyce.
1969	The Internet comes into existence.
1971	American computer enthusiast John Blankenbaker devises the world's first personal computer (PC). Author Julian Reitman predicts the rise of the emuscene in his book *Computer Simulation Applications*. Magnavox releases the world's first home videogame system. It is the Magnavox Odyssey, and it based on the work of videogame pioneer Ralph Baer.
1972	Atari markets *Pong*, the very first commercial arcade videogame.
1974	MITS markets the Altair 8800, the world's first commercially vended PC. It is the brainchild of Ed Roberts and Bill Yates.
1975	Microsoft is founded in the United States by Bill Gates and Paul Allen.
1976	Apple Computer is founded in the United States by Steve Jobs and Steve Wozniak. Fairchild markets the Channel F, the world's first cartridge-based videogame system. The long-running debate over violent content in videogames officially kicks off with a public outcry over Exidy's *Death Race*, in which players are awarded points for running over people.

1978	Atari releases the legendary cartridge-based Video Computer System (VCS) videogame console, later redubbed the Atari 2600.
1989	Motorola releases the MC68000 CPU, the first processor for a PC powerful enough to permit the possibility of sophisticated software-based emulation on these systems.
1989	Microsoft releases the Z80 Softcard, the very first computer system emulator designed for use with a PC. It permits the use of CP/M software on a suitably equipped Apple II PC.
1981	IBM releases the IBM PC, which for better or for worse has become the system that has shaped both the design and direction of most mainstream PC technology ever since. Because of this, the term "PC" is now nearly synonymous with IBM compatible personal computers. Acorn Computers Ltd. releases the legendary BBC Minicomputer System, one of the most venerated of the classic British PC designs. Computer code is deemed eligible for copyright protection In *Tandy v. Personal Microcomputer*. An American dies of a heart attack while playing the arcade videogame *Berzerk*. He is the videogame industry's only confirmed fatality for almost two decades.
1982	Commodore Business Machines releases the legendary Commodore 64, the first PC to incorporate both multichannel sound and multicolor graphics into its basic design. It would go on to sell more units during its lifetime (some 4 million) than any other basic PC design and would also become the first PC to break the US$1 billion mark in sales. Compaq Computer Corporation releases the Compaq Portable PC, the first legal clone of the IBM PC and the first portable "PC clone." The legality of reverse engineering a computer system BIOS is validated by the subsequent lawsuit, *IBM v. Compaq*. Intel releases the i80286 CPU, which contains within its internal code the first known firmware emulator for a personal computer part. This code permits the 16-bit i80286 to run practically all programs originally designed for use with the older 16-bit i8086. Coleco releases the ColecoVision Expansion Module #1, the world's first videogame system emulator. It permits the playing of Atari VCS games in a suitably equipped ColecoVision console. The legality of unlicensed videogame system emulation by pure hardware/firmware techniques is

	established in the subsequent lawsuit, *Atari v. Coleco*. Atari releases the Atari 5200, the world's first back-compatible videogame console.
1983	The illegality of the dumping of computer code stored within the ROMs of videogame hardware by the average user is established by *Atari v. JS&A Group*. The legality of copying and using protected forms of expression for noncommercial purposes is established by the famous "Betamax case," *Sony v. Universal*. The final ruling is not issued until the following year.
1984	Four legendary computer systems are released or made ready for release in America during this year, four that would forever change both the PC and videogame console industries. They are the Apple Macintosh (Mac), the IBM PC-AT, the Commodore Amiga, and the Nintendo Entertainment System (NES, the import version of the Famicom). The illegality of incorporating unlicensed BIOS code into a competing computer product is established by *Apple v. Franklin*.
1985	Microsoft releases the very first version of *Microsoft Windows*. It is an obvious rip-off of the Apple Macintosh operating system designed for use on IBM compatible systems. Nobody notices, and nobody cares - except Apple, who promptly file a copyright infringement lawsuit against Microsoft. Commodore's Amiga development team devises *Transformer*, the first "true" (software-based) computer system emulator of any kind for a PC platform. It allows a stock Amiga to emulate a stock IBM PC.
1986	Intel releases the i80386 CPU, the first processor powerful enough to permit sophisticated software-based emulation on an IBM compatible platform. The illegality of the unauthorized modification of computer code stored in ROM is established by *Kramer v. Andrews*. Avant Garde Systems releases *PC-Ditto* for the Atari ST, the first "true" computer system emulator for PCs to hit the market. It allows a stock Atari ST to emulat a stock IBM PC. Commodore's *Transfomer* software is rushed to market under the banner of Insignia Solutions in direct response to the release of *PC-Ditto*.
1987	Commodore releases the Amiga 2000, the first PC specifically designed

	with emulation in mind. The idea that on-screen textual displays of an informative nature can be protected by copyright is established by *Digital v. Softklone*. The possibility that an emulator could violate vendor patents is established under the doctrine of equivalents by *Penwalt v. Durand-Weyland*.
1988	Atari devises the *Rabbit* emulator as a means of bypassing the 10NES lockout on Nintendo's proprietary format for NES videogames. It is based on illegally obtained Nintendo source code. It subsequently markets several games utilizing *Rabbit* under the Tengen banner. Nintendo eventually sues Atari a couple of years later over unlicensed games released by Atari utilizing *Rabbit* technology. The legality of archiving copy-protected computer programs is established by *Vault v. Quaid*. Gadgets-by-Small releases *Aladdin* for the Atari ST, the world's first Mac emulator. Sega releases the 16-bit MegaDrive videogame console in Japan. Among its more notable features is back-compatibilty for Sega Master System games, achieved via a form of firmware emulation. The console would be exported to the United States the following year as the Sega Genesis. Readysoft unveils *A-Max* for Amiga systems at the World of Commodore show in November. Designed by Simon Douglas, it is the Amiga's first Mac emulator. Unlike *Aladdin*, it provides almost perfect emulation of a stock Mac on a stock Amiga - right down to native Apple floppy format support on stock Amiga disk drives. Subsequently, Apple threatens a lawsuit.

Salad Days: The Golden Age of Emulation (1989 - 1998)

1989	A hacked bootleg version of *A-Max* becomes the emuscene's first emulator to require a BIOS dump for proper operation. The widely anticipated lawsuit by Apple against Readysoft over *A-Max* never materializes. As a result, what is now termed "the *A-Max* affair" legitimizes unlicensed PC emulation, setting off a virtual explosion in emulation technology that is one of the chief hallmarks of the Golden Age.

	The high cost of math coprocessors for PCs of the day results in a number of software-based FPU (floating point unit) emulators being released into the public domain. The most popular and by far the most notable of these is Ron Kimball's *87EM* for IBM compatible systems.
1990	The U.S. Secret Service launches Operation Sun Devil, a nationwide sweep against all forms of illegal computer activity. Although hailed as a victory, the actual results were mixed and its impact only temporary at best. Anticompetitive EULA language, such as the denial of reverse engineering rights, is held to be unlawful in *Lasercomb v. Reynolds*. Videogame enhancement technology is determined to be noninfringing in *Galoob v. Nintendo*.
1991	Yuji Naka, author of *Sonic the Hedgehog*, devises the very first software-based videogame system emulator. It is an unreleased NES emulator for the Sega Genesis/MegaDrive (G/MD) videogame console. The ban on advertising on the Internet is finally lifted in October - a decision that many will rue in the years to come.
1992	The concept of intermediate copies is defined by *Sega v. Accolade*. The possibility that an emulator could violate copyright protection is established under the notion that it could infringe "protectable expressions" or go beyond its "stated purpose" in *Nintendo v. Atari*. Argonaut Software devises *Gameboy 68000* for Amiga systems, long held as the emuscene's first software-based videogame system emulator (until Yuji Naka's admission to *GameWeek* in mid-1999). It still retains its claim as the first videogame system emulator to hit the infant Internet emuscene. The notion that only limited, specific portions of on-screen graphical displays can be protected by copyright is established by *Apple v. Microsoft*. *AmIBM*, the first-ever hoax emulator, is released.
1993	German emucoder Christian Bauer releases *Shapeshifter*, the first public domain Mac emulator for Amiga systems.
1994	The Careless Gamer releases *MegaDrive*, the first working Sega G/MD emulator and the first ever developed by and for the emuscene. It is a "true" (software-based) emulator, as preferred by the emuscene, and all

	later references to emuscene-derived emulators will be of this type unless otherwise noted. The illegality of unlicensed distribution of commercially derived ROMs via telecommunications techniques is established by *Sega v. MAPHIA*.
1995	Activision releases the first of its *Activision Classics* multigame packs, the first time that software-based emulation is employed in a commercially vended videogame product. The Internet emuscene begins as the very first dedicated emusites go online. Legendary emucoder Marat Fayzullin releases *Virtual Game Boy*, the first cross-platform videogame emulator and the first with open source code.
1996	The oldest continuously operated emusites still operational today go online. Among their number is the legendary Vintage Gaming Network, originally founded as Dave's Video Game Classics and still by far the most popular emusite on the Internet today. Legendary emucoder Nicola Salmora releases the very first version of *M.A.M.E.* (Multiple Arcade Machine Emulator). Gary Henderson and Jeremy Koot release the very first version of *SNES9X*, the first (and still the best) cross-platform Super Nintendo (SNES) videogame console emulator. Emulation comes full circle with the first release of Bernard Schmidt's cross-platform *UAE* (Universal Amiga Emulator).
1997	Chris George's *VSMC* becomes the first target of an emulator "hack" by the emuscene. Duddie and Rafu release the first verion of *PSEmu*, the world's first PlayStation (PSX) emulator. Bloodlust Software releases three remarkable videogame emulators within the span of a few months - *Genecyst* (G/MD), *NESticle* (NES), and *Callus* (Capcom CPS-1). They are long the standard by which competitors are measured due to their astounding excellence, and there is a sudden explosive rash of ROM sites to support them. Legendary emucoders _Demo_ and zsknight release *ZSNES*, the best of the DOS-based SNES emulators and still the "friendly rival" of *SNES9X* in terms of excellence. Its appearance causes a marked increase in the

	number of SNES-themed ROM sites. The first formal protest by a vendor against the emuscene is filed by Zyrinx Software against The Dump: Genesis over an unauthorized copy of *Zero Tolerance* posted there. RPGe releases the very first full English translation for a import videogame ROM, *Final Fantasy V* for SNES. The sudden surge in ROM sites on the Internet attracts the attention of the videogame vendors, including such notable names as Nintendo and Sony. It also gains the notice of one of the umbrella organizations for the vendors, the Interactive Digital Software Association (IDSA).
1998	On 22 March 1998, the IDSA launches the "great sweep" against the burgeoning Internet emuscene, bringing the Golden Age to a sudden and harsh end.

Paradigm Shift: The NextGen Wave (1998 - ????)

1998	As a result of the IDSA's "great sweep," practically all of the major ROM sites are shut down within a six-month period, along with a fair amout of emusites to boot. Only a few of the major emusites survive, and even some of these are briefly shut down for a time. It takes the emuscene over a year to recover from the IDSA action. The unreleased sequel to the arcade videogame *Q*Bert* becomes the first arcade videogame to be release into the public domain specifically for the enjoyment of the emuscene. The term "NextGen emulation" is employed for the first time. It is used is to describe developing emulators for 32-bit and 64-bit videogame consoles, such as the PSX and N64. President William Jefferson Clinton signs the Digital Millenium Copyright Act (DMCA) into law. Connectix releases *Virtual Game Station* (VGS), the first commercially vended PSX emulator. They are subsequently sued by Sony, and forced to withdraw it from retail sale for almost a full year.
1999	Episilon and RealityMan release *UltraHLE*, the world's first working N64 emulator. Sega releases the *Sega Smash Pack* by emucoder Steve Snake, the first

	time ever that a public domain videogame console emulator (*KGen*) "went commercial." David Herpolsheimer and Randy Linden release *bleem!*, the first commercially vended PSX emulator for IBM compatible systems. They are almost immediately sued by Sony. Nintendo announces that it will abandon the cartridge format for its console videogames, making it the last videogame vendor to do so. ASCII Software of Japan becomes the first vendor to threaten legal action over a ROM translation patch. Judge Charles Legge of the 9th U.S. District Court throws out Sony's third and final claim for an injunction against retail sales of *bleem!*, citing a conclusive lack of evidence that the emulator had violated any Sony copyrights. This ruling is held to have been the first to truly legitimze videogame emulation in the eyes of the general public, doing for it what the resolution of the *A-Max* affair did for PC emulation a full decade earlier. David Lloyd launches *OverClocked*, the emuscene's first comic strip. Sega becomes the first vendor to openly support the Internet emuscene. They do so by announcing the impending establishment of its own ROM site, specifically geared for Dreamcast customers.
2000	The unquestionable legality of unlicensed emulation is finally established by the U.S. 9th Circuit Court of Appeals in the case of *Sony v. Connectix*. Sega, in conjunction with NEC, launches its DreamLibrary service in Japan. This provides Dreamcast owners with hundreds of classic MegaDrive and PC Engine videogames for use under emulation on their consoles at a nominal fee. A Korean man dies of malnutrition after repeated marathon sessions playing his favorite RPG. He becomes the videogame industry's second known fatality to date. ***... and the story continues ...***