

## A BRIEF HISTORY OF THE DEMOSCENE\_

by Lassi Tasajärvi

### Generation 0/1

The generation that lived through early adolescence in the 1980s was the first to grow up with home computers and computer games. The computers available for home use were fairly rudimentary, as far producing graphics or music went, but using them for creative endeavors soon started to interest the kids after the novelty of the rather basic games wore off.

Traditionally, when a child starts to display a talent for art, he or she is steered towards piano lessons or drawing classes. The generation that grew up with home computers soon started to use the computers for both art

and music, at their own behest. For many the computer was just the closest tool available, but some kids chose the computer because they liked challenges and they were the ones that gave birth to a whole new culture.

The first home computers on the markets were extremely basic and using them for just about anything required an intimate understanding of how the computer worked and how to program it. The kids with creative talent and a good enough grasp of mathematics and new technology were able to use these computers for self-expression in surprisingly diverse ways.

This gave birth to the first generation of artists that started expressing themselves digitally. Many went on to study traditional art and use different media for self-expression, but the important point is that they learned to use computers to create whatever they wanted, on their own terms, without any knowledge of earlier art theories or techniques.

### Cracked by The Kid Next Door

In the 1980s, computer advertisements sold the image of a machine that had something to offer each member of the family. This primitive little box was supposed to store mom's recipes and dad the architect was going to use it to design buildings. These corny adverts painted a picture of a family arguing about whose turn it was to use the computer for the common good. What it came down to in real life was this: the arguments would deal with whether it was junior's turn to play computer games using the family TV or the family's turn to watch programs and how much money could be spent on new computer games per month.

The Commodore 64, the most popular home computer of its time, came with a tape deck that could load and save games and programs, using regular audio cassettes. The most forward-thinking radio shows would even broadcast computer programs that listeners could record on their cassettes and load into the computers. Customer service departments were clogged with calls from confused older listeners, when the request show was followed by ten minutes of infernal noise. Some thought that it was experimental music and the rest thought that their radios had broken down.

Tape decks and the early versions of the disk

drive could easily be used to copy games. Game manufacturers had naturally equipped their wares with different copy protection methods. Some were technical, but others consisted of codes you had to feed in to gain access to games. The codes came with the games and were printed in a way that prevented copying, so the kid next door had to buy his own copy or borrow the original, along with the code card.

It didn't take long, however, before kids bored with playing started to experiment. Pages and pages of code that came with the computer or in magazines were used to learn about the art of programming and the structure of the computer. We're talking about kids between the ages of 10 and 20. It didn't take long for the most adept to figure out the secrets of the computer and start looking into how games were made. If you knew what you were doing, you could dig up the part of the program code that contained the copy protection. This made it easier to disable it. The altered program was saved, copied and distributed among friends. Parents were happy, since requests for money decreased noticeably. Game manufacturers were understandably upset by this development and started developing more elaborate and advanced copy protection technologies. This, in turn, motivated the kids to work harder at cracking them. This was the start of a race that is still being run.

The original motive of these kids, crackers, was not money. It was about testing one's skills and overcoming a challenge. What better way to do this than the secretive area of copy protection code? And every time you managed to crack one method, the opposition would soon present you with a new challenge. Another motivating factor was the chance to gain credibility with your peer group. The first one to crack a tough protection measure got respect from the other crackers. And among

those who just played games, the fame of the cracker as a good source grew.

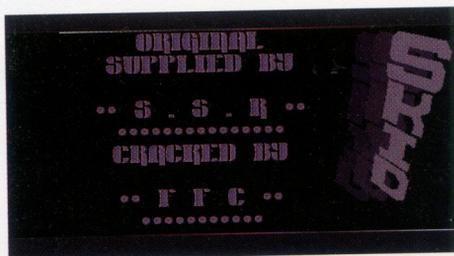
Soon the crackers had friends with extensive game collections, who, in turn, started trading games with their friends. The "distribution networks" for cracked games quickly grew from covering one group of friends to entire cities and soon countries, as well as across national boundaries. This meant that the best crackers would be known by hundreds and thousands of kids. It's worth noting that money never changed hands in the original networks. The crackers were after fame and challenges.

Early on the crackers started adding their own signature to the games they cracked, so their colleagues would be aware of who it was that got there before them and the players would know who made it all possible. They used aliases partly because their real identities were secret but also because aliases were cool. The typical text before a cracked game would be something along the lines of "Cracked by Black Hawk".

Game firms and government officials reacted to the phenomenon but could do little to prevent its growth. One of the reasons may have been their inability to understand the nature of what was happening. They thought the crackers were doing it to sell the games and concentrated their raids on small companies that tried to make money out of cracking games. These companies rarely had any real connections with the cracker scene.

The fame associated with the different cracker names and "brands" grew and pretty soon many of them weren't even playing the games they cracked, but concentrated on cracking the copy protection measures and getting the games out to the distribution network. Some concentrated on cracking only good games: "What's the point of cracking a bad

game? Who's gonna play it?" In addition to the text, crackers were starting to add graphics and sound to their "advertisements". First, they used elements taken from the game that were programmed to do the crackers bidding on the screen. Soon, they were designing their own logos and developing styles that would repeat from one "release" to the next.



## Crack Intros

The little programs or adverts attached to the games were soon named "crack intros". As the name implies, it was a piece of work displayed before the game. The typical elements included the cracker's logo, an occasionally animated text extolling his skills and speed, as well as computer graphics that moved to the beat of the accompanying background music.

Some crack intros were later developed into "trainers". In addition to removing the copy protection and borrowing game elements for their intros, crackers started experimenting changing the game itself, by inserting shortcuts that made playing easier. In conjunction with the intro, you could, for example, choose to have immortality. This would allow you to quickly play the game through. The crackers and crack intros thus became conduits for the perfect (anti)game experience. This turned up rate of demand for new games in the distribution network, since it took a day to play through most of the new cracks with their augmented capabilities. Sometimes crackers fixed bugs that the game firms had neglected

to fix, just so no one could accuse them of publishing a bad crack version.

Crackers were not the only ones bored with playing. Many of the kids that owned hundreds of games were developing an interest in the graphics and music used in their games. There was so much more there than what art or music classes at school could offer. The esthetics of games was something they could identify with. In addition to playing games, their parents really had no clue what they were doing with the computer that seemed to take up so much time. The computer classes at school had nothing to offer an advanced cracker or artist, neither in the realm of technology or esthetics, since the equipment and software used by schools was light years behind the supercharged audiovisual nature of the games.

In the beginning, the crackers made all the elements in their intros themselves or borrowed them from games, but soon they found friends who knew how to use tools for making music or graphics. Or they had programmed the tools themselves. These snippets of code and primitive programs were developed into highly specialized tools, which could be used by people who had no idea how to program a computer. They spread like wildfire among the interested and initiated, since they were often distributed on the same cassettes and disks as the cracked games.

Soon, the team responsible for cracking a game included, in addition to the programmer who cracked the game, a kid specializing in graphics and another one that made music. In addition to their personal aliases, they employed team names, like 'Evil Cracking Association' or something of that sort. The most active teams that made the best intros soon gained notoriety in the networks. There were people who started admiring the intros for their own sake. They might never even get

around to playing the game. The teams soon started competing against each other, not only in the areas of who'd crack a game first, but also by trying to make the coolest and most advanced crack intro possible. So, the artists' and the musicians' aliases were gaining fame as well.

### Share and Enjoy

Often the team included people who were engaged in actively distributing the cracked games and intros. One person might have dozens or hundreds of contacts around the country and world that he could trade games with. Preferably ones cracked by his team or contacts. These guys were called swappers, traders or couriers. The stream of kids with the blank discs and cassettes would be heading to the swappers house. Some weren't into copying games for people who weren't 'in the scene' - i.e. just played, didn't crack or make intros. Instead, they'd use the mail to trade dozens of cassettes or disks per week with their colleagues. With the help of the programmers, or by themselves, the swappers would collect the best cracks and intros onto one disk, using their own name or the name of the team, and these compilations became a sort of calling card for them.

As the competition between the different teams stiffened, notice was given to the beginning of the "production process". In the early days, games were cracked after it was bought or borrowed from a friend. In order to maximize speed and dispersion, more professional moves were needed. Going to the store on the day of the launch was not enough. People involved with the making of the games were recruited to provide a team with games before they were out for public release.

These included importers and people who wrote reviews for magazines, or anyone that had access to games before their publication. Sometimes teams would call the manufacturer and pretend to be working for a foreign games magazine and request review copies. The next level of contacts were people working at the game firm. They were the ones that were forced to hide behind an alias, so if a team had previously used their real names and included their contact information, these new developments meant that everyone in the team had to be more careful and paranoid.

The people that got the games for the team were called suppliers and they made it necessary for the game industry to start taking the phenomenon seriously. The crackers still weren't making any money for their efforts. They were satisfied with the challenge and the fame, as were some of the suppliers. Some had to be bought. Previously, the games had made it to the stores and sold quite well before they were cracked. Now, with the increased efficiency of the cracker network game sales were down, since the game might come out on the cracker networks weeks or months before its actual publication date.

Cracking became a sport based on speed. At its most intense, it was a question of hours. Back when snail mail was the primary method of distribution, this degree of precision was impossible, so victories were measured in days, but once the modem became a part of the cracker's toolbox, it was down to hours. The fastest team was able to prove that their BBS had offered the crack for download hours before the other team. Quality was important, though. If you managed to mess up the game while cracking it, your fame faded fast; in addition, the most scorn was heaped on teams that re-cracked games already done by other teams.

The stiff competition forced programmers to make a priority choice. Would they concentrate on cracking or on getting the most out of their computers by making increasingly impressive crack intros? Over the years, crack intros had developed into a rather advanced entity in their own right. They had a solid fan base. Often intros outshone the games they preceded in terms of technical and esthetic brilliance. The cracking of games was also attracting more and more attention in the law enforcement community and some of the crackers didn't really think going underground for this sort of infraction was worth it. This was especially true of the graphic artists and musicians involved in the making of the intros. They wanted more recognition for their skills, but so did many of the programmers who were employing ingenious methods to get the most out of their computers. Advertising the greatness of this team wasn't the programmer's only motivation for placing music and graphics in the intros. He also wanted to let his programming abilities shine conspicuously by having more color, more movement and more sound on screen than anyone else using the same computer. So, the programmers, along with the musicians and graphic artists, started to gain a fan base.

Teams and individuals were dividing into groups that concentrated on cracking games and ones that concentrated on making impressive intros. Originally, a team might have two different units, where one would make intros and the other would concentrate on the illegal aspects of the trade. More and more people were, however, getting into it just for the intros, both individuals and teams. Eventually, the intros were no longer distributed with the games. They gained their own distribution networks. The intros were now called emos and a new age was born.



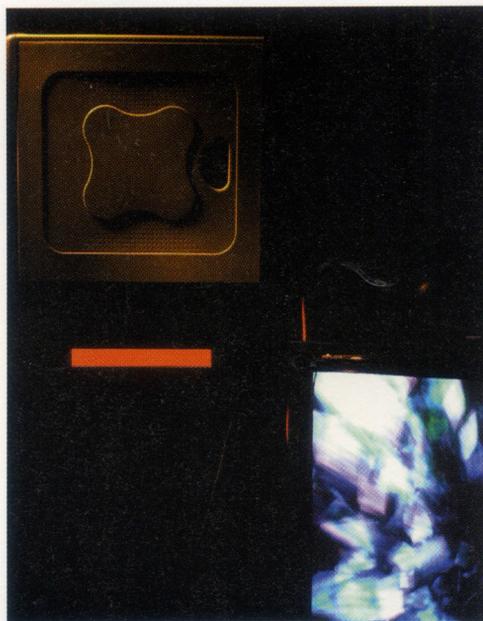
## Demos

In this context a demo is not an expression of a political opinion or the musical output of a band trying to make it out of the basement. Or even the advance version of a computer program. It refers to a stand-alone work that combines programming chops with computer graphics and music. The name "demo" comes from the word "demonstration".

The distribution network for demos was the same as the one for games. Demos were distributed on the same compilations as games and they started to interest the players, even though they had no idea what they were. The demos were skillfully made little works of art that relied on an esthetic sense familiar to all gamers. They inspired an ever-growing number of kids to check out the tools that came with the demo and game compilations or learn programming.

It's important to understand how demos differ from videos or 3D-animations. In a demo, the objects and effects you see on the screen are created in real-time, calculated and generated by the computer as you watch. They aren't played off a diskette, tape or a hard disk. The memories of the first home computers were laughably small. It wasn't large enough to hold any preprogrammed or generated animations. Each second of an animation requires many frames, each of which requires a considerable amount of memory, especially if you wanted it to look smooth and contain many colors and cover the whole screen. Besides, the same memory has to also have room for the code and the music.

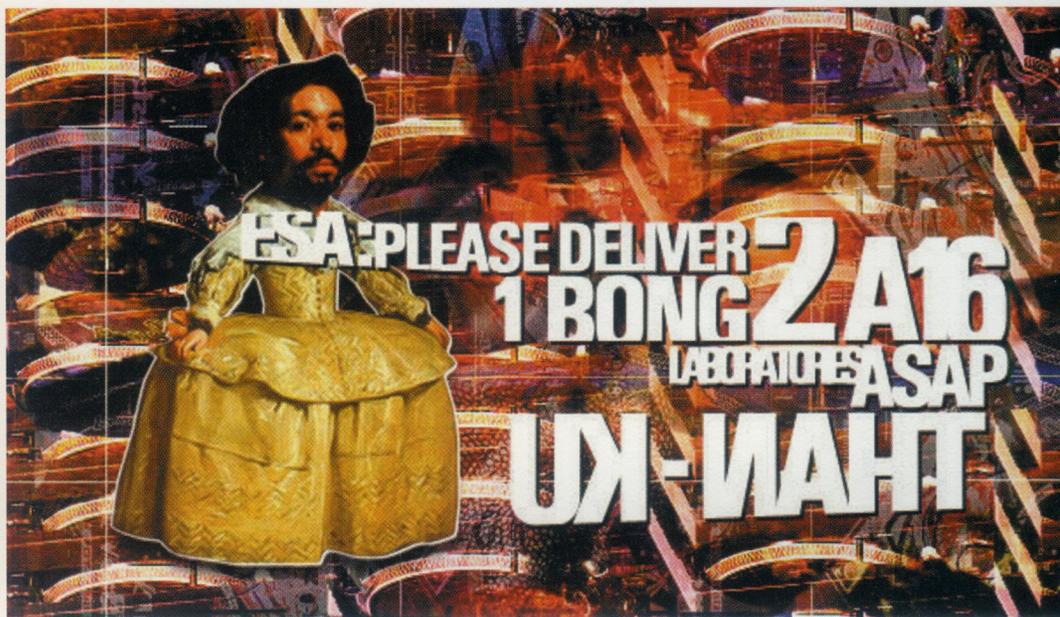
For this reason, anything with any degree of complexity to it had to be realized via code and in real-time. Movement and effects were mathematical algorithms that the computer used to calculate what kind of element was drawn on screen, how the element was



moved and shaped. In demos, everything you see on screen is drawn several times per second, so that the movement looks as smooth and impressive as possible.

The real-time principle and technology behind demos are similar to those behind games. For the player to feel that he or she is really controlling what happens in a game, things have to happen in real-time. When you yank on the joystick, the game has to react immediately. You can't play an animation. It makes its way from beginning to end, exactly the way it was filmed or animated.

The difference between games and demos is that, while in a game the player has an impact on what happens on the screen, in a demo the user is just a viewer. In a demo, all the computer's resources are used to generate, calculate and present this complex whole in real-time. Interaction would just consume vast amounts of precious resources that are best used for expanding the boundaries of what a particular computer is capable of. The makers of demos also wanted to create



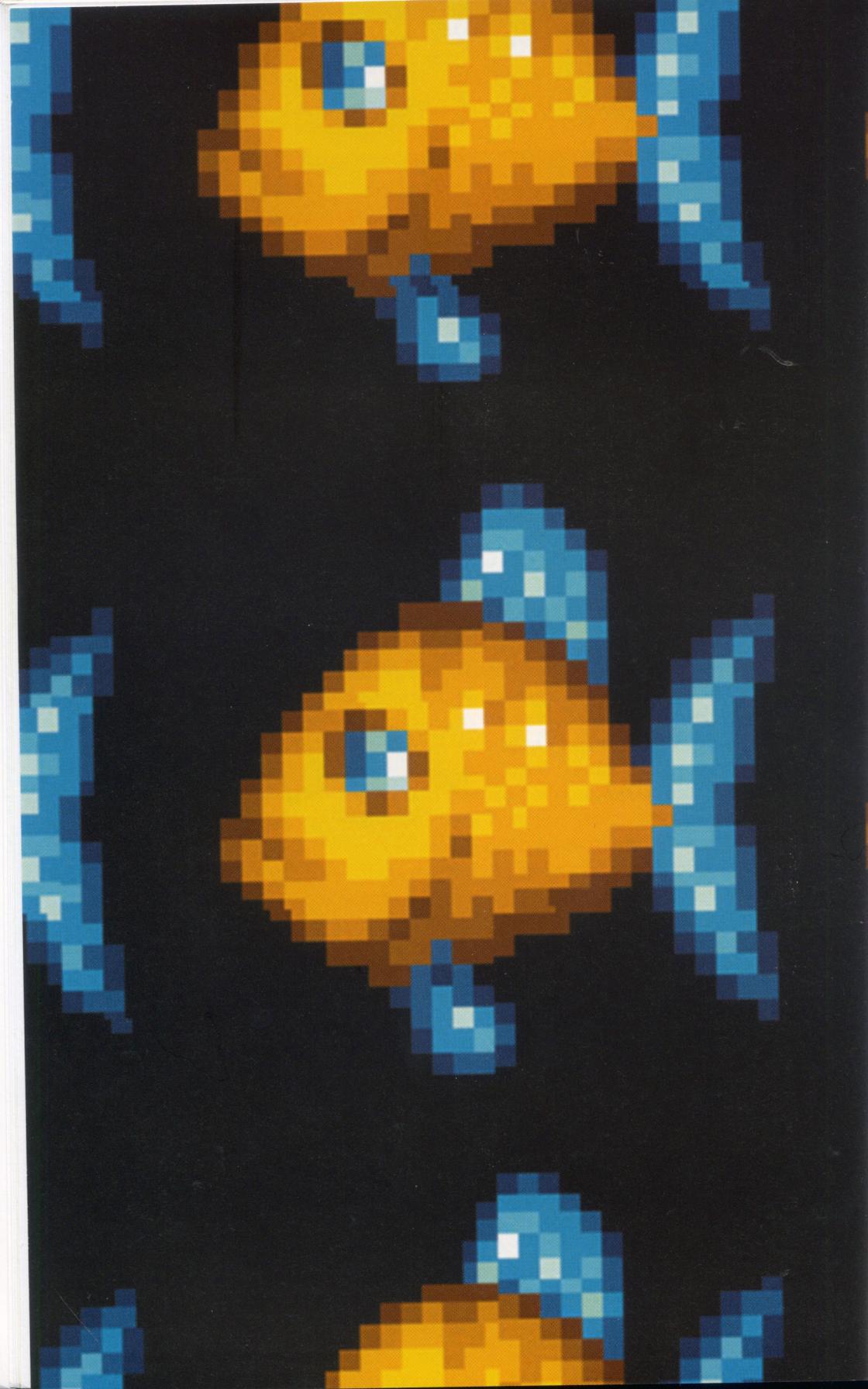
a work that they control, where they express themselves, not the viewer or the user. We are talking about a kind of "interpassive" work, which could, as far as the technology goes, contain interactive elements, but the choice not to include them has been made.

The concept of demos as a 'real-time work of art' can be clarified through the use of a more traditional comparison. When comparing the movies to theater, theater is often referred to as an interactive and direct form of art. Theater performances are live, while movies are pre-recorded. Demos are much more like theater performances than movies, videos or animations. An animation played on a computer is a movie-like, linear work that is the same every time you play it. The contents of a movie or an animation are defined when they are filmed, and the projector or computer that is used for screening the work doesn't have to expend its resources in generating it. It merely repeats it.

A theater performance (and a demo) is born in front of the viewer's eyes. Naturally, the

plot and choreography (code) is defined in advance by the writer and director (and demo programmer), and these are used to control the actors, lights and special effects (and graphic elements on screen). It also contains music. In a theater, an orchestra can be used to simultaneously play the music as the play is being performed. It's the same with demos. Someone composed the music and the computer plays the score (code), live, by generating the sounds using its sound chip or card. The strength of a theatrical performance is its actors, who always (theoretically) perform the play the same way, based on instructions given by the director. The computer's processor repeats the work the same way each time, based on the instructions given by the programmed code, since there is no user interaction.

The theater director and the demo programmer are faced with some of the same limitations. The director can't get the actors to move faster than they are physically capable of or to add more elements than they can control to their movements. A good director



or coach armed with a thorough knowledge of his or her actors and coaching methods can get the most out of the actors and produce an impressive and physically effective work. In the same way, a programmer who knows his computer and the code he uses can move elements on his stage in a more impressive manner than others.

It's important to acknowledge the difference between both forms of art and the passive movie, animation or video that lack the "it's all happening here and now" aura. The viewer should gasp at what can be done with very meager resources (processor speed and amount of memory).

For this reason demos have always been very much a programmer's art. The graphic designers and musicians are respected for how well they perform in a limited environment, but it's clear that the highest degree of creative problem solving is needed in programming in order to coax real-time media art out of primitive equipment.

The viewer has to have an understanding of the real-time nature of the works, otherwise the viewer may demand the same things from a demo that they do from a video or animation. This is a part of the reason why demos have, for a long time, been the art of a certain generation of young people, since art critics and curators lack the understanding of how computers and code work, not to mention game esthetics, necessary to truly appreciate demos.

### The demo scene

In the 1980s programmers and other kids who spent time working on their computers were usually shy and silent. Their peers thought they were odd and the time spent with computers caused either worry or amusement in the people around them. That has all changed.

Using computers and networks is seen as a right (and sometimes responsibility) inherent to every citizen. "Former" nerds like Bill Gates and Linus Torvalds are the heroes and success stories of our time.

Making demos offered kids who may have been quiet, liked math and computers and were perhaps looked down upon by others a chance to shine. No one cared how many pimples the guy behind the Black Hawk moniker had on his face. The only thing that mattered was what he could do. There were thousands of people around the world who felt how he felt and had been through the same stuff. The scene encouraged countless programmers, musicians and graphic artists to develop their skills, regardless of what the people in their immediate vicinity thought about them. The people criticizing the kids for their antisocial pastimes had no understanding of the elaborate social networks connected to the demo scene. And the networks were global. This was all before the widespread use of the Internet.

As the 1980s turned into the 1990s, the Commodore 64 had been almost totally replaced by Commodore's Amiga 500 computer as the preferred demo platform. The Amiga was challenged by the Atari ST, which had its own fans. The PC, which was primarily a computer for business, didn't really come into its own until the middle of the decade. Now, demos have been made for a number of different platforms, but it was during the reign of the Amiga 500 that the demo scene phenomenon developed into what it is today, in terms of how the different individuals and teams interact and how many people are involved, worldwide. The span of time between 1980 and the early-90s is usually referred to as the golden years or oldskool.

During the late 80s, demo teams were sprouting up in great numbers all over the world, es-

pecially in countries where home computers were popular. The groups soon became aware of each other and started trading demos. The most common way of distributing demos was on pack disks, which contained the team's swapper's contact information. This trade soon went global and became a solid network of people, now known as the "demo scene".

Daily, hundreds of demo diskettes would be making their way from point A to point B, courtesy of global snail mail. It was a new way to utilize the very old concept of pen pals. The emphasis was on trading one's own creative endeavors. The trading of new games became its own subculture after the demo-trading scene took off. A group's swapper, also known as a 'mail trader', could have hundreds of contacts all over the world and spend a considerable percentage of his allowance on postage stamps and empty diskettes. Some of the swappers became friends and, along with diskettes, the packages would convey the latest personal news.

The most hardcore swappers invested significant amounts of money in keeping the network active. The most important and active had direct contacts in all the significant cracking and demo groups, which guaranteed them access to all the newest wares (cracked games and tools) and demos. In addition to game distributors, demo swappers started competing to see who could get access to the 'O dayz old' demos.

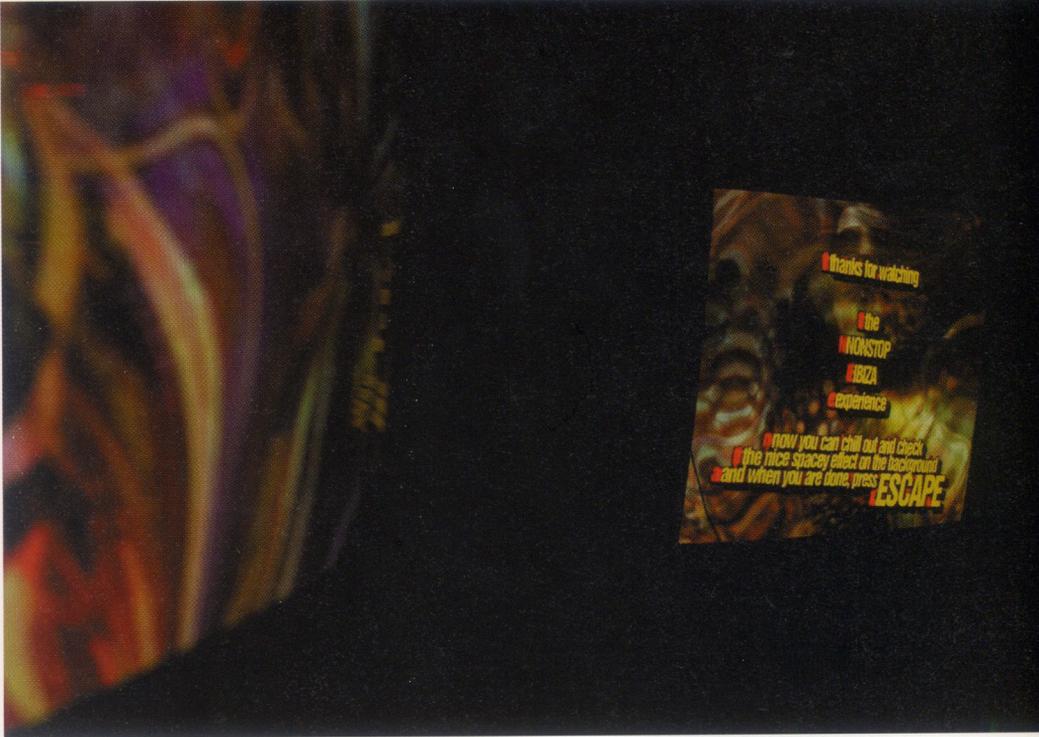
In the beginning, their own demos were an important unit of currency. A swapper who was just starting out would only have access to old games and demos, ones that had already been through several hands. There was no way you could get in touch with the elite swappers with that stuff. The only thing to do was make your own demos and compilations. Then you could be sure no one else had what you had. Making your own demos was the

best way to get contacts instead a bunch of 'no thanks' letters.

The hardcore elite swappers could request that people send them two diskettes: one they'd get back, the other would be payment to the swapper. In this way, some of the more active swappers could cover some of their costs. Another popular method was recycling stamps. All you had to do was spray a thin layer of hairspray, wax or water-soluble glue on the stamps before sending, so the post office stamp could be washed off. Swappers asked each other to return the stamps with the next package. This was especially common in overseas trades.

For the most part, the global post system was oblivious to the extent of the network. In most cases, if the stamp recyclers were caught, they could get away with it by saying they were devoted stamp collectors and just wanted the stamps back for their collection. In a worst-case scenario, it involved a visit from the police and a lecture to both the kids and their parents. Since the other party involved was usually in another country, not much was done to uncover the true extent of what was going on. The fact that the perpetrators were underage also contributed to the post offices' laissez-faire attitude. In any case, instead of getting hung up on the stamps, the cops would have been better off looking at what was inside the package, since odds are most of it was illegal. Your average policeman in the early 1990s was not informed about software piracy or the laws established to combat it.

The demo scene was a cultural phenomenon born as a result of new digital technology and the art produced using it, and the global interaction this activity resulted in among the artists. The demo scene also produced "zines", known as "diskmags" or "diskzines", with articles and criticism dealing with the

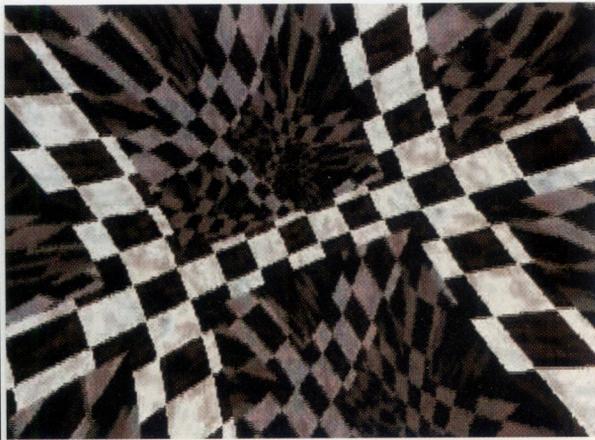


demos and their makers. The zines had an editorial board, usually a demo group. Sometimes they would have writers from several countries, writing articles about different issues. The zines contained global news about new demos, the birth of new groups or the death of old ones, fusions, people changing groups, etc. There were demo and group Top Ten-charts. Swappers were rated, too, and the zines could be used to look for new contacts. Get-togethers were written about in very colorful party reports. Some publications concentrated on publishing lists of the most popular demos and groups; these lists were known as Charts.

In the early days, diskettes were distributed via snail mail, but the new generation was quick to embrace the potential of the bulletin board systems, i.e. BBSs, in the 1980s and the Internet in the 1990s. Since the works were

digital to begin with, they were ideally suited to be distributed in this manner. So the move was made from the post carrier's leather bag to the data cable. Swappers who concentrated on distributing their wares this way were called modem traders. Every self-respecting cracking team had their own BBSes that could be accessed via a phone line. Each group typically had a World Headquarters (WHQ) and various country-specific HQs that were used to distribute wares and demos. The BBS was run by a sysop (system operator).

The early modem connections were ridiculously slow compared with today, but files were also much smaller. Many families were mystified by junior hogging the phone line for hours at a time, but not talking to anyone. Often, the pipe to the HQ might be open from late at night until early in the morning. The pro swappers had 24/7 BBSes, with ac-



cess restricted to a chosen few.

If telephone hacking or phreaking, had been an idle pastime before, used mostly to gain access to free voice or conference calls, now it made monetary sense. If you could bill your 8 hours, long distance, on someone else, you'd save a lot of money.

Naturally, the whole demo scene moved on to the Internet as soon as it was possible in the early 1990s. The Internet revolutionized both the cracking and demo scenes. The amount of traffic grew exponentially and anyone could distribute files in many different ways. Many old skool crackers and swappers feel that this is when the underground values and spirit of honorable competition disappeared. The crowd just got too big. This coincided with a series of raids that shut down many of the legendary BBSes. The rest were shut down due to lack of motivation. For demo makers and collectors, the Internet was a positive revolution, because most of the demos published here and there in the 1980s and 1990s, were available on the Internet. Individuals emptied the contents of their old hard disks and diskettes onto the data networks. Demo communities and teams beat most of the corporations in establishing their own homepages. A plethora of player and emulator programs made it possible to view

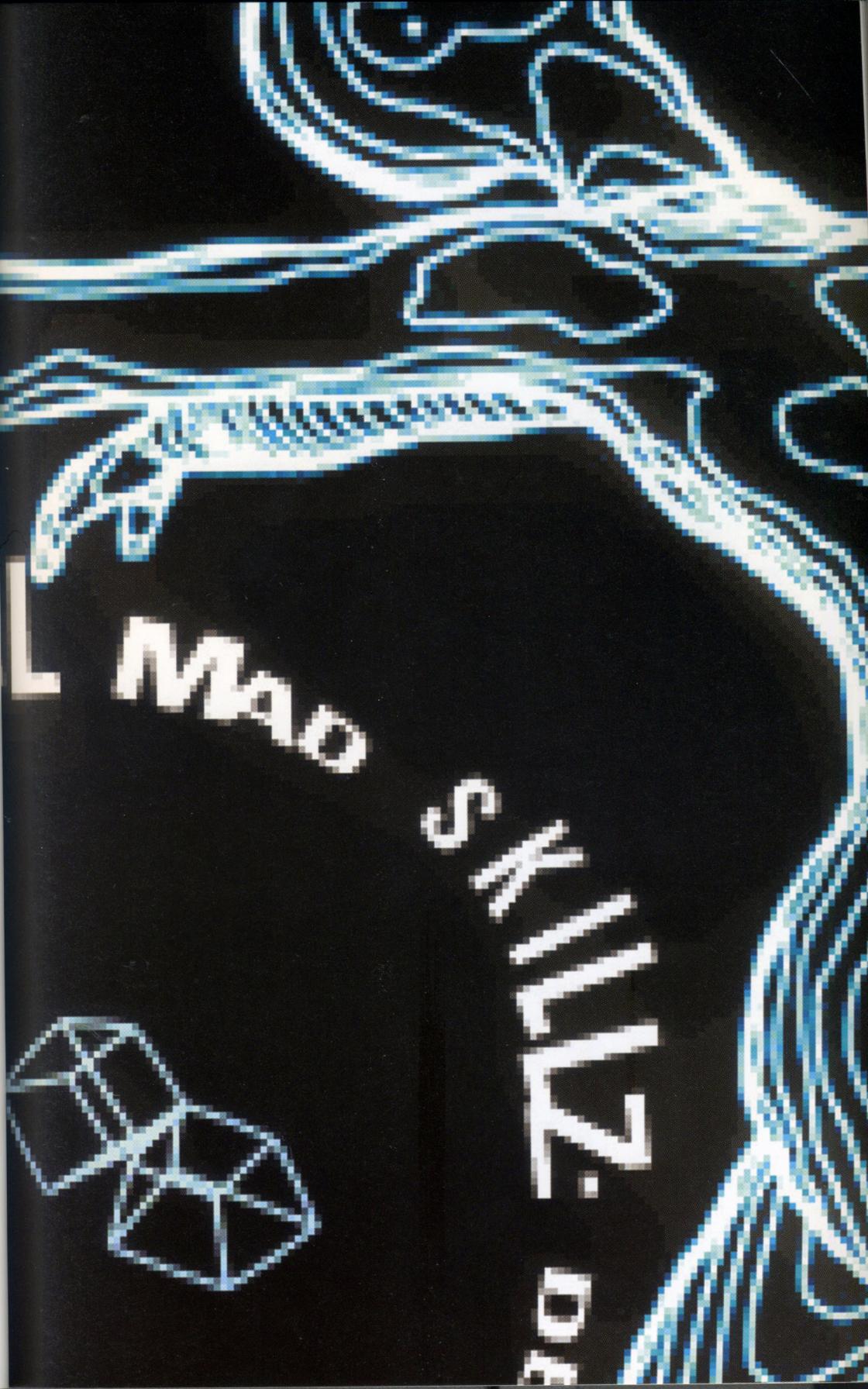
old demos on the PC.

The demo scene and its culture still unite tens of thousands of people around the world and the number of works they have produced can be counted in the hundreds of thousands. The demo scene phenomenon was strong in Europe, and especially the Nordic countries. It still is. The computers used for making demos made their biggest impact in Europe. PCs were, for a long time, too expensive and slow to compete with Commodore's 64 and Amiga machines.

Finland has always been one of the world's leading demo countries. The repercussions of this fact can still be felt in the local IT-industry. Basically, demo scenesters started all the Finnish game companies. Many new media companies and advertising agencies are full of them. The demo scene was like a school, since no education to become a digital media expert was available in the 1980s and early-1990s.

## Demos as art

In the early days of demos and home computers, no one had coined the term multimedia. Demos were clearly an early form of multimedia art, though. The makers of demos have always steered clear of the term. Generally, they've avoided any attempts to attach what they do to any existing artistic paradigm. For the people making the demos, the point is to stretch the capabilities of the computer and to look for new ways of presenting ever more complex works. The technological challenge, competing with your peers, plays a central role in developing their own form of expression. The demo is not only a work, but also a demonstration of the makers' artistic and technical skills. The aim is to outdo others.

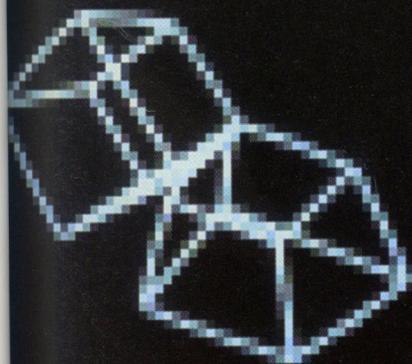


L

Mao

S K I N N E R

DO



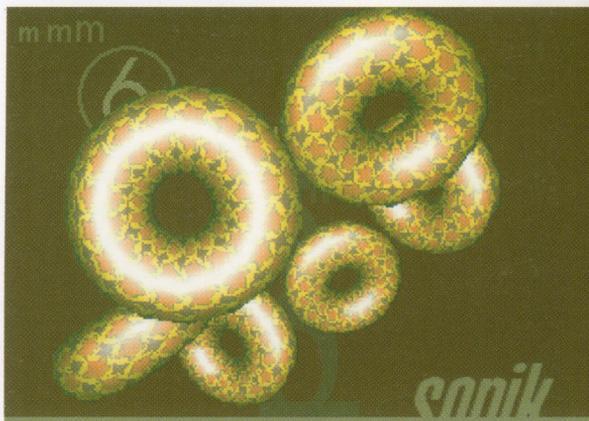
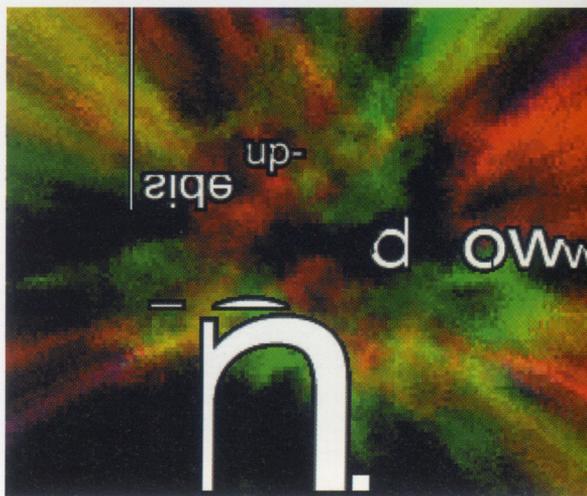


In the same way crackers race to be the first to release a cracked game, demo groups compete to be the first to use a particular effect or routine (preferably something nobody has thought of before). Making a demo may take anywhere from a few days to as long as a year. Even though technical prowess is the point in competing with demos, various cheats and 'magic tricks' are allowed. In many demos, a particular implementation is not as advanced as is claimed, or it is only partially implemented. This is allowed, as long as the cheat isn't immediately apparent. The point of a demo is to make the viewer speechless. After a period of speechlessness, the question that comes to mind ought to be "How IS that done?" By combining a suitable number of

magic tricks and true programming prowess, you can generate controversy while remaining credible.

Sometimes groups will improve on their demos and release better versions later, especially if there is a particularly bad bug in the code. This practice was not as widespread in the pre-Internet days as it is now. Back then once you released a version, it was out of your hands and in the mail. Some crack teams would take buggy demos, fix them and re-release them, complete with sarcastic greetings directed at the original group.

No one knows who made the first demo or intro. The question has generated a lot of discussion, but most of it ends up being arguments about whether it was made for a Commodore 64 or should we count the text greetings or little adverts made for other platforms. Or even the short little programs made for the first generation of programmable calculators. It's likely that as soon as the first computer that even vaguely resembled what we now consider a home computer came out of the box, someone was trying to make it do things completely outside of its normal boundaries. And the next logical step is to see who can make it do the coolest things. Generally, the first scene that fulfills the criteria given to that word these days, would have been the piracy scene that grew around the Apple II computer in the late-70s.



[

A.98



VERY, VERY, VERY RED-EYE

## Categories and Styles

Demos can be divided into several categories. This was especially the case in the early days. The general tradition of the early crack intros (also known as 'cracktros') was continued by what are known as just intros, which were a kind of minidemo. The structure was simple and they were short. They were primarily used for communications, like advertising the group BBS, changes in the group's membership or call-outs to other groups to attend a get-together (invitation intro).

Most demos were single part demos. As you can tell by the name, these demos had one part. The contents consisted on variations on a series of themes and effects. In addition to the effects, the demo might contain dozens of minutes worth of scrolling text.

Megademos were ambitious works,,

often spread over a couple of diskettes. They consisted of several parts, each of which might have different coders, graphic artists and musicians working on it. As a result, the styles of the different parts could vary quite dramatically.

The beginning of the 1990s saw the emergence of the trackmo demo. The structure was no longer based on separate parts. Instead, the program would be loading the next part as the previous one played. The term "trackmo" comes from the fact that they were continuously loading and running "track by track". The parts were mixed together, either audio visually or through a plotline. Trackmos demos loaded and worked without an operating system, so they used all the available space on a disk (or several) and couldn't be copied onto compilations. They had to be dis-

tributed as they were, individually.

When the classic intro - single part - mega-demo division was broken, a spate of subcategories was born in rapid succession. In addition to trackmos, there were e.g. dentros (half demo, half intro), multros (multi part intros) etc. As in the early days, musicians still collected their pieces onto special music-discs and graphic artists published slide shows of their work.

In the first 1980s demos, the standard elements were the group's logo, a scroller text, various small, animated graphic elements called 'sprites' and 'bobs', color bars called 'copper bars' and 'equalizers' that moved to the pulse of the music. Other popular elements that came along later included fractal graphics and different 'plasma' effects, which were multicolored, animated color surfaces that filled most of the screen. As demos developed, the complexity, amount, colorfulness and mobility of the elements increased as a result of constant competition. Texts started to run in different directions on screen and various tricks to make 2D elements look or behave like 3D elements were employed.

The basic nature of the demos started to change in the early-90s. It became more like design and the influence of traditional graphic design and art became more pronounced. More and more demos had a person involved who chose designer as their title. He was something between a director and a graphic artist. Earlier, it was the coder that took a holistic view of the project and realized it, with help from a graphic artist. Now, the designer had more input into the basic structure of the demo. At the same time, the real-time effects were being augmented with short bits of animation. This was usually the case with demos where human figures were seen dancing to mobility of the elements increased as a added in. Emulating effects like color and

noise filters further emphasized the music video esthetic. At the same time, a higher degree of synchronization and a deeper relationship between the music and the effects was sought. Since the music was real-time, having a certain effect show up using a trigger placed in the code for a specific instrument (or any part of the music code really) was logical. Certain groups concentrated on this side of their demos and became known for it.

The 1990s saw explosive development in the field of vector graphics. In the demo scene, this resulted in an almost weekly unveiling of ever more complex 3D objects and tricks. It all started with the simple wire frame vector objects of the 80s: cubes, pyramids, etc. These were then filled and textured. The objects and textures became more and more complex and started to react to their surroundings in different ways, like generating reflections. Basically, just about every effect offered by image processing or 3D modeling software was done individually and in real-time. It's worth keeping in mind the fact that the computer has to calculate 50-70 images per second on screen in an average demo. A 3d program may spend 50-70 minutes, or even couple of days, rendering ONE image.

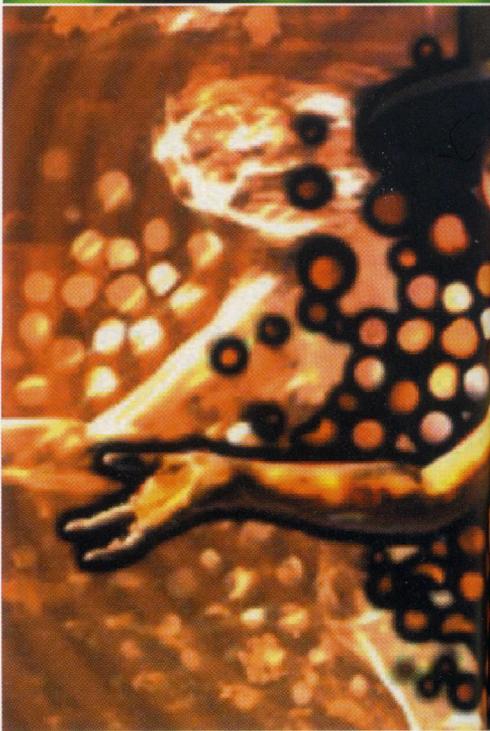
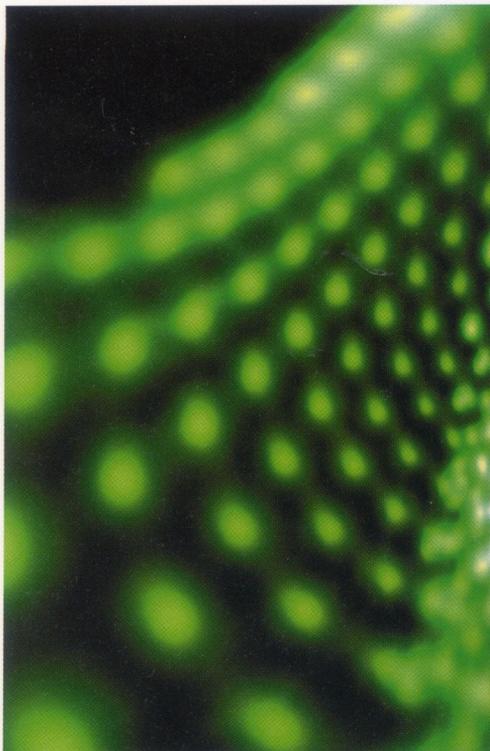
As the demos became more and more cinematic, one level of communication in the demo scene was lost. Since the first cracks, the demo screens had contained scroll texts that were used by groups to communicate with each other by exchanging greetings, news, contact information, pleasantries, as well as tell stories to hype themselves up and relate details of epic drunken parties. An integral part of the texts were also the credits (who made the intro or demo) and the greetings, where the group gave shout outs to all the other groups they traded with, respected or just wanted to be pals with.

The primary aim of every group starting out was to get as many cool, established groups on their greetings list as possible. Even more important was getting your own name on the lists of these groups. The road to the greetings list for a novice group went through their swapper contacts. They were the ones that kept the greetings lists up-to-date.

The scroll texts played an important part in how the whole culture developed. They were the birthing place for the legends of the early days. This was how information was disseminated before the diskette zines or Internet. The various track and design demos were built on a series of rapidly changing scenes, so a stream of text lasting for several minutes would have been incongruous. What were left was the credits, greetings, as well as the occasional message or inside joke that was briefly flashed on screen.

### Art Program[ming]

Generally speaking, early demos combined traditional esthetic values, new technology, and popular culture. Parents or teachers very rarely had any idea what the kids were doing with their computers. From this point of view, making demos could be considered an autonomous phenomenon. What the people in the scene considered good or bad art had little to do with traditional techniques or theories. Demos were influenced by the esthetics of sci-fi, heavy metal record covers, MTV, advertisements, fantasy and pornography, or in other words, everything that fascinated the average 10-20 year old male. It's also important to remember that there were no ready-made programs for producing music, graphics or multimedia available for the first generation of home computers. The kids programmed the tools themselves, to fulfill their particular needs. The programmer in question might have been ten years old, who'd programmed his own app, because he



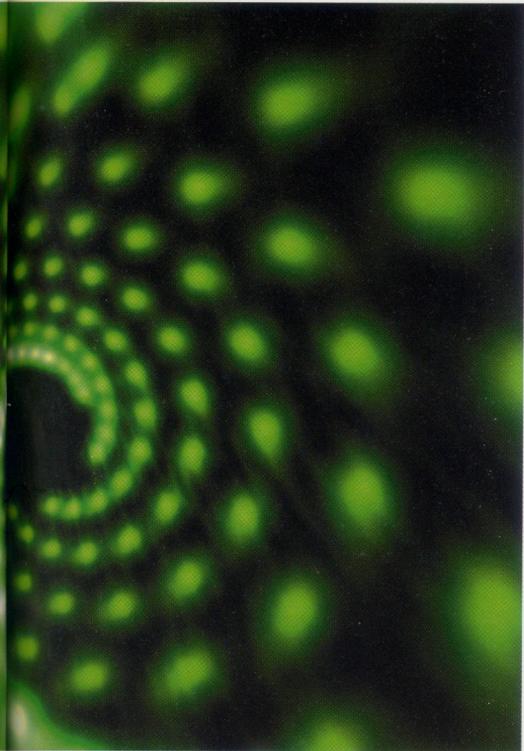
just couldn't find anything that did what he wanted to do.

The most respected demos were the ones where creative problem solving was apparent in the programming. Programming a, for example, Commodore 64 was different from what is construed as programming these days. Now we have access to advanced programming languages that make it easier. Current computers have also been designed, as far as power and parts go, to deal with heavy-duty multimedia programs, with their separate 3D cards and graphic libraries.

As a result, the older and newer generations of demo makers tend to argue about which is more important: programming chops or graphic design. The best demos are brilliant in both areas, it's just that today it's much easier to make impressive demos without any programming skills. The older generation tends to call for more technical creativity in today's demos, instead of relying on audio-visual tricks alone. The point, according to them, should be to challenge what the computer is capable of.

One of the cardinal unwritten rules of the demo scene was that the best demos should always look and sound better than the best games out at the same time. This is one of the reasons why, after DOOM for PC was published, Amiga coders spent an insane amount of time and effort trying to produce similar 3D mazes on their computers. An added incentive was the fact that making something like a 3D maze on an Amiga was incredibly difficult. This made for an interesting challenge and a source of much competition.

Now that we've reached the first years of the 21st century, demos and 3D engines are so close to each other, it's hard to tell who is leading the game. Some say that the game companies pretty much took the lead already





in the end of the 1990s. Programming a rugged 3D demo engine functions as a calling card for many programmers who've set their sights on the game industry as an employer.

Getting started with 3D programming is fairly easy these days. Current languages and tools, not to mention the powerful computers almost everyone has access to, make the threshold a lot lower than it was in the very early days, when getting the necessary power out of the computer meant hacking right into machinery. In other words, the kids wrote hexadecimal code, line by line, into the computer's memory. There was no source code to augment or tweak. Coding required a flexible mindset, as the process itself was not flexible at all. After that most of the demos were written in assembler and there was source code. Nowadays demo coders use common sophisticated programming languages.

It was the same with the early musicians and graphic artists. They had to use joysticks or the keyboard to change parameters in the code and then play it back or look at it to see what happened. Some of them had to change the code itself and recompile. There was no way you could sketch or jam. This way of working helped the makers understand the principles the operation of the computer was based on from a variety of viewpoints and make economical art using a small amount of memory. The impact was based on just a few colors or a couple of sounds.

## The Clocksmiths of Real-Time

Even though comparing demos to movies should be avoided (so there is no misunderstanding between real-time and pre-calculated), it's still interesting to look at the similarities in their development.

The fact that demos were distributed via mail, straight from one artist to another, bears more than a passing resemblance the underground video culture of a couple of different decades, or the spread of DJ mixes and remixes. There are comparisons to be drawn in the development of demos and movies, too. Both started with a single stationary camera aimed at a static stage that contained mobile elements. Little by little, movement is increased, ending up with an active and mobile camera, with the structure of the storytelling changing from a series of separate scenes to a more fluid form that tries to control, in addition to the different elements on screen, time and the viewer's experience on an expanding sensory and emotional scale.

The fact that most of the demo makers use aliases and are recognizable, mostly to other demo makers, by their styles makes demos similar to graffiti culture, and demos have been referred to as digital graffiti. The early crack intros can be compared to tags, the quick-and-dirty signatures drawn by graffiti artists in as many public places as possible. They're both a way of marking territory and making your alias famous. Many demo scene graphic artists used to paint trains, too, and there are clear similarities between the logos of some demo teams and graffiti styles and techniques.

In many ways, the demo scene was a precursor to the online communities that get talked about in conjunction with the Internet. The demo scene operated according to many of the same principles later adopted by virtual

communities and in the realm of open source work. For example, 'ripping', or taking credit for the work of someone else in the scene, was frowned upon and the punishment would be ostracism from the community and a reputation as a 'lamer'. One's reputation in the community was of the utmost importance. The software and demos were also made in a distributed manner, often in different parts of the world. So, the similarity between the demo scene and, for example, Linux and communal art is easy to see.

Demos are works of art. The people who make them (programmers included) are artists. There is no way to question this. Looking into the makers' perceptions of their actions and artistic personas is interesting, though. Many of them disavow the conceptual burdens that come with taking part in a discourse on art and the recognition offered by the art world. For them gaining recognition within their own community is more important. Naturally, the urge to dazzle the general audience, on both a technical and esthetic level, is always present, but other demo makers are often the only people capable of understanding the innovative nature of the works without an in-depth knowledge of programming, as well as the limitations of the hardware used. Thus, encouragement and criticism from one's peers and respected members of the community is a lot more important than widespread fame.

Demo art rarely "says" anything or comments on society. Still, even though the makers insist that demos are made "for the hell of it", they are commenting on society via technology and software just by engaging in the act of producing demos. The indirect, and sometimes direct, societal criticism that demos offer pertains to the uses and rationales of the chosen computers, operating systems, tools and distribution channels. Information technology is no longer separate from society or a

pile of nerdy toys. Some very important parts of society are built on code these days. In this environment, the people who are familiar with the history and operational principles of the technologies and software, as well as their potential for creative use, are the ones capable of making real political and artistic choices and statements.

A clear statement in the demo scene is the careful nurturing of innovative technical implementations and quality criteria for digital esthetics, as well as criticism based on the aforementioned. In addition to demos, it applies to computer games, mobile applications, digital media and information systems. The programmers, artists and musicians instill the values of the demo culture in the products and services of the information society and communication culture.

Demo artists are meticulous about the quality of their output, from the almost invisible little details to the overall concept. The work has to take into account the reality of both the technology used and the user community. Demo makers are probably closer to artisans than artists. Many artisans combine tradition-rich and strict quality standards with esthetic values. A clocksmith wants to create a watch that looks unique, and keeps better time than any other. A satisfied user is a rewarding experience for a clocksmith, but his professional and artistic identity depends on what other clocksmiths see in his clockworks and design. Only the community of clocksmiths can appreciate the clockworks, the methods and materials used to create it, as well as the challenges faced. And only the community can bestow true master status on the clocksmith, not the audience or society-at-large.

The people who make demos are not so much wandering conceptual artists, but the master clocksmiths of real-time digital media art.