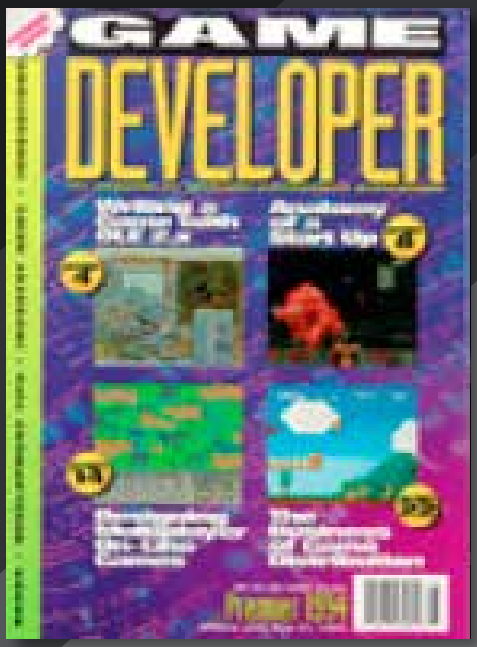
gd

GAME DEVELOPER MAGAZINE

PREMIER 1994



UBM Tech

The Dream

I have this idea about every time I play Wolfenstein. I dream about going back to the press conference where the 386 was unveiled. As a time traveler from 1994, I run past the crowds, install Wolfenstein on the sole 386 computer in the world, and start it up. The crowd oohs and ahhs as I blow away the first guard, doubtlessly dazzled by the digitized speech from the sound card I installed the night before.

I turn around and seize the microphone from the dumbfounded speaker and proclaim: "You see this world? With the proper tools, good programming, great artwork, and competent project management, a small group of people can assemble a game of this quality that will run on this processor perfectly. So, now the challenge is on for game developers to surpass this game and show the world what can be done!"

Developers of Today

But, alas, it is only a dream. The state of the art in game development today is still the state of the art, when it should be closer to the norm. Ever since the first quarter fell through the slot of the first Pong machine, the industry has been shrouded in secrecy. Has the paranoia that enveloped the electronic game industry on behalf of keeping a competitive edge served the industry well? No.

Do the game developers of today, who arguably have the some of the hardest jobs in programming, have the best tools at their disposal? Do game developers not have to worry if the system they're developing for will be viable in six months? Are the companies making tomorrow's computer operating systems consulting game companies to make sure their needs are met? And lastly, do game programmers, who have to port from the arcade to the Super Nintendo Entertainment System to Sega's Game Gear to DOS to myriad consoles, portables, PCs, and arcade machines, have the best porting tools? These questions are what I base my answer on.

For every owner of an Atari 7800, for every person who buys a cool DOS game only to find out that he or she doesn't have 610K of conventional memory, for every SNES owner who wants to play Sonic the Hedgehog II, there's one more person who is either tuned out, turned off, or has spent his or her way out of the electronic game market. Clearly, the electronic game industry isn't exactly in a state of ruin, but issues such as quality control and over-segmented markets need to be examined. That's where we come in.

Game Developer is written for developers, prospective developers, and interested spectators and discusses the issues of code, commerce, and creativity. As far as code goes, our technical articles will help novices figure it out and help the professionals do it better. Commerce topics will cover the latest and greatest in computer games, what went right for a company, where it went wrong, and why it doesn't do it anymore. Creativity will focus on the artists of the electronic entertainment industry from the graphic artists who make game images to the musicians who write the scores.

Hey Out There!

That's where we stand, take a look at this issue and tell us what you think. We've tried to cover as much ground as possible as far as the range of subject matter and level of information. Let us know who you are, what you think, and anything else that comes to mind.

You can contact me at (415) 905-2349, on Internet at 71154.676@compuserve.com, or you can write to us at Game Developer, 600 Harrison St., San Francisco, CA 94107. ■

Alexander Antoniades
Associate Editor

GAME DEVELOPER

Editor Larry O'Brien
Associate Editor Alexander Antoniades
Production Editor Nicole Freeman
Editorial Assistant Andrea Pucky

Publisher Veronica Costanza
Group Publisher Regina Starr Ridley
Group Director Donald Pazour

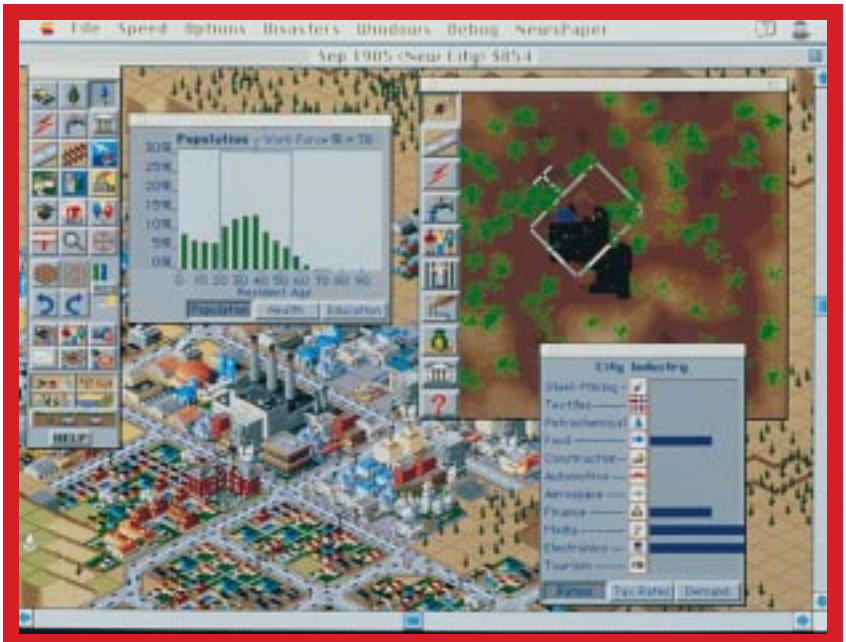
Advertising Sales Staff
New England/Midwest
Holly Meintzer (212) 626-2275
Middle Atlantic/Southeast
Michele Blake (212) 626-2322
Dave Moreau (212) 626-2318
West/Southwest
Michele Anet (415) 905-2298
Pam Grossman (415) 905-2256
Yvonne Labat (415) 905-2353

Marketing Manager Susan McDonald
Art Director/Marketing Christopher H. Clarke
Advertising Production Coordinator Denise Temple
Director of Production Andrew A. Mickus
Vice President/Circulation Jerry M. Okabe
Group Circulation Director John Rockwell
Circulation Manager John Tootill
Circulation Assistant Stephanie Blake
Newsstand Manager Pam Santoro
Reprints Andrea Varni (415) 905-2552

Miller Freeman
A MEMBER OF THE UNITED NEWSPAPERS GROUP

Chairman of the Board Graham J.S. Wilson
President/CEO Marshall W. Freeman
Senior Vice Presidents Thomas L. Kemp, H. Vern Packer, Wini D. Ragus
Vice Presidents Charles H. Benz, Vicki L. Masseria, Andrew A. Mickus, Jerry M. Okabe, Donald A. Pazour, Charles L. Wrye
Chief Financial Officer Warren (Andy) Ambrose

Let's Go Embed



Jeff Braun, president of Maxis, the company responsible for SimCity, believes games should be portable to all systems—consoles, PCs, even TVs. Radical though this vision might seem, SimCity has reached the 1 million mark in sales worldwide. Who knows, maybe this interoperability idea is a good one. Braun seems to think so, and other game developers are beginning to believe he is right.

Jeff Braun, president of Maxis, wants to be your partner. He wants to help you become rich, and he wants to help you sell games. He envisions shelves stocked with your work. There's only one small hitch; he wants them all to read "A SimCity Add-On." According to Braun, high-end console machines are dead coming out of the starting gate. "Where they have potential is as TV-top machines for interactive television," he says. But that's years in the future. In the meantime, he says, "More PCs are sold into the home in a month than 3DO dreams of selling in a year." In a world where installed base means everything, Braun sees nothing coming to dethrone the PC as the smartest target system. Additionally, the dream of partnering with small game developers cannot happen without the capabilities of a real operating system. As a matter of fact, it cannot happen without some advanced operating system capabilities that are only now becoming available on PCs.

Braun wants to open up SimCity to other programs. People who play one game are likely to play another, and Braun wants to give them as many chances as possible to interact with Maxis products, even when they're playing another company's game. Braun envisions SimCity as a virtual world in which gamers will have their choice of myriad recreational and simulation diversions. You think people want to be SimCops, SimArchitects, or SimCar-Racers? SimCity will create the environment and economic conditions, you provide the specialized functions. That's Jeff Braun's dream, and you can't do it on a console.

The Path to Interoperability

Although Braun's vision is radical for a game company, it's rapidly becoming the common wisdom in the business software community, which calls it "document-centered computing" or "component-based development." In many ways, it makes more sense for game software, with its broad-based demographics, than for business software that is dominated by only three dominant

applications (word processing, databases, and spreadsheets). If Maxis can't pull it off, someone else will.

There are three steps along the way to total interoperability:

- *Static data exchange.* Data formats are opened up. Maxis has already done this with Mallard and Virtus, which will allow you to fly and walk, respectively, through your SimCity.
- *Limited dynamic interface.* No console has the ability to run multiple processes simultaneously, but all the popular desktop operating systems (with the important exception of DOS) have some multiprocessing and interprocess communication ability. With this level of interface, gamers would be able to interact dynamically with an active and evolving SimCity, but would not be able to, for instance, dynamically change the bitmaps used in SimCity.
- *Full programmatic interface.* This is the big enchilada, a situation in which the company creating the "server application" (the gaming world) creates two entire interfaces for the program, one for the gamer and one for the add-on programmer. "Client applications" (the add-ons) are able to interact with the server application, passing data and commands back and forth while both processes are running. If your space-race program drops a Skylab on your city, your SimCity window can show the burning chunks raining down and slamming into the dome of your nuclear power plant. (Hmm.... SimStrategicDefenseInitiative, anyone?)

Obviously, the first step requires little more from the game company than a disciplined attitude toward data storage and some documentation. The second step requires a more sophisticated interprocess communication facility that's generally provided by the operating system. Traditionally, the performance requirements of games have required developers to bypass high-level languages and operating-system features. Now that the average player's PC is based on a 32-bit microprocessor with 4MB or more of RAM and ample disk storage, and game developers are able to program in higher-level languages such as C++, it is feasible for games to move to this second step through the use of such technologies as dynamic data exchange.

The third step requires an even more sophisticated interprocess capability, whereby data elements in one program can be exposed to modification by another program. This is one of the goals of so-called distributed object orientation. There are a number of technologies that could be used, but Microsoft's Object Linking and Embedding (OLE) 2.x is the most likely one.

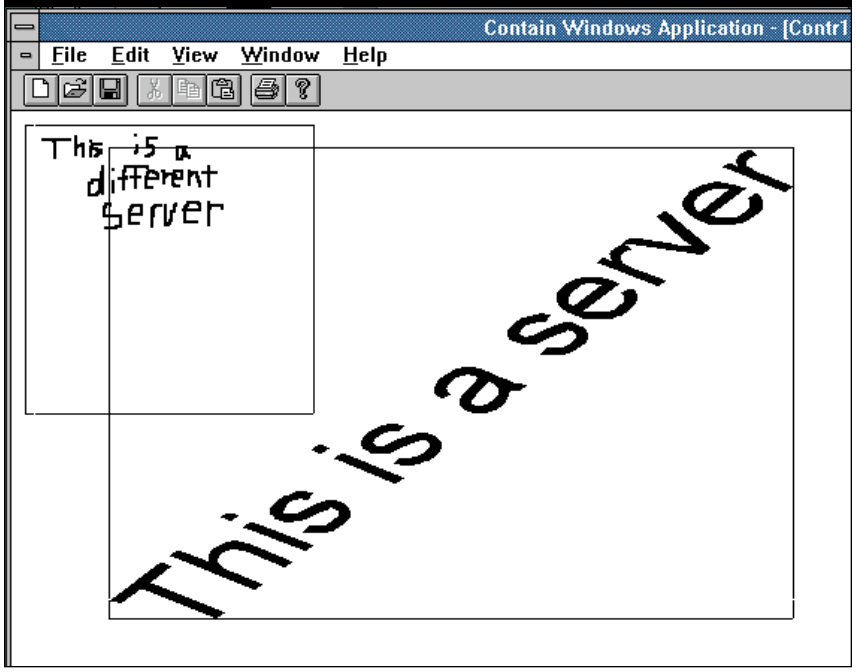
The World of OLE

This is the point where this article veers into total speculation. When I asked the people at Maxis to name the technology they would be using, they rolled their eyes and gave me a look I interpreted as "we haven't made up our minds, and we're under nondisclosure about the things we're thinking about." Since I am speculating, from now on I'll use as my example a theoretical open gaming environment.

by Larry O'Brien

Developing a game portable across many different systems is a daunting if not impossible task. With the help of high-level languages and OLE 2.x, devlopers now have a new window of opportunity.

Figure 1. A Simple Container Application



OLE 2.x is likely to be the dominant technology used for open gaming environments because the gaming industry is so sensitive to installed bases. Microsoft's technology is complex and, according to some, poorly implemented. Nonetheless, it's difficult to see how competing technologies from IBM, Apple, or Sun are going to convince any game developers that they represent viable alternatives. OLE 2.x and the C++ class libraries needed to handle it are Microsoft's latest strategic weapons in the battle for operating-system dominance.

We had an opportunity to work with a beta version of Microsoft Foundation Classes that supports OLE 2.01. This version of MFC, labeled 2.5, should be available in an interim release of Microsoft's Visual C++ package by the time you read this. The rest of this article is based on a beta copy of the compiler package. Although it is a late beta copy, anything could change by shipping time.

Although I'm generally opposed to writing about a product that hasn't shipped yet, this is the only class library that I'm aware of that supports OLE 2.x, and it's my strong belief that a class library is necessary to insulate the programmer from as much of OLE's complexities as possible. I'll try to make clear which features are part of OLE, which are part of MFC, and which are part of Visual C++.

There are four types of OLE programs:

- Containers
- Servers
- Automation servers
- Automation clients.

The naming is unfortunate, as it reflects two uses of the word "server." A container holds frames in which servers are active (if you embed a graph in a word-processing document, the word processor is the container application, and the graphing program is the server), while an automation server is programmatically driven by an automation client. So an open gaming environment would be both a container (so new graphic elements could be added) and an automation server (so, for instance, new AI routines could be plugged in at will). A gaming add-on would probably be both a server (for display) and an automation client (for logic).

Listing 1. An Example Dispatch Map

```
BEGIN_DISPATCH_MAP(CGamedevDoc, COleServerDoc)
   //{{AFX_DISPATCH_MAP(CGamedevDoc)
   DISP_FUNCTION(CGamedevDoc, "Destroy Building", DestroyBuilding, VT_BOOL, VTS_I4)
   //}}AFX_DISPATCH_MAP
END_DISPATCH_MAP()
```

OLE 2.x is likely to become the dominant technology used for open gaming environments.

Containers and Servers

Containers and servers are probably what you think of when you think of OLE. Figure 1 shows a simple container application that contains two embedded server frames. One server application is a text-manipulation program, the other is a simple drawing application. The frames of the servers overlap without a problem, and the frame of the drawing server overlaps the client area of the container and the frame of the other server.

In this case, a border is drawn around the servers; but in an open gaming environment, this would probably

not be the case, and the servers would be able to use standard Windows drawing modes to paint themselves over the background. However, servers must be rectangular, and hit detection of overlapping servers is based on a simplistic Z ordering (the last embedded object receives the command).

There are a few more aspects of containers and servers that are especially important to an open gaming environment:

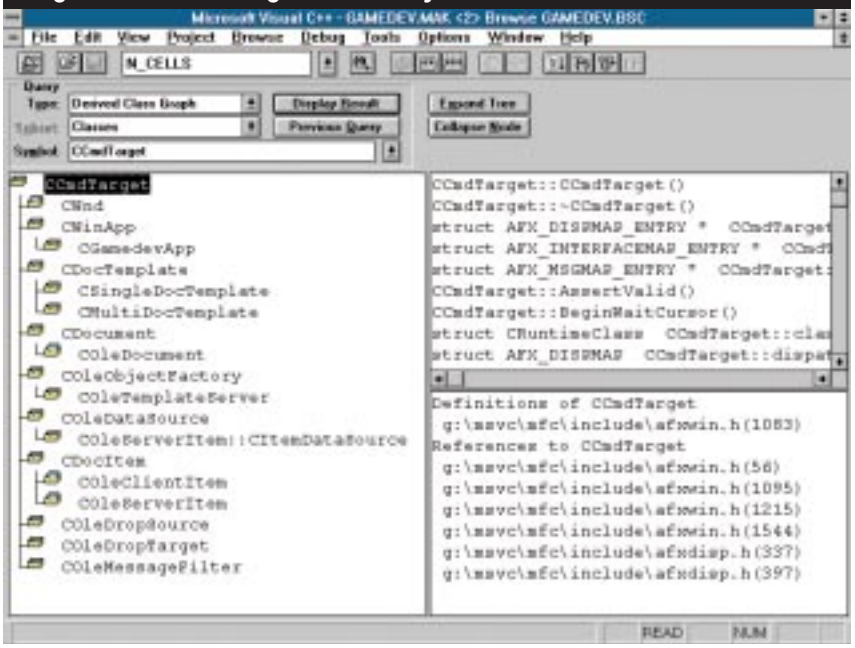
- A container and server can negotiate the display size of the embedded server. So, servers can be automatically placed and scaled within containers without excessive code.
- A server receives different Windows messages when it is activated in place than when it receives the focus as a fully opened independent application.

The really exciting capability of OLE 2.x is OLE automation. An automation server (remember, that's different than a regular OLE server) creates one or more of what are called "dispatch interfaces." This is an interface for other programmers and is the most exciting thing to happen to programming since Bjarne put the PP into C. On the other hand, like the rest of OLE, it's Yet Another Microsoft Standard, which are notorious for being not-as-portable and not-as-stable as promised.

In MFC, classes derived from `CCmdTarget` can place functions in the dispatch interface simply by calling an `EnableAutomation()` member function in their constructor via a dispatch map that looks almost identical to a standard MFC message map. Figure 2 shows the hierarchy for `CCmdTarget`. Listing 1 shows an example dispatch map. Data can be directly exposed, although this is bad programming form, since you're relying on add-on vendors to validate the data before calling your function. When they don't and a crash occurs, it will look like your program is at fault.

There are three categories of functions that would be exposed through a gaming environment's dispatch table:

Figure 2. CCmdTarget Hierarchy



- *Informational.* Every informative piece of data presented to the user by way of the graphical user interface should also be available to automation clients.
- *Programmatic Interface.* Every action available to a user through the graphical user interface should also be available to automation clients.
- *Modifications.* One of the more interesting possibilities of OLE automation is add-on automation clients that alter internal behavior of the automation server. In a gaming environment, the obvious use for this would be to alter the playability-realism-difficulty trade-offs that every game has.

Automation clients use a text-based command language to control automation servers. All commands follow the basic form `AppName.Document.Verb {Parameter1 Parameter2 ...}`. Microsoft is leading the way by providing automation clients. The latest versions of Visual Basic, Excel, and Word are all automation clients (obviously, Visual Basic is the most versatile, although the latest version of Excel has Visual Basic for Applications and is only slightly less useful).

Linking to the Future

If OLE captures the hearts and minds of the game development community, it would mean sweeping changes in the entertainment marketplace. Large companies such as Maxis, Electronic Arts, and LucasFilm would primarily work on creating worlds that were robust containers and automation servers, while smaller companies would concentrate on adding value in the form of servers and automation clients. Microsoft would have enormous leverage in the battle for the home operating system (surprise, surprise).

Although Maxis is the first game company to embrace this idea, the advantages in terms of time-to-market and saved effort are making an impact already in the field of business software development. Compared to business programs, entertainment programs have shorter lifespans, broader demographics, and lower profit margins. All these factors combine to make OLE an appealing embed mate. ■

Larry O'Brien is the editor of Game Developer, Software Development, *and* AI Expert *magazines.*

Designing On-Line, Multiplayer Games

Feudal, an on-line game available on CompuServe, is a great example of the work involved when transforming a single-player or board game into a highly interactive, multiplayer on-line game. Here, you can learn some of the pros and cons of programming an on-line game.

My first on-line game began as an in-house corporate game in 1980. Our company had an enlightened attitude and encouraged us to use spare computer time to experiment with new ideas and improve programming skills. With this time, I created a medieval wargame called Feudal. The units were chess pieces traversing a large terrain map, trying to capture other pieces and opposing kings. Up to 20 players could play in one game over a period of weeks. You could examine the map, build new units, and give orders for the day, then issue the update order to execute the orders you had given.

During one lunchtime test, we lowered the update time to one minute, and the blitz-Feudal (and my first truly interactive) on-line game was born. The entire character of the game changed. Instead of carefully studying dozens of units and carefully planning their moves, players now found they could effectively give orders to only a handful of units. Instead of having waves of troops marching inexorably forward, we now had kings marching forth with only a few hunting bishops ranging diagonally to scout enemy positions. Once it had found the enemy, the king could quickly gather a few attack pawns and move in for the kill.

Now, the players who did well were those who could think on their feet and react to quickly changing conditions. During this process, I discovered the first rule of on-line gaming—a true on-line game is not just a translation of an existing board or computer game; it must adapt and use the advantages and even the disadvantages of the on-line environment.

Most existing on-line computer games have origins similar to Feudal's. On-line games can be divided into four levels of interactivity:

- Solo-against-the-computer. The classic adventure game is an example of these solitaire games. They were novel before the PC made its debut, but there's no reason to pay on-line charges when you can play similar games on a stand-alone PC.
- One-on-one. This includes everything from tic-tac-toe to chess, as well as advanced wargames. These games also waste host-machine capabilities and resources; direct modem-to-modem connections can provide the same experience.
- Some-on-some. Two to 10 players

by Steve Estvanik

Thinking of getting into the multiplayer game scene? Steve Estvanik shares his experience programming his multiplayer game SNIPER!. Programming multiplayer games can be challenging—and fun!

participate in these games. My Air Traffic Controller and SNIPER! games are examples.

- Many-on-many. For these games, I see hundreds if not thousands of players taking an active part in an ongoing scenario. A few games tap the potential of this last category, but none really fulfill that potential.

The SNIPER! Project

I'll be focusing on the challenges a programmer or designer must keep in mind when working in the on-line environment. I have drawn the examples from SNIPER!, a project I first started in 1987. The first version of SNIPER! was released on CompuServe in 1989, and the first release of its graphics interface Scope was in 1991. Updates and enhancements continue.

SNIPER! is based on a board wargame from TSR Inc. and runs on CompuServe. It's a multiplayer, tactical combat game, simulating infantry actions during World War II. You

Figure 1. Game Overview for SNIPER!

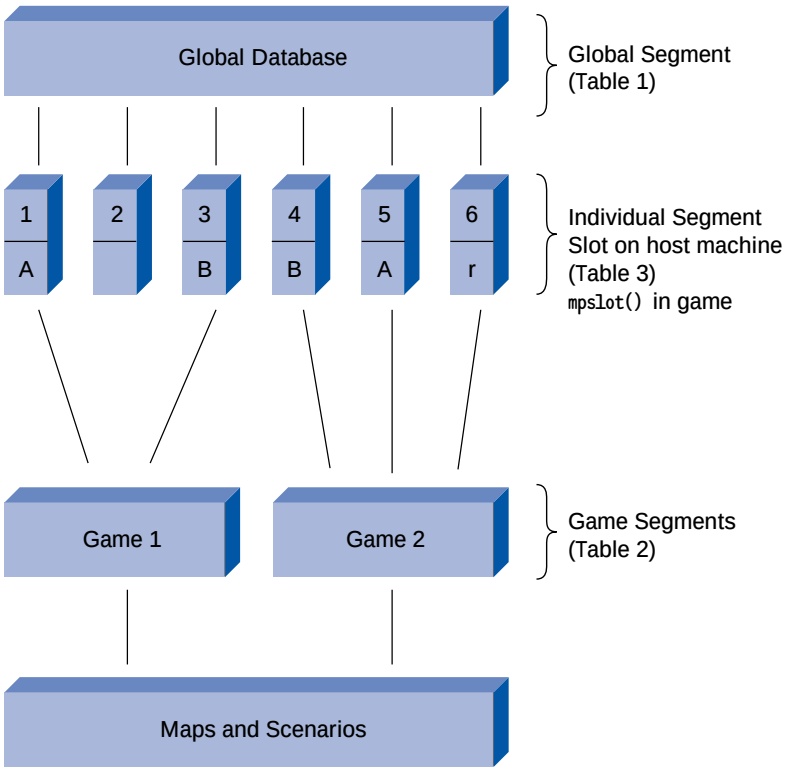


Table 1. Global Database Segment

For the entire session, keep track of:

`gstarted`	Track games started
`gfinished`	Track games finished
`highwater`	Maximun players during a session
`totplayer`	Total players during this session
`starttime`	Time that first player started (used to count total elapsed time)
`highgame`	Highest number of games in play at a time
`highslot`	Highest current slot

For each player, keep track of:

`plmission()`	Selected mission
`plscen()`	Selected scenario
`plrank()`	Player rank
`plnatl()`	Player nationality in current game
`pljob()`	Job number of each player
`poff()`	Offset for player, for example, if there are 12 units, seven in the first faction and five in second, `poff(1) = 0` and `poff(2) = 7`; thus the first unit of a side will be `unit(poff(i)+1)`
`plunits()`	Number of units for each faction
`plside()`	Faction of this player, normally 1 or 2

Game-specific global data:

`plfflag()`	For each faction, alerts players that the other player's program has done something to the unit data
`pluflag()`	Flags which units have been changed
`plsmkflag()`	Indicates smoke has changed
`plrubflag()`	Indicates rubble has changed

command a squad of soldiers and play against one or more other human players from around the world. There's a wide variety of missions and scenarios, each with a unique map and various rules, goals, and challenges.

In addition to playing SNIPER! against other people, you can conduct a reconnaissance of any other current SNIPER! game. Reconnaissance lets you watch your future opponents for style of play. Do they always attack aggressively? Or do they spend time carefully setting up their attack? New players can get strategy hints or other tips by watching two experienced players. And as the designer, I can get ideas for improvements by directly watching what people are trying to do.

On-line games have a lot of browsers, similar to lurkers in forums, who far outnumber the more active participants. Although it wasn't feasible in SNIPER!, my goal in future games is to involve these lurkers in the actual game mechanics by having them drive the economy. So even if they don't come back, something in the game will have changed or been updated by them being there.

Playing vs. Designing

Playing on-line games is as different as designing them. You don't have the long delays that are standard in phased computer games in which each player takes control of the computer to enter his or her turn. Instead, you must constantly weigh choices and decide whether to spend time on one unit or spread your time among several. In the game, this can be used to an advantage against your opponent.

You can place one sniper in a secure position and just give it orders to fire. Your opponent will be forced to react to this annoyance, while you can be moving other units for an attack elsewhere on the map. When your opponent uses the same or similar tactics against you in a different area of the map, you'll have the additional choice of which of many sectors to concentrate on. These complex, chess-like possibilities and decisions reveal the different levels of involvement and challenge that are available with on-line gaming that cannot yet be achieved against a computer opponent.

Scope and other real-time game interfaces allow immediate communication, but the user can also freeze some portions of the action, at least momentarily. Delays of even a second or two can break concentration, so the user should not be interrupted when he or she is trying to enter a command. Thus, in Scope, you can enter a command at any time.

Scope achieves this by treating incoming reports in an asynchronous fashion, reading data from the host character by character. Only when a proper terminator is found does the program decide whether to display the on-screen results or merely store the new data for later queries. While this is occurring, you can enter new commands or move the map or gather other information. You can even combine these operations.

For example, while entering a command, you can stop, use the mouse to scroll the map, gather specific information about unit location or terrain, then return to the half-completed command and use that information to give more precise coordinates. This contrasts with a mode-based system in which you are either in command mode or map mode and cannot easily mix operations from the two modes.

Eliminating modes makes a more realistic interface for the user, but places extra efforts on the designer and programmer to ensure that all features are available and cooperate with each other. This is one of the keys to interactive on-line games. The player should always be able to execute some type of action. Games such as chess and backgammon are usually less interesting as on-line games because players must wait for their opponents to take their turn. (This is particularly true of games with a connect charge from an on-line service.)

Table 2. Game Segment Data

Items	Description
trucenegotiated	Game ended with a truce
firstshotfired	Has first shot or grenade been fired?
tpanicrecovery	Time at which next panic recovery can occur
endofgame	Can be triggered by either player; this is the variable that plucks players from the game
smoking	There is smoke somewhere on the map
tdrift	If smoking, the time that the smoke will drift
mpslot()	Slots of the participating players
nrubble	Number of rubble areas
rublist()	For scope, lists all rubble coordinates
smap()	Current terrain for map coordinates
umap()	Shows where units are on the map for easier checking
msgflag()	Signals a message is waiting for this player; allows any number of real players
smoke()	Smoke locations
smokeopen()	Open slots in smoke array
order.stack()	Pending orders stack
unit()	Current soldier information

Some soldier parameters include:

Parameter	Description
OWNER	Slot of player
FACTION	Same as plside()
GROUP	Subdivisions of a faction
ILOC	Row location
JLOC	Column location
FACING	Direction unit faces
ACTIVATION	Activation rating, a value from 1 to 5
TACSTAT	Tactical status, one of the following: normal, moving, evading, prone, and cargo
COND	Condition, one of the following: normal, wnd, inc, kia, oob
TPANIC	Time of next panic check; 0 equals no panic, otherwise time for next check
RPANIC	Relative panic rating from 2 to 5
LASTMOVE	Time of last move
NEXTTASK	Time for next move
SIGHT	Sighting and exposed flag
WPN	Weapon number
WPNSTAT	Weapon status
SGRENADE	Number of smoke grenades
FGRENADE	Number of fragmentation grenades

Considerations

On-line, multiplayer games pose challenges similar to those of designing other real-time programs, so some experience with interrupts, interjob communications, and shared data is helpful. Compared with other computer games, multiplayer games have several additional design considerations, including game start up and ending and interplayer communication. Related design problems include keeping the game interesting for both sides at all times and providing an enjoyable gaming experience in a designated amount of time.

Most board games and many computer games use some sort of pulsed or sequential movement. It's difficult to make a reasonable, simultaneous action game without a computer. For board games, the best that can be done is to allow players to write orders and try to execute all the orders at the same time. Or you can use a third, nonplaying umpire to control the game.

A real-time game has a completely different feel to it. In SNIPER!, the action is constant. You never need to wait for your opponent. Instead, you have your squad carry out your plans and react to the enemy's moves as you become aware of them. Continuous action necessitated a major overhaul of the original board game's features. Pulses, rounds, and other stepwise parts of the game are gone. No longer are there recovery phases, but then, in a real battle, the two sides don't stop for several minutes while people go running around trying to help others recover from panic or passing out "end of round" indicators and activation chits.

The basic feel of most board games is that of a series of specific segments with specific tasks to be performed in each segment. For on-line games, you never want to leave one player waiting while the other player is making decisions. In SNIPER!, I reworked the original board game phases to form a smooth-flowing background of activity. An important element in achieving this goal was my redesign of the activation concept

Instead of moving a set distance or sighting and waiting for one of several phases, all activities take place whenever a particular soldier is ready to perform them. Once any activity occurs, an activation rating determines the next time a soldier can act. The main processing portion of the host program looks to see if either side has orders that are ready to be executed. Each command takes a variable amount of time to be executed, so orders are "stacked" for each soldier. This lets you give multiple commands to one or more soldiers, then move elsewhere.

A welcome side effect of this strategy is the minimization of the influence of baud rate (speed of transmission) used by the opponents. Each person's job will execute the next order on the

stack, no matter which side that unit belongs to. So some orders of each side are executed by the slow player's job and some by the fast one.

I changed the activation ratings from the board game concept to create simultaneous, real-time action. In the board game, a die roll decides whether particular units can move or act. In my game, activation indicates the probable time until the next action. From the game charts, I extracted four variables that describe a unit's activation: an overall evaluation of the performance of a unit—the track and activation ratings, the number of expected rounds, and a small random factor.

Thus a track A unit in the board game could expect to be active up to six rounds. The time between simple actions in the on-line game is between four and 14 seconds. Contrast this to a type H unit whose time between similar actions is 10 and 25 seconds. These numbers were determined experimentally, and the tracks vary according to nationality and scenario. They have to have some spread, but not too much, or the game will be unbalanced. The details aren't published. Instead, players know that a track A is faster than a track B but not necessarily by how much.

Mapping and Sighting

Another major change in moving from a board game to a computer game was the map shift. People relate more naturally to a square grid, where left, right, and diagonal have obvious meanings, while a hex map simplifies the designer's job. On the computer, diagonal and sideways movement are easy to calculate correctly. Thus, while your squad faces in one direction, it can move in several other directions relative to its facing. This lets you fall back, for example, while maintaining covering fire against anyone foolish enough to run after your retreating squad.

Sighting requirements are different for on-line games. In SNIPER!, any unit can sight and fire once a unit is in sight. This eliminates the need for specific-opportunity fire rules. We avoid

Table 3. Individual Player Data During a Mission

Flags:	
`endofwatch`	Allows end of this player's mission without ending game for other players
`lastrubble`	Last known rubble square (for requesting an updated rubble list)
`lastSmoke`	Last known smoke (used to determine when to request updates)
Data:	
`mpwho`	Which element in `mpslot()` is current player?
`rank`	Current rank
`totpts`	Total points
`cumhitpoints`	Carryover of hitpoints (up to 100)
`tourney()`	Current tournament scores
`scope`	Is this player using Scope?

Table 4. Interprocess Communications as IPCF Structures

Two-Player	Multiplayer
`pljob` (sender)	`pljob` (sender)
`curgame` (slot)	`curgame` (slot)
`mapnum`	`mapnum`
	`ptr` to next player (5, 6, 7)
	Second player
	Third player
	Fourth player

Table 5. Global Lock Parameters

Lock Variable	Items Affected	Locked By
`lock1`	Game segment variables	`initgame()`
`lock2`	`plfflag()`, `pluflag()`	`flagunit()`
`lock3`	Not used	
`lock4`	`gamepoints`, `hitpoints`, `preservation`	`resolvegame()`
	`curgame`, `gstatus`, `lvlopp`	`zerogame()`, `clearslot()`
`lock5`	`radioflag()`, `radiomsg()`	`pushradio()`, `popradio()`
`lock6`	Hall of Fame updates	

Table 6. Individual Game Lock Parameters

Lock Variable	Items Affected	Locked by
`glock1`	Unit array	`moveunit()`
		`panic_recovery()`
		`paniccheck()`
`glock2`	`nactive`	
`glock3`	`orderstack`, `norders`, `hasorders`	`poporder()`, `initindividual()`
		`cancel_orders()`
`glock4`	`msgflag`, `msgstack`, `nmsg`	`pushmsg()`, `popmsg()`
`glock5`	smoke, `nrubble`, `rubble()`	`pushsmoke()`, `popsmoke()`

the board game problem of having units that are in known positions but that can't be fired on because there's no current spotter.

Grenades and Smoke

There's added realism beyond the board game by having walls, floors, bridges, and trees turn to rubble when hit by an explosion. Similarly, rubble can be added to maps before play starts, increasing the unpredictability, since the rubble will be random in degree and placement.

Smoke is generated from any explosion or from special smoke grenades. Each game has a wind direction so the smoke drifts, persists for awhile, then fades. Smoke inside a building persists longer, making it difficult to find people inside. Fires sometimes start and spread out of control.

Technical Challenges

This section describes the very different approach to program design that's required for an on-line game. Sharing data among users in traditional, single-microcomputer games is quite simple, since everything is either in memory or on a disk. For an on-line game, you need to decide how players will communicate. You need to design different segments of data that can be shared by different classes of users. This affects subsequent programming problems.

Figure 1 shows my solution for SNIPER!. At the top is the global dataset, and the contents of this segment are shown in Table 1. This general information is available to any player at any point in the game. It includes player names, current status of each player, the state of the game, messages among players, and timing information. Once a game is started, the only people who need the individual game data are the actual players and any watchers, so each game has its own data segment attached, as shown in Table 2.

Thus, you can swap these segments in and out of memory as needed rather than try to guess some optimum amount of memory at the start and then either have too much unused memory or not enough for the demand. Finally, users carry with them a quantity of personal data, such as points scored, as shown in Table 3. So, in Figure 1, players 1 and 3 are playing sides A and B in game 1. They have their own data segments and share that for game 1. They also share the global data area with all other players. Similarly, when player 6 decides to recon game 2, being played by players 4 and 5, player 6's job locates the game 2 data segment and attaches to it to be able to watch the game. Player 2 is logged into the game, but not attached to any game segment.

All these different data segments are, of course, transparent to the players. But this choice gives me flexibility I might not have with other schemes. For example, when a game starts, I first read a copy of the map into the game segment. By making copies of the maps, each game starts with the same map but is free to change it as the game progresses. Thus, explosions can have immediate effects on the terrain. Or, random pregame bombing can be invoked to destroy part of the terrain. Thus, the maps, while similar, differ slightly for each game.

A series of interprocess communication facilities (IPCF) sends messages from one job to another. One job starts a game, sets up the map and terrain, then signals the other player or players that the game is ready. Table 4 shows the IPCF used for one-on-one and multiplayer synchronous starts. In the two-player version, the first job sends an IPCF consisting of the sender's job number, current game number, and map number.

The receiving player's job uses the current game number to look up the address of the game segment, and the map number to ensure that both players are using the same map. (Current game numbers are not predictable, since many games can begin and end during a session, and the numbers are recycled.)

For multiple players, the IPCF is more complicated, since no player should be allowed to enter orders until all have been linked into the game. Because of locking, only one player at a time can gain access to the game segment, so a cascading design works here.

Player A starts the game as before, then sends an IPCF to player B. Now, the IPCF includes the slot numbers of players C and D, too. When player B is set up, his or her job sends a similar IPCF to player C. (The only change is that the pointer now points to player D, rather than C). After D is attached to the game segment, his or her job still has the addresses of the other players, so a message can be sent to all jobs that allows the players to start the game.

Normally, each attachment to a game segment takes about one second, but if the system is slow, the entire process might take 10 seconds or more. This design ensures that the startup procedure will work no matter what delays occur. It can also handle the sit-

uation in which a player is disconnected between the request for the game and the actual start of the game.

For asynchronous starts, the first player just begins a game, and others can join at any time. Recon is a special case of an asynchronous start in which a player has a more limited command list.

Shared locks

The game-level segment contains all the data, even though players are unable to see all the data for the opposing side. By giving both jobs access to all the information, a faster job can execute orders for either side. Since this is a host-run system, only information needed for displays is returned to the individual PCs, so there's no way to steal a look at an opponent's information. To protect against simultaneous updates, variables or arrays can be locked.

A series of locking variables on the global and game levels allow flexibility without destroying response time. Table 5 shows the global level, while Table 6 shows the game level. In most cases, only two to four players contend for game locks. For global information like player scores or overall status, all players in the game contend, so these functions need to be minimized, and any updates need to be fast and reliable.

End of Game Resolution

When the game ends (for normal or abnormal reasons such as a disconnect), a signal is sent to all players, and their final status is determined. They're then pulled out of the game, and the game segment memory is freed. A game-resolution module then assigns points, enters players in appropriate Halls of Fame, and returns players to the conference area with everyone else not currently involved in a game. Watchers are shown a short message, then popped back to the conference area. ■

Steve Estvanik is a professional games designer in Seattle. He concentrates on strategy, puzzle, and interactive multiplayer games. He can be reached by e-mail at 76703.3046@compuserve.com.

From A Train to Zork: The Business of Game Distribution

Nintendo is famous for Mario, the plumber whose name has gone out on over 100 million cartridges. Developers, distributors, and publishers all dream of a Mario that will propel them into success.

Games do not sell themselves any more than they play themselves. People buy them to run on a computer or video game platform from someone who got them from someone else, who may or may not have created them. This process can be confusing, even to many game developers, but it need not be. Unlike books, games run on platforms: either personal computers or dedicated home video game consoles. This is where we begin.

The Video Game World

Between 35 and 50 million U.S. homes have a video game console. That number comprises at least 90% of homes with children and from a third to a half of the 95 million households in the country. Nintendo of America almost single-handedly recreated today's home video game industry on the ashes of Atari's empire, selling more than 34.5 million 8-bit Nintendo Entertainment System (NES) consoles in the U.S. between October 1985 and the end of 1993.

That console is nearly obsolete now. Its immediate successor is the 16-bit cartridge console, sitting in an estimated 25 million homes by the end of 1993. (Not all 16-bit owners have an 8-bit machine; thus the ambiguity over the number of video-gaming households.) Turbo-Technologies Inc. and Sega of America rushed their TurboGrafx-16 and Genesis 16-bit consoles to market in December 1988 and February 1990, well before Nintendo's Super NES (SNES) appeared in August 1991.

Analyst Lee Isgur of Volpe, Welty & Co. scoffed at the notion that Sega's 18-month lead could have prevented Nintendo from retaking dominance of the 16-bit market, but he notes that Sega Genesis currently outsells SNES. Nevertheless, Nintendo alone generated sales of $4.3 billion in its 1992 fiscal year out of a total home video game retail market of almost $5.5 billion. Both numbers are undoubtedly at least $1 billion bigger already.

Over 700 titles were released for all video game platforms in 1993. Sales volumes of 100,000 are considered successful and only 20% of games released sell this magic number. Sonic 2 for the Sega Genesis has sold over 5 million copies worldwide. Starfox for the SNES was

the first cartridge to ship an initial million units. Mortal Kombat shipped 3 million cartridges within two months of its September 13 Mortal Monday release and will certainly have surpassed 4 million by the new year. Street Fighter II, among all its incarnations and across all platforms, has sold over 8 million cartridges worldwide. Most cartridges sell between 30,000 and 50,000 copies.

Video games appeal to a younger audience. "The hard-core game-playing public is still 12- to 14-year-old males, so running, jumping, and shooting games are the most popular," says Scott Pelland, editor of *Nintendo Power* magazine. Indeed, recent figures show action games generate 31% and sports games 27% of total cartridge sales; adventure games is the last major category. Of the sports total, football makes up a full third, basketball a fifth, baseball and hockey an eighth apiece, and boxing and golf each about one twentieth. The proportion of sports sales for the Sega Genesis is higher than for the SNES. Fighting games and what Bob Botch of U.S. Gold calls "critter" games are currently very popular. Table 1 summarizes the platforms available today.

Nintendo reported a ratio of over three cartridges for each SNES platform sold in fiscal 1993; that ratio for the Game Boy was closer to six to one. Nintendo is famous for Mario, the plumber whose name has gone out on over 100 million cartridges. By autumn of 1993, over 1,300 titles had been released and over 313 million cartridges sold for Nintendo's three platforms: 231 million for NES, 28.5 million for SNES, and 54 million for Game Boy.

The Street Fighter II cartridge for the SNES was available a full year before the Sega Genesis version came out. However, Nintendo's milder version of a similarly successful arcade game, Mortal Kombat, is being outsold more than two to one by the bloodier, simultaneously released Sega version. Nintendo's vaunted FX chip, enabling better graphics in 16-bit cartridge games, does not make up for the lack of a Nintendo CD-ROM-playing add-on.

The Sega mascot is Sonic the Hedgehog, with a higher Q-rating (an indicator of public awareness) among children than any other video game character. A typical Sega Genesis buyer in the U.S. will spend about $255 more on cartridges and peripherals during the first two years, says Ellen Beth Van Buskirk of Sega of America. Sega has a big head start with its CD-ROM Genesis add-on and other peripherals. Distribution has finally caught up to Nintendo's, too, with nearly 18,000 retail shelves carrying Sega products. Less than half as many publishers support the Sega Genesis, but there are half again as many Genesis titles as there are for the SNES.

Another company, Turbo Technologies Inc. has introduced new technology even more aggressively than Sega. TTI was formed in April 1992 as a joint venture of Japanese hardware giant NEC and $400 million-a-year, Sapporo-based Hudson. A 21-year old technology house, Hudson was the first Nintendo licensee and the first to publish 16-bit CD-ROM software in 1987. Its TurboGrafx-16 or PC Engine is the second most popular platform in Japan

by James Cooper

Once you have a great game, what do you do with it? Finding a distributor in the complicated world of retail is a tricky proposition. Learn where your best options lie—and what to avoid.

Table 1. Platforms Available Today (Continued on p. 26)

	Nintendo NES	Nintendo SNES	Sega Genesis	SNK NeoGeo	Nintendo Game Boy
Release Date	10/85	8/91	2/90	11/90	6/89
Installed Base	34.5 million	14 million	12 million	95,000	15 million
Retail Price	$40 to $50	$90 ($130 for full system)	$90 ($120 for full system)	$650	$50
Cartridge/CD Price	$25 to $50	$50 to $80	$40 to $70	$170 to $240	$20 to $35
Titles Available (as of 12/93)	693	335	Over 500	41	330
Titles Released in 1993	55	185 (7 NOA)[1]	Over 110 (10 SOA)[2]	7	75 (6 NOA)[1]
Third-Party Developers	77	77	33	12	77
Notes	8-bit	16-bit	16 bit	24-bit arcade quality	8-bit black and white

1. Indicates Nintendo of America release
2. Indicates Sega of America release

behind Nintendo's Super Famicom, the equivalent of a SNES.

Although U.S. distribution cannot compare to Nintendo and Sega, TTI products are carried by retailers including Toys R Us, Electronics Boutique, and The Good Guys. Working Designs is the only third-party publisher; titles from Namco, Irem, ICOM Simulations, and other developers are released under the Hudson Soft label.

The NeoGeo from SNK Home Entertainment remains the only 24-bit home game console. Literally an arcade machine, it also is unique in allowing players to carry saved games from home to the arcade and back using a memory device the size of a credit card. Only twelve developers are licensed to this platform, and no more are anticipated.

Other CD home entertainment platforms include the Philips CD-Interactive or CD-I, the Commodore CD32 replacing its CDTV, the Tandy Video Interactive Service or VIS, and Pioneer's new LaserActive. Seven thousand Radio Shacks sell the VIS; the CD32 is the first with a double-speed CD-ROM drive; and the CD-I is backed by one of the largest multinationals in the consumer electronics industry. All are available now for about $400, except the LaserActive. It costs as much as the other three put together, $1,200, and can play either Sega CD or TTI Duo games: to play both costs $1,500. None of the four is clearly targeted at the game market, as the few games in their libraries show, but developing titles for them would provide useful work experience.

Portable video game players make up 15% of the video game industry, according to Atari director of communications Bob Brodie. The 8-bit Nintendo Game Boy and its cartridges are the least expensive and have by far the largest installed base, larger even than any home console. The others offer color. The Lynx and TurboExpress are 16-bit machines. Both Game Gear and TurboExpress can be turned into portable TVs with tuner accessories costing $60 to $100. Rated best of the portables, TTI's TurboExpress is the only one to play the same 16-bit cartridges as its home counterpart.

An even newer generation of home video game machines is available now: the 32-bit 3DO Multi-player introduced in October 1993, and the 64-bit Jaguar introduced by Atari over Thanksgiving 1993. Even more impressive than its technology is the business model 3DO advances: no annual limits on the number of titles per publisher, no limits on the source or quantity of production, and licensing fees and production costs at least an order of magnitude lower than what Sega and Nintendo charge. CDs are cheap to make, too: about a dollar each in quantity.

Which Comes First?

This raises the chicken-and-egg conundrum of every computer product: which comes first, hardware or software? Table 1 shows that hundreds of titles are available for the most popular platforms. Yet during 1993, Nintendo itself published only seven SNES cartridges and Sega only 10 Genesis titles. The vast majority of games come from third-party publishers, so-called because they are neither the consumer nor platform manufacturer. Third-party licensees support a platform by publishing cartridges that run on it. Just as platform providers need publishers

Sega Game Gear	Atari Lynx	TTI Turbo Express	3DO	Sega CD
6/91	12/89	1/89	10/93	10/92
3 million	Not available	150,000 worldwide	90,000 to 100,000	300,000 to 400,000
$90	$80	$130	$700 (with Crash 'N Burn)	$230 (with Sewer Shark)
$35 to $45	$25 to $50	$20 to $50	$40 to $50	$50 to $60
About 150	72	200 (83 available)	10 to 20	45
About 40 (10 SOA)[2]	8	40	10 to 20	25 (9 SOA)[2]
14	3	0	400	14
8-bit, color head-to head, $100 TV tuner add-on	16-bit, color, multiple player	16 bit, color, five-player, plays TG-16 cartridges, $60 TV tunner add-on	32-bit, CD, RISC	CD add-on to Genesis

to field sufficient titles to create demand for their game machines, publishers often enlist independent design houses to execute game development.

Sculptured Software of Salt Lake City is perhaps the oldest and most prolific of these design houses; it has created over 150 titles since 1984 including the SNES Mortal Kombat for Acclaim in 1993. Averaging 20 projects a year, Sculptured Software has developed for the PC, Macintosh, Amiga, Atari ST, and Commodore 64 platforms. Currently, it is only supporting the SNES, Sega Genesis, Sega CD, and Nintendo Game Boy. Founder and president George Metos employs over 70 people and deals with two dozen publishers at any given time.

Alexandria Inc. is another such third-party publisher, developing for the 3DO, SNES, Sega Genesis, and Sega CD, although it is a licensed publisher for none of them. Alexandria president Ken Balthaser comments on the platforms available today: "We're rooting for 3DO. It's better for us if there's a high-quality platform out there. It's less risky, and the royalty is a couple of bucks. It is their technology, but they're not milking us. Atari is the same, being reasonable about licensing. You can get CDs manufactured locally and in small quantities vs. cartridges from Sega or Nintendo: with a minimum of 30,000 at $10 or $12 per cartridge, you're talking millions of dollars. Five thousand CDs and a minimal royalty is a whole lot less risk. Following the 3DO model makes it easy on publishers."

His only complaint is that, for now, "As a developer, we gave up choosing 3DO. Even with a royalty, we'll make no money other than the guaranteed fee." But for the future, he says, "We've had big companies like Philips come out with new products and fail. Sega is a video game company. They know their target market. 3DO needs to do the same thing."

The Cartridge World

If the availability of programmers does not dictate platform, criteria in choosing one might include the installed base, owner demographics, development cost or difficulty and the inevitable, arguable question of potential gameplay quality.

Installed Base. Robert Botch, president of cartridge publisher U.S. Gold, thinks, "One million and growing is a good platform critical mass; 100,000 is too low. On the other hand, top programmers prefer to work with the latest technology. You want to keep them happy, so it may not be a business decision after all."

Owner Demographics. SNES players are skewed toward males 12 to 14 and younger. The older NES machines are being handed down to younger siblings, reports Capcom public relations coordinator Erin Skiba, so the publisher of Street Fighter II is still releasing NES titles such as Rescue Rangers. The Sega Genesis has been positioned more toward teenage and older audiences. Portable play by females and adults is surprisingly high. Nintendo reports almost half the Game Boy primary users are adults, of which nearly half are female.

Development Cost or Difficulty. 3DO provides a complete development environment, including clips from movies in the Time Warner library; it's friendlier to nonprogrammers and costs about

Table 1. Platforms Available Today (Continued from p. 24)

	Atari Jaguar	TTI Duo	TTI TurboGrafx16
Release Date	11/93	10/92 TG-CD 12/89	12/88
Installed Base	30,000 to 50,000	Worldwide: 50,000 plus 100,000 TG-CD	1.17 million worldwide
Retail Price	$250 plus $250 for CD	$300	$80
Cartridge/CD Price	$50 to $80	$30 to $60	$20 to $50
Titles Available (as of 12/93)	4	50	200 (83 available)
Titles Released in 1993	4	20	40
Third-Party Developers	Over 29	1	0
Notes	64-bit, RISC	16-bit, CD, and SuperCD	first 16-bit home console

$15,000 if you count the Apple Quadra it runs on. The Jaguar offers "freedom, a lot more power with no operating system to get in the way," says Atari's Brodie.

Gameplay Supported. Some games require the more advanced platforms. Sticking with 16-bit performance, Capcom will not release Street Fighter II on any portables or CD, says Skiba. Sonic the Hedgehog offered unprecedented speed when it debuted with the 16-bit Genesis system, and Starfox made a similar impression in Jan. 1993 with its FX chip. High-end applications will no doubt showcase 3DO and Jaguar strengths shortly.

The cartridge world may seem frenetic, but it was designed to accommodate advancing technology and has done so remarkably well. Cartridges keep new technology compatible with the large installed bases of older platforms. T&E Soft's John Eaton points out that when Nintendo introduced the FX chip, its Starfox game remained playable on the same SNES consoles. "It's an interesting thing. It's something that you can't do with a PC, and you are not going to be able to do with, say, a 3DO machine. Once you've got that hardware, it's a very expensive prospect to upgrade it. There are all kinds of entertainment products that a lot of people cannot play on their computers because they are not the latest and greatest."

The PC World

Kirk Green of Walt Disney Software sums it up, "We are inundated with so many different platforms. It is hard enough to keep up with something as well established as the home PC."

Of over 125 million PCs and 10 million Macintoshes worldwide, 30 million are in U.S. homes, claims Steve Eskenazi of Alex, Brown & Sons; however, he doubts even half are used for games. Paul Wheaton of Dataquest notes that of the 12.5 million PCs sold in the U.S. in the past year, 3.4 million or 27% went to homes. Research supplied by Mark Pendergrast of the Software Publishers Association (SPA) shows that 1992 PC game purchases totaled $267 million for DOS, $31 million for Macintosh, and almost $30 million for Windows games.

Historically, game developers have been obliged to support graphics standards from Hercules to VGA, XGA, and SVGA; one, two and three-button mouses; extended and expanded memory; joysticks; and several flavors of sound card. Currently optional but well recognized as growth areas are Windows, CD-ROM, modem and network capability, full-motion video, decompression hardware and software, and voice recognition.

Every PC peripheral and software standard is essentially a mini-platform. Sometimes platform lifecycles are judged by nonintuitive standards. Gametek vice president of product development Gordon Walton relates how Software Etc. once discount-racked all PC games packaged with $5\frac{1}{4}$-inch disks. Many of the same titles with $3\frac{1}{2}$-inch floppies were still selling strong.

Computer games in North America sold $342 million at retail in 1992 by SPA reckoning—29% more than $265 million in 1991, and sales of $69 million for second-quarter 1993 were 22% above $56 million for the same period in 1992. Between 600 and 900 titles are released annually, of which the top 25 will sell 100,000 in a year, says Gametek's Walton. Half break even, 15% make money, and, if it lasts for three months, the average title might sell 40,000. A classic like SimCity is approaching 1 million worldwide after four years. Civilization from MicroProse and Falcon 3.0 from Spectrum HoloByte, two resounding PC success stories from companies that happen to be merging, have sold 250,000 to 300,000 each.

The SPA reports that home education continues to beat out entertainment and personal finance for the strongest growth among its three consumer categories. Driven primarily by Macintosh software, home education sales grew 55% in the second quarter of 1993 over the same period a year ago, and total sales of $146 million in 1992 jumped 47% from $99 million in 1991. The ratio of entertainment to education sales was 2.65:1 in 1991, 2.34:1 in 1992, and less than 2:1 in the first half of 1993.

Available sales figures for Windows educational software were well over twice those for Windows entertainment software in 1993. Kirk Green of Walt Disney Software notes that edu-tainment is growing popular, even for preschoolers. "Mickey's ABCs and 123s

were just certified SPA Gold with 100,000 sales. That's pretty good for a product aimed at three-year olds."

Are the rules any different for cartridges and PC games in return policies, discount racking, and so on? Denny Thorley of Sega licensee Extreme Entertainment says no, "They are exactly the same. The only difference is the profit margins."

Retailers

"Games are like fish. If they sit on the shelf a week, they start to stink." Norman Weiss is sole proprietor of the Software Library in Orinda, Calif. He is a retailer. When it comes to tracking down those hoards of consumer dollars, the game industry relies on people like Weiss.

PC games are sold in about 15,000 stores in North America, video games in about 18,000. The top 10 retailers account for 75% to 85% of all PC and video game sales. For video games they are:

1. Toys R Us
2. Kaybee Toys
3. Wal-Mart
4. Target
5. Sears
6. Electronics Boutique
7. Software Etc
8. Egghead Software
9. Babbages
10. Blockbuster Video.

Publishers have direct relationships with these retail outlets through inside sales forces or independent sales representatives.

Three primary retail channels for cartridge sales are:

- Toy stores (40% of sales)
- Mass merchants (40% of sales)
- Electronics and software stores (20% of sales).

The six primary channels for PC games, in approximate order of overall sales, are:

- Software specialty stores, such as Electronics Boutique, Software Etc, Egghead Software, Babbages, and G+G/Captron
- Computer superstores, such as Comp-USA, BizMart, CompuMart, MicroCenter, Computer City, and Incredible Universe (Tandy)
- Mass merchants, such as Sears, SAMs, K-Mart, Target, and Montgomery Ward
- Consumer electronics stores, such as The Good Guys, Circuit City, Silo, The Wiz, and Radio Shack (Tandy)
- Toy stores, such as Toys R Us and Kaybee Toys
- Discount warehouses, such as Wal Mart, CostCo, Price Club, and Office Depot.

Specialty stores continue to dominate, but computer superstores, discount warehouses, and mass merchandisers like SAMs and Target are gaining fast. Toys R Us does not sell PC titles at all in most of its stores, following a bad experience in the 1980s. Tandy is the opposite; although it does not stock video games, if Radio Shack is carrying your PC game, it means a sale volume of 7,000 to 28,000, and Tandy does not return products. Norman Weiss of the Software Library is tired of seeing his wholesale costs exceeding some sale prices of large chains. He and 60 other retailers are banding together to negotiate group purchasing arrangements.

Bob Botch of U.S. Gold says buyers want to see three things. "In order of importance: a marketing program behind a product, well-recognized licenses affiliated with the product, and quality in the product itself." LucasArts Entertainment enjoys a strong brand advantage, and marketing manager Mary Bihr agrees the game industry is more and more hits-driven. The trend is obvious, "Computer superstores and discount warehouses stock hits, not breadth." Gametek's Walton confirms that 90% of relevant storefronts carry 25 PC titles or less. Video-game store buyers base their decisions more and more on the top 10 or 20 lists in consumer trade magazines like *GamePro* and *Electronic Gaming Monthly*, according to Brad Berglund of cartridge publisher Hot-B USA.

Retailers think in terms of turnover, returns, and sales per square foot. Turnover had better be at least one PC game per store per month or your title goes straight to the discount rack, says Walton of Gametek. Customers return about 8% of PC games, up to 50% for particularly bad ones, and they have no deadline for doing so. Publishers are expected to pay freight.

In the mall, nobody averages $1,000 per square foot in annual sales like the food servers or a very strong jewelry store. Clothing, book, and gift stores would be very happy with half that, but a Software Etc or Electronics Boutique should be able to turn $500 or more per square foot a year easily. To see sales per square foot and shelfspace perfected, visit the video game aisle at Toys R Us.

Atari's Jaguar Console

Retailers collect people's money in exchange for shrink-wrapped boxes. That said, it may come as a surprise how little retailers want to deal with people or products. They dislike walking customers through software installations, explaining AUTOEXEC.BAT and CONFIG.SYS files and how best to get past a puzzle. Worst of all, retailers hate selling something so bug-ridden it gets returned. More forgivable are hard-to-stack, oddly shaped boxes as long as they walk out the door.

Speaking of handling boxes, Maxis director of sales Ileana Seander lists just a few requirements retailers might impose on a supplier: special handling such as preticketing, merchandising support like counting stock in the field and automatically reordering, nonstandard pallets or skids (288 PC games fit a normal pallet; that's two gross), and directly shipping to as many as 250 separate locations instead of a few central warehouses.

A retailer's franchise comes from building relationships with the community. Egghead Software's CUE Card program has almost 1.8 million members or about 0.75% of the population. That is a good-sized herd of disposable income to be able to track down.

Distributors

The biggest distributors of game titles are Ingram/Micro, Merisel, Baker & Taylor (parent of Soft Kat), Handleman's, Avco, American Software, Beamscope Canada, and Fidelity. Nintendo maintains an official list of 10 to 20 recommended distributors.

"Our distribution mechanism is just a big engine," explains Ingram/Micro senior director Jeff Davis. To fuel this engine, Davis and a staff of eight evaluate 2,000 product submissions a year. "We look for titles with sales potential of $30,000 a month." At this self-proclaimed world's largest wholesale distributor of microcomputer products, over 100 people compose a creative department for marketing and sales services whose duties include publishing an inch-thick phonebook-like catalogue and sending it to every retailer serviced every month.

"Believe it or not, we try hard not to own our stock. Think in terms of selling through instead of selling to a distributor," advised Davis at a November 1993 seminar for the Multimedia Development Group of San Francisco. He meant it. Ingram/Micro keeps only about 22 days of inventory in stock. Distributors can bend the rules in their favor in many ways. Stock balancing lets them return older products to a publisher in exchange for newer releases; however, retailers returning products will be credited with a price in effect at the time of return. This difference could be between the $36 Weiss originally paid for the game and the $10 he eventually gets back.

Mary Bihr tells why LucasArts Entertainment recently chose a direct relationship with a distributor after being an Electronic Arts affiliated label for many years. "Baker & Taylor is the largest distributor of CD-ROM products today. With the new arrangement, we will not have to accept any returns, and they will place us in new storefronts including college bookstores and comic book stores. This works out well with our CD-ROM emphasis and the release of our first games based on comic books, the Sam & Max series by Steve Purcell."

Distributors also prefer high prices and the bigger margins that go with them. This can put them in conflict with publishers, who want to grow their markets by driving price points down.

Publishers

"In a whole new business, you can make up the rules," avows Tom McGrew, vice president of sales and marketing for the past six years at CD-ROM content publisher Compton's NewMedia. In the

WHERE VIDEO GAMES COME FROM

John Eaton, president of North American operations for Japanese-owned PC and cartridge publisher T&E Soft, explains where most video games come from. "The financial model that Nintendo and Sega use is to control all the manufacturing for the most part. They put the licensing fee on top of that." Every 16-bit cartridge costs a third-party publisher between $16 and $30, of which $10 is generally considered a licensing fee. An 8M cartridge with battery backup might cost $24 from Nintendo and $20 from Sega. Nintendo charges more, explaining why the Genesis Mortal Kombat costs $10 less at retail, and Nintendo cartridges cost $5 to $10 more.

"All the manufacturing" means just that: cartridge, label, manual, box, even warranty cards are made and assembled in Japan. A minimum first order is 30,000, and reorders are at least 10,000, except for European titles that must be translated into more languages. Until mid-1993, all licensees were limited to three releases per year; however, any game scoring high enough on Nintendo's or Sega's evaluation scale would not to be counted.

"You have to buy your cartridges from them in advance. You have to put up a letter of credit to make the order, and it takes three months for your product to get manufactured in Japan and shipped to the U.S.; you pay for it 'on the dock' at Kobe. It winds up in the store about four months after you ordered it, and you wind up getting paid about six months after you put up your letter of credit," says Eaton.

A 50,000 initial shipment of a 12M cartridge like Street Fighter II with battery back-up would cost $25 or more—$1.25 million floated for six months. Even a bare-bones 30,000 run of the smallest cartridge would tie up nearly $500,000, and you never stop owning the cartridges, even after the shopper carries them out of Toys R Us.

Eaton summarized, "There's a tremendous inventory risk and a lot of money that gets tied up. A lot of people believe that there are very few software people actually making money in the cartridge business. And that's one of the things that is going to drive the move away from the cartridge business and towards the CD-ROM, 32-bit business. Or that's one theory."

absence of a platform provider pursuing the same agenda, he is right, and there is no Nintendo in the PC CD-ROM world.

Publishers deal extensively with everyone: consumer, platform provider, distributor and retailer, and of course developer, so we will examine each of these relationships.

If platform providers create a market, publishers can grow it. Compton's drove the price point down to $39.95 for CD titles, and their sales climbed from $5 million in 1991 to $15 million in 1992 and $35 million in 1993. Says McGrew, "For two and a half years, we have been a market maker. We want CD-ROM to be ubiquitous. If you do not make it ubiquitous to the consumers, then you'll suffer the peaks and valleys of the software business." As of November 1993, Compton's International Encyclopedia had sold over 600,000 copies, including bundling with IBM, Compaq, and Apple.

Electronic Arts is an example of an assertive game publisher. It has always pioneered platforms and is now growing the multiplayer market with a four-way Genesis adapter, multisided strategy games like General Chaos, and eight-page inserts in magazines like *Electronic Gaming Monthly*, promoting Electronic Arts sports tournaments around the country.

More than any other group except retailers, publishers make it their business to understand consumer behavior. Spectrum HoloByte conducted focus groups in the course of marketing Iron Helix, reports developer Drew Huffman of Drew Pictures. Such market research is par for the course. Warranty cards also help these game makers understand game players.

Publishers and platform providers can protect a market. Sega did so in 1993 by introducing a video game ratings system, but the best example is, of course, Nintendo. Stringent content restrictions, gameplay standards, title limits, and quality-driven quantity restrictions throughout the 1980s all ensured that no Atari fiasco would strike the ecosystem Nintendo had scrupulously nurtured. The new ratings system may amount to no more than free publicity for Sega, since no Nintendo title will ever get anything other than the equivalent of a G rating. This is a restriction at least 77 third-party publishers have been willing to accept.

Publishers' fates follow those of their platforms. Incentives for the two are therefore usually aligned, but, in one recent case, the platform provider was at a rare disadvantage. Capcom undoubtedly cut into Sega sales when Street Fighter II for the Genesis, a genuine hit, was delayed four extra months from June to September 1993.

How do developers respond to the newest platforms? "It's an interesting time right now in the business. Everyone is wondering where to put their bets. It's risky for publishers and developers—but more so for publishers because they have more riding on it," observes developer Ken Balthaser.

Two-thirds of the 33 Sega Genesis licensees also support Nintendo platforms. Eight of 14 Sega CD third-party publishers do. Of the Jaguar's 20 initial licensees, only two were either Sega or Nintendo third-party publishers, but six of the next nine announced by December 1993 are. With over 400 licensees and counting, 3DO has clearly positioned itself as more than a video game console.

To play the game, developers must agree to take all unsold merchandise, or they will not be asked back. Distribution consultant and former Broderbund marketing vice president Leigh Marriner says breaking into the retail channel is extremely difficult, partly because channel members demand protection against returns. Publishers must demonstrate that they will be around next year, that they will be able to make good on all unsold inventory, and that they are willing to do so.

The importance of two other publisher functions is waning, however: securing broad retail exposure, and fielding a product line. U.S. Gold's Bob Botch says, "Getting more storefronts is not so much a priority; getting the stores to carry our whole line is." Marriner concurs: "1993 has been the first year to see some of the major publishers not get all their products picked up by the retail chains."

Sometimes a publisher can dictate market terms. Compton's NewMedia and Tom McGrew established a whole new margin level for CD-ROM products: "Content has more value than software. This is not the software business, this is a bastard child. Because of that, you can't apply the old model. The software business is dead." How much of being a pioneer is obligation, and how much is opportunity? "We were the first with standardized packaging for CD-ROM products. Now guess what standard Sears and K-Mart are enforcing."

Overly assertive publishers run the risk of antagonizing their resellers and jeopardizing future access to consumers. Norman Weiss resents the terms imposed by a PC publisher in the San Francisco Bay Area: "They're like Nazis. They made me sign up for a program. I practically had to fill out my life story, then I got copies of every new product whether I wanted it or not. They wouldn't even let me quit. To this day, returning even defective products to these people is a nightmare. It reminds me of IBM. Years ago, I applied to be an authorized retailer but they turned me down. Too small. Now they're knocking at my door, begging to sign me up. I won't do it. I can't do it. For the last three years, I've told my customers never to buy IBM. I can't go back on that. And besides, the quality isn't there." What about that PC publisher's products? "I only order them if I have a customer request."

"Think mass market," exhorts Tom McGrew. There will be changes when it is reached, warns Norman Weiss. "You can't buy Microsoft or Symantec products from the factory. Try it. They will refer you to a distributor or your local retailer. There are none of those toll-free 800 numbers you see in magazine ads for computer games, either."

Publishers want a developer they can trust to deliver on the promises they make down the chain. Botch of U.S. Gold looks for developers who are proven or with whom he has worked

Table 2. A Game Developers Glossary of Terms

Affiliated label	A developer who shoulders most of the monetary burden except gaining access to a publisher's distribution channels, while keeping its own name on the box. Marketing assistance is negotiable, and independence is possible after developing a brand identity.
Bundling	Including another product in a package, typically software with hardware. For example, a Virgin Games 7th Guest game was bundled with with Media Vision CD-ROM drives, Sonic cartridges were bundled with with Genesis consoles, and Crash 'N Burn cartridges were bundled with 3DO consoles.
Copublish	See "affiliated label."
FOB	Freight-on-board indicates who pays shipping and how it is calculated. For example, "FOB Kobe" means the developer pays shipping across the Pacific.
Inventory	Inventory is counted in dollars (wholesale), units, or days (before the games will probably get sold).
Listing fee	What a mail order catalogue charges a publisher to include a product.
MDF	Market development funds; also called cooperative advertising.
OEM	Original equipment manufacturer. See "bundling."
POP	Point of purchase. Can refer to promotions ("impulse buy") or data collection.
Port	A direct conversion of the a game title from one platform to another.
Price protection	A distributor can collect the difference between its original purchase price and what the product has been discounted to.
Program	An arrangement between a distributor and either a vendor (publisher) or reseller (retailer) that typically includes minimum sales volumes, return conditions, prices, or some rack jobbing. Programs are more prevalent in game software than business applications due to lower per-unit prices, $45 for a game vs. $250 for WordPerfect.

before. If it's time to release a new race game, whoever did his last one can be trusted. Mary Bihr of LucasArts says you need developers who accept creative direction, work to specification, and are reliable—but most important of all is "to have an extremely creative development team who understand on a gut level what is fun."

Either party can evolve into the other. U.S. Gold is building an in-house development team. Maxis grew from two men in a garage to a Broderbund-affiliated label to a publishing house with its own affiliates. On the other hand, Sculptured Software reports no such publishing ambitions.

Developers

Developers only deal directly with platform providers and publishers. PC game developers can still conceivably publish themselves, but the industry has matured well beyond the point where it is advisable. Video game designers must deal with either a platform provider, a licensed third-party publisher, or at least a design house. 3DO offers the only exception.

The wealth of video game platforms is a mixed blessing to developers. Older established houses like Sculptured Software see an opportunity to perform more ports, conversions of the same title from one platform to another. Younger firms like Alexandria face developing on what could be a losing platform: "The difference between 32-bit and 64-bit starts becoming meaningless. It's 'The Battle of the Bits.' I'll be so happy when the technology reaches the state where the output of the machine will allow TV-like quality. Then the consumer won't care about the box, and we can get down to creating the content: that's what we're all waiting for." Ken Balthaser probably speaks for most developers.

Among PC games, there is a crop of new mini-platforms. Microsoft conducted free developer conferences around the country recently to jumpstart multimedia title development. With its hobbyist origins, Apple has always devoted significant resources to supporting developers of all varieties of applications. When Apple declared it intended to seed the market with CD-ROM drives at cost, it was good news for everybody, except CD-ROM-drive distributors.

The relationship between developer and publisher is probably the most significant for readers of this magazine. It can be an uneasy alliance. We know what publishers want from developers and why. Now let's explore how developers feel about their publishers, how they ought to deal with them, exactly what publishers do to earn their keep, and (something every developer wonders) is there any alternative? Table 2 shows a glossary of terms used in distribution channels, that is, by publishers and each different distribution channel between them and the consumer.

"We've always agonized over marketing people's actions," confides Darren Bartlett of the Illusions Gaming Company, a cartridge developer. "They

change our names. They change our audience. We developers are a little older now than our target audience. They like cute animals. We like to blow things up. But our next game is about an autistic kid in a straitjacket." Mutual respect is the norm.

Remember three things when dealing with publishers. First, decide how much publisher you want: a half, a whole, or none at all. Second, limit all agreements with publishers and distributors by geographic region; don't give up overseas rights, for example. Third, limit all agreements by platform; you wouldn't want a cartridge publisher handling your DOS port.

Copublishing or affiliated label arrangements are the half-a-publisher option, allowing developers and publishers to do what each does best, in theory. Maura Sparks of Pop Rocket, an Electronic Arts affiliated label, describes their relationship as "perfect for a small company, teaming up with all that marketing muscle." Developing exclusively for CD-ROM, the four-person Pop Rocket is looking to single-handedly create a music-video-adventure genre.

Three good reasons to copublish are:

- You plan to become a publisher someday and want to develop a brand identity
- You can afford to invest all the marketing, production, inventory, and service costs up-front
- You want to maintain control and ownership of the product.

Copublisher's royalties equal their publisher's. Alternatively, they might agree to sell to the publisher at a discount 5% to 10% better than they give any other distribution channel.

Drew Huffman offers the following advice to anyone negotiating with a publisher on the strength of his experience releasing Iron Helix through Spectrum HoloByte:

- Ask for help from family, friends, associates, lawyers, distribution consultants, and even competitors.
- Match the product to the publisher.
- Decide what you want, including the absolute minimum you will accept, and stick to it.
- Don't lose your cool; it's the publisher's job to bargain with you.
- Understand what it is that you don't know. Admit that to yourself, make a note, and find out later. Never commit to anything until you understand.
- Never make a decision for the wrong reasons.

For CD titles, former Broderbunder Leigh Marriner recommends the following affiliated label programs: Compton's NewMedia, Sony, Apple, Electronic Arts, Maxis, Knowledge Adventure, Davidson, Edmark, Magic Quest, Paramount Interactive, Accolade, and Broderbund.

The whole-publisher approach is far more common. Sculptured Software controller Shelley Dahl says, "A lot of the design takes place in-house, but we work closely with a publisher. After all,

Table 2. A Game Developers Glossary of Terms

Rack jobber	A distributor guarantees a dollar sales figure per linear foot of shelf space for retailers and performs merchandising like shipping, counting old and new stock, restocking, assuring proper placement, and stickering.
ROP	Run-of-press, includes newspaper advertising, Sunday inserts, and so on.
Sell-in/sell-through	Sell-in is filling the distriction channel by getting the product out the publisher's door. Sell-through is consumers removing it from the distibution channel, hopefully not temporarily.
Sell sheets	Descriptions of products written by publishers for retailers.
SKU	Stock-keeping unit. This is how everyone in the distribution process keeps track of product and includes a specific title-medium combination. The same game title for Genesis, SNES, Game Boy, and Game Gear would have four different SKUs.
To SKU	To formally introduce a product to the market. This entails assigning a product a unique UPC code, listing a suggested retail price, and indicating a national street date with a time frame for release.
SPIF	Sales performance incentive fund is a wholesale rebate paid to distributors for each unit sold through to the next distribution level, typically $1. Retailers are not usually "SPIFfed."
SRP	Suggested retail price.
Stock balancing	The right of a distributor to exchange older, slow-moving products for newer units from the same publisher.
Street price	The price a consumer can get with a little looking, usually about 25% less than the SRP.
Terms	Specific payment and delivery agreements.
Top of mind	A marketing metric indicating how much publishers, distributors, and retailers are aware of a product's existence.
Turnover	Indicates how quickly inventory is replaced; divide retail sales by average inventory—the higher the better.

Sega's Sonic the Hedgehog CD

it is their product." Indeed, most publishers originate 85% to 90% of their releases. As Cary Hammer of Unexpected Development explains, the average video game "starts with a publisher purchasing a license. They may then ask a developer to 'do it all' or to execute their own in-house design, to execute another party's design, or just to do a straight platform port."

Publishers rarely accept proposals from developers and even more rarely accept entire games done on spec. But sometimes a publisher will approach a developer with a license and ask if it has a game ready and waiting? They don't just mean an engine but a whole game, where the graphics are added, and the game is done in a matter of weeks. Konami is one exception. Dealing exclusively with Japanese teams, it accepts only finished spec games from U.S. developers, but it no longer publishes PC games.

The mechanics of publishing are simple once a project is on track, but it is imperative to maintain close communication with all channel partners. Coordinating the logistics of product launches nationwide is at least one full-time job. The following prescription for a product launch, combining input from half a dozen publishers, illustrates some of the ways publishers earn their keep.

Publishers start contacting distributors at least six months prior to product release. A good way to think of it is one to two trade shows before release. So for a Christmas product, supply tidbits at the January and June Consumer Electronic Show. The dialogue starts in earnest three to six months before shipment. The publisher has alerted the channels to the new title, and you must assign it a stock keeping unit, a UPC code, street price, short description, and a ballpark time frame for when it will hit the street.

U.S. Gold's Botch calls it a case of "Hollywood-style counter-programming; announce for the Fall, just not when exactly," as publishers jockey for position against each other's releases. Nobody wanted to release a game too close to Mortal Monday, September 13, 1993, for example. The distribution channel will need sell sheets explaining the product and empty boxes for advertising photo shoots. The publisher spreads demo disks around to buyers, reviewers, and user groups. It makes plans for advertising and promotions in stores, magazines, at special events, with coupons, and whatever it can think of. Six weeks before shipment, the publisher takes a look at the market trends and competitive environment.

Hot-B USA's Brad Berglund says you must ship by the end of October or early November to be on shelves by Thanksgiving, the start of the Christmas buying season. He and Botch agree that the Christmas quarter is as good as any two quarters' sales, 50% to 60% of the entire year. Botch adds that January and February are good too, with people buying titles for the hardware systems they got over Christmas. Summer is the worst, thanks to longer daylight hours, vacations, and travel.

A typical figure for market development funds is 3% of wholesale volume,

deducted off invoice. Sales performance incentive funds can be used to push the product further into the distribution channel. Remember, it's the sales department's job is to push a product into the distribution channel, and the marketing department's job is to pull the product through. It can do that through advertising, promotions like contests, coupons, favorable press reviews, user-group-generated word-of-mouth, and of course a high-quality product starting with the box. Maxis sales director Ileana Seander says the two biggest factors affecting store buyers' decisions are their relationship with the publisher and the box. But don't go too far. Everyone in the distribution channel serves a prequalifying function. Handleman's and Tandy will not carry games with overtly sexual, violent, or occult box art.

Gametek's Gordon Walton also says product-centered advertising is a waste of money. If you must advertise, focus on company image and use home-buyer media or magazines like *Computer Gaming World*, *Strategy Plus*, and *PC Entertainment*, not trade, general interest, or computer industry magazines. Co-op ads in direct mail catalogues may be worth the money if mail order is a chosen distribution channel. Nintendo was thought to have spent $100 million promoting all titles this Christmas. Promoting Mortal Monday reputedly cost Acclaim $5 million. Your results may vary.

What are your alternatives? Self-publishing is not recommended. Says Marriner, "Breaking into the retail channel is extremely difficult. It takes time to work distribution deals. Going it alone means less distribution. And, because you are marketing your company at the same time as the product, it will cost half again as much. But, most important is the pace of technology in this arena. By the time you get distribution, your product may no longer be state of the art."

Distribution consultant Solange Van Der Moer of Infinity Partners paints a brighter picture. "Money still talks; you just have to turn up the volume. Match your product to the right target audience. Think out of the box. You're providing information; it's just a different medium." She likens Maxis to Wyndham Hill Records: "Over 12 years, they created the New Age category of music; the name of each new artist was irrelevant. You've already established an ongoing metaphor, and the consumer wants the next chapter." For distribution work-around, she finds potential for multimedia placement in museums, sporting goods stores, and even running the PC blockade at Toys R Us.

Three other viable distribution channels are worth considering.

Mail order accounts for 12% to 15% of IBM-compatible PC game sales but 40% to 50% for the Macintosh, of which 90% goes through three houses: Mac Zone, Mac Warehouse, and Mac Connection. "They will ask for 'favored nation' pricing, but don't give it to them," warns Ingram/Micro's Davis. "Keep a level playing field across all channels." Multimedia publishing consultant Robert Risse of New Media Partners notes catalogues charge buy-in fees of $2,500 to $10,000 just to be listed and will request even more for advertising within their pages.

Shareware works. Take it from Scott Miller at Apogee, the bulletin board proprietor and shareware publisher that brought Wolfenstein 3D to market. It has sold over 20,000 copies since May 1992, generating over $1 million. Apogee has more than 20 titles, one to two dozen under production, 70 people collecting paychecks, and design teams from Redmond, Wash., and Canada to London and Sydney. Miller is particularly excited by a variant trend he calls "rackware," which he considers free distribution through thousands of grocery, stereo, and software stores as well as Wal-Mart, K-Mart, and Radio Shack. "All you need is a disk duplicating machine and screen shots for them to put in the bags and on the boxes."

Electronic distribution is another topic that deserves its own article, but look for The Sega Channel soon, allowing game downloads one at a time into a special Genesis peripheral. Network games are viable already, however, and they allow group play. The main networks, their owners, and subscriber bases, according to *The Wall Street Journal*, are CompuServe (H&R Block, 1.4 million), Prodigy (IBM and Sears, 1 million), GEnie (General Electric, 400,000), and America Online (350,000).

GEnie has been the most aggressive of the big networks in courting game players, lowering its on-line charge to $3 per hour. The Multi-Player Game Network (MPG-Net) has around 10,000 subscribers. Delphi and National Videotex are in transition. The ImagiNation, formerly The Sierra Network, has 30,000 to 40,000 subscribers. Director of marketing Jeff Leibowitz says, "We are a games network, but games aren't really the point: it's playing games not against the computer but against other people. It can be a difficult message, because people don't want to hear that they're lonely."

Assuming you stay with the more traditional distribution channels, what payoff can you expect? Consider terms, feedback, royalties, and margins.

Terms. The vast majority of payments are advances against royalties, usually guaranteed so a developer will not end up owing anything if the product dies in the market. A schedule of eight payments is common: six milestones bracketed by signing and completion or approval payments. Weighting can be equal, heavier at signing and delivery stages, backloaded toward delivery, or otherwise.

A returns fund is maintained until a product is retired. Out of sales to a distributor, 20% may be held by the distributor for returns. This fund is adjusted every quarter, paid down over time, and closed out when the title is finally removed from distribution.

Whatever your publisher threshold was, Robert Risse of New Media Partners lists critical details you would be wise to either expect or insist on seeing in a contract:

- Publishers' offers often come in the form of a term sheet. This is a good thing, pinpointing key terms without any confusing legal terminology.
- Sublicense fees should be computed at a higher rate than royalties. Your royalties on a publisher's royalties can be up to 33%; for ancillary products like

T-shirts and TV series, 7.5% to 10% is typical.

- Recoupment of the advance is negotiable; a recoupment percentage of 50% means developer royalties are paid out in cash as quickly as they are applied to pay down the advance.
- Right of first refusal on future titles should be diluted to a right of first negotiation.
- Payment schedules are negotiable. Typically developers are paid 30 days after publishers, which could amount to 120 days without cashflow. Advances might be made bigger, or payment can be adjusted.
- Ask for a commitment on how much the publisher will spend on advertising and promotion.
- After verbal agreements are obtained on key points, call in the lawyers. And read what they draw up to make sure it agrees with everything you negotiated.

Feedback. PC games can expect a window of sales no more than six months before being discount-racked. Cartridges need to survive 60 days, and some last for years. LucasArts's Bihr describes the market as "pretty aware; it has a quick adoption rate. We hope to recoup with the first shipout both marketing and development costs." Sometimes it takes two or three months after release to know if a product will succeed. Other times, in the case of X-Wing, the entire initial shipment of 100,000 was yanked through the channel, blowing out all forecasts and inventory, over the first weekend. Sell-in from March to December 1993 approached half a million, without even a Christmas.

Royalties. Ballpark royalties are 7% to 15% of wholesale, or $1.00 to $3.00 per unit; they can be computed as a percent of sales or in dollars per units sold. Name developers get more. For example, a $50 suggested retail price game with a street price of $35 and wholesale revenue of $18 may return $2. A copublishing developer would get closer to $9. A late developer might be penalized a point or two, so the rate might not be known until a project is completed. Sometimes there is a bonus for high volume sales.

Margins. Handleman's demands at least 60% off suggested retail price. Tandy/Radio Shack expects 62.5% if not 65% off retail, but will not ask you to accept returned products. Others average 55% off street retail. Street prices at Electronics Boutique and Software Etc or Babbages may dip 20% from $49 to $39, for example, affecting your wholesale revenue. The toy business is enviable, where all calculations are made off landed cost. Overall, retailers and distributors count on 10%, according to Jeff Davis of Ingram/Micro. Compton's NewMedia takes 50% off suggested retail price for distributors, 45% for CompUSA, and 40% for retailers.

The Future

I see three trends playing themselves out prominently in 1994, and two more operating in the background.

CD-ROM. 1994 will be the year of the CD-ROM. Cheaper and easier to manufacture than floppy disks or cartridges, each one can hold an unprecedented 600M of data. But the technology is imperfect; data transfer rate is bad, and seek time is worse. Even double-speed and triple-speed drives are slow. David Walker, a technical director at Electronic Arts and co-founder of the Computer Entertainment Developers Association, likens it to the data cassette of 10 years ago. By Eric Goldberg's standards, it has already failed to be adopted by 50% of the populace within seven years, as the VCR, microwave and audio CD have been.

Nevertheless, the installed base of PC and Macintosh CD-ROM drives is estimated to have grown from half a million in 1992 to 6 million at year-end 1993 and to reach up to 20 million by the end of 1994. That would approximate today's installed base of 16-bit video game consoles.

Tom McGrew reiterates the kind of success Compton's NewMedia is enjoying with nongame multimedia titles: "This is not the book business, where 25,000 hardbacks is a success. We're preselling 15,000 to 18,000 CD-ROMs at a time. CompUSA already has 120 CD titles on the shelf, and when we ask if they would be interested in another, they just say, 'Ship it.' "

Capstone's Angie Niehauf thinks "multimedia is what is going to change everything. Blockbuster and Sound Warehouse's rental of software just like cartridges is expanding pretty fast. Everybody's looking for CD product. The retailers are asking for it. Let's see if the consumers are as excited."

Rental. Rental may be the way to jump-start new hardware standards. Blockbuster and other major video rental outlets will be renting cartridges and CD-ROMs as well as their players, including 3DOs, Jaguars, and Sega CDs. Nobody knows the net effect to date of cannibalization of potential cartridge sales vs. heightened awareness, but making state-of-the-art platform units available will raise the floor of installed bases if nothing else.

Interestingly, Electronic Arts has already experimented with a rental-only Sega cartridge. The John Madden Championship Edition was later rereleased exclusively via direct mail. If you consider arcades a sort of time-rental, then count Electronic Arts in on this trend yet again. They formed a coin-operated division in August 1993.

Platforms. What a war there will be in 1994 over platforms. Regardless of which ones survive, any hit games that materialize will be ported to the others over time. Therefore, in the long run, consumers will only be inconvenienced. Retailers and distributors are indifferent. For the developers, publishers, and providers, it could be deadly. So let's squint once again into the horizon.

"The Jaguar lets you bang on the hardware and blow by the OS," says Atari's Brodie. "It is a 55 MIPS computer with five processors, including a 64-bit RISC, 32-bit RISC, and 68000 chip just for joystick control and configuration setup." One advantage the Jaguar clearly has over the 3DO is a cartridge slot alongside its upcoming CD add-on. Plus, the Jaguar is still a cartridge machine, capable of assimilating new technology as T&E Soft's John Eaton described. An MPEG decompression cartridge for full-motion video will be available sometime this year.

3DO initially followed Philips' CD-I into 2,000 to 3,000 stereo stores like The Good Guys, Circuit City, and Silo, positioned as a consumer electronic device. It suffered there. Disappointing from a sales standpoint, the distributions channel strategy may have helped establish 3DO as more than just a game box. A respectable first impression safely made, it is now sold at Capcom's video game retail subsidiary G&G/Captron, among others, and will be rented at Blockbuster Videos.

Sega's 32 or 64-bit CD game project, known as Saturn, is still an unknown quantity, but Alexandria's Ken Balthaser ventures an opinion sight unseen: "I have to bet on Sega, just because of their track record. They've proven they can create new technology, they can introduce it, they can take the software and launch it. The question is, when will they launch. Like anyone with a successful product in the marketplace, you don't want to kill the goose laying the golden eggs. So near term, two to three years: Sega. Long term: Sony or Matsushita, the big ones. When it becomes a commodity in the marketplace like TVs, they're going to come out winners. You've got to go with the people who can produce mass quantities at low prices."

SNK will release a NeoGeo CD attachment by the end of 1994, shortly before Sony's CD game machine appears. Nobody's willing to guess when or if Nintendo's 64-bit Reality Engine will appear.

On the PC side, 80% of all software development in the DOS world is for the Windows environment. Yet the SPA reported only $3.7 million in Windows game sales out of $68.9 million total home entertainment software sold during the second quarter of 1993. Games will have to catch up.

Two less obvious trends will subtly affect the U.S. game market in 1994.

Overseas Markets. The worst thing that can happen in publishing is to push too many units into the distribution channel, according to Denny Thorley of Extreme Entertainment. It killed the Atari game industry, and even the 8-bit NES suffered a look-alike library in the late 1980s. Europe now offers a safety valve. Unexpected Development's Cary Hammer, developing primarily for portables and the SNES, says the European market is taking off; Game Boy titles do even better in Europe than in America. With distribution favoring hits in the U.S., overflows have a place to go without bankrupting publishers or leaving distribution-channel partners holding stale inventory.

Impending Shakeout. Even with expansive new game markets, there's only so much room at the top. The casualties may not hit the ground until 1995, but Jeff Davis at Ingram/Micro already sees "way too much product looking for not enough distribution." Everyone agrees the industry is being hits-driven by licenses and marketing. Neither is necessarily related to gameplay quality, yet that is the keystone of this industry. The lessons of Atari and the 8-bit NES have been learned, but who knows if they'll be remembered.

Dick Larkin of Hudson Soft voices the wisest publishers' concerns when he says, "The challenge is how do you make it fun? That's something we work on and work on and work on, and after all the pieces are put together, sometimes it's still just not there." It was there for SuperBomberman. The multiplayer hit game attracted unprecedented interest in TurboTechnologies's Duo and TurboGrafx-16, which Hudson Soft parlayed into exclusive rights to a five-player SNES attachment. Their Multi-Tap is now bundled in a Nintendo SuperBomberman Party Pak.

The man behind Sid Meier's Civilization put it simply at the April 1993 Game Developers Conference. Interactivity is the special quality that computer games offer—not sound, not music, and not graphics. So given a limited budget, spend it on developing gameplay.

Cary Hammer feels the labor pool of experienced developers has grown big enough that he expects to see more spec development of cartridge games. Ileana Seander of Maxis admonishes any such developers to take the time to conduct predesign: "Look at the market; talk to retailers and focus-group consumers." Alert to the irony of her message coming from a company founded on the original game-without-a-category, SimCity, she adds, "The market has changed since five years ago. You have to know the competition, pricing, and just how limited shelf space is out there." Today, Maxis owns a large share of the Windows and an even bigger share of the Macintosh game markets. Don't expect her to give them up easily.

Distribution and You

Your ideas can become dollars. That's the good news. The bad news is that the mix of inspiration and perspiration is no longer just between you, your muse, and the latest authoring tools. Everybody's silver bullets must cross an economic landscape on the way to the firing range on a journey that is more and more by invitation only. To negotiate this terrain, it will help to have learned about the inhabitants and their perspectives, the rituals they follow and why, and what looms on the horizon for the entire ecology.

Packard's Law still holds true. Every 18 months, computer chip price performance increases tenfold. That empowers the electronic entertainment industry with a dangerous ability to recreate itself every other year. It is happening again. The PC world keeps splintering across 286s, 386s, 486s, and Pentiums, with and without multimedia CD, sound, or video capabilities. Ken Balthaser could just as easily be speaking for PCs, as he views a similar landscape on the video game front: "It's a watershed time, a transition period. 1993 was the last Christmas for 16-bit. Thirty-two is coming." ■

James Paul Cooper sold decision support software to institutional investment managers for a global consulting firm. In 1978, he joined an Arizona computer game company instead of matriculating at the University of California. Since then, he has founded or co-founded several game and software ventures and acquired an M.S. in Industrial Administration from Carnegie Mellon University, where he taught marketing. He can be reached on Internet at 72147.2102@compuserve.com or through Game Developer *magazine.*

Drew Pictures: The Making of Iron Helix

by Alexander Antoniades

On the road from video production company to interactive entertainment developer, you can travel to some strange places, from an old liberty ship in San Francisco to a trip to the local butcher shop.

If Drew Picture's game Iron Helix had a birthplace, it would be the Jeremiah O'Brian, a floating museum located in the Marina district of San Francisco, Calif. It was there that the big idea of Drew Pictures founders Vincent Carrella and Drew Huffman finally clicked. It was an idea that had its roots from when Carrella and Huffman were working at Paracomp.

Huffman was working as product manager for a computer animation program called Film Maker. Carrella was a strategic marketing person who did hands-on demonstrations on roadshows with Huffman. It was during these roadshows that Carrella and Huffman spent a lot of time together and talked about their future plans. Huffman wanted to start his own production company.

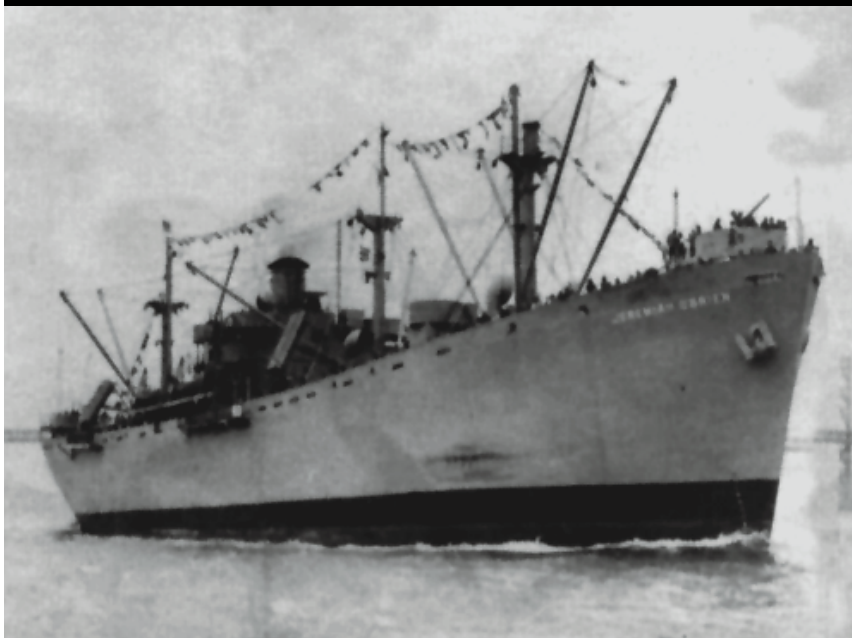
Huffman eventually left Paracomp to start Drew Pictures. A year later, Carrella came on board. The company's main focus at that time was video production and animation for commercial television and a variety of hi-tech clients. That all changed when the company was asked to work on a module of a large CD-ROM project and came upon a great idea.

The Big Idea

Drew Huffman's original concept was designing a game around the limitations of the CD-ROM. By storing the main program on the hard drive, the designers could use the massive storage

The main interface screen of Iron Helix was the final result of a balancing act between technical and gameplay considerations.

Figure 1. The Original Jeremiah O'Brian



potential of the CD-ROM to store sound and video the program could access as needed.

One of Huffman's original ideas involved piloting a robot around a three-dimensional virtual environment with a view through a small window. The challenge was to justify the small window size. Drew Pictures designers developed a storyline where the robot was exploring a large ship that was too dangerous for humans to explore. This led to an interface design in which the player is controlling a robot from a safe vantage point, making what the robot sees in the view screen just a small part of the overall scenario.

While Huffman continued working in video production, Carrella started investigating the Macintosh CD-ROM market. They made a favorable business plan in which he stated that based on the success of CD-ROM games such as Spaceship Warlock and Reactor, the Macintosh CD-ROM market was potentially profitable. However, Huffman and Carrella still weren't sure, because of technical limitations, developing a CD-ROM was the right thing to do.

There were two main objectives Huffman wanted to accomplish in his game:

- To get smooth playability off a CD-ROM drive
- To suspend disbelief that someone was playing a game.

The CD load times were very frustrating, but they now knew they could overcome that limitation. The release of the Quicktime 1.0 animation engine for the Macintosh promised consistent playback of video animation and finally convinced the pair that the game was technically possible.

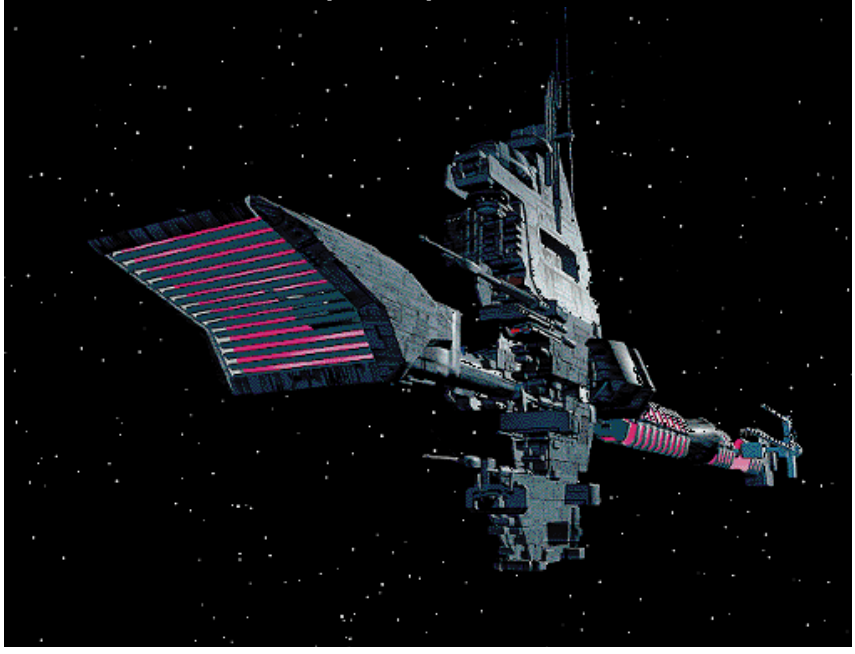
A visit to the Jeremiah O'Brian, shown in Figure 1, finally made all the pieces fit. The atmosphere on that historic ship gave them the feeling of what they were trying to convey in the game. They even thought of doing a documentary about the Jeremiah O'Brian. But in the end, they decided to go into the game business instead of testing the treacherous waters of educational multimedia. They did name the game's main spaceship after the ship in homage.

Inspiration and Perspiration

They decided to use Macromind Director from Macromind as the authoring tool to create the game. J.A. Nelson, an old associate of Huffman, joined the team as its Director expert. Rich Cohen, a technical director at Industrial Light and Magic, also joined the team as art director.

For the next year and a half, everyone worked out of Huffman's apartment, trying to build a game and a game company. Their experience showed them how important it is to understand business and not just game development to become a successful company. To learn what they were getting into, they

The Jeremiah O'Brian Spaceship



talked to professional game developers and distributors.

It was about three months into development that they realized Quicktime 1.0 wasn't going to cut it. It could not maintain the consistent frame rate the designers wanted. What they ended up using in the final game was an animation engine called PACo from The Company of Science and Art. While Quicktime isn't used in the final version of Iron Helix, the designers insist that the game couldn't have been made without it.

Instead of writing the story first and then developing the game, they developed both in conjunction, allowing themselves to make adjustments as they ran into technical difficulties. One of their major complaints about CD-ROM games was the character interaction. One of their pet peeves about CD-ROM games is that the CD-ROM is too slow to have live video interactions. To get around this limitation, all the characters in the game are dead—the player views video clues they have left behind.

As they worked more on the game, eventually the whole plot was fleshed out. The story revolves around a runaway spaceship containing a biological weapon and a doomsday device. The player controls a robot probe that is piloted around on a spaceship. All the crew is dead, so the player looks around the ship for pieces of the crew's DNA, which is used to gain access to more of the ship. However, every time certain parts of the ship are accessed, the Defender robot patrolling the ship will become aware of the player's presence. The player accesses video clues left behind by the former crew, which show how to destroy the Defender robot and the dangerous spaceship.

The Darwin Probe

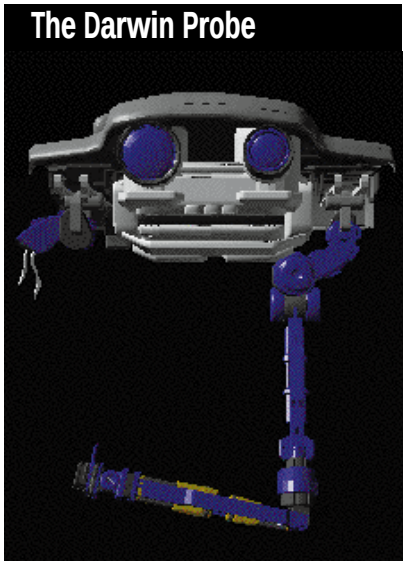


Figure 2. Meat or Microbe? You Make the Call!

Iron Helix

The way the story was written, most of the animation takes place inside two windows on the screen. The primary window contains the robot probe's remote camera view, and the other contains the interface where the player views videos and the DNA scans, among other things.

The images in the primary display window are all computer-generated three-dimensional images. The tools used to make these were Macromedia's Swivel 3-D Professional and Macromodel. Every segment was animated individually in development and is pulled off the CD-ROM in real-time when the probe is moved in the game. The original plan called for the probe to be able to tilt and pan in its movement, but it was too much work, and the developers felt that it didn't enhance gameplay. In the end, the probe would be able to turn at 90º angles and move at fixed increments with transitional animation linking every move.

Of course, in the instances where the Defender kills the probe, there must be additional animation sequences. The developers calculated all the places where the player could possibly die and decided to only include the most probable scenarios. In the end, out of 600 locations where the player could die, they made 250 unique death sequences and a generic death scene for the rest.

The secondary animation window in the main interface screen displayed, among other things, the video images of the dead crew. For those video pictures, they used mainly the Drew Pictures staff and friends as actors. They filmed the sequences in the office and later changed the backgrounds with Adobe Premiere.

The Creative Process

In the part of the game where the player has to find the dead crew's DNA, a close up scan appears in the same window as the videos. The original idea was that the player would manually match the DNA of the dead crew members. Carrella spent two months studying DNA matching. He got out all his old biology texts and even spoke to an FBI lab. In the end, he found that real DNA matching was too hard to simulate, so that any simulation would be so simplified it wouldn't have any educational aspect, which was one of the main reasons he wanted to do it.

Making the actual image for the DNA samples, shown in Figure 2, was

An Unrendered Swivel Picture of the Bridge

another story. Carrella was fond of electron micrographs and wondered how he could get the same effect from photographing other things. He photographed hunks of meat and chicken innards and used a program called TextureSynth to make the background, mixed them in Adobe Photoshop, and threw in other touches to make the shots look authentic.

Early in the development process, the designers started looking for a distributor because they knew the distributor would be crucial to the game's success. They sent Gilman Louie, chairman of Spectrum Holobyte, a description of what they were working on. Louie was interested, but he could not negotiate with them at that time. Five months later, after they had talked to Electronic Arts, Sony, Broderbund, and Educorp, Drew Pictures negotiated the distribution contract with Spectrum Holobyte.

Spectrum Holobyte helped the team jump a big hurdle in its development process by taking one of the alpha versions to a focus group. It was in that focus group that many of the playability features were corrected. In the original game, the robot probe had to manually open a door, which wasn't very popular with the playtesters. As a result of that meeting, the designers added more DNA , more levels of difficulty, and the ability to momentarily jam the Defender. The focus group consisted of hard-core gamers as well as novices to address the games impact on a larger audience.

Money and Markets

Oddly enough, the Japanese version of Iron Helix was available before the English version. This was because Spectrum Holobyte was more demanding than Drew's Japanese Distributor. The whole game remained in English for the Japanese version, as did the French and German versions, with only subtitles for the specific foreign language. The foreign version was distributed as a test only. Currently, the foreign CD market is too small, and foreign sales account for less than 10% of Drew Pictures's total profit.

After the team successfully completed the Macintosh version of Iron Helix, it started work on a PC version. The designers chose to do it for Windows instead of DOS, so they wouldn't run into driver problems with the DOS CD-ROM drivers. One problem they did run into, however, was that the performance of the Windows version of the Director player wasn't good enough to use for the game. It was at this point that Bill Zettler, their director of engineering, programmed his own engine in C. The designers will continue using Director for development and are working with Macromedia on the speed issue.

Despite problems with the environment, the Windows version of Iron Helix outsold the Macintosh version in its first week. "Each version is a refinement," says Carrella, "The PC version is a better game than the Mac version." Although, he stresses that it's just the gameplay that's better, not the experience. In his opinion, the PC game market is very finicky; games like Hell Cab and the Journeyman Project will do better on the Macintosh. "There's less room for error on the PC," points out Carrella.

The Future of Iron Helix

The next version of Iron Helix will be for the Sega CD. The designers were thinking of doing a 3DO version, but Sega's large installed base changed their minds. If they do make a 3DO version, they want to write a new version of the game instead of porting the Macintosh version, and the Sega version will be redesigned to be faster. They are not planning to develop a CD-I version of Iron Helix, but the Atari Jaguar looks like a promising platform. They do, however, expect 3DO to become a dominant player in the future.

Since the original success of Iron Helix, Drew Pictures has been working on new projects and looking in new directions. The designers want to make games that combine cinema styles with arcade-level play. Drew Pictures sees itself primarily as an interactive entertainment company that makes games instead of a game company. As Huffman put it, "One thing I can confidently say is that Drew Pictures will never make a shoot 'em up game." ■

Alexander Antoniades is the associate editor of Game Developer *and assistant editor on* OS/2 Magazine.