The Art of Computer Game Design

by Chris Crawford

Preface to the Electronic Version

This text was originally composed by computer game designer Chris Crawford in 1982. When searching for literature on the nature of gaming and its relationship to narrative in 1997, Prof. Sue Peabody learned of The Art of Computer Game Design, which was then long out of print. Prof. Peabody requested Mr. Crawford's permission to publish an electronic version of the text on the World Wide Web so that it would be available to her students and to others interested in game design. Washington State University Vancouver generously made resources available to hire graphic artist Donna Loper to produce this electronic version. WSUV currently houses and maintains the site.

Correspondance regarding that site should be addressed to Prof. Sue Peabody, Department of History, Washington State University Vancouver, peabody@vancouver.wsu.edu.

If you are interested in more recent writings by Chris Crawford, see the "Reflections" interview at the end of The Art of Computer Game Design. Also, visit Chris Crawford's webpage, Erasmatazz.

This document was convert by Mario Croteau, from the Web site of the Department of History of Washington State University at Vancouver.

Chris Crawford (the author) and Sue Peabody (of department of History of Washington State University at Vancouver) gave me a great support in my project: making that important document available to everyone.

The Art of Computer Game Design

by Chris Crawford

Table of Contents

Acknowledgement 1

Preface 1

Chapter 1 - What is a Game? 5

BOARD GAMES 5
CARD GAMES 6
ATHLETIC GAMES 6
CHILDREN'S GAMES 6
COMPUTER GAMES 7
REPRESENTATION 7
Formal 8
System 8
Subjectively Represents 8
Games versus Simulations 8
Subset of Reality 9
Summary of Representation 9
INTERACTION 9
Games versus Puzzles 10
Games versus Stories 10
Games versus Toys 11
Significance of Interaction 12
Nature of Interaction 12
CONFLICT 13
Games without conflict? 13
Summary of Conflict 14
SAFETY 14
Summary of Safety 15

Chapter 2 - Why Do People Play Games? 16

Fantasy/Exploration 17
Nose-Thumbing 18
Proving Oneself 19
Social Lubrication 20
Exercise 20
Need for Acknowledgement 20
Summary 21
MOTIVATION VERSUS SELECTION 21
Game Play 21
Sensory Gratification 22
INDIVIDUAL TASTES 22

Chapter 3 - A Taxonomy of Computer Games 25

SKILL-AND-ACTION GAMES 25
Combat Games 26
Maze Games 27
Sports Games 29
Paddle Games 29

Race Games 30
Miscellaneous Games 30
TRATEGY GAMES 30
Adventures 31
D&D Games 32
Wargames 33
Games of Chance 34
Educational and Children's Games 34
Interpersonal Games 34
CONCLUSIONS 34

Chapter 4 - The Computer as a Game Technology 36

GAME TECHNOLOGIES 36
COMPUTERS 38
DESIGN PRECEPTS FOR COMPUTER GAMES 41
Precept #1: GO WITH THE GRAIN 41
Precept # 2: DON'T TRANSPLANT 43
Precept #3: DESIGN AROUND THE I/O 43
Precept #4: KEEP IT CLEAN 44
Precept #5: STORE LESS AND PROCESS MORE 45
Precept #6: MAINTAIN UNITY OF DESIGN EFFORT 47
CONCLUSION 48

Chapter 5 - The Game Design Sequence 49

CHOOSE A GOAL AND A TOPIC 49
RESEARCH AND PREPARATION 51
DESIGN PHASE 52
I/O Structure 52
Game Structure 54
Program Structure 57
Evaluation of the Design 57
PRE-PROGRAMMING PHASE 58
PROGRAMMING PHASE 58
PLAYTESTING PHASE 58
POST-MORTEM 60

Chapter 6 - Design Techniques and Ideals 63

BALANCING SOLITAIRE GAMES 63
Vast Resources 63
Artificial Smarts 64
Conclusions on Artificial Smarts 68
Limited Information 68
Summary 68
RELATIONSHIPS BETWEEN OPPONENTS 69
Symmetric Relationships 69
Asymmetric Games 69
Triangularity 70
Actors and Indirect Relationships 71
SMOOTH LEARNING CURVES 72
THE ILLUSION OF WINNABILITY 73
SUMMARY 74

Chapter 7 - The Future of Computer Games 74

FAD OR FIXTURE? 74

THE TECHNOLOGICAL EXTRAPOLATION 75

ASSESSMENT: TECHNOLOGICAL REVOLUTION 76

THE NATURE OF CHANGE 78

The Mass Market 78

The Flowering of Heterogeneity 79

CONCLUSIONS 80

Chapter 8 - Development of Excalibur 81

BEGINNINGS 81

EARLY WORK: JANUARY-APRIL, 1982 82

THE LONG HAUL: MAY-DECEMBER 1982 84

Interview 85

Web Links 90

ACKNOWLEDGMENT

I am deeply indebted to Madeleine M. Gross for her painstaking and thorough criticisms of this book. In many cases she invested greater efforts into her criticisms than I had put into my original thoughts. She strove to restrain my wild hyperbole and place my arguments on a firmer foundation of rigorous logic. The logical consistency and reliability in this book I owe to her; the speculative flights of fancy must be laid at my doorstep.

PREFACE

The central premise of this book is that computer games constitute a new and as yet poorly developed art form that holds great promise for both designers and players.

This premise may seem laughable or flippant. How could anybody classify the likes of SPACE INVADERS and PAC MAN as art? How can TEMPEST or MISSILE COMMAND compare with Beethoven's Fifth Symphony, Michelangelo's Pieta, or Hemingway's A Farewell To Arms? Computer games are too trivial, too frivolous to be called art. They are idle recreation at best. So says the skeptic.

But we cannot relegate computer games to the cesspit of pop culture solely on the evidence of the current crop of games. The industry is too young and the situation is too dynamic for us to dismiss computer games so easily. We must consider the potential, not the actuality. We must address the fundamental aspects of computer games to achieve a conclusion that will withstand the ravages of time and change.

There are many definitions of art, few of which make much sense to the uninitiated. I will present my own pedestrian definition: art is something designed to evoke emotion through fantasy. The artist presents his audience with a set of sensory experiences that stimulates commonly shared fantasies, and so generates emotions. Art is made possible only by the richness of the fantasy world we share. Art is nevertheless difficult, because there are so many practical problems associated with stimulating fantasies deep inside another person's mind. A major problem is getting the attention or participation of the audience. Most art allows very little participation. You sit quietly and listen to music that other people created and perform, or you stroll through a museum and stare at pictures or statues other people made. You sit passively and read a novel, or a poem, or a short story. With all of these art forms, the role of the audience is passive. The artist does all the active work, makes the biggest emotional investment. The audience is expected to absorb quietly the fruits of the artist's exertions. Active participation is severely curtailed. Without participation, attention dwindles and impact crumbles away.

This is in no wise a criticism of art or artists. The technologies of art preclude participation. If we had every klutz jump into the orchestra pit, or prance on the opera stage, or slop paint with

Picasso, we would have some great parties but no art. it seems the curse of art that artists can say so much in their work and most people will hear so little because they cannot participate in the art.

Enter the computer. Conceived long ago, born in war, reared as the servant of business, this now adolescent technology has exploded out of the computer room and invaded shopping centers, pizza parlors, and homes. Popular characterizations of the computer alternate between the old image of the computer as omniscient, cold blooded, giant calculator, and the new image of the computer as purveyor of video thrills and 25 cent fixes. Originally developed as a number cruncher, the computer assumed a new personality when it was given graphics and sound capabilities. These capabilities gave the computer a powerful asset: it could now communicate with the human, not just in the cold and distant language of digits, but in the emotionally immediate and compelling language of images and sounds. With this capability came a new, previously undreamed of possibility: the possibility of using the computer as a medium for emotional communication art. The computer game has emerged as the prime vehicle for this medium. The computer game is an art form because it presents its audience with fantasy experiences that stimulate emotion.

Unfortunately, the current generation of microcomputers cannot produce a sensory experience as rich as that produced by, say, a symphony orchestra or a movie. This weakness is more than offset by a fundamental advantage lacking in most other art forms: a game is intrinsically participatory in nature. The artist has here a tool that is more subtly indirect than traditional art. With other art forms, the artist directly creates the experience that the audience will encounter. Since this experience is carefully planned and executed, the audience must somehow be prevented from disturbing it; hence, non participation. With a game, the artist creates not the experience itself but the conditions and rules under which the audience will create its own individualized experience. The demand on the artist is greater, for s/he must plan the experience indirectly, taking into account the probable and possible actions and reactions of the audience. The return is far greater, for participation increases attention and heightens the intensity of the experience. When we passively observe someone else's artistic presentation, we derive some emotional benefit, but when we actively participate in a game, we invest a portion of our own ego into the fantasy world of the game. This more sizable investment of participation yields a commensurately greater return of emotional satisfaction. Indeed, the role of participation is so important that many people derive greater satisfaction from participating in an amateur artistic effort than from observing a professional effort. Hence, games, being intrinsically participatory, present the artist with a fantastic opportunity for reaching people.

Until now, games in general and computer games in particular have not been very impressive as art forms. The computer games especially are downright puerile. This is because the technology of computer games has been in the hands of technologists, not artists. These guys (and they are almost all male) can write beautiful operating systems, languages, linking loaders, and other technological wonders, but artistic flair has heretofore been treated as subordinate to technical prowess.

Another contributor to the fecklessness of our current computer games is the timidity of the marketplace. These machines are new; the public is unfamiliar with them and the manufacturers are hesitant to press the public too hard too fast. We therefore opt to build inhibited little games pathetically whispering some trivial emotion. Truly intense emotions or situations such as pathos, ecstasy, majesty, rapture, catharsis, or tragedy intimidate use. We hide behind the defense that we are in the entertainment business, not the art business, but that defense only betrays a profound misunderstanding of art. Art can be starchily elitist, but good art can also be a foot stomping blast. Elitism arises from the intellectual content of art; impact springs from its emotional honesty.

Fortunately, times are changing. Already, we see a backlash developing against computer games. It expresses itself in many ways: in ordinances against the placement of arcade games in some areas, in statements by educators denouncing the games, and in more vigilant regulation of children's game activities by parents. This backlash is viewed by smaller minded members of the industry with anxiety. More visionary thinkers watch the backlash with eager interest rather than defensiveness. The American people are telling us something here, something very important. It is imporant enough to them that they are willing to compromise their traditional reluctance to interfere with other people's business. While the arguments presented in public debates normally focus on formal issues such as delinquency from school, creation of large groups of rowdy teenagers, and so forth, the concerns expressed privately reflect a distaste for the games, a vague suspicion that the games are a waste of time. You can't fool all of the people all of the time; they are beginning to realize that the world of computer games is as yet a vast wasteland.

Computer games are much like candy, comic books, and cartoons. All four activities provide intense or exaggerated experiences. Whether they use sugar, exclamation points, or animated explosions, the goal is the same: to provide extreme experiences. Children appreciate these activities because their novelty value is still strong. Adults, jaded by years of experience with such things, prefer diversions with greater subtlety and depth. We thus have the panoply of culinary achievement, the vast array of literature, and the universe of movies as the adult counterparts to candy, comic books, and cartoons. Yet, we have no adult counterpart to computer games. This deficit is pregnant with possibilities, for it suggests a momentous upheaval in computer game design.

This developing revolution has nothing to do with the rapid technological developments of the last few years. While technological improvements will surely continue, we are no longer hampered primarily by the limitations of the hardware. Our primary problem is that we have little theory on which to base our efforts. We don't really know what a game is, or why people play games, or what makes a game great. Real art through computer games is achievable, but it will never be achieved so long as we have no path to understanding. We need to establish our principles of aesthetics, a framework for criticism, and a model for development. New and better hardware will improve our games, but it will not guarantee our artistic success any more than the development of orchestras guaranteed the appearance of Beethoven. We are a long way from a computer game comparable to a Shakespeare play, a Tchaikowsky symphony, or a Van Gogh self portrait. Each of these artists stood on the shoulders of earlier artists who plunged into an unexplored world and mapped out its territories so that later artists could build on their work and achieve greater things. We computer game designers must put our shoulders together so that our successors may stand on top of them. This book is my contribution to that enterprise.

CHAPTER ONE
What is a Game?

If we desire to understand games and game design, we must first clearly establish our fundamental orientation. We must define what we mean by the word "game." We must also determine the fundamental characteristics of all games. After discussing some of the obstacles inherent in this effort, I will briefly describe the salient classes of games; then I will propose a set of attributes that characterize all games.

Games are a fundamental part of human existence. The parlance of games has insinuated itself into our language to refer to activities that are not truly games. We play along with activities we find distasteful. We play ball with those who require our cooperation. We play games when we are insincere. A willing participant is game for the enterprise. This broad penetration of gaming concepts into the entire spectrum of human experience presents us with two potential barriers to understanding games.

First, our liberal use of gaming terms promotes an exaggerated perception of our own understanding of games. We fail to render unto the subject the careful and critical analysis that we tender to more academic topics, and we blithely ignore the complexities of game design. Complete amateurs whose only relevant skill is programming undertake to design games with no further preparation than their own experience as game players. Those who overrate their own understanding undercut their own potential for learning.

The second obstacle is ambiguity. We have applied the principles and concepts of gaming so widely that we have watered down their original meanings. There is no longer a clear focus to the concepts we seek to understand. Game designers have no well defined set of common terms with which to communicate with each other. Discussions of game design frequently disintegrate into arguments over semantics. To cut through the tangled undergrowth that has grown up around gaming we shall need the bulldozer and the scalpel.

Let us begin this endeavor by stepping back for a moment and taking our bearings. Let us take a brief tour of the universe of games, glancing briefly at each of the major regions. In the course of this tour I hope to refresh the reader's memory of games and make some simple points before digging into the serious analysis of fundamental game characteristics. I perceive five major regions of games: board games, card games, athletic games, children's games, and computer games.

BOARD GAMES

We begin with the board games. These games consist of a playing surface divided into sectors populated by a set of movable pieces. In the most common arrangement the pieces are directly associated with the players, while the playing surface represents an environment beyond the players' direct control. Players maneuver their pieces across the playing surface in an effort to capture other players' pieces, reach an objective, gain control of territory, or acquire some valued commodity. The player's primary concern in these games is the analysis of geometrical relationships between the pieces.

CARD GAMES

A second class of games is the card games. These games utilize a set of 52 symbols generated from two factors: rank (13 values) and suit (4 values). The games revolve around combinations built from these two factors. Players may gain or lose possession of symbols either by random processes or by matching some combination allowed by the rules of the game. Each legal combination is assigned a victory value for final assessment of game results. Players must recognize both existing and potential combinations and estimate probabilities of obtaining the cards necessary for completing a combination. This probability must be weighed against the victory value of the combination. Since the number of combinations is very large, precise computation of the requisite probabilities exceeds the mental powers of almost all players, rendering the game a primarily intuitive exercise. Thus, the player's primary concern in these games is the analysis of combinations.

ATHLETIC GAMES

Another traditional game form is the athletic game. These games emphasize physical more than mental prowess. The rules of the game rigorously specify a precise set of actions that the player is either allowed to execute or required to execute. Skillful use of the body is the player's primary concern in these .games.

We must be careful to distinguish between athletic games and athletic competitions. For example, a race is a competition, not a game. The line of demarcation between games and competition illuminates one of the fundamental elements of all games. I distinguish the two by the degree of interaction between players. Theoretically speaking, the runners in a race do not interact with each other. Each is racing only against the clock; the presence of other runners should be immaterial. In truth, the runners do interact psychologically, for the performance of one runner can affect the performance of the other runners. Furthermore, in some races a runner (or driver or pilot or captain) can physically interpose himself in between the goal and another racer, thereby gaining an advantage. I conclude that the simplest competitions, those in which each person strives to perform some task optimally without direct interaction with the other competitors, do not constitute games but competitions. A competition that does allow interaction is a game.

CHILDREN'S GAMES

Another type of gaming activity is the children's game. Hide and Seek, Red Rover, Tag, and Kick the Can are common examples. Such games frequently take the form of group activities emphasizing simple physical play. Although these games contain simple mental and physical components, their function is not to challenge the child to perform to his or her limits in either domain. Instead, the player's primary concern in these games is the use of social skills illuminating the fundamental role of the group in human life.

A wide variety of children's activities are frequently referred to as games. When a child talks to a strip of bark, maneuvers it, and provides sound effects, we are tempted to refer to such behavior as game playing. For the purposes of this book, I ,exclude such activities from the fold of games. These improvisational games are too ill defined to provide us with any useful information about games.

COMPUTER GAMES

The next area of gaming we shall glance at is the current fad in gaming and the subject of this book, the computer game. These games are played on five types of computers: expensive dedicated machines for the arcades ("coin op" machines), inexpensive dedicated machines ("hand helds"), multi program home games, machines such as the ATARI 2600 and the ATARI 5200, personal computers, and large mainframe computers. The computer acts as opponent and referee in most of these games; in many of them it also provides animated graphics. The most common form of computer game is the skill and action ("S&A") game emphasizing hand eye coordination. These S&A games are frequently violent in nature. There are many other areas of computer gaming: adventure games, fantasy role playing games, and war games. In our cursory overview, these other computer games are eclipsed by the sheer volume of the skill and action games.

This concludes our quick survey of the most prominent groupings in the universe of games. We shall return to the subject later, to create a taxonomy of computer games, and later still to draw on specific examples of games to make points about their nature. We must now address the question which motivated our initial reconnaissance: what are the fundamental elements common to these games? I perceive four common factors: representation, interaction, conflict, and safety.

REPRESENTATION

First, a game is a closed formal system that subjectively represents a subset of reality. Let us examine each term of this statement carefully. By 'closed' I mean that the game is complete and self sufficient as a structure. The model world created by the game is internally complete; no reference need be made to agents outside of the game. Some badly designed games fail to meet this requirement. Such games produce disputes over the rules, for they allow situations to develop that the rules do not address. The players must then extend the rules to cover the situation in which they find themselves. This situation always produces arguments. A properly designed game precludes this possibility; it is closed because the rules cover all contingencies encountered in the game.

Formal

By formal I mean only that the game has explicit rules. There are informal games in which the rules are loosely stated or deliberately vague. Such games are far removed from the mainstream of game play.

System

The term 'system' is often misused, but in this case its application is quite appropriate. A game’s collection of parts which interact with each other, often in complex ways. It is a system.

Subjectively Represents

Representation is a coin with two faces: an objective face and a subjective face. The two faces are not mutually exclusive, for the subjective reality springs from and feeds on objective reality. In a game, these two faces are intertwined, with emphasis on the subjective face. For example, when a player blasts hundreds of alien invaders, nobody believes that his recreation directly mirrors the objective world. However, the game may be a very real metaphor for the player’s perception of his world. I do not wish to sully my arguments with pop psychological analyses of players giving vent to deep seated aggressions at the arcades. Clearly, though, something more than a simple blasting of alien monsters is going on in the mind of the player. We need not concern ourselves with its exact nature; for the moment it is entirely adequate to realize that the player does perceive the game to represent something from his private fantasy world. Thus, a game represents something from subjective reality, not objective. Games are objectively unreal in that they do not physically re create the situations they represent, yet they are subjectively real to the player. The agent that transforms an objectively unreal situation into a subjectively real one is human fantasy. Fantasy thus plays a vital role in any game situation. A game creates a fantasy representation, not a scientific model.

Games versus Simulations

The distinction between objective representation and subjective representation is made clear by a consideration of the differences between simulations and games. A simulation is a serious attempt to accurately represent a real phenomenon in another, more malleable form. A game is an artistically simplified representation of a phenomenon. The simulations designer simplifies reluctantly and only as a concession to material and intellectual limitations. The game designer simplifies deliberately in order to focus the player’s attention on those factors the designer judges to be important. The fundamental difference between the two lies in their purposes. A simulation is created for computational or evaluative purposes; a game is created for educational or entertainment purposes.(There is a middle ground where training simulations blend into educational games.) Accuracy is the sine qua non of simulations; clarity the sine qua non of games.

A simulation bears the same relationship to a game that a technical drawing bears to a painting. A game is not merely a small simulation lacking the degree of detail that a simulation possesses; a game deliberately suppresses detail to accentuate the broader message that the designer wishes to present. Where a simulation is detailed a game is stylized.

Consider, for example, the differences between a flight simulator program for a personal computer and the coin op game RED BARON". Both programs concern flying an airplane; both operate on microcomputer systems. The flight simulator demonstrates many of the technical aspects of flying: stalls, rolls, and spins, for example RED BARON has none of these. Indeed, the aircraft that the player files in RED BARON is quite unrealistic. It cannot be stalled, rolled, spun, or dived into the ground. When the stick is released it automatically rights itself. It is incorrect to conclude from these observations that RED BARON is inferior to the flight simulator. RED BARON is not a game about realistic flying; it is a game about flying and shooting and avoiding being shot. The inclusion of technical details of flying would distract most players from the other aspects of the game. The designers of RED BARON quite correctly stripped out technical details of flight to focus the player's attention on the combat aspects of the game. The absence of these technical details from RED BARON is not a liability but an asset, for it provides focus to the game. Their absence from a flight simulator would be a liability.

Subset of Reality

The last term I use is "subset of reality." One aspect of this term ("subset") is easily justified. Clearly, no game could include all of reality without being reality itself; thus, a game must be at most a subset of reality. The choice of matter in the subset is the means of providing focus to the game. A game that represents too large a subset of reality defies the player's comprehension and becomes almost indistinguishable from life itself, robbing the game of one of its most appealing factors, its focus.

Summary of Representation

A game creates a subjective and deliberately simplified representation of emotional reality. A game is not an objectively accurate representation of reality; objective accuracy is only necessary to the extent required to support the player's fantasy. The player's fantasy is the key agent in making the game psychologically real.

INTERACTION

Some media for representing reality are static. A painting or sculpture depicts a snapshot of reality frozen in time. Some media are dynamic; they show change with time. Movies, music, and dance are dynamic in this way. They are able to represent the changing aspect of reality more richly. But the most fascinating thing about reality is not that it is, or even that it changes, but how it changes, the intricate webwork of cause and effect by which all things are tied together. The only way to properly represent this webwork is to allow the audience to explore its nooks and crannies to let them generate causes and observe effects. Thus, the highest and most complete form of representation is interactive representation. Games provide this interactive element, and it is a crucial factor in their appeal.

Games versus Puzzles

One way to understand the nature of the interactive element of games is to contrast games with puzzles and other non interactive challenges. Compare playing a cube puzzle with playing a game of tic tac toe. Compare the sport of high jumping with the game of basketball. In each comparison the two activities provide similar challenges to the player. The key difference that makes one activity a game and the other activity not a game is the interactive element. A cube puzzle does not actively respond to the human's moves; a high jump pole does not react to the jumper's efforts. In both tic tac toe and basketball the opposing players acknowledge and respond to the player's actions.

The difference between games and puzzles has little to do with the mechanics of the situation; we can easily turn many puzzles and athletic challenges into games and vice versa. For example, chess, a game, has spawned a whole class of puzzles, the end game problems. Games can include puzzles as subsets, and many do. Most of the time the puzzles are a minor component of the overall game, for a game that puts most of its challenge value on included puzzles will rapidly lose its challenge once the puzzles have been solved.

Games versus Stories

Another way to illustrate the role of interaction is to compare games with stories. A story is a collection of facts in time sequenced order that suggest a cause and effect relationship. Frequently, the facts presented are deliberately fictitious, because the facts of a story are intrinsically unimportant. Indeed, the entire concept of fiction ("an untruth that is not a lie") only makes sense when one realizes that the facts presented in the fiction are themselves unimportant. The cause and effect relationships suggested by the sequence of facts are the important part of the story. For example, we care not whether Luke Skywalker and the Death Star really existed. We saw that Luke Skywalker was good and pure, and that the Death Star was evil, and that Luke Skywalker destroyed the Death Star. The cause and effect relationship suggested by the story was that good overcomes evil. Thus, a story is a vehicle for representing reality, not through its facts per se, but through the cause and effect relationships suggested by the sequence of facts.

Games also attempt to represent reality. The difference between the two is that a story presents the facts in an immutable sequence, while a game presents a branching tree of sequences and allows the player to create his own story by making choices at each branch point. The audience of a story must infer causal relationships from a single sequence of facts; the player of a game is encouraged to explore alternatives, contrapositives, and inversions. The game player is free to explore the causal relationship from many different angles.

Indeed, the player expects to play the game many times, trying different strategies each time. A story is meant to be experienced once; its representational value decreases with subsequent retellings because it presents no new information. A game's representational value increases with each playing until the player has explored a representative subset of all of the branches in the game net.

This does not mean that games are better than stories. Although stories trace only a single sequence of causal development, they do so with greater intricacy and detail than games. Detail is crucial to the creative success of a story, for it provides the texture, the feel of reality that makes a story compelling. The story writer unleashes a mighty swirling torrent of facts that sweeps the audience to its predestined conclusion. The game designer creates a complex network of paths cunningly crafted to show the player all possible facets of a single truth. In this respect, a story is like a statuette where a game is like a jewel. The statuette's value arises from the fineness of detail and intricacy of construction. A jewel, by contrast, has no detail; its faces must be absolutely smooth. The jewel's value arises from its ability to refract light into many different angles. A statuette is meant to be stationary; a jewel is meant to be moved. So too, is a story static where a game is dynamic.

Stories enjoy a particular advantage over the current generation of computer games: the element of surprise. A good story boasts an array of interesting plot twists. The storyteller leads us into a set of expectations and then cleverly inserts a new factor that creates a disjunction, a new and dramatically different situation. This process can be repeated many times during the course of the story. Among computer games, only adventures provide this element of surprise. Unfortunately, the surprise can only be created by limiting the player's freedom of action so as to guarantee that the player will encounter the surprise under the proper circumstances. After a while, all adventures begin to smell like primrose paths. The really exciting possibility offered by computer games is the prospect of formulating a plot twist in response to the player's actions, instead of merely dragging him down a pre-ordained primrose path. However, the ability to formulate surprise requires an ability to analyze the player's actions, deduce his expectations, and generate a believable plot twist that confutes his expectations without frustrating him. Artificial intelligence that advanced has yet to be created.

Games versus Toys

Games lie between stories and toys on a scale of manipulability. Stories do not permit the audience any opportunity to control the sequence of facts presented. Games allow the player to manipulate some of the facts of the fantasy, but the rules governing the fantasy remain fixed. Toys are much looser; the toy user is free to manipulate it in any manner that strikes his fancy. The storyteller has direct creative control over his audience's experience; the game designer has indirect control; the toymaker has almost none.

Significance of Interaction

Interaction is important for several reasons. First, it injects a social or interpersonal element into the event. It transforms the challenge of the game from a technical one to an interpersonal one. Solving a cube puzzle is a strictly technical operation; playing chess is an interpersonal operation. In the former, one plays against the logic of the situation; in the latter, one uses the logic of the situation to play against the opponent.

Second, interaction transforms the nature of the challenge from a passive challenge to an active challenge. A puzzle will always present the player with exactly the same challenge. But a game opponent reacts to player's actions, and presents different challenges in each game. This difference has major emotional significance. The person solving the puzzle must somehow divine, guess, deduce, master, or discover the key trick built into the puzzle by the designer. Emotionally, the puzzle player is working against the puzzle or its designer to unmask its secret. Once the secret is known, the puzzle is no longer interesting. The game-player, by contrast, faces different challenges each time she plays the game. Where a puzzle is dead a game is alive; the player must create her solution to the game in a manner best suited to her own personality and that of her opponent. The key distinction between a game and a puzzle is the difference between creating your own solution and discovering the designer's solution. A game acknowledges the player's existence and reacts to the player's personality; a puzzle lies down like a dead fish.

Computer games seldom provide a human opponent, and so they lack the social element that other games offer. They can, however, present an illusory personality against which the player must work. This is one of the most exciting and least developed potentials of the computer as a game technology. And regardless of the computer's success or failure in synthesizing a social element, the computer can readily make the game a highly interactive experience for the player. It can react to the player's moves with speed and thoroughness.

Nature of Interaction

Interactiveness is not a binary quantity; it is a continuous quantity with a range of values. Puzzles have little or no interactiveness, while games have more interactiveness. This suggests that interactiveness is an index of "gaminess". Some games, such as blackjack, tag, or PONG provide very little interaction between the players. Although the players may wish to interact, the games provide very limited modes of interaction (binary decision to stand or hit, running, and twisting paddle). The games do not allow players to invest much of themselves into the play, or to react in a rich way to their opponents. Other games, such as bridge, football, and LEGIONNAIRE (trademark of Avalon Hill Game Co.) allow a far richer interaction between players. Players can grapple with each other at a variety of levels. The first group of games is generally acknowledged to be dull, while the second group of games is generally regarded as more interesting. What is important about the modes of interaction is not their mechanical quality but their emotional significance. PONG is insipid because I can't express much of my personality through the medium of a bouncing ball. Bridge is better because it includes within its interaction elements of teamwork, deception, and cooperation. I can better imprint my personality traits onto a game of bridge. Thus, degree of interaction provides a useful index of "gaminess".

CONFLICT

A third element appearing in all games is conflict. Conflict arises naturally from the interaction in a game. The player is actively pursuing some goal. Obstacles prevent him from easily achieving this goal. If the obstacles are passive or static, the challenge is a puzzle or athletic challenge. If they are active or dynamic, if they purposefully respond to the player, the challenge is a game. However, active, responsive, purposeful obstacles require an intelligent agent. If that intelligent agent actively blocks the player's attempts to reach his goals, conflict between the player and the agent is inevitable. Thus, conflict is fundamental to all games.

Games without conflict?

Some people shrink' from this aspect of games. A number of attempts have been made to design "nice" games cleansed of conflict. Such games emphasize cooperative efforts rather than conflict. They have not been successful commercially; this suggests that few people enjoy them.

More importantly, these games are failures because they are not games in the first place. Conflict can only be avoided by eliminating the active response to the player's actions. Without active response, there can be no interaction. Thus, expunging conflict from a game inevitably destroys the game.

While it is impossible to eliminate conflict from a game without destroying the game, it is possible to include cooperative elements by shifting the conflict. Members of a team can cooperate with each other in the team's conflict with another agent. This other agent could be another team, an individual human, or a computer simulated player. In all cases, the opponent must be perceivable as endowed with a persona. Without at least the illusion of purposeful reaction to the player's actions, the game collapses.

This "blood and guts" view of conflict in games is reinforced by the social context in which they are often played. Our real world conflicts are always indirect, diffused over time, and tightly regulated. Moreover, they all too frequently lack resolution, for seldom does one achieve an outright victory in the conflicts of daily life. Local successes, yes, but the struggle continues without clear resolution. Because games are subjective representations of the real world, they focus our attention on a particular aspect of the world by accentuating that aspect. Conflict in games thus tends to be (but need not always be) accentuated to its most direct and intense form violence. Violence is not essential or fundamental to games. It is common in games because it is the most obvious and natural expression for conflict.

Summary of Conflict

Conflict is an intrinsic element of all games. It can be direct or indirect, violent or nonviolent, but it is always present in every game.

SAFETY

Conflict implies danger; danger means risk of harm; harm is undesirable. Therefore, a game is an artifice for providing the psychological experiences of conflict and danger while excluding their physical realizations. In short, a game is a safe way to experience reality. More accurately, the results of a game are always less harsh than the situations the game models. A player can blast the monsters all day long and risk only her quarter. She can amass huge financial empires and lose them in an hour without risking her piggy bank. She can lead great armies into desperate battles on which hang the fate of nations, all without shedding a drop of blood. In a world of relentless cause and effect, of tragic linkages and inevitable consequences, the disassociation of actions from consequences is a compelling feature of games.

This is not to imply that games are devoid of consequences. The penalties for losing a game can sometimes be a significant deterrent to game play. Losing to another person always entails some loss of dignity. This may be an attraction of computer games there is less shame in losing to a computer. The loser can keep coming back for more defeats without losing face. Moreover, true victory the total destruction of the computer's forces, is acknowledged to be impossible in most such games; this further lessens the shame of defeat.

A second penalty for losing is the less of any reward that might have been gained by winning. In almost all games the reward penalty structure is positive. That is, the loser is not punished for losing, the winner is rewarded for winning. The loser's only loss is any investment that he made to enter the game, such as a bet or entry fee. This investment is usually very small, and may rightly be regarded as a recreational fee for the services associated with the administration of the game rather than a penalty for all potential losers.

Gambling presents us with some difficult problems related to the issue of the safety of games. Gamblers risk money or goods on the outcome of a random or near random process. Losers forfeit their bets and winners reap a large reward. Hence, gambling presents a real financial risk to the player. However, two extenuating circumstances intervene: first, the recreational gambler risks very little money; second, some gamblers deny to themselves the laws of chance. They indulge in the fantasy of control. The proper intonation in the shake of the dice, the correct twist on the handle of the slot machine these things make the difference, or so they tell themselves. Thus, recreational gambling, while somewhat deviant from the mainline of game playing, probably deserves inclusion in the fold of games. Serious gambling, however, involving large sums of money expended more for anticipated financial gain than for recreation, lies on the far side of the gray zone.

A special form of gambling, deserving special consideration here, is poker. Poker is a game of bluffing; the key to success in the game lies in convincing your opponent that you have better or worse cards than you really have. Because money is at stake, the player experiences stresses that strain his ability to deceive his opponents. Thus, the risk of gambling, a mere outcome of other games, is an intrinsic part of the structure of poker. This unique aspect of poker merits special consideration. I would not hesitate to classify poker as a game.

Summary of Safety

Games provide safe ways to experience reality. Special cases abound, but the central principle remains: games are safe. In this chapter I have presented a set of characteristics that defines what I mean by the word “game”. For the most part, I have emphasized the characteristics intrinsic to the games themselves rather than the motivations of the players. Such separation of game from player is artificial and misleading, for neither exists without the other. In the next chapter, I turn to look at the players of games and their motivations.

Chapter Two
Why Do People Play Games ?

Game-playing requires two components: a game and a player. The game designer works to produce a game, and so her immediate preoccupation is with the game itself. Yet, her final goal is to educate, entertain, or edify the game-player; hence, the human player is the proper primary concern of the game designer. Why do people play games? What motivates them? What makes games fun? The answers to these questions are crucial to good game design.

One way to address the question of the purpose of games is to inquire into their history. Games now are too varied, too intricate, too involved, to indicate a single clear function. Perhaps their fundamental nature would be more evident in their earliest incarnations. How far back must we go? To MONOPOLY, created during the Depression? No, card games were played long before that. Indeed, the discoverers of King Tutankhamen's tomb found among the wealth there a wooden surface with regular divisions that appears to be some sort of boardgame. But even archaeology does not take us far enough back. If we wish to get back to the beginnings of games, we must go beyond the realm of the archaeologist and into the realm of the paleontologist. We must reach not thousands but millions of years into the past to find the earliest games, for games predate not just history but all of mankind. They are not a human invention.

Fortunately, direct recourse to paleontology is unnecessary. A trip to the zoo will suffice. There we find two lion cubs wrestling near their mother. They growl and claw at each other. They bite and kick. One cub wanders off and notices a butterfly. It crouches in the grass, creeps ever so slowly toward its insect prey, then raises its haunches, wiggles them, and pounces. We laugh at the comedy; we say that the cubs are playing a game, that they are having fun, and that they are such fun-loving, carefree creatures.

We are right on the first count: these cubs do indeed appear to be playing a kind of game. We can certainly see in their behavior all four of the fundamental game attributes described in Chapter 1: representation, interaction, conflict, and safety. We may be right on the second count; who knows if lions can have fun? But we are dead wrong on the last count. These cubs are not carefree. They do not indulge in games to while away the years of their cubhood. These games are deadly serious business. They are studying the skills of hunting, the skills of survival. They are learning how to approach their prey without being seen, how to pounce, and how to grapple with and dispatch prey without being injured. They are learning by doing, but in a safe way. Better to make mistakes with butterfly and sibling than with the horns of the wildebeest.

Games are thus the most ancient and time-honored vehicle for education. They are the original educational technology, the natural one, having received the seal of approval of natural selection. We don't see mother lions lecturing cubs at the chalkboard; we don't see senior lions writing their memoirs for posterity. In light of this, the question, "Can games have educational value?" becomes absurd. It is not games but schools that are the newfangled notion, the untested fad, the violator of tradition. Game-playing is a vital educational function for any creature capable of learning.

The incidence of game-playing in animals is itself instructive. Game-playing has been observed only in mammals and birds. The phylogenetically earlier orders (fish, insects, amphibians, and reptiles) have not been shown to engage in game-playing. (See Animal Play Behavior, by Robert Fagen, Oxford University Press.) Game play seems to be associated with that quality which we have clumsily attempted to measure with brain size, intelligence, and ability to learn. This correspondence cannot be attributed to accident; clearly game play is an important component in the development of many creatures.

We commonly associate the playing of games with children. Indeed, "play" as an activity is considered to be the almost exclusive preserve of children, and the term is applied to adults either disparagingly or jocularly. Children are expected to play games because we recognize (perhaps unconsciously) the fundamental utility of games as an educational tool. As children grow up, cultural pressures change and they are encouraged to devote less time to the playing of games so that they can devote themselves to more serious activities.

I claim that the fundamental motivation for all game-playing is to learn. This is the original motivation for game-playing, and surely retains much of its importance. This claim does not conflict with my other primary assertion that computer games constitute a new art form. Consider, for example, humans and food. The fundamental motivation to eat food is the base desire for nourishment, yet this has not prevented us from embellishing this fundamental activity with all manner of elaborate and non-nourishing customs, rituals, seasonings, and garnishes. I do not mean to imply that food is an art form; only that we humans can take an activity far beyond its prime cause without denying that prime cause.

I must qualify my claim that the fundamental motivation for all game-play is to learn. First, the educational motivation may not be conscious. Indeed, it may well take the form of a vague predilection to play games. The fact that this motivation may be unconscious does not lessen its import; indeed, the fact would lend credence to the assertion that learning is a truly fundamental motivation.

Second, there are many other motivations to play games that have little to do with learning, and in some cases these secondary motivations may assume greater local importance than the ancestral motivation to learn. These other motivations include: fantasy/exploration, nose-thumbing, proving oneself, social lubrication, exercise, and need for acknowledgment. I shall examine each in turn.

Fantasy/Exploration

A very important motivation to play games is fantasy fulfillment. Like a movie, a book, or music, a game can transport the player away from the tawdry world that oppresses him and create a fantasy world in which he can forget his problems. Games are potentially superior to the traditional means of escape (movies, books, music) because they are participatory. Instead of merely watching a movie reading a book, or listening to music, the player is actively involved in the game. Indeed, the player drives the game, controls it in a way that is quite impossible with the passive fantasies. This need to escape, to fantasize is certainly an important motivation.

Fantasy fulfillment frequently takes the form of symbolic exploration. There's a big world out there, full of exciting things, people, and places, yet most of us are confined to a world ,of asphalt, plastic, and paper. Many art forms attempt to transport the audience into a different world, to present experiences or feelings not often known in the everyday world.

Consider, for example, the success of Disneyland. This place is undoubtedly the most successful of its genre. Such parks are often called "amusement parks" or "theme parks." These terms are misleading, for the success of Disneyland cannot be attributed solely to its amusements and diversions. These elements are technically excellent, but other amusement parks sport technically excellent rides. The success of Disneyland can be summed up in one word: fantasy. Disneyland creates and supports an aura of fantasy, a context of make-believe that permeates all of the activities within the park. Within moments of entering the park, the visitor feels that s/he is in a different world. Fanatic attention to detail in signposts, walls, windows, even railings has created an environment that encourages suspension of disbelief.

Fantasy is an important component of human play. It is critical to our recreation, our art and our games.

Nose-Thumbing

A common function of games is to provide a means of overcoming social restrictions, at least in fantasy. Many games place the player in a role that would not be socially acceptable in real life, such as a pirate or a thief. An excellent (albeit extreme) example of this is the game CRUSH, CRUMBLE, AND CHOMP by Automated Simulations. In this game the player is cast as a 1950's-vintage monster going on a rampage through his favorite city. He stomps on police cars, crushes buildings, swats helicopters, and creates general mayhem. The box art shows a monster about to attack an IRS building as terrified citizens flee. This represents an extreme case of anti-social behavior made acceptable by the safety of the game.

Sometimes the player's role is itself socially acceptable, but the actions taken are discouraged in real life. MONOPOLY encourages players to engage in what the Federal Trade Commission delicately calls "predatory trade practices." Wargames encourage players to start and win wars. Some games address sexual matters, allowing players to indulge in make-believe behavior that they could never exhibit in the real world.

The most telling example of this nose-thumbing phenomenon lies in the arcade games. These games emphasize violence, and lots of it. The theme is almost universal in arcades: destroy somebody. The coup de grace is not delivered discreetly or elegantly. On the contrary, the victim is dispatched with the most colorful animated explosion possible. Like a Sam Peckinpah movie, the violence is the whole point and purpose of the enterprise. Yet, even as we pander to these distasteful emotions, we delicately mask them in less offensive garb. We never, never obliterate human beings; instead, we vaporize ugly space monsters. The monsters have perpetrated some odious interstellar crime, so the player is cast as the defender, the protector, or the avenger. The case is often presented that the game represents a time of extreme crisis ("THE FATE OF HUMANITY IS AT STAKE!!!"). This heightens the player's sense of urgency; it also conveniently justifies the use of extreme violence, thereby allowing the player to have violence without guilt. The player can thumb his nose at social strictures and engage in violence and mass murder without risking censure. The game provides a safe way to thumb one's nose.

Proving Oneself

Another function of games is as a means of demonstrating prowess. All games support this motivation to a greater or lesser degree. Many game-playing communities sponsor tournaments or player ratings. Arcade games support this function by recording and displaying the initials of the top-scoring players. There are also players who carry this to extremes. Their prime goal is not merely to win, but to beat somebody, preferably somebody worth beating. Chess has an unusually high concentration of such sharks; so do wargames. A common question asked during a wargame is "Are you playing for blood or for fun?" Such players normally prefer games that allow their skill to be properly brought to bear, so they tend towards games in which chance plays a minimal role.

Despite this concentration of such players in deductive logic games, almost all games have sharks preying on the playful players. When a shark plays for serious rewards (e.g., social dominance) and -takes serious risks of failure, the crucial element of safety is eliminated from the game, and the game ceases to be a game; it becomes a conflict.

Inasmuch as all games have the potential for being played in an overly competitive way, some people who are especially sensitive to the social risks of game-as-conflict refuse to play games, for they do not perceive the games to be safe. If they do play, they prefer to play games of pure chance, not so much to disable or discourage the shark as to create a situation in which winning is patently unrelated to prowess. If winning is arbitrary, social risk is eliminated and safety is restored.

It is impossible to design a game that is unalterably safe (i.e., invulnerable to sharks) without resorting to pure chance as the sole determinant of victory. If the game in any way allows individual prowess to affect the outcome, then the outcome is perceivable as a reflection of individual prowess. In most games, safety from social risk is conferred onto the game by the attitudes of the players, the willingness to say, "It's only a game."

Social Lubrication

Games are frequently used (especially by adults) as social lubricants. The game itself is of minor importance to the players; its real significance is its function as a focus around which an evening of socializing will be built. Card games and some light board games serve this function. An excellent example of such a social lubricant game is a game utilizing a large plastic gameboard about four feet square that is marked with colored spots. On each player's turn, a random process is used to determine which of four appendages (arms or legs) is to be placed on which spot on the board. As the players contort to fulfill the game requirements, they inevitably make physical contact with each other in innocent and foolishly humorous ways. Social interaction is thereby fostered.

Exercise

Exercise is another common motivation to play games. The exercise can be mental or physical or some combination of both; in either event, the game is an entertaining way to stay in shape. Some players like to exercise their cognitive skills, while others prefer the use of intuition. Some players prefer to exercise their athletic skills. Furthermore, players need to exercise their skills at an appropriate level. A chess player will get very little exercise out of a game of tic-tac-toe. Similarly, a person who finds tic-tac-toe challenging will get little useful exercise out of chess. These preferences sort players out and route them to the different games available.

Need for Acknowledgment

We all need to be acknowledged, to be recognized by other people. The acknowledgment we crave is not merely an acknowledgment of our existence, but of our personalities. For example, when we meet a casual acquaintance, we usually get a perfunctory acknowledgment ("Hello there, Jones.") We are more gratified when the greeting in some way acknowledges us as individuals with special personalities and problems ("Hello there, Jones; is that knee still bothering you?")

The popularity of pets provide another example of the need for acknowledgment. Why on earth do we keep in our homes animals that require food, veterinary attention, and sanitary maintenance? Because they acknowledge us. We can interact with pets; we talk to them, play with them, and emote with them. A dog is an especially responsive creature; it can read our facial expressions and interpret our tone of voice. A smile will trigger tall-wagging; a kind word will precipitate jumping, licking, barking, or some other expression of affection. Goldfish, by contrast, neither appreciate nor express emotion. Thus, even though goldfish are much easier to care for, most people prefer dogs as pets. People value acknowledgment enough to expend the effort to obtain it.

This is one reason why interaction is so important to a game; it allows the two players to acknowledge each other. A truly excellent game allows us to imprint a greater portion of our personalities into our game-playing. Such a game allows me to play in a way that only I could have played it.

My opponent must look beyond the playing pieces and acknowledge my cleverness, my rashness, my deviousness, my entire personality. When such a game ends, my opponent and I know each other better than we did before we sat down to play.

Summary

Many factors play a role in motivating a person to play a game. The original (and almost instinctive) motivation is to learn, but other motivations come to bear as well.

MOTIVATION VERSUS SELECTION

We must be careful to distinguish between factors that motivate people to play games in the first place and factors that allow people to choose between games. In other words, the answer to the question, "Why do people play games?" can be quite different from the answer to the question, "What makes one game more fun than another?" Some factors motivate a person to play games; other factors help that person select a particular game. For example, sensory gratification is such a selection factor. A player who has decided to play a particular type of game will prefer a game with excellent graphics over games with poor graphics; yet the graphics alone will not motivate many people to play games. Motivating factors get people to approach games in general; enjoyment factors help them make their choice of particular games.

Distinguishing motivation from enjoyment is not tantamount to denying correlation's between motivating factors and enjoyment factors. Clearly, any game that does not deliver the experiences implied by the motivating factor will not be enjoyed. Thus, some (but not all) motivating factors will also be used as enjoyment factors. If a player is motivated to play a game for mental exercise, that player will probably prefer those games that offer better mental exercise than do other games. A game cannot be fun if its factors do not satisfy the motivations of the player. Two enjoyment factors that are not in themselves motivational are game play and sensory gratification.

Game Play

Game play is a crucial element in any skill-and-action game. This term has been used for some years, but no clear consensus has arisen as to its meaning. Everyone agrees that good game play is essential to the success of a game, and that game play has something to do with the quality of the player's interaction with the game. Beyond that, nuances of meaning are as numerous as users of the phrase. The term is losing descriptive value because of its ambiguity. I therefore present here a more precise, more limited, and (I hope) more useful meaning for the term "game play". I suggest that this elusive trait is derived from the combination of pace and cognitive effort required by the game. Games like TEMPEST have a demonic pace, while games like BATTLEZ0NE have a far more deliberate pace. Despite this difference, both games have good game play, for the pace is appropriate to the cognitive demands of the game. TEMPEST requires far less planning and conceptualization than BATTLEZONE; the demands on the player are simple and direct, albeit at a fast pace. BATTLEZONE requires considerably greater cognitive effort from the player, but at a slower pace. Thus, both games have roughly equivalent game play even though they have very different paces. Pace and cognitive effort combine to yield game play.

Sensory Gratification

Sensory gratification is another important enjoyment factor. Good graphics, color, animation, and sound are all valued by game players. They support the fantasy of the game by providing sensory "proof" of the game's reality. We see a related phenomenon in the movies: special effects. Some of the newer movies have excited great interest because of the excellent special effects they utilize. These movies have placed us in the thick of space battles, let us meet strange and wonderful creatures, and taken us to faraway places. The things we see look so real that we believe the fantasy; we know (subjectively) that the fantasy is real. Similar processes can be applied to games. Special effects, graphics, sound, animation-these factors all help distinguish a good game from a bad game. We must not confuse their role, however; sensory gratification is a crucial support function, not a central feature. Sensory texture enhances the impact of the fantasy created by the game or movie, but wonderful graphics or sound do not by themselves make the product. A movie without a believable or enjoyable fantasy is just a collection of pretty pictures; a game without an entertaining fantasy is just a collection of interactive pretty pictures.

INDIVIDUAL TASTES

So far I have discussed motivational and enjoyment factors as if they were absolute quantities whose significance is independent of the individual player. Such is not the case; the response to a given game depends heavily on the personality of the prospective player. How are we to deal with the personality differences that dominate the individual's response to games?

One academic solution to this problem is to postulate the existence of a very large number of personality traits that determine the individual response to a game. We next postulate a like number of game traits that, taken together, completely define the psychological profile of the game. Next, we measure and catalog all of the personality traits of any given individual, presumably with an omniscient "personalitometer". Then we measure all the game traits of the game in question with an equally powerful "gamometer". We then perform a matrix multiplication of personality traits against game traits. Sometime before the sun enters its red giant phase, our monster computer returns a number telling us how much that person will enjoy that game.

This approach will for the moment remain a gedanken-experiment. We must devise simpler, admittedly less reliable means of coping with individual differences. One alternative route is to observe and catalog groups of game-players, and identify the game traits valued by these groups. This method is made difficult by the youth of the computer game industry. We can at this time identify only a few broad, vague, and overlapping groups of players: skill-and-action enthusiasts, D&D enthusiasts, and strategy gamers. There remain several other game types, but they have not attracted so large a following as to present us with a definable group of players. The passage of time and further research will certainly give us more information with which to work.

Individual tastes in games are not static; as a person changes so do the tastes. The following analogy with music illustrates this point.

As children, we are all exposed to music in a variety of forms, but it has little impact on us because our tastes are poorly developed. We sing and dance to simple songs, but a full appreciation of the emotional range of music eludes us. The power of music arises from our ability to associate musical expressions with emotions. It takes years to develop these associations, and they are made in the context of our experiences. For many in my generation, the first deep contact with music came with rock 'n roll in the 60's. The pounding beat, simple themes, and short durations were easily grasped by our adolescent and unsophisticated minds. We could understand this music. Moreover, the act of listening to and enjoying this music was itself an educational experience. As the range of our musical experience expanded, we learned more complex components of the musical lexicon and developed a wider range of associations. Soon we were able to understand and appreciate other musical compositions previously inaccessible to our untrained ears. Rock music changed to reflect this maturation; some of us stayed with rock. Others moved to jazz, country, or folk. Like some others, I moved from rock to classical in a series of stages. As I moved along this evolutionary path, the lessons of one stage enabled me to understand the material of the next stage. Other people followed their own paths, exploring and learning the areas of musical expression that most appealed to them. The common experience was that our musical tastes evolved, no matter what direction we chose. Rock music was the broad base we all shared, the entry point or ,junk out of which sprang many branches.

Just as rock 'n roll was the entry point into the world of music for an entire generation, so will skill-and-action games be the entry point into the world of games for the whole population. Like early rock 'n roll, skill-and-action games have broad appeal, and are easy to understand. As people become more sophisticated with games, their tastes will evolve down different branches. Like rock 'n roll, skill-and-action games will not go away; they will change to reflect the evolving taste of the public. We can see this happening already. The early arcade games are tame pussycats compared to the rip-snorting, fire-breathing games of 1982. Had TEMPEST been released in 1977, it would have intimidated and repelled players. Times change; people change. Skill-and-action is here to stay and will always provide an entry point for new players, but the public will not stand still. Many people will move on to explore other areas of game-playing.

People play games for many reasons. In this chapter, I have touched on a variety of these motivations. I readily admit that my treatment of the subject matter is thin, speculative, and uncompelling. People are complex creatures; we will never fully understand human motivations to play games. Yet me must appreciate the importance of these motivations and at least try to understand them if we are to master the art of computer game design.

CHAPTER THREE
A Taxonomy of Computer Games

Thousands of computer games are commercially available on a variety of hardware configurations. These games present a bewildering array of properties. Many show close similarities. Most possess some unique design feature. Given this large sample of games, we can learn a great deal about game design by establishing a taxonomy of computer games. A taxonomy would illuminate the common factors that link families of games, while revealing critical differences between families and between members of families. A well-constructed taxonomy will often suggest previously unexplored areas of game design. Most important, a taxonomy reveals underlying principles of game design. In another field of study, Charles Darwin's meticulous taxonometric work while on the Beagle led almost inevitably to his development of the theory of evolution. While we cannot hope that taxonometric work in computer game studies will be so spectacularly productive, it certainly seems worth the effort.

I will insist on an important qualification: I do not claim that the taxonomy I propose is the correct one, nor will I accept the claim that any correct taxonomy can be formulated. A taxonomy is only a way of organizing a large number of related objects. If there were some organizing agent, some underlying process that created the group of objects, then we could reasonably expect to be able to find a single correct taxonomy embodying the central organizing principle in its structure. For example, the wide array of living creatures on this earth did not arise by chance; this array is the product of natural selection. Natural selection is a reasonable, understandable, nonarbitrary process. Therefore, there is only one reasonable taxonomy for life on earth, the taxonomy that embodies the principles of natural selection. In the shape of an airplane we can see the principles of aerodynamics; so too in a taxonomy of living creatures can we see the hand of natural selection.

Such is not the case with computer games. The field is too young, the sample too small, for whatever organizing principles there may be to have asserted themselves. The games we now have are more the product of happenstance than the inevitable result of well-established forces. Without a wide array of games there is little opportunity to choose between games; without choice there can be no natural selection. It is therefore impossible for us to devise a single, absolute taxonomy. Many taxonomies are admissible. Indeed, attempting to construct several alternative taxonomies is a useful way to examine the common traits of computer games. I am not so ambitious; I shall be happy to propose just one taxonomy. I divide computer games into two broad categories: skill-and-action ("S&A") games (emphasizing perceptual and motor skills) and strategy games (emphasizing cognitive effort). Each major category has several subcategories.

SKILL-AND-ACTION GAMES

This is easily the largest and most popular class of computer games. Indeed, most people associate all computer games with skill-and-action games. All arcade games are S&A games and almost all games for the ATARI 2600 are S&A games. This class of games is characterized by real-time play, heavy emphasis on graphics and sound, and use of joysticks or paddles rather than a keyboard. The primary skills demanded of the player are hand-eye coordination and fast reaction time.

I group skill-and-action games into six categories: combat games, maze games, sports games, paddle games, race games, and miscellaneous games.

Combat Games

Combat games all present a direct, violent confrontation. The human player must shoot and destroy the bad guys controlled by the computer. The challenge is to position oneself properly to avoid being hit by the enemy while shooting him. These games are immensely popular; they are Atari's forte. There are many variations on this theme, most arising from variations on the geometry of the situation or the weaponry of the opponents.

STAR RAIDERS and SPACEWAR can be compared on these bases of geometry and weaponry. In both games the player files through space in a rocket ship and engages enemy spaceships in real-time cosmic dogfights. STAR RAIDERS presents the conflict in first-person geometry (that is, the television screen shows the same scene that the pilot would see.) SPACEWAR uses much the same weaponry and mechanisms with one crucial difference: the geometry of the game is third-person rather than first-person (that is, the player sees his own and his opponent's spaceships from a distance.) The difference in result is obvious to anyone who has played both games. The first-person game is more exciting and compelling than the third-person game. Unfortunately, the first-person geometry is so technically difficult to execute that it has been implemented on only a few games. Most games use third-person geometry.

ASTEROIDS is a shoot-em-up game that uses the same space environ that STAR RAIDERS uses. The primary difference between the two games is in the nature of the opposition. The enemy in ASTEROIDS is not a small number of intelligent opponents armed with weapons identical to the player's; instead, the enemy is a large number of stupid rocks armed only with their ability to destructively collide with the player.

MISSILE COMMAND is another combat game with several interesting twists. First, the player must defend not only himself but also his cities from descending nuclear bombs. Second, the game is a purely defensive game in that the player never has the opportunity to attack his enemy. Third, while shots in other games are very rapid events, the shooting process in this game is slower and takes time to develop because the missiles must fly to their targets before detonating. Because the time between firing and impact is so long, the player must plan his shots with greater foresight and make use of multiple explosions. Thus, although this is a skill-and-action game, there are more strategic elements involved than in many games of this category.

SPACE INVADERS (trademark of Taito America Corp.) is one of the most successful combat games of all time. It was one of the first smash hit games and contributed to the upsurge of popularity of computer games that began in 1979. While STAR RAIDERS and ASTEROIDS give the player great mobility and MISSILE COMMAND gives him none, SPACE INVADERS gives the player limited mobility in one dimension only. As in ASTEROIDS, the player must face a multitude of rather stupid opponents who can win by touching the player (landing); in addition, as in STAR RAIDERS, the monsters also shoot back. The monsters march back and forth across the screen, slowly descending onto the player. As the player kills more and more monsters, they march faster and faster. This gives the game a hypnotic accelerating tempo. SPACE INVADERS is definitely a classic.

The success of SPACE INVADERS has spawned a whole series of copies and derivatives. There are a large number of copies whose only goal was to cash in on the success of the original game. There are also several genuine derivative games. For example, GALAXIAN (trademark of Midway) is a simple variation on SPACE INVADERS. Individual invaders peel off and attack the player with more ferocity than the docile monsters of the original game. CENTIPEDE; is also a derivative of SPACE INVADERS; it is different enough to be a new design, but the internal game structure is very similar to the original. The invaders have been grouped into a segmented centipede; their side-to-side motion is bounded not by the edges of the screen but by mushrooms randomly scattered across the screen. Numerous embellishments (spiders, fleas, and scorpions) extend the game considerably. TEMPEST is a three-dimensional first-person derivative of SPACE INVADERS using vector graphics. The amount of design attention that SPACE INVADERS has attracted is a tribute to the game's originality, appeal, and durability

There are many, many other combat games. BATTLEZONE and RED BARON are two first-person combat games utilizing vector displays. Other combat games include CAVERNS OF MARS, YAR'S REVENGE, CROSSFIRE (trademark of On-Line Systems) and DEFENDER (trademark of Williams).

You may wonder why so many combat games are set in outer space. There are three reasons. First, space is easy to depict and animate with a computer---all the designer need do is draw a blank screen with a few white dots for stars. Second, space is not encumbered by the expectations of the players. A designer encountering problems can always concoct some super-duper zapper to solve any design problems with the game and nobody can object that it is unrealistic. Earthbound games constrain the designer to look reality squarely in the eye---such a tiresome burden for a "creative" mind. Third, space is an intrinsically fantasy-laden environment that encourages suspension of disbelief because it is unfamiliar to its audience.

Combat games have always been at the heart of computer gaming. Players never seem to tire of them; it appears that they will be around for a long time to come.

Maze Games

The second subgrouping of S&A games is the set of maze games. PAC-MAN (trademark of Namco) is the most successful of these, although maze games predate PAC-MAN. The defining characteristic of the maze games is the maze of paths through which the player must move. Sometimes one or more bad guys pursue the player through the maze. Some maze games (MAZE CRAZE for the ATARI 2600 is a good example) specifically require that the player make his way to an exit. Other maze games require that the player move through each part of the maze. DODGE 'EM is an early example of such a game. In either case, the number, speed, and intelligence of the pursuers then determines the pace and difficulty of the game. PAC-MAN has a very carefully balanced combination of these factors. The pursuers are just slightly slower than the human player; their intelligence and number make up for this. The overall pace of the game makes it difficult for the player to fully analyze the positions of the five pieces in real time.

Any successful game is certain to attract copies, variations, and derivatives, and PAC-MAN is no exception. One of the first such games for the ATARI Home Computer System was the first edition of JAWBREAKERS (trademark of On-Line Systems). This game, now removed from the market, clearly demonstrates the difference between structural changes and cosmetic changes. Structurally, it is indistinguishable from PAC-MAN. The play of the game is almost identical to that of PAC-MAN. Cosmetically, there are a number of differences: the pursuers are faces rather than ghosts; the player is a set of teeth rather than a head with mouth; the maze is laid out differently; the sounds are different. This game provides a good example of the methods that can be used to copy games while attempting to minimize legal problems.

Another PAC-MAN derivative is MOUSKATTACK (trademark of On-Line Systems). This game shows some structural changes relative to PAC-MAN. The player is again pursued through a maze by four computer-controlled creatures, but the basic scenario contains a number of embellishments. First, merely passing through every point in the maze is not enough; some points, randomly chosen by the computer, must be passed through twice. Second, the player is allowed to fight back against the pursuers in a very different way (setting mousetraps). The strategic and tactical effects of this counterforce capability yield a game that plays rather differently. Finally, there is a very interesting two-player game that allows both cooperative and competitive strategies. In MOUSKATTACK we see the basic structure of PAC-MAN with a number of embellishments and extensions that produce a distinct game.

The appeal of maze games can be attributed to the cleanliness with which they encapsulate the branching structure that is a fundamental aspect of all games. The reader will remember from Chapter One that a game has a tree structure with each branch point representing a decision made by the player. In a maze game, each branch point is neatly depicted by an intersection in the maze, and the options available to the player are visually presented as the paths available at the intersection. Thus, a maze game presents a clear visual representation of the branching structure of the game.

Even more fascinating is the looping structure possible with maze games. A player can return to an intersection in the maze many times. Yet, each time he does so, the options he has take different meanings because the other maze-inhabitants have moved in the interim to a different pattern of positions. In this way, a small number of displayed intersections can represent a huge number of branch-points in the game-tree. The analogy with a computer program, in which a small number of program instructions, through looping and branching, can address a large number of specific cases, is striking.

Sports Games

These games model popular sports games. They are anachronisms derived from the early days of computer game design when computer games had no identity of their own. People without original ideas for games fell back on the sports games as models around which to design. This also served a useful marketing purpose: why would a conservative consumer buy a game with a title and subject completely alien to his experience? Better to offer him a game he is already familiar with. Thus we have games based on basketball, football, baseball, soccer, tennis, boxing, and others. All of these games take liberties with their subject matter to achieve playability. The most enjoyable aspects of the computer game have very little to do with the real game. This is fortunate, for a slavish attempt at replication would have produced a poor computer game. Only by substantially altering the original games were the authors able to produce a decent design. Even so, sports games remain the wallflowers of computer gaming. I suspect that sports games will not attract a great deal of design attention in the future. Now that computer games have an accepted identity of their own, the need for recognizable game titles has diminished.

Paddle Games

I use the title "Paddle Games" to cover the PONG-based games. PONG is certainly one of the most successful and fertile of game designs, for it has many grandchildren and great-grandchildren. The central element of the game, that of intercepting a projectile with a paddle-controlled piece, has been used in endless variations. The original PONG pitted two players in an electronic version of ping-pong, hence the name. BREAKOUT was a solitaire version that required the player to chip away at a wall with the ball. The player received points for each brick destroyed. SUPER-BREAKOUT introduced variations on this theme with moving walls, extra balls, and other tricks. CIRCUS ATARI introduced parabolic trajectories for the projectiles and a complex moving wall of balloons. WARLORDS; took the genre even further; up to four players (one in each corner) defend brick castles against a projectile bounced around the field by their shield-paddles.

In the above games, the player uses the ball as a weapon to batter; in other paddle games the player must only catch the ball, or many balls, rather than deflect it. AVALANCHE is one such game. In this game, the player is at the bottom of the screen and large numbers of rocks are failing; each one must be caught with the player’s piece. The game becomes quite frantic as more and more rocks fall at a faster and faster pace. Another game, CHICKEN, (trademark of Synapse Software) expands on this theme by replacing the rocks with eggs and making each one hatch on striking the ground, forcing the player-hen to jump over it as she moves about.

The paddle game-system is a very simple one; although I doubt that it has much development potential remaining, I am hesitant to pronounce such a durable old system dead.

Race Games

Some computer games involve a straightforward race. Most of these games allow the player to move at constant speed, but extract time penalties for failure to skillfully negotiate an assortment of hazards. Thus, a player in the APX skiing game DOWNHILL must avoid the trees and rocks; the player's score is based on his time to complete the course. MATCH RACER by Gebelli Software is a car-racing game with oil slicks and obstacles. NIGHT DRIVER is a car-racing game featuring a first-person view of the road. One problem with all of these games is that they are not true games but puzzles, for there is no real interaction in a race between a player and his opponent. Indeed, it is difficult to identify the opponent in these games.

A more involved variation on the race game is DOG DAZE by Grey Chang. This is a true game, not a puzzle. It presents a two-player competitive race game with variable goals and asymmetric obstacles. Each player has a dog; hydrants pop onto the screen at random locations; the players must race to be the first to touch the hydrant, thereby claiming it as their own. Players may not touch hydrants owned by their opponents on pain of being temporarily paralyzed. The game has many interesting twists and turns without being overly complex; it demonstrates that the race game can be a flexible vehicle of game design.

Miscellaneous Games

My taxonomy is flawed; there exist a number of games that do not fit into this taxonomy very well. The first I will mention is DONKEY KONG, (trademark of Nintendo) a game that looks vaguely like a race game with intelligent obstacles. FROGGER (trademark of ________) is another game that defies classification in this taxonomy. It could perhaps be called a maze game with moving walls or obstacles, but the fit is poor. APPLE PANIC by Broderbund Software also defies my taxonomy. In some ways it is like a maze game and in some ways it is a combat game. The pace of the game is oddly slow. I don't know what to call this game. The fact that these games do not fit my taxonomy does not bother me overly much; I certainly don't want to create ad hoc categories for individual games. I am content to wait and see other developments before I create new categories or revise old ones.

STRATEGY GAMES

Strategy games comprise the second broad class of computer games. These games emphasize cogitation rather than manipulation. I do not mean to imply that S&A games are devoid of strategic content; some S&A games do indeed have a strategic element. The major distinguishing factor between strategy games and S&A games is the emphasis on motor skills. All skill-and-action games require some motor skills; strategy games do not. Indeed, real-time play is rare in strategy games (this is changing; LEGIONNAIRE from Avalon-HIII is a notable real-time strategy game). Strategy games typically require more time to play than S&A games. Strategy games are nonexistent in the arcades; they are rare on the ATARI 2600; they are almost exclusively restricted to personal computers. I divide strategy games into six categories: Adventures, D&D games, wargames, games of chance, educational games, and interpersonal games

Adventures

These games derive from one of the oldest computer games, called "Adventure". In these games the adventurer must move through a complex world, accumulating tools and booty adequate for overcoming each obstacle, until finally the adventurer reaches the treasure or goal. Scott Adams created the first set of Adventures widely available for personal computers; his software house (Adventure International) is built on those games. The Scott Adams games are pure text adventures that run in a small amount of memory, so they do not need disk drives; they are also readily transportable to different machines. A short time later Ken and Roberta Williams built On-Line Systems with THE WIZARD AND THE PRINCESS (trademark of On-Line Systems), an adventure that presented pictures of the scenes in which the adventurer found himself. The game itself was not particularly new; the innovation was primarily the use of graphics. Both firms have expanded their lines with more games using the systems they pioneered. Most of these derivative games are structurally similar to the originals, differing in detail, polish, and size.

The next variation on the adventure theme was the giant adventure, of which there are several. TIME ZONE by On-Line Systems is one of these. These giant adventures use multiple diskettes to link together a gigantic adventure. As the player solves the puzzle in one environment he moves on to another environment on another disk. The games are structurally identical to earlier games; the only difference is one of magnitude. They take many weeks of play to solve.

A new variation on the adventure game genre is DEADLINE (trademark of Infocom), a detective adventure with a number of interesting twists. Its heritage as an adventure is evident in its lack of graphics and its use of an excellent sentence parser. This adventure puts the player in the role of a detective attempting to solve a murder. The game is played in a real-time mode that adds to the interest and challenge of the game. The player searches not for treasure but for information with which to solve the murder. This game shows the potential of the adventure system in that the same system can be used, with the storyline and goals altered, to appeal to a different audience.

One of the most clever adventures ever done is Warren Robinett's ADVENTURE on the ATARI 2600. This adventure follows the same basic format as all adventures, except that it uses absolutely no text. Instead, the user moves through a series of rooms presented in rather simple graphics. Although the graphics and input schemes are radically different, the basic feel of the adventure system has been successfully retained. SUPERMAN, HAUNTED HOUSE, and GALAHAD AND THE HOLY GRAIL by Doug Crockford are all derivatives of this game.

Adventures are closer to puzzles than to games. As discussed in Chapter One, puzzles are distinguished from games by the static nature of the obstacles they present to the player. Adventures present intricate obstacles that, once cracked, no longer provide challenge to the player. It is true that some adventures push closer to being games by incorporating obstacles such as hungry dragons that in some way react to the player. Nevertheless, they remain primarily puzzles.

D&D Games

A completely independent thread of development comes from the D&D style games. Fantasy role-playing was created by Gary Gygax with Dungeons and Dragons (trademark of TSR Hobbles), a complex noncomputer game of exploration, cooperation, and conflict set in a fairytale world of castles, dragons, sorcerers, and dwarves. in D&D, a group of players under the guidance of a "dungeonmaster" sets out to gather treasure. The game is played with a minimum of hardware; players gather around a table and use little more than a pad of paper. The dungeonmaster applies the rules of the game structure and referees the game. The dungeonmaster has authority to adjudicate all events; this allows very complex systems to be created without the frustrations of complex rules. The atmosphere is quite loose and informal. For these reasons, D&D has become a popular game, with endless variations and derivatives.

D&D first appeared in the mid-70's; it didn't take long for people to realize that it had two serious limitations. First, the game needed a group of players and a dungeonmaster, so it was impossible to play the game solitaire. Second, the game could sometimes become tedious when it required lengthy computations and throwing of dice. Many people recognized that these problems could be solved with a microcomputer. The first company to make a D&D style computer game available was Automated Simulations. Their TEMPLE OF APSHAI program has been very successful. They also market a number of other D&D-style games.

So far, however, few games have been marketed that truly capture the spirit of D&D. There are several reasons for this. First, most D&D-players are young and don't have the money for such packages. Second, the adventure games have slowly absorbed many of the ideas of the D&D games. There was a time when we could easily distinguish an adventure from a D&D game with several factors.Adventures were pure text games, while D&D games used some graphics. Adventures were puzzles; D&D games were true games. Adventures were by and large nonviolent, while D&D games tended to be quite violent. Lately, we have seen adventures taking on many of the traits of D&D games, so that it is now harder to tell the difference between them.

An ideal example of this phenomenon is ALI BABA AND THE FORTY THIEVES (trademark of Quality Software), a game with the basic elements of both adventures and D&D games. The player must search through a large maze to find and rescue a princess, but on the way he must fight monsters and thieves. The player, as Ali Baba, possesses personal characteristics (dexterity, speed, etc.) that are reminiscent of a D&D game, but he must explore the maze as in an adventure. For these reasons, I feel that this game cannot be classified as either an adventure or a D&D game, but rather is a solid example of the merging of these two genres into a new class of games, the fantasy role-playing ("FRP") games. This suggests that we will see more such games combining the "search and discover" aspects of adventure games with the "defeat opponents" aspects of D&D games.

Wargames

A third class of strategy games is provided by the wargames. Noncomputer wargames as a gaming form have a long heritage. Commercial wargaming goes all the way back to the 1880's with an American wargame design using wooden blocks. The British have long had a dedicated group of wargamers using miniature models of soldiers and very complex rules. Their games, called miniatures games, have grown in popularity and are now played in the USA. But the largest segment of wargamers in recent years has been the boardgamers. This hobby was founded in the late 1950's by Charles Roberts, who founded the Avalon-Hill Game Company and created such classic games of the 60's as BLITZKRIEG, WATERLOO, and AFRIKA KORPS (all trademarks of the Avalon-Hill Game Company). During the 1970's a new company, Simulations Publications, Inc., turned board wargaming into the largest segment of wargaming.

Wargames are easily the most complex and demanding of all games available to the public. Their rules books read like contracts for corporate mergers and their playing times often exceed three hours. Wargames have therefore proven to be very difficult to implement on the computer; we have, nevertheless, seen entries.

The computer wargames available now fall into two distinct groups. The first group is composed of direct conversions of conventional boardgames. COMPUTER BISMARK, COMPUTER AMBUSH, and COMPUTER NAPOLEONICS (trademarks of Strategic Simulations, Inc.) are examples of this group of games. These games illustrate the folly of direct conversion of games of one form to another. They parrot successful and respected boardgames, but are themselves not as successful. Because they attempt to replicate boardgames, they are, like boardgames, slow and clumsy to play.

The second group of computer wargames are less slavish in their copying of board wargames. My own EASTERN FRONT 1941 is generally considered to be the best of this lot, primarily because of its graphics and human engineering features. Many of the games in this category are experimental; hence the successes are outnumbered by the failures. Avalon-Hill's first entries into the computer wargaming arena were such experiments. My own TANKTICS game is an early experiment that once was the most advanced commercially available wargame (it was the ONLY commercially available wargame when I first released it in 1978). It is now generally regarded as a mediocre game. It can safely be said that computer wargaming is not a well-developed area of computer gaming. For the moment, computer wargaming is too closely associated with board wargaming in the minds of the public and most designers; until it can shake free from the constraints of boardgames and, establish its own identity, computer wargaming will evolve slowly.

Games of Chance

Games of chance have been played for thousands of years; their implementation onto computers is therefore quite expectable. They are quite easy to program, so we have seen many versions of craps, blackjack, and other such games. Despite their wide availability, these games have not proven very popular, most likely because they do not take advantage of the computer's strong points. Furthermore, they lose the advantages of their original technologies. These games demonstrate the folly of mindlessly transporting games from one medium to another.

Educational and Children's Games

The fifth category of strategy games is that of the educational games. Although all games are in some way educational, the games in this set are designed with explicit educational goals in mind. This group is not heavily populated as yet, perhaps because the people interested in educational uses of computers have not yet concentrated much attention on game design. The Thorne-EMI puzzles are good entries in this field, and APX sells a collection of very simple children's games that have some educational value. Several of the classic computer games are educational: HANGMAN, HAMMURABI, and LUNAR LANDER are the three most noteworthy of these early educational games. SCRAM (a nuclear power plant simulation) and ENERGY CZAR (an energy economics simulation) are two of the more complex programs in the educational games field. My favorite entry to date is ROCKY'S BOOTS (trademark of The Learning Company), a children's game about Boolean logic and digital circuits. The child assembles logic gates to create simulated logical machines. This game demonstrates the vast educational potential of computer games. Educators are becoming more aware of the motivational power of computer games; with time we can expect to see more entries of the caliber of ROCKY'S BOOTS.

Interpersonal Games

I have been exploring a class of games that focus on the relationships between individuals or groups. One such game explores gossip groups. The player exchanges gossip with up to seven other computer-controlled players. The topic of conversation is always feelings, positive or negative, expressed by one person for another. Adroit posturing increases popularity. Similar games could address corporate politics, soap-opera situations, gothic romances, international diplomacy, and espionage. Although the category is undeveloped, I believe it is important because it addresses fantasies that are very important to people. Many other art forms devote a great deal of attention to interpersonal relationships. It is only a matter of time before computer games follow a similar course.

CONCLUSIONS

This concludes the description of my proposed taxonomy. Obviously, this taxonomy has many flaws. This is primarily because the basis of division is not any grand principle but is instead historical happenstance. There is no fundamental reason why wargames should be treated any differently than D&D games. Yet, both game systems evolved separately and are historically quite distinct. Similarly, the creation of an educational games category is my response to the efforts of educators to create educational games. With the passage of time, market forces will assert themselves, and a more organized and consistent taxonomy will become possible. People have tried to create educational games, so we now have them. My taxonomy is a patchwork because the set of available computer games is a patchwork.

This taxonomy suggests a number of observations about the state of game design with computers. For example, it should be obvious that there are very few basic scenarios for skill-and-action games, each scenario taking one category. The archetypical game in each category spawned a whole family of imitators, variations, and improvements. Moreover, the archetypical game in each category was seldom the big moneymaker; instead, the archetypical game was followed by several successor games that improved on it until one game hit the nail on the head. Thus we have COMBAT leading to SPACE INVADERS in the combat category, DODGE 'EM leading to PAC-MAN in the maze category, and PONG leading to SUPERBREAKOUT in the paddle category.

Another lesson that arises from this taxonomy is that the Analogy games are still in a very poorly-developed state in comparison to the S&A games. While S&A games have fairly clear-cut categories that make sense, the categories in strategy games are less satisfying and the distinctions between categories are muddier. This ambiguity suggests that much creative opportunity remains in the strategy games field.

A taxonomy reflects the body of material it attempts to organize. The state of computer game design is changing quickly. We would therefore expect the taxonomy presented here to become obsolete or inadequate in a short time. New taxonomies must be created to reflect the changes in the marketplace in the next few years. For the present, however, the proposed taxonomy can provide us with an organized way to view the menagerie of games while suggesting new areas to explore.

CHAPTER FOUR
The Computer as Game Technology

GAME TECHNOLOGIES

Every art form is expressed through a physical medium. The control and manipulation of this physical medium is a technical problem that the artist must master before she can express herself through it. Thus, the sculptor must thoroughly understand the limitations of marble, brass, or whatever medium she uses. The painter must fully understand the technology of paint and the behavior of light. The musician must be deeply skilled in the technology of sound creation. So too must the computer game designer thoroughly understand the medium with which she works. The computer offers special possibilities and imposes special constraints on the designer. In this chapter I will discuss the nature of these possibilities and constraints. A few examples of a game technology operating at a simpler level may help establish basic principles.

Cards are one such simpler game technology. We have here a very simple set of physical equipment---52 pieces of cardboard, imprinted on one side with a uniform pattern, and on the other side with distinct symbols. The key traits of this equipment can be summarized as follows:

1) There are many cards.
2) Each card is unique.
3) Each card possesses a numeric value.
4) Each card possesses a suit, a two-bit value.
5) The identity of a card can be selectively revealed.
6) Each card is easily assignable to an owner.

These six characteristics are the fundamental properties of the card, technology that constrain the design of all card games. Each characteristic carries implications for game design with cards. Some things are easy to do with this technology and some things are hard to do with it. For example, games of probability are easily implemented with this technology, for the two characteristics (numeric value and suit) can be combined into many, many sets according to laws of probability. The limitations on information created by the cards can be used to create games of guesswork and intuition. Indeed, one of the most intriguing of card games, poker, is based not so much on cold probability assessments as on the deceptions made possible by the limited information employed in the game.

Like every other technology, cards also have their weaknesses. For example, it would be very tricky to design a card game for more than 52 players, because there are only 52 cards in one deck. It would also be very difficult to design a good skill-and-action game using cards as a technology. Another tough design challenge would be a good athletic game using cards. Games meeting these conditions could be implemented with cards, but they probably would not be very good games.

This doesn’t mean that cards are a bad game technology. Some things can be done well with cards, and other things can’t. Another game technology, that of the boardgame, is somewhat more flexible than cards. This technology is so much more flexible than cards that I cannot devise a list of defining characteristics as I could with cards. Boardgames can be described but not rigorously defined. They use a large surface of paper or cardboard on which are printed various images, normally taking the form of a stylized map. Frequently the area represented on the map is divided into discrete regions by either a regular geometric pattern (rectgrid or hexgrid), a segmented path to be traversed, an irregular division of regions, or a network of points connected by paths. The map itself remains the same throughout the game; players designate changes in the situation with a set of markers that can be moved about on the map. Sometimes a randomizing machine is used to determine outcomes of random processes; a spinner or dice are most frequently used for this purpose. Sometimes cards from a special set are drawn to provide this randomizing function.

This technology has proven to be very successful for game designers. It easily accommodates groups of players, and with appropriate game design can address a very wide range of gaming situations. Chess is certainly the all-time classic boardgame. MONOPOLY (trademark of Parker Brothers), a successful early boardgame, concerns real estate transactions. Other boardgames have addressed such topics as life goals, solving a murder, and race relations. The most ambitious modern boardgames are the wargames. Among these are games with boards of some 25 square feet, several thousand movable pieces, and a rules manual 50 pages long. A small industry has sprung up around these designs, complete with historical research, star designers, and its own jargon.

Boardgames provide a flexible and powerful technology for game designers. In recent years, however, we have seen a stagnation in designs with the board technology. Many new boardgames look like cheap copies of MONOPOLY. Wargames, after showing a burst of creative energy in the 60’s and 70’s, have started to stagnate. Few fundamentally new ideas are being introduced in this arena. It may be that we have mined this vein to the limits of its productive capacity.

What are the limitations of this technology? First and foremost, it is very difficult to maintain privileged information in a boardgame. All players can see the board and the position of all the markers. Second, the mechanics of handling all the pieces must be managed by the players. In some cases this can become a sizable chore, as in the aforementioned monster wargame. For this reason most boardgames are long affairs, frequently filling an evening. Short boardgames playable in twenty minutes or less are quite rare. Finally, should the pieces be disturbed, a boardgame is easily ruined.

The central point of the preceding discussion is that every game utilizes some technology, and that each technology has strengths and weaknesses, things that it can do well and things that it can do poorly. The astute game designer must fully grasp the strengths and weaknesses of the technology s/he uses. Let us now examine the computer as a game technology.

COMPUTERS

The most striking feature of the computer in a game context is its responsiveness. Responsiveness is vital to the interactiveness that is so important to any game. The computer can respond to the human player's wishes in a huge variety of ways. If the action in a card game or board game starts to drag, the players have no choice but to plod through it or take desperate measures. There is no reason why a computer game in similar straits could not speed up the game on demand. It could change the length of the game, or the degree of difficulty, or the rules themselves. SPACE INVADERS (trademark of Taito America) for the ATARI 2600 provides an example of such performance. The player can select one or two-player versions, visible or invisible invaders, stationary or moving shields, fast or slow bombs, and a variety of other options. In effect, the player chooses the rules under which he plays. The game is responsive to his wishes.

This responsiveness arises from the computer's plasticity. The computer is dynamic; it imposes little constancy on any element of the game. Boardgames, cardgames, and athletic games all have invariables that constrain the designer. Once you have printed up 100,000 game boards it becomes very difficult to modify the map. Try as we may, we can't have 53-card stud; the card decks aren't made that way. And should some miracle of science produce more elastic footballs that kick further, we will not be able to simply extend football stadiums without spending many millions of dollars. The computer is far less restrictive. All of the game parameters are readily changed, even during the course of the game. There is nothing stopping us from creating a football game in which the goal post recedes from the visiting team. Territories in wargames can be switched around the map of the globe more easily than we move a chair in the living room. This flexibility is of paramount importance tothe game designer. As yet, it has been put to little use.

A second feature of great value is the computer's ability to Motion as game referee. All other game technologies demand that somebody take the time to handle the administrative responsibilities of the game. Athletic games are most demanding; they require several impartial referees or umpires to administer the rules of the game and adjudicate disputes. Card games and boardgames require that the players also function as referees. This is seldom a problem with card games, but it can be a big load with boardgames, especially the more complex ones such as the wargames. Rules disputes and administrative foul-ups are part of the unavoidable dangers of boardgames. The computer can eliminate all of these problems. It can administer the game, freeing the player to concentrate on playing it. This allows one other big advantage: the computer can implement complex arithmetic and logical rules. With other technologies, game rules must be overly simple because the humans implementing them cannot be trusted to perform simple numerical computations. The computer eliminates this restriction.

For example, in the original version of EASTERN FRONT 1941, I was able to use exceptionally complex victory calculations. Most board-level wargames about the eastern front in World War II assign victory points for captured cities, and perhaps for casualties inflicted and sustained. A more complex calculation recognizing the realities of the campaign would be too tedious for human computation. Original EASTERN FRONT 1941 was able to calculate not only cities captured and casualties inflicted and sustained, but also the eastward progress of every German unit as well as the westward resistance of every Russian unit. The game is thereby able to provide a more realistic and meaningful measure of the player's performance.

The third advantage of the computer is in real-time play. Other game technologies must have pauses and procedural delays while administrative matters are dealt with. The computer is so fast that it can handle the administrative matters faster than the humans can play the game. This makes real-time games possible. Skill-and-action games are the direct result. The speed of the computer also eliminates the need for turn-sequencing so common in card games and boardgames.

The fourth strength of computers for game design purposes is their ability to provide an intelligent opponent. All other games require a human opponent (exception: solitaire card games, but they are actually puzzles rather than games). The greatest success so far has been with chess-playing games. Programs written for microcomputers can now play a chess game well enough to challenge most non-rated players. These games represent the best we have achieved to date in game artificial intelligence. Most games are far less intelligent. Instead, they rely on overwhelming numerical advantage to make up for the superior intelligence of the human player. With the passage of time, we can expect to see more intelligent algorithms that provide more challenging play from the computer.

The fifth strength of the computer is its ability to limit the information given to the players in a purposeful way. This capability can be of great value. Limited information forces the player to use guesswork. The nature of this guesswork can be very intriguing. For example, guessing a random number between one and ten is not a very interesting challenge, but guessing your opponent's resources based on your assessment of his actions and personality is a far more interesting exercise. When the guesswork is included in the framework of a complex and only partially known system, the challenge facing the human player takes on a decidedly real-life texture.

Limited information provides another important bonus. Games are an unreal representation of a real-world problem. The player must use his imagination to make the unreal situation seem real. Limited information encourages the use of imagination. If we know all the pertinent facts, we can treat the problem as a simple problem of deduction. But if we know only a portion of the truth, our minds grope for an appropriate model on which to hang our projections. What model could be more appropriate than the reality that the game attempts to re-create? We are therefore forced by lack of information to imagine ourselves in the real-world predicament postulated by the game so that tie may deal with the problems imposed by the game. In the process, the illusion of reality is heightened. The game draws us into its fantasy world more effectively.

The sixth feature offered by computers is their ability to utilize data transfer over telephone lines for game play. The use of telecommunications for game play makes possible game structures that are out of the reach of other technologies. It allows us to create games with huge numbers of players. Until now, administrative problems have made it necessary to limit the number of players in any game. Six players is a rough upper limit for a non-refereed game; twelve players will require several referees and twenty players or more will require many referees. Obviously, games with hundreds of players will face many administrative problems. Indeed, the logistic problems of assembling all th players are themselves prohibitive. All these problems are solved by computers linked through a telecommunications network. With this technology it should be possible to design games knitting together thousands of players scattered all over the continent. Players could drift into and out of the game at their whim; with large numbers of players the coming and going of individuals will not be detrimental to the game.

Like any technology, computers have weaknesses as well as strengths. The first and most painful weakness is the limited I/O capability of most computers. The computer itself may be supremely responsive, but if the human player can't tell it what he wants, or fails to understand the computer's response, the computer's effective responsiveness is nil. In other words, the computer must communicate its responsiveness to the human; it does so through I/O. Most output is through graphics and sound; most input is through keyboard, joystick, and paddle.

Graphics are the first component of output. Good graphics are hard to come by. Even the Atari Home Computer System, boasting the best graphics in the microcomputer world, has graphics limitations that severely constrain the game designer. You simply cannot show all the graphic details that you would like to show. For example, I suspect that few boardgame boards could be duplicated on a single screen by this machine. The number of colors, the mixing of text with high-resolution graphics, and the size of the board all combine to make the task hopeless. It is possible to use a variety of tricks to produce something that is functionally similar to any given game board. We could reduce the number of colors displayed, we could dispense with text, and we could design an oversize display through which the user must scroll. EASTERN FRONT 1941 uses all of these tricks, and the result is quite usable, but the game wends a tortuous path past the graphics constraints of the computer.

Of course, the computer also boasts some graphics advantages. I have yet to see the boardgame that could show animation or change itself around the way a computer game could. These sensory features can dramatically increase the impact of any game. So the graphics picture is not all bad.

Another I/O restriction comes from the input requirements. Input to the computer must come in through the keyboard or the controllers. This can make things very difficult for the game designer. In the first place, you can't say much with a joystick or keyboard. A joystick can say only five fundamental words: "up", "down", "right", "left", and "button". A keyboard can say more, but only through a lengthy and error-prone sequence of key presses. The human who wishes to express a meaningful communication to the computer must successfully enter a long and clumsy string of simple commands. Input is made even more difficult by the indirectness of keyboards and joysticks. There is very little about such devices that directly corresponds to real-world activities. Actions that are simple and obvious with other technologies become arcane with the computer. If I give you a bat and tell you that your goal in baseball is to hit the ball, you will have few problems deciding that you should swing the bat at the ball. A computer baseball game is not so easy to figure out. Do you press H for "hit" or S for "swing" or B for "bat"? Do you press the START key or press the joystick trigger? Perhaps you should swing the joystick by its cable at the ball displayed on the television screen.

After I/O, the second weakness of the personal computer is its single-user orientation. These machines were designed for one person to use while a seated at a desk. If two people are to use it, they may be forced to exchange seats, a clumsy and distracting procedure. With joysticks or paddle controllers the problem is diminished but not eliminated. This is one reason why so many computer games are solitaire and has led to the accusation that computer games are anti-social. A boardgame invites a group of people to sit around the table. A computer game encourages one player, accepts two, and discourages more.

The final weakness of the computer to be considered here is the requirement that it's programmed. No other game technology imposes so harsh a demand on the game designer. The boardgame designer can sketch an adequate board and construct some simple playing pieces that will serve quite effectively. When the time comes to produce the game, the designer's amateur efforts can be handed to a professional who can produce a quality version of the prototypes made by the designer. For this reason the designer need not concern himself with the technical aspects of game production.

The computer game designer does not have life so easy. The design must be implemented on the computer by programming it. Programming itself is a tedious and difficult process, and it is not easily delegated, for the programming effort exerts a major influence over the design process. Implementing a design well is a major hurdle for any computer game designer.

DESIGN PRECEPTS FOR COMPUTER GAMES

How do we translate an understanding of these strengths and weaknesses of the computer into a set of guidelines for game designers? The characteristics described above imply a variety of precepts.

PRECEPT #1: GO WITH THE GRAIN

(Introducing our idiot cartoon hero. A rocket lies on its side. A wheel-less baby carriage lies nearby. Our hero is walking from the baby carriage toward the rocket, carrying some baby carriage wheels and a hammer.)

The first-precept can be summarized with the aphorism: "Work with the grain of the machine, not against it." Too many game designers set out with unrealistic goals. They attempt to force the machine to perform tasks for which it is not well-suited. In saying this, I do not excuse lazy programming. We must remember that the computer is the servant of the human; the convenience of the computer is not of interest to the designer. Our goal is to extract maximum performance from the computer, to make it work its best. We can only do this by making it perform functions which it performs well.

Case In Point: Hexgrids

An example of this principle might be illuminating. Board wargames are traditionally executed on maps that use a hexgrid system. This regularizes movement and defines positions.

Hexgrids are preferred over rectgrids for several reasons. First, rectgrids have diagonals; two units can be diagonally adjacent. This situation can be very messy; rules to cope with it are always burdensome and confusing. Hexgrids have no diagonals, so they eliminate the problem. Second, hexgrids allow a player a choice of six directions in which to move, while rectgrids offer only four directions. The greater range of choice allows the player to control more finely the movements and positioning of his pieces.

It therefore seems natural that designers of computer wargames would also use hexgrids for their maps. Indeed, most computer wargames do so ---but it is a terrible mistake. The hex does have advantages, but it imposes a penalty on computer wargames that does not apply to boardgames. You can print anything you desire on a piece of paper, but the graphic display of the computer is not so accommodating. The display system of the television set is fundamentally rectangular in its architecture. Horizontal lines are stacked in a vertical sequence. Such a display can very easily handle rectangular shapes; hexagonal shapes just don't work very well. To draw a hex the program must draw four diagonal lines, each one composed of a set of staggered dots. To make the hexgrid recognizable the lines must be surrounded by an exclusion zone at least one pixel wide; this consumes a large portion of the screen area if the hexes are small and dense. If they are larger, less screen area is consumed by the gridwork but fewer hexes can be shown on a single screen. Moreover, joysticks cannot be easily used with hexgrids because joysticks are set up with rectangular geometry. I do not wish to imply that hexgrids cannot be implemented on personal computer displays; on the contrary, they have already been implemented on many personal computers. The problem is that they are clumsy to display, lacking in graphic detail, and difficult to use. They just don't work smoothly. A topologically identical solution has been used in a few games: horizontally staggered rows of squares ("bricks") are used in place of hexes. This system retains the flexibility of hexes while imposing fewer display problems; it remains very difficult to use with a joystick.

For these reasons I went back to rectgrid for EASTERN FRONT 1941. My decision was not based on laziness or unwillingness to tackle the problem of hexgrids; indeed, I had already solved the problem with another game (TACTICS) and could easily have transported the code. The experience I gained in working with the earlier code convinced me that hexgrids weren't so important. The success of EASTERN FRONT 1941 seems to indicate that the lack of hexgrids need not impose a handicap.

PRECEPT #2: DON'T TRANSPLANT

(Now our hero is plummeting earthward from the top of a cliff, furiously flapping makeshift wings attached to his arms.)

One of the most disgusting denizens of computer gamedom is the transplanted game. This is a game design originally developed on another medium that some misguided soul has seen fit to reincarnate on a computer. The high incidence of this practice does not excuse its fundamental folly. The most generous reaction I can muster is the observation that we are in the early stages of computer game design; we have no sure guidelines and must rely on existing technologies to guide us. Some day we will look back on these early transplanted games with the same derision with which we look on early aircraft designs based on flapping wings.

Why do I so vehemently denounce transplanted games? Because they are design bastards, the illegitimate children of two technologies that have nothing in common. Consider the worst example I have discovered so far, a computer craps game. The computer displays and rolls two dice for the player in a standard game of craps. The computer plays the game perfectly well, but that is not the point. The point is, why bother implementing on the computer a game that works perfectly well on another technology? A pair of dice can be had for less than a dollar. Indeed, a strong case can be made that the computer version is less successful than the original. Apparently one of the appeals of the game of craps is the right of the player to shake the dice himself. Many players share the belief that proper grip on the dice, or speaking to them, or perhaps kissing them will improve their luck. Thus, the player can maintain the illusion of control, of participation rather than observation. The computer provides none of this; the mathematics may be the same, but the fantasy and illusion aren't there.

In one way or another, every transplanted game loses something in the translation. It may also gain something, but it always loses something. This is because any game that succeeds in one technology does so because it is optimized to that technology; it takes maximum advantage of the strengths and avoids the weaknesses. The transplanted version uses the same design on a different set of strengths and weaknesses; it will almost certainly be a lesser product. Any memorable artistic expression is as much a creature of its vehicle of expression as it is an image of a thought. Shakespeare reads best in Elizabethan English; translation to modern English loses some of the verve and linguistic panache that we find so entertaining. The rhetoric of Isocrates, dull and drab in English, acquires a compelling cadence in Greek that thrills the listener. Great books that touched our souls when we read them almost always disappoint us when we see their movie adaptations. Why should computer games be immune to this law of loss on translation?

PRECEPT #3: DESIGN AROUND THE I/O

(Now our man is putting the final touches onto a gigantic and complex machine with pipes, valves, smokestacks, and many wires. On the front face of the machine is a sign that reads, "Make your move". Underneath it are two buttons labeled "CHOICE A" and "CHOICE B". To the right of this are a pair of illuminable signs, one reading, "YOU WIN!!!", the other reading "YOU LOSE!!!")

As I mentioned earlier, the computer's ability to calculate is a strength, but it's I/O is a weakness. Thus, the primary limitation facing the computer game designer is not in the machine's ability to perform complex computations, but in the I/O: moving the information between the computer and the human player. The game must be designed in such a way that the information given to the player flows naturally and directly from the screen layout and sound output. I have seen far too many games with good game structures that were ruined by poor I/O structures. The user was never able to appreciate the architectural beauties of the game because they were buried in a confusing display structure. Even worse are the games that sport poor input arrangements, especially poor use of the keyboard. Most game players find keyboards difficult to use smoothly. Difficulty can in some cases create challenge, but difficulties with keyboards generate only frustration. The implementation of the game will be dominated by the limitations of I/O. What can and cannot be displayed, what can and cannot be inputted, these things must decide the shape of the same.

A comparison of two of my own games provides an excellent example of the importance of I/O structures. EASTERN FRONT 1941 and TANKTICS (trademark of Avalon-Hill) are both wargames dealing with World War II. Both provide reasonably intelligent opponents, complex detailed simulation, a rich variety of options, and thought-provoking strategic challenges. In all these respects, they are roughly equivalent. They differ primarily in their I/O. EASTERN FRONT 1941 was designed around its I/O; it provides clean, informative graphics and an intuitively obvious joystick input system. By contrast, TANKTICS was designed around its game structure; its keyboard input system is clumsy and confusing and its alphanumeric; screen display is cryptic. EASTERN FRONT 1941 has been acclaimed by the critics and has received awards; TANKTICS has been panned. The quality of a game's I/O structure is crucial to its success.

PRECEPT #4: KEEP IT CLEAN

(Our hero at the controls of his custom motorcycle, 20 feet long, equipped with numerous rear-view mirrors, power steering, brakes, and throttle, adjustable seats, adjustable handlebars, windshield wipers on several windshields and on each mirror, television, hamburger dispenser, etc. The artist can use imagination here.)

Many game designers fail to keep the overall structure of their game close to heart as they develop the details of the game structure. As they encounter design problems, they resort to quick patches that are grafted onto the main game structure without due regard to the impact such grafts have on the overall cleanliness of the design. A game must have artistic unity if it is to have emotional impact on its audience. Artistic unity can only be achieved by sticking close to the theme and eschewing distracting details.

I refer to any factors that do not comport with the central theme of the game as "dirt." The debilitating nature of dirt is seldom recognized, because dirt also endows a game with "color", namely the texture or feel that makes the game seem real. It is true that proper use of this kind of color will indeed enhance a game. However, the game designer must realize that color is obtained at the price of a certain amount of dirt. The critical quantity then becomes the ratio of color to dirt. The designer always desires the highest possible ratio, but sometimes, to increase the absolute amount of color, s/he must accept some more dirt. In all cases, the inclusion of dirt into a game must be a conscious trade-off on the part of the game designer, not an accident springing from the desire to quickly resolve some irritating problem.

Dirt most often arises from special-case rules that are applied rarely. For example, EASTERN FRONT 1941 has a number of special-case rules that add dirt to the game. The worst is the rule forbidding Finnish units to attack. Inasmuch as there are only two Finnish units, this rule has very little significance to the game as a whole, yet the player must still be aware of it. It clutters up the game and the player's mind without adding much. (I had to put it in to solve a design problem: what's to stop the Finns from taking Leningrad all by themselves?)

A less dirty rule provides that Axis allies (Rumanian, Hungarian, and Italian units) fight with less determination than the Germans. There are six of these units in EASTERN FRONT 1941; thus, the rule is not quite so special a case and hence not quite so dirty.

There is a rule in EASTERN FRONT 1941 that armored units move faster than infantry units. EASTERN FRONT 1941 has many armored units; thus, this rule is not a particularly special case, because it applies to a goodly portion of all units. It is therefore not dirty.

I can generalize these observations by saying that the narrower the range of application of a rule, the dirtier it is. My precept against dirt thus requires the designer to formulate a set of rules that cover the entire game situation without recourse to special case rules. In the perfect game design, each rule is applied universally. We can never achieve the perfect design, but we can and should strive to give each rule the widest possible application. The player must consider the implications of each rule while making every decision in the game.

There is a school of game design that I derisively label the "humongous heap" school of game design. Perpetrators of this philosophy design a game by selecting a simple structure and piling onto it the largest possible jumble of special odds and ends (they call them "features"). These people design with a shovel instead of a chisel. They confuse magnitude with magnificence, intricacy with insight.

PRECEPT #5: STORE LESS AND PROCESS MORE

(Our idiot is juggling. Beside him another man is juggling five or six numbers comfortably and happily. The idiot is staring upward in stark terror, arms outstretched in a futile attempt to catch an avalanche of numbers that will simply crush him.)

The role of information storage in a computer is often misunderstood. A computer is not primarily an information storage device; it is instead an information processing device. Information storage is a necessary precondition for information processing, but it is not an end in itself. Greater amounts of stored information permit greater amounts of information processing, but if the processing capability is insufficient to realize the full potential of the storage, then that storage is wasted. The ideal program strikes the optimum balance between storage and processing. Most game programs I have seen are long on storage and short on processing. This is because data for storage facts are easier to come by than process-intensive material-program code. In taking the path of least resistance, most game designers end up going downhill.

Thus, a game that sports huge quantities of static data is not making best use of the strengths of the machine. A game that emphasizes information processing and treats information dynamically is more in tune with the machine. Relegate all static information to a rules book; paper and ink are still a better technology than personal computers for storing static information. Information that lies around and does little, that must be dusted off before using, has no place inside the microcomputer. As you look over your program listing, you should inspect each byte and ask yourself, "Am I getting my money's worth from this byte? Is it working hard for me, doing useful things frequently? Or is this a lazy byte that sits idle for hours and is used only rarely?" Fill your program with active bytes that do things, not lazy bytes.

Lazy bytes are often associated with dirty rules (they like to hang out together in sleazy pool halls). Dirty rules are special cases that occur rarely. If they occur rarely, the bytes associated with them are not used often, hence they are lazy bytes.

Another argument in favor of this precept arises from more fundamental considerations on the nature of game play. Interactiveness is a central element of game enjoyment. As mentioned earlier, the computer's plasticity makes it an intrinsically interactive device. Yet, the potential inherent in the computer can easily go unrealized if it is programmed poorly. A program emphasizing static data is not very dynamic. It is not plastic, hence not responsive, hence not interactive. A process-intensive program, by contrast, is dynamic, plastic, responsive, and interactive. Therefore, store less and process more.

One last argument has more to do with games than computers. (You will remember from Chapter One that a game is distinguished from a story by the network of options that a game has, as opposed to the single richly-developed thread of a story. Much of the quality of a story is derived from the richness of the information it contains. A story is thus all information and no processing. A game derives its quality from the richness of the network of options it presents. These options are only accessible through the process-intensive aspects of the game. Games that are information-rich and process-poor are closer to stories than to the ideal game.

PRECEPT #6: MAINTAIN UNITY OF DESIGN EFFORT

(Our hero is now a pole vaulter handcuffed to a high jumper. They are attempting to leap; their attempt is obviously going to collapse in a tangle of limbs. Their facial expressions indicate that they are aware of the likely outcome.)

Games must be designed, but computers must be programmed. Both skills are rare and difficult to acquire, and their combination in one person is even more rare. For this reason many people have attempted to form design teams consisting of a nontechnical game designer and a nonartistic programmer. This system would work if either programming or game design were a straightforward process requiring little in the way of judicious trade-offs. The fact of the matter is that both programming and game design are desperately difficult activities demanding many painful choices. Teaming the two experts together is rather like handcuffing a pole vaulter to a high jumper; their resultant disastrous performance is the inevitable result of their conflicting styles.

More specifically, the designer/programmer team is bound to fail because the designer will ignorantly make unrealistic demands on the programmer while failing to recognize golden opportunities arising during the programming. For example, when I designed the game ENERGY CZAR (an energy-economics simulation game), I did not include an obviously desirable provision for recording the history of the player's actions. During the final stages of the game's development, virtually everyone associated with the project suggested such a feature. From technical experience, I knew that this feature would require an excessive amount of memory. A nontechnical designer would have insisted upon the feature, only to face the disaster of a program too big to fit into its allowed memory size.

Another example comes from EASTERN FRONT 1941. While writing the code for the calendar computations, I realized that a simple insertion would allow me to change color register values every month. I took advantage of this opportunity to change the color of the trees every month. The improvement in the game is small, but it cost me only 24 bytes to install, so it proved to be a very cost-effective improvement. A nontechnical game designer would never have noticed the opportunity; neither would a nonartistic programmer.

There is no easy way to produce good computer games. You must start with a good game designer, an individual with artistic flair and a feel for people. That person must then learn to program. The opposite direction of development (from programmer to designer) will not work, for programmers are made but artists are born. When eventually you get that rare individual who is both designer and programmer, then you can subordinate designers and programmers underneath her, so as to multiply her creative power. In the process, the subordinates will receive valuable training. In all cases, the creative process must be unified in a single mind. Committees are good for generating red tape, deferring decisions, and shirking responsibility, but they are useless when it comes to creative efforts.

CONCLUSION

In this chapter I have discussed the computer as a technology for game design. Discussions of computers and their impact on society tend to become polarized between the "gee whiz school and the cynical school. The former group sees a rosy future of countless triumphs wrought by the computer -- "Every day in every way, better and better." The latter group sees computers as a dehumanizing threat, a waste of time, or yet another vehicle for the expression of human perfidy. In this chapter, I have tried to present computers as just another technology, like hammer and nails, clay and stone, paper and ink. Like any technology, they can do some things well. Like any, technology, they do some things poorly. The artist's role is to deviously evade their weaknesses while capitalizing their strengths to greatest advantage.

CHAPTER FIVE
The Game Design Sequence

Game design is primarily an artistic process, but it is also a technical process. The game designer pursues grand artistic goals even as she grinds through mountains of code. During the process of developing the game, she inhabits two very different worlds, the artistic world and the technical world. How does one manage the integration of such dissimilar worlds? In short, how does one go about the process of designing a computer game? In previous chapters I have touched on some of the questions related to this process; I have also laid down a few precepts. In this chapter I will suggest a procedure by which a computer game could be designed and programmed.

The procedure I will describe is based on my own experiences with game design, and reflects many of the practices that I use in designing a game. However, I have never used this procedure in a step-by-step fashion, nor do I recommend that any person follow this procedure exactly. In the first place, game design is far too complex an activity to be reducible to a formal procedure. Furthermore, the game designer's personality should dictate the working habits she uses. Even more important, the whole concept of formal reliance on procedures is inimical to the creative imperative of game design. Finally, my experience in game design is primarily with personal computers, so my suggestions are not completely applicable to arcade game designers or home video game designers. I therefore present this procedure not as a normative formula but as a set of suggested habits that the prospective game designer might wish to assimilate into her existing work pattern. With these important qualifications in mind, let us proceed.

CHOOSE A GOAL AND A TOPIC

This vitally important step seems obvious, yet is ignored time and time again by game designers who set out with no clear intent. In my conversations with game designers, I have many times discerned an indifference to the need for clear design goals. Game designers will admit under close examination that they sought to produce a "fun" game, or an "exciting" game, but that is more often than not the extent of their thinking on goals.

A game must have a clearly defined goal. This goal must be expressed in terms of the effect that it will have on the player. It is not enough to declare that a game will be enjoyable, fun, exciting, or good; the goal must establish the fantasies that the game will support and the types of emotions it will engender in its audience. Since many games are in some way educational, the goal should in such cases establish what the player will learn. It is entirely appropriate for the game designer to ask how the game will edify its audience.

The importance of a goal does not become obvious until later in the game design cycle. The crucial problems in game development with microcomputers are always problems of trade-offs. Everything that the game designer wants to do with her game costs memory, and memory is always in short supply with microcomputers. Thus, the designer must make trade-offs.

Some game features can be included, and some must be rejected. At two o'clock in the morning, when you must face the awful decision of rejecting one of two highly desirable features, the only criterion you will have for making this painful choice will be the goal you have established for the game. If your goals are clear, your decision will be painful but obvious; if your goals are murky, you may well make the wrong choice, and whatever you choose, you will never know if your decision was correct.

How do you select a proper goal? There is no objective answer to this question; the selection of a goal is the most undeniably subjective process in the art of computer game design. This is your opportunity to express yourself; choose a goal in which you believe, a goal that expresses your sense of aesthetic, your world view. Honesty is an essential in this enterprise; if you select a goal to satisfy your audience but not your own taste, you will surely produce an anemic game. It matters not what your goal is, so long as it is congruent with your own interests, beliefs, and passions. If you are true to yourself in selecting your goal, your game can be executed with an intensity that others will find compelling, whatever the nature of the game. If you are false to yourself, your game will necessarily be second-hand, me-too.

There are situations in which it is not quite possible to attain the purity of this artistic ideal. For example, I would not claim that only immature, childish people should design games for children. Nor would I suggest that good shoot-'em-up games can only be done by shoot-'em-up personalities. The realities of the marketplace demand that such games be written, and it is better that they be written by mature professionals than by simpering fools. Such emotionally indirect games, however, will never have the psychological impact, the artistic power, of games coming straight from the heart.

Once you have settled on your goal, you must select a topic. The topic is the means of expressing the goal, the environment in which the game will be played. It is the concrete collection of conditions and events through which the abstract goal will be communicated. For example, the goal of STAR RAIDERS apparently concerns the violent resolution of anger through skillful planning and dexterity. The topic is combat in space. The goal of EASTERN FRONT 1941 concerns the nature of modern war, and especially the difference between firepower and effectiveness. The topic is the war between Russia and Germany.

Most game designers start off by selecting their topic, with their goals subordinated to their topic. Indeed, they commonly describe a game under development by its topic rather than its goal. When I tell other designers that I am working on a game about leadership, I am met with quizzical expressions. Is it a space game, or a wargame, or a dungeon game, they wonder; they seem satisfied when I tell them it's a game about King Arthur. It is a serious mistake to subordinate the goal to the topic. Although your initial flash of inspiration may focus more on the topic than the goal, you must have the determination to take control of the design and impose your own goals onto the topic rather than allowing yourself to be swept away by the momentum of the topic.

Selecting a good topic can be time-consuming, for each potential topic must be carefully examined for its ability to successfully realize the goals of the game. Many topics carry with them some excess emotional baggage that may interfere with the goals of the game. For example, my most recent game design effort uses the Arthurian legends as its topic. My goal in the game is to examine the nature of leadership. I found the Arthurian legends to be a compelling vehicle for this goal. Unfortunately these legends contain a strong component of male braggadocio, the vanquishing of opponents by brute force. This theme directly contradicts some of the points I want to make with the game, thus weakening the utility of this topic for my ends. I find the legends so powerful and so malleable that I am willing to accept and work around this potential pitfall.

RESEARCH AND PREPARATION

With a goal and topic firmly in mind, the next step is to immerse yourself in the topic. Read everything you can on the topic. Study all previous efforts related to either your goal or your topic. What aspects of these earlier efforts appeal to you? What aspects disappoint or anger you? Make sure that you understand the mechanics of the environment your game will attempt to represent. Your game must give the authentic feel, the texture of the real world, and this can only be achieved if you firmly understand the environment of the game. While researching EXCALIBUR, I studied the history of Britain during the period AD 400-700. I found little in the history books that was harmonious with my goal of depicting the nature of leadership. But in the Arthurian legends I found recurring themes more closely related to my goal. You may well find yourself adjusting your goals as you perform this research function; such erratic decision-making is an embarrassing admission of poorly-defined goals, but reflects an honest willingness to adapt to the exigencies of the topic-environment. It is a departure from the ideal in which I have sinfully indulged myself many times.

During this phase it is critical that you commit little to paper and above all, WRITE NO CODE! Take long walks as you contemplate your game. Cogitate. Meditate. Let the goal, the topic, and the facts gleaned from your research simmer together in the innards of your mind. Weave them together into a whole. Take your time with this phase; impatience now will lead to mistakes that will kill the game. I give myself at least three weeks to develop a game idea in this stage before proceeding to the next step. With EXCALIBUR I expended several months on this stage. During this time I kept my fidgeting hands busy by writing an opening graphic display that had little relevance to the final game.

You will generate during this phase a great variety of specific implementation ideas for your game. They will not all fit together neatly---like any hodgepodge, they will require much sorting and rearranging before they can be used. You should not wed yourself to any of them. A large collection of candidates for implementation is a useful resource during the design phase. A laundry list of implementation ideas that must be included is a liability. Indulge yourself in creating implementation ideas, but be prepared to winnow them ruthlessly during design.

For example, I recently designed a corporate politics game in association with another person. During the research and preparation phase, we came up with a long list of clever ideas that we wanted to into the game. We had agreed that the game would have a feminist point of view without being preachy. We wanted to have a demanding boss, tough projects, deadlines, brownie points, one male chauvinist pig, neutral males, neutral females, family and home obligations, mentors, and the competition for the big promotion. We managed to include almost all of these ideas in the final design. We were not able to integrate the family elements into the game. Every design we created failed to do justice to our desires. In the end, we had to discard this desirable element.

DESIGN PHASE

You now have a clear idea of the game's ideals but you know nothing of its form. You are now ready to begin the concrete design phase. Your primary goal in the design phase is to create the outlines of three interdependent structures: the I/O structure, the game structure, and the program structure. The I/O structure is the system that communicates information between the computer and the player. The game structure is the internal architecture of causal relationships that define the obstacles the player must overcome in the course of the game. The program structure is the organization of mainline code, subroutines, interrupts, and data that make up the entire program. All three structures must be created simultaneously, for they must work in concert. Decisions primarily relating to one structure must be checked for their impacts on the other structures..

I/O Structure

I prefer to start with the I/O structure, for it is the most constraining of the three. I/O is the language of communication between the computer and the player; like any human language, it is the funnel through which we must squeeze the avalanche of thoughts, ideas, and feelings that we seek to share with our fellow human beings. I/O will dictate what can and cannot be done with the gains.

I/O is composed of input and output. Unlike human languages, the two are not symmetric. The computer has two means of output to the human: graphics on the screen and sound. In the future, we may see more exotic devices for output for games, but for the moment these are the two most common. Graphics are the most important of the two, perhaps because we humans are more oriented towards vision than hearing. For this reason, many game designers devote a large portion of their energy towards the design of quality displays. Indeed, some designers go so far as to design the display first and let the game develop from the display, as extreme an example of goalless design as ever there could be.

Don't make the common mistake of creating cute graphics solely to show off your ability to create cute graphics. Graphics are there for a reason: to communicate. Use graphics to communicate to the user forcefully and with feeling, and for no other reason. Plan functional, meaningful graphics that convey the critical game information while supporting the fantasy of the game. Don't use graphics tricks as a crutch for a bad game design. If the game is dull and boring, no amount of graphics gift-wrapping is going to fix it. The worst examples of this mistake are the games that alternate boring game segments with cute but meaningless graphics displays. Use of sound should follow the same rules: use it to tell the player what's going on in the game. The only place where striking but uninformative graphics and sound can be useful is at the beginning of the game, and then only if they help to establish the mood or tone of the game.

Storyboards are a graphics design tool that tempt many game designers, for they are a well-developed technology from the film industry. They are not appropriate to games, because storyboards are an intrinsically sequential technology. Games are not sequential, they are branching tree structures. The game designer who uses an intrinsically sequential tool risks having her designs made subtly sequential. The tool shapes the mind of its user; the saw suggests that we cut wood, and the freeway suggests that we drive wherever it takes us, not where we choose to go. In like manner does a storyboard impress its sequentiality upon our games.

Devote special care to the input structure of the game. The input structure is the player's tactile contact with the game; people attach deep significance to touch, so touch must be a rewarding experience for them. Have you ever noticed the tremendous importance programmers attach to the feel of a keyboard? Remember that players will do the same thing with your game. A case in point is provided by the games JAWBREAKER and MOUSKATTACK (trademarks of On-Line Systems). In both games the joystick entry routine admits an unfortunate ambiguity when a diagonal move is entered. This gives the player the impression that the joystick is unresponsive. I have seen players slam down the joystick in frustration and swear that they would never play the damn thing again. Remember this well as you plan your input structure: will your input structure frustrate and anger your players?

The input structure lies at the heart of a fundamental dilemma all game designers must face. An excellent game allows the player to interact heavily with his opponent, to invest a great deal of his personality into the game. This requires that the game offer the player a large number of meaningful options, enough options that the player can express the nuances of his personality through the choices he makes. Yet, decisions must be inputted, and a large number of options seem to require an extensive and complicated input structure, which could well be intimidating to the player. Our dilemma, then, is that an excellent game seems to require a hulking input structure.

The dilemma is resolved through the designer's creativity in designing a clean input structure that allows many options. This does not come easily. Many schemes must be considered and rejected before a satisfactory solution is found. Yet, such a solution is often possible. In designing SCRAM, a nuclear power plant game, I faced the following problem: how can a player control an entire nuclear power plant with only a joystick? At first glance, the task seems hopeless. Nevertheless, the solution I eventually discovered works very well. The player moves a cursor through the plant display. With the cursor adjacent to a piece of controllable equipment, the player presses the joystick button and pushes the stick up to turn on or increase power, and down to turn off or decrease power. The system is simple and easily understood once the player has seen it.

There is a general solution, at the theoretical level, to the dilemma of option richness versus input cleanliness; I call this solution "the webwork". To design a webwork game, we start with a small number of pieces. We then define a relationship that applies to all pairs of pieces. The set of relationships between pieces constitutes a webwork. The webwork can easily become quite complex, yet few pieces are required to create the webwork. In general, the number of pairwise relationships is equal to N*(N-1), where N is the number of pieces. Thus, four pieces can generate 12 pairings, 8 pieces can generate 56 pairings, and 16 pieces can generate 240 pairings. With fewer pieces to manipulate the player faces fewer I/O problems without sacrificing a rich set of relationships in the game.

Backgammon illustrates the simplicity and power of webwork games. Backgammon has only 30 pieces and 26 positions for them to occupy. The relationships between pieces are fairly simple and are expressed through the ability to move and bump. Yet, on any given move, each piece has an offensive, defensive, blocking, or blocked relationship with most of the other pieces on the board. This is partly because almost every other board position in front of the piece can be reached, given the right die roll. It is no accident that the length of the playing area (24 steps) is exactly equal to the maximum die roll. It had to be that way to squeeze all of the pieces into range of each other, thereby maximizing the number of significant pairwise relationships.

Most webwork games rely on spatially expressed webworks; these are easy to depict and easy for the player to visualize. Few games have non-spatial webworks; my own GOSSIP is one such game. Curiously, GOSSIP uses a spatial webwork for its internal computations even though the game webwork is non-spatial. This may imply that game webworks are intrinsically spatial; it may equally well imply that I cannot shake my mind-set free from spatial webworks.

The choice of input device is an important design decision. I maintain that a good game designer should eschew the use of the keyboard for input and restrict herself to a single simple device, such as a joystick, paddle, or mouse. The value of these devices does not arise from any direct superiority over the keyboard, but rather in the discipline they impose on the designer. Simple input devices go hand-in-hand with simple input structures. Complex input devices encourage complex input structures.

The I/O structure is the most important of the three structures in a computer game, for it is the face of the game that the player sees. It is the vehicle of interaction for the game. It is also the most difficult of the three structures to design, demanding both human sensitivity and complete technical mastery of the computer. Give it the care it deserves.

Game Structure

The central problem in designing the game structure is figuring out how to distill the fantasy of the goal and topic into a workable system. The game designer must identify some key element from the topic environment and build the game around that key element. This key element must be central to the topic, representative or symbolic of the issues addressed in the game, manipulable, and understandable. For example, in EASTERN FRONT 1941, I started with the enormous complexity of modern warfare and extracted a key element: movement. Movement dictates the dispositions of the military units. Moving into an enemy's position initiates combat with him. Moving behind him disrupts his supplies and blocks his retreat routs. Moving into a city captures it. Movement is not equitable with all aspects of war; it is, instead, the key element through which many other aspects of war are expressible. It is easily manipulable and immediately understandable.

A more difficult design challenge came from the game GOSSIP. This game addresses social relationships. The enormous complexity of the subject matter and the intricate twists and turns of human interaction together suggest that the subject is beyond treatment in a game. After much thought I was able to isolate a key element: the "statement of affinity". One way or another, many of our social interactions boil down to one of two declarations: a first-person statement of feeling ("I rather like Sandra"), and a third-person statement ("Well, Tom told me that he doesn't like Sandra one bit"). The key element encapsulates the grander array of human interactions rather well. It is easily manipulable; indeed, it is quantifiable. And it is quite understandable. The isolation of the statement of affinity as the key element of human interaction made possible the game GOSSIP.

The nature of manipulability assumes tremendous importance to the success of the game. The key element must be manipulable, but in a very specific set of ways. It must be expressively manipulable; that is, it must allow the player to express himself, to do the things that he wants or needs to do to experience the fantasy of the game. For example, in a combat game, shooting is almost always a key element. If the player's freedom to shoot is heavily restricted, the player cannot live the fantasy. At the same time, the manipulability must be concise. To use the combat game example again, if the player is required to declare the amount of gunpowder to be expended on each shot, he may well find the manipulability a hindrance to the game. The manipulability must be meaningful to the fantasies of the game. Finally, the manipulability must be focused: the options from which the player chooses while manipulating the key element must be closely related. For example, in the game GOSSIP, the key element (statement of affinity) assumes a linear sequence of values ranging from hatred through love. ENERGY CZAR violates this principle by requiring the player to choose from a large, disconnected set of options. Menu structures and use of the keyboard both arise from unfocussed key elements.

Many games employ multiple key elements. For example, most combat games include both movement and shooting. This is not necessarily bad; if both key elements are kept simple, or if one key element retains primacy, the game can be successful. However, too many key elements violating too many of these principles will rob the game of its focus.

Your main problem with creating the I/O structure is overcoming constraints; your main problem with creating the game structure is realizing possibilities. Your previous work with the I/O structure defines the limitations on the structure of the game. You can take more liberties with the internal structure because the player will not directly encounter it. For example, for the game TACTICS I developed a very complex combat algorithm that realistically calculates the effects of armor-piercing shot. The complexity of this algorithm would have confused the player had I tried to explain it. But the player does not need to understand the internal workings of the algorithm; he need only grasp its effects. I therefore did not feel constrained to design a simple-minded and intuitively obvious algorithm.

Concentrate an providing enough color to guarantee that the game will convey the authentic feel of reality. Keep your sense of proportion while adding details. It will do your game no good to provide exquisite detail and accuracy in one sphere while overlooking the most fundamental elements in another sphere.

A very common mistake many designers make is to pile too many game features onto the game structure. In so doing, they create an overly intricate game, a dirty game. As I discussed in Chapter 4, dirt is undesirable; a game is a structure that must fit together cleanly and well, not a brushpile. Dirt creates a second problem not mentioned in Chapter 4: it gums up the I/O structure of the game. For example, the long-range scan feature of STAR RAIDERS does provide some nice additional capabilities, but it adds another keystroke to be memorized by the player. That's dirty input. Fortunately this problem is overridden in STAR RAIDERS, because the fantasy puts the player at the controls of a starship, and so the player finds the intricacy of the control layout a supporting element of the fantasy rather than a hindrance. In most games, you may well be forced to give up nice elements in the game structure in order to maintain the quality of the I/O structure. On the other hand, you may be forced to go back and change the I/O structure to incorporate a game feature you are unwilling to abandon. If you do so, do not simply tack on a now command; rethink the entire I/O structure and modify it so that the new command fits well with the rest of the I/O structure.

Designing the game structure is emotionally very different from designing the I/O structure. While designing the I/O structure, the designer must thread a precarious path between the Scylla of expressive power and the Charybdis of expressive clarity, even while the storms of hardware limitations toss her design to and fro. While designing the game structure, the designer finds herself on a placid sea stretching flat to the horizon. The challenge taunting her now is "Where do you go?"

Program Structure

The program structure is the third object of your design attentions. This structure is the vehicle which translates the I/O structure and game structure into a real product. One of the most important elements of the program structure is the memory map. You must allocate chunks of memory for specific tasks. Without such safeguards, you may end up expending excessive quantities of memory on minor functions, and having insufficient memory remaining for important tasks. Definitions of critical variables and subroutines are also necessary. Finally, some documentation on program flow is important. Use flow charts or Warnier-Orr diagrams or whatever suits your fancy. This book is not primarily concerned with programming; if you need guidance on program development, consult any of the many excellent books on program development.

Evaluation of the Design

You now have three structures in hand: the I/O structure, the game structure, and the program structure. You are satisfied that all three structures will work and that they are compatible with each other. The next stop in the design phase is to evaluate the overall design for the most common design flaws that plague games. The first and most important question is: does this design satisfy my design goals? Does it do what I want it to do? Will the player really experience what I want him to experience? If you are satisfied that the design does pass this crucial test, proceed to the next test.

Examine the stability of the game structure. Remember that a game is a dynamic process. Are there any circumstances in which the game could get out of control? For example, if the game has money in it, could a situation arise in which the player finds himself the owner of ridiculously large amounts of money? In short, does the game structure guarantee reasonable upper and lower bounds on all values? If not, re-examine the game structure carefully with an eye to structural changes that will right the situation. If you have no other options, you may be obliged to put them in by brute force (e.g., "IF MONEY > 10000 THEN MONEY 10000")

Now probe the design for unanticipated shortcuts to victory. A player who can find a way to guarantee victory with little effort on his part will not derive the full benefit of your game. Insure that all unintended shortcuts are blocked so that the player must experience those processes that you want him to experience. Any blocks you place must be unobtrusive and reasonable. The player must never notice that he is being shepherded down the primrose path. An example of obtrusive blocking comes from the game WAR IN THE EAST (trademark of Simulations Publications, Inc). This wargame deals with the Eastern Front in World War 11. The Germans blitzed deep into Russia but their advance ground to a halt before Moscow. To simulate this the designers gave the Germans an overwhelming superiority but also gave them a supply noose whose length was carefully calculated to insure that the Germans would be jerked to a dead halt just outside Moscow. The effect was correct, but the means of achieving it were too obvious, too obtrusive.

The last and most crucial decision is the decision to abort the game or proceed. It should be made now, before you commit to programming the game. Do not hesitate to abort the game now; even if you abort now you will still have I earned a great deal and can say that the effort was worthwhile. A decision to give up at a later stage will entail a real loss, so give this option careful consideration now while you can still do it without major loss. Abort if the game no longer excites you. Abort if you have doubts about its likelihood of success. Abort if you are unsure that you can successfully implement it. I have in my files nearly a hundred game ideas; of these, I have explored at length some 30 to 40. Of these, all but eight were aborted in the design stage.

PRE-PROGRAMMING PHASE

If the game has made it this far, you are now ready to commit your ideas to paper. Until now your documentation has been sketchy, more along the lines of notes and doodles than documents. Now you are ready to prepare your complete game documentation. First, commit all of your design results from the previous phase to paper. Define the I/O structure and the internal game structure. The tone of this documentation should emphasize the player's experience rather than technical considerations. Compare this first set of documents with your preliminary program structure notes; adjust the program structure documents if necessary.

PROGRAMMING PHASE

This is the easiest of all the phases. Programming itself is straightforward and tedious work, requiring attention to detail more than anything else. Seldom has a game failed solely because the programmer lacked the requisite programming skills. Games have failed to live up to their potential because the programmer did not expend enough effort, or rushed the job, or didn't bother to write in assembly language, but in few cases has talent or lack of it been the crucial factor in the programming of a game; rather, effort or lack of it is most often the responsible factor. If you place all of your self-respect eggs in the programming basket, I suggest that you get out of game design and work in systems programming. Otherwise, write the code and debug it.

PLAYTESTING PHASE

Ideally, playtesting is a process that yields information used to polish and refine the game design. In practice, playtesting often reveals fundamental design and programming problems that require major efforts to correct. Thus, playtesting is often interwoven with a certain amount of program debugging.

Sometimes playtesting reveals that the game is too seriously flawed to save. A nonfatal, correctable flaw is usually a matter of insufficiency or excess: not enough color, too many pieces, not enough action, too much computation required of the player. A fatal flaw arises from a fundamental conflict between two important elements of the game whose incompatibility was not foreseen.

You must have the courage to trash a game with such a fatal flaw. Patching after the game is programmed can only achieve limited gains; if the game is badly deformed, abortion is preferable to surgery.

If playtesting reveals serious but not fatal problems, you must very carefully weigh your options. Do not succumb to the temptation to fall back on a quick and dirty patch job. Many times the problem that is discovered in playtesting is really only a symptom of a more fundamental design flaw. Be analytical; determine the essence of the problem. Once you have determined the true nature of the problem, take plenty of time to devise a variety of solutions. Don't rush this process; sometimes the ideal solution comes from an unexpected angle. Choose a solution for its promise of furthering the faithfulness of the game to your goals. Do not opt for the easiest solution, but the solution that best meets your goals.

For example, while designing EASTERN FRONT 1941, I ran into a severe problem with unit counts: there were far too many units for the player to control conveniently. After wasting much time trying to devise ways to shrink the map or directly reduce the number of units, I eventually stumbled upon zones of control, a standard wargaming technique that extends the effective size of a unit. The inclusion of zones of control in the game not only solved the unit count problem; it also made the logistics rules more significant and gave the game a richer set of strategies. I set out with the narrow goal of reducing the unit count, but I found an improvement with much broader implications.

If your initial design was well-developed (or you are just plain lucky) the game will not face such crises; instead, the problems you will face will be problems of polish. All of the little things that make a game go will be out of tune, and the game will move like a drunken dinosaur instead of the lithe leopard you had envisioned. Tuning the game will take many weeks of work. For the short term you can scrimp on the tuning while you are working on other problems, for tuning the game requires delicate adjustments of all the game factors; any other changes will only throw off the tune. Therefore, defer final tuning work until the very end of the polishing stage.

There are actually two forms of playtesting. The first is your own playtesting done in the final stages of debugging. The second form comes later when you turn over the game to other playtesters. The salient-difference between the two lies in the nature of the bugs exposed. Your own playtesting should reveal and eliminate all program bugs (arising from flaws in the program structure) and many of the game bugs (arising from flaws in the game structure). The game you give to the playtesters should be free of program bugs; they should discover only bugs in the game structure. There is no point in showing an incomplete game to playtesters, and indeed there is a danger in contaminating their objectivity by showing them a version of the game too early. But the time will come when you feel that the game is very close to completion, and your own stock of ideas for improvements is dwindling. This is the time to show the game to a few select playtesters.

Playtesters must be selected and used with great care. You cannot simply grab a few friends and ask them what they think of the game. You need playtesters who possess a deep familiarity with games, playtesters who can analyze and criticize your game with some basis of experience. Ideally the playtesters would themselves be game designers, for they would then share your appreciation for the trade-offs essential to good game design. You should also know the player well, both his personality and his game taste. You should never use more than five or six playtesters. A surplus of playtesters only insures that you will not be able to assess carefully the reaction of each playtester.

A variety of other systems have been used for playtesting. Most rely on gathering large groups of "real people" and assessing their reactions to the game. I have little respect for such systems. Although they are scientific, objective, and democratic, they seldom yield useful design information, for consumers make lousy critics. The suggestions they make are inane and impractical; they don't know enough about computers or games to make practical suggestions. Such methods may well work with detergent and shaving cream, but I very much doubt that any great movie, book, or song was created through market research of this kind. I will concede that such methods can prove to be a useful way to guide the mass production of cheap games by designers of limited talents; this book is not directed to persons of such a mentality. The playtesters will need a preliminary manual for the game. It need not be a finished product any more than the game itself---just enough orientation information to get the playtester going with the game. Make sure that there is enough in the manual that the playtester doesn't waste time critiquing problems of the game that will be solved by the manual. Do not sit down with the playtester in advance and coach him through the game; you will only contaminate his objectivity. The playtester's first reaction to the game is your best feedback on the success of the manual . Let the playtester experiment with the game for perhaps a week before you meet with him. Do not ask the playtester to keep lengthy written records of play performance; he won't do it. Instead, include in the manual a few suggestions about potential problems that worry you. The most for which you should ask in writing is a simple record of game options selected and subsequent scores.

Schedule along interview with the playtester after he has had enough time to digest the game. Come to the interview prepared with a set of standard questions that you ask all playtesters. Do not lead the playtester's answers and don't solicit praise. Your job is to find flaws; accolades come later. While it is more scientific to use a third person to conduct the interview (thereby assuring more honest answers), this imposes a middleman between you and your playtesters. I prefer to get the information directly from the playtester. I also prefer to take a very negative tack during the interview, encouraging the playtester to criticize the game along with me and to suggest means of improving it.

Playtesters' criticisms are difficult to evaluate. Most criticisms must be rejected for a variety of reasons. Some are incompatible with your goals; some are not achievable in the-memory space remaining. Some are reasonable, but would require major software surgery incommensurate with the gains offered. Do not hesitate to reject 90% of the suggestions made. The remaining 10% are right; waste no time implementing them. How do you tell the good 10%? This is the stuff of wisdom; I certainly don't know.

The final stage of the design cycle is devoted to polishing the game. The polishing stage is actually concurrent with the later stages of playtesting and may involve several iterations with the playtesters. This stage is critical; the designer has been working on the game for a long time by now and the luster of the new design has worn off. It is now only a big job that should have been finished months ago. The playtesters love it, the publisher loves it and wants it right now, and the designer is sick of it. The urge to dump the damn thing is overpowering. Resist this urge; press on relentlessly and polish, polish, polish. Keep testing the game, fine-tuning it, and adding tiny embellishments to it. Once it's out the door, it's gone forever. Every single game I have done has followed the same pattern: I polished the game until I was sick of it and never wanted to see it again. When at last I sent the game out, I rejoiced; I was free of that dog at last. Within a month I was regretting my impatience and wishing I could have a chance to clean up that one embarrassing bug that I had never noticed. Within three months my regret had turned into shame as I discovered or was told of many more bugs. I have programs out there whose patrimony I hope never becomes widely known.

One of the last tasks you must perform before releasing the game is the preparation of a game manual. Manuals are frequently given short shrift by just about everybody associated with computer games. This is a serious mistake, for the manual is a vital element in the overall game package. A computer has many limitations; some can be overcome with a good manual. Much of the static information associated with a game can be presented in a manual. The manual is also an excellent place to add fantasy support elements such as pictures and background stories. Finally, a well-written manual will clear up many of the misunderstandings that often arise during a game.

You must write your own manual for the game, no matter how poor a writer you are, and even if a professional writer will prepare the final manual. The attempt to write your own manual will increase your respect for the skills of the professional writer, making it more likely that you will have a productive relationship with the writer. Writing your own manual will also provide feedback on the cleanliness of the game design. Clumsy designs are hard to describe, while clean designs are easier to describe. Finally, your own manual will be a useful source document for the professional writer. You should be prepared for the writer to throw out your manual and start all over---a good writer would rather create a new manual than polish an amateur's crude efforts. You must cater to the writer's needs, answering all his questions as completely as possible. Only a close and supportive relationship between designer and writer can produce an excellent manual.

POST-MORTEM

Once the program is out, brace yourself for the critics. They will get their filthy hands on your lovely game and do the most terrible things to it. They will play it without reading the rules.

If it’s a strategic game, they will castigate it for being insufficiently exciting; if it’s an S&A game, they will find it intellectually deficient. They will divine imaginary technical flaws and speculate incorrectly on your deep psychological hang-ups that led you to produce such a game. One critic of mine concluded that TANKTICS was obviously slapped together on a rush schedule; actually, the time between first efforts and final publication was five years and two months. Another roasted ENERGY CZAR (an energy economics educational simulation) because it wasn’t as exciting as his favorite arcade game. Don’t let these critics affect you. Most critics are far less qualified to criticize programs than you are to write them. A very few critics with the larger publications are quite thoughtful; you should pay attention to their comments. With most critics, though, you should pay heed only to views shared by three or more independent critics. Remember also that even a good critic will roast you if your goal is not to his taste.

The public is another matter. If they don’t buy your game, you lose two ways: first, you or your employer make little money on the game; and second, you don’t reach as many people with your message. It doesn’t matter how beautiful your message is-if nobody listens to it, you have failed as an artist. One failure is nothing to worry about; every artist bombs occasionally. Two failures in a row are bad; three should initiate a serious reconsideration of artistic values. Are you willing to be a noble and starving artist, or a somewhat wealthier artisan? Look within your heart, long and hard. If deep down inside you know that you met your goals, then ignore the critics and the public.

CHAPTER SIX
Design Techniques and Ideals

Every artist develops her own special techniques and ideals for the execution of her art. The painter worries about brush strokes, mixing of paint, and texture; the musical composer learns techniques of orchestration, timing, and counterpoint. The game designer also acquires a variety of specialized skills, techniques, and ideals for the execution of her craft. In this chapter I will describe some of the techniques that I use.

BALANCING SOLITAIRE GAMES

A solitaire game pits the human player against the computer. The computer and the human are very different creatures; where human thought processes are diffuse, associative, and integrated, the machine's thought processes are direct, linear, and arithmetic. This creates a problem. A computer game is created for the benefit of the human, and therefore is cast in the intellectual territory of the human, not that of the computer. This puts the computer at a natural disadvantage. Although the computer could easily whip the human in games involving computation, sorting, or similar functions, such games would be of little interest to the human player. The computer must play on the human's home turf, something it does with great difficulty. How do we design the game to challenge the human? Four techniques are available: vast resources, artificial smarts, limited information, and pace.

Vast Resources

This is by far the most heavily used technique for balancing a game. The computer is provided with immense resources that it uses stupidly. These resources may consist of large numbers of opponents that operate with a rudimentary intelligence. Many games use this ploy: SPACE INVADERS, MISSILE COMMAND, ASTEROIDS, CENTIPEDE, and TEMPEST are some of the more popular games to use this technique. It is also possible to equip the computer with a small number of opponents that are themselves more powerful than the human player's units, such as the supertanks in BATTLEZONE. The effect in both cases is the same: the human player's advantage in intelligence is offset by the computer's material advantages.

This approach has two benefits. First, it gives the conflict between the human and the computer a David versus Goliath air. Most people would rather win as apparent underdog than as equal. Second, this approach is the easiest to implement. Providing artificial intelligence for the computer's players can be difficult, but repeating a process for many computer players takes little more than a simple loop. Of course, the ease of implementing this solution carries a disadvantage: everybody else does it. We are knee-deep in such games! Laziness and lack of determination have far more to do with the prevalence of this technique than game design considerations.

Artificial Smarts

The obvious alternative to the use of sheer numbers is to provide the computer player with intelligence adequate to meet the human on equal terms. Unfortunately, artificial intelligence techniques are not well enough developed to be useful here. Tree-searching techniques have been developed far enough to allow us to produce passable chess, checkers, and Othello players. Any other game that can be expressed in direct tree-searching terms can be handled with these techniques. Unfortunately, very few games are appropriate for this treatment.

An alternative is to develop ad-hoc artificial intelligence routines for each game. Since such routines are too primitive to be referred to as "artificial intelligence", I instead use the less grandiose term "artificial smarts". This is the method I have used in TANKTICS, EASTERN FRONT 1941, and LEGIONNAIRE, with varying degrees of success. This strategy demands great effort from the game designer, for such ad-hoc routines must be reasonable yet unpredictable.

Our first requirement of any artificial smarts system is that it produce reasonable behavior. The computer should not drive its tanks over cliffs, crash spaceships into each other, or pause to rest directly in front of the human's guns. In other words, obviously stupid moves must not be allowed by any artificial smarts system. This requirement tempts us to list all possible stupid moves and write code that tests for each such stupid move and precludes it. This is the wrong way to handle the problem, for the computer can demonstrate unanticipated creativity in the stupidity of its mistakes. A better (but more difficult) method is to create a more general algorithm that obviates most absurd moves.

A second requirement of an artificial smarts routine is unpredictability. The human should never be able to second-guess the behavior of the computer, for this would shatter the illusion of intelligence and make victory much easier. This is may seem to contradict the first requirement of reasonable behavior, for reasonable behavior follows patterns that should be predictable. The apparent contradiction can be resolved through a deeper understanding of the nature of interaction in a game. Three realizations must be combined to arrive at this deeper understanding. First, reaction to an opponent is in some ways a reflection of that opponent. A reasonable player tries to anticipate his opponent's moves by assessing his opponent's personality. Second, interactiveness is a mutual reaction---both players attempt to anticipate each other's moves. Third, this interactiveness is itself a measure of "gaminess". We can combine these three realizations in an analogy. A game becomes analogous to two mirrors aligned towards each other, with each player looking out from one mirror. A puzzle is analogous to the two mirrors being unreflective; the player sees a static, unresponsive image. A weakly interactive game is analogous to the two mirrors being weakly reflective; each player can see and interact at one or two levels of reflection. A perfectly interactive game (the "gamiest game") is analogous to the two mirrors being perfectly reflective; each of the two players recursively exchanges places in an endless tunnel of reflected anticipation's. No matter how reasonable the behavior, the infinitely complex pattern of anticipation and counter-anticipation defies prediction. It is reasonable yet unpredictable.

Unfortunately, a perfectly interactive game is beyond the reach of microcomputers, for if the computer is to anticipate human moves interactively, it must be able to assess the personality of its opponents---a hopeless task as yet. For the moment, we must rely on more primitive guidelines. For example, my experience has been that algorithms are most predictable when they are "particular". By "particular" I mean that they place an emphasis on single elements of the overall game pattern. For example, in wargames, algorithms along the lines of "determine the closest enemy unit and fire at it" are particular and yield predictable behavior.

I have found that the best algorithms consider the greatest amount of information in the broadest context. That is, they will factor into their decision-making the largest number of considerations rather than focus on a small number of particular elements. To continue with the example above, a better algorithm might be "determine the enemy unit posing the greatest combination of threat and vulnerability (based on range, activity, facing, range to other friendly units, cover, and sighting); fire on unit if probability of kill exceeds probability of being killed".

How does one implement such principles into specific algorithms? I doubt that any all purpose system. can ever be found. The best general solution I have found so far for this problem utilizes point systems, field analysis, and changes in the game structure.

First, I establish a point system for quantifying the merit of each possible move. This is a time-honored technique for many artificial intelligence systems. A great deal of thought must go into the point system. The first problem with it is one of dynamic range: the designer must insure that the probability of two accessible moves each accumulating a point value equal to the maximum value allowed by the word size (eight bits) approaches zero. In other words, we can't have two moves each getting a score of 255 or we have no way of knowing which is truly the better move. This problem will diminish as 16-bit systems become more common.

A second problem with the point system is the balancing of factors against each other. In our hypothetical tank game used above, we agree that climbing on top of a hill is good, but we also agree that moving onto a road is good. Which is better? If a hilltop position is worth 15 points, what is a road position worth? These questions are very difficult to answer. They require a deep familiarity with the play of the game. Unfortunately, such familiarity is impossible to attain with a game that has yet to be completed. The only alternative is broad experience, intimate knowledge of the situation being represented, painstaking analysis, and lots of experimenting.

A second element of my general approach to artificial smarts is the use of field analysis. This is only applicable to games involving spatial relationships. In such games the human relies on pattern recognition to analyze positions and plan moves. True pattern recognition on the level of human effort is beyond the abilities of a microcomputer. However, something approaching pattern recognition can be attained through the use of field analysis. The key effort here is the creation of a calculable field quantity that correctly expresses the critical information needed by the computer to make a reasonable move. For example, in several of my wargames I have made use of safety and danger fields that tell a unit how much safety or danger it faces. Danger is calculated by summing the quotients of enemy units' strengths divided by their ranges; thus, large close units are very dangerous and small distant units are only slightly dangerous. A similar calculation with friendly units yields a safety factor. By comparing the danger value at its position with the safety value at its position, a unit can decide whether it should exhibit bold behavior or timid behavior. Once this decision is made, the unit can look around it and measure the net danger minus safety in each position into which the unit could move. If it is feeling bold, it moves towards the danger; if it is feeling timid, it moves away. Thus, the use of fields allows a unit to assess a spatial array of factors.

Another technique for coping with artificial smarts problems is so simple that it seems like cheating: change the game. If an element of the game is not tractable with artificial reckoning, remove it. If you can't come up with a good way to use a feature, you really have no choice but to delete it. For example, while designing TANKTICS, I encountered a problem with lakes. If a lake was concave in shape, the computer would drive its tanks to the shore, back up, and return to the shore. The concave lake created a trap for my artificial smarts algorithm. I wasted a great deal of time working on a smarter artificial smarts routine that would not be trapped by concave lakes while retaining desirable economies of motion. After much wasted effort I discovered the better solution: delete concave lakes from the map.

Ideally, the experienced game designer has enough intuitive feel for algorithms that she can sense game factors that are intractable and avoid them during the design stages of the game. Most of us must discover these things the hard way and retrace our steps to modify the design. Experiencing these disasters is part of what provides the intuition.

A special problem is the coordination of moves of many different units under the control of the computer. How is the computer to assure that the different units move in a coordinated way and that traffic jams don't develop? One way is to use a sequential planning system coupled with a simple test for the position of other units. Thus, unit #1 moves first, then #2, then #3, with each one avoiding collisions. I can assure you from my own experience that this system replaces collisions with the most frustrating traffic jams. A better way uses a virtual move system in which each unit plans a virtual move on the basis of the virtual positions of all units. Here's how it works: we begin with an array of real positions of all computer units. We create an array of virtual positions and initialize all virtual values to the real values. Then each unit plans its move, avoiding collisions with the virtual positions. When its move is planned, it places its planned final position into the virtual array. Other units then plan their moves. After all units have planned one virtual move, the process repeats, with each unit planning its move on the basis of the interim virtual move array. This huge outer loop should be convergent; after a sufficient number of iterations the routine terminates and the virtual positions form the basis of the moves made by the computer's units. This technique should be useful for coordinating the moves of many units and preventing traffic jams.

No matter how good an algorithm is, it has a limited regime of applicability. The odds are that a specific algorithm will work best under a narrow range of conditions. A good game design must offer a broad range of conditions to be truly interesting. Thus, the designer must frequently create a number of algorithms and switch from one to another as conditions change. The transition from one algorithm to another is fraught with peril, for continuity must be maintained across the transition. I well remember a frustrating experience with algorithm transitions with LEGIONNAIRE. The computer-barbarians had three algorithms: a "run for safety" algorithm, an "approach to contact" algorithm, and an "attack" algorithm. Under certain conditions a barbarian operating under the "approach to contact" algorithm would decide on bold behavior, dash forward to make contact with the human, and make the transition to the "attack" algorithm, which would then declare an attack unsafe. The barbarian would thus balk at the attack, and convert to the "run for safety" algorithm, which would direct it to turn tail and run. The human player was treated to a spectacle of ferociously charging and frantically retreating barbarians, none of whom ever bothered to actually fight. I eventually gave up and re-designed the algorithms, merging them into a single "advance to attack" algorithm with no transitions.

The artificial smarts techniques I have described so far are designed for use in games involving spatial relationships. Many games are non-spatial; other artificial smarts techniques are required for such games. One of the most common types of non-spatial games uses systems that behave in complex ways. These games often use coupled differential equations to model complex systems. LUNAR LANDER, HAMMURABI, ENERGY CZAR, and SCRAM are all examples of such games. The primary problem facing the designer of such games is not so much to defeat the human as to model complex behavior. I advise the game designer to be particularly careful with games involving large systems of coupled differential equations. HAMMURABI uses three coupled first-order differential equations, and most programmers find it tractable. But the complexity of the problem rises very steeply with the number of differential equations used. ENERGY CZAR used the fantastic sum of 48 differential equations, a feat made believable only by the fact that many constraints were imposed on them. In general, be wary of more than four coupled differential equations. If you must use many differential equations, try to use parallel differential equations, in which the same fundamental equation is applied to each element of an array of values.

To help keep the system balanced, each differential equation should have a damping factor that must be empirically adjusted:

new value = old value + (driving factor / damping factor)

A small damping factor produces lively systems that bounce around wildly. A large damping factor yields sluggish systems that change slowly. Unfortunately, recourse to simple damping factors can backfire when a relationship of negative feedback exists between the "new value" and the "driving force". In this case, large damping inhibits the negative feedback, and one of the variables goes wild. The behavior of systems of differential equations is complex; I suggest that designers interested in these problems study the mathematics of overdamped, underdamped, and critically damped oscillatory systems. For more general information on solving systems of differential equations, any good textbook on numerical analysis will serve as a useful guide.

Conclusions on Artificial Smarts

The application of all of these methods may well produce a game with some intelligence, but one's expectations should not be too high. Even the expenditure of great effort is not enough to produce truly intelligent play; none of my three efforts to date play with an intelligence that is adequate, by itself, to tackle a human player. Indeed, they still need force ratios of at least two to one to stand up to the human player.

Limited Information

Another way to make up for the computer's lack of intelligence is to limit the amount of information available to the human player. If the human does not have the information to process, he cannot apply his superior processing power to the problem. This technique should not be applied to excess, for then the game is reduced to a game of chance. It can, nevertheless, equalize the odds. If the information is withheld in a reasonable context (e.g., the player must send out scouts), the restrictions on information seem natural.

Limited information provides a bonus: it can tickle the imagination of the player by suggesting without actually confirming. This only happens when the limitations on the information are artfully chosen. Randomly assigned gaps in information are confusing and frustrating rather than tantalizing. A naked woman can be beautiful to the male eye, but an artfully dressed woman can conceal her charms suggestively and thus appear even more alluring. The same woman randomly covered with miscellaneous bits of cloth would only look silly.

Another way to even balance between human and computer is through the pace of the game. The human may be smart, but the computer is much faster at performing simple computations. If the pace is fast enough, the human will not have enough time to apply his superior processing skills, and will be befuddled. This is a very easy technique to apply, so it comes as no surprise that it is very heavily used by designers of skill and action games.

I do not encourage the use of pace as an equalizing agent in computer games. Pace only succeeds by depriving the human player of the time he needs to invest a larger portion of himself into the game. Without that investment, the game can never offer a rich challenge. Pace does for computer games what the one-night stand does for romance. Like one-night stands, it will never go away. We certainly do not need to encourage it.

Summary

These four techniques for balancing computer games are never used in isolation; every game uses some combination of the four. Most games rely primarily on pace and quantity for balance, with very little intelligence or limited information. There is no reason why a game could not use all four techniques; indeed, this should make the game all the more successful, for, by using small amounts of each method, the game would not have to strain the limitations of each. The designer must decide the appropriate balance of each for the goals of the particular game.

RELATIONSHIPS BETWEEN OPPONENTS

Every game establishes a relationship between opponents that each player strives to exploit to maximum advantage. The fundamental architecture of this relationship plays a central role in the game. It defines the interactions available to the players and sets the tone of the game. Most computer games to date utilize very simple player-to-player relationships; this has limited their range and depth. A deeper understanding of player-to-player relationships will lead to more interesting games.

Symmetric Relationships

The simplest architecture establishes a symmetric relationship between the two players. Both possess the same properties, the same strengths and weaknesses. Symmetric games have the obviously desirable feature that they are automatically balanced. They tend to be much easier to program because the same processes are applied to each player. Finally, they are easier to learn and understand. Examples of symmetric games include COMBAT for the ATARI 2600, BASKETBALL, and DOG DAZE by Gray Chang.

Symmetric games suffer from a variety of weaknesses, the greatest of which is their relative simplicity. Any strategy that promises to be truly effective can and will be used by both sides simultaneously. In such a case, success is derived not from planning but from execution. Alternatively, success in the game turns on very fine details; chess provides an example an advantage of but a single pawn can be parlayed into a victory.

Asymmetric games

Because of the weaknesses of symmetric games, many games attempt to establish an asymmetric relationship between the opponents. Each player has a unique combination of advantages and disadvantages. The game designer must somehow balance the advantages so that both sides have the same likelihood of victory, given equal levels of skill. The simplest way of doing this is with plastic asymmetry. These games are formally symmetric, but the players are allowed to select initial traits according to some set of restrictions. For example, in the Avalon-Hill boardgame WIZARD'S QUEST, the players are each allowed the same number of territories at the beginning of the game, but they choose their territories in sequence. Thus, what was initially a symmetric relationship (each person has N territories) becomes an asymmetric one (player A has one combination of N territories while player B has a different combination). The asymmetry is provided by the players themselves at the outset of the game, so if the results are unbalanced, the player has no one to blame but himself.

Other games establish a more explicitly asymmetric relationship. Almost all solitaire computer games establish an asymmetric relationship between the computer player and the human player because the computer cannot hope to compete with the human in matters of intelligence. Thus, the human player is given resources that allow him to bring his superior planning power to bear, and the computer gets resources that compensate for its lack of intelligence.

Triangularity

The advantage of asymmetric games lies in the ability to build nontransitive or triangular relationships into the game. Transitivity is a well-defined mathematical property. In the context of games it is best illustrated with the rock-scissors-paper game. Two players play this game; each secretly selects one of the three pieces; they simultaneously announce and compare their choices. If both made the same choice the result is a draw and the game is repeated. If they make different choices, then rock breaks scissors, scissors cut paper, and paper enfolds rock. This relationship, in which each component can defeat one other and can be defeated by one other, is a nontransitive relationship; the fact that rock beats scissors and scissors beat paper does not mean that rock beats paper. Notice that this particular nontransitive relationship only produces clean results with three components. This is because each component only relates to two other components; it beats one and loses to the other. A rock-scissors-paper game with binary outcomes (win or lose) cannot be made with more than three components. One could be made with multiple components if several levels of victory (using a point system, perhaps) were admitted.

Nontransitivity is an interesting mathematical property, but it does not yield rich games so long as we hew to the strict mathematical meaning of the term. The value of this discussion lies in the generalization of the principle into less well-defined areas. I use the term "triangular" to describe such asymmetric relationships that extend the concepts of nontransitivity beyond its formal definition.

A simple example of a triangular relationship appears in the game BATTLEZONE. When a saucer appears, the player can pursue the saucer instead of an enemy tank. In such a case, there are three components: player, saucer, and enemy tank. The player pursues the saucer (side one) and allows the enemy tank to pursue him unmolested (side two). The third side of the triangle (saucer to enemy tank) is not directly meaningful to the human---the computer maneuvers the saucer to entice the human into a poor position. This example is easy to understand because the triangularity assumes a spatial form as well as a structural one.

Triangularity is most often implemented with mixed offensive-defensive relationships. In most conflict games, regardless of the medium of conflict, there will be offensive actions and defensive ones. Some games concentrate the bulk of one activity on one side, making one side the attacker and the other side the defender. This is a risky business, for it restricts the options available to each player. It’s hard to interact when your options are limited. Much more entertaining are games that mix offensive and defensive strategies for each player. This way, each player gets to attack and to defend. What is more important, players can trade off defensive needs against offensive opportunities. Triangular relationships automatically spring from such situations.

The essence of the value of triangularity lies in its indirection. A binary relationship makes direct conflict unavoidable; the antagonists must approach and attack each other through direct means. These direct approaches are obvious and expected; for this reason such games often degenerate into tedious exercises following a narrow script. A triangular relationship allows each player indirect methods of approach. Such an indirect approach always allows a far richer and subtler interaction.

Actors and Indirect Relationships

Indirection is the essence of the value of triangularity to game design. Indirection is itself an important element to consider, for triangularity is only the most rudimentary expression of indirection. We can take the concept of indirection further than triangularity. Most games provide a direct relationship between opponents, as shown in the following diagram:

Since the opponent is the only obstacle facing the player, the simplest and most obvious resolution of the conflict is to destroy the opponent. This is why so many of these direct games are so violent. Triangularity, on the other hand, provides some indirection in the relationship:

With triangularity, each opponent can get at the other through the third party. The third party can be a passive agent, a weakly active one, or a full-fledged player. However, it's tough enough getting two people together for a game, much less three; therefore the third agent is often played by a computer-generated actor. An actor, as defined here, is not the same as an opponent. An actor follows a simple script; it has no guiding intelligence or purpose of its own. For example, the saucer in BATTLEZONE is an actor. Its script calls for it to drift around the battlefield without actively participating in the battle. Its function is distraction, a very weak role for an actor to play.

The actor concept allows us to understand a higher level of indirection, diagrammatically represented as follows:

In this arrangement, the players do not battle each other directly; they control actors who engage in direct conflict. A good example of this scheme is shown in the game ROBOTWAR by Muse Software. In this game, each player controls a killer robot. The player writes a detailed script (a short program) for his robot; this script will be used by the robot in a gladiatorial contest. The game thus removes the players from direct conflict and substitutes robot-actors as combatants. Each player is clearly identified with his own robot. This form of indirection is unsuccessful because the conflict itself remains direct; moreover, the player is removed from the conflict and forced to sit on the sidelines. I therefore see this form of indirection as an unsuccessful transitional stage.

The next level of indirection is shown in a very clever boardgame design by Jim Dunnigan, BATTLE FOR GERMANY. This game concerns the invasion of Germany in 1945. This was obviously an uneven struggle, for the Germans were simultaneously fighting the Russians in the east and the Anglo-Americans in the west. Uneven struggles make frustrating games. Dunnigan's solution was to split both sides. One player controls the Russians and the west-front Germans; the other controls the Anglo-Americans and the east-front Germans. Thus, each player is both invader and defender: Neither player identifies directly with the invaders or the Germans; the two combatants have lost their identities and are now actors.

The highest expression of indirection I have seen is Dunnigan's RUSSIAN CIVIL WAR game. This boardgame covers the civil war between the Reds and the Whites. Dunnigan's brilliant approach was to completely dissolve any identification between player and combatant. Each player receives some Red armies and some White armies. During the course of the game, the player uses his Red armies to attack and destroy other players' White armies. He uses his White armies to attack and destroy other players' Red armies. The end of the game comes when one side, Red or White, is annihilated. The winner is then the player most identifiable with the victorious army (i.e., with the largest pile of loser's bodies and the smallest pile of winner's bodies).

The indirection of this game is truly impressive. The two combatants are in no way identifiable with any individual until very late in the game. They are actors; Red and White battle without human manifestation even though they are played by human players. There is only one limitation to this design: the system requires more than two players to work effectively. Nevertheless, such highly indirect player-to-player architectures provide many fascinating opportunities for game design. Direct player-to-player relationships can only be applied to direct conflicts such as war. Direct conflicts tend to be violent and destructive; for this reason, society discourages direct conflicts. Yet conflict remains in our lives, taking more subtle and indirect forms. We fight our real-world battles with smiles, distant allies, pressure, and co-operation. Games with direct player-to-player relationships cannot hope to address real human interaction. Only indirect games offer any possibility of designing games that successfully explore the human condition.

SMOOTH LEARNING CURVES

As a player works with a game, s/he should show steady and smooth improvement. Beginners should be able to make some progress, intermediate people should get intermediate scores, and experienced players should got high scores. If we were to make a graph of a typical player's score as a function of time spent with the game, that graph should show a curve sloping smoothly and steadily upward. This is the most desirable case.

A variety of other learning curves can arise; they reveal a great deal about the game. If a game has a curve that is relatively flat, we say that the game is hard to learn. If the curve is steep, we say the game is easy to learn. If the curve has a sharp jump in it, we say that there is just one trick to the game, mastery of which guarantees complete mastery of the game. If the game has many sharp jumps, we say that there are many tricks. A particularly bad case arises when the player's score falls or levels off midway through the learning experience. This indicates that the game contains contradictory elements that confuse or distract the player at a certain level of proficiency. The ideal always slopes upward smoothly and steadily.

Games without smooth learning curves frustrate players by failing to provide them with reasonable opportunities for bettering their scores. Players feel that the game is either too hard, too easy, or simply arbitrary. Games with smooth learning curves challenge their players at all levels and encourage continued play by offering the prospect of new discoveries.

A smooth learning curve is worked into a game by providing a smooth progression from the beginner's level to an expert level. This requires that the game designer create not one game but a series of related games. Each game must be intrinsically interesting and challenging to the level of player for which it is targeted. Ideally, the progression is automatic; the player starts at the beginner's level and the advanced features are brought in as the computer recognizes proficient play. More commonly, the player must declare the level at which he desires to play.

THE ILLUSION OF WINNABILITY

Another important trait of any game is the illusion of winnability. If a game is to provide a continuing challenge to the player, it must also provide a continuing motivation to play. It must appear to be winnable to all players, the beginner and the expert. Yet, it must never be truly winnable or it will lose its appeal. This illusion is very difficult to maintain. Some games maintain it for the expert but never achieve it for the beginner; these games intimidate all but the most determined players. TEMPEST, for example, intimidates many players because it appears to be unwinnable. The most successful game in this respect is PAC-MAN, which appears winnable to most players, yet is never quite winnable.

The most important factor in the creation of the illusion of winnability is the cleanliness of the game. A dirty game intimidates its beginners with an excess of details. The beginner never overcomes the inhibiting suspicion that somewhere in the game lurks a "gotcha". By contrast, a clean game encourages all players to experiment with the game as it appears.

Another key factor in maintaining the illusion of winnability arises from a careful analysis of the source of player failure. In every game the player is expected to fail often. What trips up the player? If the player believes that his failure arises from some flaw in the game or its controls, he becomes frustrated and angry with what he rightly judges to be an unfair and unwinnable situation. If the player believes that his failure arises from his own limitations, but judges that the game expects or requires superhuman performance, the player again rejects the game as unfair and unwinnable. But if the player believes failures to be attributable to correctable errors on his own part, he believes the game to be winnable and plays on in an effort to master the game. When the player falls, he should slap himself gently and say, "That was a silly mistake!"

SUMMARY

In this chapter I have described a number of design methods and ideals that I have used in developing several games. Methods and ideals should not be used in grab bag fashion, for taken together they constitute the elusive element we call "technique". Technique is part of an artist's signature, as important as theme. When we listen to Beethoven's majestic Fifth Symphony, or the rapturous Sixth, or the ecstatic Ninth, we recognize in all the identifying stamp of Beethoven's masterful technique. If you would be a compute game designer, you must establish and develop your own technique.

CHAPTER SEVEN
The Future of Computer Games

In this book, I have explored computer games from a number of angles. I have presented my claim that computer games constitute an as-yet untapped art form. Implicit in this claim is the hope that this art form will someday be tapped. Unfortunately, history bears out the fears of cynics more often than the hopes of dreamers. I must therefore separate hopes from predictions. Where are computer games going? How will they change in the years to come? Will we see them emerge as a true art form? There are a number of divergent trends apparent now; analysis of them is complicated by conflicting interpretations of the current state of computer game design. I shall begin by addressing the most commonly cited arguments, and proceed to the framework I prefer.

FAD OR FIXTURE?

The first and most important question concerns the very survival of the computer games industry. One school of thought maintains that computer games are merely a fad, a temporary infatuation that will quickly pass when their novelty value is exhausted. Proponents of this view compare the computer game to other fads that swept into society with equal force. They maintain that computer games lack sufficient fundamental appeal to insure any staying power. Eventually, these people say, computer games will go the way of the hula hoop.

This line of thought is breezily rejected by all members of the industry, but I fear that the confidence people express is little more than the Titanic syndrome---the confidence that arises from mere size. They tend to blindly extrapolate into the future the astounding growth rates we have experienced in the past. It is certainly hard to give credence to doomsayers when the curve of growth slopes upward so steeply. However, few industry optimists can provide justification for their extrapolations. Just because the industry doubled in 1982 does not mean that it will double in 1983 or 1984. Indeed, it cannot continue to annually double much longer; if it did, only eleven years' time would be needed for Atari alone to engulf the entire Gross National Product like some monstrous PAC-MAN.

Furthermore, size alone generates negative forces that will certainly reduce the growth rate. In the simple days of the seventies, when computer games were counted by the thousands rather than the millions, nobody much cared about their effects because they were a minor component of our society. But now, they are everywhere. They are such a powerful force that they are affecting society in such a way as to generate negative feedback. We now have a backlash developing against computer games, with ordinances against arcades popping up all over the country. Parents are beginning to restrict their children's access to the games. Editorialists warn against the dire effects of playing the games. Already several preliminary studies have been undertaken to determine the effects of computer games on children; so far, the as-yet speculative results have been mildly favorable, but the day will certainly come when the crap game we call research comes up snakeyes, and a blockbuster report is issued demonstrating that computer games cause cancer in laboratory rats.

Bigger critters than Atari have bitten the dust; bigger industries than ours have shriveled and died. Size and past success are no guarantee of permanence. We need substantive reasons for confidence in the future rather than simple extrapolations of past history. I am convinced that substantive reasons for optimism exist; the full presentation of my reasoning will come later in this chapter. For now let me say that computer games satisfy a fundamental desire for active recreation, and as such are assured of a bright future.

THE TECHNOLOGICAL EXTRAPOLATION

The most commonly cited future for computer games is the technological extrapolation. Adherents of this school point to the undeniably steady march of technology and the rapid improvements that we have seen in the hardware for delivering games. They then extrapolate these trends directly to project a future populated by supercomputers with fabulous games chock-full of unbelievable graphics and incredibly realistic experiences. These people emphasize technological factors as the primary agents of change. They claim that the big breakthroughs will come with the use of bigger and faster processors, megabytes of RAM, new languages, and better display hardware. Holography, trackballs, laserdisks, body sensors-these are the coin of the realm among the technological extrapolators.

I cast a jaded eye on such predictions. This is the same line of thought that extrapolated computer development in the late 60’s to predict ever-larger, ever-faster mainframes as the primary avenues of development in the computer industry for the 70’s. Computers did indeed become larger in that decade, but the development of larger computers was not the dominant event of the 70’s. Instead, the maturation of minicomputers and the genesis of microcomputers were the major developments of the 70’s. The extrapolators never foresaw the coming of microcomputers, because micros didn’t fit into their "bigger and better" extrapolations.

I do not deny that technology will improve; it will. The real issue is not whether or not technology will improve, but whether or not technological limitations are the primary constraints on the game designer. I do not deny that technological limitations do impose severe constraints on all computer games, and I readily acknowledge that technological advances will remove many of these constraints. Thus, technological immaturity, the weakness of current 8-bit, 64K, 1 MHz systems---is a crippling limitation. Yet I maintain that artistic immaturity is an even more crippling limitation.

Consider two extreme hypothetical future worlds. The first world has no technological development and the second world has no artistic development. In the first world I am stuck with an Atari 800 as my sole medium for game design. This does not worry me too much; I could explore the possibilities of this machine for five or ten years before beginning to feel trapped. The second world, though, is a bleak place indeed; I am doomed to write ever-fancier variations on STAR RAIDERS and BREAKOUT, with more colorful explosions, snazzier sounds, and 3-D photon torpedoes, but never anything new or different. I would feel trapped immediately.

Neither of these worlds will happen; we will have both technological development and artistic development. Yet, we must remember that the technological development, while entirely desirable, will never be the driving force, the engine of change for computer games. Artistic maturation will be the dynamo that drives the computer games industry.

The relative importance of technological development and artistic maturity is made clear by a comparison of modern movies with the silent movies. The modern movies boast gigantic technological advantages---sound, color, and fabulous special effects. When used with skill and artistry, the new technologies are indeed magnificent. Yet, all these advantages cannot make up for a lack of artistic quality: the computer-graphics blockbuster TRON compares poorly with any of Charlie Chaplin's movies. if Chaplin could do so much with black and white film and no sound, why cannot we do good work with 8 bits and 48K?

ASSESSMENT: TECHNOLOGICAL REVOLUTION

To explain my own assessment, I must present some background about how I view technological revolutions. The first great technological revolution I will draw on is the revolution in transportation that swept American society in the first half of the twentieth century. The automobile was invented in the late 1800's; by the turn of the century it was available as a consumer product. However, many problems plagued the automobile. It was expensive and unreliable. It lacked the software (support services such as service stations and appropriate roads) to make it truly practical. It required considerable skill and dedication to operate. Furthermore, it was unnecessary; American culture had developed quite successfully without it, so there was little existing need for it. Thus, the automobile was not a practical tool; it was a plaything of the wealthy.

With the passage of time, these problems with the automobile lessened in severity. Mass production lowered the cost and increased the reliability; more service stations and better roads became available. More and more automobiles were purchased; by the late twenties the automobile was a common fixture of American life.

The third stage became obvious in the 1950's. The automobile changed the face of American society. Housing patterns began to change. Commuting became practical. Urban sprawl sprawl. Drive-in restaurants and theaters became common. The technology changed the society.

The fourth stage began asserting itself at about the same time. As the automobile changed American society, so too did society change the automobile. Originally designed as a device to transport people and property from point A to point B as quickly, safely, and reliably as possible, it was transformed into a form of self-expression, a recreational device, and ultimately an end in itself. Could Henry Ford have anticipated dune buggies, vans with waterbeds, low-riders, and naked-lady hood ornaments? I doubt it.

Let me summarize the four stages that occurred in this transportation revolution. First, the technology was initially desirable to only a small part of the public. With time, conditions improved and the technology conquered society. Then it began to change society. In the process, society began to change the technology. The direction of this change was away from the pragmatic and towards the recreational.

Let us now examine the second great revolution of this century, the entertainment revolution sparked by the television. When television first became available in the late 1940's, it was expensive, unreliable, and lacking sufficient software (programs) to make it anything more than a toy for the wealthy. With time, these problems were overcome. Televisions became cheaper, more reliable, and offered more programming. They swept into society with great force. In the process, they dramatically changed the lifestyles of the American people. Nighttime entertainment was now readily available. Leisure time activities changed accordingly. But the public worked its will on television. It evolved from "visible radio", or a means of presenting lectures, plays, and speeches, into a medium with its own personality. Thus, the same four stages outlined for the automobile occurred with television: pioneer, conquest, transformation of society by the technology, and transformation of the technology by society.

The same sequence of stages is occurring with computers. At the moment, personal computers are still expensive, unreliable, hard to use, and lacking software. The situation is changing rapidly; prices are failing, machines are becoming friendlier, and software availability improves daily. All observers agree that personal computers will take society by storm. The only differences of opinion are those of magnitude. Will 1990 see 5 million computers in American homes, or 10 million, or 20 million? No one knows, but everyone agrees that the figure will be large.

We therefore expect that personal computers will change the face of American society. We expect that networking will allow more Americans to participate in economic activities from the home, decreasing the load on transportation and accelerating the pace of economic life. The ease of manipulating information will give information an even more prominent role in our society. Our financial system will become less dependent on currency. Our lives will be changed by these machines.

But we ourselves will not be changed. The computer will change our habits and our leisure time, but it will not change our personalities, for emotionally we are still the same people who built the pyramids, fought the Crusades, and colonized the New World. Our analysis of the two previous revolutions leads us to expect that the relationship between society and the computer will be one of reciprocal transformation. We further expect that the nature of this transformation will be a shift from the pragmatic toward the recreational, from the functional to the frivolous. This leads us to suspect games as the primary vehicle for society to work its will on computers.

Ten years ago, even five years ago, this suggestion would have seemed ridiculous. Computers were the awesome creatures of man's cleverness, the intelligent progeny of the machine age. They were perceived to be powerful, endlessly capable, and not a little fearsome. Most people's only concern with computers was whether they would be man's slave or his master. The possibility that they might be his playmate never crossed anyone's mind.

We were wrong, for the computer game has already established itself as a primary form of use of the computer. By any number of measures, computer games are already a major portion of the world of computers. Consider, for example, the number of computer games in existence. What is the most reproduced program in human history, the one program with more copies in existence than any other program in the world? At the moment, the #1 program is undoubtedly COMBAT, the game cartridge supplied with every ATARI 2600. Millions and millions of copies of this cartridge have been distributed. Perhaps you object that this measure is unfair because nobody buys the program by itself. Very well, then, consider PAC-MAN, ASTEROIDS, SPACE INVADERS, and MISSILE COMMAND, each of which has sold millions of copies. Indeed, were we to compile a "Top Forty" lit of the best-selling programs of all time, I very much doubt that Visicalc (trademark of Visicorp) or any serious piece of software would make the list. Games dominate. Perhaps you object that numbers alone do not adequately measure social significance. Perhaps you would prefer to measure economic significance. Very well, let's try a comparison. Visicalc, the most successful personal computer serious package, has sold, say, 400,000 copies at, say, $200 apiece. That amounts to $80 million gross. By contrast, if Atari sells, say, 5 million copies of PAC-MAN at $30 apiece, that's $150 million. And that's just one title; there are many other games generating large sales figures.

Thus, games are already a primary form of use of computer technology. They have established themselves as a major component in the world of computers. In the accelerated world of the 80's, the fourth stage (transformation of technology by society) is upon us even as the second phase (conquest) is beginning.

THE NATURE OF CHANGE

Games are the vehicle with which society will change the computer. How will the games themselves be changed by society? We can expect two processes to affect games: the mass market and the flowering of heterogeneity. In some ways, these processes work against each other.

The Mass Market

As computer games become a mass market item, they will fall prey to the homogenizing forces of the mass market. The emphasis will not be on originality or creativity, but rather on adhering to the time-honored formulas. Just as movies and television fell prey to the formulas of sex and violence, cops and robbers, sitcoms, and the other mechanical incantations of the mass media, so too will games fall victim to the tyranny of the mass market. (Are my biases showing?) We will see an emphasis on delivering the same game over and over in new clothing. My guess is that we are already caught in the grip of this force, for we are producing little more than variations on a single theme: "blast the monsters!". This has sold well, so we stick with it.

This cynical view of the mass market is countered by the realization that the mass market is occasionally capable of sustaining a real blockbuster. Hollywood may grind out an army of soulless clones, but every now and then something really interesting comes out. When this happens, the mass market responds fabulously. 2001-A SPACE ODYSSEY, STAR WARS, and RAIDERS OF THE LOST ARK are examples of original, creative ideas coming out for the mass market and enjoying success. Just because something works in the mass market does not mean that it must be junk.

The Flowering of Heterogeneity

The games market differs from the movie market and the television market in that it is less centralized and has fewer economies of scale. In this respect it is closer to the books market and the records market. For this reason, I expect the games market to exhibit a greater degree of heterogeneity and less slavish obeisance to mass tastes.

I therefore expect a host of baby markets following in the train of the mass market. While the baby markets will never be as lucrative as the mass market, they perform two valuable services. First, they provide a testing ground for new ideas that, if successful, will be swallowed up by the voracious mass market. Beyond, the baby markets will always provide a haven for the refugees from mediocrity and a playground for those whose tastes aren’t average.

You may ask why baby markets have not yet developed very far to date. I answer the question with a little story. Suppose that you were the first astronaut to land on a newly discovered planet, and there you found a civilization every bit the equal of ours, but for a single exception: they had no literature. No novels, no poetry, no children’s books, no textbooks, no magazines, nothing that we have, with one exception: they did have comic books. On further study, you discovered the reason for this oddity. Reading was a new discovery only recently popularized by teenagers and shunned by the majority of adults who felt intimidated by this newfangled skill. Thus, literature was used by teenagers to express the fantasies and interests they enjoyed: confronting authority, violent resolution of conflict and so forth. Hence comic books. Could you not look on this situation and recognize the seeds of the future in it? Would not the flowering of other forms of literature be expected as the kids grow up and develop new interests? Would not novels, short stories, westerns, gothic romances, poetry, and other genres be incipient in the situation you found?

So it is with computer games. Until now the preserve of teenage males, these games are bursting into society at large. While they have satisfied until now the fantasies of twisted computer-nerd minds, they will soon blossom into a much richer array of fantasies. We will have country-western games, gothic romance games, soap-opera games, comedy games, X-rated games, wargames, accountant games, and snob games. The society that invented the hot tub, CB radio, and dune buggies will have no reservations about impressing its character on computer games.

Eventually, games will be recognized as a serious art form. The exploration of games as a serious art form will be restricted to a tiny fraction of the total activity. Most of the effort will always be more along the lines of pop-art. Yet this tiny group of games-artists will be responsible for creating the future classics of games, the games that endure.

CONCLUSIONS

To conclude: I see a future in which computer games are a major recreational activity. I see a mass market of computer games not too different from what we now have, complete with blockbuster games, spin-off games, remake games, and tired complaints that computer games constitute a vast wasteland. I even have a term for such games---cyberschlock. I also see a much more exciting literature of computer games, reaching into almost all spheres of human fantasy. Collectively, these baby market games will probably be more important as a social force than the homogenized clones of the mass market, but individual games in this arena will never have the economic success of the big time games.

By 1985 software stores will be as common as record stores; by 1990 they will be as common as bookstores. On entering the software store, you will be confronted by racks and racks of games, with serious software occupying a smaller portion of the floorspace. Just as in a bookstore or record store, you will see aisles devoted to particular tastes in games. You can browse through collections of cowboy games as your companion explores the latest space games. Perhaps you will look for the latest product of your favorite author, all of whose works are collected in alphabetical order. On the walls you will see posters announcing the latest smash hit games by software superstars. After evaluating a number of games you will make your choices and purchase them. Then you'll go out to the parking lot to discover that some idiot has dented the fender of your car. Some things never change.

Chapter Eight
The Development of Excalibur

In Chapter 5, I presented an idealized game design sequence. I attempted to describe a general purpose method that properly recognized the concepts developed throughout this book. It is a sad truth that the practicality of the schemes we devise is inversely proportional to the idealism they embody. I have never designed a game in complete accordance with the system described in Chapter 5. My real designs have followed considerably rockier courses. In this chapter, I will describe the development of EXCALIBUR, a recent design. The contrast between the real process, jerky and mistake-prone, and the ideal process should help the reader bridge the gap between theory and practice.

BEGINNINGS

In December of 1981, I began working for Alan Kay in his new Corporate Research unit at Atari. Given total creative freedom, I resolved to do a game worthy of the vast faith that Dr. Kay had invested in me. I wanted this game to be grand and glorious, a game so lofty in its goals and play that it would put all others to shame. Since marketing considerations were not significant to the game, I resolved that this game would run in a 48K disk-based environment. This afforded me plenty of computer resource with which to work.

My background is in wargames, and I naturally thought in terms of a wargame. War is the most extreme expression of human conflict, the greatest evil of human existence, and the highest tragedy of our species; it is therefore an obvious starting point for a serious artist. I wanted to break away from the conventional treatment of war in wargames, which either glorifies war as an expression of misconceived heroism, or trivializes war as a fascinating intellectual exercise. I wanted something more than a wargame, something that placed war in a meaningful context. My game would include war as a viable option that must sometimes be exercised, but not frivolously. I wanted a game that warmongers would inevitably lose, because I deeply believe that peaceful strategies are often the most practical ones. This game would address statecraft as a human enterprise; as such it would necessarily focus on leadership. Another fundamental goal I established was that the game would actually consist of a number of games linked together. This would allow me to show policy, statecraft, and war at a variety of scales, from the most strategic and indirect level to the most tactical and direct level.

My next task was to determine the fantasy context for the game. I boiled the possibilities down to two contenders: a game dealing with the USA after a major nuclear war, and a game about Britain in the Dark Ages after the collapse of Roman authority. Both contexts deal with societies attempting to reorganize themselves after a calamity. I decided that the first fantasy was too morbid for my purposes. Furthermore, the second fantasy context was shrouded in the legends of King Arthur, an intrinsically interesting subject. I therefore chose the Arthurian context.

The player in this game would be King Arthur, and his goal would be to unify Britain and bring peace to the troubled land. The challenge of the game would arise from the unwillingness of the other kings to submit to Arthur's authority. The player would be required to use a variety of techniques to establish his authority, only one of which would be military action. Indeed, I resolved that overuse of military methods would brutalize the nation and result in endless insurrections and anarchy. With these noble goals established, I began serious design work on the game.

EARLY WORK: JANUARY-APRIL, 1982

I first turned to the question, what is leadership? The answer to this question is central to the game. It was essential for me to determine the essence of leadership at the national level and reduce this essence to a form manageable in a game. I needed to extract the central decisions of leadership and design a form for expressing them. The military aspects of leadership are the most obvious and easiest to work with. I would have had no difficulty designing a game in which the player must make all the correct military decisions. Yet, this was not satisfactory to me: I wanted to address wider issues. My game had to address the social, diplomatic, and interpersonal aspects of leadership. How was I to represent and manipulate these factors in the course of the game? These problems vexed me for months.

I quickly grew impatient with the struggle with such fundamental problems. The child in me wanted immediate gratification. To satiate these impatient impulses, I wrote the title and ending scenes for the game. These were not crucial to the structure of the game, but they. gave me an opportunity to explore some interesting graphics techniques without compromising the integrity of my design. The ending scene posed some interesting problems. It shows the sword Excalibur twirling through the air over a lake, falling into a hand that abruptly rises out of the water to catch it, and then recedes beneath the waves. I spent a great deal of time trying to add the lonely sound of the wind whistling against the blade of the sword, but I was never able to obtain satisfactory results. I therefore turned to the idea of accompanying the title and ending scenes with some appropriate music. I chose as my two prime candidates a section from Siegfried's death and funeral in Wagner's Siegfried, and a portion of Dvorak's Seventh Symphony.

I also determined the fundamental structure of the game at this time. There were to be four fundamental nested games. The first, CAMELOT, would concern Arthur's activities within his castle. These would include the management of his own kingdom, the conduct of diplomacy, and the preparation of the army. The second game module, BRITAIN, would allow Arthur to travel around the island of Britain with his army and engage in strategic military activity. The third game module, BATTLE, would allow Arthur to fight battles with enemy armies. If Arthur himself managed to encounter an enemy king on the battlefield, then he would enter the fourth module, JOUST. This last module was intended to be a simple skill-and-action game in which Arthur attempted to unhorse his opponent. The game would use a full first-person view of an advancing horseman, lance leveled, with the whole scene bouncing up and down with the galloping of Arthur's own horse. I entertained myself by devising clever graphics algorithms that would generate true 3D first-person graphics. After I had expended a great deal of effort, though, I realized that the JOUST game would take only a few seconds to play and would not provide much challenge. So I started over with a new idea: a swordfight game. The first problem I faced was, how can I simulate the motion of a sword through joystick commands? I got out a yardstick and spent hours in my living room, swinging the yardstick, trying to divine some sort of pattern to it that could be represented cleanly with a joystick. My difficulties arose from the fact that the motion of a sword in a swordfight is a very complex motion, and a joystick simply cannot adequately express all the intricacies of such motion. I eventually found a reasonable system. The side-to-side motion of the joystick controlled the angle of attack of the sword, from horizontal swing from the left, through a vertical swing over the player's head, to a horizontal swing from the right. Backward motion on the joystick swung the sword backwards in preparation for a stroke; forward motion of the joystick sent the sword forward in its stroke.

This problem solved, I began work on some new graphics routines that would show an opposing swordsman in first-person graphics. This proved to be a very difficult task. I eventually gave up on the swordfight game for much of the same reasons that had led me to abandon the joust game. Besides, I didn't want Arthur to be able to hack his way to victory. If swordfights cannot assure success, what's the point of having them in the game?

By now it was March. I began work on the BRITAIN module. This was a .scrolling map with a number of embellishments thrown in. I had earlier done .scrolling maps in EASTERN FRONT 1941 and LEGIONNAIRE, so the implementation of this module was easy for me. Since I had lots more memory for this game, I decided to splurge and make a gigantic scrolling map. I ended up with a 6K map of Britain that is quite large.

Slowly the design was taking shape in my head, but a fundamental question remained unanswered: was this to be a historical game or a fictional game? That is, was this a game about Britain in the sixth century AD or was this a game about King Arthur? I read every book I could lay my hands on about both subjects. This research led me to conclude that Britain in the sixth century was a chaotic and depressing place. The native Celts were defending their homeland against invading Anglo-Saxons landing on the eastern coast of the island. For two centuries the Anglo-Saxons slowly pushed the Celts westward. King Arthur was actually a Celtic general who led a brief counteroffensive against the Anglo-Saxons, winning the battle of Mount Badon and halting the Anglo-Saxon offensive for about 50 years. But Arthur's success was only a brief respite; in the end, the Celts lost. Thus, the historical record does not support my needs for a society struggling to reorganize itself. Instead, the story of Britain in the Dark Ages is the story of one people being relentlessly driven out by another.

Yet, from the dreams of the vanquished arose the legend of the conquering King Arthur, a legend that passed through the ages and agreeably molded itself to suit the needs of any storyteller. As I read the many incarnations of these legends, I was struck by their surpassing flexibility. Each artist who took them up impressed a different character upon them. They worked just as well as religious inspiration, ribald tales, or expositions of the chivalric ideal. Even Mark Twain turned them to good use for his characteristic blistering social comment.

A major turning point in the design process came when I watched the movie EXCALIBUR. This is a magnificent film that beautifully captures the best elements of the Arthurian legends yet makes its own statement. I watched it over and over, reveling in the richness of the tale. This movie shamed me with its excellence. I realized that I had been compromising the important artistic issues in my game in order to play with cute graphics. I climbed a lonely hill and spent a day meditating. I rededicated myself to the lofty artistic goals I had earlier set for myself. I also knew that I could not realize them alone; I had to got help. I enlisted the aid of Larry Summers, and hired Valerie Atkinson to help me. With new determination, we set to work.

THE LONG HAUL: MAY-DECEMBER 1982

Here is where we stood in May, 1982: I had established the broad design but had left many details unfinished. A number of disparate chunks of code had been written, but they did not fit together at all. There was no overall design document. Faced with so many things to do, I foolishly opted to finish some of the more obvious minor things. I wrote the CALIG module that draws Gothic characters onto the screen. Valerie set to work preparing the bit map tables for the routine. Larry worked on finishing the title scene by adding the music and the dissolve routines. This work, never intended as more than flashy window-dressing, unfortunately consumed nearly two months.

In June we began work on the CAMELOT module, with Valerie taking primary programming responsibility. This module was actually a set of illustrated menus. Each room (menu) had four options described by a single-word entry. A vertical band allowed the player to move his crown-cursor to the menu selection. To the right of the vertical band we placed a graphic window for showing some critical bit of information. For example, in the Round Table Room, we showed a circle depicting the Round Table and a set of shields representing the knights of the Round Table. Their spatial positions in the room indicated their social relationships. In the Treasury Room we had intended to show piles of coins; we had to delete that feature later on to show more detailed economic data. We had also intended to use a kernelled display that would have allowed much more color on the screen; later on we gave up on that idea, for it would have consumed too much execution time.

As Valerie set to work on this sizable job, I began working on the social game associated with the Round Table. I plunged into the task without realizing the magnitude of what I was attempting. I wanted to produce a small game that would require Arthur to manage a social group. I quickly realized that the most interesting features of such a situation lay not the radial relationships (the relationships between Arthur and the other knights) but in the circumferential relationships among the knights. Although Arthur must perforce deal with knights radially, the circumferential relationships may well be the deciding factors. I found this system fascinating and worked intensively with it. I developed a set of algorithms that model group behavior in a most interesting way. I was so pleased with the algorithms that I threw together a short BASIC program that turned them into a stand-alone game. This game seemed very promising to me; particularly impressive was my wife's reaction. A woman who takes a dim view of silly games, she took an instant liking to this game. Surprised and gratified that I had finally produced something she could enjoy, I decided to pursue this new game, originally a study for EXCALIBUR, as a completely new project. Aric Wilmunder was hired to execute the design, called GOSSIP.

In July we entered a long and slow period of frustrating progress. I began devoting a larger share of my time to the writing of this book. Other duties further distracted me. Without my active daily participation, the project began to flounder. Larry and Valerie plugged away at their work, making the best of a weak situation. For months they slowly built on the system we had created, fleshing out the skeletal system I had so briefly described. Since I had so little time to devote to the project, I did a great deal of designing by the seat of my pants. In our regular weekly meetings, they would present me with the latest design flaw they had uncovered. Having no clear memories of previous decisions, I would hack together an ad hoc solution. My intuitions are fairly good, and many times I got away with these deplorable techniques. However, many of my on-the-fly decisions fell apart and wrought havoc with the overall design. Poor Valerie put features into the CAMELOT module, only to have have them stripped out, then later re-installed.

Our records for this period indicate a great deal of wasted effort. We had intended that the treasury room in Camelot would be illustrated with piles of coins indicating quantities of wealth. A great deal of time was expended writing coin-drawing routines. In the end, we realized that we didn't have enough screen space to show these piles of coins, so we had to use simple numbers drawn onto the screen. Indeed, the list of things we designed, programmed, and later dropped is a revealing measure of my own failure to plan ahead. The list includes declarations of war (dropped but later incarnated as "Attack"), alliances, sieges, demands for tribute, armies moving around in Britain, and a host of minor patches.

Six months were consumed in this muddle. These six months were not a total loss; indeed, much progress was made: Larry completed the economics processing, the BRITAIN module, disk swapping of modules, the presentation of diplomatic news, and a number of major consolidations of the ever-burgeoning code. Valerie took the CAMELOT module much further, linking it to the new features and making it the largest and most complex module in the entire game. Yet, all of this could have been completed in half the time had I been more organized and devoted more energy to the project. By Christmas, everybody was tired of the project, demoralized, and despairing that the project would ever be completed. Those were dark days indeed.

RENEWED EFFORT (JANUARY - APRIL 1983)

In January 1983 I was able to return EXCALIBUR to its rightful place as my highest priority project. I plunged into the project with a cold determination to get this project done and out the door. Gone were the grand and lofty feelings of 1982, the misty-eyed vision of a truly grandiose game.

In their place was a bitter resolve to finish the game at all costs. I met long and frequently with Larry and Valerie. Ruthlessly I slashed at the design, ripping out vaguely defined or non-essential sections. The design discipline that I had sought to escape by consuming vast computer resource was forced on me by my inability to complete the project. At home, I worked on the artificial intelligence routines for the knights in the Round Table Room. This took a few weeks. Then I tackled the BATTLE scene. During February and March I wrote, debugged, and playtested this module. I was possessed, driven to complete the game by my self-imposed deadline of April 1. My records indicate that I averaged 300 bytes of debugged code per day. Industry averages are 75-100 bytes per day. Larry and Valerie were caught up in the frenzy. They worked furiously on integrating all the pieces of the program together and resolving the myriad inconsistencies thereby produced. Entire modules handling Merlin's room, economics, vassalage, tithes, and swapping code were designed, coded, and debugged.

Despite this, we failed to make our April 1 deadline. We moved it back to April 15. Even this became impossible to meet. Nevertheless we made April 15 an important milestone -- all coding would be completed by this date.

The first two weeks of April were consumed in a wild orgy of effort. Meeting every day, sometimes for four hours at a stretch, we hammered out what was undoubtedly the toughest part of the design: the artificial intelligence algorithms.

I had reserved this task for last, for the AI routines must reflect every aspect of the design. The design must therefore be complete, and all variables completely defined, before AI algorithms can be designed. Moreover, the creation of the AI routines tends to freeze the design, since significant design changes after the AI is done can ruin the entire AI design.

The AI for EXCALIBUR is easily the most difficult I have ever attempted. It must consider the personalities of the different kings, economic factors, military factors, and geometric factors. The system we developed uses intermediate variables that express concepts such as the amount of military prestige a king has, how much prestige he has an economic manager, and how well-liked he is. Personality traits factored into the algorithms include ambition, stupidity, and defensiveness.

FINAL WORK (MAY - JUNE 1983)

We almost succeeded in meeting our milestone of having all code completed by April 15. The code remaining was quite trivial. We all took a break for two weeks. In May we began final work on EXCALIBUR. Larry and Valerie began rooting out and eliminating all the bugs in the program. As I write this, they are still working on the task. In June, we will begin tuning and polishing the game. I would like to spend more time polishing this game, but it is long overdue. It will have been in development for 18 months, and will have consumed 3 programmer-years of effort. In these days of six-week development times of quicky games, EXCALIBUR may well be one of the most sweated-over games ever done. It is certainly one of the most ambitious designs ever attempted. It may not be successful, but if it fails, it will not be for want of effort.

Crawford's 1998 note: We shipped EXCALIBUR in July.

Interview with Chris Crawford,

Fifteen Years After Excalibur and The Art of Computer Game Design

By Sue Peabody , Asst. Prof. of History, Washington State University Vancouver

Date: Tue, 17 Jun 1997

Prof. Peabody asks:

I'm interested in what you think of the changes that have occurred in the last decade since you wrote this -- what did you correctly anticipate? What was obscured in your crystal ball? Is there anything that you would like to add to the piece now that you couldn't or didn't when you originally wrote it?

Gee, it's actually been fifteen years since I wrote that in 1982, so I can be even less humiliated by its errors. I will not try to evaluate specific statements, but rather respond to the overall tone. I was pretty much on the mark in guessing the approximate rate of growth of revenues in entertainment software. The industry is indeed much bigger and better-funded than back in the early 80s. Where I was way off the mark was my optimism about the broadening of the marketplace. I believed that by this time we'd be seeing a wide range of entertainment software addressing a wide range of tastes. That has not happened; computer games now are completely unchanged in terms of their basic appeal. They are precisely the same fast-action shoot-em-ups or nerdy strategy games that we were dishing out 15 years ago.

What became of Excaliber? (I gather that it was very successful.)

Indeed not. It came out just as Atari collapsed and was lost in the dust of the disaster. Those few people who saw it, though, seem to have been impressed. I know that most designers regard it as a minor landmark in game design.

Do you think that the computer game lends itself better to certain kinds of history?

Absolutely! And this is both its strength and its weakness. Every form of historical examination has biases built into it. The stuff and substance of history -- documents -- has a built-in bias towards big shots. We know lots of details about Charlemagne, but damn little about the few million peasants who lived under his rule. We know some things about the Bronze Age better than the Iron Age, because bronze doesn't rust away.

Of course, computer games aren't evidence, but they are a prism through which we can look at the evidence, and they bias our view, too. This bias can be a strength, especially when it forces us to take an operational view of history rather than a mythological view. By this I mean that history can be "wondrous stories" or it can be "natural processes." Thus, the mythological style would tell us that Napoleon won so many battles because he was a brilliant strategist -- hooray for Napoleon! But we can also wargame out his battles, follow what he actually did and why he did it, and it makes a great deal more sense. What also emerges from an operational approach is that Napoleon was a lot more ruthless than his opponents in terms of "living off the land" (taking all the peasants' food).

I suspect that the written word is weaker for operational thinking than it is for mythological thinking. Most written descriptions of the Battle of Midway love to tell of that dramatic moment when the Japanese admiral looked up and saw the American dive bombers overhead, and in that one instant, the battle was lost. But how did they get to that juncture? Yes, written accounts do manage to communicate the intricate sequence of events that led to such a profound reversal of fortune, but the written explanations are either impossible to follow or have a mythological feel, as if this battle were some grand Greek drama acted out in the Pacific Ocean. When you actually play out the thing, you get a greater sense of how microscopically logical processes can lead to macroscopically astounding results.

A computer game, like any history, can be used to emphasize some aspect of history. For example, I designed a game some years back that I called Guns & Butter, in which I presented the thesis that technological development arises automatically from economic growth. Most histories of technology have a "great man" flavor to them, so I presented the alternative view that new technologies arise automatically as soon as an economy is large enough to utilize them. (By the way, would this be termed a Marxist view of technological history?) I won't claim that this thesis is necessarily correct, but it certainly offered a different view of historical processes. The tendency of polities to agglomerate at ever-larger levels came through quite clearly in the game.

Obviously, there's plenty of room for abuse here, and the relative opacity of the designer's assumptions and biases (compared with print) could make computer games a greater source of mischief than enlightenment. Goebbels was so frightening because he had a pretty good grip on how to use modern media for propaganda purposes. Right now, we're all too dumb to figure it out. Someday we'll have our interactive Goebbels.

One way to characterize the difference between the "thesis" of a historical game and the "thesis" of a book or article is that the game thesis can be written in present tense (e.g. "the French Revolution resulted from a government fiscal crisis, an economic emergency and a lessening of monarchical authority") whereas a conventional textual thesis is in past tense.

Interestingly, I was just clearing out some old paper wargames from SPI days, and they all sport, across the top of the box, the legend "The time is: 0600 hours, Thursday, May 21st, 1476" or some such. The sense of being in the present is vital to simulation -- and one of its most powerful attractions. Isn't the whole idea of history to make the past accessible to the present?

I'm caught up in the midst of a software deadline... Gotta go now.

Chris

WEB LINKS

THE ART OF COMPUTER GAME DESIGN:

http://www.vancouver.wsu.edu/fac/peabody/game-book/Coverpage.html

Chris Crawford: http://www.erasmatazz.com/
Email: chriscrawford@wave.net

Prof. Sue Peabody, Department of History,
Washington State University Vancouver:

http://www.vancouver.wsu.edu/fac/peabody/peabody.htm
Email: peabody@vancouver.wsu.edu

Mario Croteau: **email**: kalid@sympatico.ca