New Zork Times | *http://www.csd.uwo.ca/Infocom/Articles/NZT/zorkhist.html*

The History of Zork

by Tim Anderson & Stu Galley

First in a Series

In the beginning, back in the 1960's, DEC (Digital Equipment Corporation) created the PDP-10, a medium-sized computer. The "10", as it was called, became popular at many research installations, and a great deal of software was written for it, some of which is still far in advance of systems on more modern machines. At MIT's Artificial Intelligence Lab, an operating system called ITS (Incompatible Time-sharing System) was written for the 10. ITS was designed to make software development easy. The designers of the system assumed that it would have a small, knowledgeable, friendly group of users, so they did not include any security features.

Around 1970, the ARPAnet was invented. ARPAnet made it possible for researchers all over the country (indeed, all over the world) to communicate with each other, and to use each other's machines. In those halcyon days, access was unrestricted; you could get on from any machine connected to the net, or by knowing an appropriate phone number. Budding hackers from around the country soon discovered that this made a wonderful playground. They also discovered that there were some computers at MIT with some neat stuff on them and no security—anyone who could connect to the machines could log in.

Also around 1970, a language called MUDDLE (later renamed MDL) was developed as a successor to LISP. It never succeeded in fully replacing LISP, but it developed a loyal user community of its own, primarily at MIT's Project MAC (now called the Laboratory for Computer Science) and especially in the Dynamic Modelling Group (later the Programming Technology Division). The Dynamic Modelling Group (DM), in addition to its other accomplishments, was responsible for some famous games. The first of these was a multi-player graphics game called Maze, in which players wandered around a maze shooting each other. Each user's screen showed the view of the maze that his or her computerized alter-ego saw, updated in real time. Dave Lebling was among those chiefly responsible (to blame?) for the existence of the game.

The next game of note was Trivia (who says research labs aren't ahead of their time?), an ongoing "can you top this" contest for the truly crazed. Trivia, unlike Maze, could be played by network users, and achieved wide popularity on the ARPAnet. Marc Blank wrote the second version, and I maintained/hacked it; it was actually a legitimate test of a database system the group used for a research project.

In early 1977, Adventure swept the ARPAnet. Willie Crowther was the original author, but Don Woods greatly expanded the game and unleashed it on an unsuspecting network. When Adventure arrived at MIT, the reaction was typical: after everybody spent a lot of time doing nothing but solving the game (it's estimated that Adventure set the entire computer industry back two weeks), the true lunatics began to think about how they could do it better. Adventure was written in FORTRAN, after all, so it couldn't be very smart. It accepted only two-word commands, it was obviously hard to change, and the problems were sometimes not everything one could desire. (I was present when Bruce Daniels, one of the DM'ers, figured out how to get the last point in Adventure by examining the game with a machine-language debugger. There was no other way to do it.)

By late May, Adventure had been solved, and various DM'ers were looking for ways to have fun. Marc Blank was enjoying a respite from medical school; I had just finished my master's degree; Bruce Daniels was getting bored with his Ph.D. topic; and Dave Lebling was heartily sick of Morse code. Dave wrote (in MUDDLE) a command parser that was almost as smart as Adventure's; Marc and I, who were both in the habit of hacking all night, took advantage of this to write a prototype four-room game. It has long since vanished. There was a band, a bandbox, a peanut room (the band was outside the door, playing "Hail to the Chief"), and a "chamber filled with deadlines." Dave played and tested the game, saw that it was pretty awful, and left, to spend two weeks basking in the sun.

Marc, Bruce, and I sat down to write a real game. We began by drawing some maps, inventing some problems, and arguing a lot about how to make things work. Bruce still had some thoughts of graduating, thus preferring design to implementation, so Marc and I spent the rest of Dave's vacation in the terminal room implementing the first version of Zork. Zork, by the way, was never really named. "Zork" was a nonsense word floating around; it was usually a verb, as in "zork the fweep," and may have been derived from "zorch." ("Zorch" is another nonsense word implying total destruction.) We tended to name our programs with the word "zork" until they were ready to be installed on the system.

By the time Dave got back, there was a (more-or-less) working game. It probably wasn't as big as Adventure, and was certainly less than half the size of the final version, but it had the thief, the cyclops, the troll, the reservoir and dam, the house, part of the forest, the glacier, the maze, and a bunch of other stuff. The problems were not as interesting as those that came later: it took time for people to learn how to write good problems, and the early parsers wouldn't support complicated solutions anyway. What we had done right was all in the "substratum." There was a well-defined (and easily-changed) theory governing interactions among objects, verbs, and rooms. It was easy to drop in new parsers, which happened frequently, since everyone and his uncle tried his hand at writing a parser (Marc finally became obsessed with it, and wrote the last 40 or 50 of them himself). And it was easy to add new rooms, objects, and creatures (I won't discuss the difficulty of adding new concepts yet).

Zork, like Adventure, survived only because it was played by people outside the small community that developed it. In the case of Adventure, this was possible because it was written in FORTRAN and could run on practically any machine. Zork was written in MUDDLE, which ran on only some PDP-10s. Its user community was the group of "net randoms" that infested the MIT systems; remember that we had no security at all at this time. DM had developed an active community largely because of Trivia. Since Trivia was pretty dead by the time Zork came along, there weren't many other things for the randoms to do, so they hung around waiting for the next game. Early players of Zork ranged from John McCarthy, the inventor of LISP (we actually have a copy of the connectivity matrix that McCarthy used instead of a map), to twelve-year-olds from Northern Virginia. No one ever officially announced Zork: people would log in to DM, see that someone was running a program named Zork, and get interested. They would then "snoop" on the console of the person running Zork, and see that it was an Adventure-like game. From there, it only took a little more effort to find out how to start it up. For a long time, the magic incantation was ":MARC;ZORK"; people who had never heard of ITS, DM, or PDP-10s somehow heard that if they got to something called "host 70" on the ARPAnet, logged in, and typed the magic word, they could play a game.

Although Zork in June 1977 was infinitely more primitive than, say, Zork I, it still had pretty much the same flavor. The Flathead family was represented, in the person of Lord Dimwit Flathead the Excessive, ruler of the Great Underground Empire; and the official currency was the zorkmid. Bruce was responsible for the purplish prose where these were first mentioned.

Many of the details of the GUE were whimsical (if not silly), but we weren't completely immune to reality. In those days, if one wandered around in the dark area of the dungeon, one fell into a bottomless pit. Many users pointed out that a bottomless pit in an attic should be noticeable from the ground floor of the house. Dave came up with the notion of grues, and he wrote their description. From the beginning (or almost the beginning, anyway), the living room had a copy of "US News & Dungeon Report," describing recent changes in the game. All changes were credited to some group of implementers, but not necessarily to those actually responsible: one of the issues describe Bruce working for weeks to fill in all the bottomless pits in the dungeon, thus forcing packs of grues to roam around.

The first major addition to the game, done in June 1977, was the river section, designed and implemented by Marc. It survives largely unchanged in Zork I, but illustrates very well the problems of building reality. There were minor problems of consistency—some parts of the river were sunlit (and even reachable from outside), but others were dark. The major problem resulted from the new concept Marc introduced: vehicles. In the original game, there were rooms, objects, and a player; the player always existed in some room. Vehicles were objects that became, in effect, mobile rooms. This required changes in the (always delicate) interactions among verbs, objects, and rooms (we had to have some way of making "walk" do something reasonable when the player was in the boat). In addition, ever-resourceful Zorkers tried to use the boat anywhere they thought they could. The code for the boat itself was not designed to function outside the river section, but nothing kept the player from carrying the deflated boat to the reservoir and trying to sail across. Eventually the boat was allowed in the reservoir, but the general problem always remained: anything that changes the world you're modelling changes practically everything in the world you're modelling.

Although Zork was only a month old, it could already surprise its authors. The boat, due to the details of its implementation, turned into a "bag of holding": players could put practically anything into it and carry it around, even if the weight of the contents far exceeded what a player was allowed to carry. The boat was two separate objects: the "inflated boat" object contained the objects, but the player carried the "deflated boat" object around. We knew nothing about this: someone finally reported it to us as a bug. As far as I know, the bug is still there.

Second in a Series

When last seen, Zork(R) was a small game (probably slightly more than half the size of the final mainframe version) that ran on one computer. Although it was only six weeks old, and had never been advertised, it had a relatively large user community from all over the country. In some ways it was better than the classic Adventure at this time, but mostly it was the next game to come along, and it wasn't even the only contender.

The characters: MIT-DM, a PDP-10 running ITS; MDL (aka Muddle), a language that ran only on PDP-10s; Marc Blank, Bruce Daniels, Dave Lebling, and Tim Anderson, intrepid implementers; and assorted net randoms.

July 1977 saw two major additions to the game, the last for several months (we weren't exactly hired to write the thing, after all). The first of these was another BKD special: Bruce didn't write much code, but he was willing to design problems. We went to him, and asked for a particularly nasty section; the result was the coal mine. His design was originally nastier than the final implementation, since the maze was just about as horrible as the original one in the game; it got simplified due to popular demand. The problems were improving in quality, and the coal mine maze was a late example of making things hard by making them tedious.

The volcano section was Marc's second vehicle implementation, but is perhaps more noteworthy for the loving portraits of Lord Dimwit Flathead the Excessive that decorated the coin and stamp found in the section. The river (see Part I) and volcano sections, in addition to vehicles, required a better concept of time: both the boat and the balloon moved more or less on their own, and the volcano required the use of explosives and fuses. Marc added a clock daemon, which processed a queue of events that would happen some fixed number of moves later. This handled, in addition to the movement of the vehicles, the fuse, the lantern burning out, and the mysterious gnomes that occasionally appear. The first of these was in the volcano: if the player got trapped in the upper reaches of the volcano by losing his balloon, after a few moves a volcano gnome would appear and offer freedom in exchange for a treasure. We were just being nice; most players weren't allowed to save their games, so they had no way of backing out if they made such a mistake. The gnome allowed them to keep playing, albeit with no chance of getting all the points.

Even before the volcano section, we'd talked about a problem that involved flying; Dave had a preference for something with an eagle, and its aerie, but we could never figure out how to restrict things enough—it wouldn't do to have a parallel map of the game viewed from the air. Once again, we worried about restraining a new concept, so the balloon had no way of leaving the volcano. And once again we were bitten by a new concept. When the player used the explosives in the wrong place, and didn't get out of the way, he'd end up with 20,000 pounds (or was it tons?) of rock on his head. This made a certain amount of sense in the underground section, but not out in the forest.

No more sections were added to the game for several months after July, but it continued to improve. In addition it finally moved to machines other than DM, thus greatly expanding the number of players.

Although Muddle ran primarily on DM, a version for TENEX (the most popular PDP-10 operating system on the Arpanet) had existed for some time; the TENEX version could, with some minor modifications, run on TOPS-20 as well. We finally succumbed to one of the requests for a copy of Zork when we were given an account on a TOPS-20 machine on the net. After we made the necessary software modifications, of course, many copies could be made; a mailing list of Zork owners developed, so They could get whatever updates appeared.

Although people could get runnable Zorks, they couldn't get sources. We tried two approaches to protecting the sources (remember, there was no protection of any sort on DM): they were normally kept encrypted; and we patched the system to protect the directory where we kept the sources (named CFS, for either "Charles F. Stanley" or "Computer Fantasy and Simulation"). This worked pretty well, but was finally beaten by a system hacker from Digital: using some archaic ITS documentation (there's never been any other kind), he was able to figure out how to modify the running operating system. Being clever, he was also able to figure out how our patch to protect the source directory worked. Then it was just a matter of decrypting the sources, but that was soon reduced to figuring out the key we'd used. Ted had no trouble getting machine time; he just found a new TOPS-20 machine that was undergoing final testing, and started a program that tried every key until it got something that looked like text. After less than a day of crunching, he had a readable copy of the source. We had to concede that anyone who'd go to that much trouble deserved it. This led to some other things later on.

Players hadn't been able to save their Zorks because the method we used at first took several hundred thousand bytes for each save, and even on a time-shared system that was excessive. Marc, around this time, invented a new way of saving that cut the size down to something more reasonable, with the slight disadvantages that any new rooms or objects added to the old game would break existing save files, and that it never quite worked right anyway. However, it did make it easier to play the game, and we still had the silly notion of being nice to our users.

Fall '77 saw two major additions to the game, as Marc took another break from medical school (yes, fans, he did graduate on time), and Dave got into coding in a big way. The Alice in Wonderland section, complete with its magic bucket and robot, was installed. The robot was the first "actor," an object that could perform some of the same tasks the player could. The style of address was familiar: "ROBOT, TAKE THE CAKE." The implementation of this required another change in the game's flow of control, and changes to anything else that one could reasonably talk to.

The first version of fighting was added about the same time. Dave, an old Dungeons and Dragons player, didn't like the completely predictable ways of killing creatures off. In the original game, for example, one killed a troll by throwing a knife at him; he would catch the knife and gleefully eat it (like anything else you threw at him), but hemorrhage as a result. Dave added basically the full complexity of D&D-style fighting, with different strengths for different weapons, wounds, unconsciousness, and death. Each creature had its own set of messages, so a fight with the thief (who uses a stiletto) would be very different from a fight with the troll and his axe.

As a result of the purloined sources at DEC, a lunatic there decided to translate Zork into FORTRAN. We had always assumed this would be impossible: Muddle is very (oops, *very*) different from FORTRAN, and much more complicated, and we'd used most of its features in designing Zork. The guy who did it was mostly a hardware person, so perhaps he didn't know what he was up against. At any rate, shortly after the Great Blizzard of '78 he had a working version, initially for PDP-11s. Since it was in FORTRAN, it could run on practically anything, and by now it has.

Unfortunately, at some point in the preceding year we (no one will now admit to suggesting the idea) had decided to change the name of the game. Zork was too much of a nonsense word, not descriptive of the game, etc., etc., etc. Silly as it sounds, we eventually started calling it Dungeon. (Dave admits to suggesting the new name, but that's only a minor sin.) When Bob the lunatic released his FORTRAN version to the DEC users' group, that was the name he used. I'm sure many people have noticed a curious similarity between the Dungeon game they played on their friendly IBM 4341 and the Zork I they played on their equally friendly IBM PC; now you know why.

Fortunately for us, a certain company (which shall remain nameless) decided to claim that it had trademark rights to the name Dungeon, as a result of certain games that it sold. We didn't agree (and MIT had some very expensive lawyers on retainer who agreed with us), but it encouraged us to do the right thing, and not hide our "Zorks" under a bushel.

The next section that was added was intended to be the last: after a player had accumulated all the points in the game, he could play the End Game, designed largely by Dave. This became the section of Zork II with the Dungeon Master, and at the time was certainly the most involved, and hardest (as it should have been) thing in the game. The implementation was, if anything, more involved than the problem. Less than two months later, though, Marc had come up with something worse, probably during a boring anatomy lecture. The bank section has probably been fully deciphered by fewer people than anything else in the game; even those who solve it on their own don't usually understand what was going on. I can only say that it makes sense if you understand it.

For some time, we'd been getting bug reports, fan mail, and suggestions for new problems from all sorts of people. We were beginning to run a little short on ideas anyway, and one of the ideas we got was very good. During a lengthy dinner at Roy's, our favorite Chinese restaurant, we worked out the details of the jewel-encrusted egg, purple prose courtesy of Dave. Many people on the net had long since solved the game, but went back in and did any new problems that came along; one of them had played D&D with Dave, and called him up about a day after the egg was announced. "I've gotten the egg opened, but I assume you losers have some nonsense where you do something with the canary and the songbird. Dave, no fool, said "Cough, cough, ahem, of course," and immediately went off and added the brass bauble.

The remaining puzzles, the Royal Zork Puzzle Museum and the palantirs, were added in the late summer and fall of 1978. The puzzle was designed (several times) primarily by Bruce, who in theory was back trying to finish his dissertation. Finding the minimum number of moves required to solve it was a popular pastime among dedicated Zorkers for a while.

The last (lousy) point was a tribute to the final point in the original Adventure, which involved leaving a particular object in a particular room for no particular reason. When we first solved Adventure in 1977, Bruce finally figured this out by using a machine-language debugger on the running game (since Adventure was not written in machine language, this was not easy). The major difference between that and our version (a stamp worth One Lousy point) is that it would be harder to find ours without the source of the game.

The last puzzle was added in February of '79. We (mainly I, at this point) kept fixing bugs for almost two more years—the last mainframe update was created in January of '81. No new puzzles were added because none of the implementers had time or inclination, and because we had no more space available: at the time, we were limited to a megabyte of memory, and we had used it all up. The first article about Zork appeared in April of '79, and attracted a great deal of interest; some of this may have been because we offered to give people the game (if they didn't already have it), and gave them parts of the sacred sources as well.

Infocom was incorporated in 1979 by various people from the DM group, including Marc, Dave, and me. It was not founded to sell Zork; rather, it was founded to give group members somewhere to go from MIT. Marc and Joel Berez (both exiled to Pittsburgh) determined that it would be possible to make Zork run on something cheaper than the $400,000 PDP-10, and the company eventually went along. See the next NZT for further details.

In the meantime, we still get requests for hints on the mainframe Zork (sometimes it's called Dungeon, and often it's on something other than a PDP-10). The most recent request for a copy came in on April 1, but I think it was serious.

The Final (?) Chapter: MIT, MDL, ZIL, ZIP

The year: 1979. As Tim Anderson has recounted in previous installments in this series, Zork was one large computer game, about a megabyte in size—as large as it could be and still fit in its original home, a DECsystem-10. Marc Blank and Dave Lebling designed and wrote the program, with the help of Bruce Daniels and Tim. They had met and worked together in a research group at M.I.T., and now the group was losing valuable talent through graduation and the lure of "the real world." Several members of the group believed that they could still produce outstanding computer-based products in almost any category—from programming languages like MDL (an important influence on modern Lisp) to data bases, electronic mail and artificially intelligent systems—if only centrifugal force didn't separate them.

The problem: What sort of product could the group work on together, and to whom could they sell it? As early as 1976, they had discussed the potential marketability of various computer games that had been designed or implemented by group members just for fun. Now their attention was focused on various potential products based on mini-computers, some involving custom hardware as well as software. The group was ignoring the potential of a mass market for micro-computers, not only from lack of experience with them (the group's unofficial motto is "We hate micros!") but also from serious concerns about software piracy.

Joel Berez had graduated from the group and was working in his family's business in Pittsburgh. Marc had finished medical school (and moonlighting on Zork development) and was starting his medical residency in Pittsburgh. These two had long been friends, and they liked getting together for a Chinese dinner and conversation.

One topic of conversation was "the good old days" at M.I.T., and one reason that the old days were good was Zork. They wished that Zork's wonderfulness (or "taste and winnage" in M.I.T. jargon) could somehow be brought to more people. But very few people had access to the large computers that could run Zork. More and more people were beginning to buy personal computers—like the Radio Shack TRS-80 Model I or the Apple II—but those computers were too small to run Zork. Or were they?

Joel and Marc began some seat-of-the-pants design work (much of it on Joel's parents' coffee table) on how much Zork could be compressed, and how to do so in a flexible way to allow for different, incompatible personal computers in the future. They considered using available "portable" tools for programming, like UCSD Pascal, but it soon became clear that Zork had too much text in it. (Keep in mind that a standard personal computer at this time came with 16K bytes of memory and no disk drive.) They finally concluded that, by inventing a programming system specifically for Zork, they could fit about half of it into a computer with 32K bytes of memory and one floppy-disk drive.

Meanwhile, the group at M.I.T. was in the process of forming a corporation—choosing "Infocom" as the name least offensive to everyone—and searching for a project that would quickly produce a product to start generating income for the company. Among the projects they considered were systems for keeping track of documents, handling electronic correspondence, and processing text. When Zork was added to the list of possibilities, Joel and Marc worked intensively during the summer and autumn creating the programming tools for their design. And they had to work for IOUs, since the company treasury—which started with only $11,500—could afford to pay only for the hardware they needed at the time.

The key to their design was an imaginary computer chip called the "Z-machine." This chip would be able to run Zork (or at least part of it) if the program were coded in a special, very compact language. Then the design called for each personal computer to have a program (called a Z-machine Interpreter Program or ZIP) that would interpret the special Z-machine language and make the computer act the same way that a real Z-machine would. In order to get Zork written in this special language, another language was invented, called Zork Implementation Language (ZIL), similar in many ways to MDL. Marc built a two-stage translator program that would translate a ZIL program, first into an assembly language and then further into the Z-machine language. He also built a ZIP so that a DECsystem-20 could emulate the Z-machine.

There was still the problem of cutting Zork in half. Dave examined his complete map of Zork and drew a boundary around a portion that included about 100 or so locations: everything "above ground" and a large section surrounding the Round Room. The object was to create a smaller Zork that would fit within the constraints established by the design of Joel and Marc. Whatever wouldn't fit was to be saved for another game, another day.

In the process of being converted from MDL into ZIL, the program became "cleaner" and friendlier. The geographies of the maze and the coal mine were simplified so that the connections were less arbitrary, and in other places complexity was removed whenever it didn't serve a justifiable purpose. For example, there was originally a barrel sitting near the top of Aragain Falls, but it was just a red herring; its only purpose was to lure unsuspecting adventurers inside and carry them over the falls to destruction. The Rainbow Room had its name changed to On the Rainbow, and that meant removing the silly joke about Rockefeller Center and the NBC Commissary. Since the Land of the (Living) Dead (the word "Living" was removed in order to fit the name on the status line) no longer led to the stairway where Zork III later began, the crystal skull (a brand-new treasure) was put there instead.

By late 1979, Joel and Marc had both moved back to Boston. Joel had been elected president of Infocom and started business school, and Zork I was shaping up as Infocom's first product. Zork I first saw the light of day on a DECsystem-20 on which the company was renting time, then on the PDP-11 in Joel's bedroom. Scott Cutler (who had graduated from the group a couple of years before) used his TRS-80 Model II to create a ZIP for a TRS-80 Model I. As 1980 dawned, Infocom spent a large portion of its bank account to purchase a Model I, and Scott and Marc demonstrated that Zork I was alive in it by starting the game and actually collecting points with the incantation "N.E.OPEN.IN." (It's certainly no less inspiring than "Come here, Mr. Watson; I want you!")

Mike Dornbrook was enlisted to test Zork I for bugs and other bad features, because he had some experience with computers but no experience with the original Zork, exactly like our intended audience. (One of his contributions was the alternate—and, some say, more logical—solution to the Loud Room puzzle, which was added only after the first users of Zork I asked so often for hints for that puzzle.) He played it so much that he memorized the entire geography, and he fell in love with the game. He was convinced that it would attract a cult following, although others thought it would last maybe a year on the market and then fade away, like a video game. He urged the company to start planning spin-off products, like maps, hints, posters, T-shirts, etc. So the first published release of Zork I had another feature added, a "small piece of paper" in the artist's studio that said something like "Write to Infocom, P.O. Box 120, Cambridge, Mass. 02142 for info on other products, including Movement Assistance Planners (M.A.P.S) and Hierarchical Information for Novice Treasure Seekers (H.I.N.T.S.)." Besides leaving the door open for an after-market in Zork accessories, we wanted to start building a mailing list of customers for future direct mailings (like the one you are reading!).

Now that the company had a flesh-and-blood product, how could a small group of hackers market and sell it? One possibility was to produce it ourselves and distribute it through computer chain stores. But that meant devoting time and energy to finding suppliers, producing packages, supporting users, and so on. Another possibility was to contract with a software publisher, but which one? Joel contacted Microsoft, but they were already publishing the original "Colossal Cave" adventure game—the one that inspired Zork—and by the time Zork fan Bill Gates heard of our offer, Infocom was deep in negotiations with Personal Software Inc. (PS).

PS had several good features: it was the first true publisher of software developed by others; it was the leading publisher of computer games at the time; and it had strong ties to Software Arts Inc., where VisiCalc was invented (_requiescant_in_pace_), and where Zork I was demonstrated in February 1980. PS agreed in June to publish Zork I and sent us an advance on royalties, our first bonafide income! Sales began in December, and over the next nine months PS sold about 1500 copies of the TRS-80 version.[1]

Also in June, we paid for a search of trademark records in preparation for registering "Zork" as our own trademark. We discovered that Mattel Inc. had registered "Mighty Zork" in 1973 for a toy model motorcycle, but that registration was cancelled in October 1979. Other trademarks discovered in the search were the likes of Zorr, Zorak, Zark, and Zowees (all by Mattel); Zogg, Zon, Zak, Zok, Zot, Zonk, and Zerak; and variations on Mork and Ork (by Paramount Pictures). Not to mention the Zork Hardware Company of El Paso and Albuquerque.

We had another product in which PS had no interest: the PDP-11 version of Zork I. We sent product announcements to various places, including a newsletter for PDP-11 users, and as a result, the first copy of Zork I sold was a PDP-11 version! It came on an eight-inch floppy disk with a manual that I wrote and Joel had reproduced from a typewritten master.

By the end of 1980, the version of ZIP for the Apple II had been created by Bruce, who had designed puzzles for the original Zork before graduating from M.I.T. and going to work for Apple Computer Inc. Apple Zork I proved more popular than the TRS-80

[1] Zork I came under the wing of PS's New Products Manager, a fellow named Mitch Kapor, who later founded Lotus Development Corp.

version; PS sold over 6000 copies in eight months.

The first press reviews of Zork I were encouraging. In February 1981, BYTE magazine said, "No single advance in the science of Adventure has been as bold and exciting as the introduction of Personal Software Inc.'s *Zork, The Great Underground Empire.* ... That the program is entertaining, eloquent, witty, and precisely written is almost beside the point. Unlike the kingdoms of the Adventures for machines with 16K bytes of memory and far from the classic counter-earthiness of the Colossal Cave in the original Adventure, Zork can be felt and touched—experienced, if you will—through the care and attention to detail the authors have rendered. .... [A] most excellent and memorable work of computerized fiction."

Mike Dornbrook was enlisted again to fulfill mail orders for personalized hints. Joel collected orders from the post office box, passed orders for maps and posters to his Significant Other for fulfillment, gave requests for hints to Mike, and gave me the numerous small checks to deposit in the bank. Mike created personalized hints off the top of his head, typing them on an old office typewriter. (When Mike started business school in September 1981, he founded a separate company, the Zork User's Group, and took over all mail-order sales. Only then did he computerize the operation. In 1983, Mike came back to work for Infocom, bringing Z.U.G. with him.)

Meanwhile, Dave was designing Zork II. At first, the most straightforward approach seemed to be to use everything left out of Zork I and simply convert it from MDL to ZIL. But Dave's active imagination kept inventing new puzzles that virtually begged to be implemented. So the final design left out the Royal Puzzle and the "end-game" (both to appear in Zork III) and instead included the Wizard of Frobozz, the garden, and the new diamond maze. (The last was re-oriented to the compass based on Mike's belief that "southpaw" should be a hint.) The last of the original puzzles—the long slide and "sending for the brochure"—were left out of Zork III and didn't reappear until Sorcerer.

Zork II was offered to PS in April and licensed in June 1981, about the same time that Joel graduated and became Infocom's first salaried employee. But we had serious concerns about PS's commitment, even to Zork I. After an initial rush of advertising, Zork I seemed to join PS's range of products as just another game. We were eager to make new versions and new titles—including Zork III, "Zork: the Mystery" (Deadline), and "Zorks in Space" (Starcross)—but not if our publisher wasn't also eager. The fact was that PS was planning to drop its line of entertainment software—since their titles neither sold well over the long term nor brought in enough money to satisfy them—and to change its name to Visicorp in order to identify closely with its "Visi-" series of business products.[2]

It now appeared that we had two choices: to negotiate and contract with another publisher (and to hope for more satisfaction), or to take the plunge and *become* a publisher. We definitely preferred the second choice, but that required office space, production facilities, an advertising agency, and so on—and most of all, money. But we threw caution to the wind, and hired Mort Rosenthal (who later founded Corporate Software Inc.) as marketing manager, who found a time-shared office in Boston's venerable Faneuil Hall Marketplace, a time-shared production plant in Randolph, an ad agency in Watertown, an order-taking service in New Jersey, a supplier of disks in California, and so on. The money came both from the company's founders and from a bank loan that they personally guaranteed.

We announced Zork II and our new role as publisher with a Christmas promotion as eye-catching as we could afford. Thanks to our ad agency, we had a new style of packaging for both Zorks (the stone-built letters that are still in use), a counter display for stores, ads in major computer magazines, and direct-mail ads for dealers. We also

[2] In December 1984, after a long legal tangle with Software Arts over VisiCalc, Visicorp eventually merged into one of its own spin-off companies and disappeared.

bought PS's entire inventory of Zork I (except the TRS-80 version, which they still wanted to sell) to prevent them from "dumping" it on the market at bargain prices and lowering the public's image of "Zork" in general. Our first shipment went out just in time for Christmas sales.

On New Year's Day 1982, we moved the company to larger space at the far end of Cambridge—55 Wheeler Street. Now we had office space for everyone, especially for Marc (now vice-president for product development) to finish Zork III. And we had enough space to set up all the personal computers—instead of shuffling them from one person's home to another—for testers to use, and for programmers to create or adapt ZIPS for Atari, CP/M, IBM PC, and other machines. Zork III was finished in the autumn, about the same time that the company began hiring people to begin developing its first business product. But that's another story.