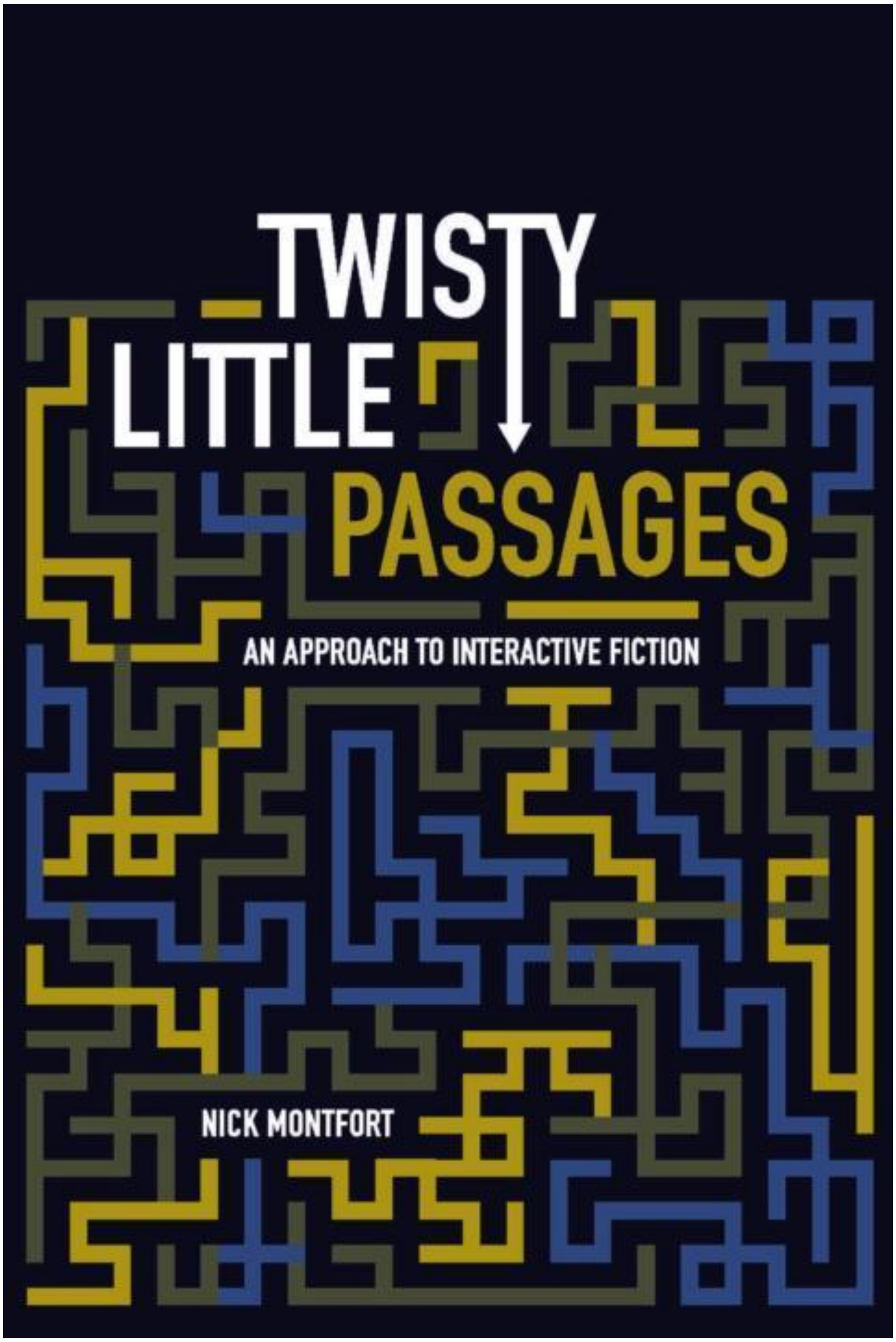

TWISTY LITTLE PASSAGES

TWISTY LITTLE PASSAGES

An Approach to Interactive Fiction

NICKMONTFORT

The MIT Press
Cambridge, Massachusetts
London, England

First MIT Press paperback edition, 2005

This book was set in Bembo and Meta by The MIT Press and was printed and bound in the United States of America.

Library of Congress Cataloging-in-Publication Data

FOUR

ZORK AND OTHER MAINFRAME WORKS

For those who had access to mainframe computer time, writing one's own Adventure-like program seems to have been almost as popular an activity as was playing Adventure. Thus, Adventure became the model for computing gaming overall; Nelson (2001b) writes that "for the five years to 1982 almost every game created was another `Advent'" (347). At Stanford, where Don Woods had augmented Will Crowther's original work to create the canonical Adventure, at least two follow-ups were written on different systems during this time: FisK and Lugi (Meier 2002). The authors of these two works are not recorded online, and the source code has not been made available. At the University of Waterloo in Canada, Marc Niemiec created a language called F (for Fantasy) for the development of interactive fiction on the Honeywell Level 66, using it to write New Adventure. Brad Templeton and Kieran Carroll wrote Martian Adventure in F, incorporating at least one puzzle that was somewhat "Martian" in the sense of the Martian Poetry movement, and shows the relationship between the riddle and early interactive fiction: the player character encounters "keys" that cannot be picked up. It turns out these are not the sort of keys one expects in an adventure; to deal with them one is supposed to type (Meier and Persson 2002). For the Hewlett Packard 1000, two IF works were programmed and made freely available: Mystery Mansion by Bill Wolpert and Warp. An incomplete adaptation of The Lord of the Rings entitled Lord, created at the University of Helsinki by Olli J. Paavola, has been called, by Nelson (2001b), "the first book adaptation in interactive fiction" (347). Nelson noted that Lord, influenced by Zork as well as Adventure, was particularly authentic in the way it created a world with the sort of detail Tolkien would appreciate.

At Carnegie Mellon University, during 1979-1982, John Laird (1997) wrote the Adventure follow-up Haunt, which he has described as "over-thetop," "quirky" and also "a bit buggy." Nelson calls it "not inspired" but notes that it is the first non-cave work of interactive fiction (2001b, 353).The outrageous prologue is worth quoting from extensively. It merely introduced the sort of treasure-hunt situation that quickly became stereotypical, but it is a great departure from Adventure and Zork, which simply begin with the player character in the woods.

> Along [sic] time ago, a young couple was picnicing [sic] near the woods on the outskirts of town. They were celebrating the birth of their first child. Unfortunately, a crazed moose inhabited that area and attacked them. The child and husband were unharmed, but the wife was gored to death by the moose.
>
> After the funeral, the man bought the land where the incident occurred and constructed a large mansion: CHEZ MOOSE. He filled it with the treasures of his family and claimed that his wife's soul was still in the area. He vowed to remain in the mansion until he had returned her soul to human flesh. He tried to bridge the gap between life and death to reclaim her. . . . Several people have entered the mansion looking for him but none of them have ever returned. There were rumors that he and his wife now haunt the house.

> That would be the end of the story except that the house still stands and is filled with priceless treasures. The house and all its contents are willed to his only descendant.... The terms of the Will say you get to keep any treasure you get to the lawn, but of course you must also get off the premises alive.... If you are insane enough to try, your adventure starts at a bus stop.

Laird (1997) himself wrote that "the puzzles were way too obscure (many based on Saturday morning cartoons from my youth)" but admitted they were certainly creative, suggesting a different, even wackier mode for interactive fiction.

One puzzle involves removing the paint from a painting to reveal a more valuable one underneath. Another requires the player character to use two straight objects to make a vampire-resisting cross.

Numerous obscene inputs are recognized by Haunt. Marijuana is one of the treasures; smoking it makes the adventurer hungry. Stranger effects are brought on by eating the sugar cube. After winning the house, the player character is confronted with a final dilemma when James Watt arrives from the Department of the Interior and asks to buy the land. Written in CMU's OPS-4 (Official Production System 4), Haunt made its way to a few other universities but was not widely available. It was notable for being the first rule-based system to have more than 1,000 rules (Laird 1997). Laird (1998) wrote it without having played Zork; the creators of Zork played it at some point before Zork was finished (Nelson 2001,353; Lebling 1997). Laird began a port of Haunt to OPS-5; this incomplete version has been used as an exercise by CMU students (Winalski 1997). CMU would later host the major academic effort in interactive fiction, Joseph Bates's Oz Project, discussed in chapter 7. Laird, now a professor of computer science at the University of Michigan, is currently developing a new domain for his artificial intelligence research, which he describes as "an adventure game where the player takes on the persona of a ghost-like energy creature trapped in a house" (Laird 2001).The graphical, 3-D system is being built using the Unreal Tournament engine and is known as Haunt 2.

The mainframe Zork, programmed at MIT beginning in May 1977, has a special place among the mainframe follow-ups to Adventure. Zork became the second widely known interactive fiction work. For less capacious home computers, it was split up and further developed into Zork I, Zork II, and Zork III -a trilogy of best-selling games that was the basis for Infocom's creative and financial success, which I discuss in chapter 5. Several specific improvements that had been made to the Adventure world model and parser were touted in an academic paper (Lebling, Blank, and Anderson 1979), and, later, in the manuals that accompanied Infocom's commercial products. Those familiar with the history of computer gaming recognize the significance of Zork. One contemporary IF author expresses a common view: "It's by far the most famous piece of IF and can be considered the father figure of the genre, much like Super Mario Bros. is with the side scroller or King's Quest for the graphic adventure" (Sherwin 1999). One rough measure of the overall popularity of Zork is that in mid-December 2001, the word "Zork" appeared in approximately 124,000 Web pages in the Google index; only about 22,000 such pages mentioned either "Space War" or "Spacewar," referring to the MIT program that is widely considered to be the first modern video game. Even taking into consideration that the word "zork" sometimes appears on the Web in its original sense-a nonsense word in MIT argot-the interactive fiction Zork is clearly well known; it is also important in the history of new media and of interactive fiction.

Zork in this chapter refers to the PDP-10 computer program, running on ITS and written in MDL, that was developed at MIT beginning in late May 1977 and augmented later that year and in the following years (Anderson 1985a, 7). For a few months Zork was called Dungeon; there is a version of a FORTRAN port of Zork that has been widely available since 1979 that uses this name. Zork also refers to ports from that FORTRAN port into Glk and from the original MDL into Inform. Zork is distinguished in this discussion from Zork I-III. I consider that trilogy and how it differed from the mainframe Zork in the next chapter. Where distinctions among different versions of Zork are important, the year of the version cited is given.

Despite the clear importance of Zork, with very few exceptions, the advances that this work brought about have been only superficially considered. This chapter describes three advances and their importance. First, Zork draws on a deep reservoir of technical humor, making its origins in the subculture of MIT evident and commenting on technology in interesting ways. It first realized the ability of interactive fiction to speak back to the culture in which it was produced. Second, while the technical advances made by the game's creators (known as Implementors) were in some ways incremental, they did lead to progress in interactive fiction. It is helpful to explain these advances and see how these proved important in the development of later works. Third, Zork was at least in some ways superior to Adventure as a system that generates satisfying narratives in response to user input. It produced narratives that were connected to adventure stories of the folktale sort in new ways-not just through offhand reference, but structurally. Neither programming improvements nor better writing, considered alone, made Zork's thief the first memorable character in interactive fiction; rather, it was the way he functioned during an interaction and throughout a successful traversal of Zork that was fundamentally better than what had been achieved before.

SEEING ZORK AGAINST ADVENTURE

The potential narratives of a successful traversal of Zork involve the adventurer wandering through an outdoor area, a house, and then a vast complex called the Great Underground Empire.To succeed the interactor must direct the adventurer to collect treasures, almost all of which are located underground. A handful of living opponents thwart the adventurer: the troll, who stays put in a single room and serves as an obstacle; the vampire bat, who can carry off the adventurer; the cyclops, who can dine on the adventurer; and the thief, who wanders around the underground areas stealing items the adventurer either is holding or has already seen. To get through the mazes, detailed mapmaking (or else extraordinary luck, or cheating) is required. Riddle-like challenges require the interactor to understand the nature of a disguised object in order to use it properly; in some cases machinery must be manipulated in order to determine its purpose. Some puzzles, such as the Bank of Zork and the Royal Puzzle, are elaborate and extremely hard. A few require that the interactor "guess the verb" and perform an action that would not be obvious from the commands available in Adventure. If a magic word from Adventure is typed in Zork, a hollow voice says "Cretin."

It makes little sense to consider certain features in Zork as if they were original, because certain features are not. Attacking Zork with a detailed psychoanalytical reading that considers the subterranean world, beneath a forested area in which a lone building sits, is foolhardy-for several reasons, but most specifically because all of these features are lifted directly from Adventure. According to the small leaflet found in the mailbox in the 1978-1979 FORTRAN port of Zork, the program "was created ... by Tim Anderson, Marc Blank, Bruce Daniels, and Dave Lebling. It was inspired by the Adventure game of Crowther and Woods, and the Dungeons and Dragons game of Gygax and Arneson." By 1981 the leaflet had been revised to replace mention of Dungeons and Dragons with "the long tradition of fantasy and science fiction adventure." This revision came after Zork was briefly renamed Dungeon, drawing the attention of certain lawyers to the Implementors and MIT (Anderson 1985b, 4). Both the original and revised statements have truth to them, of course.

Zork's Implementors had played Adventure earnestly, and they created Zork mainly in response to their enjoyment of that interactive fiction work and the opportunities they saw to improve upon it. The interests of the Implementors were not restricted to programming. As Anderson (2001) said, "One reason that some of the writing is good, if it is good, is that the people who wrote it were not nerds in the classic sense. Very few people are really nerds in the classic sense, I think. Marc, Dave, and I all exposed ourselves to a lot of really good writing."The Zork Implementors were also filmgoers, seeing Star Wars very soon after it came out, during Zork's early development;

Marc Blank was on MIT's Lecture Series Committee, the student group that screens films on campus, and had an interest in classic films. The Implementors read books and viewed films eclectically, taking in much from outside the genres of science fiction and fantasy. This included history, as one might guess from the offhand reference in Zork to Octavian's general Marcus Vipsanius Agrippa.

Few people are struck by the powerful narratives that Zork produces when compared to those produced by Adventure. Zork, like Adventure, provided nothing special to set up the initial situation, in which the adventurer is standing near a building in the woods; certainly Haunt was much more innovative when this particular aspect of writing is considered. Although the humor of Zork is widely appreciated, in some ways Zork's writing may in fact be worse than the writing in Adventure, especially in the case of the accuracy and richness of geological description. Nelson (1995a) described Zork's underground setting as "based not on real caves but on Crowther's descriptions" and as "better laid out as a game but not as convincing." Lebling (1988) explains that those at MIT who weren't cavers still had their own underground empire to explore and to base interactive fiction upon:

> When I was a student at MIT, there was a pastime called "Institute Exploring" (also known as "Tunnel Tours"). A group of students would go over to the main part of the campus at around 3am and try to visit some of the more obscure and off-limits locations.
>
> MIT is full of basements and subbasements, and these are often crammed with equipment left over from some cancelled

research project.

However many levels of simulation are involved in the room descriptions, though, Zork did substantially improve upon the art of interactive fiction. Janet Murray (1997) describes one way in which Zork presents a compelling "moment of experimental drama," early on: when the adventurer first enters the dungeon, the trapdoor is closed and the way out is barred by an unseen opponent (81-82). This moment, which has no parallel at the beginning of Adventure, creates an ominous and interesting situation in the potential narratives of Zork. It also makes for a puzzle-the interactor must figure out how to escape, going deeper into the dungeon.This improvement, one of several that made interactive fiction and new media more compelling, came about by improving the design of the IF world, not by simply programming more cleverly or by writing better bits of texts.

Situated in Subculture: Technical Humor and Critique

Zork is a literary, gaming, and computing artifact, a part of the culture and subculture in which it was created. It has been noted that masculinist rationality underlies Zork and other interactive fiction (Sloane 2000, 100); other contexts of interest also exist. The Implementors were certainly influenced by, but not mere puppets of, male academic research and its perspectives and concerns. In Zork they sought to comment on academia and technology as technologists and as writers and designers of a new kind.They were marked in certain ways by their environment, and they marked back in Zork, the simulated environment they created.

To illustrate how an MIT-specific interpretation can help us understand Zork, consider that in the Entrance to Hades a large gateway bears this inscription: "Abandon every hope, all ye who enter here." Relating this inscription to literary tradition is, if not fruitless, misleading. Zork's Hades has only superficial similarities to either Dante's Hell or the Hades of the ancient Greeks, with jeering "evil spirits" who prevent entry. Dante's Hell lies conspicuously open and unguarded-in contrast to the securely locked gate of Purgatory, watched over by a sword-wielding angel (Purgatorio 9.76-138). Christian doctrine of Dante's time held that it was easy to enter hell; entering Zork's equivalent realm is difficult, providing an intricate puzzle opportunity. A different approach to this Hades entryway is provided by Implementor Tim Anderson (2001): "You can think of it as a Dante quotation or you can think of it as something people say when they come to MIT, which is really more what we had in mind." MIT students come to learn that "Tech Is Hell," and admission to that Hell is indeed difficult. There was also a rather direct antecedent in computing, whether or not it was known to the Zork Implementors: The (rather arcane) source code for the Unisys A Series COBOL 68 compiler began with a comment containing the same quotation, in Italian (Stevens 1999).

The first glimpse of Hades reveals "a desolation, with a pile of mangled corpses in one corner." Hades is hardly a vast, mythical wasteland if it is small enough to have a pile of corpses in the corner, as in this tongue-in-cheek description. Actual lifeless corpses should also not have much of a place in either the Greek Hades (in which they would be wandering around listlessly) or Dante's Hell (where they might more appropriately writhe beneath their punishments). The rest of The Inferno is summarized in Zork at the entrance as

"Thousands of voices, lamenting some hideous fate," and within as "the sounds of thousands of lost souls weeping and moaning." (The command listen is not implemented.) One reviewer's reaction to the corresponding region of Zork I is typical: "Here I am in Hades. *yawn* Wonder if there's a gift shop around" (Stevens 1997).

However, within the land of the dead is a room that didn't make it to the commercial Zork I. This area, except for the presence of severed heads and dead bodies, might fit in perfectly well at MIT:

> Tomb of the Unknown Implementer
>
> This is the Tomb of the Unknown Implementer. A hollow voice says: "That's not a bug, it's a feature!" In the north wall of the room is the Crypt of the Implementers. It is made of the finest marble, and apparently large enough for four headless corpses. The crypt is closed.
>
> Above the entrance is the cryptic inscription:
>
> "Feel Free."
>
> There are four heads here, mounted securely on poles.
>
> There is a large pile of empty Coke bottles here, evidently produced by the irnplementers during their long struggle to win totality.
>
> There is a gigantic pile of line-printer output here. Although the paper once contained useful information, almost nothing can be distinguished now.

These signs of coding exertion (of no use to the adventurer) are situated in a place that is certainly not a direct reference to the Tomb of the Unknown Soldier, but rather to the Tomb of the Unknown Tool, a famously inaccessible area under MIT's Building 9. "Tomb" at MIT refers to, as one student publication explains, "an interesting, out-of-the way, unused spot" (Amonlirdviman 1996, 213). The heads are "securely mounted" because if they were removable the adventurer could carry them around and do things to them, thereby introducing unnecessary complexity to Zork's simulated world. The inscription punningly decorating the crypt was a common saying among the four creators (the spelling "Implementers" later being changed to "Implementors") in response to a suggestion that part of Zork be modified. It was a polite way of refusing, meaning "feel free to go ahead and make the change yourself if you like." For example, "How about you add an underwater area inside the reservoir that you can swim down into?" "Feel free."

This vein of technical humor continues into the south wall of Zork's temple. There is a prayer upon it, "inscribed in an ancient script which is hardly remembered these days, much less understood. What little of it can be made out seems to be a philippic against small insects, absent-mindedness, and the picking up and dropping of small objects." The prayer begins with an attempt to protect not against literal "small insects," of course, but against programming errors, or bugs. As for absentmindedness, it might be something the Implementors need to guard against in creating Zork, or it might be a professorial malady that students should watch out for. Clearly the last philippic is making fun of the adventurer, whose essential actions seem to be picking up and dropping things.

Zork has a purely numerical joke that may be the most elaborate in all interactive fiction-perhaps even in all computing. In the Clearing a command to count leaves brings the wry response "There are 69,105 leaves here." This reply presupposes a superhuman (and in fact computer-like) adventurer, able to count a tremendous number of objects in the thin slice of time represented by a move. Perhaps this prodigious ability to count is in keeping with the adventurer's autistic nature, as manifested in the emotional understatement and the fixation on objects that Aarseth (1997, 115-117) has pointed out. Whatever the case, the absurd, impossibly accurate count is funny, as is the "364.4 Smoots and one ear" measurement first marked on the Harvard Bridge in October 1958 by MIT students who had just finished measuring the bridge with Oliver Reed Smoot's supine body. The same sexual innuendo is insisted upon twice in the digits "69,105"-the "69" to the left of the comma is repeated to the right of the comma, since decimal 69 is octal 105, and (as is not true in general) hexadecimal 69 is also decimal 105. This number appears again in Infocom works The Witness by Stu Galley (the gun receipt is number 69105) and in Leather Goddesses of Phobos by Steven Meretzky (which has another pile of leaves). In works from the late 1990s, Adam Cadre's I-0 features 69,105 pieces of laundry in the trunk of car; Admiral Jota's in-joke Pass the Banana has a file size of 69,105 bytes. The number also is mentioned in Infocom's newsletter The New Zork Times and in the instructions to Douglas Adams's Bureaucracy, another Infocom work. But in case one's appetite for numeric allusion to mutual oral sex is not satisfied at the "Clearing," there is more in Zork along similar lines. The description of the "Studio" mentions that the "walls and floors are splattered with paints of 69 different colors."

Plenty of additional MIT-specific and computing references are to be found in Zork. Descending from the Dome Room to the Torch Room, the adventurer sees this: "Above you is a large dome painted with scenes depicting elvish hacking rites." "Hacking" at MIT refers to the exploration of restricted areas of campus and to the perpetration of extremely clever pranks, some of which are in fact ritualized. Although "dome" meant something else inside a cave, one of the most prominent sites for hacking has been MIT's Great Dome, which has had a phone booth and a model police car placed atop it (Haverson and Fulton-Pearson 1996; Leibowitz 1990). This makes it particularly appropriate for these frescos to appear in a "large dome."

There were also jabs at the large evil computer company of the day, IBM, found in the Machine Room and the Maintenance Room: "Along one wall of the room are three buttons which are, respectively, round, triangular, and square. Naturally, above the buttons are instructions written in EBCDIC." These indecipherable instructions are encoded in IBM's proprietary character code, a rival to ASCII. Other technically funny possible outputs of Zork include "Why, only last week I patched a running RSX system and it survived for over thirty seconds," and, on an unfortunate occasion, "According to Prof. TAA [Timothy A. Anderson] of MIT Tech, the rapidly changing magnetic fields in the room are so intense as to cause you to be electrocuted. I really don't know, but in any event, something has killed you." Engravings are seen to read "This space intentionally left blank," the same notice found on blank pages at the end of engineering and science textbooks, and possibly a reference to one of the Implementors as well.

Technology plays an important part in the world of Zork, which sports Flood Control Dam #3, a flashy public works project built for no discernible reason, as the Guide Book explains:

> Flood control dam #3 (FCD #3) was constructed in year 783 of the Great Underground Empire to harness the destructive power of the Frigid River. This work was supported by a grant of 37 million zorkmids from the central bureaucracy and your own omnipotent local tyrant Lord Dimwit Flathead the Excessive. This impressive structure is composed of 3.7 cubic feet of concrete, is 256 feet tall at the center, and 193 feet wide at the top....
>
> The construction of FCD #3 took 112 days from ground breaking to the dedication. It required a work force of 384 slaves, 34 slave drivers, 12 engineers, 2 turtle doves, and a partridge in a pear tree. The work was managed by a command team composed of 234 bureaucrats, 2347 secretaries (at least two of whom could type), 12,256 paper shufflers, 52,469 rubber stampers, 245,193 red tape processors, and nearly one million dead trees.

FCD #3 is a concrete symbol of engineering as worshipped by the public, similar in this way to the impressive Hoover Dam. It is also amusingly useless, almost certainly incorrectly documented (it would take rather magical engineering practices to build a 256-foot-tall dam using nothing but 3.7 cubic feet of concrete), and apparently produced with an inefficiency more monumental than the dam itself. Zork's technical humor is an achievement not just because it is funny, but also because it delivers things like this effective parody of bumbling publicly funded technology projects. Although Zork was created more efficiently than was FCD #3, it did run on the spare time of computers purchased with, and was programmed by individuals who were supported by, Department of Defense money. It was in some ways an example of an amusingly useless public works project itself; the tongue-in-cheek tone of Zork seems to encode an awareness of this.

The structural innovations in Zork's world also reflect a technological subculture. To prevail the adventurer must use vehicles, riding in a balloon and a boat. This player character also must command a robot in order to get through one section of the work. Reading Zork against MIT makes it easier to see how the literary transformation of technology is accomplished in the work. The puzzles that offered riddle-like systems, called "problems" by the Implementors, often involve recognizing various technological artifacts, real and imagined, for what they are. In many cases, once the technology is recognized, the way in which objects are supposed to be used becomes obvious. A clear example is provided in the coal mine section of Zork's IF world.

Here, the player character finds a machine that might, in another place, be used for doing laundry; a tiny slot is noticable in the top of it. What this machine does, and how to turn it on, is unclear. The solution is to put some coal found nearby into the machine, and then turn the machine on using the screwdriver. This results in the coal being compressed with great force. (The switch is described as having the right dimensions to be turned with a screwdriver of the appropriate type-of course, a flathead screwdriver. One of these can be found near FCD #3.) To figure out how to turn on the machine, one need only recognize which ordinary tool is needed, based on a literal description. To figure out what the machine does, the interactor can act as scientist and put anything inside, then observe the results. Despite the absurdity of an underground flood control dam, many of the technologies found in the Great Underground Empire have purposes, and one could imagine why they might have been placed there by previous occupants of the realm. Understanding that useful machinery is found in this IF world, and using the process of experimentation and observation in order to learn what this machinery does, allows the player character to prevail in a way that is consistent with the overall scheme of Zork. The diamond-making machine is not a profound riddle, but it was a step toward systematic IF worlds of greater power.

Zork was not the last piece of interactive fiction to comment on MIT. Later, Zork Implementor Dave Lebling, working for Infocom, wrote an IF work that referred to MIT more directly. His 1987 The Lurking Horror was a Lovecraftian horror story set at the fictitious George Underwood Edwards Institute ofTechnology, or GUE Tech, which was laid out much like the MIT campus. MIT's Green Building was there with its meteorology dome on top, but it was called the Brown Building. (The actual Green Building is, in fact, brown.) Many other features were lifted from MIT, including an infinite corridor and an inaccessible "tomb."As independent IF authorship began to hit its stride, one of many IF works set on the authors' own campus was GC: A Thrashing Parity Bit (?f the Mind, written by Carl de Marcken, Dave Baggett, and Pearl Tsai for the MIT Artificial Intelligence Laboratory Olympics in January 1993. Excruciatingly difficult, it featured Marvin Minsky and other campus luminaries wandering about in the Acme Institute, an MIT-like place filled with computer science and Al references as well as inside jokes. Zork clearly influenced not only MIT-referential IF works like The Lurking Horror and GC, but all other interactive fiction that draws a connection between the simulated world, full of devious puzzles and fantastic elements, and our own contemporary and more mundane reality.

World Model and Parser: Software Advances

One might question whether Zork's technical advances were actually so striking and of real importance to the interactor and to the development of interactive fiction. Certainly the Implementors did not think they were staggering; in reply to help, Zork stated, "You are dealing with a fairly stupid parser" Anderson (1985b) wrote that the early work was "in some ways ... 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" (4). But Zork did sport several advances that were used to good effect. These were in both essential components: the parser, which translates player-typed text into actions, and the world model, which simulates a narrative and puzzle-filled world to the degree required for an enjoyable interaction.

The parser, "fairly stupid" as it might be, was a substantial upgrade from that of Adventure, which only accepts commands of one or two words. When there was only one appropriate object for a requested action, the parser would assume the interactor wished to use that object; otherwise, it would ask a question to disambiguate the command (Lebling, Blank, and Anderson 1979).This worked particularly well in the case of actions like digging, which required tools. When a command was issued to dig with an inappropriate tool (e.g., diq in the sand with the screwdriver), the parser would generate a reply of the form "Digging with the screwdriver is slow and tedious." Since "the hands" were designated as tools, the parser, upon receiving the simple command diq, would assume-in the absence of any other tool-that "the hands" were to be used. It would then generate a response that-although unanticipated-was particularly apt and pleasing to the Implementors: "Digging with the hands is slow and tedious" (Anderson 2001). The parser folded prepositions into the different supported actions, so that "look at" and "look under" were considered as if they were separate verbs and were translated into different actions (Lebling, Blank, and Anderson 1979). Direct and indirect objects were recognized, and some verbs were allowed to take multiple direct objects.

The world model was enhanced to implement actors, who could perform actions in much the sane way that the adventurer could, and could also be commanded by the interactor. The robot, who lacked many of the adventurer's abilities but who could be commanded to solve a puzzle, was the first actor implemented. Although not the most charming character in interactive fiction (in contrast to the robot Floyd from Planet fall, who is certainly ranked among the most engaging characters in the form), the robot is an interesting part of the world of Zork. The robot is a technological artifact, almost free of personality. The best pre-robotic entity to compare it to might be the golem of Jewish folklore. (There is no hint of the more sinister proto-robot, Frankenstein's monster.) When the robot is able to accomplish some task, it emits a "Whirr, buzz, click!," while all of its other utterances are what might be termed polite refusals. After the player character had asked the robot to read something, for instance, it would reply "My vision is not sufficiently acute to do that." If commanded to eat, it somehow speaks the reply "I am sorry but that is difficult for a being with no mouth." This robot, although an uninteresting conversational partner, did first allow the player character to direct another entity to accomplish tasks on his or her behalf. This opened up new possibilities for puzzles, and also brought on interesting narrative implications, to be explored later in works like Planetfall.

Vehicles were another new part of the world. They were implemented as if they were mobile rooms, contained in the top-level rooms of the dungeon (Lebling, Blank, and Anderson 1979). The boat and the balloon were the vehicles placed in Zork. By constraining them to linear paths, in order to avoid unusual situations that might rupture other parts of the world model or require inordinate amounts of new programming, the Implementors provided novel but carefully directed experiences of travel. Other aspects of the simulated world improved upon the Adventure universe:

> Containment: Objects may have contents. Bottles can contain water and be open or closed. Some objects are transparent. Some objects must be unlocked before they can be opened. The capacity of an object is limited. (For example, a paper bag won't hold as much as a bucket.)
>
> Weight: Objects have weight.A solid gold coffin weighs a lot more than a newspaper. The amount a player can carry depends on the total number of objects carried and on the total weight of the objects and their contents.
>
> Position: An object may be in, on, or under another object. (Lebling, Blank, and Anderson 1979)

Perhaps as important as these planned-out improvements were the ad hoc changes made by the implementors in response to email requests and based on their surveillance of other users' sessions. As Anderson (2001) said, "We spied on people playing Zork. This was ITS.You could see all the output from a terminal. We would `watch' people this way." ITS was MIT's Incompatible Timesharing System, an intentionally insecure operating system made to facilitate group work. When the Implementors got a reasonable request or spied someone floundering due to what seemed like a parser failure, they would tweak Zork.

Had interactive fiction only accepted two-word inputs instead of being pushed by the Zork parser advances toward accepting text that is more like normal English, there is little chance that the appeal of interactive fiction would have lasted beyond the era of command-line home computing. Pointing and clicking would simply be good enough when compared to a verb-noun command. Instead, Zork took the first step toward a more symmetrical interaction between interactor and IF work, an exchange more like English conversation-one that cannot be easily replaced with mouse clicks (Montfort 2000). Improvements in the world model also paved the way for many interesting IF works. Michael Berlyn's 1983 Suspended (published by Infocom), for instance, used the "actor" enhancement to provide a fragmented sort of player character, whose senses were divided between different robots that could be commanded. Infocom also brought out, in 1985, Steven Meretzky's A Mind Forever hoyaging, which simulated an enormous city; some of the sense of scale was given by using vehicles of the sort developed in Zork and implementing a subway system. Hundreds of later works used containment and position to achieve a richer and more detailed world. While Zork made important progress as potential literature, its improvements to interactive fiction's technical infrastructure were definitely of value as well.

Potential Narrative: Improvements in Villainy

The thief, who appears randomly to steal treasures the adventurer is holding, is certainly the most memorable character in Zork. This is attributed variously to the better writing associated with the thief or to his nature as an IF daemon or bot (Leonard 1998, 84). In fact the thief is important to the development of interactive fiction because he functions as a true villain, not simply an obstacle or opponent. It is his role in the interaction and in the potential narratives that result that makes the thief so effective.

Phil Goetz (1994) writes that "Zork was ... the first adventure whose non-player characters had personality. The thief was a gentleman gone wrong, with good manners, a cynical sense of humour and the willingness to slit your throat in a moment." That the thief has a true personality may be an overstatement. Comparing the descriptive text that constitutes Adventure's pirate (a sort of proto-thief) with that of Zork's thief can provide some insight into the difference in personality. From Adventure:

> Out from the shadows behind you pounces a bearded pirate! "Har, bar," he chortles, "I'll just take all this booty and hide it away with me chest deep in the maze!" He snatches your treasure and vanishes into the gloom.
>
> There are faint rustling noises from the darkness behind you. As you turn toward them, the beam of your lamp falls across a bearded pirate. He is carrying a large chest. "Shiver

> me timbers!" he cries,"I've been spotted! I'd best hie meself off to the maze to hide me chest!"

Here is a selection of many brief descriptive outputs regarding the thief from Zork:

Someone carrying a large bag is casually leaning against one of the walls here. He does not speak, but it is clear from his aspect that the bag will be taken only over his dead body.

Your opponent, determining discretion to be the better part of valor, decides to terminate this little contretemps. With a rueful nod of his head, he steps backward into the gloom and disappears.

A `lean and hungry' gentleman just wandered through. Finding nothing of value, he left disgruntled.

A seedy-looking individual with a large bag just wandered through the room. On the way, he quietly abstracted all valuables from the room and from your possession, mumbling something about, "Do unto others before ..."

The other occupant just left carrying his large bag.You may not have noticed that he robbed you blind first.

The thief, a man of good breeding, refrains from attacking a helpless opponent.

The thief, forgetting his essentially genteel upbringing cuts your throat.

The thief, who is essentially a pragmatist, dispatches you as a threat to his livelihood.

Most of the thief's character is described apart from any encounter with him, in response to info:

> Of special note is a thief (always carrying a large bag) who likes to wander around in the dungeon (he has never been seen by the light of day). He likes to take things. Since he steals for pleasure rather than profit and is somewhat sadistic, he only takes things which you have seen. Although he prefers valuables, sometimes in his haste he may take something which is worthless. From time to time, he examines his take and discards objects which he doesn't like. He may occasionally stop in a room you are visiting, but more often he just wanders through and rips you off (he is a skilled pickpocket).

Clearly, much more potential text is available to describe the thief than the pirate. The text by itself cannot tell the whole story of who the thief is and how he functions within Zork. Encounters with the thief will certainly tend to vary more than encounters with Adventure's pirate. Still, this text demonstrates that Goetz's conclusion-that the thief is really endowed with personality the pirate lacks is somewhat questionable. The thief is drawn from a stock Dungeons and Dragons player character class. The brief description of the startled pirate may be more original and humorous than any particular text associated with the thief. The pirate chortles and speaks with pithy but piratic diction that is a bit more pleasing than "Do unto others before ..." However, the thief is a noticeable improvement over the pirate for one important narrative reason: the thief can be killed by the adventurer-indeed, must be killed in order for the interactor to successfully traverse Zork.

Writing about Adventure, Mary Ann Buckles (1985) noted that " [Vladimir] Propp believed villainy to be the most important function in the folktales he examined ... In my view, the lack of true villainy constitutes one of the main structural and ideational differences between folktales and Adventure" (107). She described a villain as "the symbolic representation of forces working to seemingly hinder, but actually promoting, the hero's or heroine's development" (107), adding,

> In Adventure ... I don't believe one can speak of villainy. The two types of figures who oppose the hero, dwarves and the pirate, are not capable of "evil." They do things that hinder and threaten the reader; they steal the reader's treasure and even "kill" the reader, although "death" is meaningless and not even always permanent in the game. But since Adventure's "villains" are not representations of anything else, neither parental figures nor psychological drives or impulses, their deeds are destructive without being "wicked.".. They are stick figures which have nothing to do with ethics, "good" or "evil," or aspects of personal development. (107-108)

Buckles has one more complaint regarding the pirate: "when the pirate steals the reader's treasure in Adventure, the reader regains it by finding it. The triumph and revenge aspect [of the folktale] is missing entirely" (124).

Whatever the general applicability of Propp's theories to interactive narrative may be, these points about why Adventure is structurally unsatisfying are grounded in important concerns, and they help explain the effectiveness of the thief. He can be read as a manifestation of meaningless greed-perhaps even the same greed that is driving the adventurer to loot the dungeon. (One of the sections of the dungeon is a bank that the adventurer must rob, and in the Gallery, where a lone painting is found, the initial description notes that "there is still one chance for you to be a vandal.") Seen this way, Zork's thief does begin to represent the villain Buckles finds lacking in Adventure. The most important feature of the thief is that he must be encountered and killed by the player character in a successful traversal of Zork. The adventurer can only prevail over the thief after significant experience has been acquired; a battle in which the adventurer is victorious ends with the demise of the thief and the recovery of the stolen treasures. The folktale revenge element lacking in Adventure is present in Zork.

Finally, the thief promotes the development of the adventurer in two ways: by serving as an incentive to explore more of the dungeon and improve in rank, and by unwittingly assisting the adventurer in gaining access to a treasure.

Specifically, the thief has to be "given" the jewel-encrusted egg that is found above ground, in a nest that is found up a tree outside the house. Since the egg can be taken directly to the trophy case with no chance of the thief stealing it, the interactor must choose to direct the adventurer below ground to either give the egg to the thief or have him steal it. After doing this, and then dispatching the thief in combat, the adventurer will find that the thief has carefully opened the egg, revealing another treasure.

These aspects of the thief make him a complex figure in comparison to Adventure's simple and small menagerie; he is both a helper and a villain. His role in the experience of Zork from start to successful finish, rather than his turn-to-turn behavior, is likely the reason that, in a letter to XYZZY News in which he reports a Zork I bug, one interactor wrote, "I always hated the stupid thief. Killing that lean-and-hungry not-so-gentleman with his own stolen stiletto is one of the more satisfying things I've ever done with my computer" (Gildemeister 1996).

Most of the opponents who deal with the adventurer violently and have the potential to kill the adventurer (i.e., the troll and the thief) must be killed to win Zork.An exception is the violent and potentially fatal cyclops, who is too mighty for the adventurer to dispatch but who can be dealt with conclusively in another way. (That the cyclops is dispatched non-fatally is consistent with the archetypical adventurer, Odysseus, escaping from Polyphemus without killing him.) The one opponent who never inflicts actual physical harm on the adventurer is the Wumpus-rejected vampire bat. To enter his lair without getting a one-way flight into the coal mine, the adventurer must use a nonviolent tactic, bringing an object that causes the bat to cower. The ways of dealing with opponents in Zork are therefore more consistent and systematic than they are in Adventure. In that IF work, a dragon who does not necessarily harm the adventurer must be killed-in a way that amuses but fails to even have literal meaning.

Other memorable characters in interactive fiction relied on the thief's success as a character who plays an essential role in the potential narrative. Not all of them were villains: Floyd in Steven Meretzky's 1983 Planetfall (from Infocom) had an important role as companion and helper.

Although successful, Zork was a very early and rough effort in the history of interactive fiction. The initial simple quest (accumulate all the treasures) saw an endgame added to it after 1977, incorporating an interesting twist that was absent from its contemporary IF works. As important as it was in the history of interactive fiction, Zork is not a masterwork by today's standards. MIT's answer to Adventure led interactive fiction onto a more satisfying course, however, making many advances by improving interactive fiction as an artifact that comments on technical culture, as world model and parser, and as a machine for interactively generating satisfying narratives.

Developments in the Other Cambridge

Many college campuses in the United States were one-hit wonders when it came to Adventure follow-ups. Students from MIT in Cambridge, Massachusetts, did go on to found Infocom. Another exception to the rule could be seen in Cambridge, England. Programmers there produced a massive work of interactive fiction that grew to dwarf Adventure and Zork and also devised a development system that helped others produce IF works throughout the 1980s. Shortly after Adventure and Zork made their way across the puddle, programmers at Essex University in England took off from interactive fiction in a different direction, creating the MUD, or Multiple User Dungeon, discussed in chapter 8. According to Nelson (1999b),

> The central computer of Cambridge University, England, an IBM mainframe usually called "Phoenix" after its operating system, was one of those to receive "Advent" (a.k.a. "Colossal Cave") and "Zork" (a.k.a. "Dungeon") in the late 1970s. Two graduate students,Jon Thackray and David Seal, began a game called "Acheton" in 1978-9: with the aid of Jonathan Partington it expanded for another two years. Possibly the first game written outside America, by 1981 it seems likely that it was also the largest in the world (it has 403 locations). "Acheton" was written with a game assembler contemporary with Infocom's proprietory "ZIL': unlike ZIL, Seal and Thackray's game assembler was available for public use, the public in question being all users of Phoenix c. 1980-95.

Acheton, like Adventure and Zork, features an outdoor area, underground area, and endgame part of the world-even a last lousy point is included-but it is, as one recent reviewer wrote, "quite a bit tougher and more cruel" than its American predecessors (Russotto 2000).The player character is frequently killed without warning, as when walking around, picking up a treasure, or closing an empty safe. There are numerous mazes. As Russotto recently put it, "When the player is tempted to write a Java program to discover a Hamiltonian path through a maze, the maze is perhaps a bit too difficult." While a non-programmer like Tracy Kidder could get the hang of Adventure, Acheton was clearly esoteric.

This extremely difficult challenge was appreciated, though, because Acheton effectively inspired programmers in Cambridge to create other works of interactive fiction using its development system. Fourteen such works were introduced (Meier and Persson 2001; Nelson 1999b), as shown in table 4.1.

Although, as already mentioned, another similar development system, F, was created at the University ofWaterloo in Canada, that one did not see such wide use. By creating a free development system, even one for a small, local group of programmers, Seal and Thackray helped lay the groundwork for the broader authorship of interactive fiction by individuals. In the 1990s, after commercial interactive fiction had foundered, the rise of this type of development would become central to continued progress in the form, as described in chapter 7. The most important contribution to independently authored interactive fiction in the 1990s was almost certainly Graham Nelson's development of Inform and his authorship of Curses. (Nelson (1999a) notes that Jonathan Partington, who was involved in authoring seven of the fifteen works listed here and was the most active Cambridge IF developer, was "by a curious coincidence" his topology tutor at Cambridge; Nelson was also a Phoenix user.) But before the era of individual authorship outside of academia, a commercial heyday would occur. Just as Zork saw commercial distribution to home computer users by Infocom, many of the previously mentioned works were ported to microcomputers and distributed by one of many companies in the United Kingdom involved with the development and publication of interactive fiction-a company that, in another interesting mathematical tie-in, was named Topologika. The next chapter chronicles the shift of interactive fiction development to places outside the university and the rise of commercial interactive fiction.

TABLE 4.1

Interactive fiction works developed at Cambridge in the Acheron system, with the authors' names and login names (frequently used in speech by Phoenix users) where these were available

Acheron, Jon Thackray, David Seal, and Jonathan R. Partington (JGT1, DJS6, JRP1)

Murdac, Jon Thackray and Jonathan R. Partington (JGT1 and JRP1)

Avon, Jon Thackray and Jonathan R. Partington (JGT1 and JRP1)

BrandX, Jonathan Mestel and Peter Killworth (AJM8 and PDK1), 1979

Hamil, Johnathan R. Partington (JRP1), 1980

Quondam, Rod Underwood (RU10), 1980

Hezarin, Steve Tinney, Alex Ship, and Jon Thackray, 1980

Xeno, Jonathan Mestel (AJM8)

Fyleet, Jonathan R. Partington (JRP1), 1985

Crobe, Jonathan R. Partington (JRP1), 1986

Sangraal, Jonathan Partington (JRP1), 1987

Nidus, Adam Atkinson (AJFA1), 1987

Parc, (JR26)

Xerb, Andrew Lipson

Spycatcher, Jonathan R. Partington (JRP1)

Note: The commercial release of BrandX was called Philosopher's Quest. Some dates are unknown, but the list is sorted into an approximately chronological order; Spycatcher was the last Phoenix game, around 1988-1989.