Chapter 3
Computer Game Fictions

Digital Fictions and the Eliza Effect
Imagine you've checked out your books and walked, through the damp twilight, to the bus shelter across the street. Its fluorescent tubes have flickered on—you can read the schedules and advertisements behind the Plexiglas, as well as the stickers scattered over them.

One rectangular sticker catches your eye. It has the name of no band, the number of no locksmith, the logo of no corporation, and no image of Andre the Giant. It's just a block of text. The first words read, "Why bomb libraries?"

The text is a passage from Implementation, a novel written in small chunks formatted to fit on mailing labels that can be fed through a standard laser printer. Implementation is a sticker novel, in one sense, and also a kind of digital novel. Its authors, Nick Montfort and Scott Rettberg (2004), don't primarily distribute stickers. Instead, they distribute digital files, which others print on stickers and post in provocative locations. Readers sometimes encounter these stickers. But, like most stickers, they tend to be removed pretty quickly from the benches, doors, bathroom stalls, statue plaques, lampposts, bumpers, and other locations where they're placed. Whatever longer are the photographs that people take of Implementation stickers in interesting positions, and it is through these images that most people experience Implementation, the files for these images dwarfing the other contents of the Implementation web site.

Encountering Implementation can be a mysterious experience. Finding a disconnected sticker of text, especially if well-placed, can border on disconcerting. And the images of placed
stickers can offer up their own mysteries: Where is that? What is that? But, despite the importance of digital technology for Implementation, the mystery is never about software. Implementation depends on digital processes—but ones we use every day, such that the processes of file downloading, laser printing, and photo sharing no longer attract our attention.

One might even argue that Implementation is not appropriate to discuss as a “digital” novel but rather one that is distributed and documented digitally. Regardless, in its use of digital processes, Implementation stands as a relatively extreme example of something true of many digital fictions: they avoid the dilemma of the Eliza effect by employing processes that are conceptually simple and familiar, and that are clearly exposed to the audience.

Different fictions approach this general strategy in different ways. Some, in a manner relatively close to Implementation, embed themselves in familiar digital contexts. An email novel, such as Blue Company by Rob Witzig (2001, 2002), depends on processes of email transmission and reading—but, like the processes of laser printing, these have become completely naturalized for many with computer access. Similarly, the familiar link-following functions of web browsers and web servers make it possible to construct fictions from interconnected networks of web pages, as with The Unknown by William Gillespie, Scott Retberg, Dirk Stratton, and Frank Marquardt (1999). Familiar processes are also the primary digital components of hybrid works such as the alternate reality game by Elan Lee, Sean Stewart, Jim Stewartson, and Jane McGonigal (2004) titled I Have Been, which brought players into a fictional world composed of web pages, email messages, phone calls, physical settings and elements, and live performance—but defined its novel processes as rules to be carried out by human participants (rather than by digital computation).

Another approach to digital fiction is more common than this sort of piggybacking on the processes of everyday information life, though. In this more common approach, digital fictions define their own versions of digital media processes in widespread use—or employ versions of these processes defined for use with digital media authoring tools. Such authoring tools range from Adobe’s Flash software (for interactive animations) to game engines (used repeatedly by the same developer or made available for commercial license) to tools identified with particular artistic communities (such as the Storyspace hypertext authoring system).

The most widespread fictions taking this approach are computer games. They tend to avoid the Eliza effect by employing versions of simple processes, familiar to those who play computer games within the same genre, and following conventions to expose the structure and actions of the underlying processes to their audiences. This chapter will look at two particularly well-crafted examples of computer game fiction—Star Wars: Knights of the Old Republic (Falkner, Gilmour, Hudson, et al. 2003), or KatOHR, and Prince of Persia: The Sands of Time (Mechner, Mallat, Desilets et al. 2003), or PoP—as well as two commonly employed operational logics that enable the fictions of computer role-playing games (RPGs). But, to understand all this, it is important to begin with a wider view of RPGs.
Role-Playing Games

Games employ fiction in many ways.¹ The most story-ambitious genre of computer games is probably the computer RPG—a form that traces its roots back to a noncomputer form of gaming: the tabletop RPG (see sidebar: Computer and Tabletop Role-Playing Games).

The first tabletop RPG was *Dungeons & Dragons*, created by Gary Gygax and Dave Arneson and published in 1974 (Mora 2007). It grew out of a tabletop wargaming tradition in which maps were used to represent battlefields and miniature figures representing units or individual combatants were placed on the maps. Players would then move figures and engage in battles by following sets of rules and consulting tables of numbers (the process and data of the system) with a random element at times introduced using dice. *Dungeons & Dragons* departed from this model by suggesting that each player take on a single character, that play sessions connect with one another in an ongoing campaign during which each character would develop, and that the game encompass much more of the characters’ lives (and the fictional world in which they live) than raw combat.²

The result was the birth of a genre that is, when well played, undoubtedly the most successful combination of game and fiction now in existence. RPG systems have been created for a wide variety of fictional settings (some inspired by authors ranging from H. P. Lovecraft to William S. Burroughs), are played both as tabletop games and live performances, and embody a diversity of aesthetic goals (the design goals of Greg Costikyan’s games alone range from reproducing the wacky physical comedy of classic cartoons to evoking the alienation of Brechtian

---

¹ It is the author’s belief that the story-ambitious genre of computer games is the role-playing game.
² The result was the birth of a genre that is, when well played, undoubtedly the most successful combination of game and fiction now in existence. RPG systems have been created for a wide variety of fictional settings (some inspired by authors ranging from H. P. Lovecraft to William S. Burroughs), are played both as tabletop games and live performances, and embody a diversity of aesthetic goals (the design goals of Greg Costikyan’s games alone range from reproducing the wacky physical comedy of classic cartoons to evoking the alienation of Brechtian
Quest Flags

Modern computer RPGs use many of the same technologies and techniques for representing their fictional worlds as other games. A character (or group) moving through an explorable three-dimensional world—with objects to pick up, NPCs to engage, and combat to resolve—is present in first-person shooters (e.g., the Doom series), platformers (e.g., the Prince of Persia series), and RPGs (e.g., the Knights of the Old Republic series). What changes is the emphasis in play. As the name would suggest, first-person shooters are focused on combat as the main form of play. It becomes both the primary challenge and main motivation for players moving through the world. Platformers, on the other hand, make movement itself (sometimes through a puzzling space of platforms of varying shapes and heights) the main challenge and motivation for players. RPGs, however, often work to motivate players to engage in a variety of types of play (e.g., exploration, combat, and intellectual puzzle solving) via character development set in a larger story. In particular, many RPGs give the sense that the story itself is playable by offering the player freedom to roam across a world infused with quests that operate at many scales, can sometimes be completed in different ways, and are often optional or available for partial completion. As each player chooses which quests to accept—as well as how, whether, and when to complete them—this creates a different story structure for each playing. Some of the player's nonquest activities may be directly related to this structure (e.g., taking on tasks in the world that earn money, in order to acquire enough to purchase an item needed for a quest), but the structure also provides one context in which even...
world exploration for its own sake can be situated.\footnote{5}

Despite the variety of experiences that players can have with quests, it is commonly observed by both players and authors that there are a limited number of types of quests—perhaps three, or even one. As Lee Sheldon points out in Character Development and Storytelling for Games, this observation is correct, as far as it goes, given that “the mechanism for tracking progress by the game engine is virtually identical in every quest” (2004, 227). This mechanism is the setting and checking of a collection of small pieces of data—often called “tokens” or “flags,” but sometimes known by the more formal term “variables”—as the player progresses through the world. These data flags represent the state of the world as it relates to quests of varying scope. They are generally checked and set by relatively simple “scripts” that can be edited by game designers and writers (without recompiling the entire code of the game). The state of quest flags is often explicitly presented to players in the form of a personal “journal” or “notebook” that scripts update with helpful reminder texts about the current state of each quest (at the same time that flags are updated). Other quests are, while frequently signaled equally directly, instead organized around the possession and use of game world objects without messages in the journal.

Overall, this approach is notable for its simplicity of structure. In fact, it is so simple that game companies have worked to open its authoring to players who are not software developers. For example, when game developer BioWare published the game Neverwinter Nights (Oster, Holmes, Greig, et al. 2002), it also released the Aurora toolset for creating new spaces, objects, NPCs, and quests.

The goal, in part, was to offer players a rare opportunity in the world of computer RPGs—to play a game master, in a manner somewhat similar to tabletop RPGs, crafting the game experience of other players.\footnote{6} As in many other games, the quests of Neverwinter Nights are presented to the player in the form of a virtual journal.

Using Aurora, the first step in creating a quest is to begin a new category for entries in player journals, using the special purpose Journal Editor tool (shown in figure 3.1). Each entry in a category is designed to let the player know that the appropriate scripts have been activated to set the flags that are necessary for the next stage of the quest.\footnote{7} For example, one could create a journal category with three entries:

\begin{itemize}
\item [0001] Grandfather had a signet ring that belonged to his own great-grandfather. But apparently no one has seen it since it was passed down to him—and it wasn’t mentioned in his will.
\item [0002] Grandma would treasure this ring with the rest of Grandpa’s jewelry. But the carved symbol you’ve seen before at your Uncle Bill’s, so he might be interested in it as more than a momento. Of course, it also looks valuable, and you could use the money you’d get from selling it.
\item [0003] You no longer have the ring.
\end{itemize}

Different events can trigger scripts that change the current state of the player journal. For instance, the first entry above could be added based on a conversation with the player character’s grandmother, or finding and reading a letter sent to the player character’s father. The same script could set a value of “true” for a flag with a name such as lookingForRings. Dialogue trees for NPCs could be different based on the flag, offering the player the option
quest, starting by finding the ring (jumping straight to the second stage) seems unlikely to produce any problem with the flags or journal. Setting lookingForRing to false when it is already false makes no difference, and starting with the second stage of the quest works with the journal entries as written. But, to determine such things, the game author has to carefully think through every possibility—and the more freedom given to the player (as to when, how, and whether to take on and complete quests), the more the events of different quest strands need to be considered in relation to one another.

This, in turn, brings us back to the idea of simplicity. The quest flag structure is a simple one in a variety of ways. It's easy for software developers to implement, it doesn't require much in the way of computational resources (leaving both programming and processing resources free for areas of the game such as graphics, which is something most game companies value), and it can be exposed to the audience quite straightforwardly. Yet given the goal of the audience feeling able to play within the fictional world—not just in areas such as combat but also in terms of the stories being told—the quest flag structure quickly brings game authors to a limit point of complexity, holding back the future development of the form. I'll consider this further in my discussion below of BioWare's widely lauded single-player RPG, KotOR. First, however, a brief discussion of dialogue trees is in order.

**Dialogue Trees**

Tabletop and live-action RPGs create dialogue between characters by having two human beings improvise it. No computer process can substitute for the immediacy

of asking people if they know anything about the ring, leading to clues as to its location. Once the ring is found, a script fired by that event could move the journal to the next entry as well as set lookingForRing to its earlier "false" value, while a flag such as "learningAboutRing" could be set to "true."

If the game authors wish to move all players through the quest in the same way, the ring could be in a locked attic room that isn't accessible until lookingForRing is true. Alternately, the player could have the option of finding the ring before hearing about it in any other way. Quest designs that follow this latter sort of approach (forcing players through as few "gates" as possible) certainly give the player more options for how stories unfold, making the story have some of the same feel of free exploration as the graphical world.

But seeking this feeling of exploration carries a price—one that attention to the shape of processes can lessen. For example, in the case of our hypothetical family ring,
Figure 2.3: A game's dialogue tree. The branches of the dialogue segment that begin "yes, child!" are collapsed in this view, with some player options visible.

with possible replies nested at the next level of indentation, each of which leads to a particular NPC reply (and/or the end of the conversation), which may itself have further player character statements nested at a further indentation level below. For this reason, dialogue trees can be read straightforwardly. Dialogue trees are also rarely pure tree structures. Instead, many branches make connections to other parts of the tree—as above, where both of the possible responses to the grandmother's first question (“What's that on your hand?”) lead to the same place, because of a link from “No, the ring” to “This ring?”.

Given this, understanding how players will experience a segment of a dialogue tree requires both traversing the hierarchical tree structure and following the links that connect the structure's branches.

Another type of odd ordering, that of the two primary tree segments, emerges from common storytelling practice. Dialogue trees check and set the same types of flags employed by quests—and different sets of dialogue are appropriate at different moments. The expanded segment of dialogue above (that begins "<FullName>, you're looking well!") is appropriate for the first time that the player character meets the grandmother, which only happens once, so it is placed at the bottom. The most commonly encountered dialogue is placed at the top, as the default, and only altered (or skipped) if particular flags have certain values. If the grandmother were to have only two possible segments in her dialogue tree, this could be accomplished simply by having a flag called, for example, “firstTimeTalked” that is initially set to true. The default dialogue segment would be skipped whenever firstTimeTalked is true, and entry into the second dialogue segment could trigger a script to set firstTimeTalked to false.

Chris Barzeh, in *Game Writing: Narrative Skills for Videogames*, is generally negative about dialogue trees. He writes:

"Despite the name, dialogue trees are seldom true trees but rather converging and diverging chains of conversation. They can be a nightmare to work with, and the benefits they provide are somewhat minimal. Nonetheless, some players greatly appreciate the illusion that they have control over what their character can say, with the consequence that dialogue trees remain important, especially in dRPG games. (2006, 277)"
RPGs, either accepting or declining quests (and rewards for those quests), defining tense situations or initiating battles, learning more about NPCs (and perhaps unlocking quests related to their personal lives), and changing how NPCs feel about the player character in more emotional ways (ranging from feelings of loyalty and honesty to romantic subplots).11

At a technical level, it is correct to say that all these things boil down to menu selections. But successfully traversing these menus can be a tricky business—one made easier for players who draw on their knowledge of the game’s fictional world. In addition, it is one of the most direct means by which players can express the role they imagine for their character(s)—agreeing or refusing to do things for others, acting out of kindness or mercenary motivation, telling the truth or spreading falsehood, and so on. Building such consequential choices into dialogue trees is also quite simple for authors. In figure 3.2, the lower portion of the window shows how a few menu selections are sufficient to cause a line of dialogue (“Thank you so much”) to also update the player’s journal—in this case, moving the quest in the Grandpa’s Ring category to entry number one. Figure 3.3 shows this taking place in the game.

But, despite the ease of this basic piece of authoring, Bateman is certainly correct that dialogue trees can be a nightmare to work with—as can the larger method of storytelling that they form in combination with quest flags. Further, this might be one of their smaller problems, as I discuss below in the context of a particular example. First, though, a broader look at the logic of these operations is in order.

---

11. Further, as Benjamin Grindell pointed out during the critique peer review of this book, a number of RPGs “require with the ability for even your party members to grow to respect you—often for quite complex reasons which are tied to the past and yet are...”
Quest and Dialogue Logics

The primary operational logic at work in *Eliza* is transformation—each statement from the audience is immediately transformed into *Eliza*'s next statement, using a set of processes in a manner specified by the data of the current script. In computer systems parlance, we might say that *Eliza* conversations are nearly “stateless.” *Eliza* doesn’t model the ongoing state of the conversation, using its script differently depending on prior interactions, except to avoid excessive repetition of the use of certain rules.12

The combination of quest flags and dialogue trees, on the other hand, is in some ways all about state. The quest flag logic is precisely one of milestone-based progression. Quest flag fictions are ordered sets of discrete units. What matters is where the player is located—at which milestone along the path—and this is precisely what is exposed through the mechanism of the player journal. In some cases the beginning of the progression may be skipped and the end may never be reached, but at each moment of play the fiction is at a particular point, among a small number of predefined points arranged in order.

The logic of the dialogue tree, in contrast, is essentially that of the directed graph. Rather than modeling conversation as a set of discrete exchanges, with no context (as in *Eliza*), the dialogue tree always locates the current conversational state at one particular point, among a set of predetermined points, from which navigation is possible to other points via predetermined links. As with the milestones of quest flags, it’s usually impossible to go backward—the graph is directed toward “progress” in the conversation—but it is also usually possible to loop back to the main trunk of the currently available conversation, if occasionally rather circuitously.

In other words, somewhat like the graphical logics of games, the logics of dialogue trees and quest flags are about location in a given space. But while the visual spaces of games are often simulated in a manner that supports almost innumerable possible locations, the milestones of quest flags and graphs of dialogue trees mark out all the possible positions (and transitions between them) ahead of time. This mismatch proves problematic.

An Example: *Star Wars: Knights of the Old Republic*

The Game Developers Choice Awards are the Oscars of the game industry—the award with which members of a creative industry recognize achievements of their own. In 2004, game studio BioWare walked away with three awards that are of particular note for this discussion: Game of the Year, Original Game Character of the Year, and Excellence in Writing. All of these were awarded for BioWare’s RPG *Star Wars: Knights of the Old Republic* (KotOR), which also won game of the year awards from a slew of industry publications. While certainly not the most recent major RPG, it provides a good example of the strengths and weaknesses of the quest flag and dialogue tree logics.

**KotOR’s Successes**

*KotOR* uses quest flags and dialogue trees to reward and sustain engagement with its fictional world; establish patterns that, when altered, produce small moments of surprise and pleasure; and direct the audience’s attention to a series of things that must be accomplished through
Expressive Processing

13. This is generally done quite artfully. Even an exceptionally awkward-looking opening exposition is later given deeper resonances as players progress through the main quest. It is revealed that the characters surrounding the player...

Notes continued at end of this chapter.

play—which, like a magician’s misdirection, keeps audience attention away from larger story developments until the moment they are revealed. A main quest provides the spine of the story. It represents a massive undertaking on the part of the player character to—what else?—intervene in events that will shape the history of the galaxy. In a manner typical for multitask quest design, the desired intervention requires learning the location at which these events are taking place, which requires gathering data from partial maps on several planets, which requires getting to the locations of these partial maps, which in each case can be accomplished in multiple ways, with each approach having multiple steps that may have smaller quests as prerequisites, and so on.

Against this epic backdrop (with its many components, some rather mundane) other, optional quests are presented. Some involve developing the relationships between the main player character and her or his traveling companions. Others involve taking a role in the events happening on particular planets, often with a decidedly interpersonal element (e.g., a wife begs you to find evidence to clear her husband of murder charges, which soon leads to revelations of his infidelity, which you must choose whether to discuss with them). Some are quite small, such as discovering your ship has smuggled cargo aboard, being given the code for the container, and being offered good payment for the goods by a crime syndicate representative. In cases such as this last one, the player need only decide what the characters will do with the one-stage quest—and then enact the decision in the game’s fictional world (travel to the places, interact with the objects, and talk with the appropriate NPCs).

In addition to all these activities, there are also others embedded in the fictional world—ranging from the pleasures of exploration and spatial mastery to “minigames” of card playing and racing. Some of the minigames, in turn, can unlock additional quests and NPC interactions as well as provide resources useful in completing parts of the game to which they are less directly connected.

Given all this, conventional wisdom has it that playing a major RPG takes experienced gamers something like forty hours. But fans of the genre often spend longer, indulging in many optional activities. In addition, many play through the same game multiple times. This is particularly true for games, like KotOR, that provide different gameplay options to characters that approach the world with different ethical stances.

What drives players to spend forty, eighty, or more hours with a computer RPG? If we look only at the quest journal, where the operations of quest flag logics are exposed to the player, the appearance is of a massive indulgence in the pleasures of the to-do list. And certainly there is some of that. But, more centrally, KotOR is constantly providing doses of narrative closure and transition—from the small satisfactions of quest stages to the holistic sense of a planet’s stories, characters, geography, and history that can be developed over one or many playings. Jill Walker Rettberg talks about the pleasure found in learning the quest-based fictions of a place, the “network of fragments, most of which are not necessary to experience the game fully, and yet which cumulate into a rich experience of a storied world” (2007, 310).

At the same time, beyond its narrative pleasures, KotOR is also continually providing other reasons to keep going,
to move one part of things along just one more step. A powerful element of this can be found in the rewards
given with each bit of quest closure. Some rewards are as
simple as cash that can be used to purchase items in the
game,tube reinoclate NPCs, help those in need, and
so on. When brought to fruition, many story elements
also deliver experience points (XP), another conversion
borrowed from tabletop RPGs. As characters accumulate
experience they increase in “level” and become more
capable in the game world. Each new level is achieved at
a particular number of XP, and the drive to hit the next
number is another motivator for the “just one more thing”
mind-set that can keep players at KoOR for hours after
they’d planned to stop for the night.4 It is this quantified
progression of a primary player character that has led
William Hiber (2009) to call the RPG genre a “statistical
hiddengerman.”

And at forty, eighty, or more hours, the extent of
audience engagement with KoOR is certainly more akin
to a thick German novel of personal development than,
for instance, a film (or even a season of television). But,
again, a better analogy is probably with a tabletop RPG
campaign—into which players can easily invest a similar
amount of time. As in many RPG campaigns, KoOR works
to create a sense of flexible story making which is where
exploration and character development. To this end, the
player’s character can visit planets in different orders and
multiple times; quest items can be found at different points
in the quests to which they’re connected; and necessary
items and information are often available in multiple ways.

 Nevertheless, while these things are also true of tabletop
RPG campaigns, in KoOR and other computer RPGs

they must be managed through quest flags and dialogue
trees (rather than human memory, improvisation, and
creativity). As mentioned earlier, this creates difficulties for
game authors.

KoOR’s Troubles

Much of the narrative power of KoOR comes from the
ways that it makes playable structures out of tried-and-true
narratives. A self-contained example of this can be seen in
one of the optional quests on the planet of Dantooin.
relatively early in the game: the feud between the Sandrale
and Maste families. I encountered this quest while
playing the “Platinum Hits” edition of the Xbox version of
KoOR—and it is worth noting that the experience of
those playing other versions might be different.

After being warranted that tensions between the two
prominent families were in danger of overflowing into a
violent civil conflict, my party traveled south to the Sandral
estate and spoke with the patriarch, Nunuk Sandral. He told
us that he felt great sorrow over the disappearance of
the young Maste heir, Shen, but knew nothing about it. He
told us that his own son, Cuss, had been missing for some
time—and speculated the two might have met similar fates
amid the dangers of Dantooin.

Nunuk asked us to show ourselves out. But shortly
after this his daughter, Ralasia, appeared. After I selected
some friendly things for my player character to say to her,
the revealed that her father had in fact kidnapped Shen
Maste. In typically Shakespearean fashion, she and Shen
had fallen in love. She gave my party a key to a side entry
of the Sandral estate, making it possible for us to sneak
in and rescue Shen. Once we reached Shen he refused to
leave without Rahasia—and the result was Shen, Rahasia, and the three members of my KorOR party all coming out the side entrance at once, where we found ourselves confronted by the patriarchs of both families and their band of droids (Figure 3:4).

After some tense dialogue-tree discussions (in which I chose statements supportive of the lovers and designed to diffuse the conflict), Shen and Rahasia ran off to live in the safety of the Jedi enclave, while their fathers just barely held back from igniting a confrontation. Later, while exploring a portion of the planet further north, my party came on the Malale family compound. The guard droid granted us an audience with the patriarch, Ahlan Malale, who had last seen as his son ran off to the enclave with Rahasia. Ahlan proceeded to demand at length that something be done to find his son—outlining his suspicions that the Sandril family had kidnapped Shen.  

He offered us a reward (which sounded more like a bribe) should his son be found. I suspected this inappropriate dialogue tree segment might be active because of a simple flag of the first-time-player variety, so I took my party away and then returned to the compound. But flag structure was apparently organized in a different way. When we returned, Ahlan Malale came out again to demand an investigation into the possible kidnapping of the son he had already seen rescued from kidnapping. This illustrates one type of problem with the quest flag and dialogue tree approach, a type that results in inappropriate events.

The other major type of problem was illustrated shortly, when my party discovered the body of Casus Sandril. An amateur archaeologist, Casus had apparently been killed by wild animals while undertaking a dig in a dangerous area. We immediately went to the Sandril estate with Casus's diary, in order to share it with his worried family. But the estate was shut down entirely, without even the droid on front who had greeted us on the first visit. This second type of problem is visible when the game seemingly arbitrarily shuts off quest possibilities that have the force of narrative drive behind them.

Both types of problems emerge, most commonly, at the juncture between the freely explorable fictional world and the rigid structures of quest flags and dialogue trees. I encountered these problems regularly in my playing of KorOR. Just as the game expected me to visit the Malale estate before the Sandril estate, and produced inappropriate events when I visited in a different order, the same was true on a planetary scale. For example, I visited the home planet of one character who joined my party, John Bindo, later in the game than KorOR's dialogue tree structure expected. As a result, much of the conversation between him and the main player character consisted of him darkly hinting at truths that had already been revealed in a dramatic fashion. Each conversation with Bindo undermined the sense of KorOR having a consistent fictional world. At the same time, the conversation path with another key character, Carth Onasi, was shut off entirely after not being pursued in the expected manner—despite the fact that there was clearly much to discuss.

These problems did not spring from poor work at BioWare. It would be unreasonable, for instance, to expect lead writer Drew Karpyshyn and his team to have written (and sent for voice acting) different versions of Onasi's lines for each possible state of the story. As Chris Crawford has observed, such approaches "are always too
Expressive Processing

Both of you calm down now.
You should try to sort this out together.
Why don't you try listening to your children?

Figure 3.6: Selecting what to say during the confrontation outside the Sundrales Number.

Figure 3.6: After delivering the parental lecture, a journal entry is added, experience points are gained, and side quests are given.

Figure 3.6: The updated journal summarizes the quest and its outcome, including the family confrontation and the players' move to the Jedi Academy.

Figure 3.6: Why are you so certain the Sundrales are our friends? Any information you might have would be helpful. You can't accuse the Sundrales without proof.

Figure 3.6: The unexpected conversation that followed.
much work for the designer and not enough meat for the player” (2004, 126). And it is unlikely that the tools at their disposal offered any more elegant (and tractable) method of addressing the situation. Once the sequence of expected timings between quest flags and dialogue sections was broken it might have made sense to allow the player character to ask personal questions and be brushed off, but this is essentially the same as shutting down conversation, as the Kaor team did.15

Which is to say that problems such as these spring from a poor fit between the simple, brittle structures of quest flags and dialogue trees, and the ambitious, flexible game design they are being used to support. Those of us familiar with the logic of milestone-based progression—from business plans, grant proposals, employee evaluations, and so on—know that any detailed set of milestones will generally meet one of two possible fates. First, as the situation evolves, it can become clear that the milestones will be revised for one or more reasons: the steps may not be what one originally thought, may not happen in the order originally thought, or might need to be divided up differently. Second, an alternative fate is that the milestones themselves become a fetish, irrationally driving behavior in a situation that they clearly no longer reflect. Even in a world simulated as partially as that of Kaor, precreated milestones still become an uneasy fit with the evolving situation. Unfortunately, only the second of these two fates is possible for Kaor’s milestones, which cannot be revised by the system.

Given this situation, the prospects for fiction in games may seem grim (perhaps particularly for multiplayer online games, see sidebar: Individual Fictions in Shared Worlds).

But there is a widely practiced alternative approach to the combination of game and fiction—one that is more successful in important respects.

An Alternative: Prince of Persia: The Sands of Time
At the 2004 Game Developers Choice Awards, Kaor had some competition. Another nominee for Game of the Year, a game that won the awards for Excellence in Game Design and Excellence in Programming, was Prince of Persia: The Sands of Time (PoP).

PoP is a strong example of game fiction in its own right—which is no surprise, given that its main writer and designer was Jordan Mechner, a legend in the game design field for his pioneering games Karateka (1984), the original Prince of Persia (1989), and The Last Express (1997). Mechner is credited with bringing cinematic storytelling to computer games, pushing forward techniques from the realistic modeling of human motion to the integration of an overall story into an action game structure. Mechner is also an accomplished independent filmmaker—writing, directing, and editing films such as Chinon: Ravine: A Los Angeles Story (2005), a documentary about a 1950s community evicted from land they were told would become a public housing project (where they would return to live) only to see Dodger Stadium constructed instead, through a process of greed, hypocrisy, and anticommunist hysteria.16

Bearing this in mind, it should be no surprise that PoP is one of the strongest examples of an entirely different approach to game fiction from that of Kaor.18 Rather than attempt to make the game’s story a playable

16. This is one of the reasons that PoP has received a significant amount of scholarly attention. I am aware of three authors who offer particularly extended engagements. Drew Davidson (2006) provides a detailed, thoughtful reflection on the experience ...
one (a story that changes shape depending on how play proceeds), RP has a linear, cinematic story, with a sense of inevitability artfully fused with a context for player struggle. The struggle, specifically, consists of innovative platforming action and acrobatic combat. For games such as RP, the question is how to make this kind of combination, between audience-controlled gameplay and linear story, most productive.

The typical structure for such combinations, in the computer game industry, is simple alternation. The audience is given moments and spaces of play that, when completed, yield scripted sequences that tell the story. Because this model involves so little connection between story and gameplay, game writers are often employed only at the end of game production. The game's play mechanics, spaces, and even characters are designed before the writer arrives (Buteman, Boon, Buckley, et al. 2003). The writer's task is to create a set of scripted sequences that will tie them together and provide an overall context. The results are frequently unsatisfying and can feel, for good reason, quite arbitrary. Mechmen's games—like those of other designer/writers such as Tim Schafer (the creator of Grim Fandango [1996], Psychonaut [2005], and other well-regarded titles)—instead work to conceive the story and gameplay as integrated entities.

The story of RP begins with a war of aggression fought for reasons of greed. The war's basis includes a massive hourglass containing the Sands of Time. The possession of these sands, however, turns out to be more curse than blessing. When the brief, victorious Persian king seeks to enrich his friend with a present of the Sands, the result instead is to destroy the friend's palace and turn nearly
Expressive Processing

everyone into sand demons. The prince’s dagger, looted from the same palace as the hourglass, is the key used to unleash the sand. Both of the game’s main NPCs—a woman, Farsah, taken prisoner in the looted palace, and the viceroy who betrayed the palace’s owner and incited the unleashing of the sands—suggest that the dagger can also be used to undo things. But the viceroy has removed the hourglass to the highest tower of the ruined palace.

Mechner describes this as the result of simplifying the game’s story as much as possible. In terms of traditional story, it is certainly efficient, with only three characters: a hero, a sidekick/love interest, and a villain. It is also efficient in terms of gameplay. For platforming, it provides a challenge (moving through a ruined palace) and motivation (reaching the hourglass). For interacting with most NPCs, it supplies a motivation for combat and an exploration of why other modes of interaction are not available (they’re sand demons). In other words, it focuses attention on what games do well (movement in space and combat) and offers an explanation for what they don’t do well (interactions with characters) as well as a reason for moving forward. This is already better than most games, but *PoP* goes further in a number of ways.

Centrally, it creates connections between gameplay and story. The most obvious of these is the dagger. It is both a key item in the story and the focus of what is innovative about the gameplay. The dagger can store the Sands of Time, making it possible for the hero to use powers that make movement and combat easier (e.g., turning back time when mistakes are made). The dagger only holds a limited amount of sand, which is replenished by withdrawing it from vanquished sand demons (a requirement lest they rise again). Story and gameplay also connect through plot twists. *PoP*’s story employs twists—including one in which the dagger is lost—that also alter the experience of gameplay. Finally, as Mechner (2007) points out in his essay “The Sands of Time: Crafting a Video Game Story,” *PoP* also works to give as many as possible of the key story moments to the player, in gameplay—so when the prince must face his own father, now a sand demon.

*PoP* also makes choices at the level of process that are significantly different from games like *KeO* or *O*. Obviously, given that there is only one, linear story strand, there is no complicated system of quest flags. There are also no dialogue trees. *PoP* instead uses what is sometimes called an “event-based” dialogue system. In essence, conversation is removed from the playable elements of the game. Instead of players choosing to talk with particular NPCs, certain events trigger conversation and voice-over. In most cases these are not presented as interruptions to gameplay—as happens in many games—but rather layered over it. The prince may think to himself as the player causes him to run down a hallway or swing from a pole. Parah may make comments about the prince’s progress (or failures) in moving through a particularly difficult area, as determined by the player’s actions. Or, in one case, the player may directly elicit a response from Parah—by moving the prince into a first-person camera mode and “staring” at her.

*PoP* also shares one primary tool for NPC interaction with games like *KeO* as well as nearly every type of game that includes NPCs: the finite-state machine (FSM). While on an abstract level all computers could be described as FSMS, in computer game NPC logic FSMS tend to be used a particular way. Each of an NPC’s basic behaviors
is defined as a "state" (e.g., patrol, attack, or retreat) with particular rules for transitioning from the current state to other states. For example, an NPC could be designed to move through a patrol animation (a patrol state) until an enemy is noticed (transition to an attack state) and continue attacking until the enemy is defeated (transition to patrol state) or the NPC's health is below a defined threshold (transition to a retreat state). There might be no rule allowing for direct transition from patrol to retreat—which would make sense generally, but not if the NPC is already badly wounded from previous combat. As with quest flags and dialogue trees, FSMs are simple to explain, easy to implement in software, low in their use of computational resources, and an author's nightmare over a certain level of complexity. Luckily, they're a perfect fit for the exceedingly simple behavior of sand demons, who attack on sight and never retreat, as found in PoP.

The end result, for PoP, is an experience elegantly designed to compensate for the crippling limitations of the processes used to represent story and character in today's computer games. And this opens up the next question.

The Game Fiction Dilemma

Authors of game fictions have worked hard—through conversations such as the quest-tracking journal and linear-driven conversations presented as menus—to avoid the Edge effect. Rather than conceal the operators of their processes, game fiction authors seek to expose them to the audience. But, despite this, game fictions still face a dilemma remarkably similar to that outlined at the end of the previous chapter.28

Both KotOR and PoP take advantage of what games do well—in particular, simulated movement through space and combat. The relatively free-form actions allowed to players in these areas might be seen in parallel with the free-form text composition allowed to both those interacting with Edge's Diet and the journalists involved in Garfinkel's yes/no therapy experiment. The difference, again, is in what changes to the state of the system and influence on future operations can be produced by this interaction.

PoP's fiction, like Garfinkel's experiment, has an extremely narrow range of possible responses to interaction. Either the player's actions successfully move the fiction to the next stage (a progression signaled to the player by the triggering of a scripted sequence) or they don't. The story diagram is, as players put it, "on rails"—and its structure can be completely exposed to the audience by letting them know when they are departing for the next microphonic station. Meanwhile, PoP's NPCs are mostly only available for combat (the sand demons). The major exception (Famili) will occasionally offer a linguistic interjection in response to nonlinguistic actions in the world, but this is another narrow interaction conduit.

KotOR's fiction, on the other hand, while not allowing the free-form textual input of Edge's Diet, does accept many more actions in the world as input into its system of quest flags and dialogue trees (e.g., whether, when, and how to take on quests, take quest actions, speak with NPCs, move between worlds, etc.). Further, as this chapter has shown, such a system can have many more elements to it than the railroad system of games such as PoP (authors can produce huge amounts of data for the quest flag and dialogue tree processes, different subnets and orderings of which can appear with each playing). The result, as with
an Edge conversation, is that the number of potential outcomes is huge. Unfortunately, there is a mismatch between the great variety of situations in which KoOR is expected to perform a fiction and the simple model of fiction and character embodied in KoOR's processes—just as there is a massive mismatch between the complexity of human language to which Edge must respond and its extremely simple model of conversation (as a series of transformations). The result, in both cases, is a tendency toward breakdown that takes a shape determined by the underlying processes. And, as with Edge, the processes of KoOR are of a basically uninteresting shape.

To put it succinctly, the practices of the mainstream game industry present authors of digital fiction with two bad options for going forward. One is to "design around" breakdown, as in Project, and essentially forfeit the processing power of digital media at the level of the fiction. The other option is to attempt to layer a semisensible story—organized as a set of ordered milestone progressions—over a much more flexible game world. This creates a space of play that, if embraced by players, leads to satisfying breakdowns.

In short, the time is ripe for a new approach to game fiction. But just as Doom could not be built on the approach to graphics in Amiga, this will require an approach to fiction and character that is fundamentally different, that is more expressive and flexible than quest flags and dialogue trees. Luckily, this work does not need to begin from scratch. It can start instead by building on a history that has been present as a strand of practice within the artificial intelligence community since the time of Edge. The coming chapters explore this history.

Notes

1. Term games, of course, can hardly be used to engage fiction at all. No one wants a story with their Tetris—or its "natural world" of falling blocks—any more than they want a story with their Pong. Given the sometimes-contentious nature of critical discussion around the relationship between games and fiction, I should perhaps also make it clear that I did not follow any games "as" stories or narrations in a classic narratological sense. Rather, following Espen Aarseth's call for discussion of "narrative games" (2004), I propose here to give careful attention to the specific operations logics of quests and dialogue trees in computer RPGs. For readers interested in a more detailed discussion of games and narratology, I suggest chapter 6 of Auster's Story (von-Liern-Lebre (2006)).

2. As tabletop RPGs have come to encompass much broader areas of interaction, different groups (and game groups) have diverged in their treatment of elements. One movement is toward a generalization of the statistical methods used for war game combat. They have been employed for many additional elements of character progression and play. The other movement is toward an emphasis on creative expression that may resist quantification, or at least be decided by the logics of character and fiction rather than chance, leading to "narrated" RPGs. Players who prefer each emphasis are sometimes called "roll players" and "tale players," respectively.

3. The most famous Llewellyn-inspired game is Call of Cthulhu (Petersen 2001), with Dune the Edge (Taylor 1992) in a game inspired by both Lovecraft and Philip K. Dick. At the same time, as Aarseth reminds us in the blog-based peer review of this book, some RPG systems—such as GURPS (Jones 1986)—are designed to be "generic," rather than specific to a specific fictional world or even a specific type of fictional world. Finally, the games of CoC's authors that are referenced are Trop (1994) and Beastly Acts (1995), respectively. Together with the innovative RPGs: The Extraordinary Adventures of Birdie McCalla by James Walks (1999) and Applejack by Tom Yman (1999), Beastly Acts is reprinted in Second Person (2005), a book edited with Phil Harris.

4. Though many massively multiplayer online play groups, including those in which I have participated, include local-to-home interaction among some players, and there are a variety of other similar tools that combine elements of in-person and computer RPG play, for example, during the online peer review of this book, Sarah Tease showed interest in the web site of Octosian Portal and Grexman (http://www. octosianportal.com and http://www.grexman.com), the first of these is an independently produced web site management system for RPG players that focuses specifically on "character" elements, providing a blog for chronicling adventures and a wiki (which it is to the classes for the party and game master) for keeping track of the campaign world. Grexman, on the other hand, is a site developed by publisher Atlas for online play, social networking, and general reading about games. Today's comment reminded me that we should also consider hybrids such as the Living Grayaskets campaign & Dragon's campaign of the WCO-supported RPGA. As the web site explains:

The Living Grayaskets campaign is an Internet-based game played out in regional events throughout the real world. The conflict of the Villagers in the real world is divided into several locations and political zones. These zones are mapped onto sections of the real world. Your real world location determines the different home regions for you and your characters in the game world.

When you travel to different areas of the real world, your character journeys with you to the corresponding locations of the Villagers. Each region has a special flavor and layout apart from other regions, allowing you to immerse yourself in the respects of your home region or join the fun from other regions. (Origin of the Coast 2005)
Expressive Processing

5. I will attempt to derive various types of thematic issues that might influence the development of a new genre of interactive fiction games.

6. In addition to the literary and dramatic elements, interactive fiction games can also explore complex social issues and promote critical thinking skills.

7. By engaging players in narrative-driven experiences, interactive fiction games offer a unique opportunity for players to immerse themselves in the world of the story.

8. The potential for interactive fiction games to transcend traditional narrative structures is evident in the way they allow players to make choices that affect the outcome of the story.

9. Interactive fiction games can serve as a platform for social and political commentary, offering players a chance to explore and discuss complex issues.

10. The recent trend towards more immersive and interactive storytelling in movies and television suggests that the potential of interactive fiction games is far from exhausted.

11. As technology advances, the possibilities for interactive fiction games continue to expand, offering new avenues for creative expression and narrative exploration.

12. And of course, interactive fiction games also have the potential to inspire new forms of creative work, from literature to film and beyond.

Computer Game Fictions

13. This is generally done quite skillfully. Even if the story is a bit too obvious, it is worth noting that many video games are capable of evoking profound emotional responses in players.

14. In the end, interactive fiction games offer a unique opportunity for creative expression and narrative exploration, providing a platform for players to engage with the story on a deeper level.

15. The success of interactive fiction games can also serve as a catalyst for further research and development in the field of digital storytelling.

16. As more developers explore these new possibilities, we can expect to see a wide range of innovative and engaging interactive fiction games in the future.
technology—he's just been talking about the horrible results of which we'd just seen.

17. As its writing, Miecznik is also involved in little storytelling form another angle—as screenwriter and executive producer of the upcoming Walt Disney Pictures Jersey Hadcock's feature film "Prince of Persia: The Sands of Time," directed by Mike Newell and starring Jake Gyllenhaal, Gemma Arterton, Sir Ben Kingsley, and Alfred Molina.

18. This is one of the reasons that PoP has received a significant amount of attention, yet awareness of these authors who offer particularly compelling engagements. Drew Bowden (2000) provides a detailed, thoughtful reflection on the experience of playing through the game—whether or not to pay attention to the relationship between story and play. Barry Adams (2007) offers a reflection on depth, time, and narrative (chiefly control of the game, but important to games in general) in his chapter on PoP for Videogame Player: Text. Jason Rhoades (2009, 2008) has written extensively both on the way PoP's interface establishes and controls the player's point of view and, in his review of PoP, on how game fiction must be understood dually through technology and audience. The fiction-and-narrative interface PoP operates in a manner that concerns narrative and design development of the experience.

19. These scripted sequences can be understood as "cutscenes" (essentially, computer animation files played by the game engine), in-engine cutscenes (scripted scenes rendered by the game engine, rather than another narrative system, but still removed from player control). They can also serve as brief, meaningful breaks in the main narrative.

20. The parallels with U.S. foreign policy at the time of the game's production, before and during the 2003 invasion of Iraq, soon began to seem prophetic. They are also unlikely to be accidental, given Miecznik's description of an "anti-war theme" that "underlies the whole tale" (2007, 115). It is important to be reminded that the southernmost of the United States was yielded a new tactical base.

21. I am using the words game fiction generically, to refer to fiction within games. In More Than a Game, Barry Adams uses the words game fiction to refer to those games he sees as "having a central narrative that drives the story and the player, rather than serving as an extension of the game's mechanics; a story that is not just a story, but a part of the gameplay that drives the player forward through the game."

What, then, is game fiction? In short, the term is intended to describe a category of games that draw upon and use narrative strategies to create, maintain, and lead the player through a fictional environment in order to achieve a narrative and ludic goal. . . . (Game fiction is a complex, self-reflexive, and self-aware narrative, and its primary goal is one of simulation.) Some games are not limited to a single medium, although a game's particular narrative—that is, its depiction of a game world, as well as its visual and/or mechanical characteristics—should also include a game board, cover, or even the imagination—often reveals much about the game fiction in question.

Birth: Adams and Rhoades participated in the blog-based peer review of this book—and both seem fine with the writing game fiction generically, while still using the words as a more specific term in their own work.