Basics of Game Design

Michael E. Moore

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business
AN A K PETERS BOOK

A K Peters/CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Printed in the United States of America on acid-free paper
10 9 8 7 6 5 4 3 2 1

International Standard Book Number: 978-1-56881-433-9 (Paperback)

This book contains information obtained from authentic and highly regarded sources. Reasonable efforts have been made to publish reliable data and information, but the author and publisher cannot assume responsibility for the validity of all materials or the consequences of their use. The authors and publishers have attempted to trace the copyright holders of all material reproduced in this publication and apologize to copyright holders if permission to publish in this form has not been obtained. If any copyright material has not been acknowledged please write and let us know so we may rectify in any future reprint.

Library of Congress Cataloging-in-Publication Data

Moore, Michael E.
Basics of game design / Michael Moore.
p. cm.
ISBN 978-1-56881-433-9 (pbk. : alk. paper)
1. Computer games--Design. 2. Computer games--Programming. 3. Video games--Design. I. Title.

QA76.76.C672M6158 2011
794.8'1536--dc22 2010053534

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the A K Peters Web site at
http://www.akpeters.com

CONTENTS

CHAPTER 10

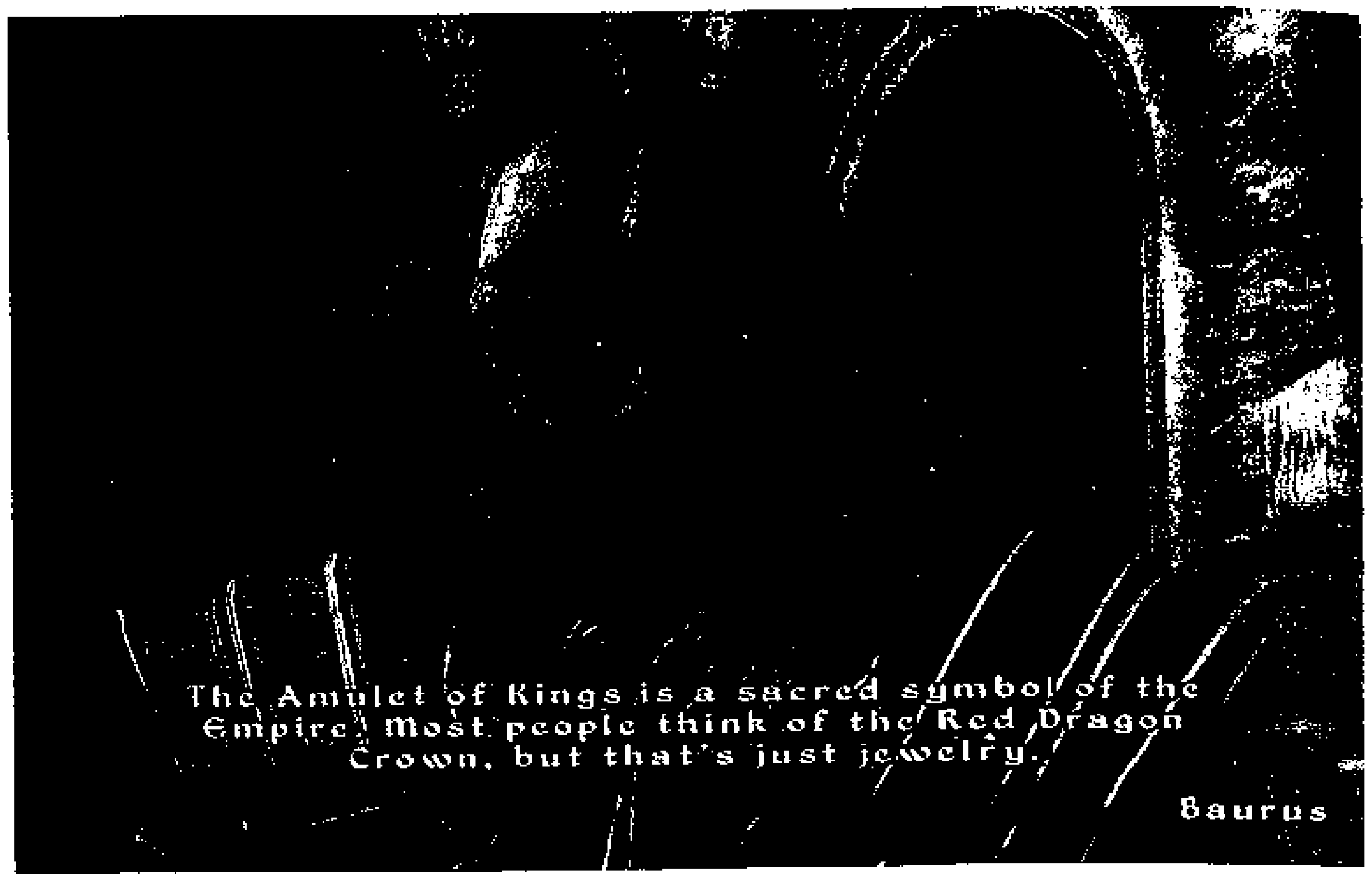
STORYTELLING IN GAMES

Until recently, the difference between movies and games was pretty dramatic. Games were about players performing a few limited actions to achieve a goal or series of goals while movies told stories of characters involved in some conflict that was resolved by the end. Early arcade and video games had little room for the elements of stories—interesting characters, well-developed conflict between them, dramatic tension leading up to a big climax, and then the denouement as the story wound down and the characters went about their lives. Movies also featured lots of dialogue, lots of action, and lots of music, all of which early electronic games lacked.

As the technology for games developed in the 1980s and the storage capacity for game platforms increased, real stories began to appear in games where the player took on the role of the protagonist whose actions and dialogue were controlled by the central processing unit (CPU). When 3D graphics became the norm in the 1990s, game worlds became much larger and more visually interesting, and using CDs for game data storage allowed even more dialogue, music, and sound effects. Nowadays, the border between movies and games is blurring, and in the future the two could possibly merge in interactive movies where viewers make their own decisions about what paths to follow through the story.

Stories in Games

While many games aren't interested in telling stories, others are anxious to narrate grand epics of massive scope in huge worlds populated with a large cast of characters. Puzzle games usually have no stories, although a few such as Interplay's *The Lost Vikings* and GT



The Amulet of Kings is a sacred symbol of the Empire. Most people think of the Red Dragon Crown, but that's just jewelry.

Baurus

Figure 10.1. *This is part of the introductory text that kicks off the main story in Bethesda Softworks'* The Elder Scrolls IV: Oblivion.

Interactive's *Oddworld: Abe's Oddysee* do have background stories that give the player reasons for solving the puzzles. Likewise, arcade games generally do not include stories, save a few notable exceptions such as Cinematronics' *Dragon Lair* and Atari Games' *Gauntlet*. The early text-based adventure games like Infocom's *Zork* series had minimal storylines, although as the genre grew, newer adventure games developed more complex stories with more interesting characters. Early action games like Nintendo's *Super Mario Bros.* and *The Legend of Zelda* have minimal stories, but as they too grew larger in scope, their stories also expanded and became more complicated.

Role-playing games usually have the largest stories where the hero undertakes a series of quests to defeat the evil machinations of the master villain (Figure 10.1). These stories usually involve the player visiting many different locations and interacting with many different characters both good and evil along the way. The stories are often structured like a huge novel divided into books and chapters, with each book involving the resolution of a central sub-quest and smaller quests at multiple locations acting as chapters of the book.

Simulation games and wargames will sometimes include stories as well, although if a wargame is based on a real battle, the story is replaced by historical facts. Real-time strategy games are often set in fantasy or science fiction worlds and tell of nations competing against one another for dominance. In such games, there is less emphasis on individual characters since they focus on armies in conflict. Where the scale,of the game is tactical (as in an FPS game) and focuses on the exploits of a few soldiers, as in Electronic Arts' *Medal of Honor* and id Software's *Return to Castle Wolfenstein*, the exploits of the soldiers are framed within a larger overarching story.

Advantages of Stories in Games

There are definite advantages of having a story in a game. First, it gives an overall structure and meaning to the game action, and provides an Aristotelian beginning, middle, and end to the action. Games without stories can end unexpectedly at any time, and the player never senses the action building towards a climax. Second, a story provides motivation for players, a reason to be running around the game world and destroying everything in their path. As long as the climax is still ahead of them, players know there is more to come, and if they've liked what they've experienced so far, they'll want to continue experiencing even more. Third, taking on the identity of a character in the game world helps the player identify more closely with the conflict that drives the action. Players feel they have a personal stake in the outcome of the game if their characters get killed while performing heroic actions. They can become so immersed in the game world they lose track of time in the real world.

Fourth, if the designers have created interesting characters, a substantial game world and a compelling story—in other words, a strong intellectual property—the game can breed multiple add-ons and sequels and make the developers a lot more money. Finally, this intellectual property can draw the interest of other industries, to make both accessories like toys and action figures and also movie and television series tie-ins. The financial reasons for including a story in a game can be as strong as the players' motivation reasons.

Disadvantages of Stories in Games

There are, of course, disadvantages of using stories in game. It takes time to figure out the plot for a story that can last through 20 hours of play and often much more. The plot can get so convoluted that players lose track of why they are undertaking the quests they have been given. Second, there is a danger that game play will take second place to storytelling, and some cut-scenes, the animated sequences where much of the plot of a game is explained, can go on and on with the player growing more and more impatient to get back to the fun of stabbing and killing. Occasionally, some of the most exciting action in a game occurs in these cut-scenes, and players can be disappointed if they can only watch events occur without being to interact with the changes in the game world. Third, creating the assets to represent the large cast of characters, multiple locations, and multitude of items is expensive and takes time since artists have to create everything from scratch. In addition, most games include voiceovers where actors record the dialogue of the many characters, and preparing for the recording sessions is expensive and can require several callback sessions for last minute changes.

Fourth, it takes considerable time and testing to check for any dangling plot elements, and the more quests and sub-quests there are in the plot, the longer the debugging process takes. Finally, there is the danger that if development takes too long, the team will be forced to make last minute changes to the plot, perhaps having to cut out whole subplots and truncate the storyline. These changes require that expensive art and audio assets be dropped, new dialogue be written to explain holes in the plotline, more voiceover sessions be made, and additional testing be done to ensure that the changes work.

Problems with Game Stories

While games have stories, they have been criticized for the weakness of the stories. There is a certain sameness to many role-playing games, especially those set in fantasy worlds. These games rely heavily on the same literary tropes (i.e., clichés) as their paper RPG ancestors, which in turn drew heavily from the fiction of J.R.R. Tolkien's epic *The Lord of the Rings*. In Tolkien's novels, the world of Middle-Earth is replete with humans, elves, dwarves (well, at least one last remnant of the proud race), and orcs, which also have been incorporated as characters and monsters in most role-playing games. It sometimes seems that designers are more interested in coming up with variations for the magic and combat systems in electronic RPGs than developing new and interesting fantasy worlds.

Weak Characters

Another criticism about game stories is the weakness of the characters. This weakness comes both from poor writing and poor acting.

Until relatively recently, the stories in games were written by the designers who created the game world and the beings in it, but nowadays professional writers are often brought in to help punch up the story and create more interesting dialogue. Some of the dialogue found in old video games is absolutely appalling, an attempt to mix middle English with modern idioms and an occasional out-of-place joke or two. The stories in these older games were also poorly structured, and the plots felt either overly melodramatic or completely non-dramatic. The hero's objective might be painfully obvious, as in rescuing a kidnapped princess from the master villain or in taking simple vengeance against the blackguards who killed his family. Recent games have started to take on more interesting plots where things change over time and the hero's motivation also changes, possibly from mere revenge to having to sacrifice himself to save the world. Still, many games want to be epic in scope where the hero has to save the land, the world, and even the universe from destruction at the hand of the master villain who plans to unleash some nameless horror upon the world...and yadda yadda yadda.

Game characters also face weak characterization because of the limitations imposed on them by the game mechanics. There are only so many actions a character can perform in a game because there are only so many controls the player can master and remember during play. The most common action of characters in games is combat, which can take up a huge portion of the active game play (as opposed to passively exploring the game world). Defining a character by his or her abilities in combat is very limiting, even if those abilities are awesome to players. As a result, the characters in games seem more caricatures than real people.

Too Much Information

To understand what the player is expected to do, there can be a goodly amount of exposition that has to be given before the real play begins. In games the situation is even worse because the player not only needs to understand the backstory of the game—what has happened up to the point where the game begins—but also has to learn the controls. As a result, the early part of a game can have a lot of explanation both about the game world and characters and about how the game mechanics work. In RPGs, in particular, the evil plans of the master villain have to be revealed in enough detail to give the player a sense of the ultimate goal of the story and what to do next. Then, as the player explores the world, she gathers allies who have to explain why they want to join in the quest. Usually, there are henchmen working for the main villain that the player faces before the final confrontation with the main villain, and their machinations and purposes for acting also need justification. As a result, RPGs are filled with extensive story information as well as details on how to play the game.

Even though a game can have a complex storyline, the game play is fairly simple and straightforward: Explore, fight, dicker, and fight some more, using controls that should be easy to learn lest the players give up in frustration. To keep things interesting for the player, the game offers new areas of explore and new monsters to fight, but eventually the game can drag down. Unlike a movie that lasts a couple of hours or a book that might take several days to finish, a game usually lasts ten hours or longer, and in the case of RPGs (especially MMORPGs), they can last for 40–100 hours or more. There are more places to visit and more NPCs and more monsters and more fights in games than in any other media.

Despite being larger than other forms of entertainment, RPGs and other games face the problem of endless repetitiveness, forcing players to do the same actions over and over again hour after hour. Where a movie can build constantly to an emotional climax, games stretch out the action for so long that players feel less emotional involvement in progressing the story and more excitement in finding new items and entities and exploring new places. When the final confrontation with the main villain comes at the end, players are often relieved—not so much with concluding the dramatic confrontation with the antagonist but in knowing that the game is close to being finished, barring the long, long fight with the villain that caps the action. RPGs often feel like long-distance marathons where only the strongest-willed survive to the end.

Structuring Stories in Games

Many people trying to get into the game industry have the mistaken belief that the designer's primary task is to come up with an interesting story on which to hang the game. While story is an important element of design, it is not the overriding one. Coming up with interesting game mechanics is far more important, and a game with a mediocre story can sell well as long as players become totally immersed in game play. A good story can make the game characters more interesting, but the real story being told during a game is the player's. A player's emotions are more invested in how cleverly he or she handles the problems presented by the game mechanics than in the character's dramatic problems. Players don't much care for other NPCs telling them what emotions they should be feeling or nagging them to hurry up and do something to stop the villain. In games, players have much less emotional attachment to the character they control than in how well they perform.

Designers do need to understand the basics of storytelling, of course, and they should take classes in creative writing at college if at all possible. There is a craft to story construction that can be learned by anyone, even though only a few learn to tell stories compelling enough to sell...or create games around. There are many sources available online and in bookstores for learning the craft of storytelling, so this chapter will look at structuring stories for games instead of teaching the basics of plotting.

Quest Structure

As described in the previous chapter, one approach to structuring the story in a game is to use quests and sub-quests. The primary quest involves stopping the major villain from completing a master plan of world conquest or whatever. To complete this quest might require that the main character fulfill a number of conditions before being able to undertake the final confrontation—for example, getting the special weapon that alone can slay the villain or acquiring a set of keys or runes or other objects that open the final path to the villain's lair. The series of actions are sub-quests with specific goals the character must achieve to continue. The sub-quests can involve stopping the minions of the main villain or doing favors for certain personages who then give the character some object needed to complete the main quest.

The advantage of such a quest structure is that the goals are all fairly clear-cut and the story itself is scalable. The designer can create a simple flow chart outlining the actions needed to complete a given goal to determine if a quest is too complex or too linear (see Figure 10.2 on page 270). If it turns out the scope the game is too ambitious and needs to be cut back, some of the goals can be removed or shortened. Likewise, if the game is too short, additional sub-quests can be added to give the player more to do. If the player controls a party instead of a single character, each member of the party can have his

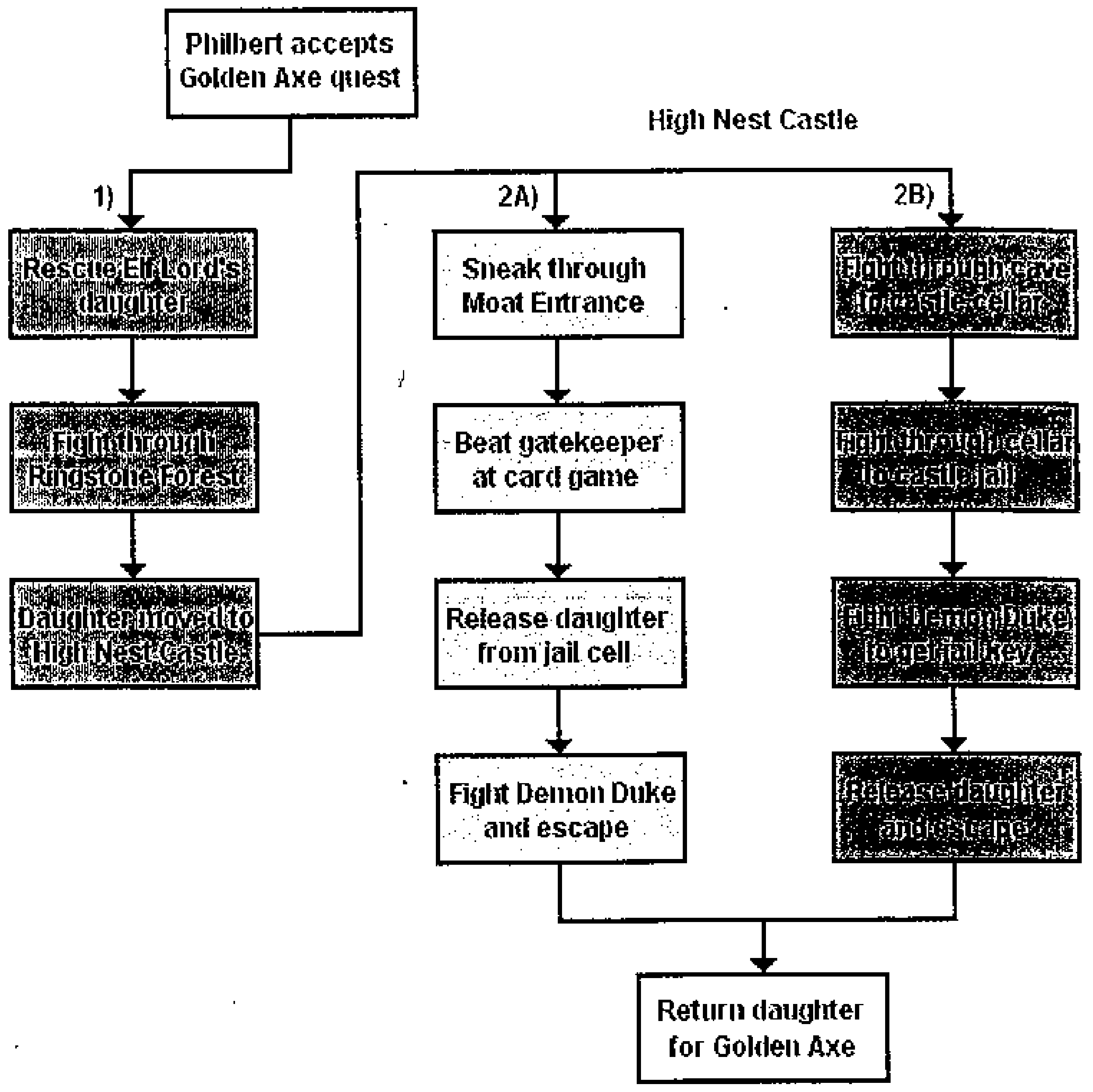

Philbert accepts Golden Axe quest
High Nest Castle
1)
2A)
2B)
Rescue Elf Lord's daughter
Fight through Ringstone Forest
Daughter moved to High Nest Castle
Sneak through Moat Entrance
Beat gatekeeper at card game
Release daughter from jail cell
Fight Demon Duke and escape
Fight through cave to castle cellar
Fight through cellar to castle jail
Fight Demon Duke to get jail key
Release daughter and escape
Return daughter for Golden Axe

Figure 10.2. *A flowchart is a useful tool for helping the design team map out the plot elements of a game.*

or her own story arch with sub-quests leading to a climax that affects the main story's quest.

Note that the story structure described above works well for a game that has a definite ending. If the project is an MMORPG, the main difference would be that there is no one overarching quests. Instead, the online game contains a series of sub-quests that can be extensive but don't substantially change the conditions in the game world. MMORPGs are persistent worlds, where things don't change much over time. The reason for this stasis is that beginning players can join the game at any time, and if some of the more challenging and rewarding quests are not available because that part of the story has been finished, the newcomers feel cheated. If some NPC has a quest to find a magic sword, that quest should be available to all players at all times, even if the NPC initiating the quest winds up with tens of thousands of copies of the magic sword.

One example of a quest structure in a game can be seen in BioWare's *Dragon Age: Origins*. The game story starts with an introductory sequence in which the player selects from six available characters and lives that character's early history (also learning the main game mechanics) until being inducted into a group called the Grey Wardens, a group dedicated to destroying the evil beings that are overrunning the world. The main story then kicks in with the player surviving the treachery of a noble who betrays his king at an important battle and the ensuing slaughter that follows. The player takes on the leadership of the Grey Wardens and is tasked with four major sub-quests: getting humans, elves, dwarves, and a council of mages to join in the final battle against their common enemy. Before the final battle can begin, however, the player has to deal with the treacherous noble. The climax is the battle with the great dragon that leads the enemy forces.

There is a problem, of course, with using quests as the backbone of a game story. Eventually, a sensation of sameness develops as the player goes from one sub-quest to the next, and there isn't always a sense of increasing tension over time nor an urgency that builds to the climax. There is, however, a sense of reward as the player completes various quests and acquires new armament and objects of power to use against the next set of enemies. As long as rewards are doled out on a regular basis, the players are content and willing to continue questing towards the final show down with the villain.

Zigzag Structure

A quest-based story structure allows players to know things their characters don't, primarily the master villain's master plan. Occasionally, a cut-scene animation pops up and shows what the villain is currently up to, thus providing a sense of urgency that the villain's plan is coming to fruition. A different approach to telling the story is to keep the player as much in the dark about what is happening in the world as the character is. In a quest-based story structure, the players understand throughout the game what they have to accomplish before they can confront the major villain—for example, collecting all the keys or runes or arms that enable them to meet and beat the villain. If players are given only limited information and set on one quest at a time, the designer-storyteller can then change directions abruptly at any time during the story and even change the goal of the quests as the need arises. This approach is like a zigzag, where the player starts out in one direction to achieve a goal and then, as circumstances change dramatically, us forced to change directions because the goal has changed. This reversal of purpose can happen several times during the game, with each major change in the plot providing new quests and new objectives.

As an example, say that the hero, Sir Clyde, learns his fiancée, Lady Genteel, has been kidnapped by the evil wizard Gort. Clyde's first goal is to track down Gort's whereabouts to rescue the fair lady. Just as Clyde breaks into Gort's dungeon to confront the villain, a giant Ruby Dragon arrives and kidnaps Lady Genteel. Much to Clyde's surprise, he learns from Gort that Lady Genteel is actually a "daughter of the sun" and is doomed to be sacrificed to bring back the red dragon race. Putting their mutual animosity aside, Clyde and Gort team up to track down the Ruby Dragon. No sooner do they fight their way into the dragon's mountain lair then they see the sun god Heliyos abscond Lady Genteel and flee with her to the highest heavens to become his wife. The Ruby Dragon is now willing to join up with Clyde and Gort to rescue the lady so as to save dragon-kind, and so the humans leap on the dragon's back to fight their way to the gates of the Third Heaven where they find that all the gods at war and Lady Genteel is again missing, having been abducted by the Dark Lord of the Underworld... And so it goes.

One advantage of a zigzag story structure is that the player is never quite sure of what will happen next and whether or not the next destination will be the final one. This kind of story structure has been used in many Japanese RPGs, and it provides players a different kind of emotional involvement in the game action. Because one never quite knows what will happen next, the designer-storyteller can spring major surprises on the player with sudden reversals of fortune and other "I didn't expect *that*" variations.

A major drawback to the zigzag story approach in that the player usually has to be kept on a tight leash at all time and given little freedom to explore the game world. Each new area of the game world the player visits has a limited number of discoveries to be made, and the player is restricted from wandering out of the current location to explore other new areas. Going back to explore areas of the world that have already been visited generally does not provide much reward except the chance to buff up the characters. There are usually fewer sub-quests available because of the travel restrictions or the sub-quests are all localized.

One way to restrict movement is to allow travel between major locations only by using a strategic world map and forbidding entry to new locations until the current chapter of the story is finished. It is possible by the end of the game to open up the game world and allow the player access to all locations, but by then most of the new discoveries have already been revealed so there is little novelty in visiting these same areas again.

Mission Structure

Games with simpler stories, such as real-time strategy games and even shooter games, can tell their narratives in a more linear fashion by using missions. Each mission acts like a self-contained chapter in a story and advances the story a bit. In many cases, the mission has nothing to do with telling the story, but cut-scenes appear between missions to advance the plot and set up the next mission. Each mission normally takes place on a single map or level, which can have its own narrative structure, as described in the next chapter. Once a mission is completed, the player moves on to the next mission and another part of the story may unfold through a cut-scene. To add variety, the player is sometimes allowed to select the next map to visit or the mission to undertake from several available, but the storyline usually does not advance until all maps or missions are completed in a strict linear fashion.

One advantage to this structure is that the player has a pretty good idea of when the story will end. The pace of the story is continuous and there are seldom any dramatic turns of fortune to worry about. The characters can be flat and two-dimensional, and there is little time given to character development (although there can be leveling to improve characters). A drawback to this structure is that if players get stuck trying to complete a map or mission, they may decide to give up playing the game if they can't figure out how to get to the end.

One way to make a mission-based story more interesting is to force the player to switch sides once in a while, as in Bungie's first-person shooter *Halo 2* where the player switches between playing the human Master Chief and the alien Arbiter. A similar approach was used in Sony's role-playing game *Arc the Lad: Twilight of the Spirits*, which switches between the fraternal brothers Kharg of the humans and Darc of the Deimos as they seek to destroy each other's race. Seeing the same situation from multiple viewpoints adds depth to the story and allows the designer to make good use of dramatic irony.

Hero's Journey Structure

One approach to building a story for a game is to use the monomyth of the Hero's Journey, as described in Joseph Campbell's 1949 book *The Hero with a Thousand Faces*. After studying myths of many cultures in various times and areas of the world, Campbell found a pattern in many of them, which he describes in the book's introduction as "A hero ventures forth from the world of common day into a region of supernatural wonder: fabulous forces are there encountered and a decisive victory is won: the hero comes back from this mysterious adventure with the power to bestow boons on his fellow man."

The 17 Steps of Joseph Campbell's "A Hero's Journey"

Departure

The Call to Adventure
The hero is warned that his or her life is going to change.

Refusal of the Call
The hero initially refuses the call.

Supernatural Aid
A magical guide appears to prepare the hero for the journey.

Crossing the Threshold
The hero leaves the known world for the unknown.

The Belly of the Whale
The hero's lowpoint, but also the transition from the old life to the new.

Initiation

The Road of Trials
Series of tests the hero undergoes early in the transition.

The Meeting with the Goddess
The hero experiences power of love (internally or externally).

Woman as the Temptress
The hero is tempted to stray from spiritual journey by earthly desires.

Atonement with the Father
Central point of journey where hero meets powerful being that muse be defeated to continue.

Apotheosis
A period of repose for the hero before the return trip.

The Ultimate Boon
The hero's achievement of the quest goal, often transcendent in nature for the hero.

Return

Refusal of the Return
The hero is reluctant to leave and return to normal.

The Magic Flight
The hero begins the journey back with the quest goal.

Rescue from Without
Guides and rescuers assist the hero to start of continue along the return journey to everyday life.

The Crossing of the Return Threshold
The difficult return to everyday life while retaining wisdom learned during the journey.

Master of the Two Worlds
The hero learns to balance the material and spiritual in the everyday world.

Freedom to Live
The hero can now live in peace, free from the fear of death.

Figure 10.3. *The 17 stages of the hero's journey, based on Joseph Campbell's* The Hero with a Thousand Faces, *can provide a solid structure for a game story.*

Campbell talks about the seventeen stages of the monomyth, starting off with the call of duty that leads the hero to head out from the known world into the unknown world. The hero is often guided initially by some supernatural helper that helps him (or her) overcome his initial insecurity and cross the first threshold into the unknown. The hero then undergoes a series of trials and sometimes encounters death before crossing back over the threshold into the known world as a master of both worlds.

The structure proposed by Campbell (Figure 10.3) is very useful when thinking about how the main character (especially in role-playing games) goes from a position of great weakness to one of great strength and resolve, gaining the power to overcome any obstacle, even death, in the long journey. The concept of a helper is also useful in that the helper only can perform a number of functions: explaining game play, advising about what to do next, offering tidbits about the game world setting and story, and even assisting in combat and other game actions. There are problems with trying to stick too slavishly to Campbell's structure since some of the stages—for example, The Meeting with the Goddess, Woman as Temptress, and Atonement with the Father—might slow down the pace of the story. Still, Campbell's book is a goldmine of ideas for designers and should be required reading for anyone interested in creating story-based games.

Open World Structure

Many role-playing games have experimented with a more open-ended structure for telling stories that allows the player to freely roam around the game world. This approach is sometimes referred to "sandbox" style because it feels like the game world is one large sandbox that the player can explore at will. The story line in such games is often episodic and nonlinear, and often the player has multiple ways to complete the game. This structure caught on with Rockstar Games' release of *Grand Theft Auto III* in 2001. It has been used in Bethesda Softworks in several role-playing games, including *The Elder Scrolls* series, *Fallout 3*, and *Fallout New Vegas*.

In these games, there is still an overarching plot that drives the action of the game, but events are broken into smaller quests and sub-quests for players to undertake in whatever order they desire. A player might visit a location in the story that is tied to the main plot but not trigger the plot because other objectives have to be completed first. For example, the player might visit a monastery where the head abbot lives, and that abbot might trigger a major portion of the story. However, the player discovers that the abbot is not available at the moment or hides behind a door the player cannot yet open. Once the player completes other portions of the overall main quest, the conditions are now right for the abbot to appear and continue the storyline.

Even though players might feel they can go anywhere in the game world, they can trigger new plot elements only in a given order. However, there are usually many subplots for players to find all around the world, and so they remain continually challenged until they have fulfilled the conditions to start the next part of the main story. Designers have to be careful about structuring the storyline, assigning plot triggers to locations, and ensuring that variables which control trigger points are changed correctly as the player completes subplots and sections of the main plot.

It is possible to channel the players to explore the game world in a desired way by placing obstacles in their path to bar entry to certain areas until the right moment. The most obvious obstacles are locked doors and barred gates that can't be opened and having important NPCs temporarily be absent from areas. Another approach is place tough enemies in the path of the players. They can try to force their way past these enemies, but eventually they get overwhelmed. Later on, when the players are stronger and have better weapons, they can defeat the enemies and explore areas of the world previously off limits to them. In many games, the main plot is relatively short, making it easier to playtest and modify trigger conditions as necessary. The richness of these games comes from having so many different locations to explore and subplots to resolve.

Linking Plot to Game Play

In a game where the story is external to game play, such as in real-time strategy games where the story works as a framework for holding the missions together, the story does not link in directly to game play. Usually, no cut-scenes appear during a mission to give more exposition about the characters or to show what is happening elsewhere in the world, unlike role-playing and first-person shooter games, which occasionally interrupt the game play for important cut-scenes that further the narrative. In games where the story acts as a framework, the whole plot can be worked out without worrying about the status of the characters controlled by the player or what plot elements they have resolved or not resolved at the start of each mission. All the designer has to do is use placeholders for the missions while figuring out the plot details. For example, in a futuristic wargame starts dealing with the invasion of Earth by alien forces, the designer would work up the initial exposition about why the invasion occurs, the aliens' plans, the current state of the player's forces, and the goal of first mission. The placeholder for the first mission simply states that the player's goal is defeating a squad of alien soldiers and capturing some of their weapons for research. The details of the mission do not have to be worked out at this point because they don't directly affect the story. The player either completes the mission or fails, and the story doesn't continue until the player wins.

In RPG and action games, on the other hand, the plot is usually more linked into the actual game play. The backstory for the game has to be written, just as for a RTS or wargame to explain what is happening in the game world up to the point the game begins and what the player has to do in the short run (the first mission, as it were) and in the long run (defeating the master villain) to complete the story. As players begin to explore the world, they might encounter NPCs who interrupt the action to explain some plot point or game mechanic. There might even be cut-scenes that give more exposition about the overall storyline. Game play is frequently interrupted during such games to further the story, based on what the player has achieved so far, and to change the existing conditions of the game world—adding allies to a party, undertaking sub-quests, buying and selling items in stores, and so on. In this case, the designer has to write the story plotline knowing that it might branch out in an open-ended world where the player is free to roam without direct guidance.

What makes the story for an open-ended game world difficult to plot is that the designer can never predict exactly the current condition of the characters when major events are triggered. Some players might work their way through the game world slowly, buffing their characters to the maximum before initiating a major event, while others plow straight ahead and trigger major events while their characters are relatively weak and poorly armed. If players have trouble continuing because their characters are weak, then they either have to go back and grind out enough experience to get to a level where they can survive or simply give up and move on to a different game.

When designing an RPG or action game, the designer has to assume that the average player will be at a certain level and have certain equipment before a new major event is triggered. The designer can tie the plot advancement directly to the player's level and accomplishments throughout the game, for example, saying the player must stay in one location of the game world until reaching experience level 3 and solving three sub-quests, then in the next location the character is expected to reach level 6 and solve four more sub-quests, and so on to the end of the game. The restrictions placed on the player should be connected to the game's plot and someone should give the reasons as to why the player cannot advance until certain conditions are met. Even if players decide to spend lots of time buffing up their characters to much higher experience levels, they will not be able to trigger the next major event until they complete the sub-quests that advance the plotline. Note that secondary sub-quests can remain open to be completed later, as long as the player can revisit those areas that have already been opened up.

It takes considerable testing with different kinds of players to double and triple check that major events are being initiated correctly without frustrating players by forcing them to build up their characters or to flail around trying to get the last critical sub-quest completed. The design team might need to add more secondary sub-quests to a game location or increase the experience point awards to satisfy players and keep them involved in the game.

Creating a Game Story

Coming up with a plot for a story should be easy for designers, since they are supposed to be creative. It helps to carry around a notebook or a cell phone that can store messages to jot down ideas as they come to mind. It can take time to figure out all the details of the plot, especially for a story-centered game. Since designing the story for an RPG is particularly challenging, this chapter will focus on the elements that go into building the story structure.

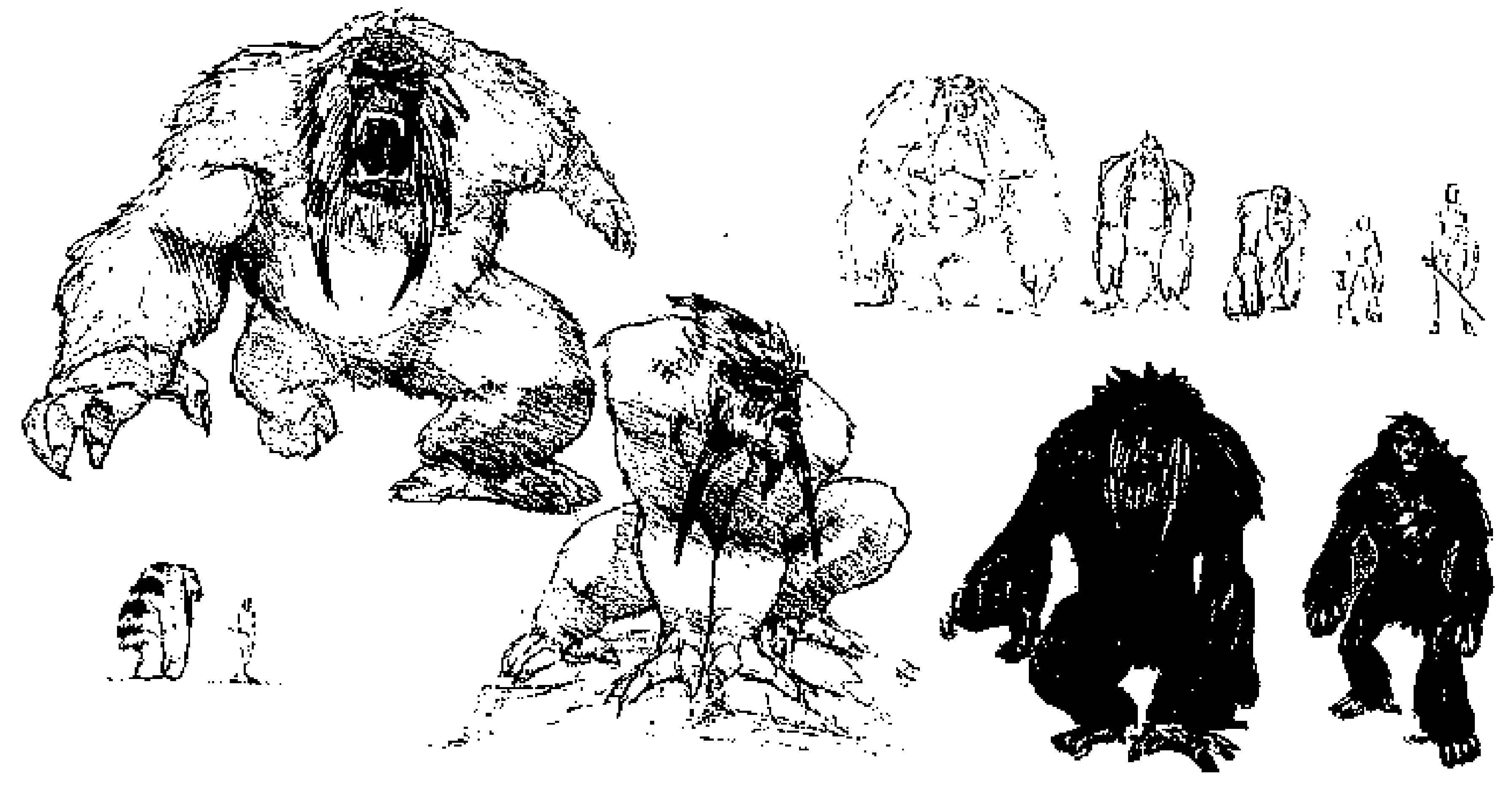


Figure 10.4. *The designers have to work with the art team to come up with different concepts of what characters and enemies look like, as in this sketch of an apelike creature created by an artist at Gas Powered Games.*

In addition to creating the main characters, the designers have to come up with a complete game world with different environments that are populated with NPCs and other entities. If different intelligent species inhabit the game universe, their histories and cultures have to be worked out at a high level, indicating why they act they way they do and how they interact with each other. The designers can fill in the details about these beings at a later time, but they should plan on working with the art team to come up with the final appearance and identifying unique actions for each race (Figure 10.4).

The master villain's master plan has to be worked out as well and the tasks the player must undertake to stymie the plan. The master plan sometimes involves the history of the game world, so the designers needs to come up with the most important events relevant to the story and, if necessary, tie them in to the histories of the races that have been or will be affected. Again at this point, the designers just need to work out the high-level concepts and leave the details for later. It is more important at this point to work out the master villain's plans in enough detail so that the major quests for the player can be explained coherently. The locations and NPCs involved in each quest have to be defined and placed on the game world to make sure that there are no problems or contradictions.

Finally, the designers have to come up with the important subplots that get sprinkled throughout the game world. The payoffs for each sub-quest should be defined, especially any that relate to major quests, and the locations and NPCs involved in each quest have to be identified. Inconsequential sub-quests can be added later after production begins to give more flavor to certain areas or to add more things for the player to do.

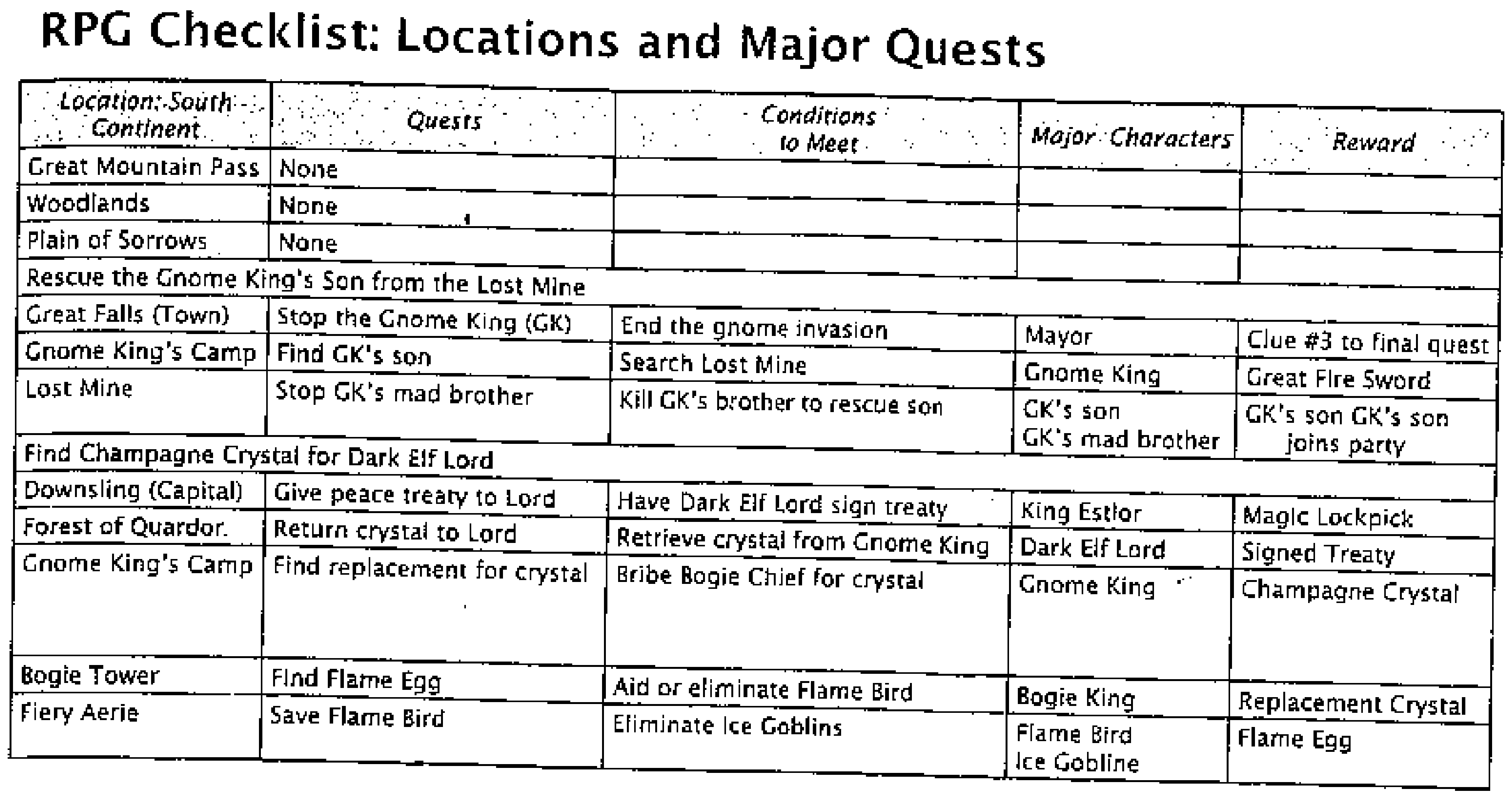
Paper Design Tools

At this point, it advantageous for the design team to create a world map on paper, showing the major locations of the quests and important sub-quests. The map does not have to be highly detailed, but it should show the environment, the races and creatures for each location. Ideally, the map should be fairly large and should be readily changeable and reproducible.

The design team should mount the map on a large cork bulletin board and perhaps hang another map in public where the whole team can refer to it. While working out the details of the storyline, it helps to use index cards or post-it notes to jot down information and then tack them to the map. By creating a visualization of the storyline, the designers can make sure that all the quests and sub-quests are spread across the entire game world. If some locations prove to be unimportant to the main storyline, they can either be dropped or set aside for sequels and add-ons.

Once the main storyline is set, a flowchart of the action can be created to incorporate into the design document and to print out and post with the maps. Since the details of the story can change once production is underway, the story flowchart and the map should be updated regularly both in the design documentation and on the bulletin board to reflect changes. The map and flowchart are useful tools for the testers to check that all the details of the story are correct and that the dialogue also refers to the correct locations and characters. If the dialogue is to be spoken by actors, it is important to get the details correct as early as possible since any mistakes will require the voice actors to come back to the recording studio for pick-up sessions. Moreover, if the game is to be released internationally, any changes to the dialogue will require the localization houses to bring in their voice actors as well for more recording sessions.

Even though a final paper version of the whole game is likely to be too detailed for a bulletin board, the design team should keep the design document updated as details are worked out. The updates can be done either in a word processing or spreadsheet program. The document can be posted on the company's intranet for reference by



RPG Checklist: Locations and Major Quests

Location: South Continent	Quests	Conditions to Meet	Major Characters	Reward
Great Mountain Pass	None			
Woodlands	None			
Plain of Sorrows	None			
Rescue the Gnome King's Son from the Lost Mine				
Great Falls (Town)	Stop the Gnome King (GK)	End the gnome invasion	Mayor	Clue #3 to final quest
Gnome King's Camp	Find GK's son	Search Lost Mine	Gnome King	Great Fire Sword
Lost Mine	Stop GK's mad brother	Kill GK's brother to rescue son	GK's son GK's mad brother	GK's son GK's son joins party
Find Champagne Crystal for Dark Elf Lord				
Downsling (Capital)	Give peace treaty to Lord	Have Dark Elf Lord sign treaty	King Estlor	Magic Lockpick
Forest of Quardor.	Return crystal to Lord	Retrieve crystal from Gnome King	Dark Elf Lord	Signed Treaty
Gnome King's Camp	Find replacement for crystal	Bribe Bogie Chief for crystal	Gnome King	Champagne Crystal
Bogie Tower	Find Flame Egg	Aid or eliminate Flame Bird	Bogie King	Replacement Crystal
Fiery Aerie	Save Flame Bird	Eliminate Ice Goblins	Flame Bird Ice Gobline	Flame Egg

Table 10.1. *The locations and quests for a game can be kept on a spreadsheet as a handy reference during production and testing. Each quest includes the steps and conditions required to complete it, the major NPCs and the payoffs.*

level designers and artists. See Table 10.1 for a sample spreadsheet of a game location with details.

Using "Chapters" to Structure a Story

If the story for a game is epic in scope, as many RPG game stories are, then care must be taken to make sure that all the loose ends for plot elements and sub-quests are wrapped up and not left incomplete. Each quest and sub-quest should have clearly defined beginnings and ends, which are reflected in the dialogues with the many NPCs the player encounters. One method for structuring a large game story is to use self-contained "books" and "chapters," just like in large novels. A "book" in this case refers to a major section of the story that can involve several sub-quests (the chapters). The designers have to define the major plot element that begins the book and what the player has to do to finish it. Likewise, the sub-quests forming the chapter have to be laid out with their beginnings and endings.

Assume, for example, that the player is given the task of tracking down the Runestone of Thor by a powerful wizard who will in return deliver the first key to open the major villain's lair. This is the first book and the various objectives the player has to achieve to retrieve the key are the chapters of the book. To get the Runestone, the player has to undertake four sub-quests, and the wizard gives instructions on the location of the first sub-quest. At this first location, an NPC asks the player to kill a monster that threatens his village. When the monster is slain, the NPC gives a reward and directions to the second sub-quest, thus ending the first chapter of the book. The sub-quests continue until the player finally gains possession of the Runestone at the end of the fourth chapter. Along the way, the player learns from various NPCs that giving the Runestone to the wizard isn't such a good idea, since the wizard is *non compos mentis*. Still, to complete this sub-quest the player must bring the Runestone back to the wizard, which results in a battle with the wizard who is indeed bonkers. By defeating the wizard, the player gains the first key to the master villain's lair and a clue to where the next key is hidden. The *Wizard's Runestone Quest* book is now finished and the play proceeds to the next book.

Testing Story Cohesiveness

A book and chapter structure for a large storyline provides breakpoints that can be checked to make sure the internal logic for all the scripts is correct and that items are correctly added and removed from the player's inventory as required by the story. This approach lets the designers make a checklist for themselves of what needs to be done to complete each chapter and book. For example, are all NPCs saying the correct dialogue at the same time and then changing their responses as conditions of each sub-quest are met? Are locations correctly affected by the resolution of the sub-quests—for example, the mad wizard's tower in the mountain path gets destroyed during the battle and therefore blocking the passage between two areas for the rest of the game? Are the loose ends all tied up by the end of the book so that NPCs no longer talk about finding the Runestone and the sub-quest can't accidentally be restarted?

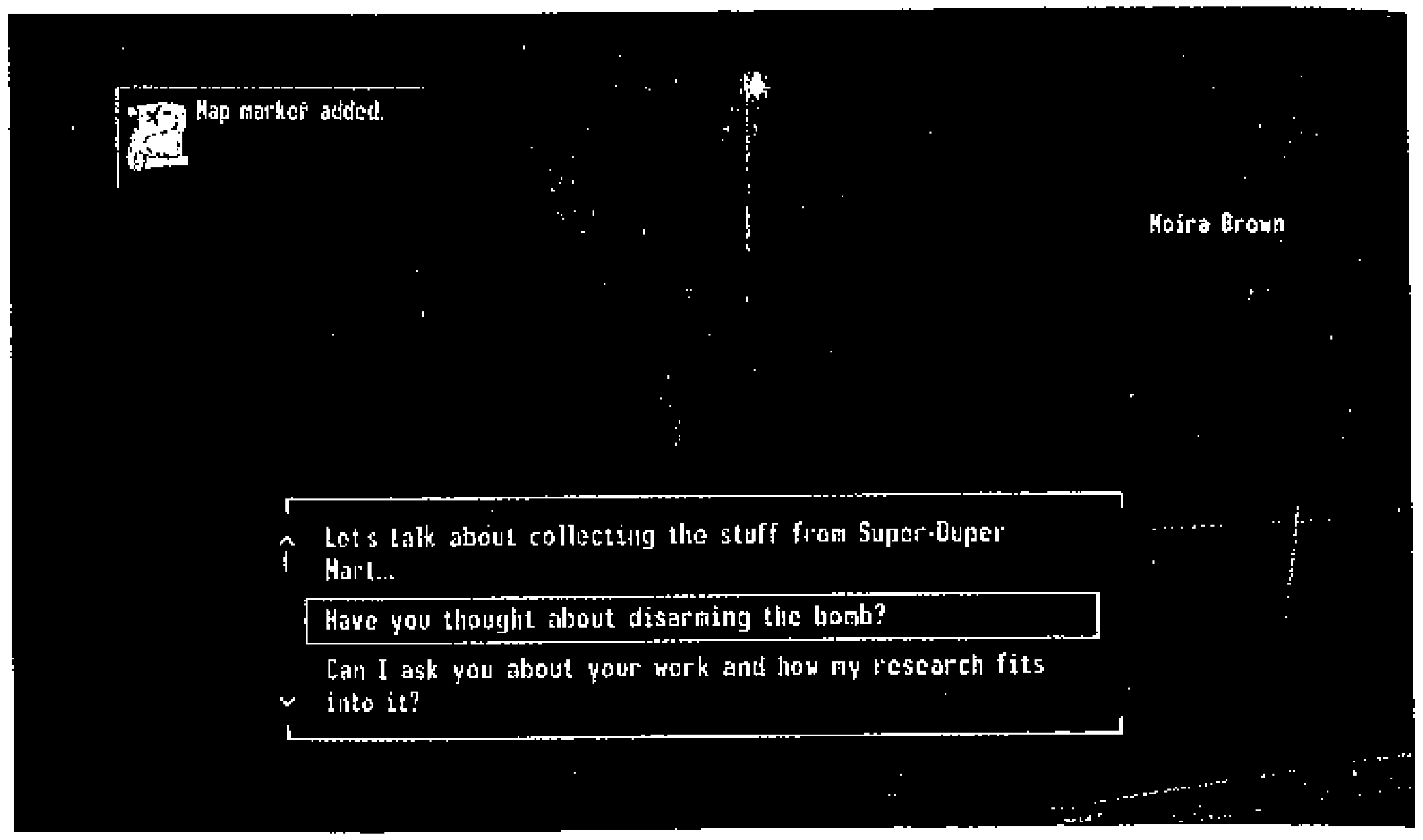
This structure works well for plots that are linear, meaning that plot elements from one chapter do not spill over into a new chapter. Where this structure works less well is when the story structure is more episodic and plot elements can overlap one another. For example, if the player triggers the "Find the Green Key" episode while the "Make My Witch Ally Happy," "Earn Enough Gold to Buy Sword of Entitlement from Dwarf," and "Find the Secret Passage to the Underworld" episodes are running concurrently, the dialogue choice for NPCs involved in these sub-quests can get very complex. It requires much greater effort by the design team to ensure that the episodic structure does not cause contradictions in the script logic or problems trying to acquire items or get to important places in the game world.

As an example, suppose that in the earlier example there are four keys the player must acquire to gain access to the villain's lair. If the player can go after the four keys only one at a time, the plot is linear and the chapter structure works well. If, however, the player can go after all four keys concurrently, then there can be considerable overlap of conditions the game logic must deal with. The NPCs have to be able to respond correctly to the current state of affairs and their dialogue logic must be carefully structured so they "recognize" which episodes are currently running and which ones are complete, and so respond appropriately during conversations.

It is also possible that sub-plots can carry over from one episode to another if they aren't directly connected to the main storyline—for example, a quest undertaken by a secondary member of a party. It is better to give these sub-plots their own book that is separate from the main plot. The timing for sub-plots can be important, so the designers might want to have a clock running in the background that allows these minor plots to start and end at times that won't affect the main plot. If the player doesn't help the party member within a certain amount of time, that NPC's sub-plot never starts. Likewise, if the player doesn't help complete the sub-plot in a given amount of time, the NPC simply refuses to reward the player.

Dialogue in Games

In games where a player interacts with NPCs, the dialogue for all characters is written by the designers, although professional writers are often brought in to polish the final text to make it sound more natural when spoken by voice actors. In games where there are no dialogue choices available to the player, the text is relatively easy to write because everything is linear. The player initiates a conversation, the NPC responds, then the player responds, and so on to the end of the chat. Once the initial conversation is over, the NPC then repeats some generic text until circumstances in the game world change—for example, if the player has completed a quest initiated by the NPC. After another exchange in which the player is rewarded, the NPC then repeats another text message until the state of things in the game world changes and causes the NPC's dialogue to change again. A given character can have many changes in dialogue, depending on



Let's talk about collecting the stuff from Super-Duper Mart...

Have you thought about disarming the bomb?

Can I ask you about your work and how my research fits into it?

Figure 10.5. *In Bethesda Softworks'* Fallout 3, *the player interacts with NPCs by selecting dialogue options. Sometimes the player is offered the option to select from friendly, neutral, and hostile.statements.*

how intricate the game plot is and how often major changes to the game world demand additional text.

In many games, the player is offered choices during a dialogue, in which case the process of writing the text becomes much more challenging (Figure 10.5). For example, if the player is allowed to show the character's mood during a conversation, there might be several choices for text: for example, one text that is jovial and friendly, one that is neutral and non-committal, and one that is threatening and antagonistic. Depending on the choice the player makes, the NPC can respond in kind and his or her responses can change dramatically based on the NPC's new mood toward the player. This type of dialogue is referred to as branching dialogue because each text choice allows a number of possible responses, which can grow over time like the branches of a tree. While this form of dialogue makes the NPCs feel more realistic, it can become daunting to write because the responses can spiral out of control into sheer confusion.

Scripting the Dialogue

A scripting system (or scripting language) is used to create branching dialogue and to change responses as the conditions of the world change. The system uses variables (or flags) to note the current conditions of game states, and when the states change the variables are incremented or decremented accordingly. For example, if the player is about to visit an NPC that will initiate a quest, the variable that tracks changes in the dialogue is set to zero, and as the player performs actions, the variable's value is changed according to the new game state. The logic for the change might look like this:

```
If (KING_QUEST_CHAT = 0)
   SAY ("The king bids you retrieve his
sword from the Bog of Doom.")
           SET (KING_ QUEST_CHAT_1)
Else if (KING_QUEST_CHAT = 1)
   SAY ("The king asks you if you've found his
   sword yet.")
Else if (KING_QUEST_CHAT = 2)
   SAY ("The king thanks you for retrieving
   his sword and gives you a bag of gold.")
           SET (KING_ QUEST_CHAT_3)
           TAKE_INVENTORY (KING_SWORD)
           GIVE_REWARD (1000_GOLD)
Else if (KING_QUEST_CHAT = 3)
   SAY ("The king thanks you again for
bringing back his sword.")
```

In the above example, the first text appears when the player first meets the king who initiates the quest to retrieve his sword. Because the quest in now begun, the variable is incremented by one (it could be set to any value used by the scripting system), and the king keeps asking about his sword until it is found and brought back. Once the sword is in the player's possession, the variable controlling the king's dialogue is incremented to 2 (`KING_QUEST_CHAT = 2`). At that point the king thanks the player for completing the quest, and the sword is removed from the character's inventory and the player's gold is increased by 1000. Since the quest is now complete, the king keeps repeating his thanks to the player until something else happens in the game world that causes the king's dialogue to change.

When the player initiates conversation with the king, the game engine checks the current value of the `KING_QUEST_CHAT` variable to determine the appropriate text to show on the screen. Other variables can be used to change the mood of the king's responses as well, for example, if the king gets more and more upset that the player hasn't yet found the sword yet, the logic for the variable "KING_QUEST_CHAT = 1" can have its own set of variables:

```
Else if (KING_QUEST_CHAT = 1)      AND
   (KING_PATIENCE_LEVEL = 0)
   SAY ("The king asks you nicely if you've
   found his sword yet.")
         SET (KING_PATIENCE_LEVEL = 1)
Else if (KING_QUEST_CHAT = 1)      AND
   (KING_PATIENCE_LEVEL = 1)
   SAY ("The king is impatient for you to
   return his stolen sword.")
         SET (KING_PATIENCE_LEVEL = 2)
Else if (KING_QUEST_CHAT = 1)      AND
   (KING_PATIENCE_LEVEL = 2)
   SAY ("The king sputters in rage about
   your incompetence. 'Where is my sword?'
   he roars.")
```

In this case, the king gets angrier each time the player speaks to him until the third response, which repeats pending the player bringing him the lost sword. It should be obvious that as using branching dialogue can get to be complicated because the number of variables that are used to measure the NPCs' temperaments can increase rapidly.

Reducing Dialogue Options

While branching dialogue can add considerable flavor to a game, it also makes the task of testing more difficult. It is easy to forget to modify the value of a variable, resulting in the wrong text appearing, and when the text options grow exponentially in branching dialogue, it is almost inevitable that problems will arise. Add to this problem the fact that the game might be translated into multiple languages and the task of testing the dialogue in all versions to guarantee that every character says the correct text at the appropriate time can become overwhelming.

There are several ways to reduce the clutter produced by branching. One approach is to reduce the number of dialogue choices the player can select or to use them only in the most important circumstances of the story. However, the sudden addition and reduction of text choices not only breaks the convention used for dialogue but also can confuse a player who is suddenly expected to make an important decision that has repercussions on the plotline.

Another approach is to limit the number of actual responses by NPCs to what the player says. For example, if the player always has the option to select a pleasant, a neutral, or an angry text message, the NPC might respond with the same text to the pleasant and neutral choices and respond differently only to the angry text. This reduces the number of branches from three to two, making it much easier to write. Of course, players might wonder why they were given three text options, since NPCs respond only one of two ways.

It is also possible to simply have the NPC stop responding if the player is acting abusively. Each time the player selects an angry response, a variable is incremented and after a certain threshold is reached, the NPC simply stops responding to the player, either temporarily until another dialogue is triggered or permanently until conditions in the story change. Of course, it can be important plotwise to employ angry messages to get stubborn or undecided NPCs to give up some important information. In this case, the text should be written so the player knows the NPC is slowly opening up from intimidation and not just shutting down out of contempt.

Gating

Another method for limiting branching is "gating," which forces all the dialogue decisions either to be resolved at the end of a chapter or to be tagged by a variable to continue into the next chapter. In effect, the end of the chapter is like a wall and only the correct text messages are allowed to go through the "gate" into the next chapter.

When using this approach, each branch of text messages can include a variable that identifies it as starting in a certain chapter. When the story element that concludes the chapter is finished, the text variables for the current chapter are all reset to "0" except for those that are still available in the next chapter. It then becomes easier to check all the NPCs to ensure that their text dialogues end correctly. This method can be used to change the text messages for generic NPCs when major conditions in the game world change. A nice tool the programming team should consider building for the design team is a simple text editor that allows changes in dialogue to be tracked throughout a chapter or even throughout the text in the whole game.

Flowcharting NPC Dialogue

Another tool the design team can create for themselves is a flowchart for the text responses by each NPC, especially if using a branching dialogue structure. The chart can be created either in a word processing document or spreadsheet, and the designers should settle on an appropriate format structure for NPC responses. The real text can then be plugged into the format structure so that the logic can be worked out ahead of time before trying it out in the scripting language.

Here is an example of a format structure for an NPC initiating a quest:

- *Greetings.* Personal introduction and start of the quest.
- *Waiting.* Response while quest is underway.
- *Payoff.* When player completes quest and gets reward.
- *Follow-up.* Chat after quest is over (until some event triggers new dialogue).

Many NPCs have no direct influence on the events of the game and are used as simple window dressing to make the game world feel more populated. In such cases, these NPCs are given a single line of text that is repeated throughout the game. In some cases, however, these NPCs are given multiple responses, even if the player isn't given multiple dialogue choices when first initiating contact with an NPC. These NPCs often give considerable information about the town and local environments as well as the various important characters the player will meet. Usually, a simple variable system is used to key the changing dialogue responses:

```
If (NPC1_CHAT = 0)
    SAY (“Welcome to the town of Abbadabba.
    You’ll find some fine shops to the north
    of town and the famous Cave of Aladdin to
    the east.")
                    SET (NPC1_CHAT_1)
Else if (NPC1_CHAT = 1)
    SAY (“It’s good to see you again. I hope
    you’re enjoying your visit.")
SET (NPC1_CHAT_2)
Else if (NPC1_CHAT = 2)
Else
SAY (“I’ve heard there’s an old widow in
the forest to the north who is looking for
someone to help her find a lost brother.")
                    SET (NPC1_CHAT_0)
```

Using this approach, an NPC can be given any amount of text to say to the player, and variables can be set and checked to see if the player has undertaken local sub-quests, so the NPC's responses change based on which quests have been completed. Of course, as with all dialogue in the game, the design team should keep in mind that the text will have to be translated for oversea markets, and the challenge of testing for bugs in dialogue responses grows as the game text grows.

Conclusion

Creating an interesting story that gives structure to a game is a lot of fun but is also a huge amount of work. In addition to working out details of the main plotline, the designers have to create the history of the locations in the game and the characters who inhabit them. The documentation needed to describe a large game world can take up hundreds of pages, most of which is not seen by anyone outside the team. Where a novel is completely linear with only one path from the beginning to the end of the story, a game often allows players to create their own paths through the story. Using a strictly linear story structure in a game helps cut down on the work, but can feel confining to players since they can't control where in the world to explore next. A quest/sub-quest structure allows designers to create more open-ended game worlds for players to explore while still providing specific goals for players to achieve as they work towards the climax of the story.

Likewise, the task of writing dialogue for a large game is not trivial, and requires much more careful planning than simply working out the plot, especially if the designers want to give the characters choices in their attitude towards NPCs. While using flowcharts to map out the dialogue for major character and minor NPCs might seem pedestrian, it is much better to invest the time during the design phase rather to wait for some muse to strike with the divine fire of inspiration two-thirds of the way through the development process to help resolve some plot problems. The testers will also be much happier if they have roadmaps of the plot and dialogue to check against during the long process of debugging the game.

Exercises

1. Do some research on the Internet to find three examples of games where the story was important to the progress of the game and three examples where the story was unimportant. Write up an analysis of your findings as to why the stories made some games better while other stories added little to game.
2. Draw up lists of ten novels or movies that have greatly influenced the stories in console and computer games.
   a. Create a list for each of these genres: science fiction, fantasy, and horror.
   b. Compare your lists with others and note those works you have not yet read or viewed.
   c. Read or review one such book or movie and write an essay on its effects on games.
3. Take a story structure discussed in this chapter (quest, zigzag, mission, or Hero's Journey) and write up the outline for the plot of a game.
   a. If your game project is a role-playing, real-time strategy or first-person shooter game, expand the main story you came up with so that it becomes the backbone for all game action.
   b. If your game project is a simulation or wargame, write up the historical or sociological information the player should know before beginning play.
   c. The outline should include all major locations in the game, the major characters involved, their motivations for acting, and what the player must do to complete the story.
4. Once you have the outline from Exercise 3, create a flowchart or spreadsheet for the story/history showing the major quests/missions involved. (You don't have to include minor quests at this point.) Prepare a presentation of your story for others to critique, so you can get feedback on its cohesiveness and whether the overall story is too flimsy or too complex.
5. Think about what players see and learn at the very start of your game project and create the opening scene or animated sequence.
   a. Create simple storyboards for the backstory or historical narrative of the game.
   b. Write up the dialogue for the characters and/or narrator.
   c. Prepare a presentation of the storyboards so you can get feedback from others to make sure the understand not only the main goals of the game but also what they need to do once they play begins.
6. Create a short but emotionally charged dialogue between two characters where they three possible dialogue options based on their changing emotional states.
   a. Draw out the responses as branches on a dialogue tree, making sure that each response is unique. Use an "if-else if-then" structure to determine the logic for the responses.
   b. Test the dialogue by running through every variation.
7. From Exercise 6, you probably were aware that your dialogue tree quickly grew massive and unwieldy after a number of exchanges. Now use "gating" to reduce the number of unique responses to a more reasonable number. Test the dialogue again to see that all the responses make sense.
8. Include two versions of the story or background information for your game design document.

a. Create a short summary of the plot/history for the first section of the document, which is aimed at management and marketing.

b. Start working on a longer version where you go into more detail about the plot/history and also the environment, the characters and the important items in the game world. This longer version can appear as an appendix.

CHAPTER 11

DESIGNING PLAYFIELDS

Most commercial console and computer games are set in three-dimensional settings inside large worlds for players to explore. However, a number of games, especially those designed for less powerful platforms or as casual games, still use two dimensional playfields where figures are flat and animate against flat backgrounds almost like cartoons. Three-dimensional games are much more interesting visually, but they are also much more challenging to design and build. While most designers can quickly learn to use 2D map editors to build flat playfields, it takes additional skills and artistic talent to build 3D levels in a graphics program like Autodesk's 3ds Max or Maya or using commercial game engines like Epic Games' Unreal Engine or id Software's Tech 5.

Whether in 2D or 3D, the playfield for a game has to be appropriate for the game world and enjoyable to explore. There should be surprises for players to stumble upon as well as dangers to face. There should also be logic to the way the playfield is constructed so the players can figure out how to get through the map without getting completely lost or having to retrace their steps in search of some unobvious path to a new area. Secret paths, of course, can lead to hidden areas with extra goodies, but the main pathway through an area should always be obvious. Good map/level design can make or break a game, so designers must understand the basics of what goes into building a playfield.

For purposes of clarity, the term *map* will be used to refer to two-dimensional playfields, whether shown top-down or from the side. The term *level* will refer to three-dimensional playfields, both inside buildings and other structures as well as external environments. *Playfield* will refer to the type of map used in a game, be it 2D