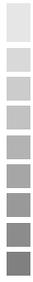
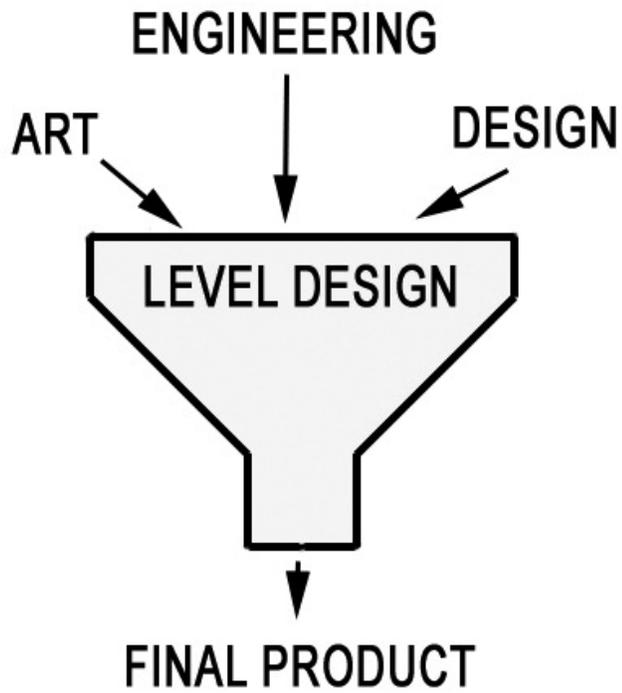
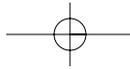


1



Introduction to Level Design





2 Game Level Design

In This Chapter

- Game Design
- Level Designers
- Anatomy of Level Design
- Defining Levels
- Brief History of Levels
- Level Design Today
- Summary
- Interview with Richard “Levelord” Gray of Ritual Entertainment

One of the hardest things about being a level designer is trying to explain to people what you do. This chapter will explain what levels are, where they came from, who makes them, and what “level design” means for the production of a modern video or computer game.

GAME DESIGN

Everything that is made has a designer. A designer formulates plans for creating products from concepts. In games, the designer is the person who often conceives the original ideas, puts them on paper to present to others (in the form of a design document or rough demonstration), and supervises the transition from design to a working video game.

Being the player’s advocate is the highest function of a game designer during the entire process of making a game. Simply put, this means that designers are the “eyes and ears” of the player, and represent the interests of the audience during the production. If a problem occurs in creating a game such that the player’s needs are not met, the designer must find a solution. When someone on the team wants to add something he feels is really cool, it’s the designer’s job to evaluate the addition’s potential risks, how much players will really use it, and what changes it will make to the players’ experience, good or bad. In the end, we make games for the players, not for ourselves, and designers are the people on the team who must always be able to see the game through the eyes of a player, rather than through the eyes of a tired developer who knows the product inside and out.

On a day-to-day basis, game design is primarily about creating and interconnecting all the elements that make up a game—the mechanics—and creating an appealing world in which to house them. Different types of nonplayer characters (NPCs) and their behaviors, weapons, and tools that the player will use and their effects; locations; items; on-screen interfaces; mood; emotional reaction; controls;



and camera views—all these things need to be considered in the early stages of developing a game. These days, a design team handles the work of documenting and implementing design decisions. This will be discussed in a later chapter, but the size of modern games means that the days of a single designer making all the decisions are quickly coming to an end.

LEVEL DESIGNERS

When it comes to actually creating the game from these beginning elements, a specialist is needed to implement the design. This is the essence of level design—the application of the team’s ideas in a playable form. A level designer is the point of convergence for programming, cinematography, audio, art, and design—all of the components of a modern computer or video game as shown in Figure 1.1. Game designers create rules and systems that form the backbone of every game, but a level designer implements them and makes them work properly. In addition, level designers carve out environments, create interesting visuals, monitor the performance of the game, make sure that technical problems are resolved before the product hits the shelves and fixes problems in the game. That’s a pretty exhausting list of responsibilities. As such, level design is an extremely important role in today’s production team—ultimately, the player experiences the game through a game’s levels.

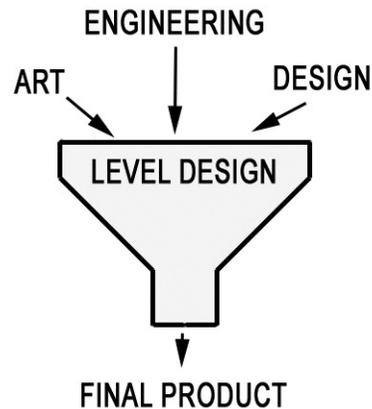
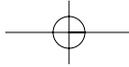


FIGURE 1.1 Art, design, and code all funnel into level design.

The level designer is an omnipotent power in the game, responsible for leading the player through the experience. However, the less the players feel the designer’s presence, the more they will feel in control of their own virtual destiny. A good level



4 Game Level Design

designer will create a level that is full of decisions players make. A *great* level designer will allow players to feel like they are making the right decisions, even if they really aren't. Partly, this can be through the illusion of choice—allowing the player three ways to choose that all lead to the same room, for example, is an easy way of letting the players make the surface decisions (“Which way do I go now?”) while maintaining control of their ultimate destinations. This can be taken further with concepts like systemic level design where players are given a high degree of freedom in the environment, but can still be guided along a narrative path. We'll discuss the different approaches to level design flow in Chapter 5.

On a visual front, level designers use the same art of illusion to create spaces that feel much bigger than they really are. For a game like *Unreal Tournament*, this might mean creating the illusion of an underwater world outside the window of an undersea base—even though there's nothing really out there. For a title like *Need for Speed: Underground*, this could be the multitude of inaccessible but plausible side streets and landmarks seen between buildings that give players the sense they are racing through a city rather than just on a single track.

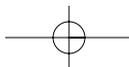
Regardless of the type of game or what platform it is for, until we work out a way to create worlds with as much rich detail and level of immersion as real life has, level designers will have to rely on the art of illusion to create believable and enjoyable game spaces.

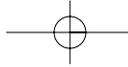
ANATOMY OF LEVEL DESIGN

Level Design is really a composite role, bringing together several disciplines: art, design, and engineering.

Art

In the past, games could often reach critical acclaim without the need to be visually stunning. These days, to create an interesting and atmospheric environment, a level designer must have some measure of artistic or architectural sense. A level that is well balanced, fun to play, and packed full of surprises will still face player criticism if the environment is crudely built or features a lot of obviously amateur art. Likewise, an architecturally impressive map with nothing to do in it is going to cause players to complain. Balancing artistic considerations with gameplay needs is an everyday struggle for the modern level designer. Although some designers can create many of their own art assets and take the visual quality of their maps into their own hands (schedule permitting), others may have more knowledge in a specific area such as modeling, texturing, lighting, or simply taking a few primitive shapes and evoking just the right emotions and imagery in the audience.





As games get more complex and level designers are responsible for more ambitious content, many teams are adding art support staff to help shoulder the burden of creating engaging aesthetics and allowing the designers to focus on the play experience. Regardless of who makes a level's assets, however, the level designer still holds the vision of the level and will be required to lend direction and vision to his team members during production. Some of the best level designers don't have an artistic background but, rather, use books and images to help them make interesting spaces. Level design does not require an art degree by any means; however, a level designer should be able to illustrate or describe the artistic needs and aesthetic requirements of the map to his team members.

If you're reading this and despairing—don't worry. Most of what makes a good artist is imagination, and the fact that you are, or want to be, a level designer is a pretty good indication that imagination is something you possess. Learning how to use your imagination wisely is something that can be learned—many great books teach the fundamentals of architecture, lighting, texture creation, and the like, examples of which will be included in later chapters.

Design

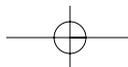
Although the amount of pre-design that goes into a level before building starts varies, there will always be times when a level designer needs to make a design decision in the process of constructing the map. We'll talk about this later in the book, but the level designer should be able to handle the implementation of the game design to achieve the goals for that map.

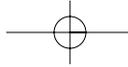
After the initial placement of game elements—after you've put in your enemy's units, your traps, puzzles, powerups, and everything that the player is going to interact with, you'll need to “tune” it all. Early drafts of levels are often disjointed and unbalanced, and unacceptable to release to the public. A level designer's game intuition is vital at this stage to go through the level and polish it, tweaking parameters, editing the variables for NPCs, trying to anticipate potential problems and ultimately designing an enjoyable experience for the player.

Level designers also need to be able to spot problems as they work and report them to the designer or producer. If the game designer is the general directing the game from above, level designers are scouts, on the front line of production and able to see potential trouble up close and personal, if they just know what to look for.

Engineering

Although the gulf of knowledge between scripting a level event and actually programming game engine functionality is sizable, some aspects of level design are closer to coding than anything else. Games frequently have an internal “script” system that allows designers to access parts of the game code in a more user-friendly





6 Game Level Design

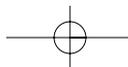
manner. The means differ from project to project, and some level designers need to be more versed in their game's scripting language than do others who might use simpler or more streamlined tools for setting up in-game events or editing level elements. However, the process is still the same—level designers will invariably be called on to plan, execute, and debug special situations in a level.

As games support larger worlds and more intricate stories, many developers rely more heavily on scripting to provide a sense of realism and action to the environment, as well as to create bigger and more elaborate situations for the player. Boss battles, patrols, the behavior of certain objects when hit with a projectile or the behavior of a civilian when seeing one of the player's units—all these things are potentially scripted by the level designer. As such, any knowledge about scripting or programming can come in quite useful when making playspaces.

Another aspect of level design is technical in nature—performance. Level designers are usually expected to bear a large responsibility for how their environments run. Every game has limitations in how complex the world can be, how many moving characters can be calculated, and how many textures or lights can be displayed in a scene before the game engine is overtaxed and the performance of play degrades. This often results in loss of frame rate—the view becomes jittery and the controls become hard to use. Further problems such as objects overlapping the same space, or errors in the geometry, can cause technical problems too. In general, the performance issue is one that becomes more and more important as the project nears the final shipping date, and a level designer needs to know not only how to spot these problems in a map but also how best to deal with them—be it a work-around, remaking that part of the level, or even amputating the whole section from the map.

DEFINING LEVELS

The term *level* is synonymous with “map,” “mission,” or “stage” in many games. The original term *level* in games most likely comes from the early arcade machines and home game systems where the play experience was divided into increments of difficulty, called stages or levels. For instance, once the player had finished the first wave of enemies, he was considered to have finished “Level One” of however many levels of difficulty the game allowed. These levels were descendants of “Dungeon levels” in early role playing and tabletop games like *Dungeons and Dragons*, which divided the game environments—most often dungeons and subterranean structures—into vertical floors, which not only determined how deep the players were, but also gave an indication of how powerful the creatures would be. Level Five creatures were obviously going to be a much bigger challenge than mere Level Ones, being further from the surface and the safety of retreat.



A modern game level has a wide range of forms. A common example is a single *Deathmatch* or *Capture the Flag* map you might play in your favorite shooter. Or it could be a track in a racing game, or simply the maze from *PacMan*. At its most basic, a level is simply an environment for gameplay. Does a level have discernible characteristics? Well, it has physical boundaries. It has entrances and exits. It has goals, and it has a beginning and an ending—or it has many of them. A level can contain almost all the game’s systems and mechanics, or it can focus on a single activity. Some levels are unique, such as a boss level. Some levels are crossed through repeatedly like the parts of the city that compose those of *Grand Theft Auto 3*.

Every game takes place in an environment, and that’s what level designers must provide—putting the “ground” in playground. A level is really a container for gameplay.

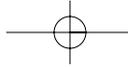
BRIEF HISTORY OF LEVELS

As long as there have been games, there have been environments to play them in. Almost every culture has its version of chess, along with a board to play it on. Even in the absence of a board, players have scratched playfields in the dirt or scribbled them on paper like tic-tac-toe. Gameplay needs a vessel in which to exist. Similarly, although the craft of creating interactive environments for video games is fairly new, there is a great deal of history behind it.

Creating Pinball—The Mother of Level Design

Although the level designer position as a team role has only been around for the past 10 years or so, games have always needed play fields. In fact, the first examples of “playfield design” started back in the days when pinball was becoming a national pastime. Early versions of pinball—called bagatelle—were random affairs. The ball was entered into the playspace and found its way down through the layout of pins until it came to rest in a numbered hole. The player really didn’t have much control of the ball once it was in play. Although there was some excitement watching the ball progress through the pins, it was more akin to pulling the lever on a slot machine, or watching a movie—once the initial interaction of starting the process was over, the participant could only watch helplessly as events unfolded.

When pinball designers began to add in the element of interactivity, such as the addition of flippers or the ability to guide the ball into reward-rich areas (i.e., a part of the board with a cluster of high-scoring bumpers, or triggering the release of bonus balls), the game made its move from passive to active entertainment. Much in the way that even though building a game level shares many common elements



8 Game Level Design

with building a movie set or describing a location in a book, what sets it apart is interactivity—the player has the opportunity to choose and alter the flow of events to his desires. That’s the “play” in gameplay.

It is interesting to note the similarity between pinball design and modern level design. Both were concerned with the funneling of an avatar—in pinball’s case, the player avatar was a small metal ball—through an interactive playfield full of rewards and hazards. With each generation of pinballs, the designers had to create new variants on old favorites and develop original ideas to keep players interested. Level designers would do well to look back to the golden age of pinball because these are our real roots—the first examples of interactive environment design.

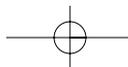
From Pinball Machines to Super Computers

As computers began to appear in universities in the 1970s, eager engineers started turning them to recreational uses, and the transition from the pinball table to video screen began. Unfortunately, the capabilities of computer-driven playfields were vastly inferior to the long-established mechanical pinball machines. In addition, the people making video games were almost always engineers and students taking a break from their real work, rather than professional game designers, so the art of playfield design had to start all over again, accounting for the new display and control methods.

In the Beginning There Was *Space War*

Widely considered the grandfather of all computer games, *Space War* was actually displayed on an oscilloscope and contained only a single planet at the center for two players to fight around against a backdrop of stars. This could be considered the first video game level. The planet was not just for decoration—it exerted gravitational influence on the players’ ships and projectiles. Thought went in to creating an interesting playspace when really, if it had simply been a blank background, no one would have complained.

As games matured, their playspaces matured also. More attention was given to the way game environments looked, and the kinds of experience different environments could give the player. Care was taken to ensure the player was steadily challenged through shifts in environmental parameters. Music and audio played more important roles in both inviting players to the game and providing feedback about their performance. Gradually gameplay went from one-screen action (like *Pong* or *PacMan*) to multiscreen or scrolling environments like *Pitfall* and *Tempest*, where the player was suddenly given greater opportunity for discovery and greater freedom of movement. Playspaces became richer, and gameplay rules more complex. *Defender*, for instance, featured a rapidly changing environment, intense special effects, and audio feedback. *Defender* was one of the first games where the player was informed of things happening in another location by audio cues—when a “human”



was converted into an enemy unit, a specific sound effect played. Although the levels allowed the player to travel left and right over the landscape, randomly moving opponents of varying speeds and accuracy meant simple travel in a straight line was impossible and the experience of each stage was always slightly different. Even though the controls were fairly simple, the sheer complexity and intensity of the levels made *Defender* a favorite for hard-core arcade junkies.

Similarly, for home systems, the Atari game *Adventure* had a randomization routine that meant the player didn't know where all the necessary items in the game were each time he played. The game was laid out on a number of screen-sized rooms that the player would travel between, dodging dragons and collecting required components to beat the game. The first fledgling elements of level design were being born to the gaming world.

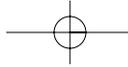
The Rise of Home Computing

In the 1980s, the rise of home-gaming on consoles and personal computers meant gamers were hungry for greater challenges, and developers quickly responded with more advanced level design concepts. Armed with more computing power and increased storage capacity on modern gaming machines, the basic elements of earlier genres such as moving platforms and enemies with simple, looping attack patterns were combined and evolved in different ways to create new challenges for the player. Designers strove to encourage exploration by hiding special rewards or even entire levels for discovery by the careful player. Environments became more interactive, introducing complex puzzles to block progression and produce richer and more varied gameplay to keep the player challenged. Narrative became an important focus as games suddenly came with richer back stories and character development rather than simply suggestive box art. Early text adventures, for example, relied on more complex story lines and descriptive text to keep the player engaged. A classic adventure in this style was *Planetfall*, which is widely regarded as being one of the first games to make players cry because of the death of a character.

However, as involved as these new game environments were, there still wasn't a specialized role for their creation yet. Video games were made by only a handful of people, who handled everything required—programming, art, and design. Audio expectations were low enough that the programmers often handled those aspects too. In the heyday of the video arcade in the 1980s, many games were designed, programmed, and decorated by a single person.

LEVEL DESIGN TODAY

Because of the explosive increase in complexity and in expectations of modern interactive entertainment, it's not uncommon to find production teams of 30, 50, or



10 Game Level Design

even more than 100 developers working for years to complete a single title. In such an environment, work is divided up into very narrow specializations, and more often than not one of these specializations is you—the level designer.

Contemporary level designers have a considerably larger responsibility in game production today. Fortunately, they also have a much bigger palette of tools and a huge amount of support in production as well. However, levels are not simply around because they have a history. Having levels helps a game in many ways, including the following:

- Overcoming memory constraints
- Narrative chapters
- Dividing the workload

Overcoming Memory Constraints

In their earliest forms, interactive games were usually simple affairs. Earlier we mentioned that *Space War* was played using an oscilloscope to display two ships and a planet. *Pong* had variable speeds and opponent response, but the playfield never changed. Game graphics were limited by ridiculously meager computational power by today's standards, and often took place in limited or repetitive environments. Most importantly, the technology at the time meant that games needed to load into whatever memory the machine had and stay there until the game was switched off or reset. Given that available memory capacities at the dawn of the computer age were minuscule compared with today's—the need to keep games as simple as possible was a predominant concern. Later, with the introduction of portable storage media like floppy disks and tape cassettes, games expanded enormously in both size and scope, and it was impossible to load the whole thing into computer memory all at once. The concept of levels (or chapters) became more prominent as a way to break up a game into sections that would only be brought in when needed. A game that was broken into sections could be much larger than the available memory of the gaming machine. When each section was finished, it would be replaced with another section loaded from tape or disk.

Early home computers such as the Commodore 64 or Sinclair Spectrum subjected the player to long waits while stages of the game loaded from tape cassette. Thankfully, storage media can be read fast enough now that consoles and computers can quite easily load in specific parts of the game they need from the CD or DVD without the player ever knowing. However, levels have also expanded in size, often having large amounts of unique textures, decorative meshes, character models, scripted sequences, and a host of support content that makes them impossible to load other than one at a time. Thus, the original concept of breaking a game into smaller pieces—levels—is still necessary to avoid straining the processor and to allow epic-sized games to parcel themselves out into bite-sized pieces.

Narrative Chapters

Very commonly, a game's levels are set up in a narrative fashion, telling a story within a story. The player character enters the level, explores his surroundings, encountering increasing challenges and dangers along the way, until the end is reached.

Most games have some form of story or narrative that draws the players along, and many games use levels as a book would chapters—dividing the story into segments allowing story arcs, the introduction of new characters, resolution of goals, unexpected return of old enemies, and so on. In many cases, a level is like a novella—a short, self-contained story that has an introduction, a series of encounters and challenges, and a final resolution. As games start to create broader, less linear story lines, levels begin to contain many story possibilities, which we'll explore in greater depth later in Chapter 5 as emergent gameplay.

Levels encompass areas of connected gameplay and provide logical breaks between key story locations. For instance, one level of a game that uses time travel as a story element might have the players in Berlin in 1800, and the next level has players in the same city in the year 3000. Separating these two periods into levels is logical, as they can be bridged by a cinematic, scripted sequence, or a simple voice-over, to create a more dramatic transition between the two locations.

Dividing the Workload

Level design arose out of a need for specialization within game production teams. As game sales grew, the one-man shows faced new challenges in keeping up with increasing consumer demand for quality and quantity of content. To maintain constant levels of production quality, game teams began to grow in numbers. Aspects of game development that one person had done previously were gradually being done by two or three people. With larger teams, programmers who might have handled both programming and the art were being relieved by full-time professional artists. Similarly, new positions such as game designers, sound effects engineers, and character animators developed to help spread the effort of creating a computer game over a wider team of individuals, each with a narrower set of tasks.

The main advantage of creating a game in stages is that it can be built faster, and production speed can be a huge factor in gaining a publishing deal or getting a milestone out the door in time. The more you can subdivide your game into distinct levels, the more designers can work on them simultaneously. In addition, with the advent of specialized level editors and working environments, the ease of importing and exporting assets (the individual art pieces used to decorate a level—props, characters, textures, etc.) into a level has improved dramatically. This means that a level designer can be working on a map while artists, programmers, and audio engineers all work on content for it, all of which can be imported easily.

12 Game Level Design

This is not to say that designers should seek to break their games into the greatest number of levels possible. Like everything, there is a point at which simply throwing more people at a problem becomes counterproductive. It does mean, however, that identifying and capitalizing on logical breaks in story, gameplay, and visual themes by separating them into levels can help reduce the risk and length of a project.

SUMMARY

This chapter covered the reason that level designers exist today. Having an idea for gameplay is one thing, putting it into practice is another. Level designers oversee the convergence of materials into the final package that players experience. Historically, level design is the extension of early forms of playfield design—from millennia back where game boards were drawn in the sand to the latest in photo-realistic game environments.

Levels have been used in games for many reasons: to allow for larger games, to separate the game experience into narrative or geographical locations, and to allow the team to work on the whole game at once.

INTERVIEW WITH RICHARD “LEVELORD” GRAY OF RITUAL ENTERTAINMENT

Richard, you’ve been making levels for quite a while now, and on a variety of projects. How did you get started as a designer?

Like many of the old veterans, I started with *Doom*. I still remember when DEU (Doom Editing Utility) came out. I downloaded it from CompuServe. The whole time it was transferring over my 256K modem, I was thinking “This can’t be for real. Nobody would let you freely create content for their proprietary game.” I installed DEU, loaded up E1M1, and removed one of the walls. Run the perverted E1M1 in the game and . . . “Oh my god! I can make my own Doom levels!” I was forever hooked!

I then spent every waking hour of the next six months making four new Doom levels, which I uploaded to CompuServe’s Action Forum. These caught the attention of both Q Studios, then working on *Blood* for Apogee, and Apogee themselves. I was hired as a contract level designer by Q Studios and worked for them for almost a year. I was then asked to work on *Duke Nukem 3D* full-time and came to Dallas, Texas, where I’ve been ever since.

→

How has the position of the level designer changed in your opinion, as teams and budgets get bigger?

When I started, the level designer was responsible for many different tasks. These included geometry, asset placement, gameplay, player flow, lighting, balancing, scripting, and some texturing. Now, everything is far more complicated. Most of these tasks are now specialized and performed by one person or subgroup of level designers. The role of the level designer is just as important, the position just requires more people to do it, and they need to be more specialized.

It is very similar to a movie. Watch the credits in a 1930s–1950s movie. They're scrolled by in less than a minute. Now there is enough time to play two or three full songs as the thousands of people's names go by. I expect the game industry will grow in this fashion many times over in the coming years.

Conversely, do you think the quality and sophistication of level design have increased with the scope of games?

Absolutely! The quality and sophistication are incredibly evolved, and that has broadened the level designer's scope. Just reload one of your games from five years ago . . . even two years ago. It's hard to look at it and remember that it was cutting edge in its time.

Do you see any standards emerging in the design and construction of game spaces? Are there tools, or a language, common to level design?

Standards are still ephemeral. They will emerge, such as tool sets and asset pipelines, as one game becomes popular and their way of doing things is in style. For instance, the *Quake* tools were a standard for a long time. Now, many level designers are using sophisticated 3D tools such as 3D Max and Maya for almost all game engines.

When you ask about standards, I presume you mean like in the software industry where engineering disciplines are used such that individuals can bounce from one application, project, or company to another with little re-education. This sort of scale of standardization has not happened yet.

Can you impart some critical lessons you've learned in your career so far?

Yes! . . . making games is not fun-and-games, . . . it's work-and-games! 'Tis true, it's a dream job and I would not replace it with any other career. However, it is not the same as playing games at home and thinking you're the next great game designer. It is also not like sitting at home and making mods and

→

14 Game Level Design

such. It is, in fact, long hours of sometimes very tedious work. It is months, at times, of 12–14 hour days, 6–7 days a week. It is coordinating with the egos of other teammates, and meddling producers and publishers, and the press that at times can be brutal, and fellow developers. Your creative juices are most often sucked into someone else's sponge. There are heartbreaks, with months of work thrown out, never to be seen by anyone again.

For someone wanting to land a level design position today, what sort of steps should they be taking?

There are a few great ways to enter, but you **MUST** make sure you really want to do this. I will warn any poser of this question that simply by asking the question makes me suspicious. Even if I had not started in 1994, you would not have to tell me how to get hired today. The hunger, the true desire, would have me looking under every rock for a niche to fit in.

This is what I would do, this is what I did in 1994: Find a game you enjoy playing that allows access to level editing. Most games do. Make some levels. Play test them well! Get a Web page going to exhibit them. This is a portfolio. Get feedback from people (friends and others that have played your levels). When you think you are polished, start emailing companies and keep an eye out for job offerings.

There are also some very good schools today, if you can afford the tuition. Southern Methodist University, for instance, here in Dallas has a great program setup by many of the leading local developers.