Name: _____ Date: _____

How Are Video Games Made?

Reading level 11 (7th grade) · 1301 words

How are video games made? Start with one moment. You press a button, and a character jumps. In that instant, the game has taken in your press, worked out where everything goes next, drawn a new picture and played a sound. Then it does the whole thing again, dozens of times a second. This book follows that one jump backward: into the engine that runs it, the people who built it, and the stages a game goes through before you can play it.

Most games are built on a game engine, a framework of software made for building video games. It usually comes with ready-made parts for drawing, physics, audio, animation, artificial intelligence and networking. A studio can write its own engine, but many use one that already exists. In the 2026 survey from the Game Developers Conference, 42 percent of developers named Unreal Engine as their main one, ahead of Unity. Godot, another engine, has gained some use with newer indie developers, but less with more established studios.

Inside the engine runs the game loop. The programmer Robert Nystrom writes that almost every game has one, and that no two are exactly alike. He calls it the one pattern his book could not live without. Each turn does three jobs. It takes in what the player did, without stopping to wait. It updates the game by one step, running the AI and the physics. Then it renders, which means drawing the picture so you can see what happened. One turn of the loop is called a tick, or a frame.

How fast does the loop turn? Count the turns in a real second and you have the frames per second. Many screens refresh 60 times a second, which leaves about 17 milliseconds to finish each frame. A millisecond is a thousandth of a second. At 30 frames per second a game has twice as long for every frame, but you see half as many pictures, so the motion looks less smooth. A game that cannot finish in time has to give something up. One guide to web games says such systems lose visual quality to keep running.

There is a trap here. Say a bullet moves the same distance every frame. On a computer that draws many frames, it flies fast. On a slow one, it crawls. Robert Nystrom says a loop must keep the game at a consistent speed on different hardware. One way, a variable time step, scales each move by the real time that passed. Nystrom warns that this has a serious problem of its own: the game becomes less predictable. Glenn Fiedler calls equal slices of time, like 1/60th of a second, the simplest way to move a physics simulation forward. But on some machines, he says, it runs faster or slower than you meant.

What does the loop draw? In many 2D games, things are sprites, which are flat pictures drawn over a larger scene. In a 3D game, an object is a model, and its surface is a mesh: a collection of points, edges and faces. The faces are usually triangles. Either way, an artist

makes the pictures, and the engine redraws them on every frame, while its physics part decides where things are and its audio part plays the sounds. Early on, rough art stands in for the final art in prototypes. More final art is made as production goes on.

Who made the jump you started with? A game designer decided how it should work, and a lead designer coordinates the other designers, who may also write the story. A game artist drew the character, and an art director oversees the art. Programmers wrote the code, and some specialize in physics, graphics, sound or input. Sound engineers handle effects, like the thud of a landing, and where sounds seem to come from. Testers look for defects. Teams can be many hundreds of people, a small group, or even a single person.

Where do the levels come from? A level designer builds them with special tools, and programmers often write those tools. But a computer can build levels as well, a method called procedural generation. It mixes human-made content with computer-made randomness. In the original Spelunky, first released in 2008, every level is a 4 by 4 grid of rooms. Minecraft goes further. Its world generator turns a starting number, called a seed, into a whole world, and the same seed gives the same terrain every time.

What stages does a game go through? Game-production books give a usual order. A concept comes first. Then comes pre-production, when prototypes, which are rough early versions, are written and the plan for the whole game is made. Then comes production, the full-scale work. In alpha, the key gameplay is in but the assets are only partly finished. In beta, the game has all its features and art, and only bugs are being fixed. Studios use these words a little differently, and when a game ships with bugs, most developers quickly release patches.

A real timeline, from the small end. Eric Barone began making games while he studied computer science, starting with really small, simple games. Then he worked on Stardew Valley for at least four and a half years, mostly by himself. He did the code, the music and the art with ordinary programs, and he worked at least eight hours a day by his own count. It came out in February 2016. After its fourth update, he formed a small team to help him keep improving it.

A middle size: Supergiant Games' creative director said the team for Hades was just under 20 people in 2020. The game came out first in early access, in December 2018, after a little over a year of work. Early access means players can buy a game before it is finished and give feedback while it grows. The team had grown from 12 people on its two earlier projects. The director also said small teams are not automatically better than big ones, but that being small makes talking simpler and decisions faster.

Now the big end. Rob Nelson, a producer of Red Dead Redemption 2, said roughly 2,000 Rockstar people worked on it. Guinness World Records lists those years as 2010 to 2018. Take this with care. It is a studio's own statement about one game. Credits, outside help and people moving between projects mean no one outside can give an exact team size, so this is

a round number. Still, set it beside a farming game made mostly by one person, and you can see how far apart the two ends are.

Testing runs through every stage, and bugs are real. In Pac-Man, from 1980, a few top players reached level 256 and found half the screen scrambled. The game keeps its count in one byte, which holds 0 to 255. At level 256 the code that draws the bonus symbols added one, and the count wrapped to zero. The routine then ran far too long. It drew from memory that was not its symbol table. Testers write bug reports that say, if possible, how to make the bug happen again.

So how long does it take, and how do you become a game developer? The answers here run from a little over a year, for the first version of Hades, to the years 2010 to 2018 for Red Dead Redemption 2. Eric Barone, asked how he began, said it was in school, with really small, simple games, and that he worked his way up. Now go back to the jump. A loop took in your press, the game drew a picture and played a sound, and all of it fit inside a sixtieth of a second. Someone spent years so that it would feel like nothing.

Questions

1. What are the three jobs of one turn of the game loop?

A. Draw, save and quit

B. Take in input, update the game, and render the picture

C. Test, sell and patch

2. What does scaling each move by the real time that passed help with, and what is the catch?

A. It keeps game speed steady on different computers, but the game becomes less predictable

B. It makes sprites bigger, but it costs more memory

C. It makes the sound louder, but only on slow computers

3. What does "seed" mean in Minecraft's world generator?

A. A plant you grow

B. A starting number that makes the same world every time

C. A kind of mesh

4. Why does the book call the Red Dead Redemption 2 team figure a round number?

A. Credits, outside help and moving staff make an exact count hard

B. Game studios never hire anyone

C. The number was made up

5. What happened in Pac-Man on level 256?

A. The game ended with a victory screen

B. Half the screen turned into scrambled symbols

C. The ghosts stopped moving

6. According to the book, which stage has all features and art, with only bugs being fixed?

A. Pre-production

B. Concept

C. Beta

Words to know

engine: game framework
frame: one picture
prototype: rough early version
mesh: 3D surface
seed: starting number
beta: nearly finished version

Answer key: How Are Video Games Made?

1. B. Take in input, update the game, and render the picture
2. A. It keeps game speed steady on different computers, but the game becomes less predictable
3. B. A starting number that makes the same world every time
4. A. Credits, outside help and moving staff make an exact count hard
5. B. Half the screen turned into scrambled symbols
6. C. Beta