“Backstory of Video Games”

Amir Alizadeh

Introduction

Have you ever had so much fun in a video game that made you forget your business tasks or your school assignments? Have you ever been enchanted by the beauty and vividness of those views and scenery behind games like *Horizon: Zero Dawn* or *Uncharted 4: A Thief’s End*? Have you ever been interested in making your own games and letting them be played by your friends and maybe by the people all around the world? But the most important question is: Have you ever wondered how these masterpieces are made and what is the “**backstory of video games**”?

If so, I’m here to be your pilot over next four pages to help you answer all the questions above. Our plane takes off from inside of a game studio, ascends to overall knowledge about game engines and finally lands in a point where you know how to kick off making a game.

Back in 1980, video games were considered nothing but toys by most adults, and the software that made them was so specialized to a specific title and optimized for a specific hardware and also too hard to maintain. But now, game has come a long way turning to a multi-billion-dollar industry, one that frequently brings in more revenue than Hollywood movies. And the software behind these fantastic three-dimensional games are either proprietary game engines like RAGE (Rockstar Advanced Game Engine), EA’s Frostbite and Guerrilla’s Decima or publicly licensed game engines like Epic Games’ Unreal Engine 4, Crytek’s Cry Engine and Unity game engine by which developers can build almost any game imaginable. Technically all game engines have some core components in common, including physics and collision detection engine, artificial intelligence system, rendering engine, animation system, audio system and some newly added features like AR/VR engine and so on, but implementation of each one may vary among game engines.

Let’s Get Inside!

Depending on a game project and studio itself, the number of work forces is various, but game studios are usually composed of five main sections: engineers, artists, game designers, producers and management and support staff. We have some subdivisions for each section. Here we take a quick look at each section.

1 – Engineers

From technical perspective, all games are made from 1’s and 0’s. knowing this fact, we can now understand why computer engineers are in the heart of a game studio. However, they may not stand out as much as other sections to the public. Engineers are categorized into two smaller sections: *tool* programmers (who work on making tools for development team to work efficiently) and *runtime* programmers (who work on the game engine and game itself). In addition to this, some other programmers are generalists, they monitor overall development process and tackle potential technical problems of all kinds. Sometimes engineers are promoted to technical leadership role in which they have more responsibilities. The highest position in a game studio related to engineering section is chief technical officer (CTO). CTO is appointed to oversee technical direction of a project and they also take part in decision-making positions.

2 – Artists

All we see in a game visually has started from a basic pencil sketch of an artist, so quality of their work is very important regardless of their category. The subdivisions are: *Concept artists*, *3D modelers*, *Texture artists*, *Lighting artists*, *Animators* and so on.

3 – Game Designers

What they do is simply what we refer to as "Fun & Joy". Their job is mainly to make a game engaging enough to absorb players for a long period of time. While some designers work on the story arc and overall sequence of levels and chapters (high-level goals), others are responsible for more in-game designs like world map geometry, determining how the NPCs (None Playable Characters) should react to players' decisions and predict if the players can act upon enemies, where to place health supplies, weapons, safe points or even traps, designing puzzle elements like what we see in VOID's Pursuit of Redemption or 2D boy's World of Goo.

4 – Writers

Video games tell a story and this story is written by the writers. We all can name games that deeply touched us emotionally, games that we will remember forever. Although not all video games have stories but they all do tell a story. That's why not all games have writers but they all have someone who defines the story that the game wants to tell. Writing games is a lot different from writing books or even film scripts, because in games, the player decides what to do and the writer should be able to write multiple stories for a situation or fork the story. This can be seen in RPG (Role Playing Games) or choice-based games like Telltale's *The Walking Dead* or Quantic Dream's *Heavy Rain*.

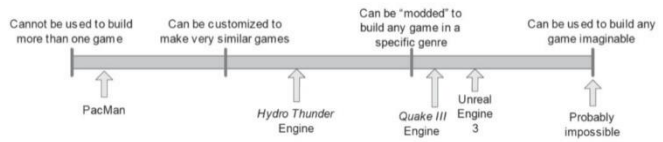
Video Games as Simulations

In most of video games we see the world in pre-built modeled objects which is a simplified yet real form of reality. The reason of simplification is we cannot afford designing every small detail of real objects but it does not necessarily lead to lack of qualification in visual perspectives. The games are also *agent-based* which means that so many in-game objects called *entities* are interacting with each other within the scope of the game. For example, entities like vehicles, characters and so on. Influenced by these relationships, game programmers can rely on object-oriented designing patterns offered by today's programming languages. And finally, video games are *interactive real-time simulations*, that is because they simply interact with users and respond appropriately as well as presenting their own stories.

What is a Game Engine?

The term "game engine" was first used in the mid-1990s. Back then, John Carmack developed the first game engine known as *Doom* engine, the engine behind *Doom II*. The core components of Doom engine were intentionally separate from the game elements and its art assets in order to make it easier for game developers to make their own games based on existing engine components. This led to "reusability" concept of game engines and marked the birth of the "mod community" – a group of game developers and independent studios that built new games by modifying existing ones, using toolkits embedded in base engines. Today, game developers can freely choose and license their game engines according to the key features they offer and also type of the game they have in mind. Although this may involve significant investment in engineering, this is much more economical for independent studios than developing their own engine from scratch.

When a game is being developed, it contains a lot of logic and rules and specific-case code to reach the goal of it, but it is not the case with game engines. When we talk about game engines we should always consider **reusability** and **extensibility** founded by which developers can make different games without major modification.



Although every game engine should be responsive for different games, it does not necessarily mean that every game engines can be employed for all genres. Sometimes a game engine is dedicated for a specific genre and hardware. Therefore, even the most general-purpose game engines are perfectly suitable for building games in one particular genre. However, some companies like EA are integrating their game engines into one, *Frostbite* in this case.

Game Engines & Genres

As mentioned earlier, game engines are game specific. To clarify, a game engine used in a racing game is so much different from its counterpart in a first-person shooter (FPS) game. Nevertheless, all game engines have a lot of tools in common with each other, regardless of genre. To illustrate, all 3D engines need a system to control user inputs from the joypad, keyboard or mouse, a system for rendering 3D models, an audio system and etc.

Your Turn!

Now, our plane has started to descend and approach to **your turn** point! As a game programmer, you gained some necessary information on how a game engine works and the story behind it so far. However, it is not enough for breaking into game industry. Generally, creating a video game is a daunting task and you will definitely face so many challenges and problems from the very beginning to the end of the project, but it is normal and somehow inevitable for game programmers/developers. Honestly, budget plays an important role in this field. Big name companies like Ubisoft, Guerrilla, Rockstar, … dedicate budgets reaching into millions, as well as a group of 500-to-1000 professional designers and programmers to their project in order to bring in the next AAA game to your doorstep. In spite of this, we have seen a lot of video games from small teams with relatively low budget which attract our eyes to their astonishing art and gameplay, in a way that pushes the boundaries of what a "game" can be. Moon Studios' *Ori and the Blind Forest*, servers as an example. Hence, money and a big group of professional people, neither of which seems to be the most important part … if not so what is? -Believe me my friend, "Love" is the answer. By analyzing the sale statistics of indie games in recent years, we understand that it **IS** possible to possess one of the hit title in the market with a group of passionate people with a low budget. However, the purpose of this article is not to admire others success; as a game programmer, I am here to share what I have learned so far with you.

As a Game Programmer

As a game programmer, you are expected to design core features of the game, supervise game testing, collaborate with designers and so on. There are several fields available for a game programmer to work in, including gameplay, production and artistic concept.

Take a look at table below stating career requirements provided by *study.com.*

Degree Level	Bachelor's degree (minimum)
Degree Fields	Computer science or computer engineering
Key Skills	Critical thinking, problem solving, quality control analysis, judgement, decision-making, operations analysis, systems evaluation, advanced computer skills with programming languages, especially C, C++, Perl, Assembly, and Lua
Median Salary (2015)*	$79,530 (for computer programmers, which includes gaming programmers.)

Do not get cold feet if you do not qualify for one or more. It is OK. There are a lot of courses both online and on-campus available for you. Moreover, you can achieve them by actually making games, having some homemade games in your portfolio is always considered an extra bonus on part of employees. Keep in mind that a game programmer should also have strong knowledge of at least one modern game engine/framework regarding to the project for which they have been hired. It is highly recommended to master a programming language, preferably C++, because C++ is arguably the most commonly used language in the game industry. However, you can count on other object-oriented languages like C# or Java. In addition to these high-level languages, every programmer should also learn at least some *assembly language* programming. Mathematics are vital when we talk about game programming, because, as you might know, game is a virtual mathematical form of reality.

Get the Ball Rolling!

If you have reached a basic skill level according to what explained earlier, it is time to get your hands dirty and start making a game. In my point of view, a step-by-step guide could be like:

1-Team up: you have to find your men who are specialized in other parts of the game development and sign a simple contract with. As a rule of thumb, it is always better to find the people who are in the same stage of professionality as you to prevent further problems, additionally, it is a good idea to enhance your skills and learn together.

2-Idea: you have to set some brainstorming meetings in which you agree on the idea of your game. It is critical to know:

- The type of game you want to make (i.e. a platformer, shooter, RPG).
- The budget
- The overall goal of your game (details can be added later)
- The local or global market (App Store or Google Play) – in Iran, most of developers release their games on Cafe Bazaar
- Your skill level

3-Engine: this is a bit tricky and controversial step. Game engine of your game is like foundation of your building, so, it is so much important to choose the right one. There are quite a lot of game engines available out there and since this list goes on and on, it might be confusing to know what to look for when using them. I would strongly suggest you to start off by looking at Unity, Unreal Engine 4, Game Maker Studio, these are all free engines with some similarities but also big differences. I personally prefer Unity, because it is optimized for both 2D and 3D. Do not hurry up! Just take your time and work around the game engine you think best suits you.

4- Production: after doing all the steps, you are ready to commence development phase.

5- Quality Control: after producing a playable version (Beta) of your game, ask your friends or professional testers to play your game and comment on it. In this phase, you can analyze behavior of your players and use them to your advantage by adding some **data mining** features to your game logging some information online. From technical perspective, spotting bugs and crashes is possible in this phase.

6- Publication and Marketing: depending on the quality of your final build and its business canvas, you can negotiate with some publishers and persuade them to take care of marketing and financial side of your project. It has been proven that the more famous your publisher, the more chance you get to hit the nail on the head!

Welcome to Game Industry

As I promised, our plane is getting prepared to land in a point where you know how to kick off making a game. You have a hard way ahead but this way is full of fun and joy, just lay back and enjoy the game!

Amir Alizadeh

2017-2018

References

- Game Engine Architecture by Jason Gregory (here is the link)
- study.com (here is the link)
- digitaltrends.com (here is the link)