Lecture 2

Game and Development Platforms

Assoc. Prof. Dr. Burkay Genç

29 Şub, 2024

Game Platforms

Types of Game Platforms

- There are three major platforms to play games

    - Console

    - PC

    - Mobile / Tablet

- There are also minor, mostly obsolete or not very popular platforms

    - Handheld

    - VR

    - Smart watch

Console

- This is the oldest gaming system
    - Remember Brown Box (Ralph Baer) from prev week's slides

    - 1967-1968
- Aimed at families
- Has games for all age levels
- Moderate pricing
- More stable and expected gameplay
- Doesn't have many options

PC

- This is the medium aged system
  - 1971 Kenbak-1 (not really what you expect from PCs nowadays)

  - 1980 Micral 80-20 (A bit more like a classic PC)

PC

- Mostly aimed at medium-advanced users
- Has a very large spectrum of games
- Moderate to expensive pricing
- Can be unstable at times
- Extreme variation in performance
- Has an almost unlimited amount of configurations

Mobile / Tablet

- Most recent addition to the group
  - Although they were invented earlier (1992), reached masses in 2000s.
- Aimed at casual gamers
- Cheap to expensive (very large spectrum)
- Mostly stable and expected game play
- Moderate variation in performance
- Has many configurations available

Consoles

What is a console?

A console is an electronic device that outputs a video and audio signal that can be channeled to a display device (mostly TVs), and controlled by one or more specific game controller units, used to play digital games.

Advantages

- Designed and built for gaming
- Minimal variation among different sets
    - Results in stable gaming
- You know what you are buying, no surprises
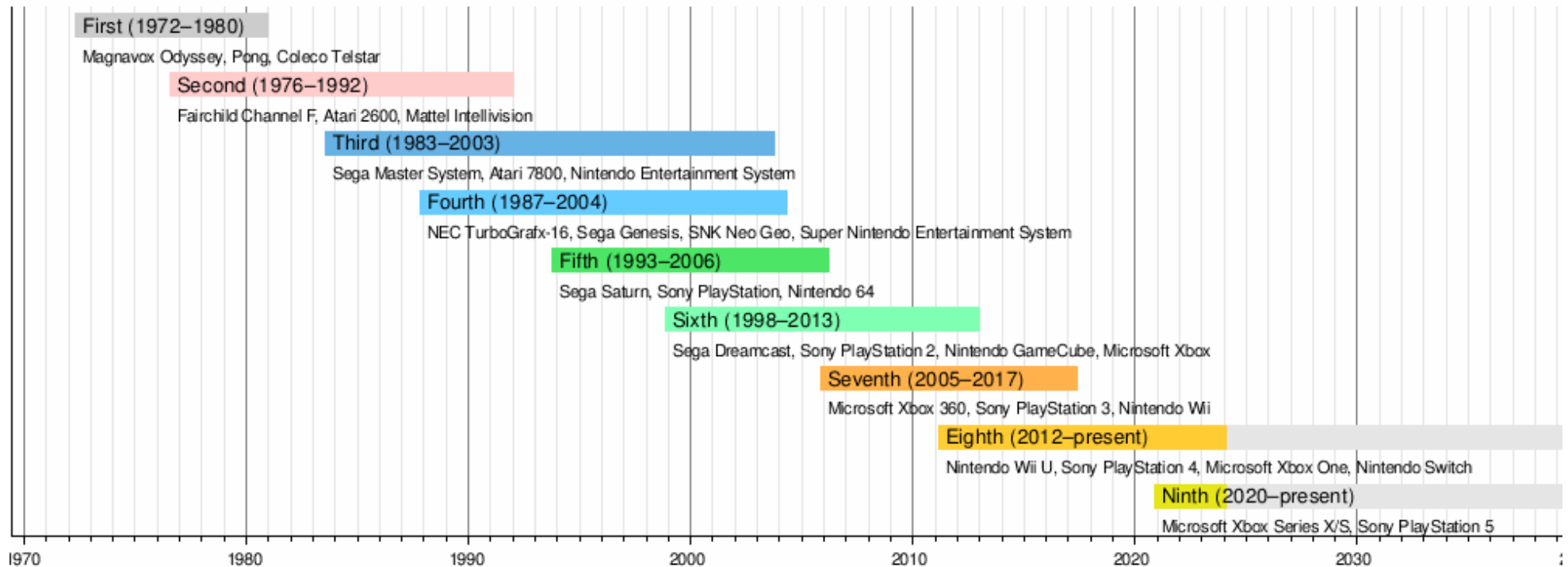- Recently, converged to PCs

Disadvantages

- Very limited options

- Missing a lot of functionality served by PCs and mobiles

- Manufacturers keep console prices low and later make profit from games licensing fees

	PC	Console	Savings %
Control	9	31	0,70967742
Tekken 8	51	65	0,21538462
Remnant 2	25	36	0,30555556
GTA 5	8	21	0,61904762
Witcher 3	10	31	0,67741935
Last of Us Part 1	32	56	0,42857143
COD MW	16	20	0,2
Amored Core 6	45	44	-0,02272727
Resident Evil 4	30	45	0,33333333
Dead Space Remake	24	39	0,38461538
Elden Ring	28	37	0,24324324
Persona 5 Royal	27	28	0,03571429
Cyberpunk 2077	30	51	0,41176471

- Cannot be upgraded

- Controllers can be annoying for some people

Generations

- Consoles are usually attributed to the so called *generations* of 5-7 year periods

First (1972–1980)
Magnavox Odyssey, Pong, Coleco Telstar

Second (1976–1992)
Fairchild Channel F, Atari 2600, Mattel Intellivision

Third (1983–2003)
Sega Master System, Atari 7800, Nintendo Entertainment System

Fourth (1987–2004)
NEC TurboGrafx-16, Sega Genesis, SNK Neo Geo, Super Nintendo Entertainment System

Fifth (1993–2006)
Sega Saturn, Sony PlayStation, Nintendo 64

Sixth (1998–2013)
Sega Dreamcast, Sony PlayStation 2, Nintendo GameCube, Microsoft Xbox

Seventh (2005–2017)
Microsoft Xbox 360, Sony PlayStation 3, Nintendo Wii

Eighth (2012–present)
Nintendo Wii U, Sony PlayStation 4, Microsoft Xbox One, Nintendo Switch

Ninth (2020–present)
Microsoft Xbox Series X/S, Sony PlayStation 5

1970   1980   1990   2000   2010   2020   2030

Loading games

- Traditionally consoles used cartridges to switch games
- Nowadays you can download games from the Internet
    - First generation -> dedicated to a single game
    - Second to fourth generation -> cartridges
    - Fifth and sixth generation -> CDs
    - Seventh+ generation -> Internet
    - Eight+ generation -> Cloud gaming

Pricing

Console release prices (in U.S. Dollars) and total sales[49][50]				
Console	**Release year (U.S.)**	**Introductory price (U.S.)**		**Global Sales (Units)**
		Originally[note 1]	**2020 inflation**[note 2]	
First generation				
Magnavox Odyssey	1972	$100	$553	350,000[51]
Second generation				
Atari 2600	1977	$200	$882	30,000,000
Intellivision	1979	$300	$996	3,000,000
Atari 5200	1982	$270	$740	1,400,000
Colecovision	1982	$175	$480	2,000,000
Third generation				
NES	1985	$200	$490	61,900,000
Atari 7800	1986	$150	$380	3,770,000
Master System	1986	$200	$470	13,000,000
Game Boy	1989	$110	$234	118,690,000
Fourth generation				
TurboGrafx-16	1989	$200	$426	5,800,000
Genesis	1989	$190	$405	30,750,000
SNES	1991	$200	$384	49,100,000
CD-I	1991	$400	$768	1,000,000
Neo Geo	1991	$650	$1248	980,000
Sega CD	1992	$300	$561	2,240,000

Pricing

Fifth generation				
Atari Jaguar	1993	$250	$453	250,000
3DO	1993	$700	$1267	2,000,000
32X	1994	$160	$282	665,000
PlayStation	1995	$300	$516	102,490,000
Sega Saturn	1995	$400	$688	9,260,000
Nintendo 64	1996	$200	$334	32,390,000
Sixth generation				
Dreamcast	1999	$200	$314	9,130,000
PlayStation 2	2000	$300	$459	155,000,000
GameCube	2001	$200	$294	21,740,000
Xbox	2001	$300	$441	24,000,000
Game Boy Advance	2001	$100	$147	118,690,000
N-Gage	2003	$300	$416	3,000,000
Seventh generation				
Nintendo DS	2004	$200	$278	154,020,000
PlayStation Portable	2004	$250	$348	82,000,000
Xbox 360	2005	$400	$540	84,700,000
PlayStation 3	2006	$500	$680	87,400,000
Wii	2006	$250	$326	101,630,000

Pricing

Eighth generation				
Wii U	2012	$350	$399	13,560,000
Nintendo 3DS	2011	$250	$293	75,280,000
PlayStation Vita	2012	$250	$293	15,900,000
PlayStation 4	2013	$400	$448	117,200,000[note 3]
Xbox One	2013	$500	$560	51,000,000 (Estimate)
Nintendo Switch	2017	$300	$318	122,550,000[note 3]
Current				
PlayStation 5	2020	$400 / $500	$400 / $500	50,000,000[note 3][52]
Xbox Series X/S	2020	$300 / $500	$300 / $500	18,000,000 (Estimate)[note 3][note 4]

Game Development for the Console

Game Development Kit

- To be able to develop a game for a console, you must have a *Game Development Kit*

- Game development kits (GDK) are specialized hardware and software used to create commercial video games for game consoles.

- They may be partnered with game development tools, special game engine licenses, and other middleware to aid video game development.

- GDKs are typically not available to the public, and require game developers to enter an agreement, partnership, or program with the hardware manufacturer to gain access to the hardware.

Xbox One - GDK

Microsoft maintains multiple developer programs for people wanting to develop games for their platforms

- The *ID@Xbox* program allows qualified game developers to self-publish their games to the Xbox One, as well as access free middleware and use two development hardware kits for free.

- The *Windows Dev Center* allows developers to create apps and games on Windows 8, Windows 8.1, and Windows 10 platforms as part of the Universal Windows Platform system.

PlayStation

- The PlayStation developer program allows registered developers to publish their games across the PlayStation Network, making their games accessible on the PlayStation 3, PlayStation 4, PlayStation Vita, and PlayStation TV all through one program.

- The PlayStation 4 development kits were known as *Orbis*, though this was just a codename.

- Academic institutions can register to receive PS4 development kits for educational use, and are not region-restricted unlike regular PlayStation Developer Program members.

PS5 DevKit Video Link

Personal Computer

What is a PC?

- A personal computer, often referred to as a PC, is a computer designed for individual use, hence the name "personal computer".

- It is typically used for tasks such as word processing, internet browsing, email, multimedia playback, and gaming.

- Personal computers are intended to be operated directly by an end user, rather than by a computer expert or technician.

- A PC usually comes along with a keyboard, a mouse and a monitor.

- PC monitors are specially designed for long hours of usage in close distance.

Advantages

- A PC can be uniquely built by an end user, allowing maximum configuration variance.
    - Hence, it can be arranged according to your budget.
- Laptop's are mobile PCs and can be easilt carried around.
- PCs are the most powerful computing/gaming devices in the market.
- You can easily upgrade a PC when the need arises
- Keyboard and mouse combo is way more usable than a single controller

Disadvantages

- Dealing with a PC requires a bit more knowledge than a console
- Especially, to build one, you must be confident in computer hardware choices
- A cutting edge PC build can be many times more expensive than a console
- PCs can be highly varied in their gaming performances
- Some may even completely fail to execute certain games
- PC software can also greatly vary, affecting gaming performance

Loading games

- Traditionally PCs used diskettes and discs to load software

    - 8" Diskette -> 230kb

    - 5.25" DS Diskette-> 360kb

    - 3.50" DS HD Diskette -> 1.44mb

    - CD Compact Disc -> 600-700mb

    - DVD Digital Versatile Disc -> 4.7gb or 8.5 (Dual)

- Nowadays downloading from the Internet or cloud services is the norm

    - Steam

    - Epic Games Store

    - Gog

Developing for the PC

Hardware requirements

- One can develop and play a game on the same hardware.

- One can also develop games for other configurations of PCs.

- You don't need a special development kit.

- However, developing for one hardware does not guarantee that the game will work on all hardwares.

- Moreover, the performance will be affected significantly.

Software requirements

- There is no *must* software for PC game development
- One can use any programming language to develop games
- One use any library
- One can use any game engine
- However, not all programming languages, libraries or engines are suitable for all types of games

Programming Languages

- *Assembly* is one of the oldest programming languages

- It is as low level as it goes

- Very difficult to learn and master

- Requires tens of lines of code to do the simplest things

- However, once properly mastered, it is the fastest you can get

- Also optimizes space requirements

- Roller Coaster Tycoon is fully written in assembly

Programming Languages

- *C* is one of the most used game development languages
- It is a nice blend of fast and readable
    - More readable than assembly
    - Much faster than Java, Python etc.
- Only main issue is the lacking of object oriented design
- Doom was one of the most popular games ever written in pure C

Programming Languages

- *C*++ is C with added OOP

- It is the de facto programming language if you are going to make a AA or AAA game

- It is fast, easy to read, and fully OOP

- Most gaming libraries are written in C++

- Many game engines prefer C++

  - Unreal

  - Lumberyard

  - Godot

  - Cryengine

Programming Languages

- *Java* is similar to C++, but it is interpreted, not compiled.

  - Runs in a virtual machine

  - This makes it much slower

  - But also much *tamed*

- Much better readability and smoother learning curve compared to C and C++

- *Minecraft* the best selling game ever, is written in Java

- Java is also the main programming language for the Android ecosystem

- Java is extremely adaptable to different platforms. One code runs *almost* everywhere.

Programming Languages

- *C#* is Microsoft being Microsoft and creating its own language

- Easier than C++, faster than Java (is the claim)

- For gaming used as a script language

    - The most popular engine -Unity 3d- uses C# as the scripting language

- Has a large community, and Microsoft behind it

Programming Languages

- *Javascript* is the scripting language of the web

- If you want to write a game for the web, you will most likely deal with Javascript

- It is not the perfect language, however.

    - Terribly slow,

    - Terribly designed,

    - Prototype hell

Programming Language

- *Lua* is yet another common language among game developers

- It is used as a scripting language

- It is extremely lightweight and can be embedded into other applications

    - this makes it a good choice as an *extension/mod* language

- Love2D game library works with Lua

Programming Languages

- Python is not a popular game development language

    - But it is popular everyhere else!

- Easy to learn, easy to use

- PyGame library allows quick game development

- Helps with learning GDscript, which the language of the Godot engine

Game Libraries vs Game Engines

.

Game Engines

What is a game engine?

- A game engine is a collection of tools, editors and language libraries to help with game development
- Consists of many components
    - Graphic renderer
    - Physics engine
    - Audio engine
    - AI engine
    - Scripting
    - Networking
    - Scene editor
    - Blueprint / no-code development
    - Asset store
    - Input control …

When to use a game engine

- You have a small team
- You are not an expert on game development
- You have limited resources (time and money)

When not to use a game engine

- Efficiency and optimisation matters
- You have the manpower to make your own renderer, physics engine etc.
- You don't want to transfer a quotient of your earnings to another company

History of game engines

- Game enginees got popular after 90s
- Before that, every game had to be coded separately
- Very few code was reusable
- Atari 2600 games were coded in Assembly
  - Almost each game has a separate code base

History of game engines
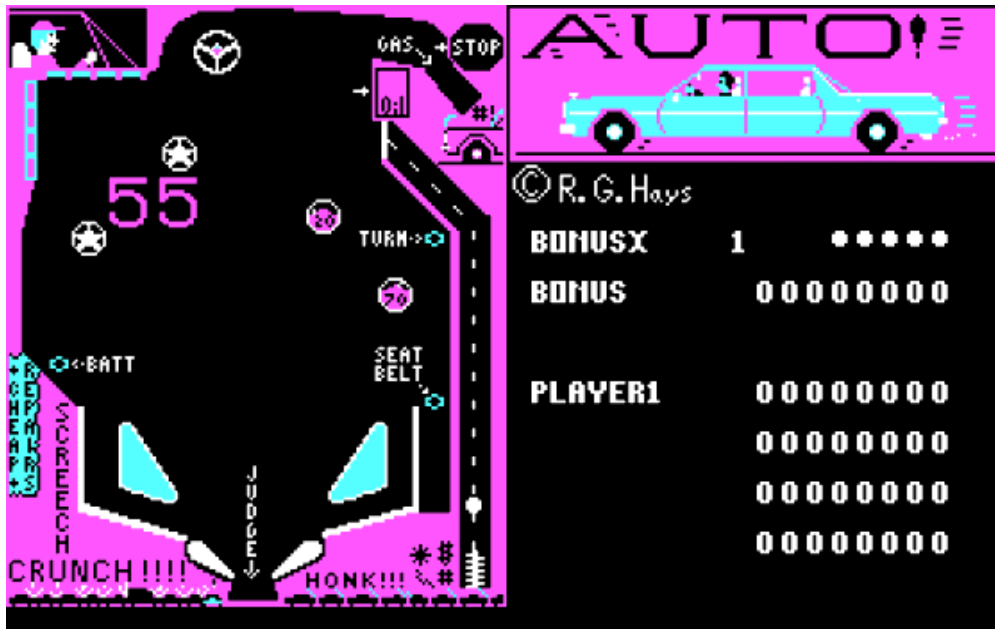
- The '80s saw the release of several 2D game development kits – known as game creation systems
  - the Pinball Construction Set (1983),
  - the Adventure Construction Set (1984), and
  - the Shoot-'Em-Up Construction Kit (1987)
- These are early examples of the idea of a game engine.
- Each kit was specialised for a specific type of game

History of game engines

- The first example of code reuse was Super Mario Bros in 1985
- Shigeru Miyamato reused the code for smooth scrolling they have written in Excitebike (1984)
- This way Mario could smoothly move and even be accelerated.

History of game engines

- Garry Kitchen's GameMaker (1985)
    - released by Activision for various home computer systems,
    - was the first integrated development environment for making games,
    - can be considered a proto-game engine
- GameMaker allowed users to
    - create background graphics,
    - movable objects known as sprites,
    - sound effects and
    - in-game music,
    - also included a game programming language that allowed the developer to code more features.

Video Link

History of game engines

- The first game engine as we understand the term today was a code base created by ID software for the production of a new game trilogy for their Commander Keen franchise.

- From the outset, id recognised that creating a single piece of software that provided common functionality for multiple games was a more significant accomplishment than making a game, and even tried to license the so-called *'Keen engine'* (1991), but met with little success.

Video Link

History of game engines

- The first game engine as we understand today is the Doom engine.
- John Carmack and John Romero of ID named the software a game engine as they were car enthusiasts.
- The engine was a big hit and many companies used it to create their own games:
    - Heretic (1994), Hexen: Beyond Heretic (1995) and Strife: Quest for the Sigil (1996)

History of game engines
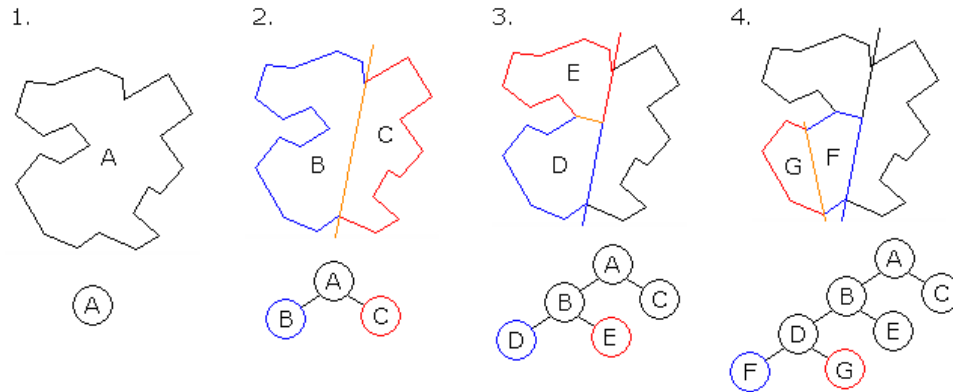
- The Doom engine was originally crawling to a halt on slow machines
- Carmack then read scientific papers on how to solve this problem and discovered the concept of Binary Space Partitioning
    - He used this technique to only render important items on the screen
    - And *cull* (not render) things that are not visible or distinguishable
- This made the Doom engine a very fast renderer for its time

Doom Engine

- Although it looks 3D, the world of Doom consists of 2D sprites
- The Doom engine arranges the sprites to look 3D
    - Closer sprites are larger, while far away ones are smaller

History of game engines

- Ken Silverman took cues from Doom to create the Build (1995) engine at the age of 19.
- The Build engine refined the illusion of 3D with the ability to look up and down
    - Duke Nukem 3D (1996)
- Silverman also added tags to parts of the game world
    - use for teleporting the player

Video Link

History of game engines

- Soon after Doom, ID released the Quake engine (1996)
    - True 3D real-time rendering
    - Support for 3D hardware acceleration
    - Quake (1996), which was made with the engine, is yet another milestone in gaming.

Video Link

History of game engines

- Bethesda had earlier attempted a true 3D engine of its own, called XnGine (1995)

  - but it suffered from bugs and stability issues.

- Eventually, Bethesda achieved success with the engine by creating the huge, procedurally-generated world of The Elder Scrolls II: Daggerfall (1996)

  - They later abandoned XnGine for NetImmerse (1997), the predecessor to Gamebryo.

History of game engines

- The developers at ID then outdid themselves with id Tech 3 (1998)
    - Medal of Honor: Allied Assault (2002) and Call of Duty (2003) uses this engine
- It featured next-gen graphics using shaders
- The engine was one of the first to allow for curved surface geometry as well.
- All these innovations – especially the shader system – were computationally expensive and would have crippled the engine's renderer, which is why the developers implemented the now legendary *fast inverse square root algorithm*.

Video Link

History of game engines

- Quake III Arena was challenged by a relative newcomer – Epic Games' Unreal Tournament (1999).
- Epic's multiplayer title was made with the Unreal Engine,
    - First used to create Unreal (1998),
- The engine itself surpassed id's technology in key areas, especially graphics

Video Link

Unreal Engine

- Initially, UE was designed for software (CPU-based) rendering, but would later be able to make use of dedicated graphics hardware.

- From the outset, it used 16-bit colour and supported visual effects such as

  - soft shadows,

  - dynamic lights,

  - volumetric fog

  - texture filtering

- These were features missing in id's Quake Engine

Unreal Engine

Other effects that Sweeney would dream up created room for even greater visual realism. Sweeney, an avid runner and in-line skater, remembers, "I was out on a six-mile run late at night after a rain storm, with the streetlights casting luminous spheres of light in the fog. I was thinking, 'Gee, I should be able to program that effect.' But when I got back to the office, I second-guessed myself. A few months later, on a whim, I spent a couple of hours whipping up code for this, and to my amazement, it worked perfectly." Sweeney's efforts in this area draws more praise from Carmack: "Light blooms [the spheres of light], fog volumes, and composite skies were steps I was planning on taking, but Epic got there first with Unreal."

Unreal Engine

- By late 1999, about sixteen projects had been developed using Epic's engine,

    - Deus Ex (2000), The Wheel of Time (1999).

- Unlike id Software, whose engine business only offered the source code, Epic provided support for licensees and met with them to discuss improvements to its game development engine.

Unreal Engine

- Unreal also provided tools that were more user-friendly for non-engineers, a crucial factor for the Deus Ex developers, whose team included many designers and whose goal for the game was to go beyond the standard FPS.

- Today, Unreal is still known for creating user-friendly tools that greatly extend the capabilities of developers in creating cutting-edge games.

- The current version of the Unreal Engine has been hailed as a game changer because of its feature set, is free to use and download (there are premium licensing options as well) and is known for its forgiving learning curve.

- The engine is free for internal projects,

    - Users mustpay a 5% royalty fee if and when their product earns over $1 million.

Unreal Engine

- Unreal Engine 4 introduced the *Blueprints* system,
  - a versatile visual scripting language that allows developers to prototype game elements and gameplay mechanics quickly by connecting nodes and other basic building blocks.
- April 2022, UE 5 features a host of innovations
  - Nanite
  - Lumen
  - World Partition System

Nanite

- This is a *virtualised geometry system* that speeds up the creation of LODs (levels of detail).

- In any game, objects are rendered at decreasing levels of detail based on how far they are from the player (or how important they are in a game environment).

- Before Nanite, developers had to author LODs by hand (the engine would then use the appropriate LOD for an object based on the player's position),

- Nanite allows one to import high-quality assets that are automatically rendered at the correct level of complexity with respect to the player's point-of-view.

Video Link

Lumen

- Lumen is used to manage and rapidly update scene lighting
    - scenes change to reflect time of day accurately,
    - new light sources are immediately integrated into the overall lighting profile,
    - sudden flashes of light entering the shot affect the scene realistically.

Video Link

World Partition System

- This utility greatly enhances the Unreal Engine's functionality in terms of building open worlds.

- The system uses a grid to map an entire universe into manageable sub-level chunks, which can be loaded and unloaded as a player traverses the landscape.

Unity

- Unity is the brainchild of three developers

    - Nicholas Francis, Joachim Ante, and David Helgason

- The company was founded in Copenhagen and the engine began life as a graphics tool for Mac OSX

- The team then recruited a diverse group of developers, all of whom brought their own ideas to the table and the result was an engine applicable to a variety of use cases.

- Immediately, commercialising the product became a priority,

    - Especially for use by indie developers

Unity

- The creators of Unity felt that their engine would foster creators in the indie game space and democratise game development.

- They released their first version in 2005,

  - won an award from Apple for 'Best Use of Mac OSX Graphics' the next year.

- The developers soon provided support for Windows and browsers

  - Still no support for Linux, though

Unity

- It was in 2008 that Unity skyrocketed in popularity,
    - when it became the first engine to provide support for Apple's new App Store
- The engine rapidly rose to prominence as iPhones became ever more popular.
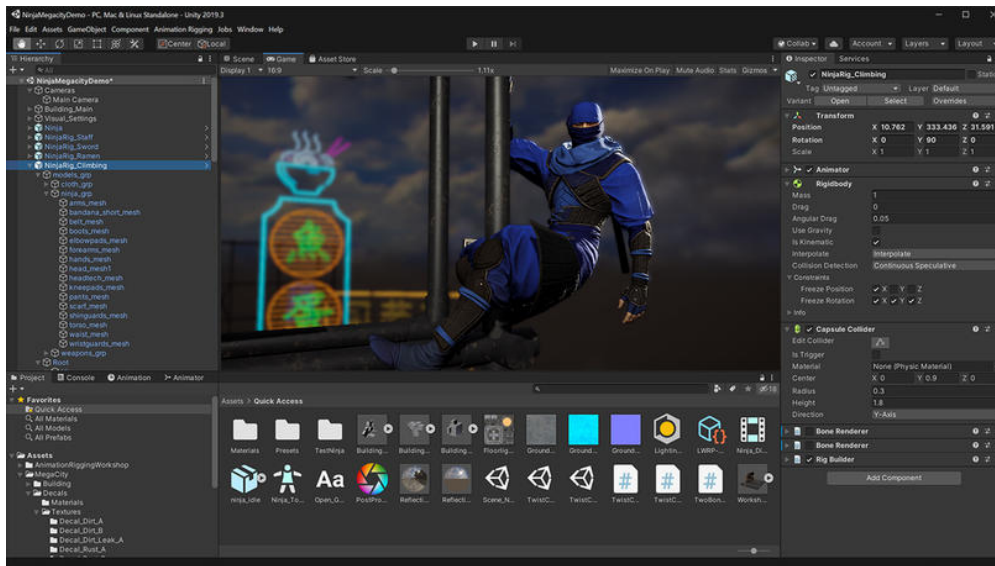
Unity

- Unity 3.0 (2010) featured Android support,
    - provided advanced graphics features for desktop and console platforms,
    - high-end development tools to indie game makers at very affordable rates
- Unity has continued to deliver high-end tools to developers who would otherwise have no access to them,
- By 2018, CEO John Riccitiello claimed that half of all the games on the market were made using Unity.

Unity
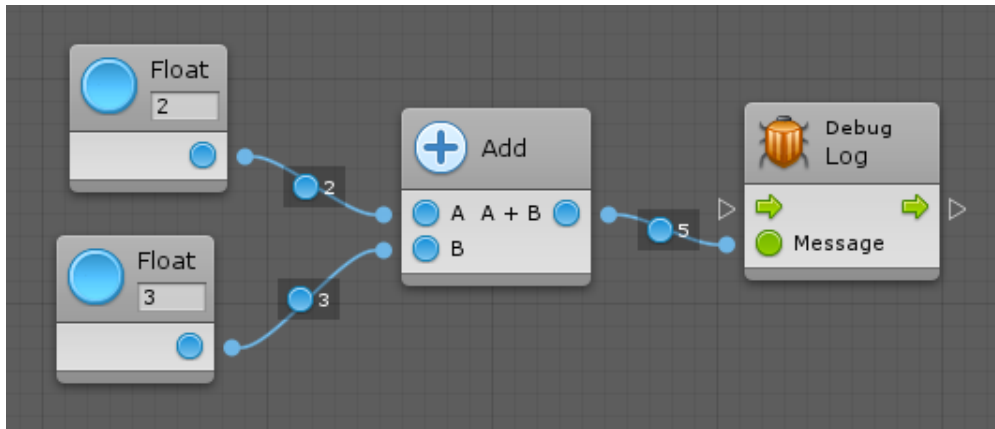
- Unity now supports over 25 platforms

- Unity has an advanced physics engine,

- Integrates with Autodesk asset-creation tools like Maya and 3DX Max

- Matches most of what UE offers

  - recently, it introduced the Bolt system to compete with Unreal's Blueprints

  - However, while even novice programmers can use Blueprints to develop the entire logic of a game, Bolt's functionality does not extend that far.

Unity

- Unity has a massive community of users, and learning it is a breeze because of the wealth of tutorials and guides available online

    - it is known for being beginner-friendly.

- Like UE, Unity is the engine behind a huge number of games

- It enables the creation of 2D, 3D, VR and AR content

Unity

- The engine is still considered best for designing mobile games
    - the fastest-growing market in gaming
    - it was ranked as the most popular for mobile game development
- By 2020, Unity had a 50% market share in mobile game development
    - a 60% share in VR and AR game development.