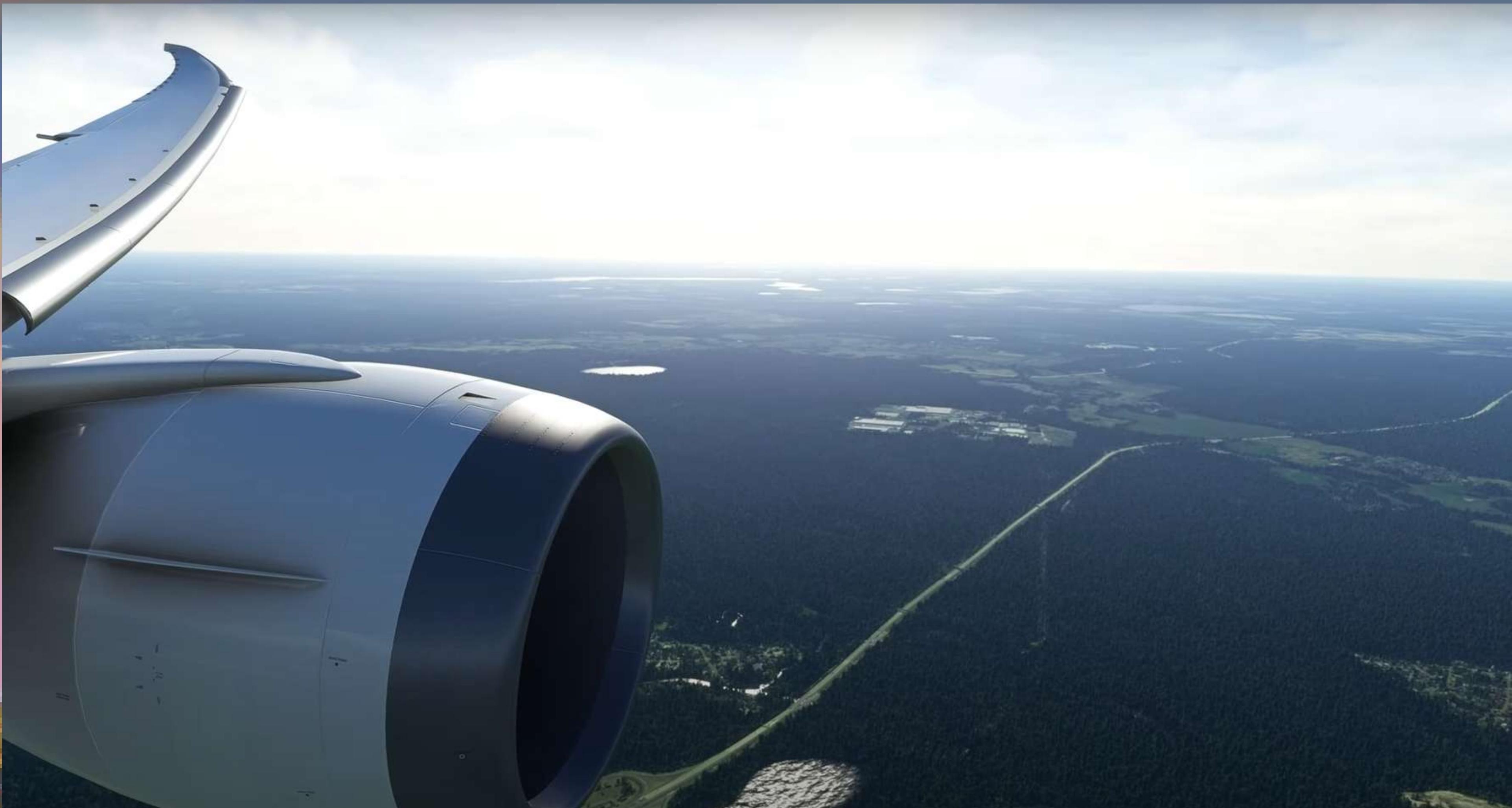


# Extended Realities

Introduction to realtime & XR

MA Creative Technologies  
27.10.2023

# INSANE REALISM Landing At Berlin Airport | 4K



<https://youtu.be/hsKJ3yOZRKI?feature=shared>

# Hong Kong and mainland China gamers clash on GTA V

① 23 December 2019



A player dressed as a Hong Kong protester on GTA V

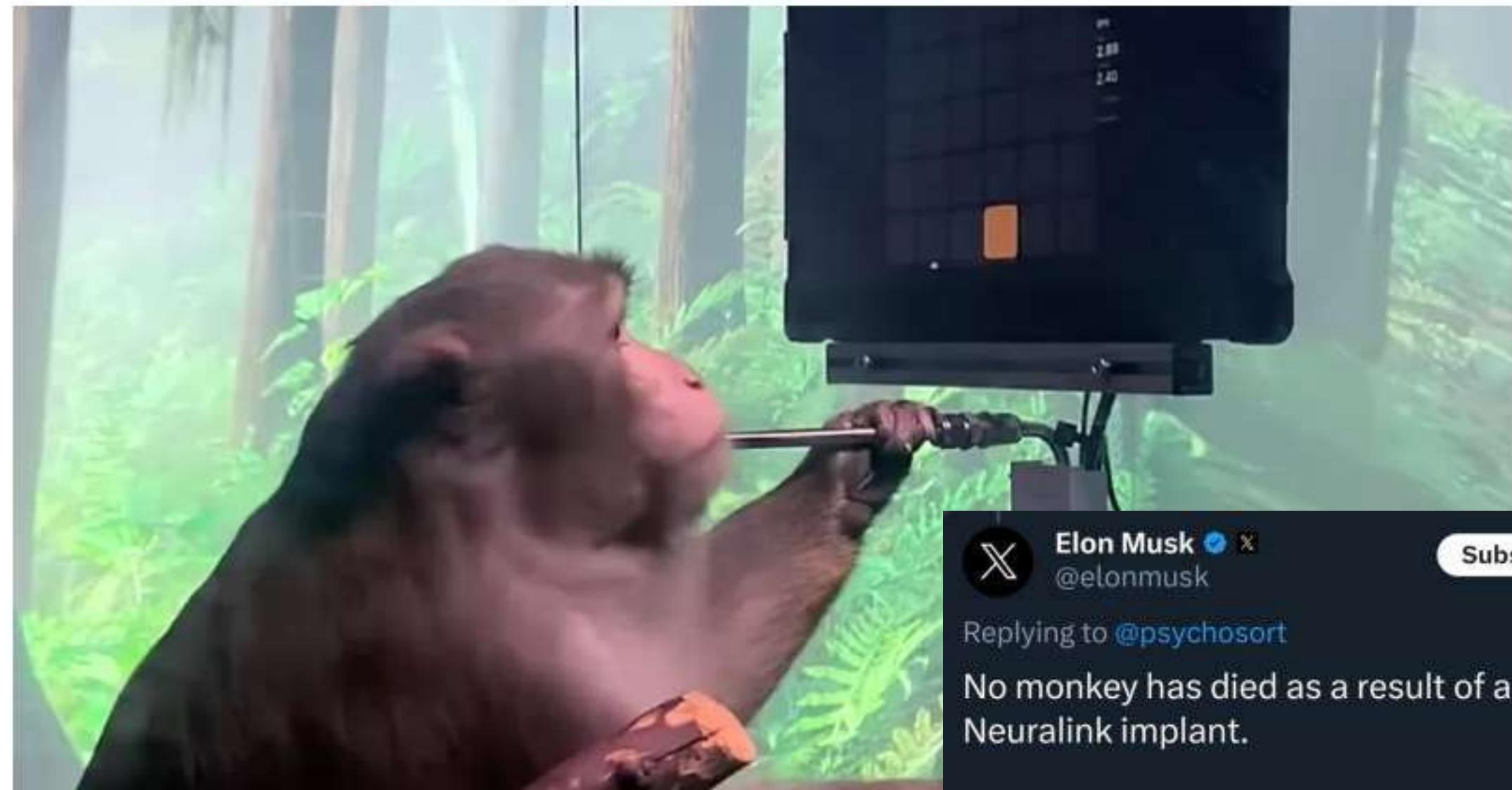
**The Hong Kong protests are being played out on Grand Theft Auto (GTA) V online.**

Players in Hong Kong realised they could dress up as protesters after a new update for the game was released earlier this month.

They spread the word on LIHKG - which has been called the Hong Kong

# Elon Musk's Neuralink 'shows monkey playing Pong with mind'

9 April 2021



Pager first played Pong with a joystick, rewarded with a smoothie via a meta

<https://www.bbc.com/news/technology-56688812>



Elon Musk  
@elonmusk

Subscribe

Replying to @psychosort

No monkey has died as a result of a Neuralink implant.

First our early implants, to minimize risk to healthy monkeys, we chose terminal moneys (close to death already),

6:57 AM · 10 Sep 23 · 147K Views

237 Retweets 110 Quotes 2,376 Likes

83 Bookmarks

# Outline of topics

## COVERED:

- Background and overview of the notions of "realtime" and "extended reality" and how they relate to the modern media landscape
- Overview of technologies available to build realtime experiences
- Introduction to the Unity game engine
- Design criteria/challenges when building VR/AR experiences using game engines
- Basic game engine programming
- Getting started with Oculus Quest VR headsets

## NOT EXPLICITLY COVERED:

- 360-degree film/animation
- TouchDesigner and other non-game engine realtime creation environments
- Hardware beyond Oculus Quest (but feel free to try out some of the other devices that are available in CTech for VR/AR experiences!)
- More advanced game engine functions (e.g. physics programming, shaders, various advanced programming patterns, tools for optimisation etc.)

# Schedule

DAY 1	DAY 2	DAY 3	DAY 4
<b>MORNING</b> <ul style="list-style-type: none"><li>• Introduction to realtime and XR</li><li>• Overview of technologies and available tools</li></ul> LUNCH	<b>MORNING</b> <ul style="list-style-type: none"><li>• Setting up a scene for XR development</li><li>• Troubleshooting installation and setup</li></ul> LUNCH	<b>MORNING</b> <ul style="list-style-type: none"><li>• Prototyping personal projects</li></ul> <p>HOMEWORK: Continue working on the prototype, iterating over wireframe model</p>	<b>MORNING</b> <ul style="list-style-type: none"><li>• Finishing up on personal projects and testing others</li><li>• Evaluating the process</li><li>• Directions for further learning and research</li></ul>
<b>AFTERNOON</b> <ul style="list-style-type: none"><li>• Introduction to the Unity game engine</li><li>• Game engine architecture</li><li>• Basic game engine programming patterns</li></ul>	<b>AFTERNOON</b> <ul style="list-style-type: none"><li>• Topics in VR and AR</li><li>• XR design criteria and challenges</li><li>• Thinking about personal projects</li></ul>		
HOMEWORK: Install all necessary frameworks for XR development	HOMEWORK: Storyboard/Wireframe/Design document for XR experience		

# Personal projects

## AIM:

Get to know the tools and workflow for realtime XR development and develop a conceptual understanding for the limits and possibilities of the medium. The outcome will be a prototype, not a polished experience for exhibition/publication.

## GUIDELINES:

- A theme/prompt will be provided (however, feel free to develop your own concept if you wish)
- Prototype will have to fulfill a minimum set of criteria in terms of features/interactions/code etc.
- Must be developed within the two workshop sessions + 10 hours work outside the class

## PROCESS:

- Background research: can be in the form of articles, a mood board, concept art, design document etc.
- Wireframe/Storyboard: basically a rough visual representation of what you want to prototype
- *1st iteration of prototype*: setting up workflow, making sure everything is linked together properly and that you can already test the experience on your target device.
- *2nd/3rd iterations of prototype*: building out experience, adding content, bug fixing, tweaking interactions

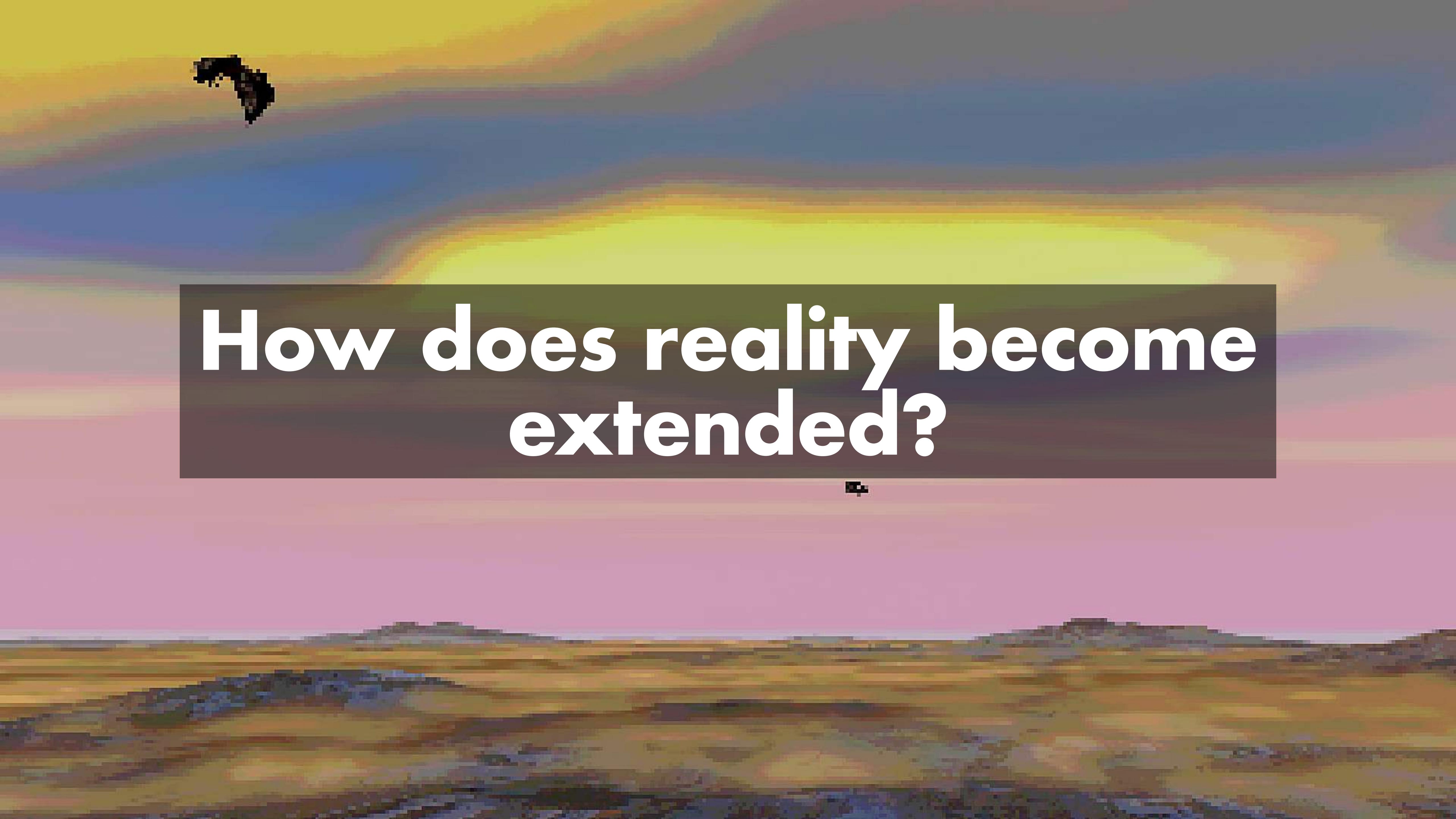
# Evaluation

Deadline:

08.12.2023

To submit: wireframe/storyboard, video walkthrough demonstrating features, unity project file (if applicable)

Evaluation is on the basis of realisation of the proposed concept, the quality of implemented features, and overall aesthetic interest



# How does reality become extended?

# How is reality established?

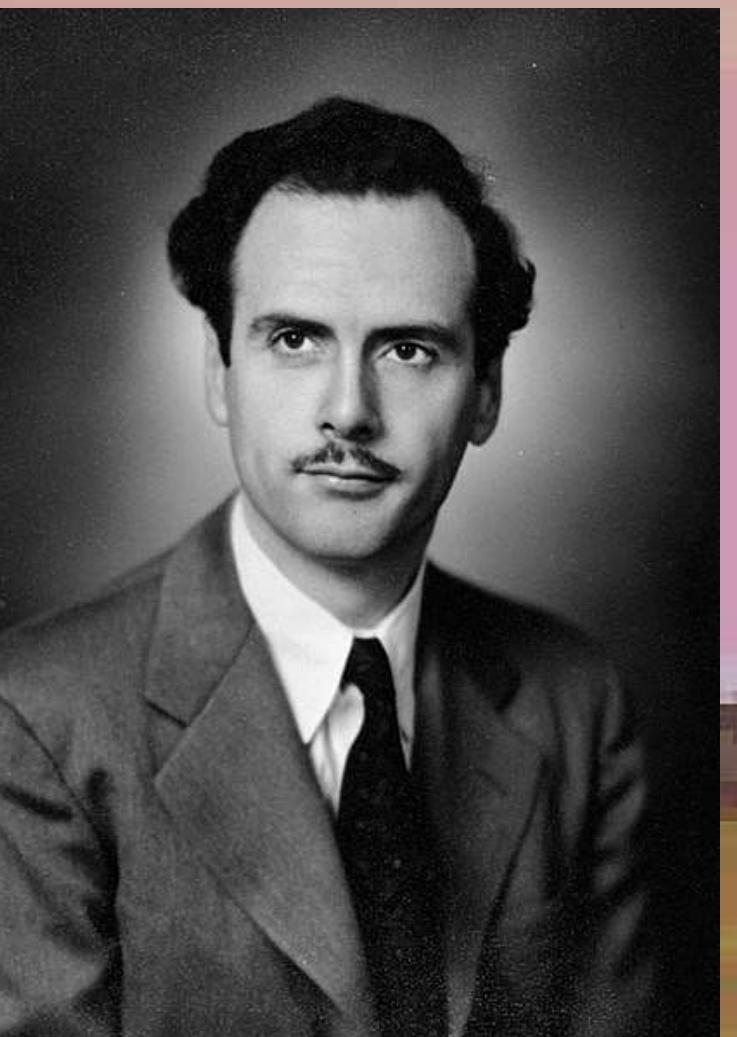
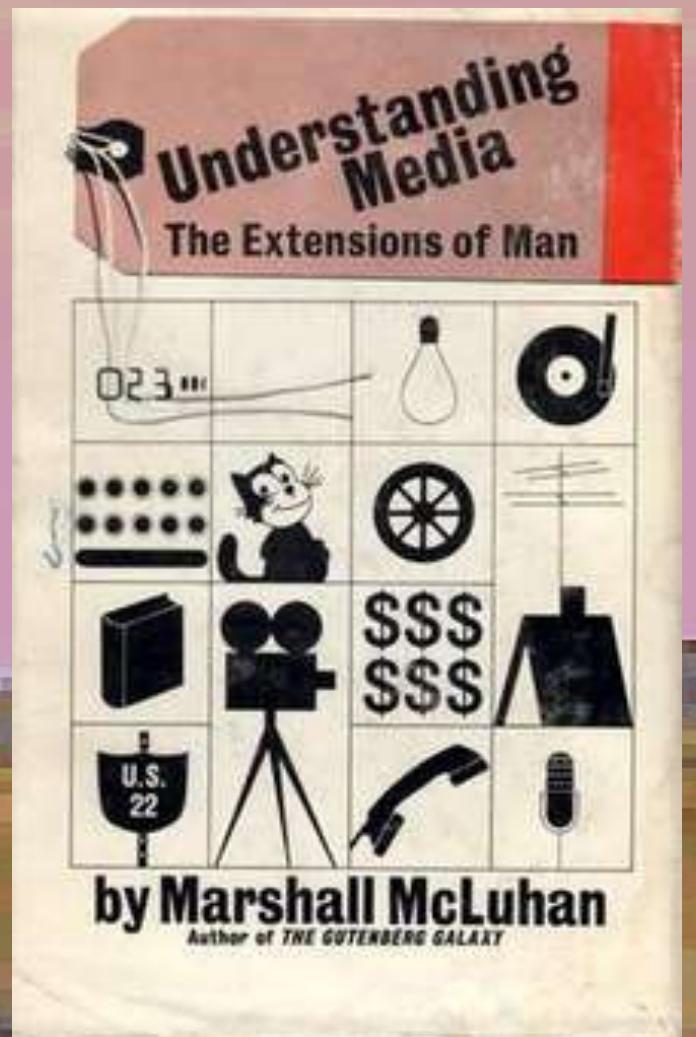
- Throughout the 20th century, film and photography had a very close relationship with the truth: what was depicted in photographic images was more or less accepted as representing a certain reality
- This is born out in the role that photographs and films were allowed to play in society (e.g. as proof in courtrooms or as documentary images in newspapers)

Right: Robert Capa, SPAIN. Córdoba front. Early September, 1936. Death of a loyalist militiaman.



McLuhan, M. 1964. *Understanding Media*. McGraw-Hill.

- McLuhan makes the point that all media are in some way involved in establishing what is real. Reality is a construction made up of different media representations.
- "The medium is the message" - he goes even further and says that the actual content of media is not so important; instead it is the way that media structure reality that we should pay attention to.
  - i.e. what allowed photography to play the role that it did was not what it depicted, but rather the way it was constructed (a person had to stand at a specific point in place and time and click a shutter; film had to be exposed to light; paper to chemicals etc.)



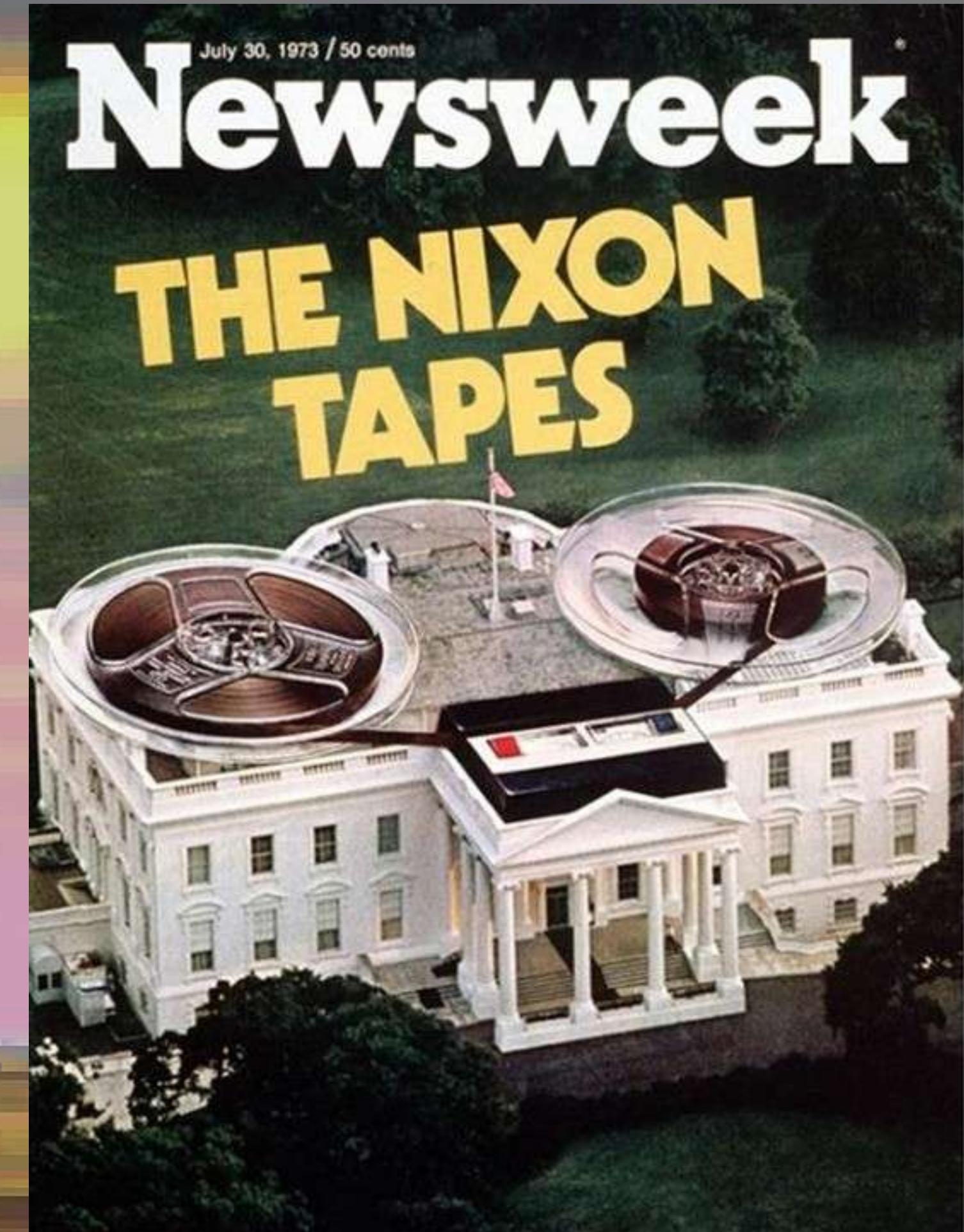
# How is reality established?

Not just images...

Audio recordings from their early days also claimed a close relationship with reality; field recordings (for example) have been important documentary sources for over 100 years...

This truth claim was greatly reinforced when audio recordings were used in the very high-profile Watergate Scandal in the USA, when Richard Nixon was found to have ordered the 1972 break-in of the Democratic National Committee headquarters in Washington, D.C.

This launched the discipline of *audio forensics*.



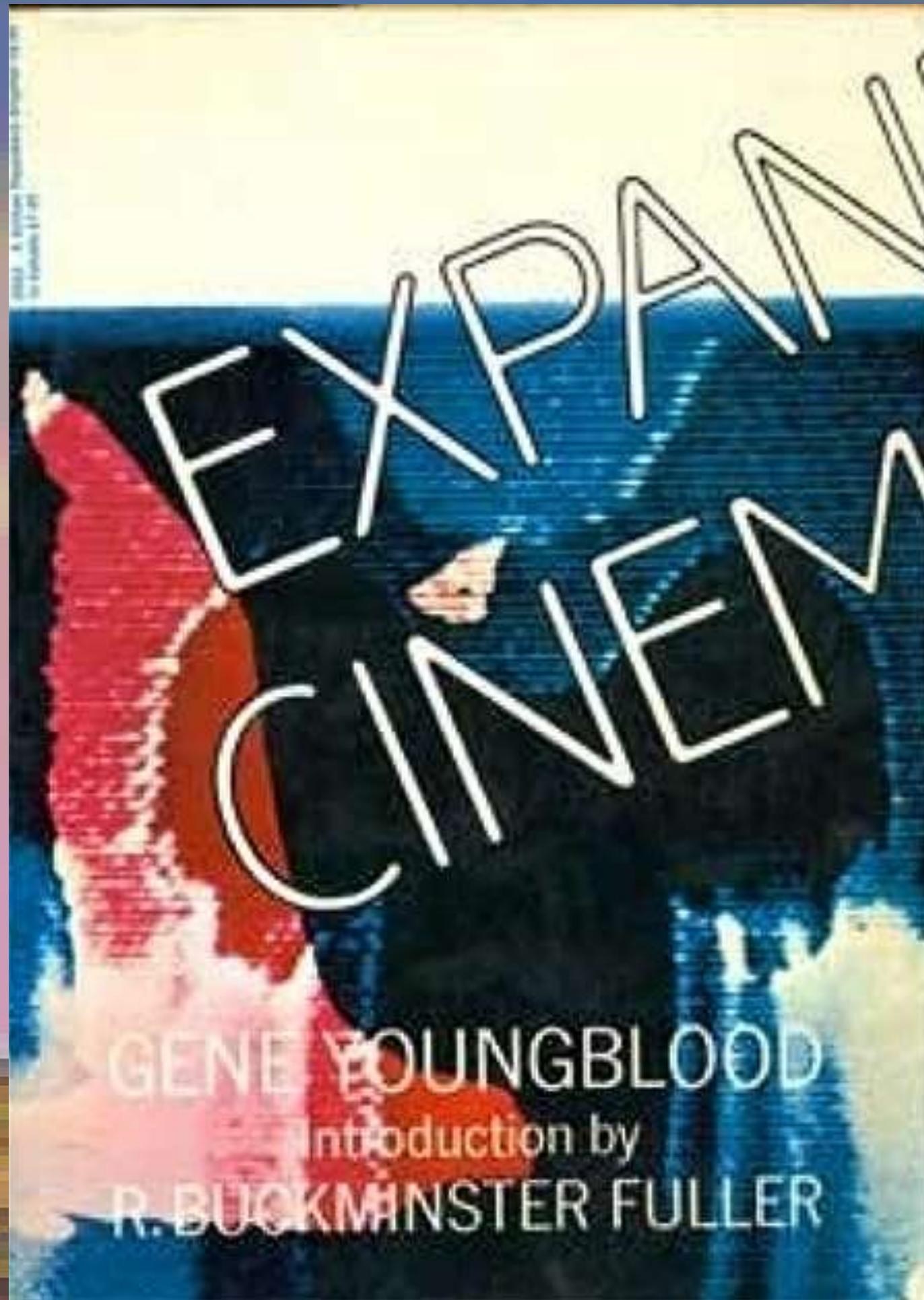


**Live broadcast of moon landing, 20 July 1969**



<https://youtu.be/9vTrmPWULfo?feature=shared>

# Expanded Cinema

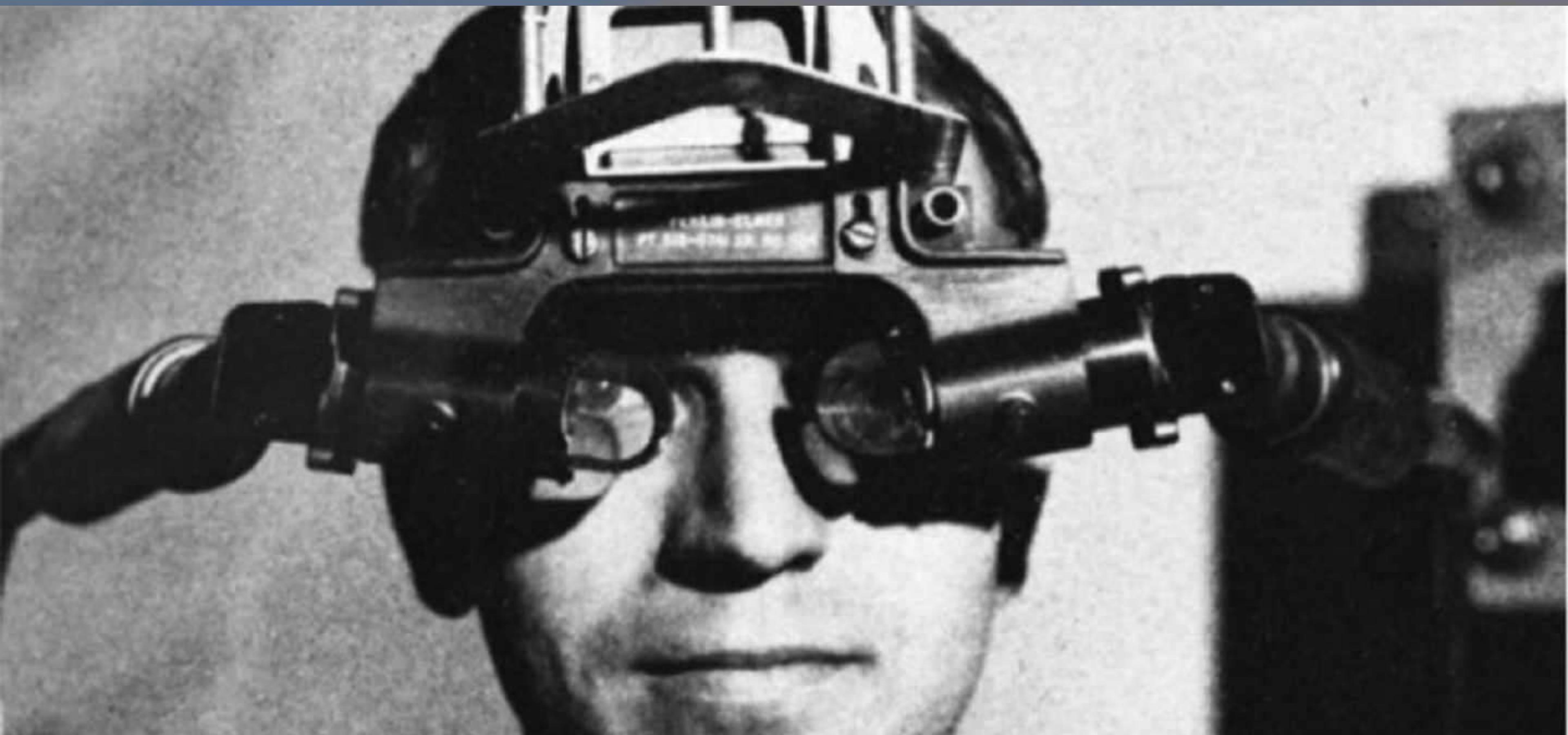


Youngblood, G. 1970. *Expanded Cinema*. E.P. Dutton & Co. New York.

- *Expanded Cinema* was a term invented to describe a number of trends that occurred in digital media in the 1960s and 70s
- "Television's elaborate movie-like subjective-camera simulation of the first moon landing" (p46)
  - Simulation surpasses film as a means of representing reality?
- Part of this new constellation of media technologies were computers, which Youngblood proposes as a new paradigm for cinema; although at the time no computer could generate images in realtime, he predicts a future where on-location shooting will be made redundant because locations can be "simulated" (...Virtual Production?)
- Draws a clear link between the historical trajectory of cinema and the future trajectory of computational technologies (e.g. video games, VR, YouTube, TikTok etc.)



## Sword of Damocles, Ivan Sutherland (1966)



<https://youtu.be/eVUgfUvP4uk?feature=shared>

## CAVE Virtual Environment (1993)



Cruz-Neira, C., Sandin, D. and DeFanti, T. 1993. Surround-screen projection-based virtual reality: the design and implementation of the CAVE. In Proceedings of the 20th annual conference on Computer graphics and interactive techniques (SIGGRAPH '93). Association for Computing Machinery, New York, NY, USA, 135–142. <https://doi.org/10.1145/166117.166134>

Baudrillard, J. 1981. *Simulacra and Simulation*. Editions Galilée.

Baudrillard, J. 1991. *The Gulf War Did Not Take Place*. (published in The Guardian)

- Meanwhile, despite the clear distance between these computational images and the appearance of the physical world, the close link between photographic images and reality was being questioned in philosophy
- Baudrillard describes 4 stages by which the relationship between images and reality is stretched and broken... Simulacra (the 4th and final stage) are images that bear no resemblance to something in the real world, and yet may still be accepted as real.
- *Hyperreality*: Essentially the cumulative effect of simulacra - a fictional "world" built upon but which transcends "reality". Hyperrealities exist all around us - influencers on social media, VR experiences that cause people to act physically in unexpected ways, AR pokemon games that cause people to end up in car accidents...
- At stake is not one's ability to distinguish between a "fake" and "real" image (although this is often impossible) - it is how we fundamentally rely on media technologies to construct reality.

Jean Baudrillard

Simulacres et simulation



## Parallel I (Harun Farocki, 2012)



<https://www.bilibili.com/video/BV1JP411779B/>

# INSANE REALISM Landing At Berlin Airport | 4K



<https://youtu.be/hsKJ3yOZRKI?feature=shared>

# "1899" Virtual Production Set



<https://youtu.be/ZMynJCgJIQk?feature=shared>

## The Beating of Faisal Al-Natsheh (Forensic Architecture, 2014)



<https://forensic-architecture.org/investigation/the-beating-of-faisal-al-natsheh>

# Pre-rendered vs. Realtime

**PRE-RENDERED:** Information is not processed at the point of being observed. Images/Sounds/Text etc. pre-recorded and played back. Media is therefore more or less the same every time it is played.

e.g. CDs, mp3s, films etc.

n.b. many elements of video games are pre-rendered, such as cut-scenes etc. Otherwise realtime applications often use re-assembled elements that have been pre-rendered or pre-recorded ahead of time.

Prerendering is typical of pre-computational media technologies, such as film and photography (although both increasingly use computational processes).



Source: Meta

*Film - traditionally a pre-rendered medium - increasingly uses realtime processes in its pipeline, such as in virtual production stages.*

# Pre-rendered vs. Realtime

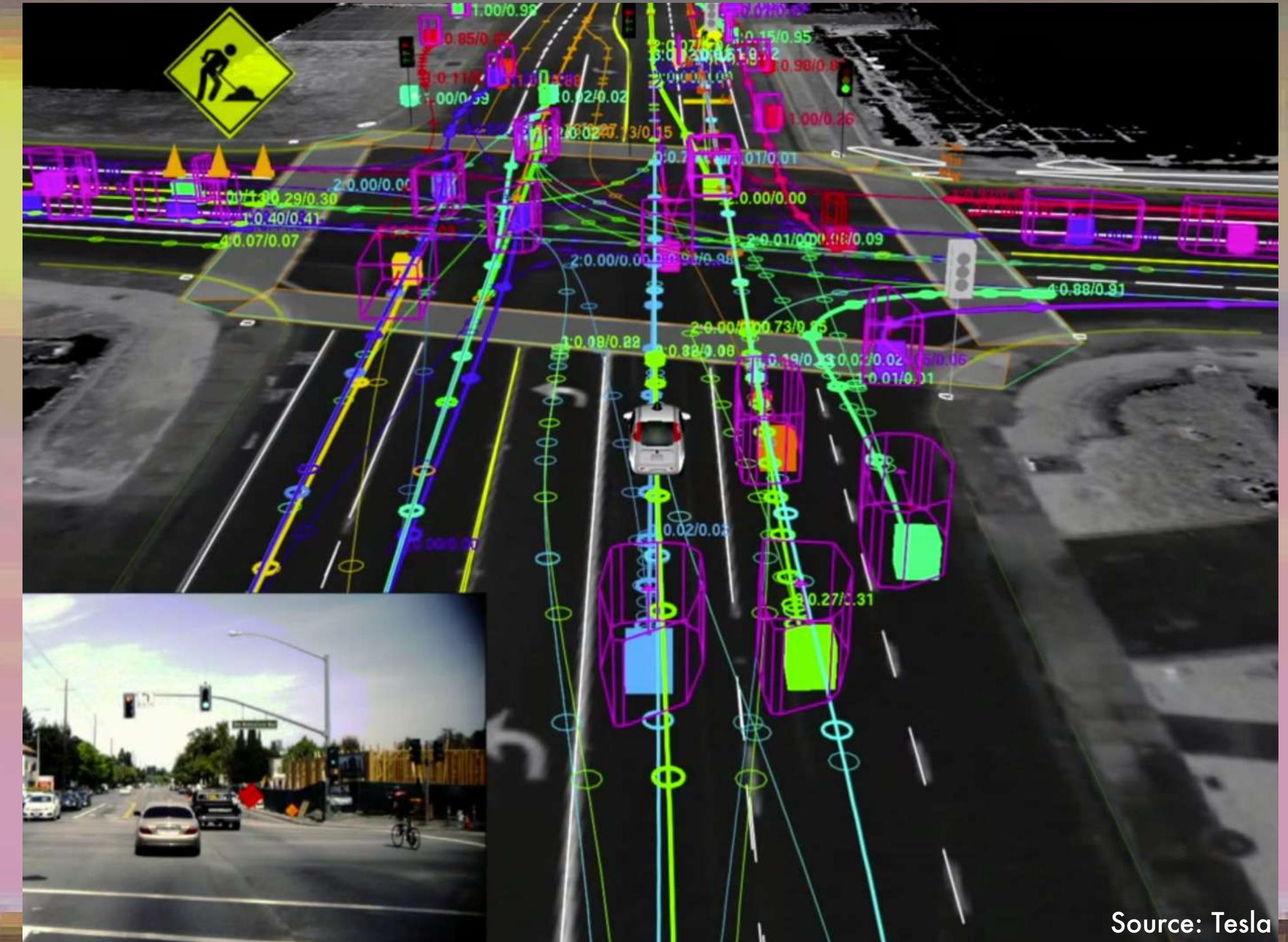
*A term that you already know implicitly...*

**REALTIME:** Information processed at (close to) the speed at which the phenomena occur in the real world.

e.g. GPS is realtime because it constantly updates with the current position of object

e.g. Video games/VR are realtime because (among other things) the image is constantly being generated/updated at the rate at which it is seen by the human eye

Realtime is a characteristic of computational systems - almost all interactive devices/apps/experiences built with a computer will have a realtime element.



Source: Tesla

Self-driving cars rely on realtime images to create a dynamic map of the surrounding environment - necessary for the control of the system and to avoid unexpected risks etc.

## HARD

:: Missing an update ("deadline") is a total system failure

### Examples

Self-driving cars, remote surgery, pacemakers, early video game systems (e.g. Atari 2600)

## FIRM

:: Infrequent deadline misses are tolerable, but impede the system's operation

### Examples

Assembly line software, high-frequency stock trading

## SOFT

:: Deadlines are not critical, and impede operation only if particularly extreme

### Examples

Video game frame rates, live audiovisual performances, software that updates airplanes flight plans etc.

*Spectrum of Realtime*

# Remote Surgery



<https://youtu.be/yFR61jjL1vo?feature=shared>

## Why is this important?

The story of the emergence of XR technologies is not just about technological advancement, but also about the medium's relationship with the notion of reality itself (hence "Extended Realities").

The inherent relationship to truth and "reality" that photography and film have has been assumed by digital images and worlds. This provides the basis on which "reality" can be "extended".

Not only, but the advent of realtime technologies has introduced a new gold standard for how media depicts reality - not just as images of something that happened in the past, but as data streams of something that is happening right now.

## Why "Extended Realities"?

What happens to our sense of reality when we use these technologies?

What does "extended" mean in this case? Are these technologies replacing reality?  
Are they supplementing reality? Is it just a convenient marketing term?

At what point should we accept virtual worlds as real? What are the consequences  
if we do?

# **Available technologies**

- 1. Game engines**
- 2. Performance/live event technology**
- 3. Web development environments**
- 4. VR/MR headsets**
- 5. Tablets/Smartphones**
- 6. Motion Capture Suits**
- 7. Other...**

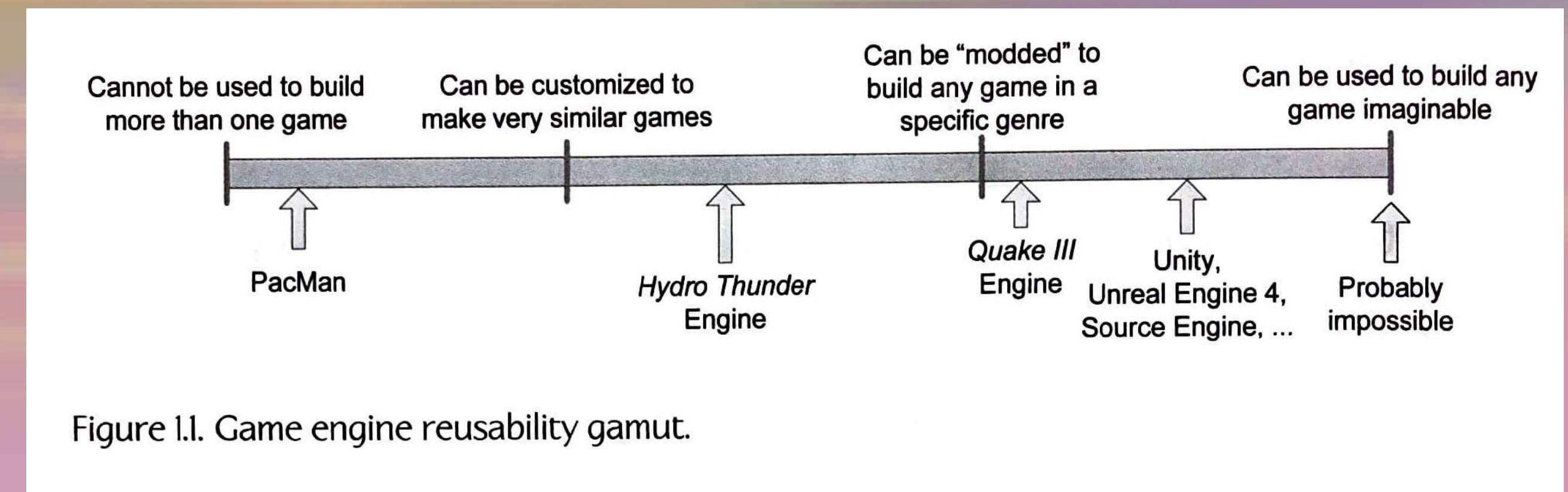
# Game Engines

## Overview

Origin in late 90s with "modding" of popular games.

Structurally, games started to be built with a clear separation between their core software components and the content/assets/game logic.

Game Engines lie on a spectrum: those that are more adaptable and can be used to make many different kinds of experiences (e.g. Unity, Unreal) and those that can be used to make basically 1 game (and therefore are indistinguishable from the game itself).



Source: Gregory, J. 2019. *Game Engine Architecture*. CRC Press.

# Game Engines

## Unity



Unity launched in 2005 specifically with the goal to "democratise" game development... which it did.

### Features

- C#
- Modular script components
- Less powerful graphics engine
- Slightly bigger community (?)
- Best for: Lightweight, mobile, 2D

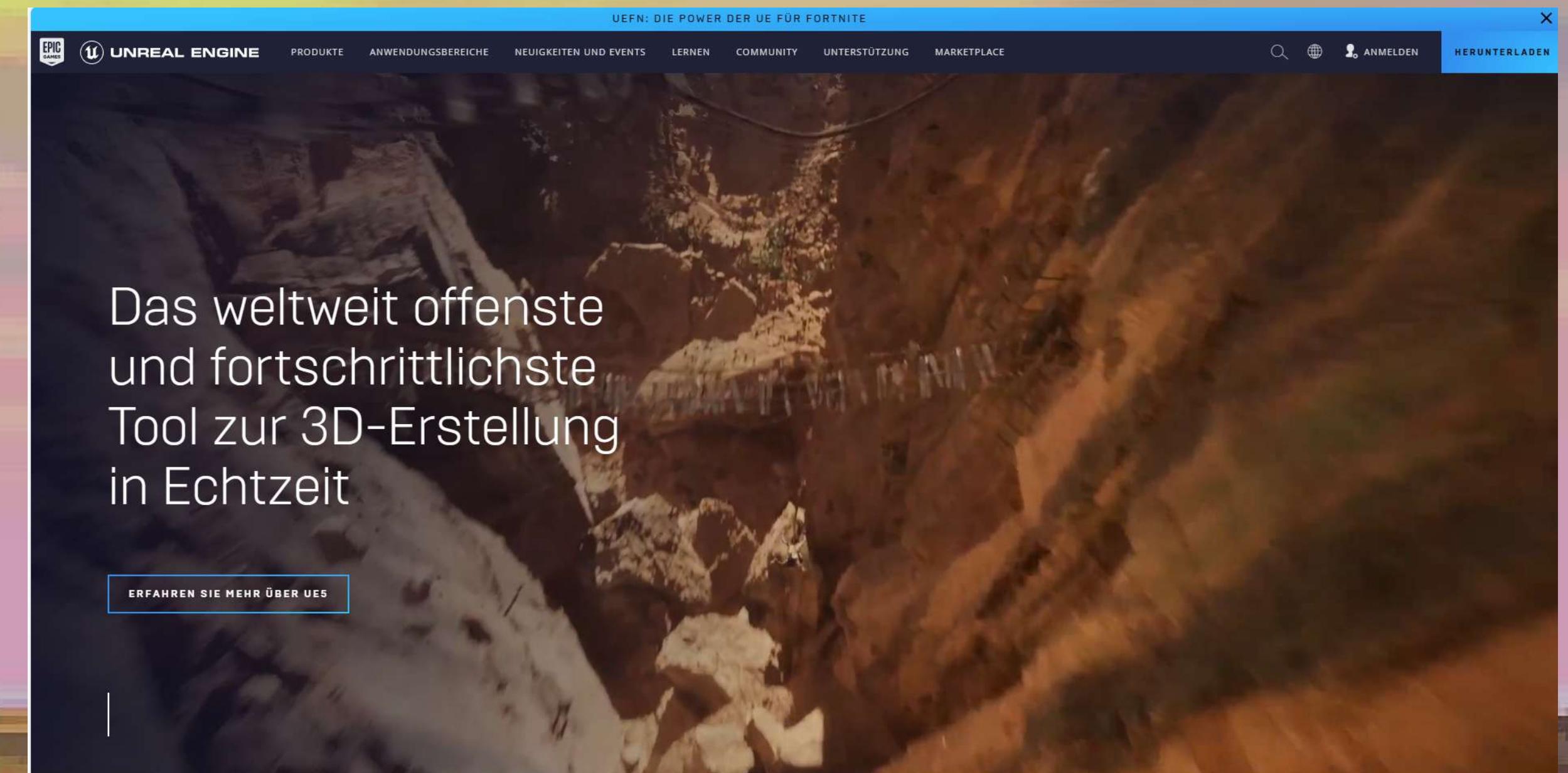
# Game Engines

## Unreal

Unreal Engine emerged out of the Unreal game franchise, very popular in early 2000s. It is a major engine for AAA FPS-style games, increasingly used in other industries (e.g. film).

### Features

- C++ (but native node-based visual coding)
- Powerful graphics engine
- Slightly smaller community
- Loads of free, high-quality assets
- Best for: High production value, photo-realistic graphics, people who think with nodes...



<https://www.unrealengine.com/de>

# Game Engines

## Other (e.g. Godot, 3JS etc.)

Unity and Unreal have been at the top a long time, but there are many other engines that may be more or less useful for your purposes...

1. Godot

2. NVIDIA Omniverse

3. 3JS/Babylon/Mozilla Hubs

4. Open 3D Engine

5. OGRE

+ Games that support modding (e.g. Minecraft, MSFS, Skyrim, Gary's Mod, Half-Life etc.)



**GODOT**  
Game engine



**hubs.** by  moz://a



**NVIDIA  
OMNIVERSE™  
ENTERPRISE**

**O3DE**  
OPEN 3D ENGINE

 **OGRE**



Source: Resident Evil

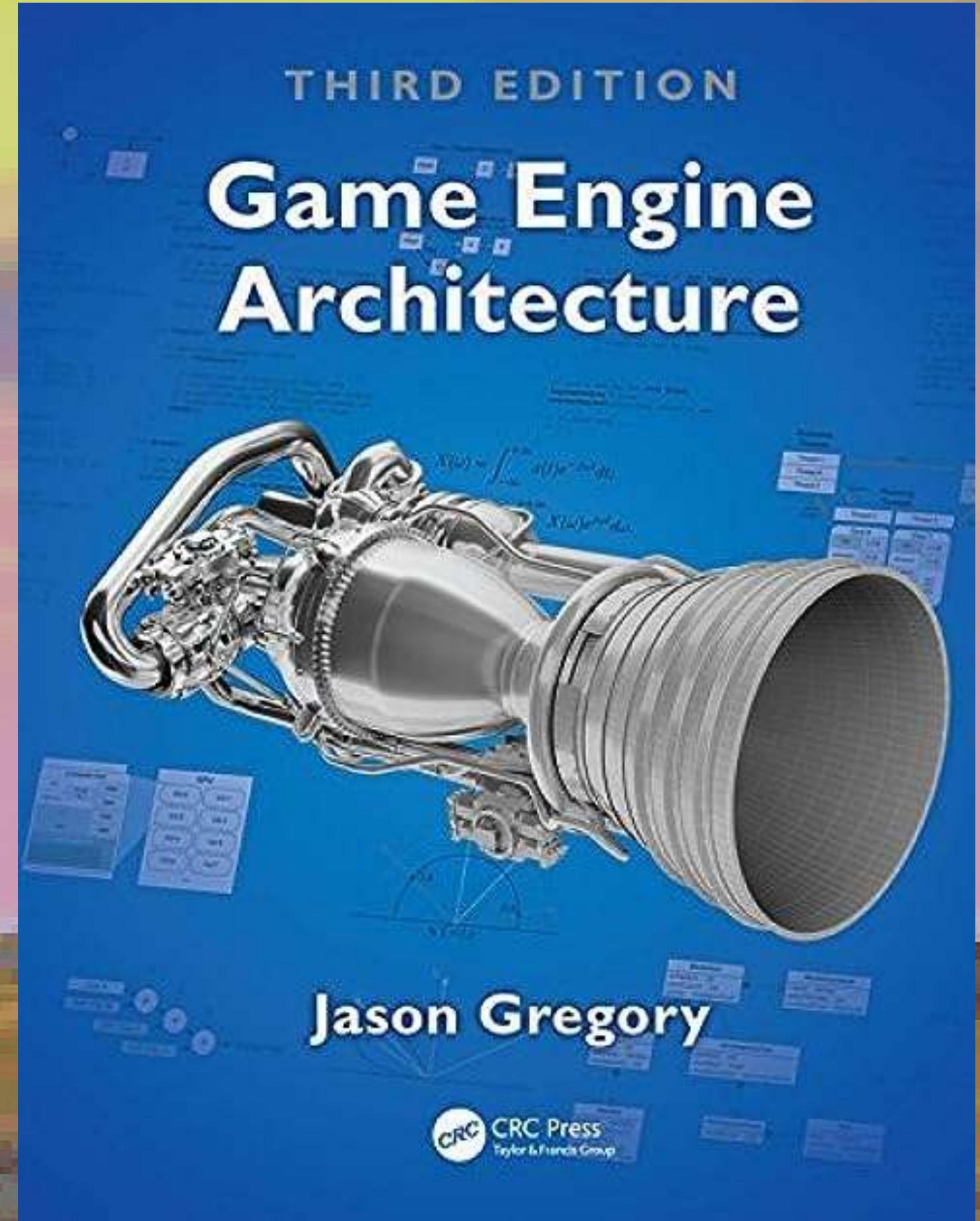
# Game Engine Architecture

## Overview

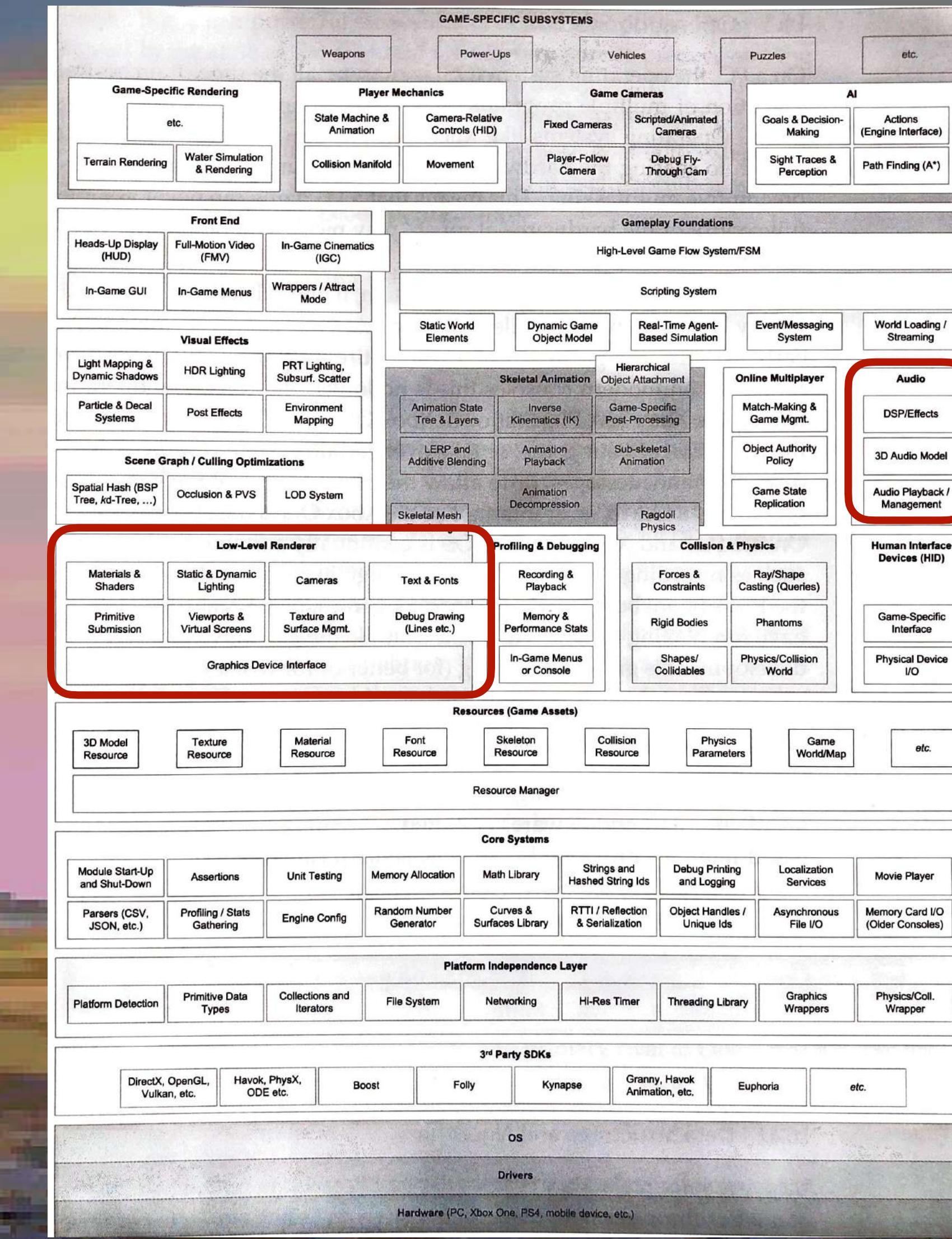
Gregory, J. 2019. *Game Engine Architecture*. CRC Press.

- Game Engines can be broadly broken down into two categories: the tool suite (i.e. what you will use to make your experience) + the runtime architecture (i.e. the behind-the-scenes components that compile your code, run the graphics, implement sound, interactions etc.)
- Structured in layers: upper layers are generally "downstream" and depend on lower layers to function.
- Here we are going to look only at the rendering engine and the audio engine, before briefly considering the tools/asset pipeline.

*nb. information on following slides comes mainly from book on right...*



# Game Engine Architecture



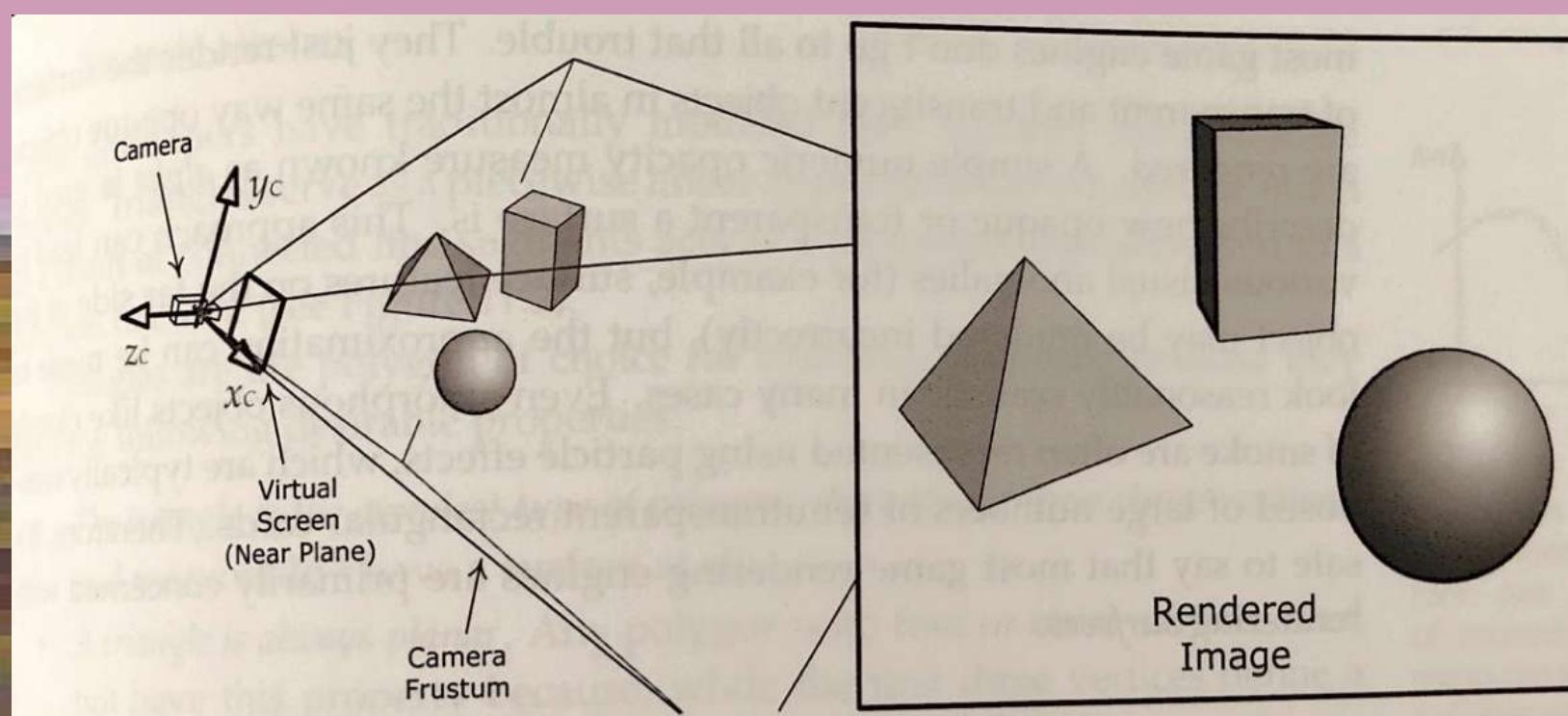
Source: Gregory, J. 2019. Game Engine Architecture. CRC Press.

# Game Engine Architecture

## Rendering Engine

### COMPONENTS OF A SCENE

- *Virtual scene*: a collection of 3D surfaces represented in mathematical form.
- *Virtual camera*: Positioned and oriented to produce desired view of the scene.
- *Light sources*: Provide all the light rays that reflect off objects in environment and into camera.
- *Visual properties of surfaces*: Describe how light should interact with each surface.



### RENDERING PIPELINE

1. *Tools stage (Offline)*: Geometry and materials are defined.
2. *Asset conditioning stage (Offline)*: Geometry and materials processed by asset conditioning pipeline (ACP) into engine-ready format.
3. *Application stage (CPU)*: Potentially-visible mesh instances identified and submitted to graphics hardware along with materials.
4. *Geometry processing (GPU)*: Vertices transformed and lit and projected into homogenous clip space. Triangles processed by geometry shader and then clipped to frustum.
5. *Rasterization (GPU)*: Triangles converted into fragments that are shaded, passed through various tests (z-test, alpha test, stencil test etc.) and finally blended into the frame buffer.

# Game Engine Architecture

## Rendering Engine



Source: Assassin's Creed Odyssey

### COMPLICATING THE PICTURE

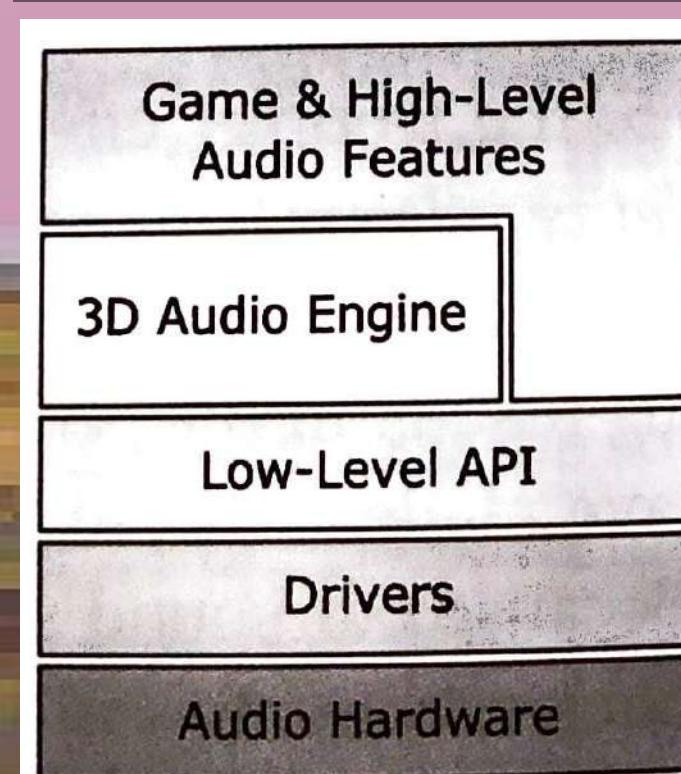
- *Shaders* (covered in another course...): Shaders tell the rendering engine how light interacts with the mesh surface or volume. In game engines, shaders are usually combined with *textures* to make *materials*, which are then applied to the objects.
- *Lighting*: Interacts with the shader in order to determine how the object will appear in the camera. There are a number of basic kinds of light sources modelled in game engines, including static lighting (calculated before runtime), ambient lights, directional lights (e.g. the sun), point lights, spot lights etc.
- *Visual effects and overlays*: Often a number of post-processing effects can be added to the virtual camera. Furthermore, VFX such as particle effects and decals are often added.
- *Virtual camera*: In addition to post-processing there are also a number of options that relate to the virtual camera itself, such as depth of field and culling.

# Game Engine Architecture

## Audio Engine

### COMPONENTS OF A SCENE

- *3D sound surfaces*: Each 3D sound in the game world consists of a monophonic audio signal, emanating from a specific position.
- *Listener*: The listener is a virtual microphone located somewhere in the game world
- *Environmental model*: Model describes the geometry and acoustic properties of the surfaces and objects present in the virtual world.



Source: Gregory, J. 2019. Game Engine Architecture. CRC Press.

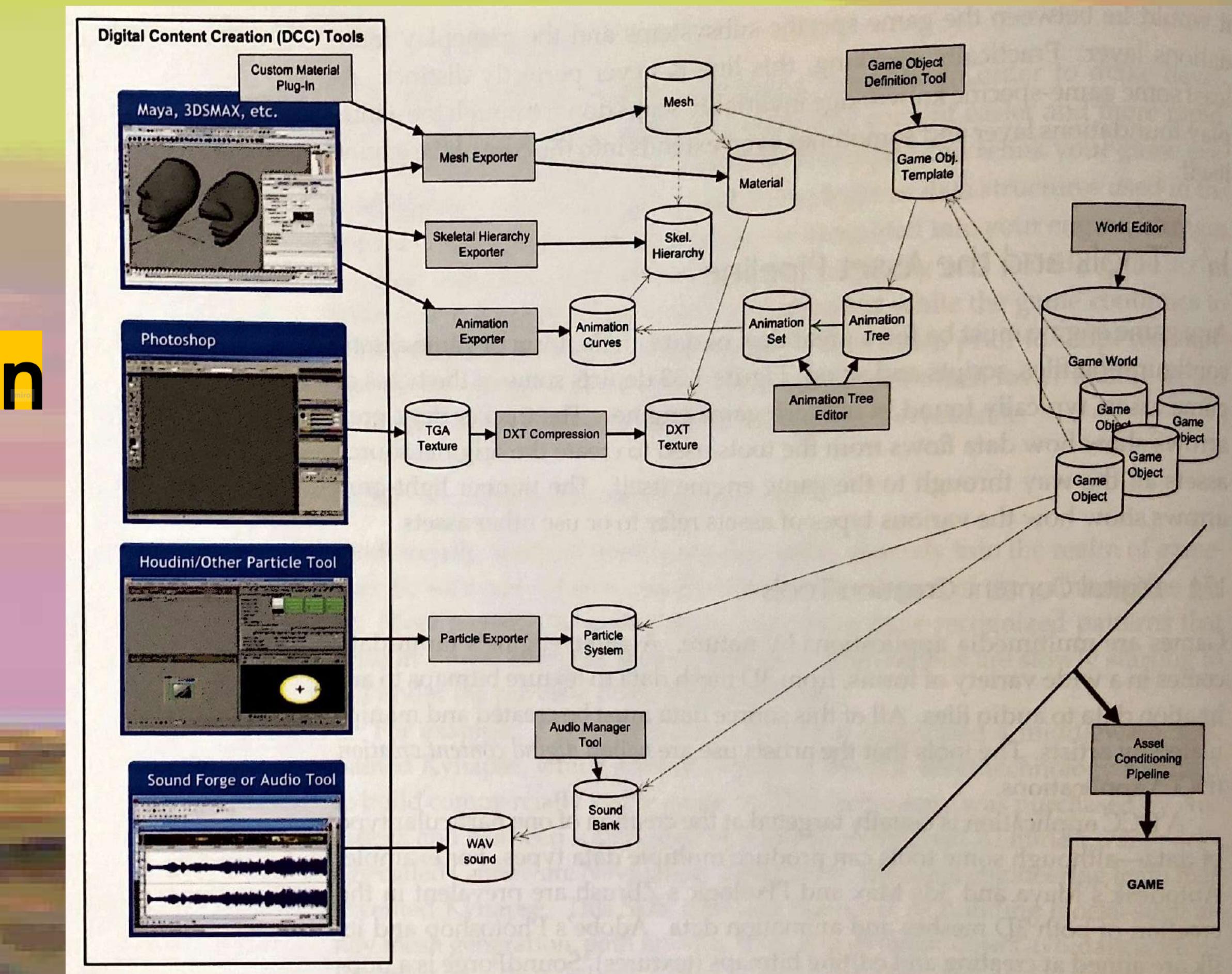
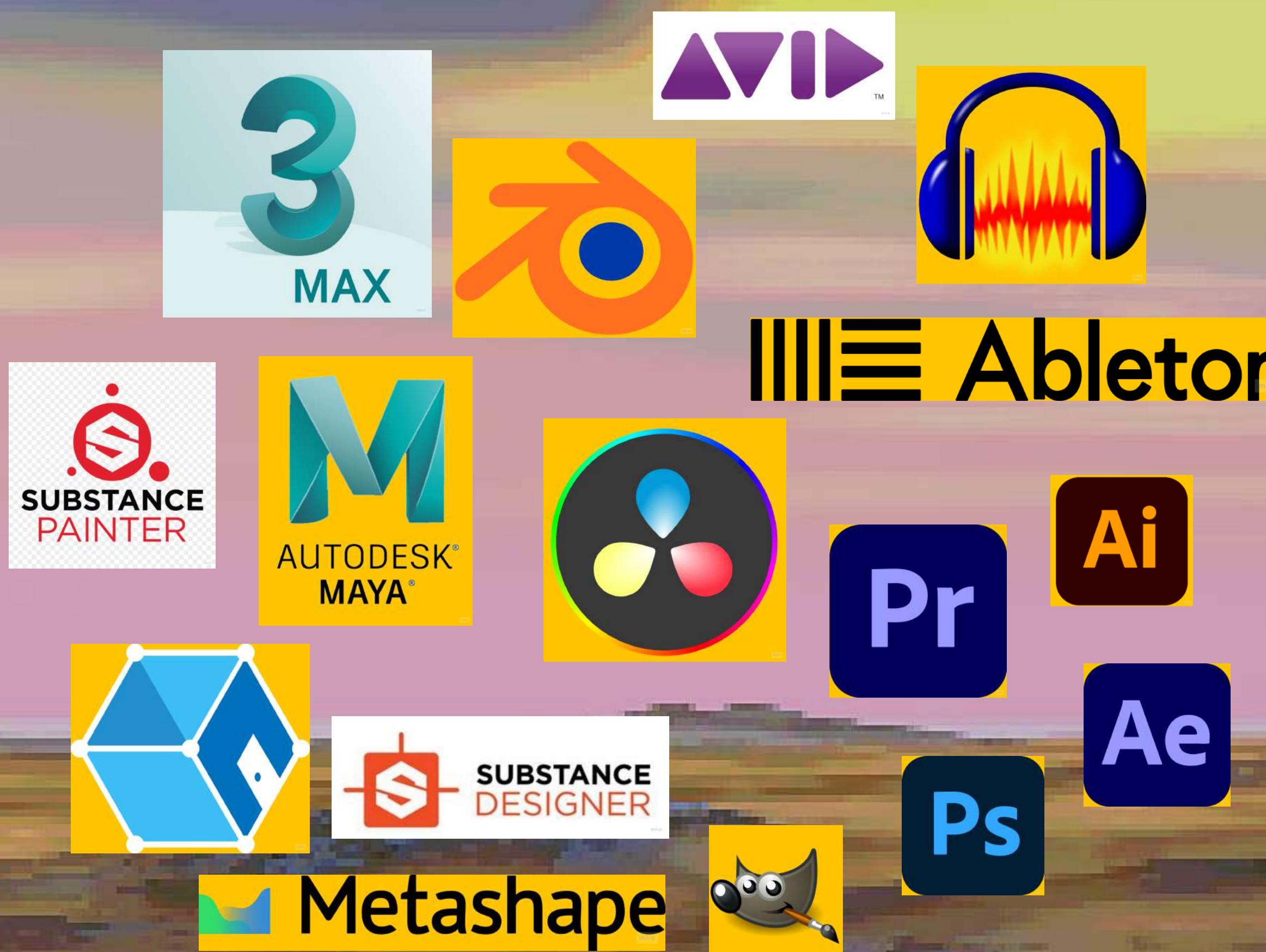
### AUDIO PROCESSING PIPELINE

1. For each 3D sound, a "dry" digital (PCM) signal must be synthesized.
2. Distance-based attenuation is applied to provide a sense of distance from the listener, and reverb is applied to the signal to model the acoustics of the virtual listening space. This produces the "wet" signal.
3. Wet and dry signals are panned to one or more speakers in order to produce the final "image" of each signal in 3D space.

nb. The audio engine in off-the-shelf game engines like Unity and Unreal are often not very advanced and work by a very simply *distance-attenuation model* (i.e. the sound gets quieter the further the listener is from the source, independent from the fact of whether or not there is a wall in the way...)

# Game Engine Architecture

## Tools and the asset pipeline



# Hardware



Source: Oculus



Source: Kristina Mirkashimova

# XR

## VR, AR, MR, XR

R  
E  
A  
L  
I  
T  
Y

### AR

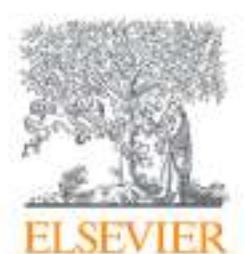
Encompasses a wide range of different applications from digital image overlays, transparent interfaces, location-based experiences, spatial audio tours etc.

Essentially any kind of app that seeks to partially immerse you in an experience through the overlay of data on the real environment.

In practice, usually involves a smartphone with a camera (and increasingly Lidar scanners...) or headset such as Hololens.

### VR

Generally a much narrower category - in order to qualify as VR, the experience must be "fully" immersive. In practice, that usually means blocking out the real environment with CGI imagery (either through VR headset or CAVE environment)



## Facebook's Project Aria indicates problems for responsible innovation when broadly deploying AR and other pervasive technology in the Commons

Sally A. Applin <sup>a</sup> , Catherine Flick <sup>b</sup>

Show more

+ Add to Mendeley Share Cite

<https://doi.org/10.1016/j.jrt.2021.100010>

Get rights and content

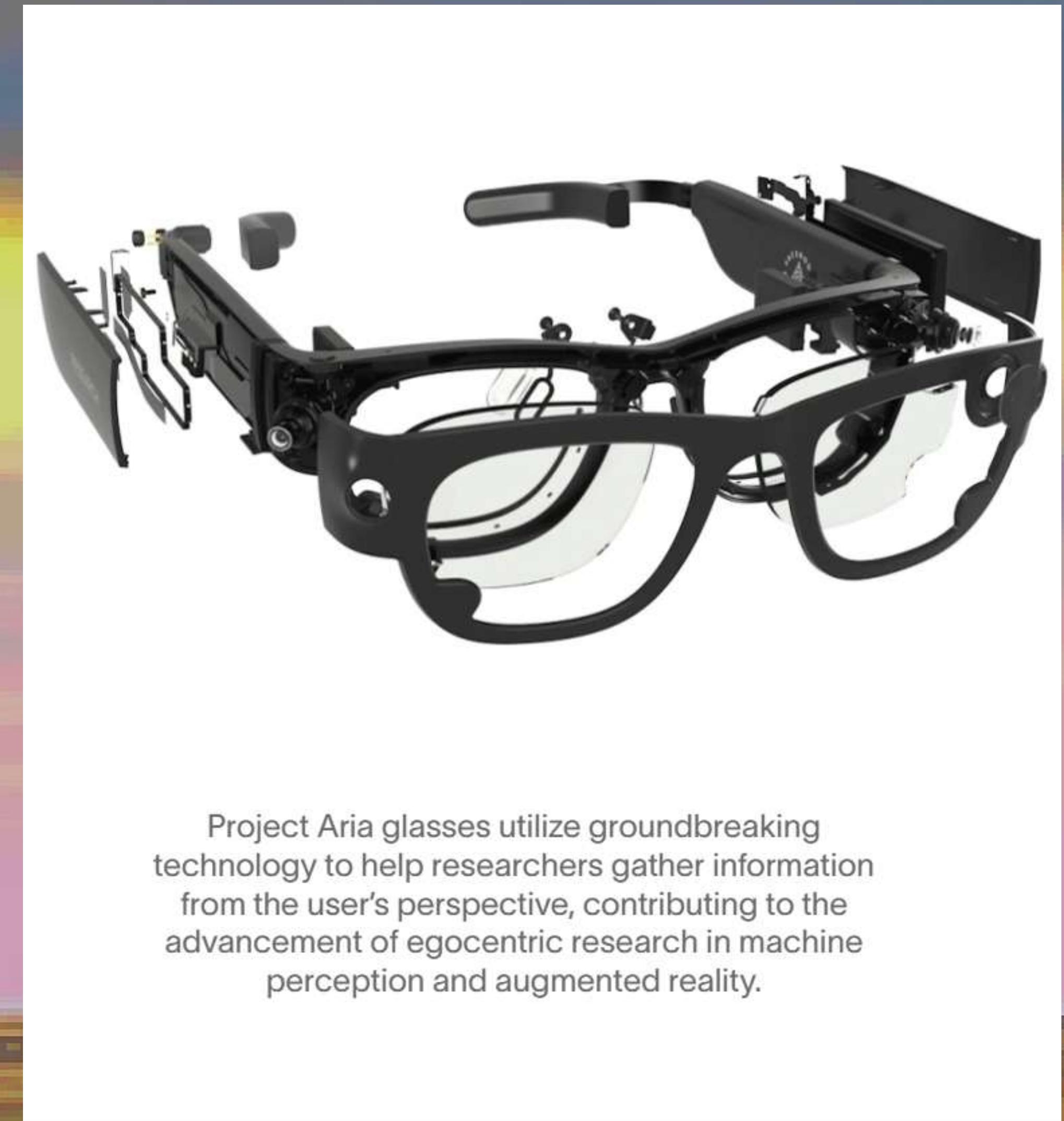
Under a Creative Commons license

• open access

### Abstract

Nearly every week, a technology company is introducing a new surveillance technology, varying from applying facial recognition to observing and cataloguing behaviours of the public in the Commons and private spaces, to listening and recording what we say, or mapping what we do, where we go, and who we're with—or as much of these facets of our lives as can be accessed. As such, the general public writ-large has had to wrestle with the colonization of publicly funded space, and the outcomes to each of our personal lives as a result of the massive harvesting and storing of our data, and the potential machine learning and processing applied to that data. Facebook, once content to harvest our data through its website, cookies, and apps on mobile phones and computers, has

<https://www.sciencedirect.com/science/article/pii/S2666659621000032>



Project Aria glasses utilize groundbreaking technology to help researchers gather information from the user's perspective, contributing to the advancement of egocentric research in machine perception and augmented reality.

<https://www.projectaria.com/>

# Extended Realities

## Introduction to Unity

MA Creative Technologies  
27.10.2023

# Introduction to Unity

## PLAN

- Setting up a new project in Unity and tour of the interface
- Overview of asset pipeline (including finding and downloading free pre-made assets from asset store)
- Overview of optimisation tools
- Series of tasks to introduce you to the main functions and features available
- Feedback/discussion of difficulties and possibilities of learning game engines



Source: Unity

# Introduction to Unity

## Resources

### DOCUMENTATION

- <https://docs.unity.com/> - includes the user manual, the scripting API, as well as further introductions to asset store and templates for demo projects

### TROUBLESHOOTING

- <https://discussions.unity.com/> - the official Unity forum, the Unity staff often respond to questions themselves
- <https://stackoverflow.com/questions/tagged/unity-game-engine?tab>Newest> - Stack Overflow has v. large and (usually) helpful community, just learn to ask questions in the correct way!
- <https://www.reddit.com/r/UnityHelp/> - Often Reddit can be of help as well due to the large community

### ONLINE COURSES

- <https://learn.unity.com/> - the official (free) Unity learning platform. Somewhat confusing layout, but generally very sound and well-structured tutorials for all levels
- <https://www.gamedev.tv/> - I am not in any way affiliated, have just taken a few of their courses in the past and found them quite helpful
- <https://www.freecodecamp.org/news/game-development-for-beginners-unity-course/> - FreeCodeCamp is a really excellent resource for all kinds of programming
- <https://www.pluralsight.com/> - Have done a few courses in the past here, also well-structured and fairly comprehensive

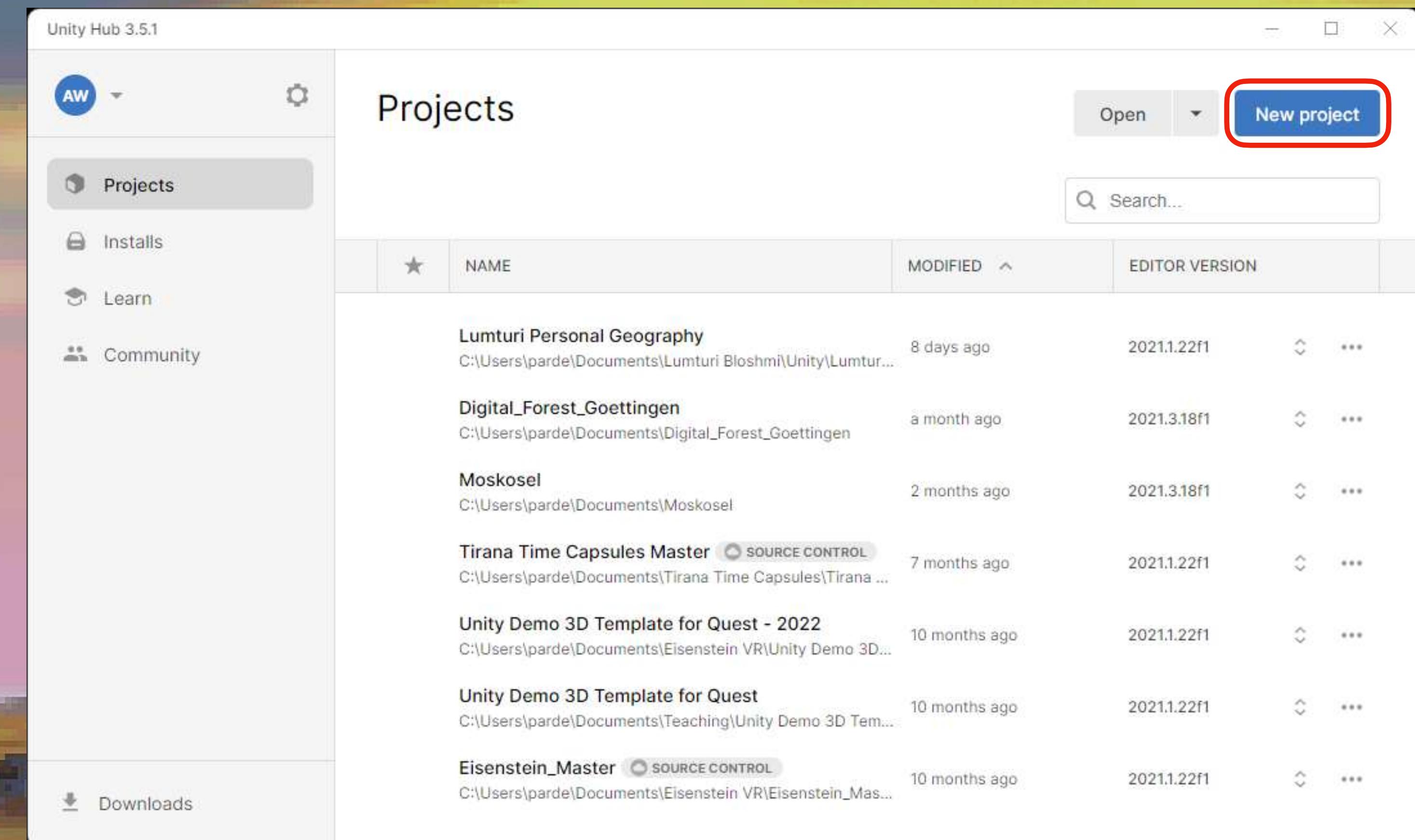
# Introduction to Unity

You have installed:

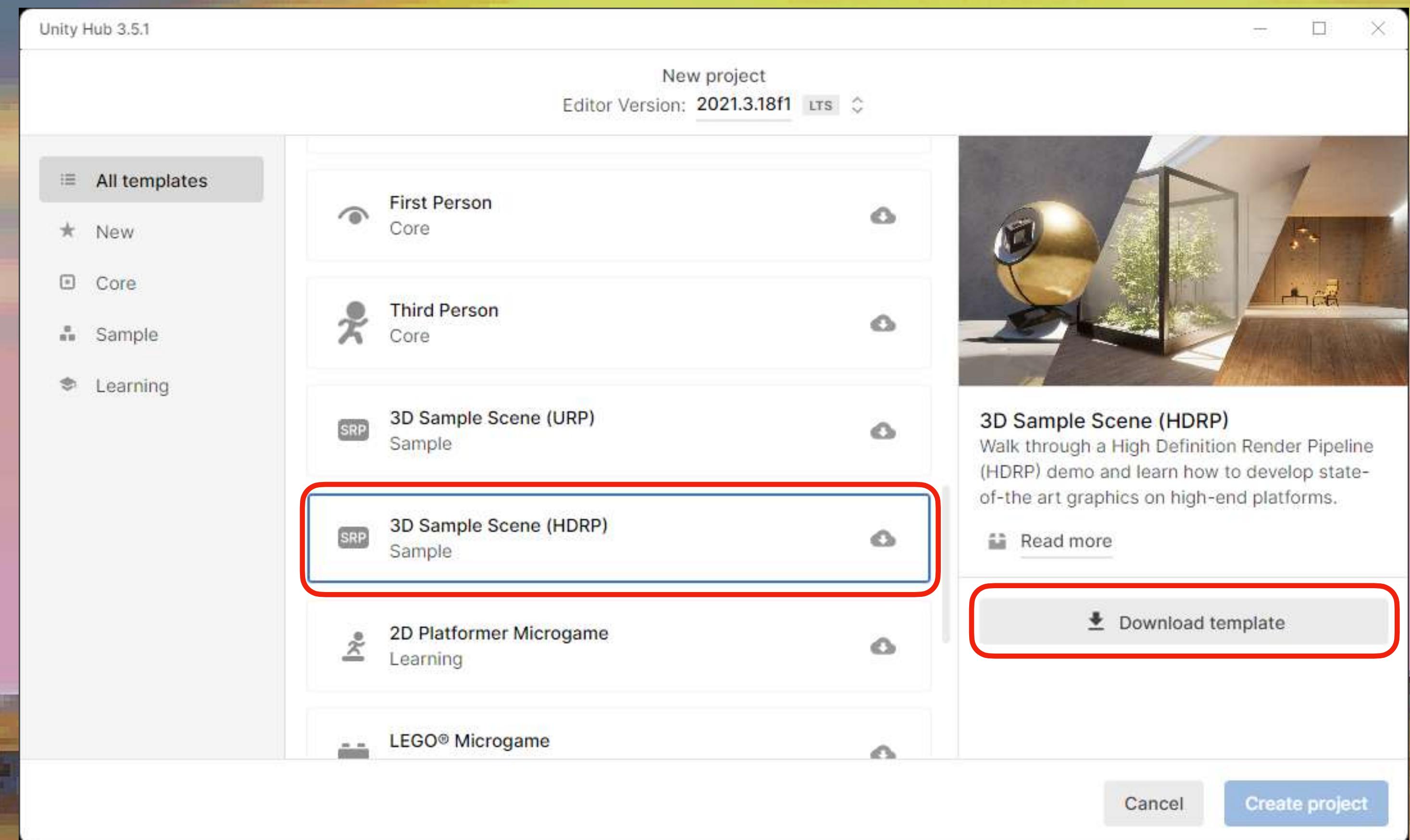
- The LTS version of Unity (at time of writing 2022.3.11f1)
- Modules: Android Build Support, iOS Build Support, Windows Build Support

*nb. you should also now have a Unity developers account.*

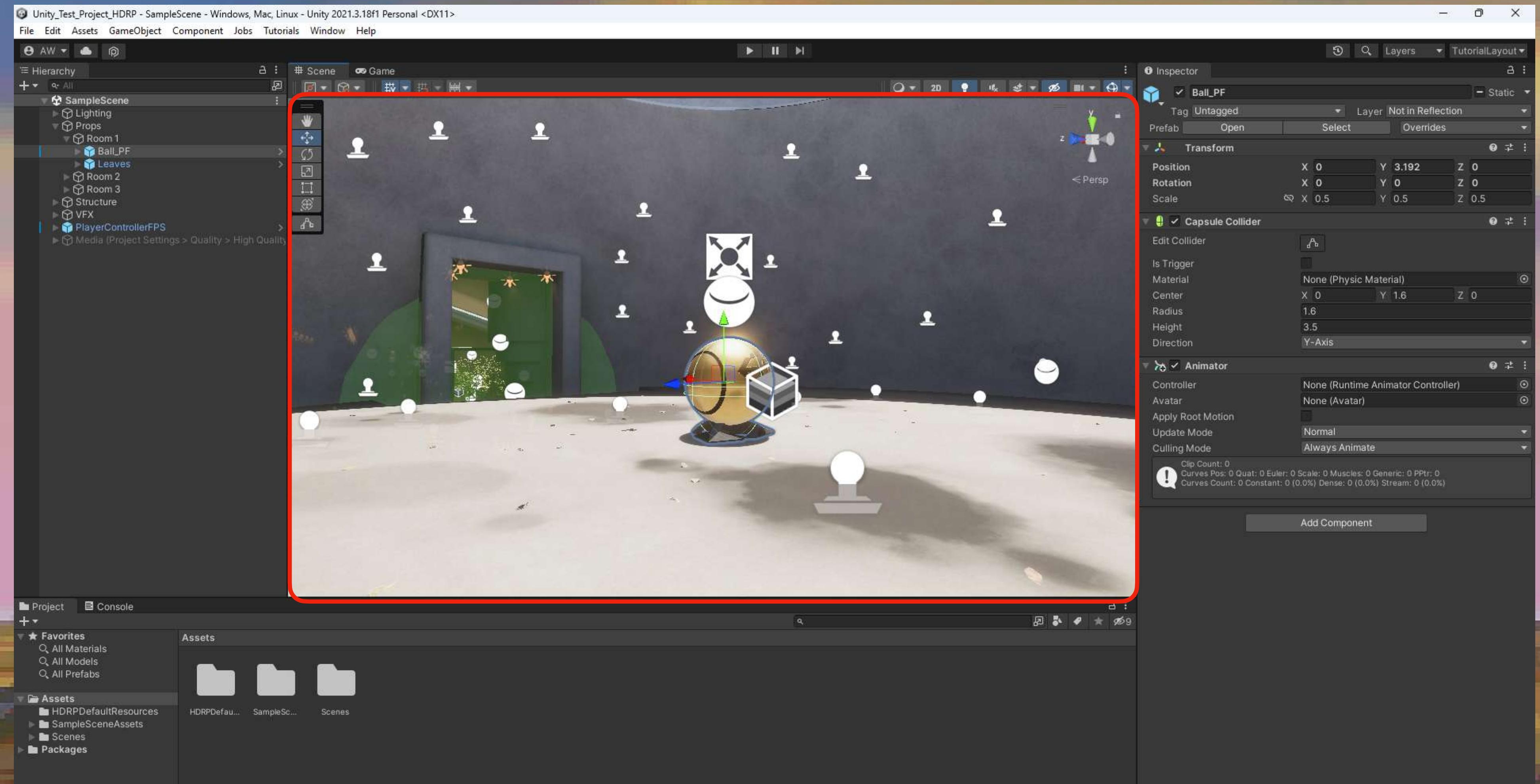
# Introduction to Unity



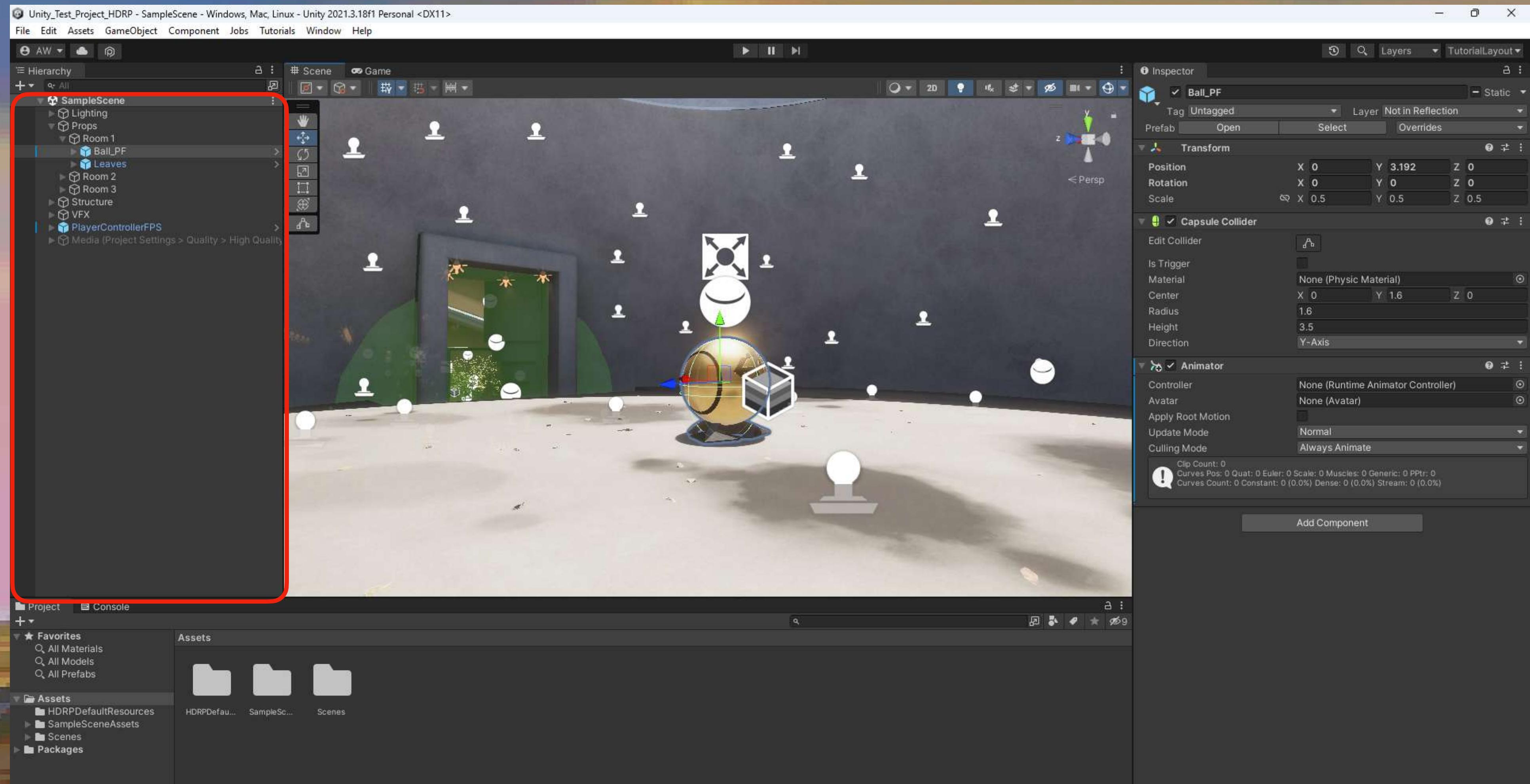
# Introduction to Unity



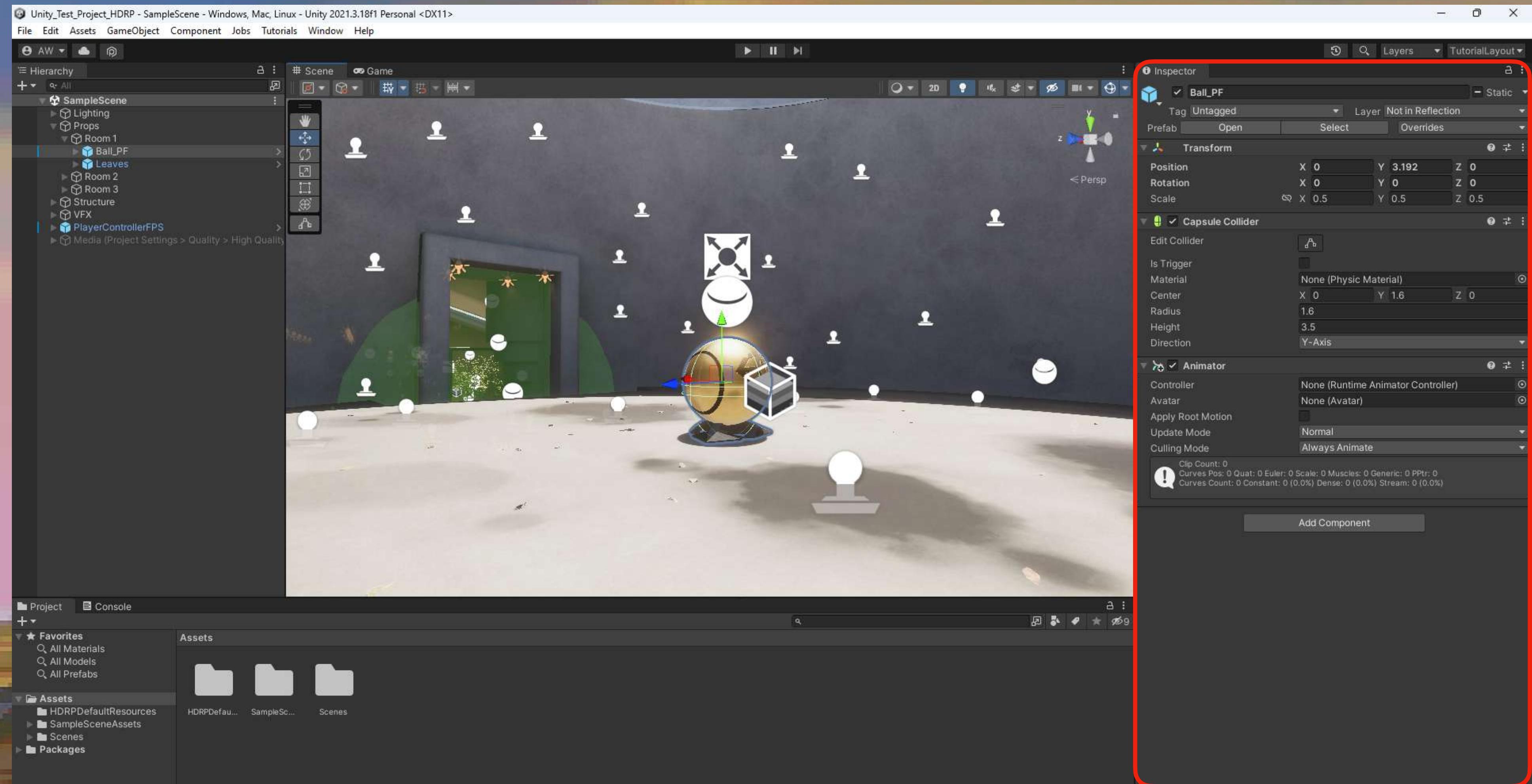
# Introduction to Unity



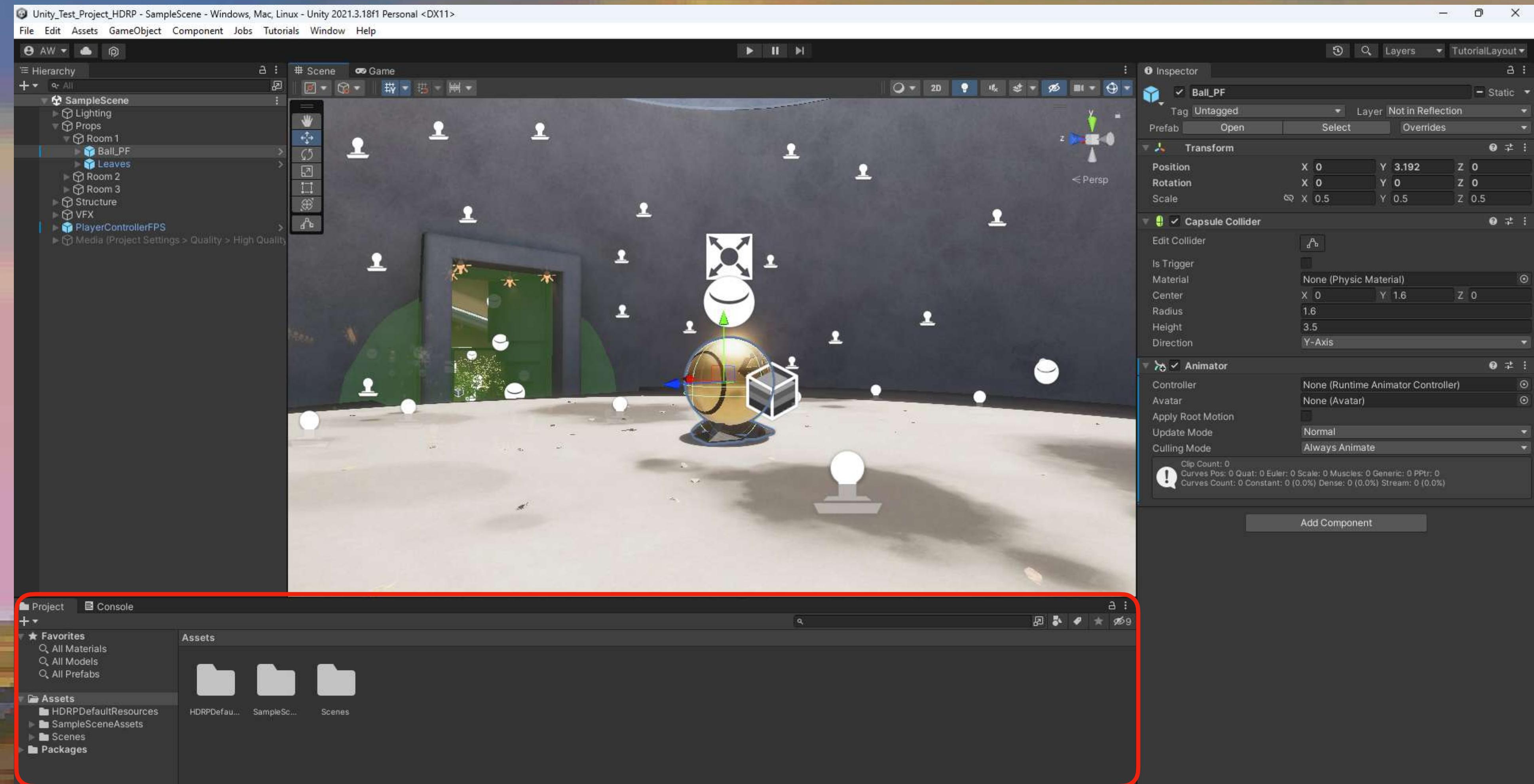
# Introduction to Unity



# Introduction to Unity



# Introduction to Unity

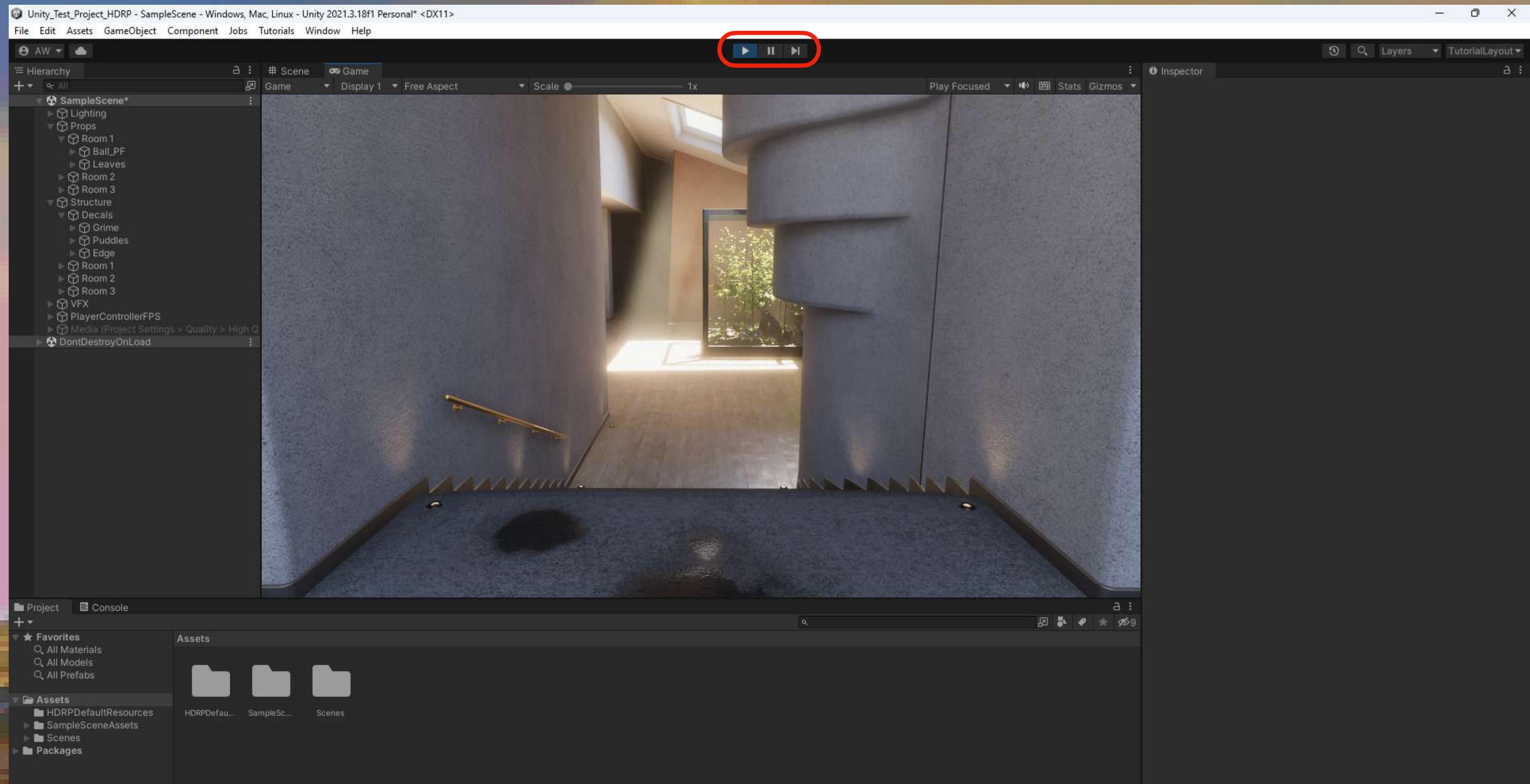


# Introduction to Unity

Take 5 minutes and navigate around your scene using the mouse.

Click on different objects in the scene and look at their properties. Try changing some of these and seeing what happens (if anything).

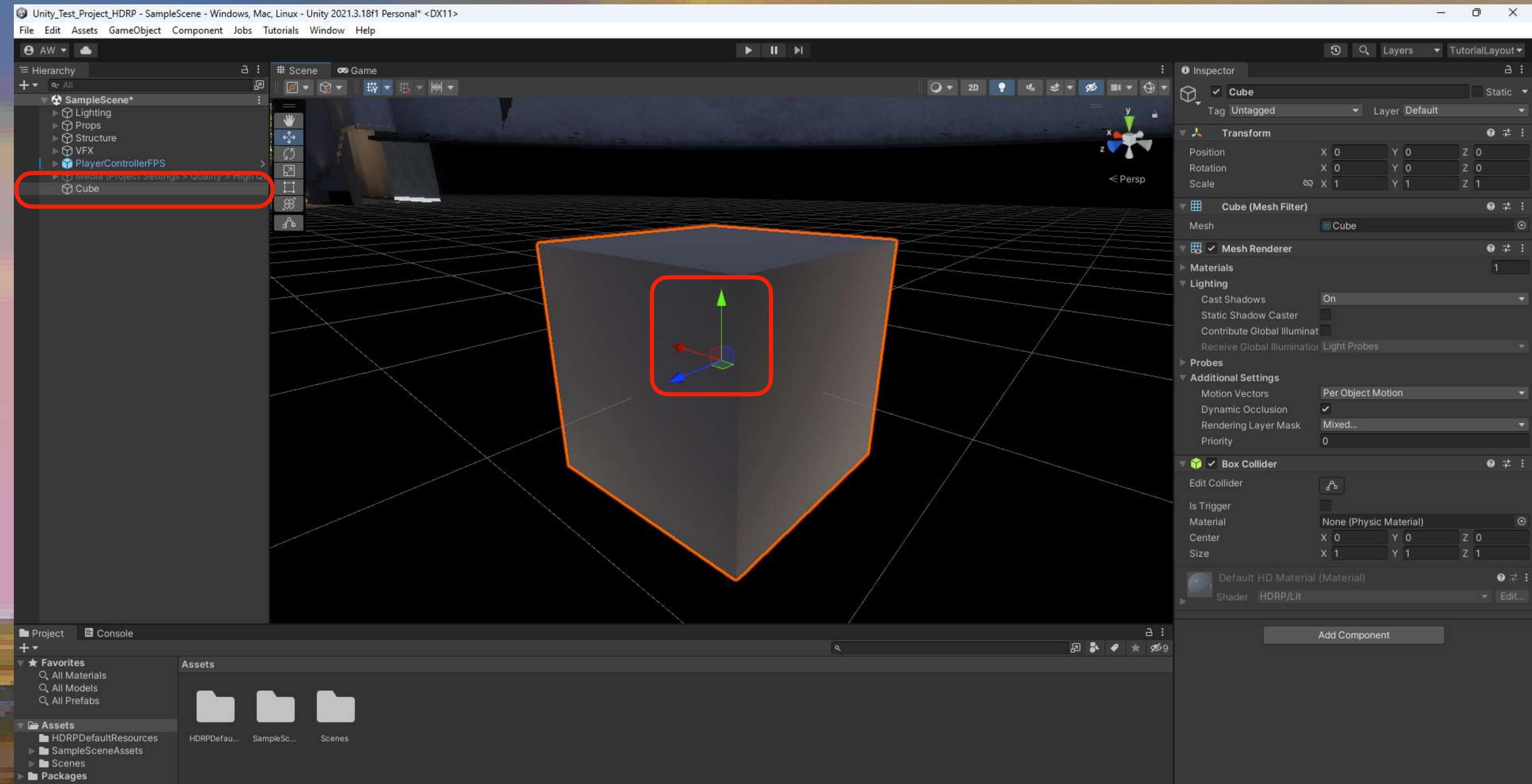
# Introduction to Unity



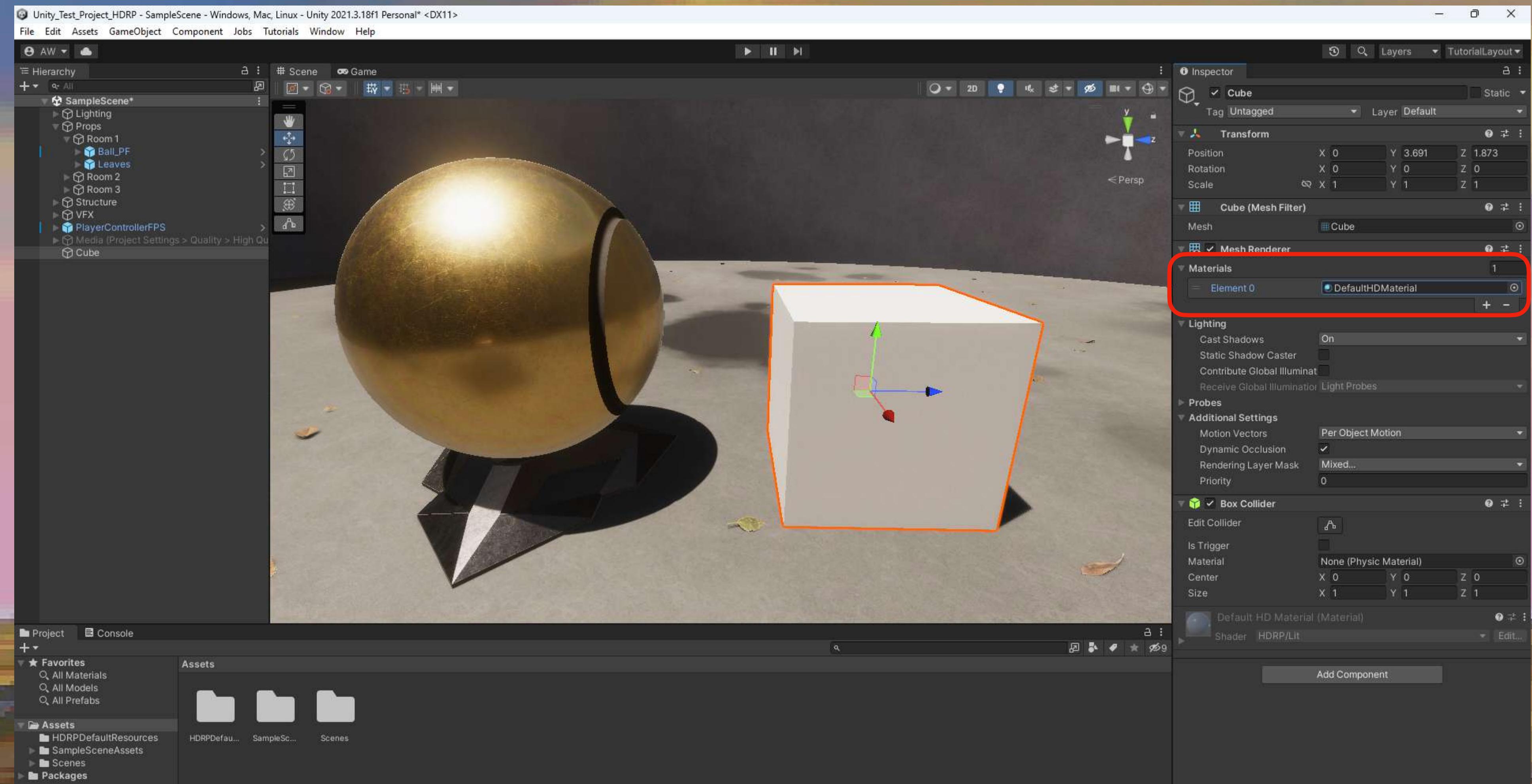
# Introduction to Unity

Click play and walk around the scene using the arrow keys/WASD and the mouse.

# Introduction to Unity



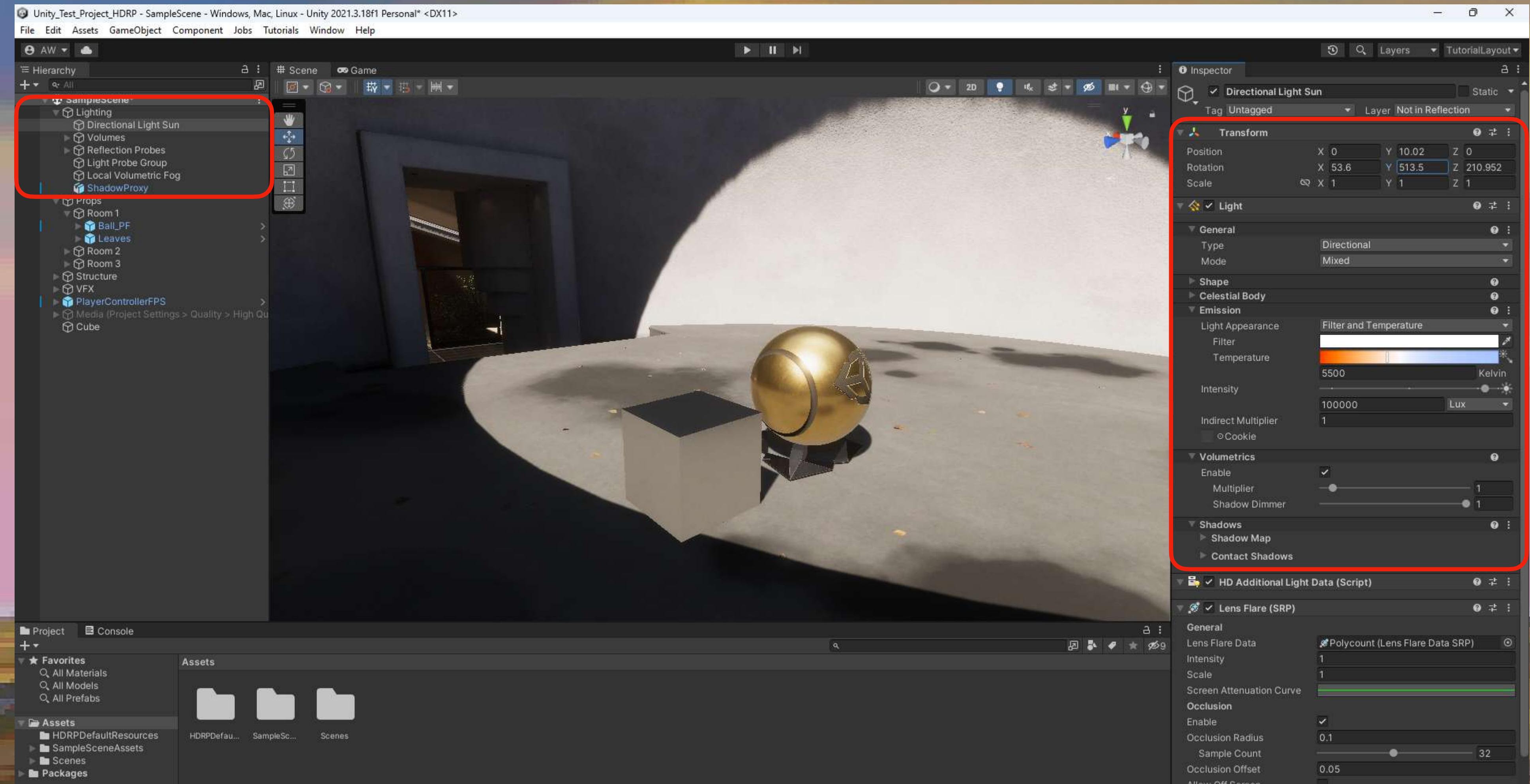
# Introduction to Unity



# Introduction to Unity

Create a new object in your scene. Apply a new material.

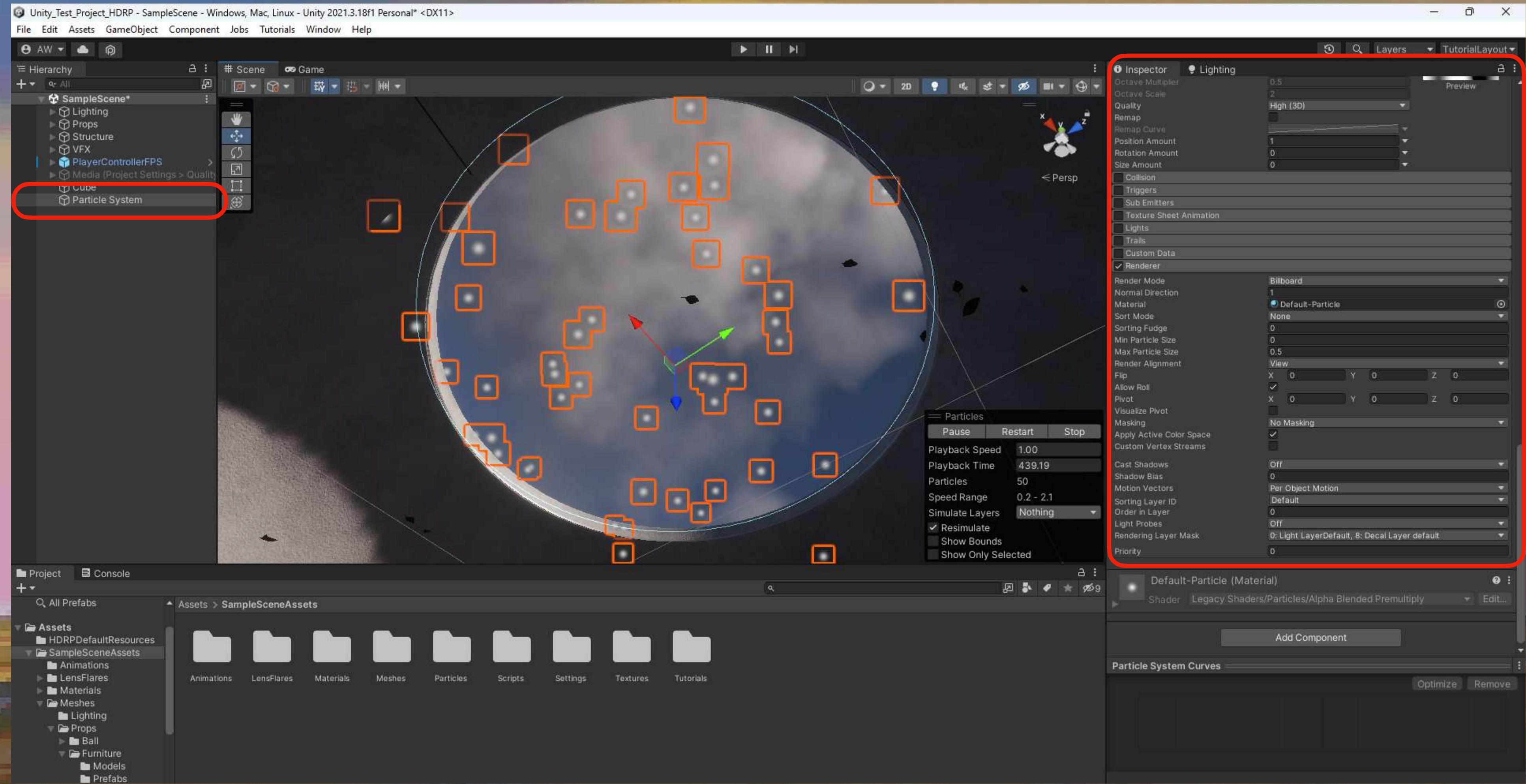
# Introduction to Unity



# Introduction to Unity

Give your scene a completely different lighting and atmosphere.

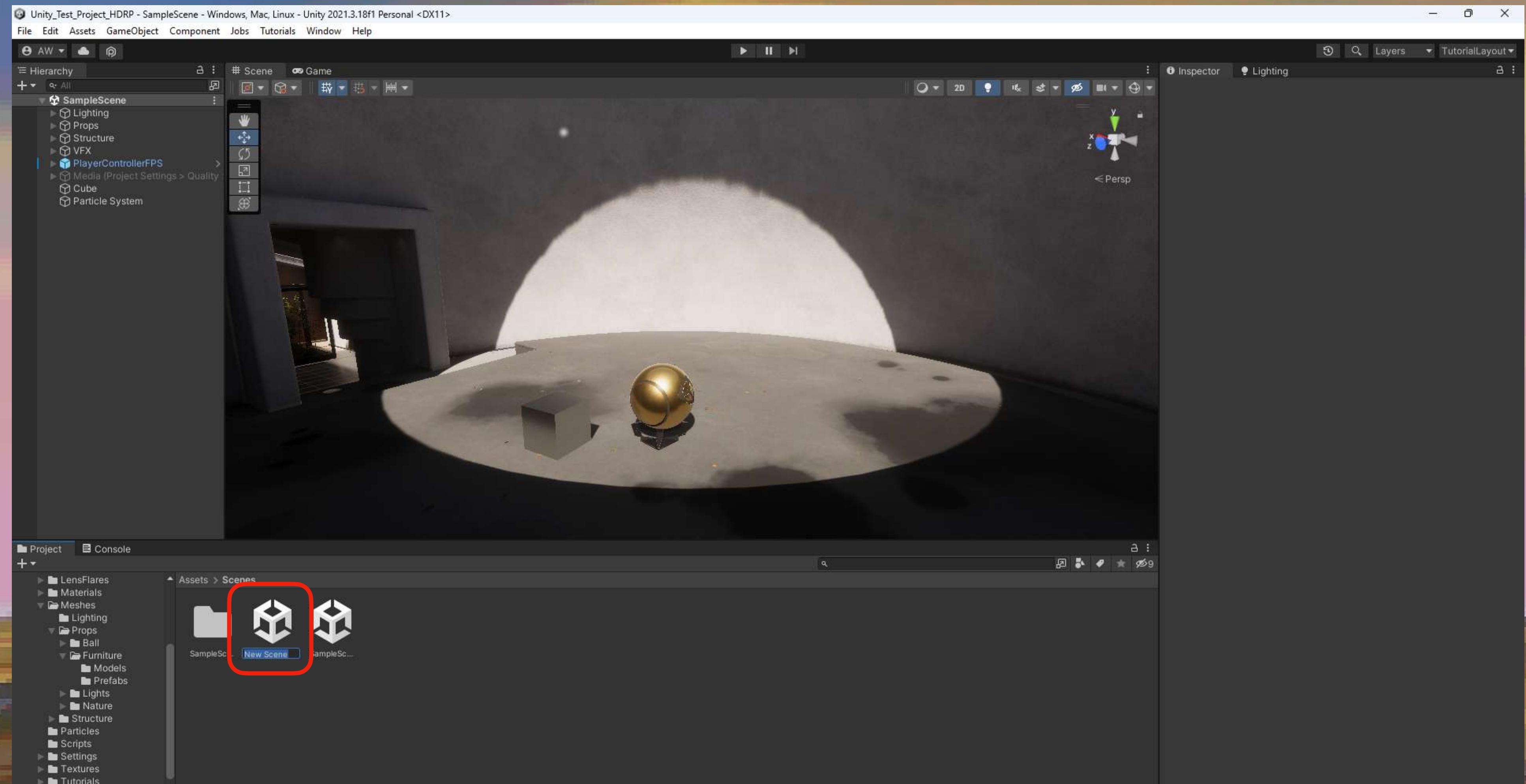
# Introduction to Unity



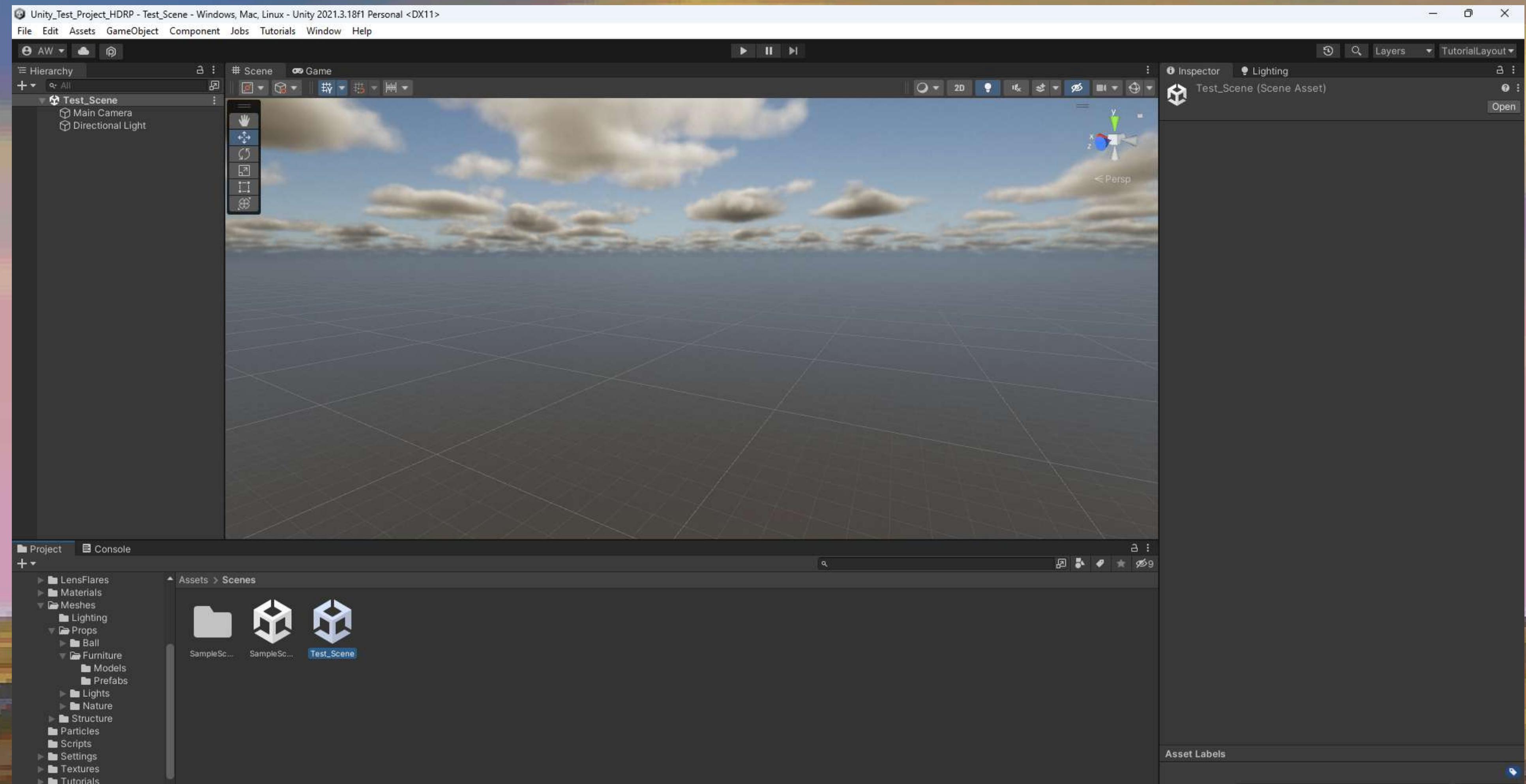
# Introduction to Unity

Create a swarm of fireflies flying overhead using particle systems.

# Introduction to Unity



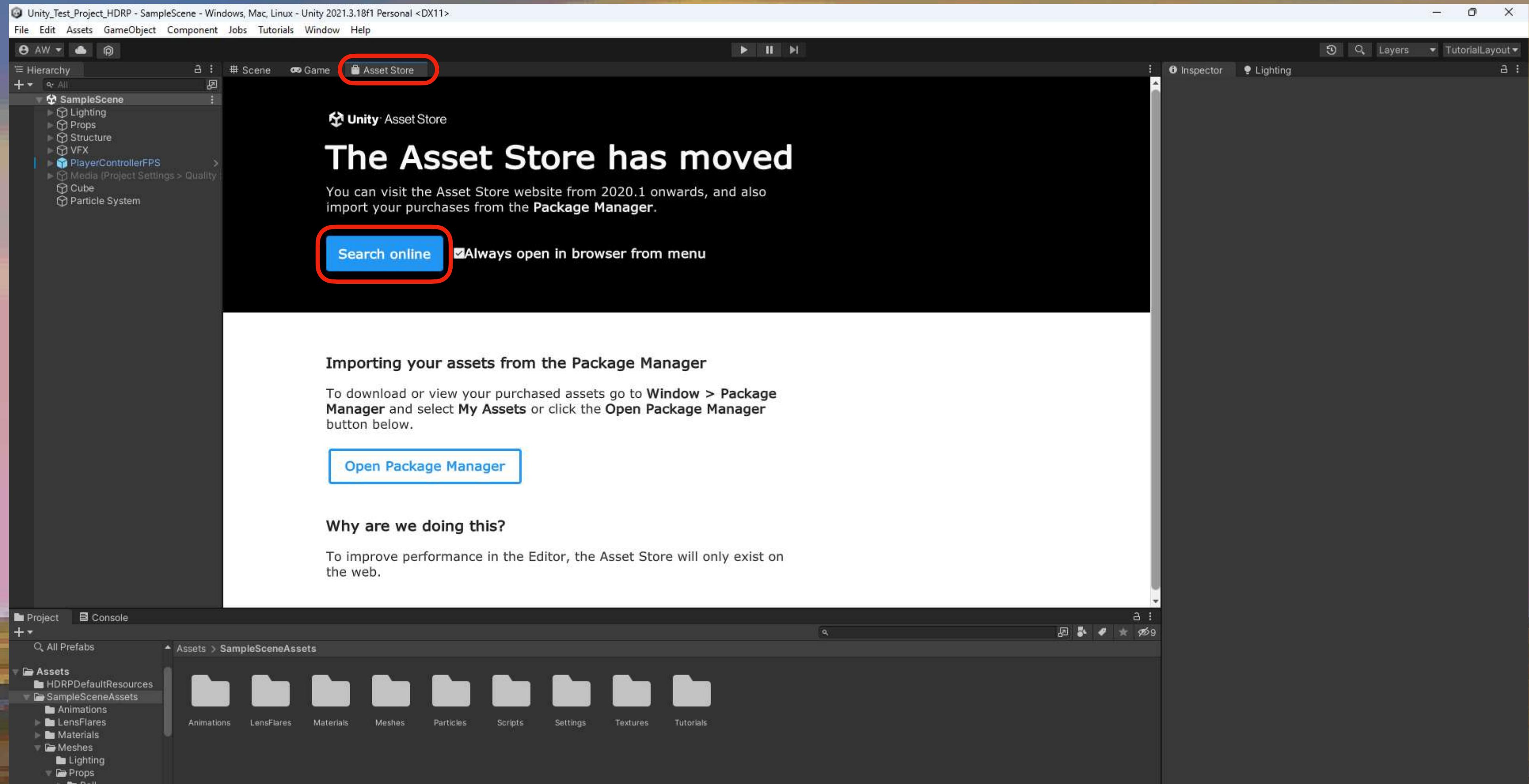
# Introduction to Unity



# Introduction to Unity

Create a new scene with no objects in it.

# Introduction to Unity



# Introduction to Unity

25-48 of 3207 results

Sort by Popularity View Results 24 Refine by clear filters

3D Free Assets

Hide Purchased Assets

All Categories

- 2D (1014)
- 3D (3207)
- Add-Ons (49)
- Audio (795)
- Decentralization (15)
- Essentials (42)
- Templates (325)
- Tools (2597)
- VFX (295)

Pricing

- Free Assets (3207)

Unity Versions

Publisher

Ratings

Platforms

Release Date

The screenshot displays the Unity Asset Store interface, specifically the search results for '3D' and 'Free Assets'. The results page shows 25-48 of 3207 available assets. The sidebar on the right contains various filtering options, with the 'Free Assets' checkbox being the most prominent, as it is highlighted with a red border. Other filter categories include All Categories, Unity Versions, Publisher, Ratings, Platforms, and Release Date.

**Assets Listed:**

- Zombie** by PXLTIGER (FREE)
- Apartment Kit** by BRICK PROJECT STUDIO (FREE)
- Guns Pack: Low Poly Gun...** by FUN ASSETS (FREE)
- Unity Terrain - HDRP Dem...** by PURCHASED (FREE)
- Sci-Fi Styled Modular Pack** by KARBOOSX (FREE)
- Quirky Series - FREE Anim...** by OMABUARTS STUDIO (FREE)
- Fantasy landscape** by PXLTIGER (FREE)
- RPG Character Mecanim ...** by EXPLOSIVE (FREE)
- Lemon Trees** by NUMENA (FREE)
- Mobile Tree Package** by LAXER (FREE)
- Lowpoly Baker's House** by EVGENIA (FREE)
- CITY package** by 255 PIXEL STUDIOS (FREE)

# Introduction to Unity

Unity Asset Store    Search for assets    Alexander alexanderwal...

3D    2D    Add-Ons    Audio    AI    Decentralization    Essentials    Templates    Tools    VFX    Sale    Sell Assets

Over 11,000 five-star assets    Rated by 85,000+ customers    Supported by 100,000+ forum members    Every asset moderated by Unity

Home > 3D > Environments > Urban > Russian buildings pack



Russian buildings pack

MadMedicSoft    ★★★★★ (7) | ❤ (296)

FREE

220 views in the past week

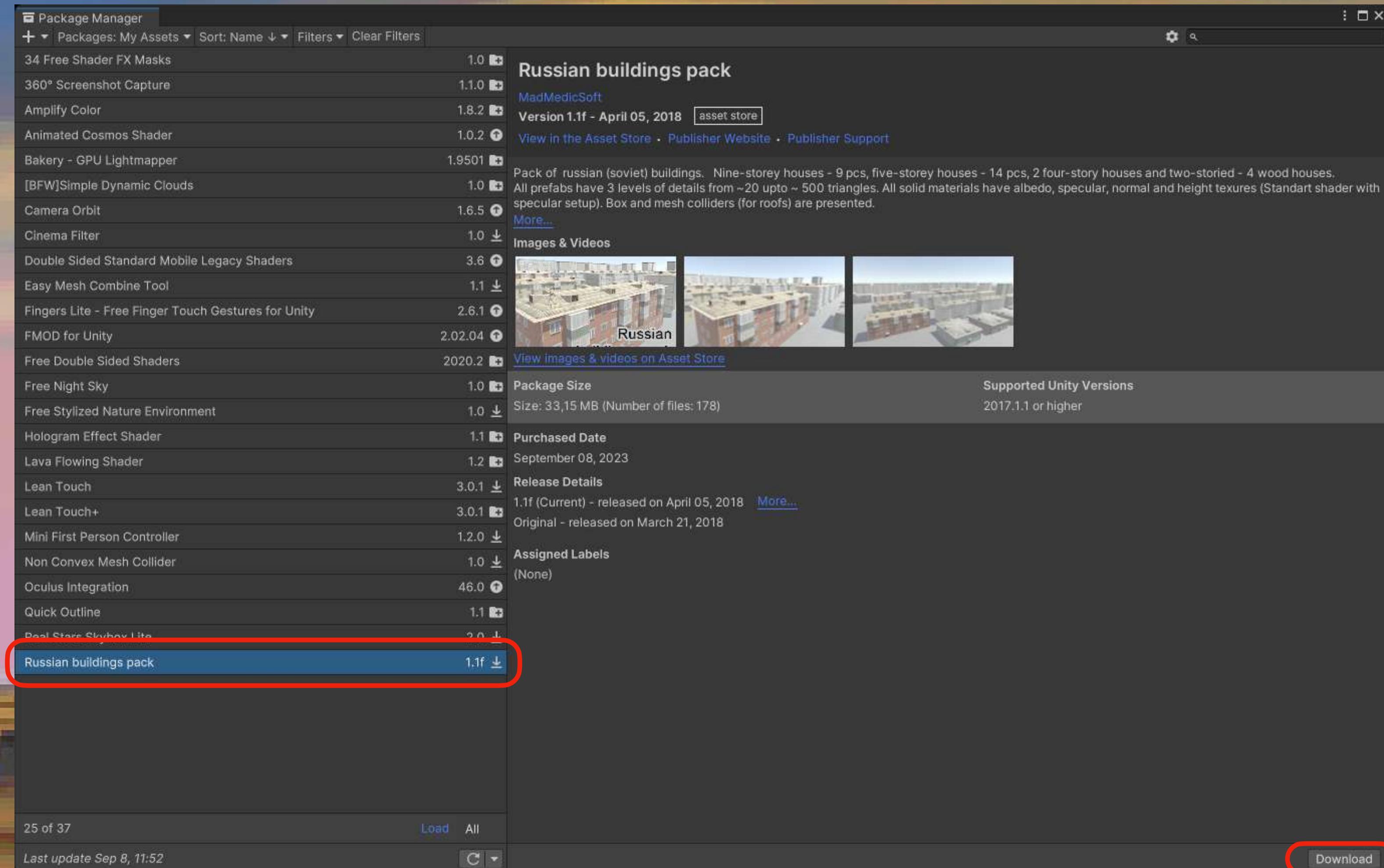
Add to My Assets

1/19

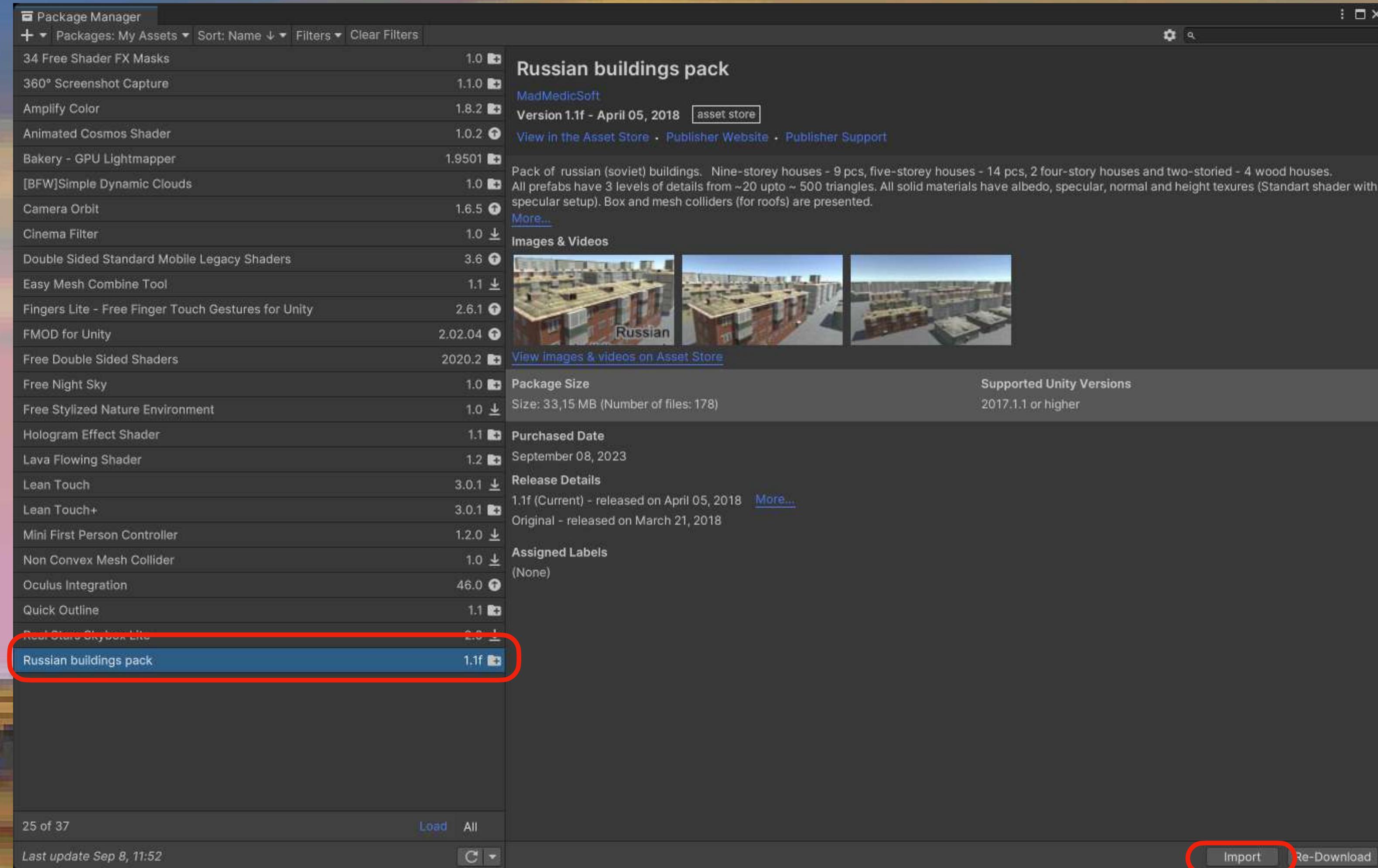
License agreement    Standard Unity Asset Store EULA  
License type    Extension Asset  
File size    33.2 MB  
Latest version    1.1f  
Latest release date    Apr 5, 2018  
Original Unity version    2017.1.1 or higher  
Support    Visit site

Frequently bought together

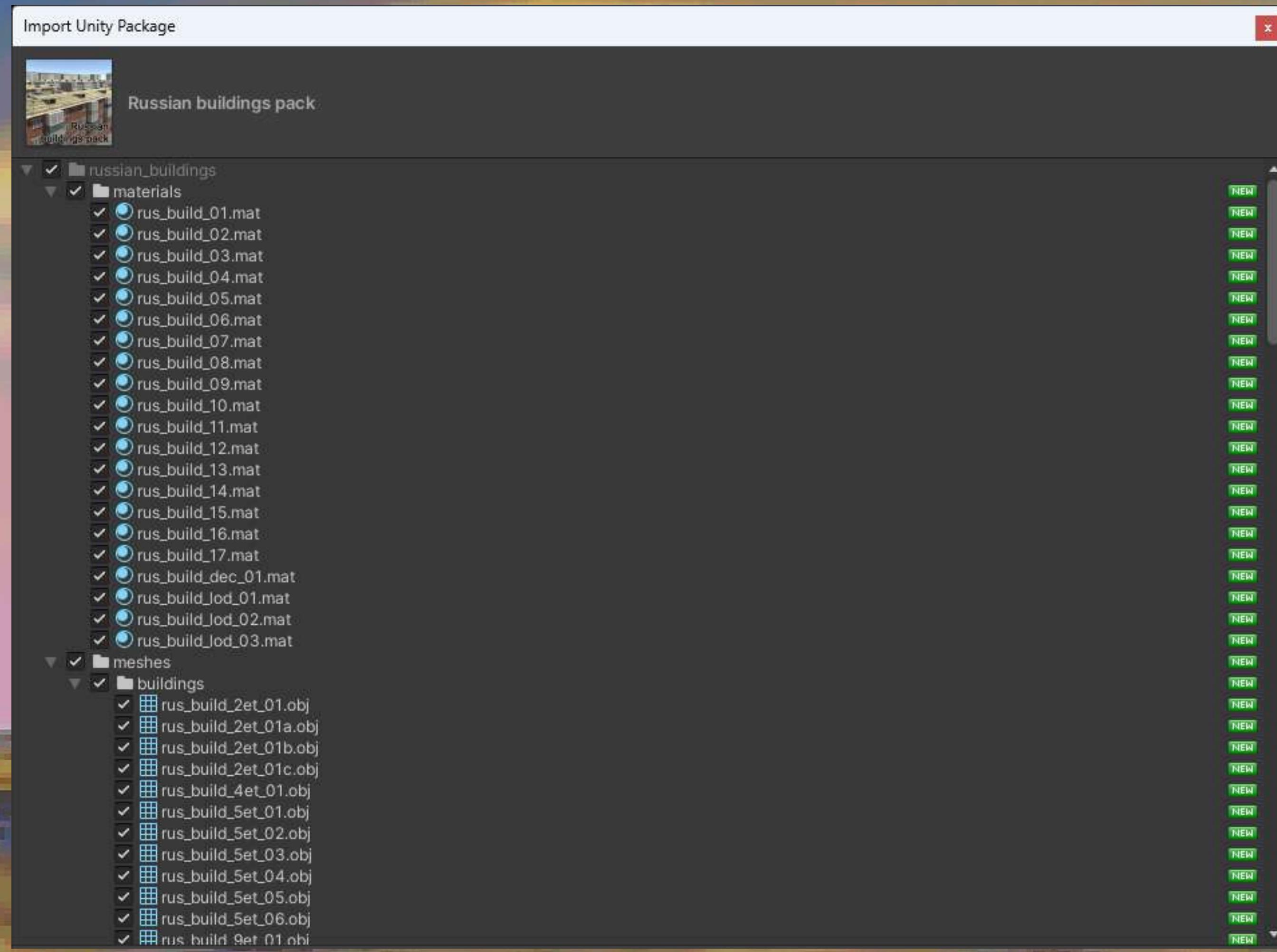
# Introduction to Unity



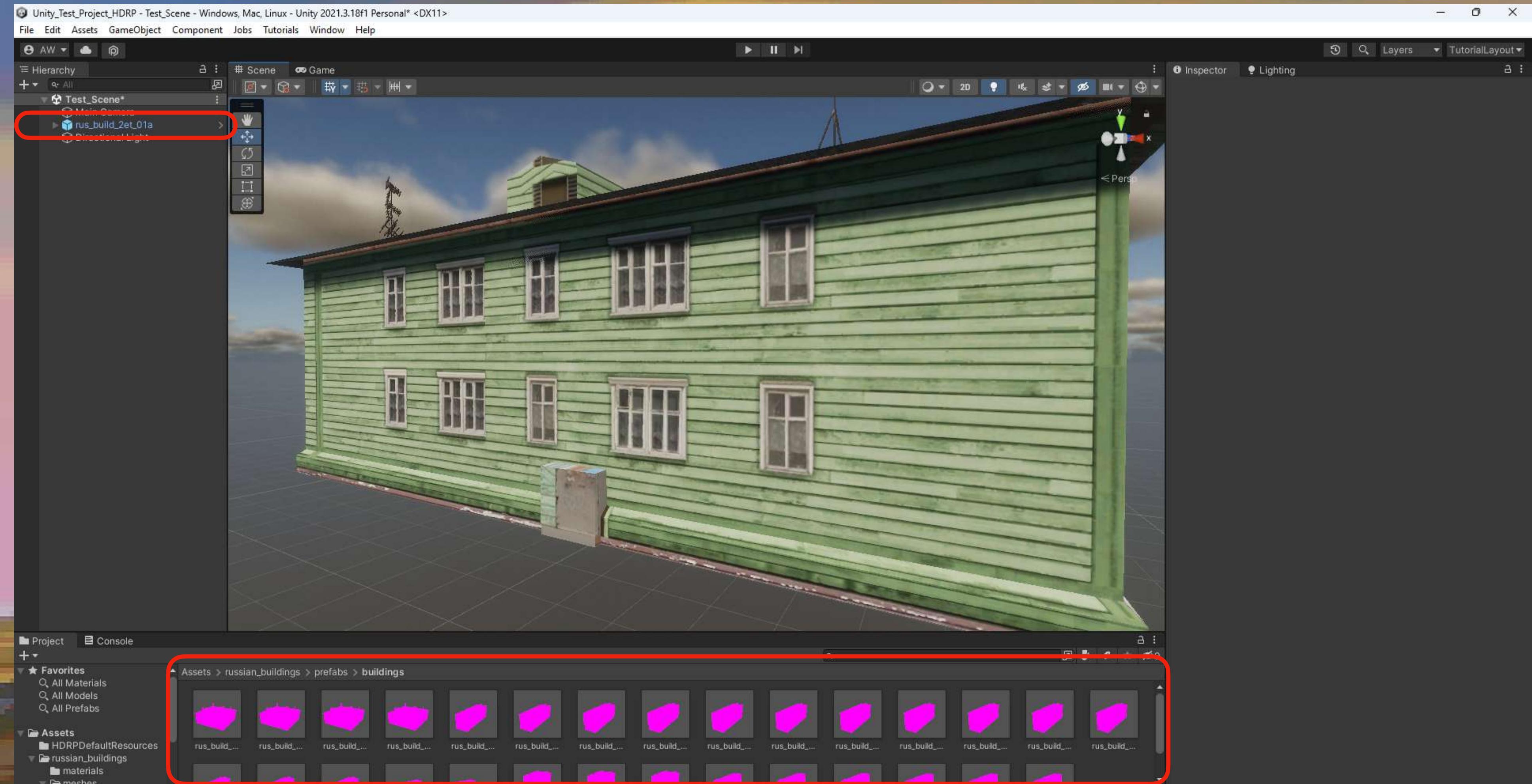
# Introduction to Unity



# Introduction to Unity



# Introduction to Unity



# Introduction to Unity

Open the Unity asset store.

Find a free 3D asset pack with enough assets to build a small scene. (TIP: Under "Pricing", tick the box that says "Free Assets").

Import package to your Unity scene.

Using the assets in asset pack and the default ones provided by the HDRP template, create a new scene from scratch that you can walk around.

# Where to go from here?

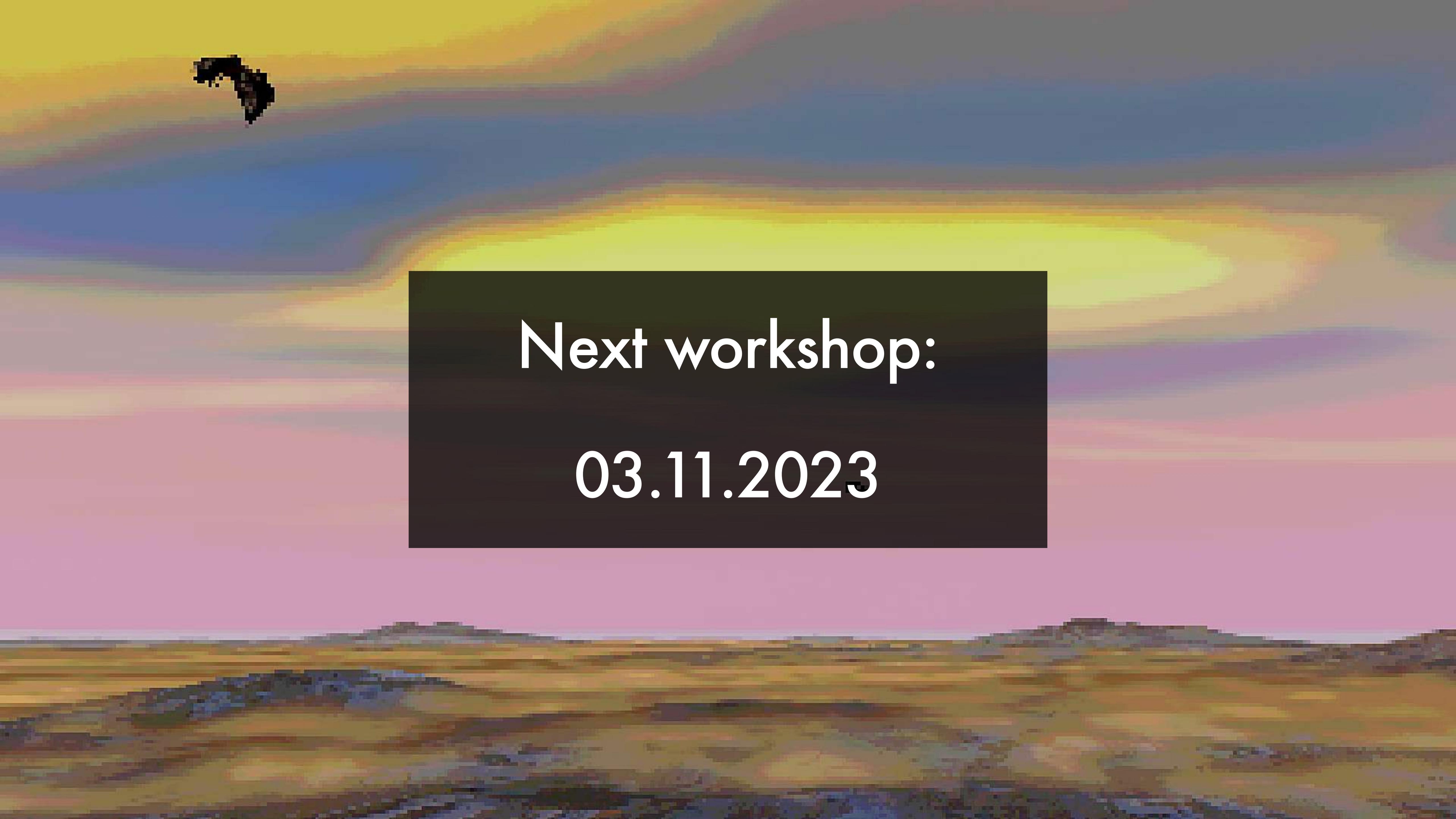
1. Continue playing around with Unity until you feel confident with the interface
2. (Bonus) Begin on the Junior Programmer pathway on Unity Learn (<https://learn.unity.com/pathway/junior-programmer>)
3. Decide whether you want to prototype a VR or AR experience for your personal project

## VR (Oculus Quest)

- Sign up for Meta account and enable multi-factor authentication
- Download Oculus mobile app
- Download Oculus desktop app (Windows only; Mac users skip this step)
- Create new project in Unity using 3D URP Core (NOT HDRP)
- Configure settings according to documentation (<https://developer.oculus.com/documentation/unity/unity-conf-settings/>)
- Download and import Oculus Integration package from Asset Store

## AR (Android/iOS smartphone)

- Create new project in Unity using AR Core template
- Download and install AR Foundation from Unity Registry (found in Plugins Manager)
- If deploying to Android device: ARCore XR Plugin
- If deploying on a iOS device: ARKit XR Plugin
- Configure settings according to documentation ([https://developers.google.com/ar/develop/unity-arf/getting-started-ar-foundation?hl=en#android\\_1](https://developers.google.com/ar/develop/unity-arf/getting-started-ar-foundation?hl=en#android_1))



A wide-angle photograph of a landscape at dusk or dawn. The sky is filled with horizontal bands of warm colors, transitioning from deep orange and yellow on the left to a darker blue on the right. In the foreground, there's a dark, silhouetted area that looks like a field or a low wall. A single, small, dark tree stands out against the colorful sky on the left side.

Next workshop:

03.11.2023