

# Assets and Art Overview

CS 415: Game Development

Professor Eric Shaffer

# What Are Assets?

## Game assets

Game assets is **a broad term used to describe every individual element that you'll find in a game**. These can include artwork, 3D models, the GUI (interface), animation, special effects, music, voiceovers, and even scripting for things like physics. Mar 7, 2022

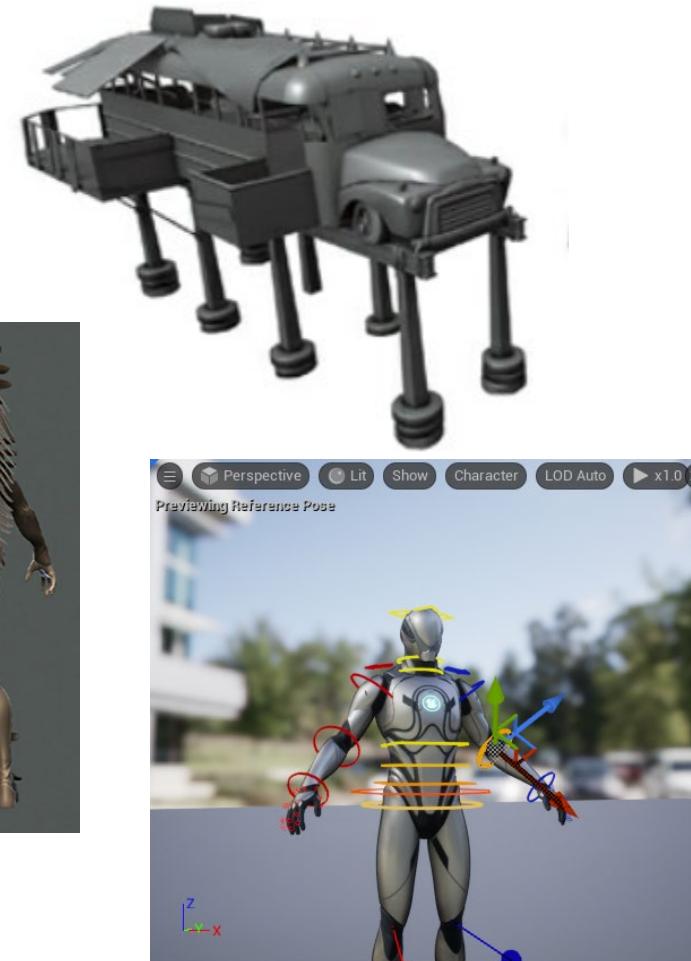
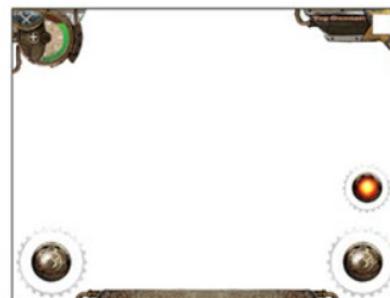
The screenshot shows the Unreal Engine Marketplace homepage. At the top, there's a banner with the text "September Free Content now available!" and several thumbnail images of different assets. Below the banner, the header includes the Unreal Engine logo, navigation links for PRODUCTS, SOLUTIONS, NEWS & EVENTS, LEARN, COMMUNITY, SUPPORT, and MARKETPLACE, along with a search bar and a globe icon.

The main content area is titled "MARKETPLACE" and features a "Showcase" section. It displays six asset packages:

- PACK 01 64 TILEABLE MATERIALS - 4K**: Includes four spherical material preview images. Price: \$29.99 **\$17.99**.
- Archer's Animations**: Includes a thumbnail of several white humanoid figures in various poses. Price: \$19.99.
- Real Bed Collection 01 (10 Models)**: Includes a thumbnail of a modern bedroom interior. Price: \$12.99 **\$6.49**.
- TV Studio**: Includes a thumbnail of a television studio set. Price: \$29.99.
- Modular Asian Medieval City**: Includes a thumbnail of a traditional Asian city street scene. Price: \$89.99 **\$62.99**.
- Tempest Combat Framework**: Includes a thumbnail of a futuristic combat scene. Price: \$149.99 **\$127.49**.

# Video Game Art

- Concept Art
- Environment Art
- Character Art
- Character Animation
- User Interface Design



# Infinity Blade

Over 1000 pieces of content free in UE MarketPlace  
Epic says that's a \$4M investment

RESULTS FOR "INFINITY BLADE"

Sort by: Relevancy

Image	Name	Developer	Status	Type
	Infinity Blade: Props	Epic Games	Free	Props
	Infinity Blade: Castle	Epic Games	Free	Environments
	Infinity Blade: Hideout	Epic Games	Free	Environments
	Infinity Blade: Temple	Epic Games	Free	Environments



## Infinity Blade

Video game

96% liked this video game

Google users



Infinity Blade is an action role-playing game developed by Chair Entertainment and Epic Games and released through the Apple App Store on December 9, 2010. It was the first iOS video game to run on the Unreal Engine. [Wikipedia](#)

**Initial release date:** December 9, 2010

**Series:** [Infinity Blade series](#)

**Publisher:** [Epic Games](#)

**Designer:** [Cliff Bleszinski](#)

**Mode:** [Multiplayer video game](#)

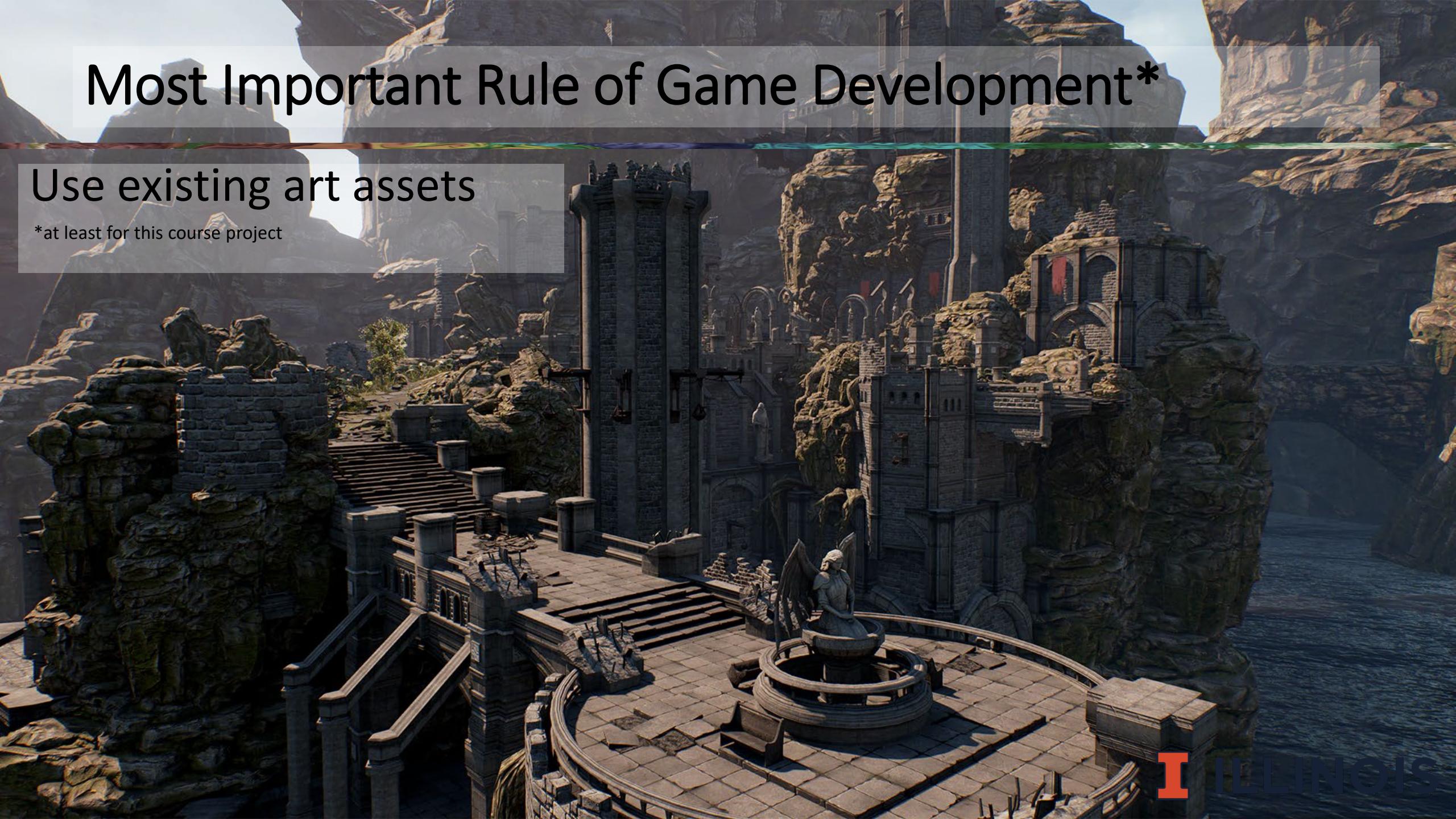
**Developers:** [Chair Entertainment, Epic Games](#)

**Platform:** [iOS](#)

# Most Important Rule of Game Development\*

Use existing art assets

\*at least for this course project

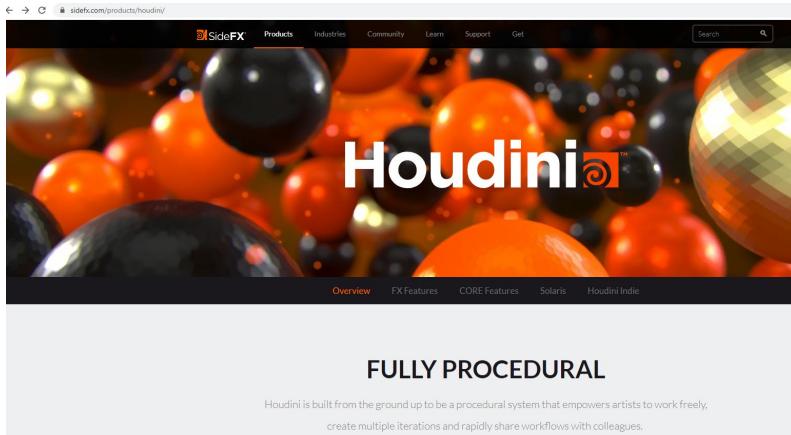


# Where to Get Assets

- UE Marketplace <https://www.unrealengine.com/marketplace/en-US/store>  
Everything for Unreal
- Unity Asset Store <https://assetstore.unity.com/>  
Everything for Unity
- Sprite Database <https://spritedatabase.net/>  
Good for 2D game art
- Games Clip Art <https://www.gograph.com/vector-clip-art/games.html>  
Good for board/card/puzzle game art
- GameDev Market <https://www.gamedevmarket.net/>  
Oriented towards indie game dev

# A Few Digital Art Tools

- **Photoshop**  
Classic...used for 2D art...texture generation, etc.
- **Blender**  
Free open source for 3D modeling, animation, etc.
- **Maya**  
Commercial Blender-like software...possibly more pro
- **Houdini FX**  
Super pro animation and special effects

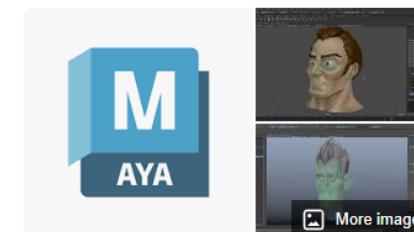


Blender  
Computer software



Blender is a free and open-source 3D computer graphics software tool set used for creating animated films, visual effects, art, 3D-printed models, motion graphics, interactive 3D applications, virtual reality, and, formerly, video games. [Wikipedia](#)

Autodesk Maya  
Computer application



Autodesk Maya, commonly shortened to just Maya, is a 3D computer graphics application that runs on Windows, macOS and Linux, originally developed by Alias and currently owned and developed by Autodesk. It is used to create assets for interactive 3D applications, animated films, TV series, and visual effects. [Wikipedia](#)

# Tools in Unreal Engine



## What's New

Information about new features in each release of Unreal Engine

## Understanding the Basics

Essential skills and concepts to help you get started in Unreal Engine.

## Working with Content

Information on using art created in external applications, importing it into Unreal Engine 4, and setting it up for use in visualization and interactive applications.



## Building Virtual Worlds

Information on the tools and techniques for interactive environment and level design.



## Designing Visuals, Rendering, and Graphics

Rendering subsystem including lighting and shadowing, materials and textures, visual effects, and post processing.



## Creating Visual Effects

Use Unreal Engine's Niagara visual effects system to create and preview particle effects in real time.



## Programming and Scripting

How to use the programming and scripting languages and tools for controlling Unreal engine programmatically at runtime.



## Making Interactive Experiences

How to create gameplay mechanics, behaviors, and conditions that make the virtual world responsive to players carrying out actions over time.



## Animating Characters and Objects

Explore Unreal Engine's animation and cinematic tools.

# Example: Foliage Generation



Unreal Engine 5 is a powerful engine able to render massive open worlds full of complex flora and fauna. The Open World Tools can be used to procedurally fill large spaces with Static Mesh assets to create outdoor spaces that feel natural and alive.



## Foliage Mode

How to render Static Mesh or Actor Foliage on the surfaces of other geometry for creating ground cover effects.



## Grass Quick Start

Learn how to add Grass textures to a landscape.



## Open World Tools Property Reference

A listing of properties and descriptions for the Open World tools.



## Open World Tools Tips

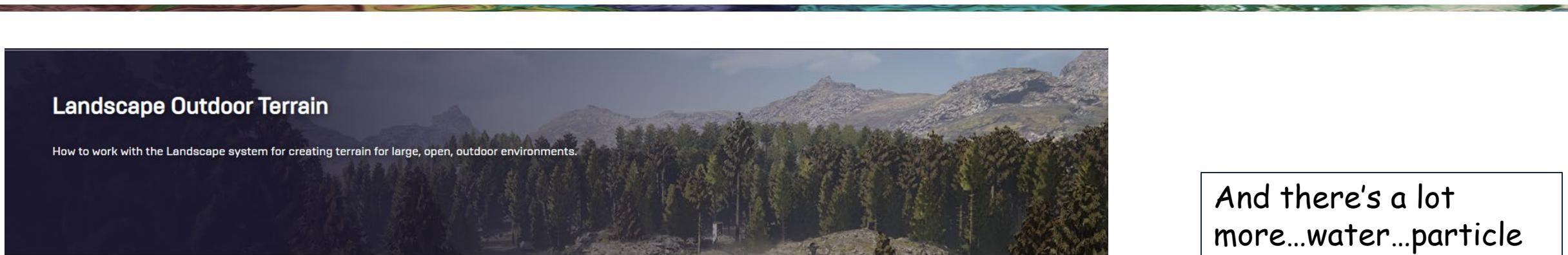
A collection of tips and tricks about how to get the most out of the Open World Tools.



## Procedural Foliage Tool

How to set up and use the Procedural Foliage tool.

# Example: Landscape Generation



## Landscape Outdoor Terrain

How to work with the Landscape system for creating terrain for large, open, outdoor environments.

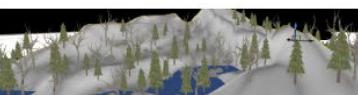
The Landscape system inside of Unreal Engine is a collection of tools that allow you to create expansive outdoor environments.

The topics below will help you Sculpt your Landscape Actors, create customized Materials, and paint them on your Landscape to create your own immersive world.



### Editing Landscapes

Guide to using the tools for editing Landscape terrains.



### Landscape Collision Guide

Collision settings for Landscape.



### Landscape Edit Layers

Explains the Landscape Edit Layers feature for non-destructive layers and splines in heightmaps.



### Landscape Material Layer Blending

Setting up materials to work with Landscape terrains



### Landscape Overview

An overview of the Landscape Outdoor Terrain System and how to use it in your



### Landscape Quick Start Guide

Getting up and running with the basics of the Landscape System in Unreal Engine.



### Landscape Technical Guide

Understanding the construction and dimensions of Landscape Outdoor Terrain.



### Creating Landscapes

Guide to creating new Landscape terrains.

And there's a lot more...water...particle system effects...you should explore the documentation...pretty much anything you need for your project can be done in just UE

# 3D Modeling

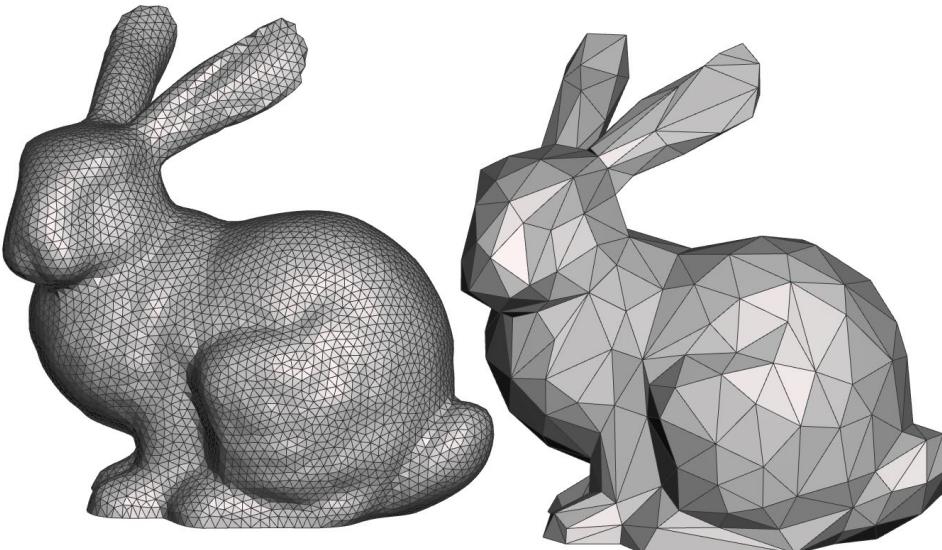
One of the main art components of many games are 3D surface meshes

- Static meshes in UE

***Surface*** because rendering simulates light reflected from surfaces

***Mesh*** because the surface representation is typically a mesh of triangles

- Modern GPUs optimized to process triangle data



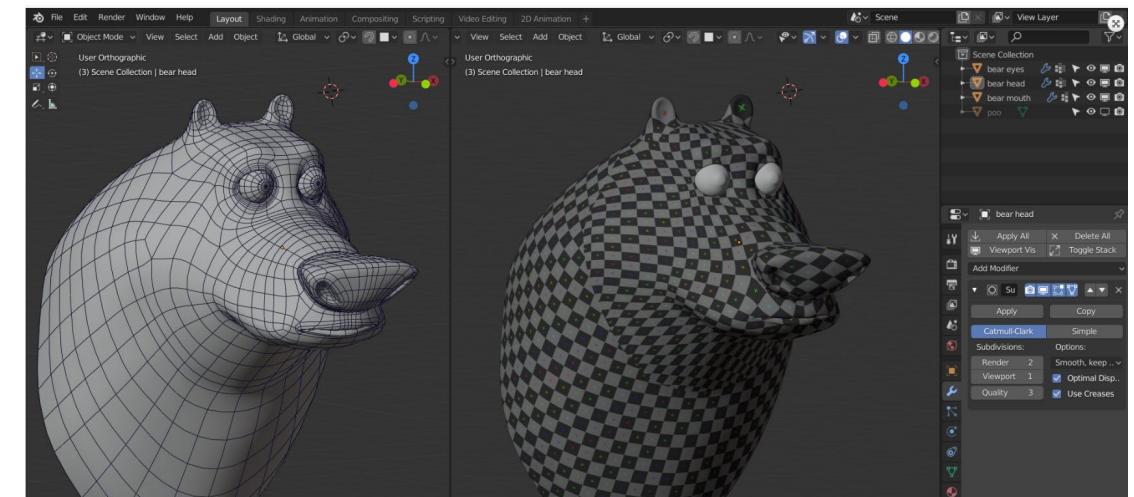
Artists often refer to models with few triangles as ***low poly*** art. What is the advantage of low poly art?

# Generating 3D Models

Many different ways to generate 3D models including

- Scanning
- Procedural generation
- Created by artist with modeling tool
  - Implicit surfaces
  - Patch-based modeling
  - Subdivision surfaces

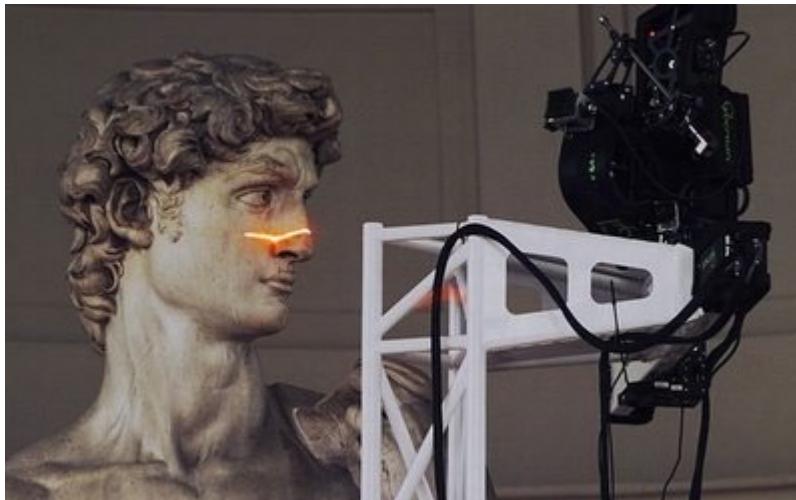
Surface may start out represented not as a polygonal mesh...but will be converted when exported...or imported...or at runtime depending on the engine and representation



# 3D Scanning

LiDAR (“light detection and ranging”)

- Bounces a laser off surface, measures distance by timing the bounce



Surface may start out represented as points...need to be processed by denoising algorithm and surface reconstruction to generate a mesh

- Can capture imagery as well for texture

Other techniques for scanning exist...photogrammetry

# Accessing Scan Data in UE

The image shows a screenshot of the Unreal Engine Bridge website at unrealengine.com/en-US/bridge. The top navigation bar includes links for BACK TO TOP, OVERVIEW, FEATURES, SHOWCASE, and GET STARTED. The main heading "Bridge—made for Unreal Engine" is displayed above a grid of thumbnail images representing various scan data assets. These thumbnails include labels such as TRAIN TRACKS, DEFINITIVE INTERIOR PLASTER, ARCTIC ICE MELT, URBAN TRASH, NEOCLASSICAL MODULAR BUILDING VOL. 2, KODIC COASTAL CLIFFS, SANDS, CONCRETE, BLOOD, and ABSURITE. To the right, there is a screenshot of the Unreal Engine interface showing the Bridge tab open, displaying a library of Megascans assets alongside a photograph of a desert landscape.

Bridge—made for  
Unreal Engine

01 OVERVIEW    02 FEATURES    03 SHOWCASE    04 GET STARTED

BACK TO TOP

TRAIN TRACKS    DEFINITIVE INTERIOR PLASTER

KODIC COASTAL CLIFFS    NEOCLASSICAL MODULAR BUILDING VOL. 2    URBAN TRASH

SANDS    CONCRETE    BLOOD

ABSURITE

Free, unlimited access to Megascans

Megascans assets are free for use with Unreal Engine. Sign in using your Epic Games account to get instant access to thousands of real-world scanned 3D assets and surfaces, with many more being uploaded every single day.

Work right inside Unreal Engine

The entire Megascans library is instantly accessible to artists right within Unreal Engine. Open the Bridge tab and simply drag and drop Megascans

# Artist Created 3D Models: Implicit Surfaces

Define a surface using a function like  $f(x, y, z) = x^2 + y^2 + z^2 - 1$  and  $f(x, y, z) = 0$

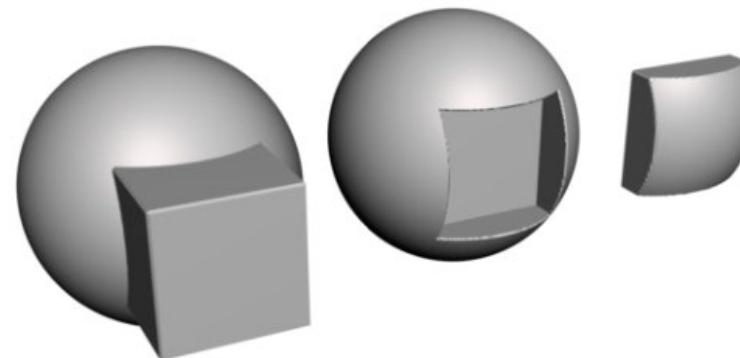
- Negative points inside surface...positive points are outside

Boolean operations between implicit functions can model more complicated shapes

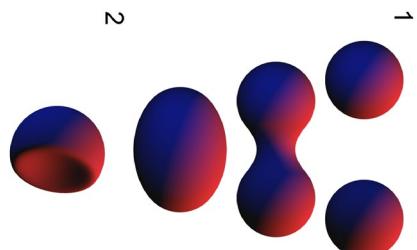
$$F_{union} = \min(F_A, F_B)$$

$$F_{intersection} = \max(F_A, F_B)$$

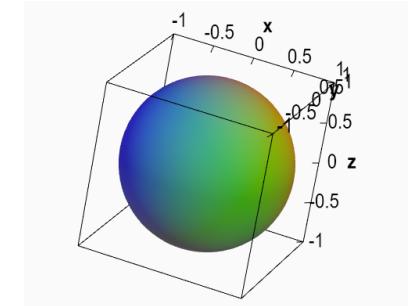
$$F_{difference} = \max(F_A, -F_B)$$



Can use functions like sums of Gaussian functions to blend implicit surfaces together



What are pros and cons of using implicit surfaces?



**RENDERMAN**  
ARTIST TOOLS

PhotoRealistic RenderMan  
Application Note #31

Blobby Implicit Surfaces  
September, 1999

**Introduction**

**Note**  
Please set your monitor gamma to 2.3 or thereabouts. Otherwise the pictures will look dark and ugly. If they're ugly but not dark, your gamma is ok.

PhotoRealistic RenderMan now supports a new surface type, **RIBlobby**, to render blobby implicit surfaces in the style of Jim Blinn's blobby molecules, Nishimura et al's *Metaballs* and Wyvill et al's soft objects. Blobby models are built from simple primitive shapes (ellipsoids and sausage-like cylinders with rounded ends in the current implementation) blending together into rounded, tubby, globular masses. They are best used in situations where users want to model amorphous shapes and are willing to trade away fine control over details of shape for ease of modeling gross features. Jim Blinn's models of complex molecular structures and the splatting raindrops in *A Bug's Life* are good applications of Blobby models.

The most important feature of blobby implicit models is *automatic blending*. Blobby shapes placed close to one another can be blended together (or not, as the modeler wishes) with no need for modelers to construct explicit round and fillet patches joining them. The two objects in the picture to the left contain identical clusters of spheres: unblended on the left, blended on the right.

# Artist Created 3D Models: Implicit Surfaces

Representation requires little storage

- Just code or text defining the functions

Very good for smooth surfaces

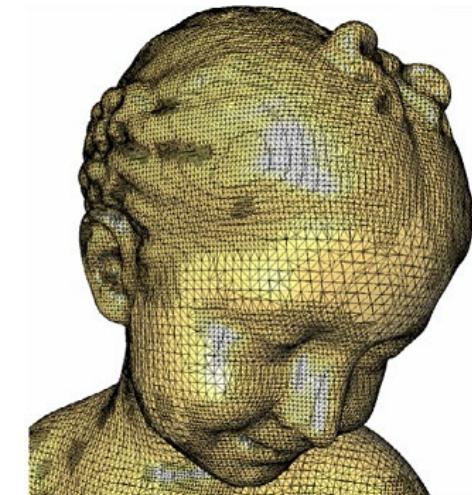
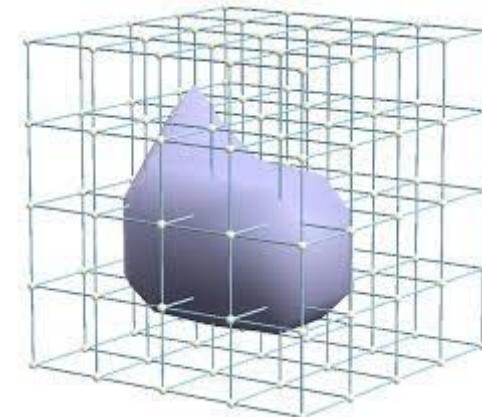
- Can do sharp features, too

Good for ray-tracing

Not great for small detailed surfaces

Polygonalization time-consuming

- Need to use isosurfacing algorithm like marching cubes
- Leads to high poly models



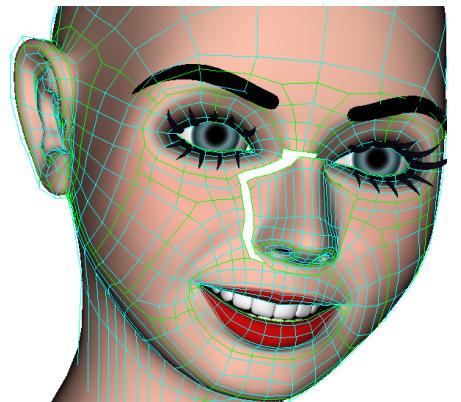
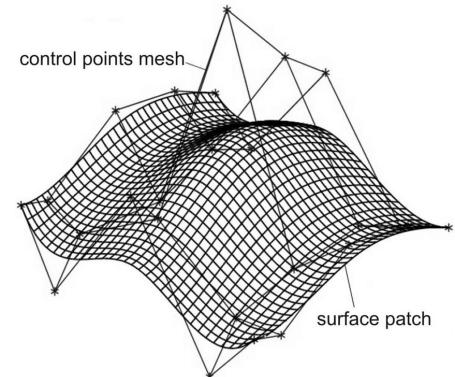
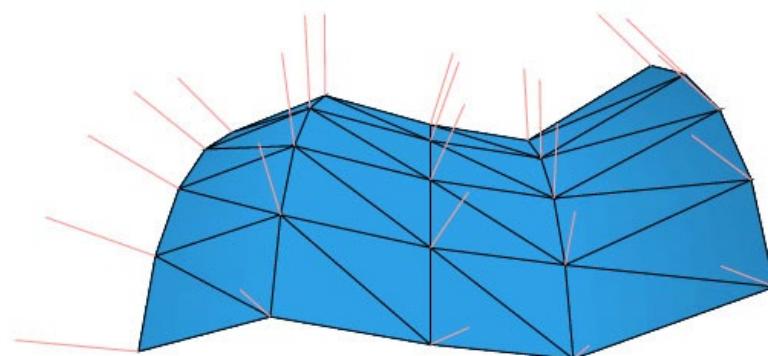
# Artist Created 3D Models: Spline Patches

Artist picks control points in space to define surface patch based on polynomials

Stitch patches together to form a surface

Can generate a grid of points that lie on the patch

Generate a pair of triangles for each quad on that grid



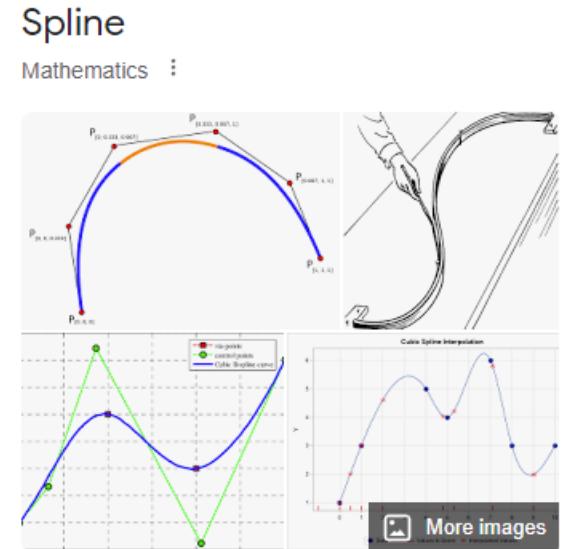
# Artist Created 3D Models: Spline Patches

Often referred to as NURBS modeling

- Non-uniform rational B-Spline
- Fairly compact representation
- Easily polygonalized and at any resolution
- Not great for ray-tracing
- Patches can crack apart due to numerical issues during animation
- Not great for sharp features

Spline curves also useful for defining camera paths.

We will take a close look at Bezier curves and patches which are very similar to NURBS but more easily understood.

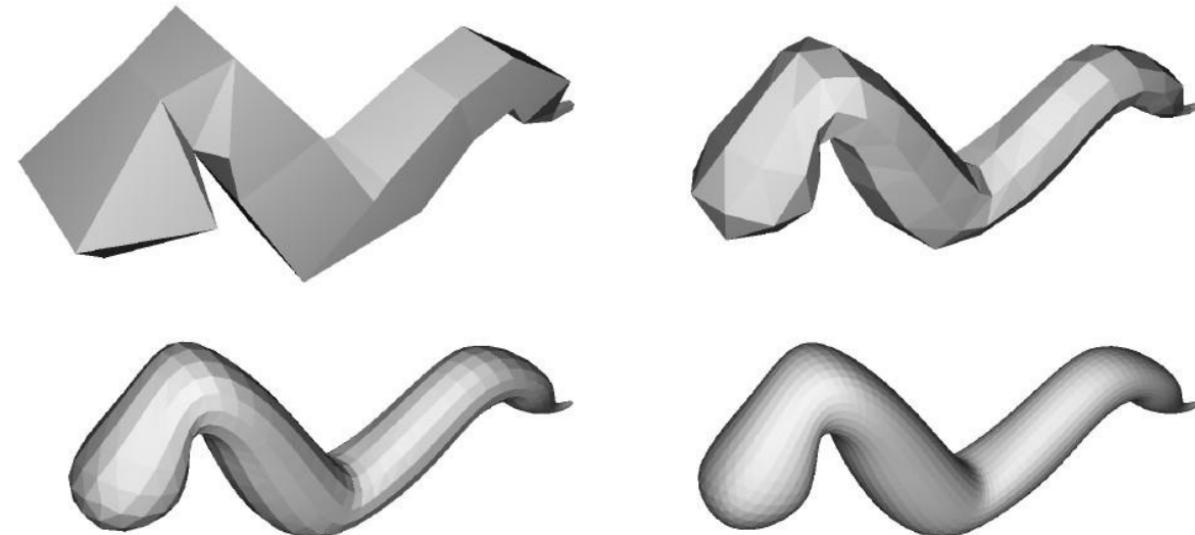


In mathematics, a spline is a special function defined piecewise by polynomials. In interpolating problems, spline interpolation is often preferred to polynomial interpolation because it yields similar results, even when using low degree polynomials, while avoiding Runge's phenomenon for higher degrees. [Wikipedia](#)

# Artist Created 3D Models: Subdivision Surfaces

Artist creates a control net of points that define a polygonal surface

- Set of subdivision rules allows computer to refine the mesh and generate smoothness



# Subd versus NURBS

Subd can keep sharp features (artists tagged edges)

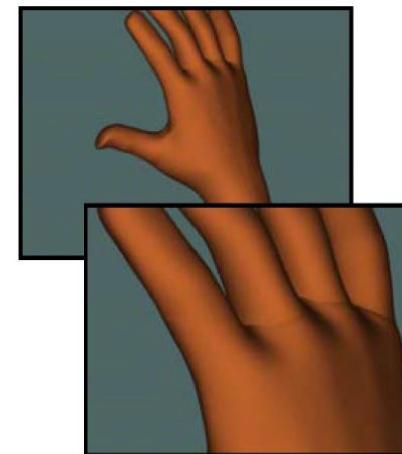
Subd can still represent smooth surfaces as well

Base level mesh usually not too large

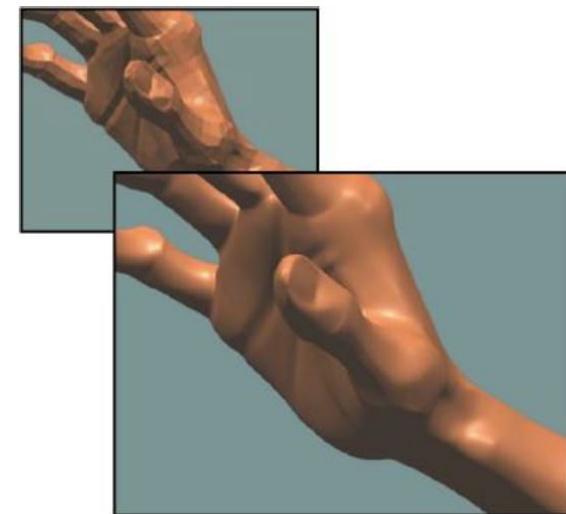
Subdivision can be done on GPU at runtime

## Example: Geri's Game (Pixar)

Woody's hand (NURBS)



Geri's hand (subdivision)



# Procedural Modeling

Have the computer generate the geometry with minimal(?) artist input!  
Good for highly detailed natural geometry like terrain  
Often based on fractal/recursive methods  
Can be done as pre-process or at run-time depending on situation



Can also do urban settings surprisingly well !

# AI Generated Art

Art is a big bottleneck in production time and it is expensive  
...a lot of the current interest in AI generated art is from the game industry



Top stories

S Smithsonian Magazine  
[Art Made With Artificial Intelligence Wins at State Fair](#)  
2 days ago

Open Culture  
[An AI-Generated Painting Won First Prize at a State Fair & Sparked a Debate About the Essence of Art](#)  
1 day ago

F Forbes  
[AI Ethics Left Hanging When AI Wins Art Contest And Human Artists Are Fuming](#)  
1 day ago