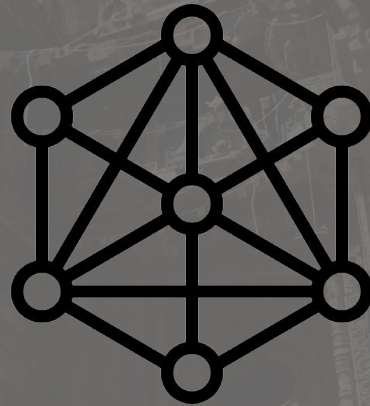


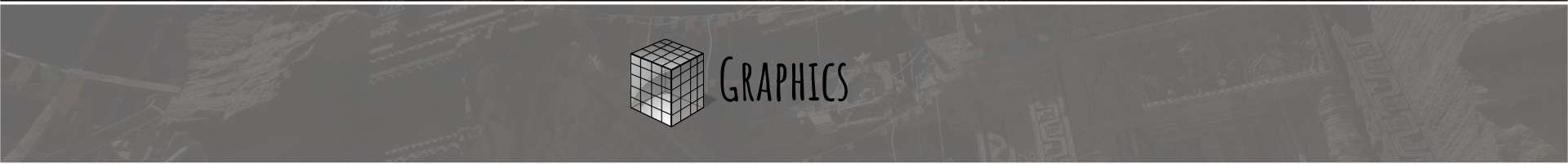
# LESSON 4

## GRAPHICS



NICOLAS SERF

SERF.NICOLAS@GMAIL.COM







GRAPHICS



MESH



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

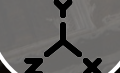
## COLLECTION



## SOCKETS



## COORDINATES



## GENERATION



## DEFINITION

BY NATURE, A MESH IS SIMPLY A COLLECTION OF VERTICES, CONNECTED ALL TOGETHER IN ORDER TO FORM A SHAPE.

## IMPORT

EXCEPT IF YOU USE SOME PRIMITIVE SHAPE, YOU'LL NOT CREATE MESHES IN GAME ENGINE, THEY ARE CREATED INSIDE EXTERNAL TOOLS AND IMPORTED IN THE ENGINE.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## SOCKETS



## COLLECTION



## COORDINATES



## GENERATION



## DEFINITION

A SOCKET IS SOMETHING YOU CAN ATTACH TO A MESH PART. EVEN IF WE GENERALLY USE THAT FOR SKELETAL MESH, THE NOTION IS PRESENT ON THE MESH LEVEL. A SOCKET WILL BE CREATED IN ORDER TO ATTACH SOMETHING ON IT LATER

## EXAMPLES

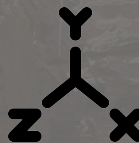
THERE IS MULTIPLES EXEMPLE, AN OBVIOUS ONE CAN BE A PLACE TO ATTACH A SMOKE EFFECT WHEN WE ARE FIRING FROM A WEAPON. IT IS USUALLY THE MUZZLE SOCKET.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## COORDINATES



## COLLECTION



## SOCKETS



## GENERATION



## DEFINITION

A SOCKET HAS A COORDINATES LIKE ANYTHING ELSE. BEING

## LOCAL VS WORLD

COORDINATE IS OBVIOUSLY A LOCAL COORDINATE RELATED TO THE PART IS IT ATTACHED TO. YOU CAN ALWAYS TRANSLATE IT INTO WORLD COORDINATE.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## GENERATION



## COLLECTION



## SOCKETS



## COORDINATES



## DEFINITION

A COMMON USED FEATURE IN EACH ENGINE IS THE POSSIBILITY TO **CREATE A MESH** AT **RUNTIME** BY FEEDING VERTICES. IT IS MOSTLY USED FOR PROCEDURAL GENERATION ROOM FOR EXAMPLE.

## RUNTIME

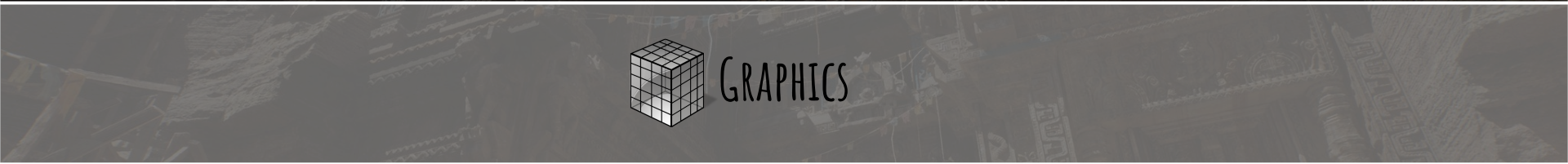
AT **RUNTIME**, IT COMES WITH **PERFORMANCES COSTS**, AS BUILDING A COMPLEX MESH WITH MANY VERTICES TO BE DEFORMABLE IF NEEDED FOR EXAMPLE IS COSTLY. IT CAN **CREATE A FREEZE** DURING THE **GENERATION**

## MERGE

ANOTHER COMMONLY EXPOSED FEATURES IN ENGINE IS TO **SIMPLE MERGE MULTIPLES MESHES** INTO A ONE. THE BENEFIT OF THAT IS THAT WE'LL ONLY NEED **1 DRAWCALL** TO DISPLAY THE MESHES. BUT YOU NOW HAVE **ONLY 1 ACTOR**.



GRAPHICS



SKELETAL MESH

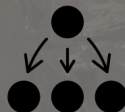




# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## SUBCLASS



## BONES



## ANIMATION



## LAYERS



## MODULAR



## DEFINITION

A **SKELETAL MESH** IS A MESH WITH MORE **FUNCTIONALITIES**. BASICALLY, **SKELETAL MESH** **INHERIT** FROM **MESHES**. JUST LIKE **MESHES**, AND EVEN MORE WITH **SKELETAL MESHES**, THEY'LL BE CREATED **OUTSIDE** THE ENGINE.

## FEATURES

THERE IS MANY **ADDITIONAL** FEATURES ON A **SKELETAL MESH**, BUT THE MOST OBVIOUS IS THE **BONES SYSTEM** AND THE ABILITY TO HAVE **SKELETAL ANIMATION**.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## BONES



### SUBCLASS



### ANIMATION



### LAYERS



### MODULAR



## DEFINITION

A **BONE** IS BASICALLY AN **INVISIBLE** THING THAT WILL REPRESENT A **PART** OF THE **MESH**. FOR COMMON CASES, LIKE CHARACTER, ETC.. THERE IS A **NAMING CONVENTION** TO FOLLOW, IN ORDER TO HAVE SOME **AUTO FUNCTIONALITIES**

## IMPLICATION

MORE TECHNICALLY SPEAKING, THERE IS AN **ASSOCIATION** BETWEEN **BONES** AND **VERTICES** OF THE **MESHES**. **VERTICES** ARE **ATTACHED** TO THE **BONES** TO FOLLOW THEM, AND THERE IS A **HIERARCHY** BETWEEN **BONES**.

## RAGDOLL

NOWADAYS, THERE IS A LOT OF GAMES THAT USES **RAGDOLL** WHEN AN **ENTITIES** DIES IN ORDER TO SHOWS THE **LOST CONTROL**. THIS IS ACHIEVE BY BASICALLY **APPLYING GRAVITY** TO EACH **BONES**.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## ANIMATION



### SUBCLASS



### BONES



### LAYERS



### MODULAR



## DEFINITION

WE'LL NOT DIVE TOO MUCH INTO ANIMATION DETAILS, BUT EVEN IF YOU CAN ANIMATE A MESH, THE REAL MAGIC COMES WHEN YOU HAVE SKELETAL MESHES AND ANIMATION THAT ARE MADE PER BONES.

## KEYFRAMES

AN ANIMATION IS BASICALLY A SET OF RULES THAT DICTATES THAT AT A CERTAIN **KEYFRAME** ON THE **ANIMATION TIMELINE**, THE BONES MUST BE IN THAT **TRANSFORM**. THERE IS THEN A **LERPING** MADE IN THE ENGINE BETWEEN KEYS

## BONES

AS STATED ABOVE, IN THE **EXTERNAL TOOLS**, THE ANIMATION WILL BE **MADE PER BONES**. WHEN YOU ARE **IMPORTING** AN ANIMATION IN ENGINE, EVERYTHING WILL BE **COMPUTED** INTO AN ASSET KNOWN AS AN **ANIMATION**.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## LAYERS



## SUBCLASS



## BONES



## ANIMATION



## MODULAR



## DEFINITION

WHEN IT COMES TO ANIMATION, YOU'LL WANT TO HAVE **CONTROLS** OVER WHICH **COLLECTION** OF **BONES** YOU WANT TO BE **ANIMATED**. IMAGINE YOU WANT TO **ALLOWS** **RUNNING** AND **ATTACKING** AT THE SAME TIME.

## UPPER/LOWER

ONE OF THE MOST KNOWN **LAYERING** IS **UPPER** AND **LOWER**. EVERY **BONES** FROM THE **HIPS** TO THE **FOOT** WILL BE IN THE **LOWER**, WHILE OTHER **BONES** WILL BE IN **UPPER**. BUT THERE IS OBVIOUSLY OTHER **LAYERING** POSSIBILITY.

## OVERRIDE

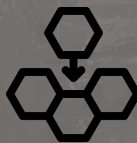
IN ANIMATION PROCESS, YOU MAY **OVERRIDE** SOME **ANIMATION** BASED ON THE **LAYERING**. IF WE TAKE GLOBALLY SPEAKING CHARACTERS, **LOWER** **LAYER** ALWAYS HAVE **PRIORITY** ON THE **OVERRIDING** PROCESS.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

MODULAR



SUBCLASS



BONES



ANIMATION



LAYERS

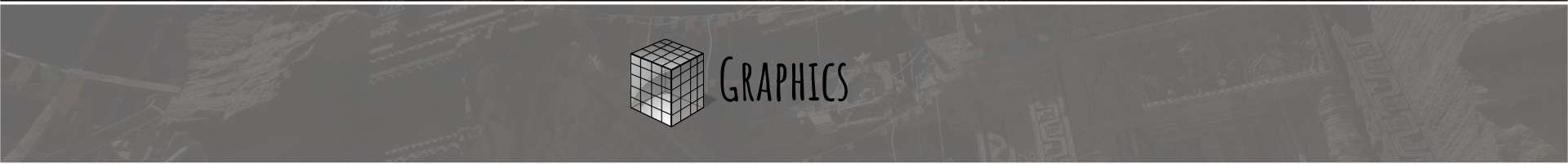


## DEFINITION

BY DEFINITION, SKELETAL MESHES ALLOW TO HAVE **MODULARITY** OVER THE CHARACTER BY HAVING **BONES** ON WHICH WE CAN ATTACH WHATEVER WE WANT.

## CUSTOMIZATION

THERE IS MULTIPLE EXAMPLE OF THAT **MODULARITY** FOR CUSTOMIZATION : CHARACTER CUSTOMIZATION, EQUIPMENT, ETC...





# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE



## DEFINITION

WE'VE ALREADY SPEAK ABOUT IT, A **MATERIAL** IS A COLLECTION OF PROPERTIES, THAT ALLOWS TO RENDER THE "PAINTING" OF A MESH. WITHOUT IT, A MESH WILL BE SIMPLY GREY.

## DRAW CALLS

MATERIAL CAN BECOME REALLY EXPENSIVE ON THE RENDER BUDGET BECAUSE EACH **MATERIAL** NEEDS 1 DRAW CALLS. IT MEANS THAT IF A CHARACTER HAS 8 MATERIALS, THERE IS 8 DRAWCALLS + 1 FOR THE MESH JUST FOR IT.

## BATCHING

THERE IS MULTIPLE TECHNIQUES IN ORDER TO COMPENSATE THE HEAVY DRAWCALLS COUNT. YOU CAN COMPACT EVERY MATERIALS INTO A SINGLE ONE, USE **BATCHING** OF MATERIALS THAT ARE EXACTLY THE SAME, ETC...



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

ALBEDO



BASIS



METALLIC



SMOOTHNESS



TEXTURES



SHADER



WAY MORE



## DEFINITION

THE ALBEDO IS ALSO KNOWN AS THE COLOR OF THE MATERIAL. BASICALLY, THE ALBEDO WILL DICTATE THAT AT A UV COORDINATE, THE PIXEL WILL BE OF A CERTAIN COLOR.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

METALLIC



BASES



ALBEDO



SMOOTHNESS



TEXTURES



SHADER



WAY MORE



## DEFINITION

DEFINES WHETHER YOUR MATERIAL BEHAVES AS A METAL OR NONMETAL. METALLIC IN MOST CASES, IS TREATED A BINARY PROPERTY, EITHER 0 OR 1.

BLACK AND WHITE MASK PASSED INTO THE METALLIC INPUT ARE OFTEN USED IN ORDER TO HAVE MORE COMPLEX METALLIC MAP



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## SMOOTHNESS



BASIS



ALBEDO



METALLIC



TEXTURES



SHADER



WAY MORE



## DEFINITION

How **SMOOTH** a MATERIAL'S SURFACE IS. IN THE MATERIAL THIS **MANIFESTS** AS HOW **SHARP** OR **BLURRY** REFLECTIONS APPEAR ON THE MATERIAL

A ROUGHNESS OF 0 (**SMOOTH**) RESULTS IN A **MIRROR** REFLECTION.

A ROUGHNESS OF 1 (**ROUGH**) RESULTS IN A **DIFFUSE** OR **MATTE** SURFACE.

IT IS MOST LIKELY THAT ARTIST WILL BE USING A **GRAYSCALE** TEXTURE TO REPRESENT ROUGHNESS AS IT IS MOST LIKELY COMPLEX ON A MESH



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## TEXTURES



### BASIS



### ALBEDO



### METALLIC



### SMOOTHNESS



### SHADER



### WAY MORE



## DEFINITION

WE'VE SPEAK OF MATERIAL PROPERTIES THAT GIVE PBR RESULT. BUT IN PRACTICE, THE PROPERTIES WILL BE FED BY TEXTURES, SPECIFYING PER UV THE METALLICNESS, SMOOTHNESS, COLOR, ETC... WITH MULTIPLES TEXTURES

## EXAMPLE

ALBEDO IS USING CLASSICAL TEXTURE, METALLICNESS USES MOST LIKELY BLACK AND WHITE TEXTURE, AND SMOOTHNESS A GRAYSCALE TEXTURE



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## SHADER



BASIS



ALBEDO



METALLIC



SMOOTHNESS



TEXTURES



WAY MORE



## DEFINITION

MATERIAL UNDERNEATH ARE SHADERS. EVEN IF IN MOST ENGINE, THERE IS NOW SOME GRAPH BASED CODING FOR THE MATERIALS, THIS IS SHADER CODE THAT IS RUNNING.

## COMPLEXITY

SHADER ARE EVERYWHERE GIVEN THERE IS MATERIALS EVERYWHERE. COMPLEXITY OF IT IS IMPORTANT TO NOT DESTROY FRAME BUDGET. EVEN IF IT RUNS ON GPU, YOU MUST REDUCE AS MUCH AS POSSIBLE THE COMPUTATION.

## USAGE

EVEN IT IS EXPOSED DIFFERENTLY FROM ENGINE TO ENGINE, GLOBALLY SPEAKING, THERE IS SPECIFIC SHADERS BASED ON USAGE. FOR EXAMPLE, IF YOU WANT A SHADER WITH TRANSPARENCY, YOU'LL USE A DIFFERENT SHADER THAN CLASSICAL



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

WAY MORE



BASIS



ALBEDO



METALLIC



SMOOTHNESS



TEXTURES



SHADER



## NORMAL MAP

NORMAL MAP IS OMNIPRESENT IN GAME DEVELOPMENT NOWADAYS. BASICALLY, IT IS A BLUESCALE TEXTURE THAT FAKES DETAILS AND SCULPTING BY FEEDING THE NORMALS OF A VERTICES, TO BE SHADED WITH LIGHTING

## DISPLACEMENT MAP

DISPLACEMENT MAP IS KIND OF A NORMAL MAP WITH AN HEIGHTMAP TEXTURE WITH A REFINEMENT. WHILE NORMAL MAP FAKE DETAILS, THE DISPLACEMENT MAP ACTUALLY MOVES THE VERTICES IN ORDER TO GIVE DETAILS.

## HEIGHT MAP

HEIGHT MAP IS A TEXTURE A BIT PARTICULAR WHICH REPRESENT ELEVATION IN LEVEL. IT CAN BE USED IN DIFFERENT SITUATION BUT AS THE NAME STATES, IT IS MOSTLY A GRAYSCALE TEXTURE.



GRAPHICS



POST PROCESSING





# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

BASIS



CAMERA



PERFORMANCES



## DEFINITION

POSTPROCESSING IS A TECHNIQUE USED IN EVERY GAMES NOWADAYS THAT ALLOWS TO ALTER THE FINAL IMAGE WHICH IS RENDERED ON SCREEN.

## FILTERS

WE'LL NOT GO INTO DETAILS FOR FILTERS, BUT POSTPROCESSING OFFERS A VARIETIES OF TOOL LIKE COLOR GRADING, AMBIENT OCCLUSION, BLOOM, DEPTH OF FIELDS, ETC... ALL OF THIS ARE CONFIGURABLE.

## VOLUME

IN MOST ENGINE, YOU HAVE THE POSSIBILITIES TO DEFINE VOLUME IN WHICH POSTPROCESSING IS PRESENT. IT ALLOWS FOR EXAMPLE TO HAVE A GRAYSCALE VIEW IF YOU ENTER IN A SPECIFIC AREA.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

CAMERA



BASIS



PERFORMANCES



## DEFINITION

A POST PROCESSING IS ACTUALLY AFFECTING THE **CAMERA**, THE CAMERA IS RESPONSIBLE FOR BEING THE **VIEWPORT** TO BE **RENDERED** ON SCREEN. POST PROCESSING AS SO, IS **AFFECTING** DIRECTLY THE RENDER OF THE CAMERA.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## PERFORMANCES



## BASIS



## CAMERA



## DEFINITION

POST PROCESSING IS A COSTLY OPERATION. THE MORE FILTER YOU HAVE, THE HEAVIER THE COMPUTATION WILL BE. SOME ENGINE COMES WITH A LOT OF POST PROCESSING OUT OF THE BOX, LIKE UNREAL, WHILE OTHER OFFERS MORE FREEDOM

## OPTIMIZATION

EVEN IF POST PROCESSING IS AN HEAVY OPERATION IF USING A LOT OF THINGS IT IS IMPORTANT TO NOTICE IN SOME CASES, IT CAN ENHANCED THE VISUAL QUALITY AND ALSO BE MORE OPTIMIZED

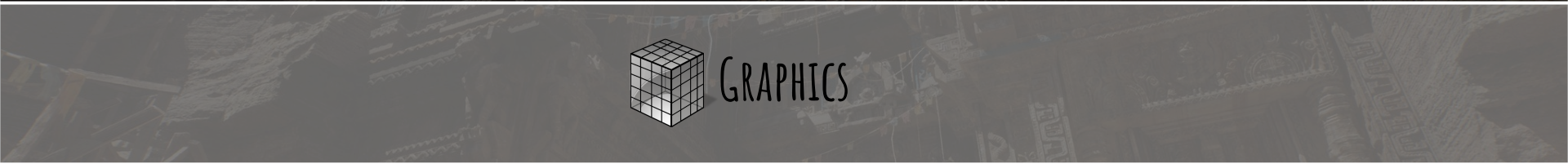
## GRANULARITY

FROM A SIMPLE WARMER COLOR TO A COMPLETELY DIFFERENT LOOK, POST PROCESSING ALLOWS TO HAVE VARIOUS GRANULARITY OVER THE GLOBAL LOOKS OF YOUR RENDERING.





GRAPHICS



LIGHTING





# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## CONCEPT



### AMBIENT



### DIRECTIONAL



### POINT-LIGHT



### SPOTLIGHT



### SHADOWS



## DEFINITION

LIGHTING IS A KEY FEATURE IN ANY GAME ENGINE. LIGHTS BRINGS LUMINOSITY AND VISIBILITY INTO WORLDS, BRINGING ATMOSPHERE, AMBIANCE AND MUCH MORE. IT CAN ALSO BE HEAVY ON PERFORMANCES.

## STATIC VS DYNAMIC

AN IMPORTANT ASPECT WHEN IT COMES TO LIGHTING, IS STATIC VS DYNAMIC LIGHTS. STATIC LIGHTS WILL BE BAKED INTO LIGHTMAPS, CREATING A STATIC ENVIRONMENT. DYNAMICS OTHERWISE BRINGS REALTIME LIGHTINGS AND LIFE.

## RAYTRACING

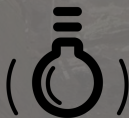
WHEN IT COMES TO COMPUTING STATIC LIGHTING, OR HAVING REALTIME LIGHTING WITH SHADOWS, IT IS DRIVEN BY RAYTRACING, AND IS DIRECTLY USING PHYSICS. IT COMES WITH VARIOUS PARAMETERS LIKE BOUNCING COUNT, ETC..



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## AMBIENT



### CONCEPT



### DIRECTIONAL



### POINT-LIGHT



### SPOTLIGHT



### SHADOWS



## DEFINITION

AMBIENT LIGHT ALSO KNOWN AS DIFFUSE ENVIRONMENTAL LIGHT. YOU'LL MOST LIKELY NOT BE ADDING MORE THAN 1 AMBIENT LIGHT AS THE NAME SUGGESTS, BECAUSE IT GIVES THE ATMOSPHERE OF YOUR WORLD.

## MINIMUM

IT IS AN IMPORTANT LIGHT IN KIND OF EVERY GAMES, EVEN IF YOU DON'T WANT TO HAVE A SPECIFIC AMBIANCE, BECAUSE IT COULD ALLOWS TO MAKE THE WORLD BRIGHTER WITHOUT TOUCHING INDIVIDUAL LIGHTS

## DIRECTIONLESS

BY NATURE A DIFFUSE LIGHT IS DIRECTIONLESS, IT DOESN'T EMIT FROM A SOURCE NOR HAVE A DIRECTION. IT IS JUST A LIGHT EVERYWHERE WITHOUT ANY NOTION OF DISTANCE.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## DIRECTIONAL



### CONCEPT



### AMBIENT



### POINT-LIGHT



### SPOTLIGHT



### SHADOWS



## DEFINITION

DIRECTION LIGHT CAN BE THINK AS DIFFUSE LIGHT WITH A DIRECTION. THEY CAN BE CONSIDERED LIKE IN FAR FAR DISTANCE, OUTSIDE OF THE WORLD.

## DIRECTION

BECAUSE OF THE DIRECTION, THE DIRECTIONAL LIGHT WILL HAVE SHADOWS OBVIOUSLY, AND BECAUSE THERE IS NO CONCEPT OF DISTANCE LIKE THE DIFFUSE, IT IS SUPPOSED TO BE A GLOBAL LIGHT FOR THE SCENES

## EXAMPLE

SUN, MOON, ETC...



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## POINT-LIGHT



CONCEPT



AMBIENT



DIRECTIONAL



SPOTLIGHT



SHADOWS



## DEFINITION

A POINT LIGHT IS THE MOST COMMON LIGHT, AS IT IS THE ONE EMITTING LIGHT IN ALL DIRECTION AROUND IT. THE MAIN LIGHTING OF YOUR SCENE WILL BE DONE WITH POINT LIGHT

## AROUND

BEING AN **EMITTER** ALL **AROUND** THE CENTER POINT, IT IMPLIES THAT WE HAVE A **STRENGTH** OF **EMISSION** FOR THIS LIGHT. ALSO, THE LIGHT IS **SPREAD** ALL AROUND WITH AN **EQUAL POWER**

## GAMEPLAY

THERE IS VARIOUS EXAMPLE ON GAMEPLAY SIDE TO USE A **POINT LIGHT** : A CANDLE, A LANTERN, A TORCH, ETC...  
BASICALLY, MOST **NATURAL LIGHTS** ARE POINT LIGHT.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## SPOTLIGHT



## CONCEPT



## AMBIENT



## DIRECTIONAL



## POINT-LIGHT



## SHADOWS



## DEFINITION

A SPOTLIGHT ON THE OTHER END IS NOT EMITTING IN EVERY DIRECTION BUT IN A GIVEN DIRECTION, WITH A GIVEN ANGLE. SPOTLIGHT ARE BASICALLY USING CONE SHAPE TO DIFFUSE.

## INNER & OUTER

IN MOST ENGINE, THERE IS AN INNER CONE AND AN OUTER CONE, YOU'LL SEE THAT DIFFERENCE BECAUSE ON THE INNER CONE, THE LIGHT IS STILL AT FULL POWER, AND WILL GRADUALLY DECREASE TO 0 ON THE END OF OUTER



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## SHADOWS



CONCEPT



AMBIENT



DIRECTIONAL



POINT-LIGHT



SPOTLIGHT



## DEFINITION

SHADOWS ARE AN IMPORTANT ASPECT IN LIGHTING AS IT BRINGS REALITY TO AN AMBIANCE. IT CAN ALSO BE USED FOR GAMEPLAY PURPOSE, LIKE KNOWING AN ENEMY IS COMING FOR EXAMPLE, BUT IT IS LESS COMMON.

## PER LIGHT

IT IS IMPORTANT TO NOTICE THAT SHADOWING CAN BE ALLOWED PER LIGHT. THAT MUST BE USED TO YOUR ADVANTAGES IN ORDER TO COMBINE PERFORMANCE AND REALISM.

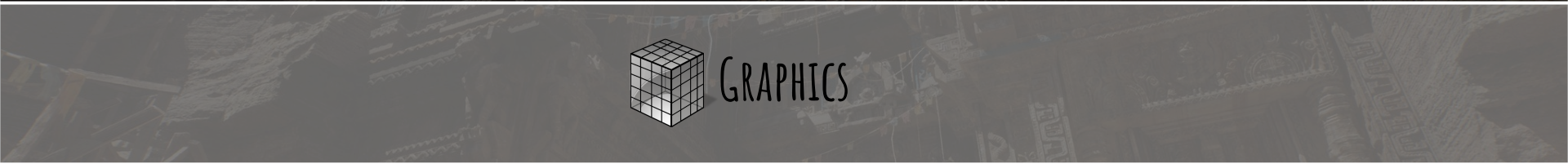
## TECHNIQUES

SHADOW MAPS, VIRTUAL MAPS, RAYTRACED, DISTANCE FIELD, ETC...

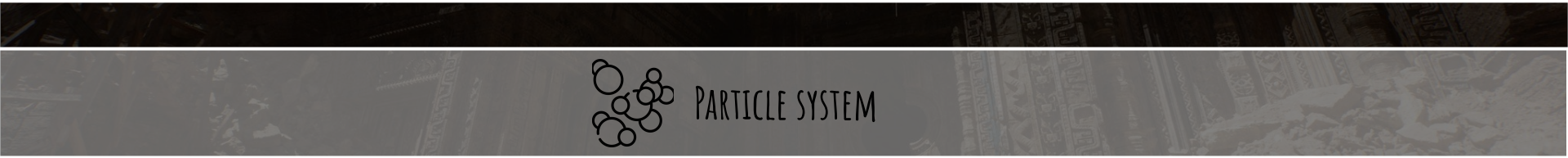




GRAPHICS



PARTICLE SYSTEM





# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## PRINCIPLE



## PARTICLE



## EMISSION



## SIMULATION



## RENDERING



## PARAMETERS



## PHYSICS



## DEFINITION

A PARTICLE SYSTEM IS USED FOR CREATING IMPRESSIVE VISUAL EFFECT, IT CAN GOES FROM A SIMPLE BLOOD EXPULSE, TO A COMPLEX AREA SPELL INDICATOR.

## COMPLEXITY

PARTICLE SYSTEM LIKE EXPLAINED BRIEFLY IN THE FIRST LESSON IS A MIX OF CPU AND GPU USAGE, AND IS A REALLY COMPLEX PIECE OF SHADER CODE AND MATHEMATICS.

## PERFORMANCE

PERFORMANCE-WISE, PARTICLE SYSTEM ARE **EXPENSIVE**, YOU MUST BE REALLY **CAREFUL** WITH HOW IMPACTFUL IT WILL BE ON THE **GAME PERFORMANCES**.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## PARTICLE



### PRINCIPLE



### EMISSION



### SIMULATION



### RENDERING



### PARAMETERS



### PHYSICS



## DEFINITION

A PARTICLE SYSTEM IS OBVIOUSLY COMPOSED OF PARTICLE. THERE IS POTENTIALLY THOUSANDS OR MILLIONS OF PARTICLE, EACH ONE BEING COMPUTED KIND OF INDEPENDENTLY.

## LOCATION

THE MOST IMPORTANT PROPERTY OF A PARTICLE IS THE LOCATION. THERE IS VARIOUS PROPERTIES AND SYSTEM THAT WILL MOVE THE LOCATION OF THE PARTICLE, BUT THE LOCATION WILL DISPLAY THE PARTICLE AT THE LOCATION.

## PROPERTIES

BEING INDEPENDENT, A PARTICLE HAS VARIOUS PROPERTIES LIKE VELOCITY, COLOR, A LIFETIME, SIZE, ETC...



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## EMISSION



### PRINCIPLE



### PARTICLE



### SIMULATION



### RENDERING



### PARAMETERS



### PHYSICS



## DEFINITION

THE FIRST STEP OF A PARTICLE SYSTEM IS TO ACTUALLY EMIT THE PARTICLES. FOR THAT, IN EVERY ENGINE, THERE IS MODULE EXPOSING VARIOUS PARAMETERS WHICH WILL DICTATE HOW THE PARTICLE ARE SPAWNED (COUNT, ETC...)

## PERFORMANCES

THE EMISSION IS THE MOST IMPORTANT ASPECT ON THE PERFORMANCE SIDE, BECAUSE THE MORE PARTICLE YOU ARE SPAWNING, THE HEAVIER THE PARTICLE SYSTEM WILL BE.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## SIMULATION



### PRINCIPLE



### PARTICLE



### EMISSION



### RENDERING



### PARAMETERS



### PHYSICS



## DEFINITION

AFTER BEING SPAWN, AND UNTIL THE PARTICLE IS DESTROYED, YOU'LL BE SIMULATING ON A FRAME BASIS THE PARTICLE. OBVIOUSLY, THIS SIMULATION WILL BE ON GPU FOR PERFORMANCES REASONS.

## UPDATE

JUST LIKE **MONOBEHAVIOR**, YOU'LL HAVE AN UPDATE METHOD THAT IS RUNNING THE SIMULATION. YOU COULD CREATE MODULES TO BE RUNNED ON THE UPDATE SIMULATION FRAME.

## FORCES

THERE IS OBVIOUSLY **FORCES** THAT ARE APPLIED EACH FRAME. THE **FORCES** HAS TO BE DEFINED BY YOURSELF, BUT THERE IS ONE EXISTING IF THE MODULE IS ACTIVATED : **GRAVITY**



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## RENDERING



### PRINCIPLE



### PARTICLE



### EMISSION



### SIMULATION



### PARAMETERS



### PHYSICS



## DEFINITION

AS EVERYTHING PRESENT ON SCREEN, PARTICLES NEEDS TO BE RENDERED, AND AS SO, IT IS USING THE RENDERING PIPELINE. BUT THERE IS NO PRINCIPLES OF POLYCOUNT ETC... ON A PARTICLE, SO RENDERING IS QUITE DIFFERENT.

## VISIBILITY

ABOUT THE RENDERING AND VISIBILITY, PARTICLES ARE RENDERED AS BILLBOARDED WITH TEXTURED BILLBOARDED QUAD, IT MEANS THAT THE PARTICLE IS ALWAYS FACING THE CAMERA.

## 2D VS 3D

BY DEFINITION, A PARTICLE IS A 2D RENDERING, YOU'LL BE GIVING TEXTURE AS INPUT FOR THE ASPECT OF THE PARTICLE. BUT THERE IS POSSIBILITY TO PROVIDE A MESH AND MAKE THE PARTICLE BECOMES 3D REPRESENTATION.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## PARAMETERS



### PRINCIPLE



### PARTICLE



### EMISSION



### SIMULATION



### RENDERING



### PHYSICS



## DEFINITION

WHEN YOU HAVE A PARTICLE SYSTEM, YOU'LL BE CONFIGURING IT THROUGH **PARAMETERS**. EVEN IF YOU CREATE NEW MODULES, YOU'LL BE **EXPOSING PARAMETERS** SO VISUAL ARTIST CAN CREATE THE EFFECTS.

## PLENTY

THERE IS A LOT OF PARAMETERS, IT CAN BE **OVERWHELMING** AT FIRST, BUT IT IS NEEDED IN ORDER TO CREATE ALL THE IDEA OF **VFX** THAT WILL BE NEEDED FOR A GAME.



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## PHYSICS



### PRINCIPLE



### PARTICLE



### EMISSION



### SIMULATION



### RENDERING



### PARAMETERS



## DEFINITION

YOU HAVE PROBABLY UNDERSTAND IT WITH THE SIMULATION PHASE, BUT **PHYSICS** IS INVOLVED. THAT'S WHY THIS IS A **COMPLEX TOPIC**, AND SOME ENGINE DECIDED TO REWRITE AN INTERNAL PHYSIC ENGINE.

## EVENTS

YOU CAN CONFIGURE **EVENTS** ON PARTICLE. FOR EXAMPLE, IF YOU WANT **COLLISIONS** EVENT WHEN PARTICLE ARE COLLIDING, **EVENTS** WHEN **PARTICLES** DIES, ETC...

## CPU VS GPU

DECIDING BETWEEN USING **CPU** OR **GPU** WILL BE BASED ON MULTIPLE **FACTORS** : ARE **COLLISIONS** / **EVENTS** NEEDED, HOW IS THE **FRAME BUDGET**, IS THERE A LOT OF **PARTICLES**, ETC...





GRAPHICS



RENDERING PIPELINE





# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## PRINCIPLE



## COMPROMISE



## DEFINITION

WE'LL NOT DIVE INTO DETAILS AS IT IS NOT THE PURPOSE OF THE LESSON. YOU SIMPLY MUST BE AWARE THAT A RENDERING PIPELINE IS SOMETHING TO BE CONFIGURED TO YOUR NEEDS.

## PIPELINE

RENDERING FOLLOWS A PIPELINE, IT MEANS A SET OF INSTRUCTIONS, IN A SPECIFIC ORDER, THE OUTPUT OF A STEP BEING MOSTLY USED AN INPUT OF THE NEXT STEP.

## PLATFORM

THE RENDERING PIPELINE ENSURE TO BE PLATFORM INDEPENDENT, AS MUCH AS POSSIBLE, AND ARE MOSTLY BASED ON SOME LOW LEVEL API LIKE DIRECTX, OpenGL, Vulkan, etc...



# GRAPHICS

1. MESH
2. SKELETAL MESH
3. MATERIAL
4. POST PROCESSING
5. LIGHTING
6. PARTICLE SYSTEM
7. RENDERING PIPELINE

## COMPROMISE



## PRINCIPLE



## DEFINITION

BASED ON PLATFORM, GAME'S NEEDS, ETC... YOU'LL NEED TO DO COMPROMISE IN ORDER TO HAVE VISUAL FIDELITY, PERFORMANCES.

## PARAMETERS


IN ALL ENGINE, YOU'LL BE ABLE TO CONFIGURE THE RENDERING PIPELINE WITH A SET OF PARAMETERS. IN MOST CASES, THERE IS A LOT OF PARAMETERS EXPOSED, LIKE SHADOWS, LIGHTING, OCCLUSION, ETC...



The background image shows the ruins of an ancient Egyptian temple, likely the Temple of Isis at Philae. The temple is carved into a dark, rocky cliff face. It features several large, ornate doorways with hieroglyphs and papyrus motifs. The architecture is partially obscured by debris and rubble in the foreground. A semi-transparent horizontal band across the middle of the image contains the text and a logo.

 LIVE DEMONSTRATION



The background image shows ancient stone ruins, possibly Mayan or Aztec, with intricate carvings and hieroglyphs. The scene is dimly lit, with a central horizontal band of light gray containing the text. The ruins are built into a hillside, with some structures featuring large doorways and decorative friezes. The overall tone is mysterious and historical.

? QUESTIONS ?