



Unity: Lighting

By Chris Bernstein, Shane Ferguson, and Chandaravuth Non



Introduction

- Why should we care?
- Different types of light in Unity and their applications
- Reflections tutorial



Global Illumination

- Technique of simulating realistic lighting events as they play across a scene and bounce off objects
- Simulating lighting is very complex and involves a lot of mathematics and processing power
- Most desirable way of handling lighting is through computing and rendering them before the scene, avoiding the added complex of having them being processed as the scene plays



Lighting Techniques

- Realtime Lighting
 - Most basic way to implement lighting in Unity
 - Realtime lighting emits direct light to the scene on every update frame
 - Objects that move through the scene directly interact with realtime lighting, creating realistic lighting visuals
 - Direction, Spot, and Point Lights



Lighting Techniques

- Baked GI Lighting
 - Where the term “baking” comes from
 - Lighting created using this technique are static, can’t be manipulated
 - Significantly improved performance over realtime lighting



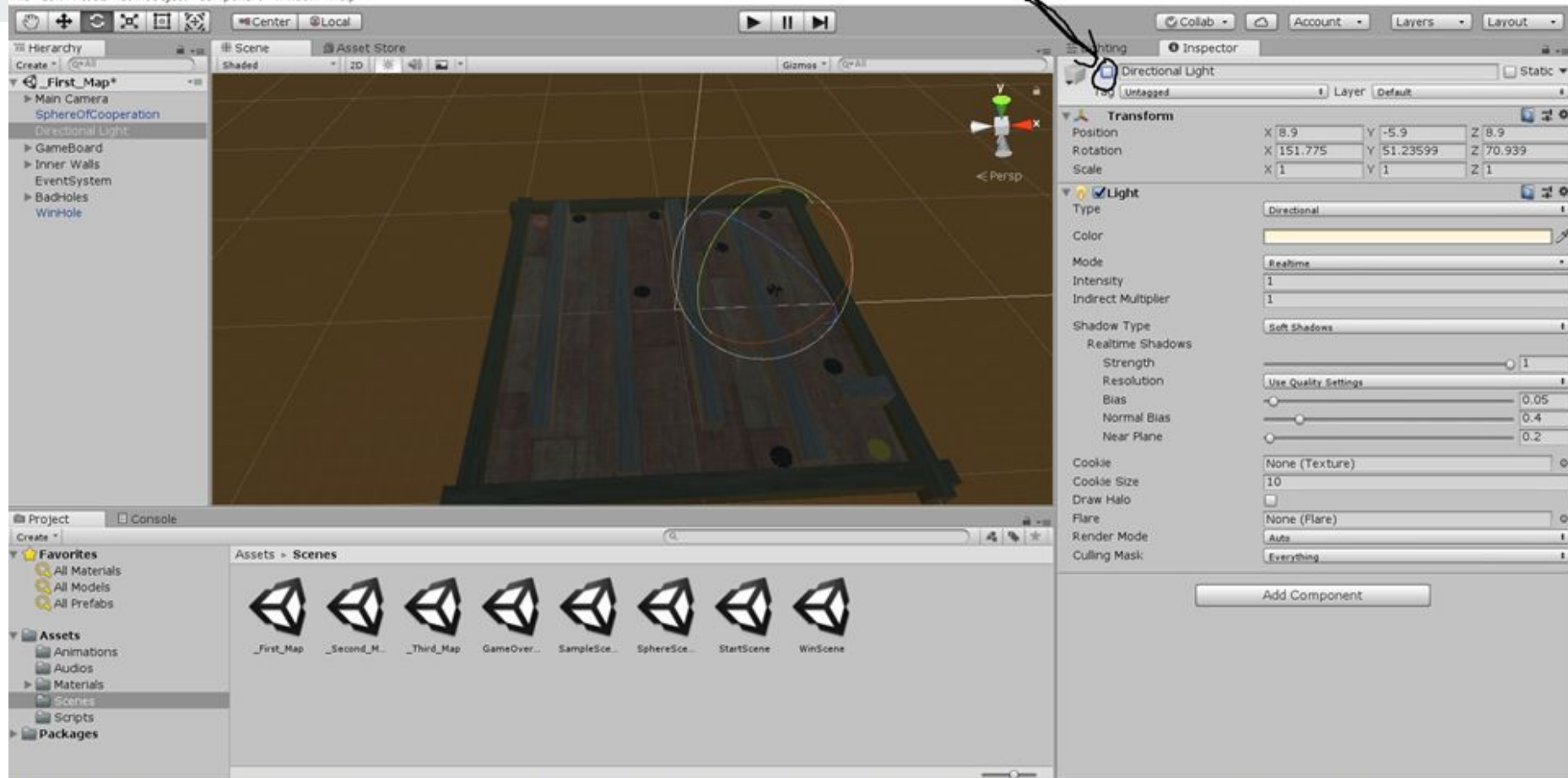
Lighting Techniques

- Precomputed Realtime GI Lighting
 - Cross between realtime and baked
 - Good way to simulate live lighting effects without the heavy costs of using realtime techniques
 - A great example would be a time of day system



Directional lights

- they are emit parallel light rays in a single direction but the light reaches out into infinity
- They can be used in outdoor scenes for the sun or moon light
- Can rotate the light in any direction
- This light is perfect when needing to illuminate a city or large open spaces
- Run on real time
- Least expensive realtime lighting to the graphics processor
- Not work on dynamic object





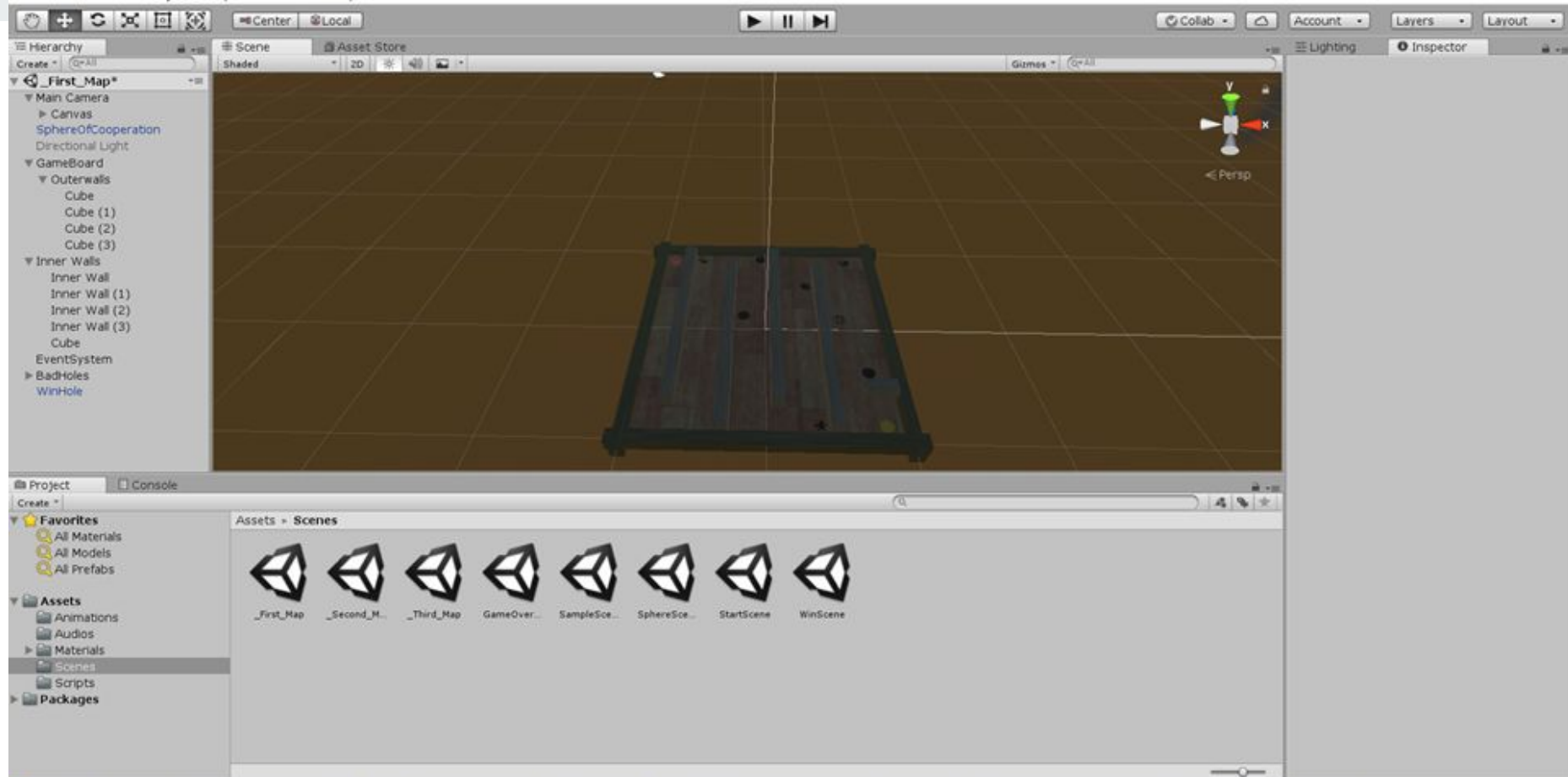
The screenshot displays the Unity 2018.2.11f1 Personal interface. The main view is a 3D scene titled "_First_Map" showing a perspective view of a game environment with a grid floor and several rectangular blocks. A directional light is positioned in the scene, casting a yellow cone of light. The Hierarchy panel on the left shows the scene's structure, including the Main Camera, SphereOfCooperation, Directional Light, GameBoard, Inner Walls, EventSystem, BadHoles, and WinHole. The Inspector panel on the right shows the properties of the selected Directional Light, including its Transform (Position, Rotation, Scale) and Light settings (Type, Color, Mode, Intensity, Indirect Multiplier, Shadow Type, Realtime Shadows, Strength, Resolution, Bias, Normal Bias, Near Plane, Cookie, Cookie Size, Draw Halo, Flare, Render Mode, Culling Mask). The Console panel at the bottom shows a message: "You are trying to create a MonoBehaviour using the 'new' keyword. This is not allowed. MonoBehaviours can only be added using AddComponent(). Alternatively, your script can inherit from ScriptableObject or no base class at all".

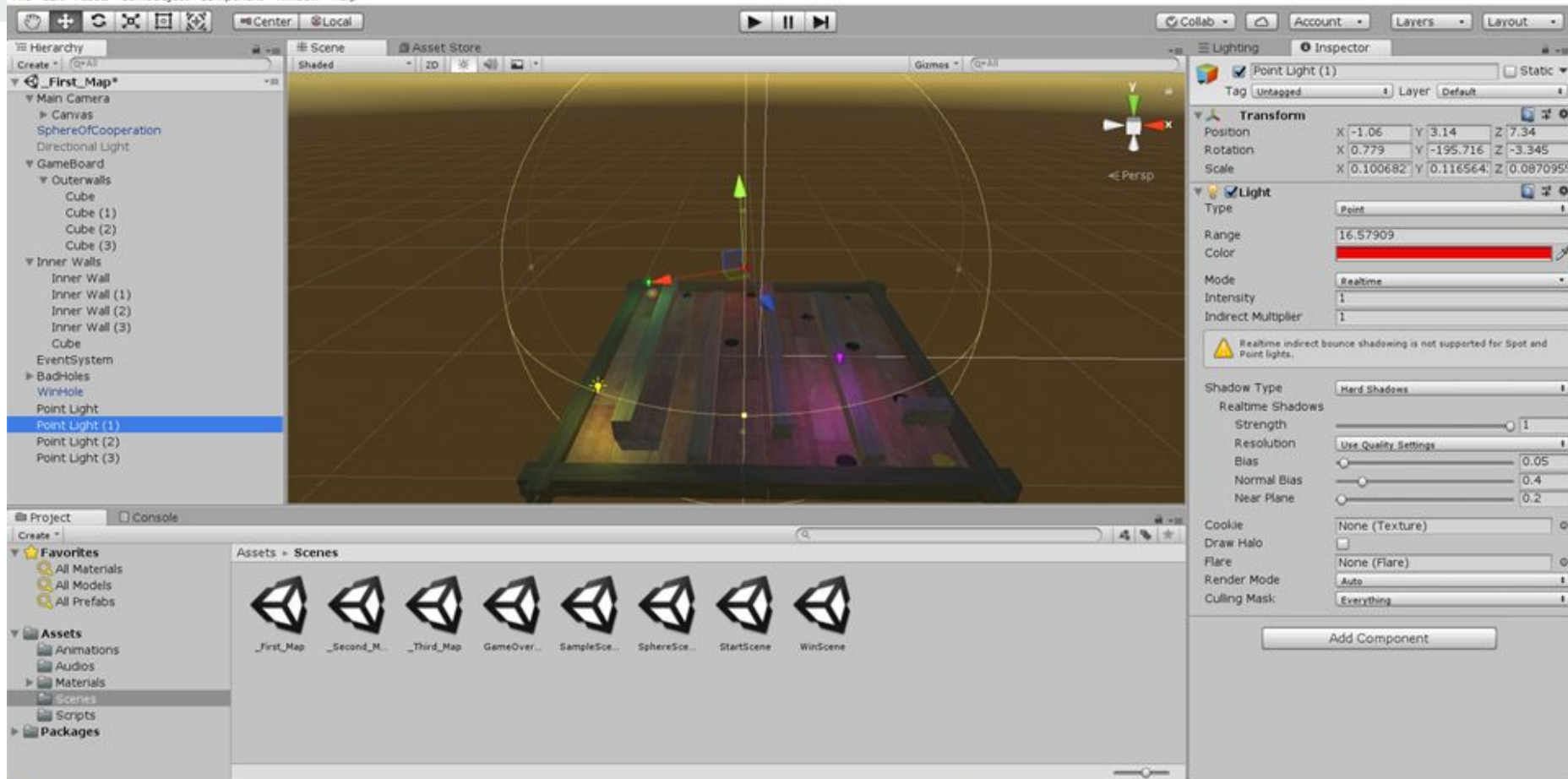


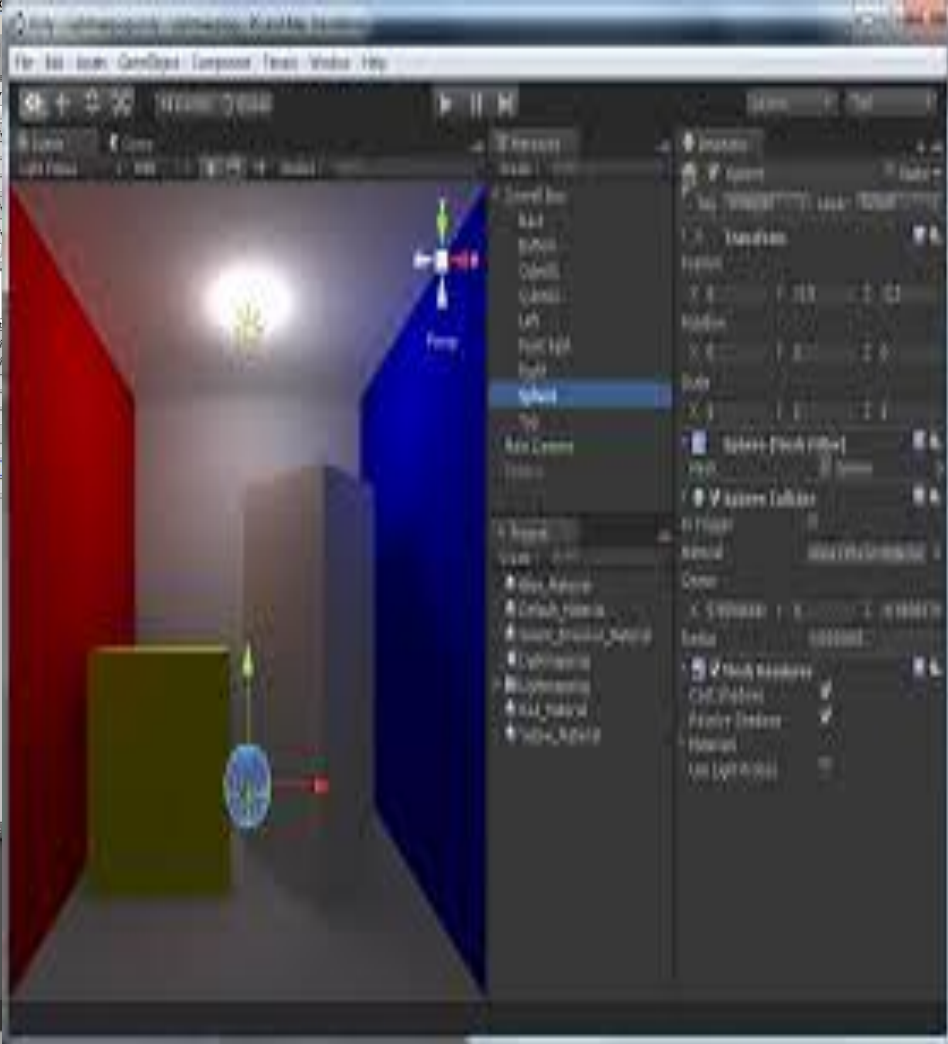
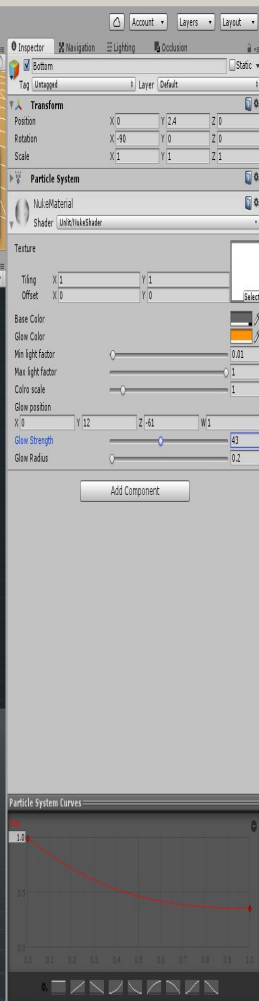
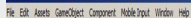
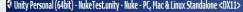


Point Lights

- Point light simulates rays shining out from the point in all directions
- Objects closer to the light will be brighter, and objects further away will be darker
- Very useful for simulating lamps, candles and other local sources of light in a scene such as spark or explosion
- Point lights are the most common light types in game
- Run on real time
- Average cost in realtime lighting for graphic processor but point lights shadow are the most expensive
- Not work on dynamic object



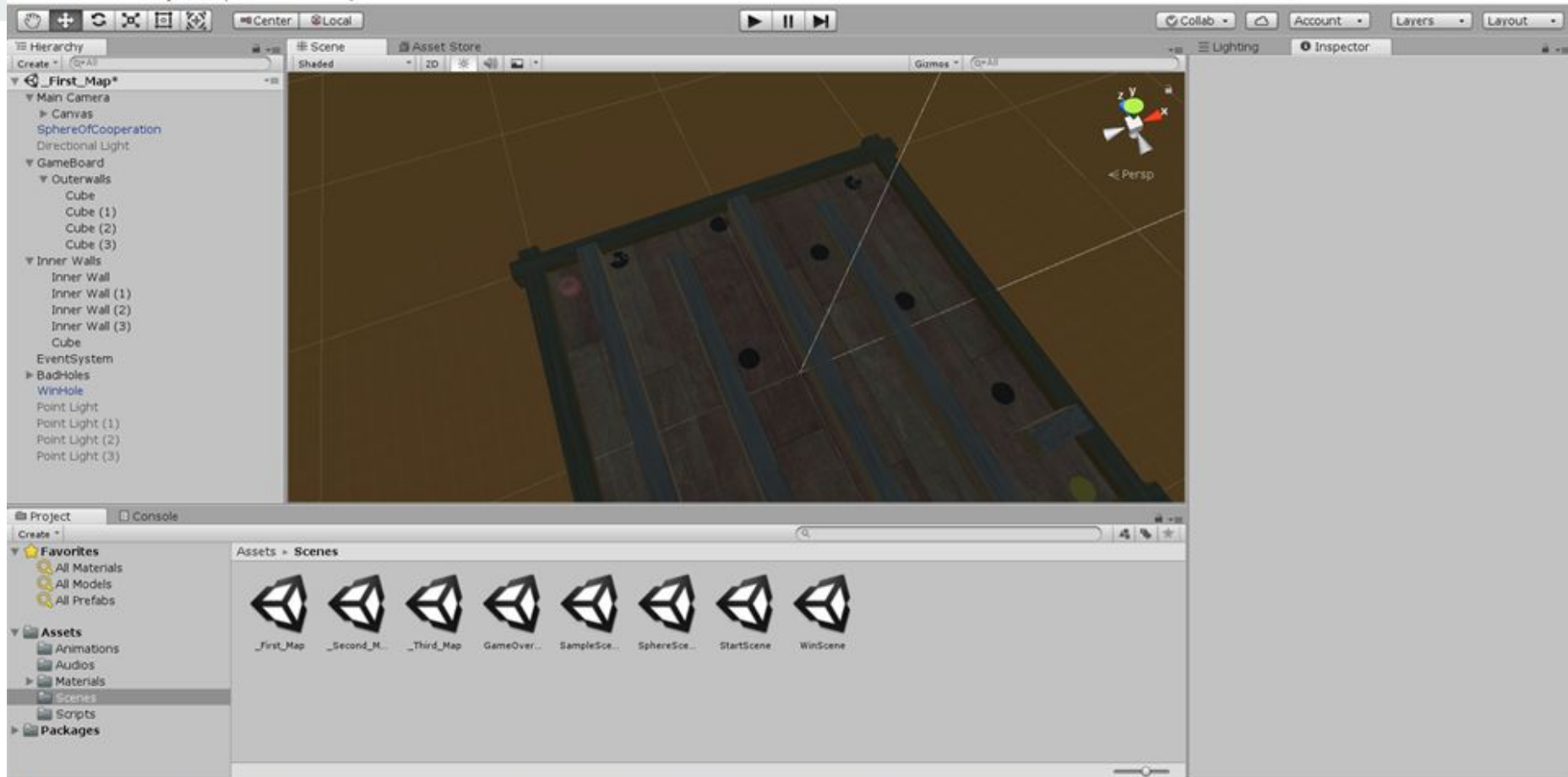


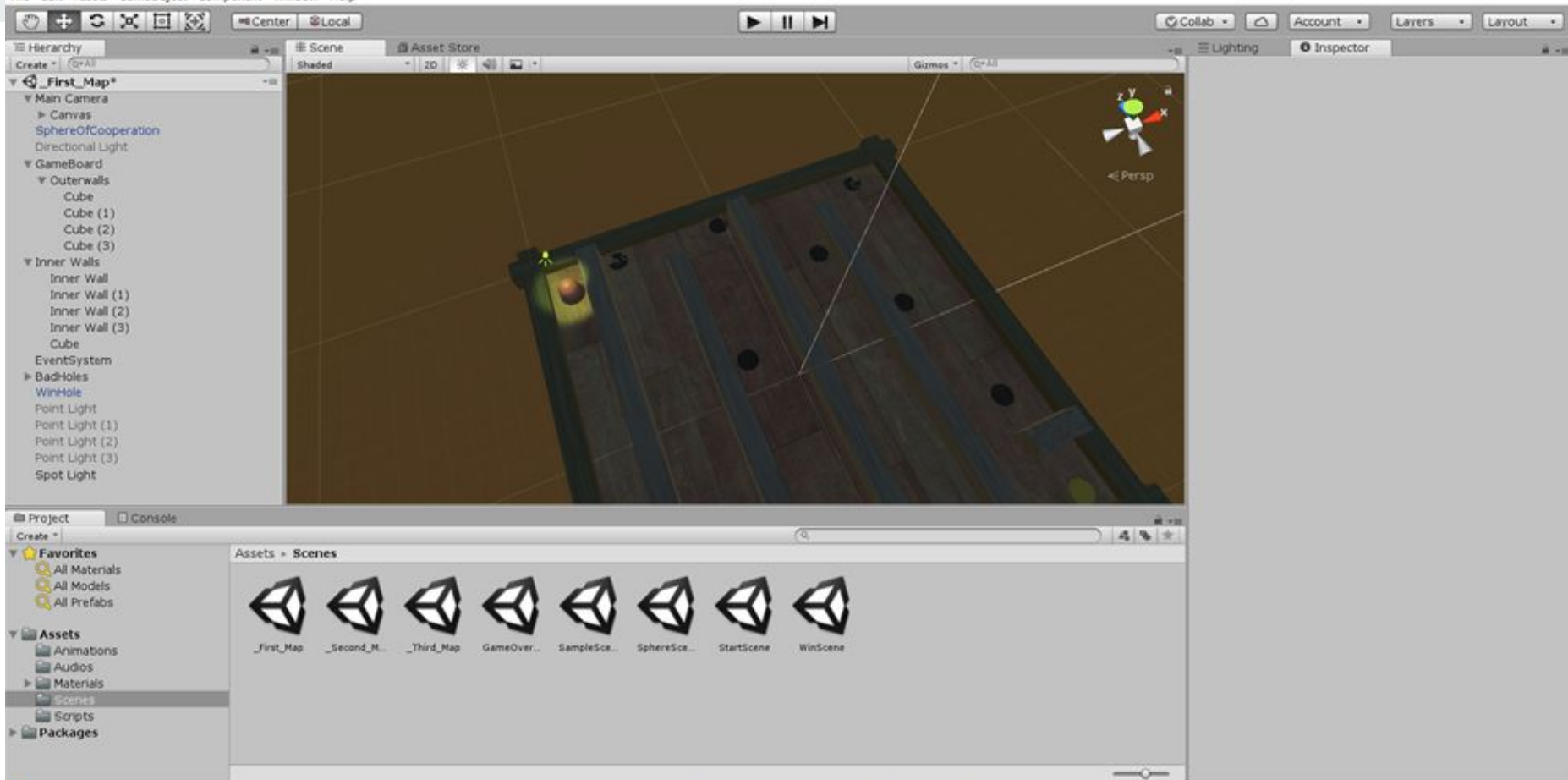


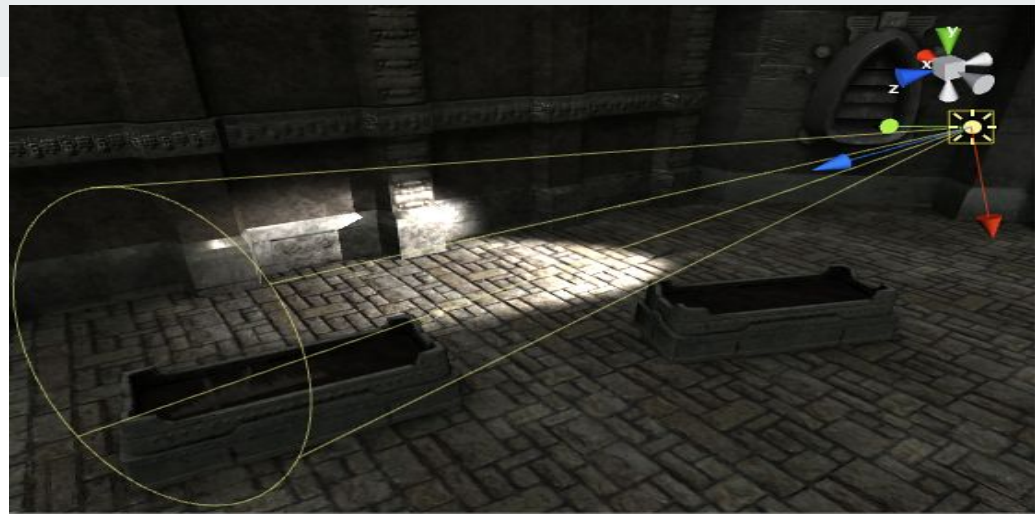
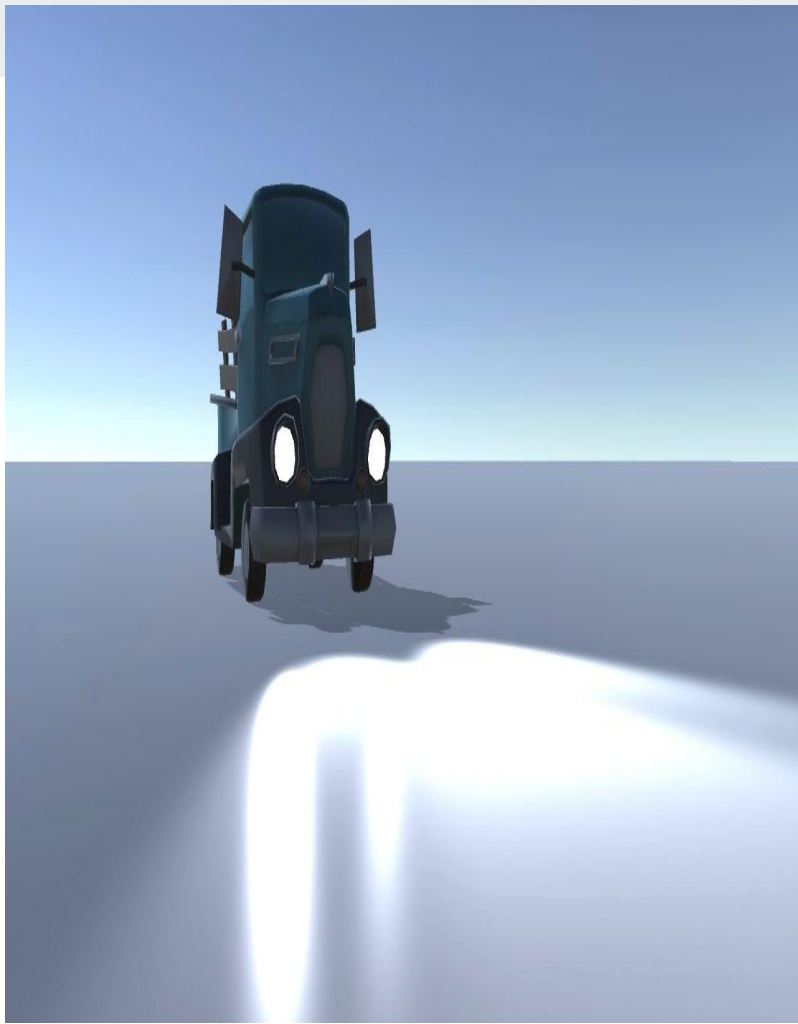


Spot Lights

- Spotlight shine from a point in a direction and only illuminate objects within a cone
- Objects closer to the spotlight will be brighter, and depending on the how wide the cone is the light will either be softer or harder
- Used for artificial light sources such as flashlights, car headlights or lamp posts
- Direction can controlled from a script or animation
- Moving spot light will illuminate just a small area of the scene and create dramatic lighting effects
- Run on real time
- Most expensive realtime lighting to the graphics processor
- Not work on dynamic object



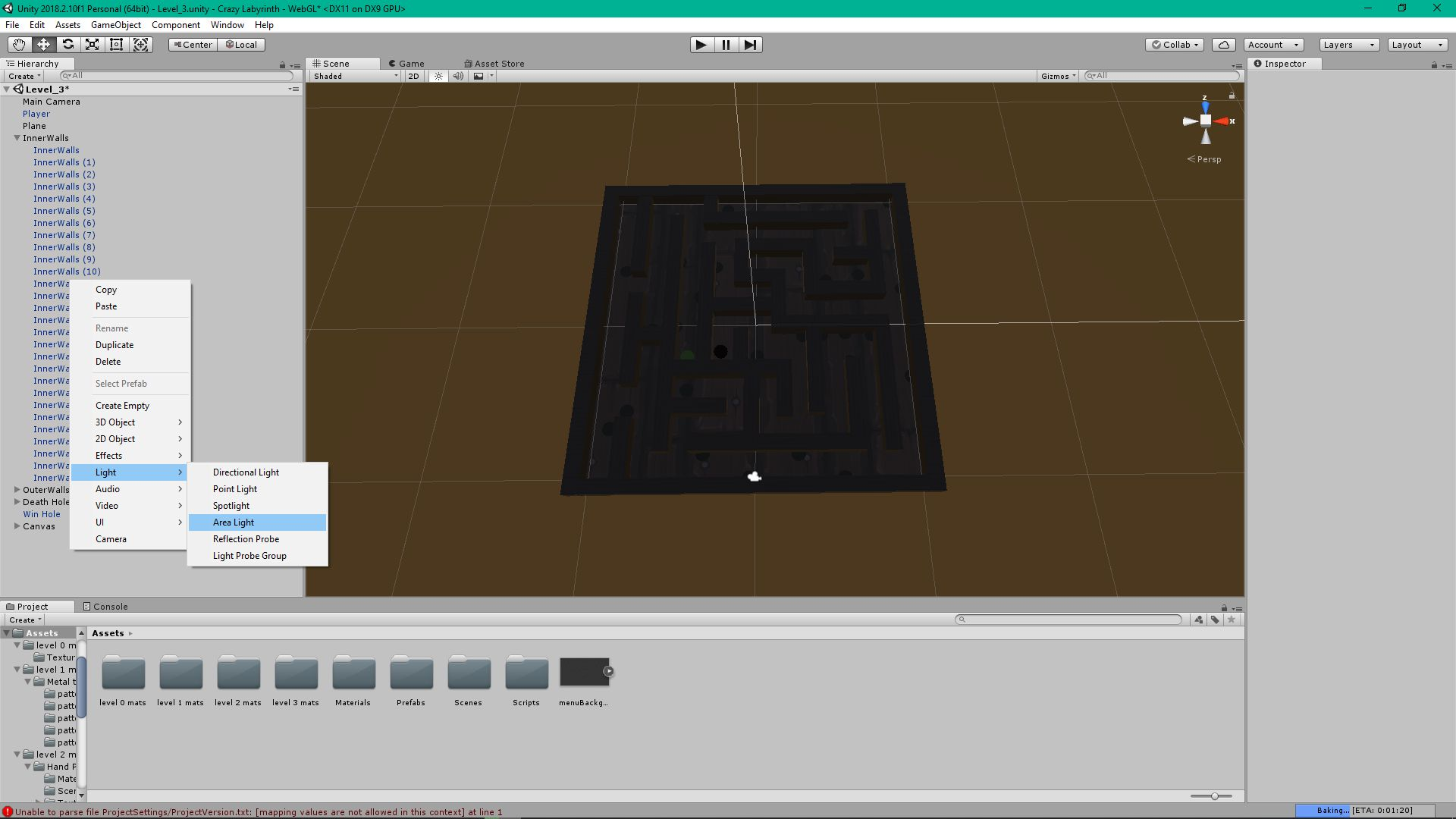


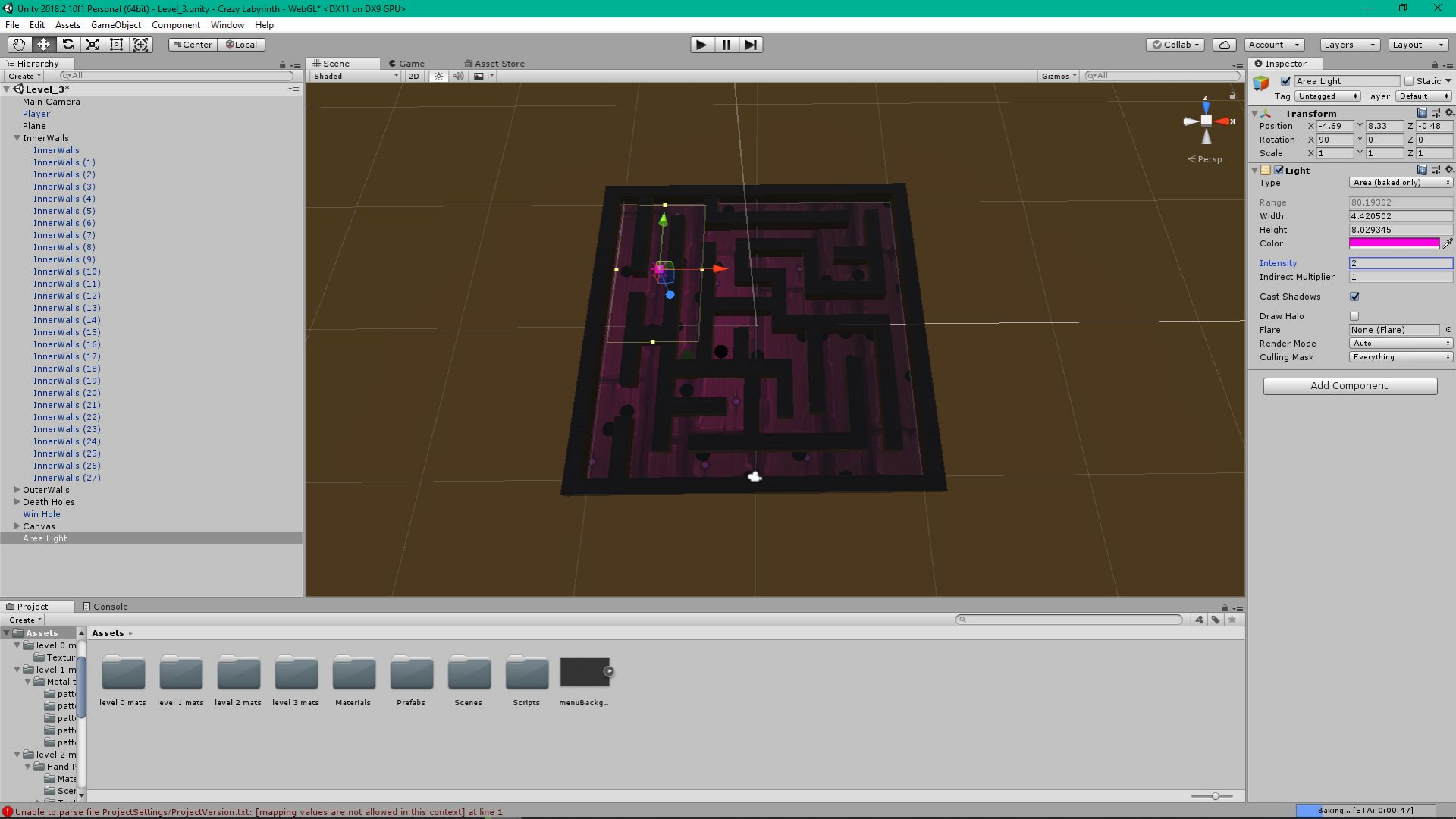


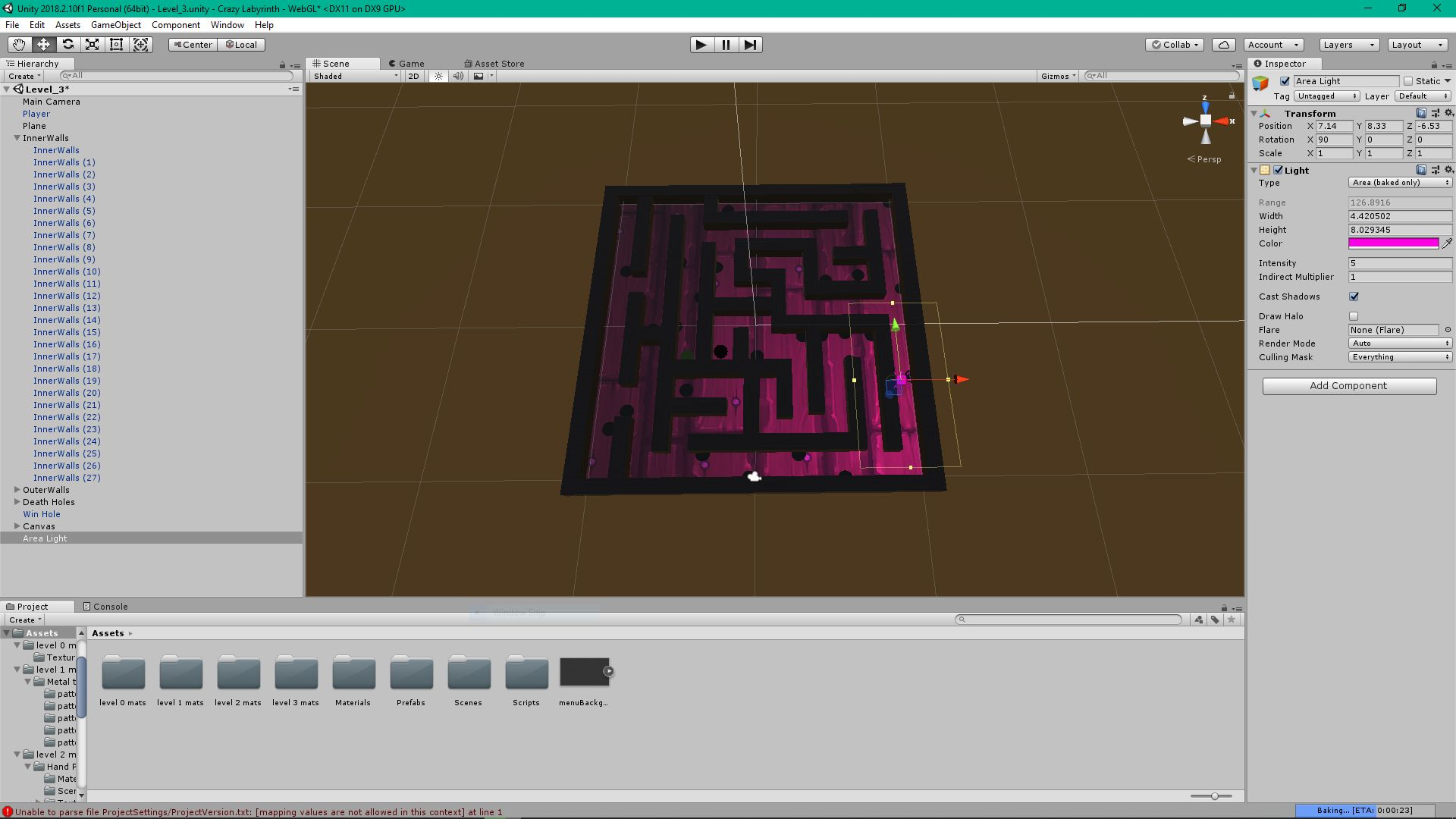


Area Lighting

- An area in the shape of a rectangle is selected
- Light is emitted uniformly across the entire shape, from one side
- Brightness fades as the distance from the source increases
- Considered to be CPU heavy
- Example uses include creating a realistic effect of the interior lighting of a house



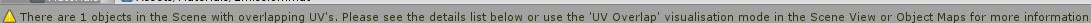


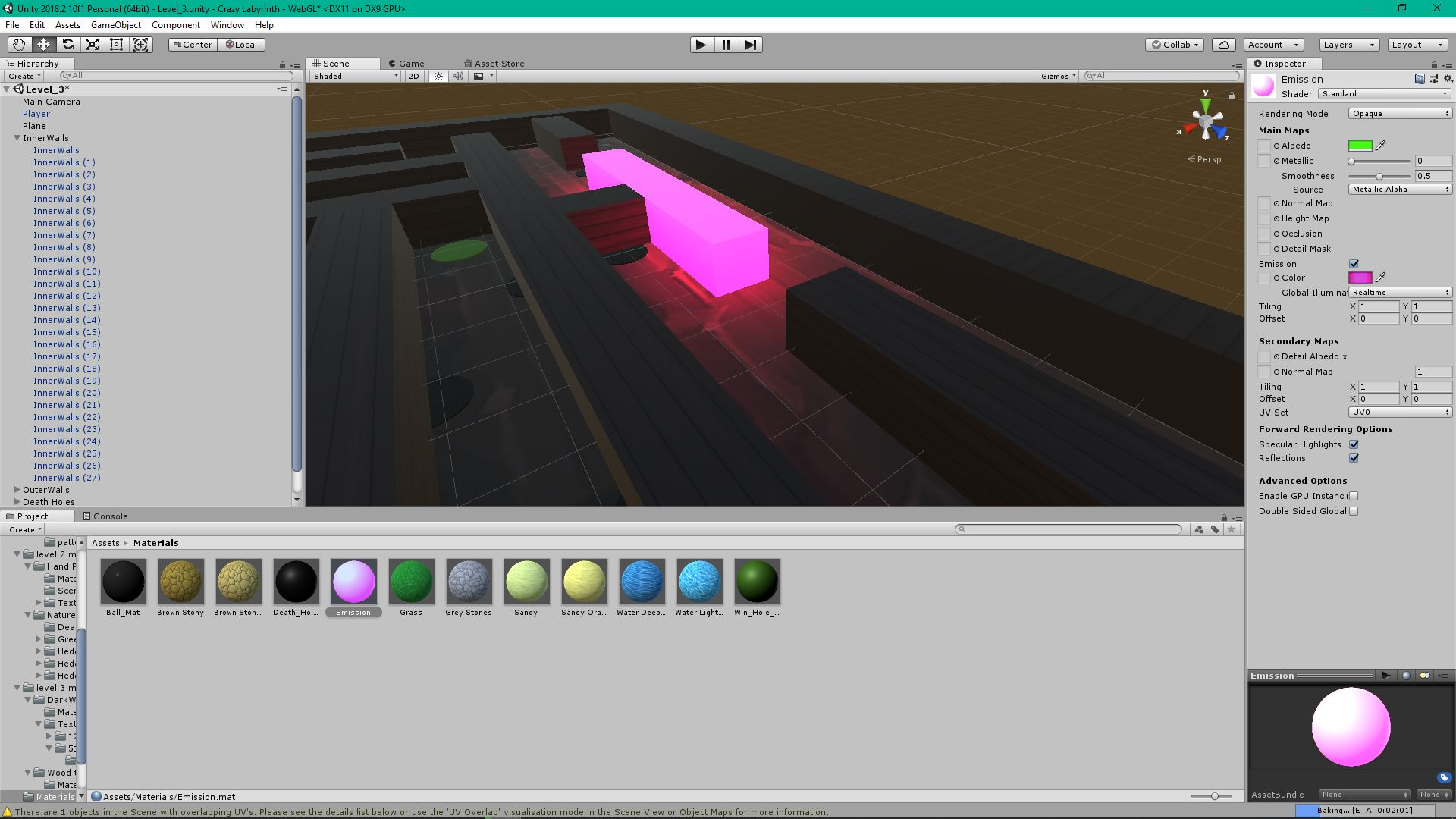




Emissive Materials

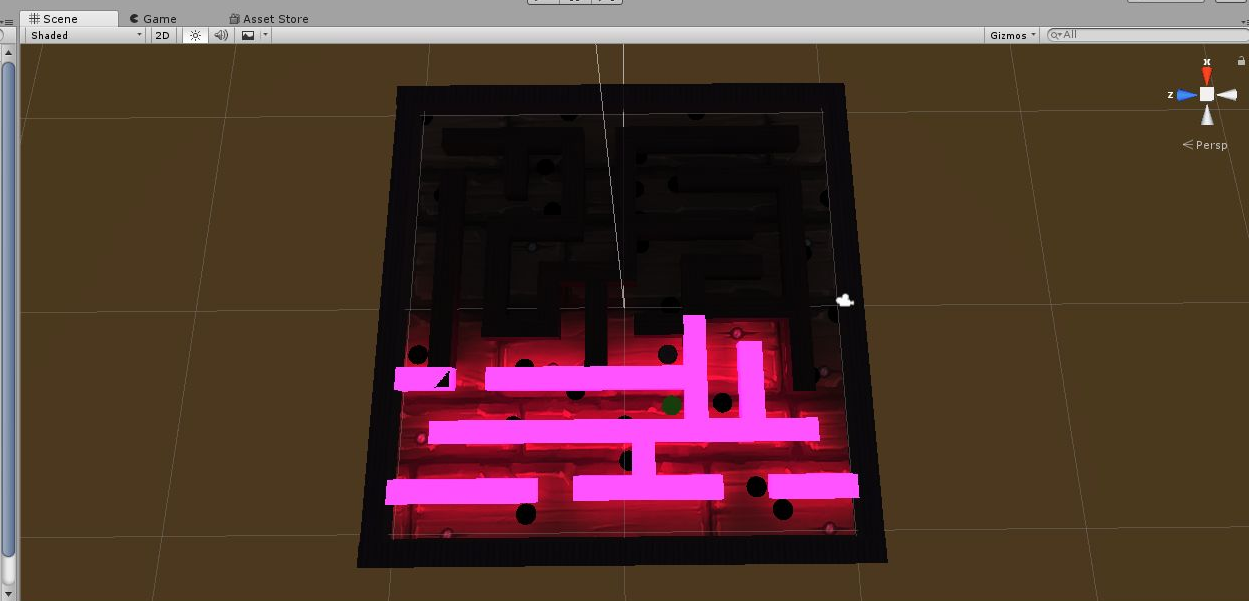
- Light is emitted off the entire surface of the selected object
- Properties of the lighting can be manipulated during gameplay
 - Color, strength, etc.
- Any static object in the scene is eligible to become an emissive material
- Lighting emitted will only affect static objects in the scene
- The light from these objects is visible across the entire scene, making emissive materials great for creating signs, especially neon, traffic lights, etc.





Hierarchy

- Create +
- Level_3*
 - Main Camera
 - Player
 - Plane
 - InnerWalls
 - InnerWalls
 - InnerWalls (1)
 - InnerWalls (2)
 - InnerWalls (3)
 - InnerWalls (4)
 - InnerWalls (5)
 - InnerWalls (6)
 - InnerWalls (7)
 - InnerWalls (8)
 - InnerWalls (9)
 - InnerWalls (10)
 - InnerWalls (11)
 - InnerWalls (12)
 - InnerWalls (13)
 - InnerWalls (14)
 - InnerWalls (15)
 - InnerWalls (16)
 - InnerWalls (17)
 - InnerWalls (18)
 - InnerWalls (19)
 - InnerWalls (20)
 - InnerWalls (21)
 - InnerWalls (22)
 - InnerWalls (23)
 - InnerWalls (24)
 - InnerWalls (25)
 - InnerWalls (26)
 - InnerWalls (27)
 - OuterWalls
 - Death Holes



Inspector

Emission

Shader: Standard

Rendering Mode: Opaque

Main Maps

- ☐ Albedo
- ☐ Metallic
- ☐ Smoothness
- ☐ Normal Map
- ☐ Height Map
- ☐ Occlusion
- ☐ Detail Mask

Source: Metallic Alpha

Emission

- ☒ Color
- Global Illumination: Realtime

Tiling: X 1 Y 1

Offset: X 0 Y 0

Secondary Maps

- ☐ Detail Albedo x
- ☐ Normal Map

Tiling: X 1 Y 1

Offset: X 0 Y 0

UV Set: UV0

Forward Rendering Options

- ☒ Specular Highlights
- ☒ Reflections

Advanced Options

- ☐ Enable GPU Instancing
- ☐ Double Sided Global

Project

Assets > Materials

Ball_Mat Brown Stony Brown Ston... Death_Hol... Emission Grass Grey Stones Sandy Sandy Ora... Water Deep... Water Light... Win_Hole_...

Assets/Materials/Emission.mat

Emission

AssetBundle: None

Baking... [ETA: 0:02:32]



Ambient Lighting

- Very unique form of lighting from the rest
- Lighting here illuminates the entire scene, with no clear indication of where the light is coming from
- Can be said to control the “master” brightness of a scene, without needing to mess around with all the other lighting in the scene
- VERY cheap to use

Lighting

SceneGlobal MapsObject Maps

Environment

Skybox MaterialDefault-Skybox

Sun SourceNone (Light)

Environment Lighting

SourceSkybox

Intensity Multiplier1

Ambient ModeRealtime

Environment Reflections

SourceSkybox

Resolution128

CompressionAuto

Intensity Multiplier1

Bounces1

Realtime Lighting

Realtime Global Illumination

Realtime Global Illumination is not supported on SM2.0 hardware nor when using GLES2.0.

Mixed Lighting

Baked Global Illumination

Lighting ModeShadowmask

Mixed lights provide realtime direct lighting. Indirect lighting gets baked into lightmaps and light probes. Shadowmasks and light probes occlusion get generated for baked shadows. The Shadowmask Mode used at run time can be set in the Quality Settings panel.

Lightmapping Settings

LightmapperProgressive

Auto Generate

Generate Lighting

0 Non-Directional Lightmaps0 BNo Lightmaps

Memory Usage: 0.0 MB

Occupied Texels: 0.0

Total Bake Time: 0:00:00

Lighting

SceneGlobal MapsObject Maps

Environment

Skybox MaterialDefault-Skybox

Sun SourceNone (Light)

Environment Lighting

SourceGradient

Sky ColorHDR

Equator ColorHDR

Ground ColorHDR

Ambient ModeRealtime

Environment Reflections

SourceSkybox

Resolution128

CompressionAuto

Intensity Multiplier1

Bounces1

Realtime Lighting

Realtime Global Illumination

Realtime Global Illumination is not supported on SM2.0 hardware nor when using GLES2.0.

Mixed Lighting

Baked Global Illumination

Lighting ModeShadowmask

Mixed lights provide realtime direct lighting. Indirect lighting gets baked into lightmaps and light probes. Shadowmasks and light probes occlusion get generated for baked shadows. The Shadowmask Mode used at run time can be set in the Quality Settings panel.

Auto Generate

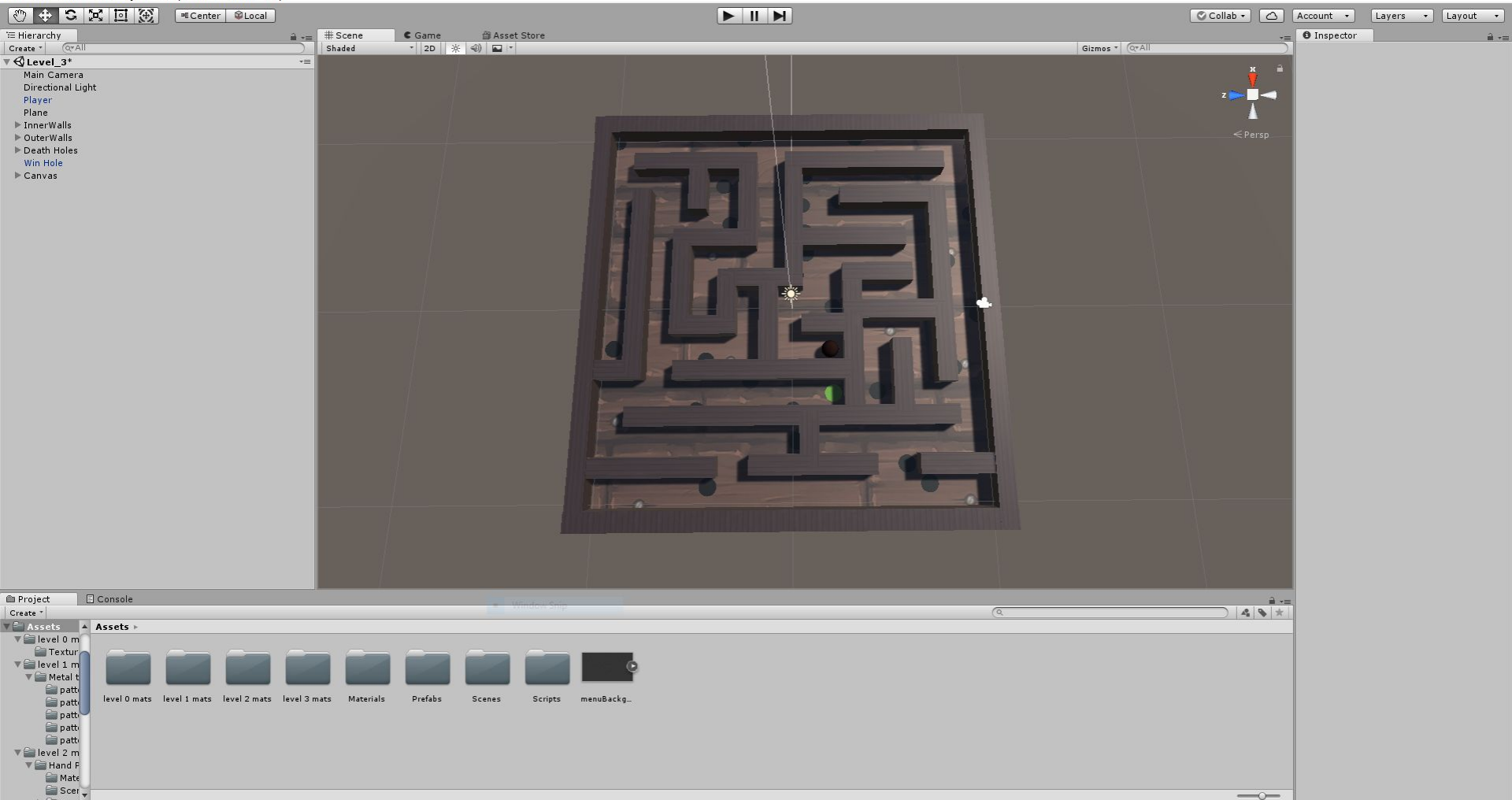
Generate Lighting

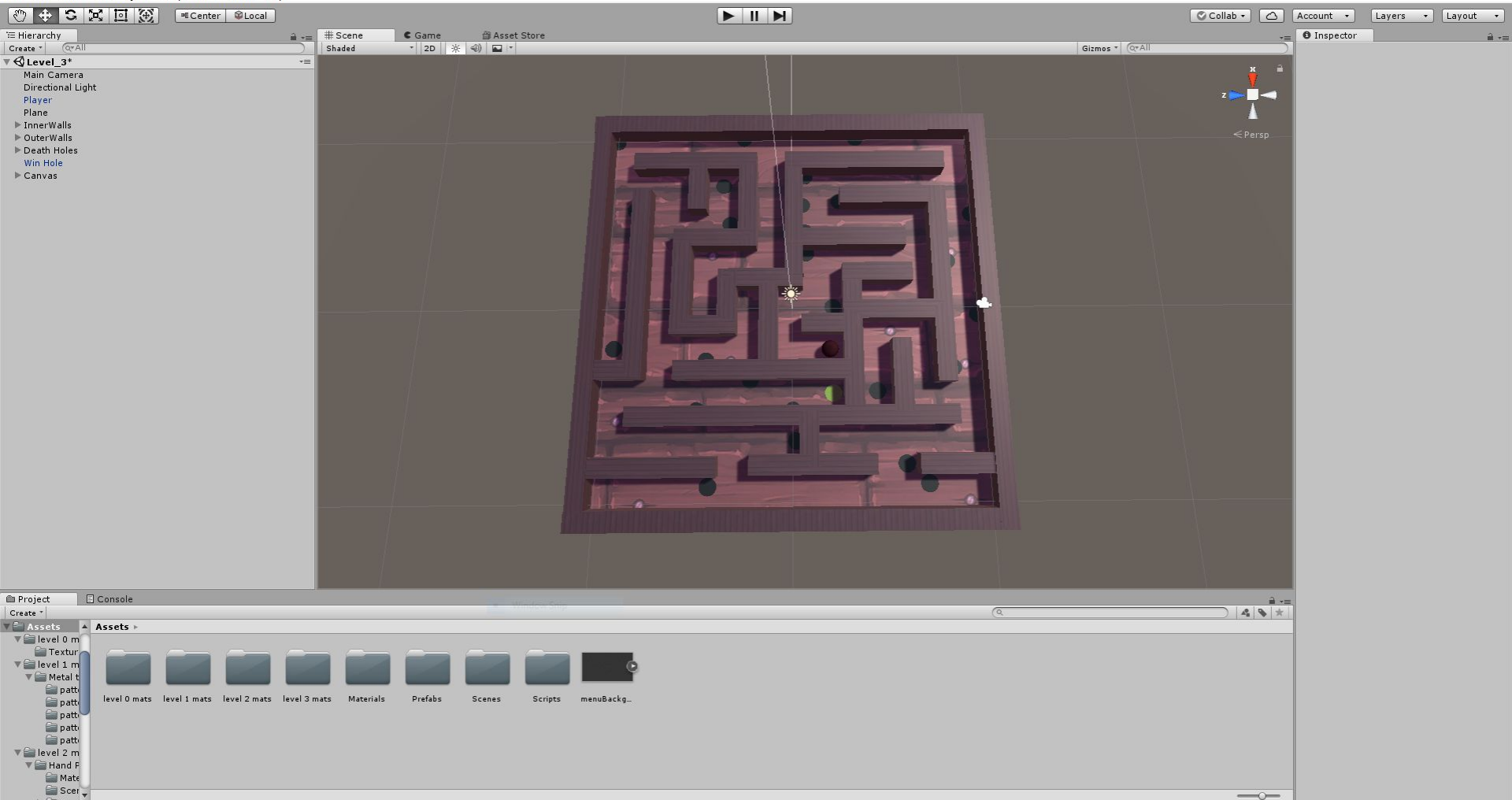
0 Non-Directional Lightmaps0 BNo Lightmaps

Memory Usage: 0.0 MB

Occupied Texels: 0.0

Total Bake Time: 0:00:00





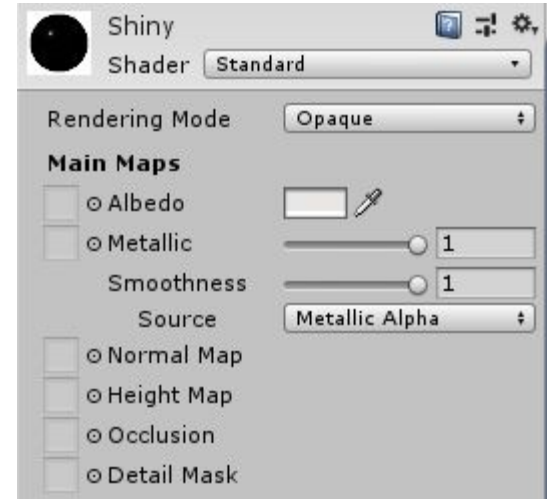
Reflections

- Add depth and realism to games
- Implementation is important
- Reflection probe component



Reflection Tutorial

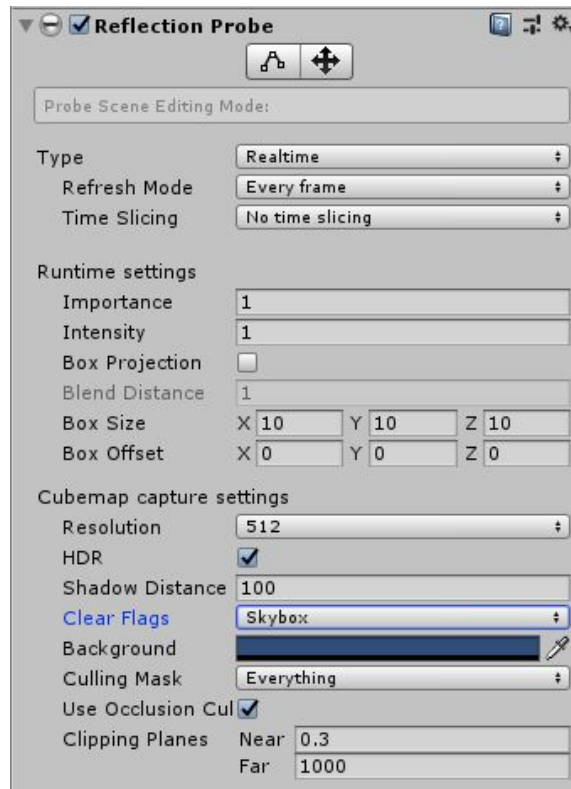
- Create a new material,
- Set metallic to 1
- Set smoothness to 1, color to light gray
- Apply material to ball
- Add component Rendering -> Reflection Probe to ball
- In Reflection Probe component:
 - Type: Realtime
 - Refresh mode: Every frame
 - Time slicing: no time slicing



Reflection Probe Breakdown

Most important parameters:

- Type
 - Baked, Realtime
- Refresh mode
 - On Awake, Every Frame, Scripting
- Time slicing
 - All faces, individual, no time slicing
- Resolution
 - Lower looks worse, performs better





Before Adding Reflections



After Adding Reflections





Conclusion

- Lighting adds atmosphere and feeling to our Unity creations.
- There are different types of lighting for a variety of different situations.