light.shadowAttenuation如何计算？

MixRealtimeAndBakedShadows(realtimeShadow, bakedShadow, shadowFade);

realtimeShadow如何计算？

SampleShadowmapFiltered(shadowCoord);

shadowCoord如何计算？

float4 TransformWorldToShadowCoord(float3 positionWS)

{

#ifdef \_MAIN\_LIGHT\_SHADOWS\_CASCADE

    half cascadeIndex = ComputeCascadeIndex(positionWS);

#else

    half cascadeIndex = 0;

#endif

    float4 shadowCoord = mul(\_MainLightWorldToShadow[cascadeIndex], float4(positionWS, 1.0));

    return float4(shadowCoord.xyz, cascadeIndex);

}

bakedShadow如何计算？

#if defined(SHADOWS\_SHADOWMASK) && defined(LIGHTMAP\_ON)

    half4 shadowMask = inputData.shadowMask;

#elif !defined (LIGHTMAP\_ON)

    half4 shadowMask = unity\_ProbesOcclusion;

#else

    half4 shadowMask = half4(1, 1, 1, 1);

#endif

inputData.shadowMask如何计算？

inputData.shadowMask = SAMPLE\_TEXTURE2D\_LIGHTMAP(unity\_ShadowMask

, samplerunity\_ShadowMask, uv);

shadowFade如何计算？

half GetShadowFade(float3 positionWS)

{

    float3 camToPixel = positionWS - \_WorldSpaceCameraPos;

    float distanceCamToPixel2 = dot(camToPixel, camToPixel);

    //z: oneOverFadeDist, w: minusStartFade

    half fade = saturate(distanceCamToPixel2 \* \_MainLightShadowParams.z + \_MainLightShadowParams.w);

    return fade \* fade;

}

MixRealtimeAndBakedShadows如何计算？

half MixRealtimeAndBakedShadows(half realtimeShadow, half bakedShadow, half shadowFade)

{

#if defined(LIGHTMAP\_SHADOW\_MIXING)

    return min(lerp(realtimeShadow, 1, shadowFade), bakedShadow);

#else

    return lerp(realtimeShadow, bakedShadow, shadowFade);

#endif

}