Lightweight Render Pipeline

(翻译)

Lightweight Render Pipeline (LWRP)，轻量级渲染管线，是一个Unity预制的Scriptable Render Pipeline (SRP)。LWRP可以为移动平台提供图形渲染功能，但你也可以在高端主机和PC上使用LWRP。LWRP使用简化的、基于物理的光照和材质。

LWRP使用single-pass 前向渲染(forward rendering)。使用LWRP，可以在几个平台上 得到优化了的实时渲染性能。

LWRP支持以下平台:

- Windows and UWP
- Mac and iOS
- Android
- XBox One
- PlayStation4
- Nintendo Switch
- All current VR platforms

Lightweight Render Pipeline有两种可用的模板: LWRP和LWRP-VR。LWRP-VR具有预先开启的VR设置(pre-enabled settings for VR)。两者的文档是一样的。

注意: Built-in和 custom Lit Shaders无法和LWRP一起工作。因为LWRP有一套新的standard shaders。如果你升级(upgrade)当前项目来使用LWRP，你需要升级built-in shaders到LWRP对应的新的shaders。(LWRP提供了相应的工具)

注意: 使用LWRP的项目无法与 High Definition Render Pipelin及Unity built-in rendering pipeline兼容。因此开发前，你要想清楚用哪一个渲染管线。

Getting started with LWRP

为了使用Lightweight Render Pipeline，你可以新建一个项目或者升级已存在的项目。

一旦你有了一个使用LWRP的项目，你必须创建一个Scriptable Render Pipeline (SRP) Asset (LWRP提供了工具)，然后配置项目的Graphics settings。

接下来三个子章节为细节讨论：

- 创建一个新项目来使用LWRP
- 升级旧项目来使用LWRP
- 配置LWRP，包括创建SRP Asset，修改Graphics settings。

Using LWRP in a new Project

如果你想在新项目中使用LWRP，你可以用LWRP模板(templates)来创建新项目。

使用用模板创建新项目(适用于Unity 2018.1后的版本):

1. 打开Unity。在Unity主页(Home)，点击**New**来创建一个新项目(new Project)。模板(**Template**)下拉框中选择LWRP模板。
2. 点击**Create Project**。Unity会自动为你创建一个带有全部LWRP功能的新项目。

关于更多关于模板使用的信息，请查阅Project Templates

注意: 创建完项目后，在开始使用LWRP前，你必须配置LWRP，即创建SRP Asset，修改Graphics settings。具体做法，请继续看本文下面的** Configuring LWRP for use**。

Installing LWRP into an existing Project

如果你是新建项目来使用LWRP，可以直接跳过这一小节，直接看下一小节**Configuring LWRP for use**。

你可以通过Unity的Package Manager System来下载和安装最新版本的LWRP。

为已存在的项目安装LWRP:

1. 在Unity中，打开项目。在顶部的导航条，点击 **Window > Package Manager**来打开**Package Manager**窗口。选择**All**标签页。这个标签页显示了当前版本的Unity可以使用的包(packages)。
2. 从packages列表中选择Lightweight Render Pipeline。在窗口右上角点击**Install**。这会直接为你的项目安装LWRP。

注意: 在开始使用LWRP前，你必须配置LWRP，即创建SRP Asset，修改Graphics settings。具体做法，请继续看本文下面的** Configuring LWRP for use**。

注意: 把一个已存在的项目切换到LWRP会消耗大量时间和资源。LWRP使用custom lit shaders，是不和built-in Unity lit shaders兼容的。你必须手动修改或转换大量内容。嫌麻烦的话，可以考虑使用LWRP启动一个新项目。

Configuring LWRP for use

为了配置和使用LWRP，你必须首先:

- 创建Lightweight Render Pipeline Asset，然后
- 添加此Asset到项目的Graphics settings中。

下面有更详细的步骤。

Creating the Lightweight Render Pipeline Asset

Lightweight Render Pipeline Asset控制着项目的全局rendering和quality settings，并且创建rendering pipeline instance。rendering pipeline instance包含中间资源(intermediate resources)和render pipeline的实现。

创建一个Lightweight Render Pipeline Asset的步骤：

1. 在Editor中，转到Project window。
2. 在Project window中右击，选择**Create > Rendering > Lightweight Render Pipeline > Pipeline Asset**。或者在顶部菜单栏中点击**Assets > Rendering > Lightweight Render Pipeline > Pipeline Asset**
3. 对于此新创建的Asset，可以使用默认的名字或者输入一个新名字。现在你已经创建了一个LWRP Asset。

提示: 你可以为不同平台(或不同测试环境)创建多个具有不同配置的LWRP Assets。

Adding the Asset to your Graphics settings

为了使用Lightweight Render Pipeline，你必须把刚刚新创建的LWRP Asset添加到Unity的Graphics settings中。如果不这样做，Unity会尝试使用built-in render pipeline。

1. 在顶部菜单栏点击: **Edit > Project Settings > Graphics**。
2. 在**Render Pipeline Settings**域中添加你刚创建的LWRP Asset。添加完成。

Lightweight Render Pipeline Asset

LWRP Asset为Lightweight Render Pipeline控制着几个graphical features和quality settings。它是一个scriptable object，继承自`RenderPipelineAsset`。当你把LWRP Asset设置到Graphics settings时，Unity会从built-in render pipeline切换到LWRP。你可以直接在LWRP中调试相应的配置，而不用去别的地方了。

你可以拥有多个LWRP assets，并在它们之间进行切换。例如，你可以有一个打开阴影的LWRP Asset和一个关闭阴影的LWRP Asset。然而，你不能在HDRP/SRP和LWRP assets之间进行切换，因为它们之间是互不兼容的。

Shader Stripping

Unity可以从单个Shader源文件中编译出多个Shader变体(Shader Variants)。Shader Variants的数量取决于你的Shader中包含了多少关键字。在默认着色器中，Lightweight Render Pipeline为lighting和shadows使用了一系列关键字。LWRP可以根据你在LWRP Asset中激的features，排除一些Shader variants。

当你在LWRP Asset中禁用某些featurers时，pipeline会从构建中(from the build)剥离相关的Shader variants。剥离Shaders使你的build sizes更小，需要更短的build times。如果你的项目不会用到某些features或keywords，Shader Stripping就很有用。

例如，你的某个项目中directional lights可能用不到shadows。没有Shader stripping的话，带有directional shadow支持的Shader variants仍然存在于build中。如果你知道你根本用不到shadows，你可以在LWRP Asset中uncheck **Cast Shadows**。这样LWRP会把这些Shader Variants从build中剥离。

Built-in/LWRP comparison

这里有一张Feature comparison table，显示了LWRP和Unity Built-in render pipeline支持的features的对比。

Shading models in Lightweight Render Pipeline

一个shading model定义了材质的颜色是怎么随着影响因素(如：surface orientation, viewer direction 和 lighting)变化的。你选择哪个shading model决定于艺术倾向(artistic direction)和程序的性能预算。LWRP提供的Shaders具有以下shading models:

- Physically Based Shading
- Simple Shading
- Baked Lit Shading
- No lighting

Physically Based Shading

Physically Based Shading (PBS) 基于于物理定律通过计算表面(surface)的反射光问题来模拟物体在真实世界中的样子。利用它可以生成逼真的物体(objects)，即真实感渲染。

PBS模型遵守以下两条规则:

Energy conservation - 能量守恒，表面反射的光的总量决不会大于总的入射光量。除非物体本身也是发光体，如霓虹光。Surfaces在微观层次具有微几何形态(Microgeometry)。一些物体具有光滑的微身体形态(smooth Microgeometry)，这使物体看起来像“镜子”。别的物体具有粗糙的微身体形态。使用LWRP，你可以模拟被渲染物体的表面的光滑度。

当光到达被渲染物体表面时，部分光被反射，部分光被折射。反射的光叫镜面反射，*specular reflection*。它会随着Camera的方向和表面上的到达点而变化。在这个shading model中，高光区域(specular highlight)的形状是用GGX function来近似的。

对于金属物体，其表面会吸收和改变光。对于非金属物体(non-metallic objects)，其表面会反射部分光。

光的衰减只受光的强度(light intensity)影响。这意味着你不必为了控制衰减而增加光的范围(range)，因为没用的。

下面这两个LWRP Shaders使用Physically Based Shading:

- Lit
- Particles Lit

注意:PBS模型不适用于低端手机硬件。如果你的目标平台是这种低端手机硬件，请使用下面要讲的**Simple Shading**模型。

Simple shading

这个shading模型适用于风格化的视觉效果(stylized visuals)或者运行于非高端硬件平台(less powerful platforms)上的游戏。使用这个shading model的材质并不具有真正的(truly)photorealistic。这个shading model不遵守能量守恒定律，是基于Blinn-Phong模型的。

在Simple Shading model中，材质反射漫反射光和镜面高光，两者之间没有关联。材质反射的漫反射光和高光的问题取决于你设置的材质属性，并且反射光的总量可以超过超过入射光的问题。镜面反射仅随camera direction变化。

光的衰减(light attenuation)仅受光强影响。

下面这些LWRP Shaders使用了Simple Shading:

- Simple Lit
- Particles Simple Lit

Baked Lit shading

Baked Lit shading没有实时光照(real-time lighting)。材质可以接受来自lightmaps或Light Probes的baked lighting。这以较小的性能代价为场景增加了一些深度。使用这个shading model的游戏可以运行于less powerful platforms。

LWRP Baked Lit shader是唯一使用Baked Lit shading的shader。

Shaders with no lighting

LWRP提供了一些不使用光照的Shaders。这意味着它们没有directional lights，也没有baked lighting。因为没有光照计算，这些shaders的编译速度会比有光照的Shaders快。

下面这些LWRP Shaders没有光照：

- Unlit
- Particles Unlit

参考:

1. Lightweight Render Pipeline