

## The very basic Renderman ( 三 )

### 二、Renderman进阶

在学会了Prman的一些渲染流程后，我们将测试一下Renderman的一些高级工具：Displacement、Volume、Multi-segment Motion Blur和使用自己写的Shader。

Renderman除了上面的SoftShadow外，还有几样杀手锏——Displacement、MotionSample、RiCurve等强劲效果。其中Displacement是众多渲染器中最好效果的、最真实的。

Renderman的Displacement有两种方式——Do Displacement和Bump。Do Displacement是真实的贴图置换，效果惊人，而Bump只是简单地改变法线的方向，效果还算可以。

一般渲染真实的Volume需要光线跟踪，而Renderman的Volume效果就不用光线跟踪都可以渲染出漂亮的体积光。

Motion blur是运动模糊，它可以表现十分有动感的运动效果。一般的Motion blur都是简单的帧与帧之间的运动插值，欠缺真实感。而分布式光线跟踪虽然可以做出漂亮的效果但计算时间又太长。RAT套件里的MotionSample就可以在短时间内做出漂亮的效果，实现Multi-segment Motion Blur。

renderman的一大特点是可以方便地使用自己写的Shader。自己写Shader的条件十分简单，只要有纯ASCII码的文本编辑器，要shader.exe，还要脑子里的一点编程经验。

### 第三章 分析Slim

#### 一、使用Slim

Slim是RAT包里的一个强大的ShadingNetwork工具，可以做出十分厉害的Shader效果。

注：关于怎么用Slim做出漂亮的Shader以及Slim的使用技巧，Quake007斑竹回在它的文章里作详细的分析和举例，在此我就越过不说了。

知道了Slim是怎么工作的，那我们就可以大大地扩充Slim的功能、加快Slim工作速度，而且可以做出无与伦比的shader。更厉害的是一它在不用编程的情况下都可以做出极其漂亮的Shader！！！！

不用编程！这对于很多想学Prman而又怕学编程的人来说是天大的喜事！

其实Slim的使用和Maya的HyperShade使用差不多，而且Slim的管理更方便一些，扩展性也更好些。当然，它和Maya的兼容性就比较差，要和Maya无缝结合就要写一写表达式。

Slim的使用是建立在sl和slo上的，也就是说，slim是简化或隐藏编程步骤的程序，它可以按用户的意志生成sl文件，也就是Shader的源文件，然后自动编译成slo文件——完成编译的Shader文件。

首先Slim会加载一些Template文件，这些Template文件是对Shader源代码的定义，包含了自动生成SI

文件的所有东西，通过Template，Slim可以知道在sl的哪里放入什么语句、哪里连接贴图等。

当Slim在做ShadingNetwork连接时，会把Template的Shader函数代码像砌积木似的堆积起来，就成了一个十分复杂的Shader。这个Shader的内容和形成过程用户无须知道，只要在Slim里做出你想要的结果就可以了。

Slim加载的Template为.slim文件，可以用notepad打开看看，懂得用的可以修改或者添加。

到\Pixar\RenderMan Artist Tools 4.5\lib\slim目录，可以看到有很多.slim文件，这里就是放.slim文件的地方。把magic\_color.slim、magic\_displacement.slim和magic\_layer.slim下载，Copy到这里。

回到\Pixar\RenderMan Artist Tools 4.5\etc目录，打开slim.ini，可以看到

```
# Slim Extensions
```

```
# put customuis first 'cuz templates refer to them.
```

```
set templates [file join [RATGetDir lib] slim]
```

这里就是安放Template的地方。

加入：

```
#magic project Extract Shaders
```

```
LoadExtension slim [file join $templates magic_color.slim]
```

```
LoadExtension slim [file join $templates magic_displacement.slim]
```

```
LoadExtension slim [file join $templates magic_layers.slim]
```

重新启动Maya。

这时候你就可以在Slim的Create菜单里看到加入了magic Color、magic displacement、magic layer这三个Shader。

如果要自己制作Template也不难，但就要有一点写Shader的经验了。可以分析那些.slim文件，通过分析可以知道编写的规律。

Mayaman里有很多精彩的Shader，要把他们用到Slim里去就要花点工夫，把他们一个一个地制作成Slim Template。有时间我把我已经做了的贴出来。

## 二、Slim Shader分析

我们来看一些用途广泛的Slim Shader。

Prman里的Shader总地分为Surface、Displacement、Light、Volume。但Slim的Shader分类就更细致。

Ensemble：我建议初学Slim的朋友多用这个Shader做ShadingNetwork，它可以妥善地管理Surface、Displacement、Atmosphere、RIB box等Shader，轻易地做ShadingNetwork连接。

AdditiveFX：一个非常好的FXShader，十分厉害，可以在一种ShadingModel的基础上添加五层贴图！使用它很容易就可以做出十分漂亮的Shader。

Blinn：常用的Shader，和Maya的差不多。

BushedMetal：其实就是各向性Shader，做金属不错，但不要用在表面比较平坦的物体上。

Cel：ToonShader，可以渲染出卡通效果，十分好用，但要扩展其功能就要下载更好的rat\_toon Shader。

Ceramic：做一些晶莹剔透的东西十分管用。

Clay：黏土Shader。

Constant：常量Shader，就是不管光照情况，都保持同一颜色的Shader。

Gooch：做气球比较好。

Fresnel Plastic：加入了Blinn里的Fresnel项的Phong，做透明物体特棒，而且可以模拟折射。

Jewel：做宝石、珠宝等有效，但不常用。

Layer：被誉为所有的渲染器中最好的Layer Shader，可以添加各种Surface，做各种融合，十分厉害，很容易就做出复合Shader，建议多练练多使用。

Matte：最简单的Shader，类似Lambert。

Metal：和Max的Metal差不多。

Mix：混合Shader，可以决定混合的因子，比Layer灵活，但不如Layer强大。

Plastic：典型的Phong，和其他渲染器的Phong没有区别。

Shared：可以把一个SurfaceShader的一部分拿过来Share，但可以修改另一部分属性而不影响源Shader。

Skin：皮肤Shader，但没有什么实际用途。

ThinkPlastic：用于表现比较薄的塑料。

Velvet：控制高光和物体边缘的距离。

### 三、使用Slim自定义界面

Slim的界面可以由用户自己定义，以适应各种制作的不同需要。Slim的界面由Sli文件定义，源自Template里的定义，可以在Slim里修改。

注：关于Slim更高级的用法以及Slim Template的编写我将在我的下一篇教程《Renderman Solution》里给出。

## 第四章、开始制作特效

Renderman是一个复杂的大系，它不同于其它的渲染器，必须有一定的基础才可以向上层发展。所以在前面花了很大的篇幅来说明基础性的东西。

过于复杂的特效不属于very basic的范围，这里只介绍简单的特效制作方法，更多的特效要自己去开发。

## 一、毛发

RAT的毛发效果十分吸引人，特别是Ultra Fur，可以用“十分惊人”来形容。

Fur可以直接添加毛发效果：

但Fur的使用不方便，特别是要慢慢地渲染预览，十分麻烦。Maya Fur的交互式操作很方便，如果可以用Maya的Fur编辑毛发，而用Renderman渲染就好了。

UltraFur就提供了这样的解决方案，它支持Paint Tool、Attractor等MayaFur的特性。但实际使用的时候还会有一点小问题，要积累使用的经验。

## 二、RiCurve

RiCurve是Prman的一个极其强大的工具，它一直受外国制作公司的青睐。

RiCurve的正式用法是在Renderman的API中调用RiCurve的函数。但RAT把调用的过程简化了，集成在界面中使用。

## 三、CSG

CSG是一种很Cool的工具，可以做出腐蚀、融合、侵蚀等强劲的效果。

CSG和布尔运算十分相似，但比布尔运算更先进。布尔运算用真实的几何体破坏化地进行造型生成，但CSG只是在渲染中用消隐算法渲染出布尔效果，速度快、效果好，而且消隐面上是目标物体的材质。

更厉害的是，它可以实现Nurbs和多边形之间的布尔运算！而且不是单一的布尔运算，是可以嵌套的，多层的布尔运算。

它的使用方法比较简单，只要把要做布尔运算的物体Group，然后用Renderman->Attributes->CSG->Primitive命令，再用CSG下的Different、Union、Intersection命令做差、和、交的集。