# “你”究竟是什么样子

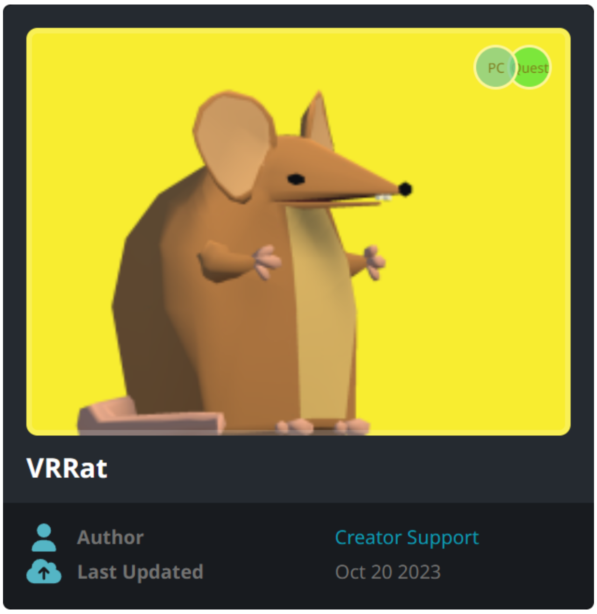
模型的组成

- 一、虚拟形象
- 二、骨骼
- 三、网络
- 四、材质
- 五、动画器

# 一、虚拟形象

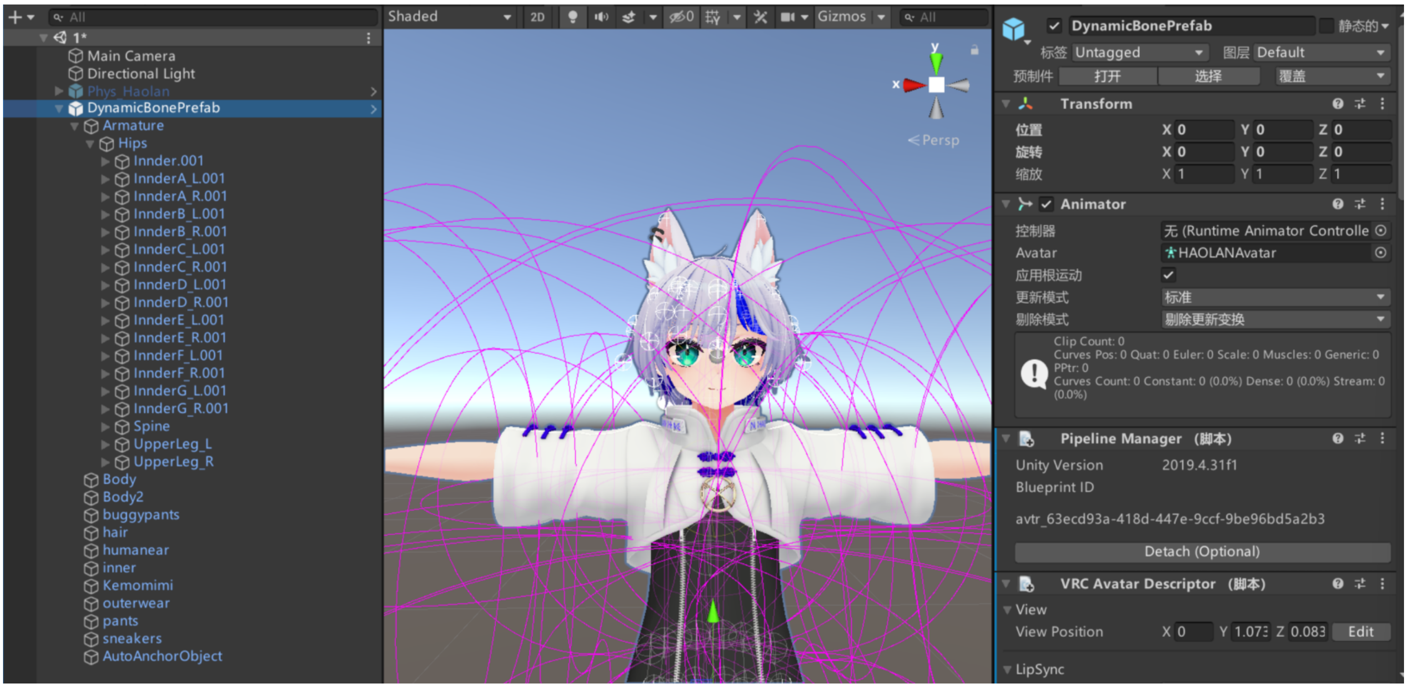
## 虚拟形象是什么?

虚拟形象是您在 VRChat 上可以使用，复制的各种“模型”，例如:


PC
Quest
VRRat
Author
Creator Support
Last Updated
Oct 20 2023

而在 Unity 这边，虚拟形象是：您导入的从别处获得的标准人物模型，加上VRChat对模型的特殊组件而形成的，一个包含很多东西的物品。

下图是一个标准的，已经配置好的虚拟形象，通常长这样：


All
1*
Main Camera
Directional Light
Phys_Haolan
DynamicBonePrefab
Armature
Hips
Innder.001
InnderA_L.001
InnderA_R.001
InnderB_L.001
InnderB_R.001
InnderC_L.001
InnderC_R.001
InnderD_L.001
InnderD_R.001
InnderE_L.001
InnderE_R.001
InnderF_L.001
InnderF_R.001
InnderG_L.001
InnderG_R.001
Spine
UpperLeg_L
UpperLeg_R
Body
Body2
buggypants
hair
humanear
inner
Kemomimi
outerwear
pants
sneakers
AutoAnchorObject
Shaded
2D
Gizmos
All
y
x
Persp
静态的
预制件
打开
选择
覆盖
Transform
Animator
Pipeline Manager （脚本）
Detach (Optional)
VRC Avatar Descriptor （脚本）
Edit
LipSync

☑ DynamicBonePrefab

标签 Untagged　图层 Default

位置 X 0 Y 0 Z 0
旋转 X 0 Y 0 Z 0
缩放 X 1 Y 1 Z 1

控制器 无 (Runtime Animator Controlle
Avatar HAOLANAvatar
应用根运动 ☑
更新模式 标准
剔除模式 剔除更新变换

Clip Count: 0
Curves Pos: 0 Quat: 0 Euler: 0 Scale: 0 Muscles: 0 Generic: 0 PPtr: 0
Curves Count: 0 Constant: 0 (0.0%) Dense: 0 (0.0%) Stream: 0 (0.0%)

Unity Version 2019.4.31f1
Blueprint ID
avtr_63ecd93a-418d-447e-9ccf-9be96bd5a2b3

View
View Position X 0 Y 1.073 Z 0.083

它拥有以下组件:

- 标准人物模型
- 动画器
- Pipeline manager (中译蓝图管理器)
- VRC Avatar Descriptor (中译模型描述器)
- Physbone (中译物理动骨, 可选)
- Dynamic Bone (中译动态骨骼, 过时, 作为老模型的中介, 不作为上传使用)
- Contacts (中译触发器, 可选)

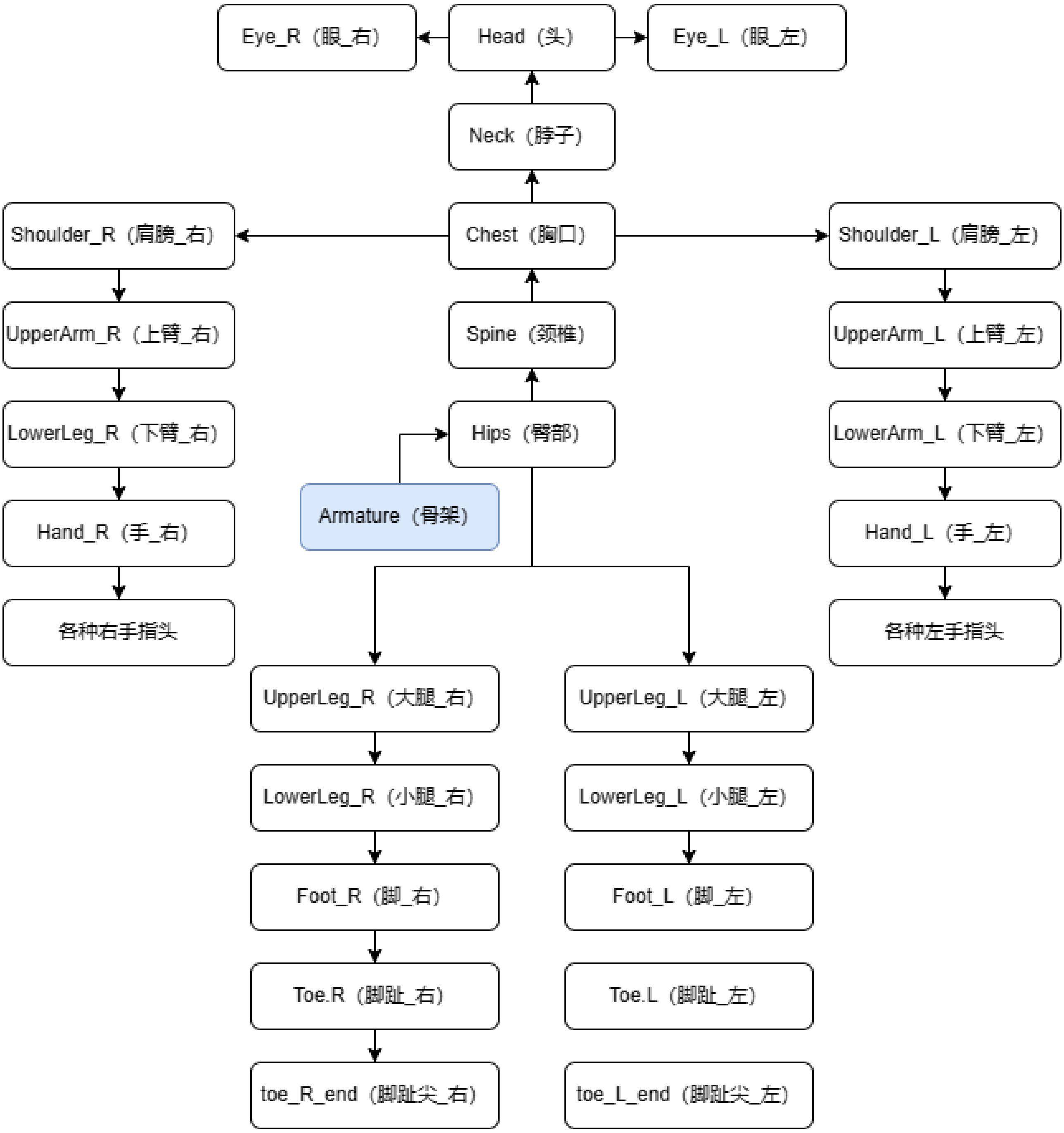
> **HAOLAN**，中所使用的模型: https://docs.vrcd.org.cn/books/vrchat/page/b9228

# 二、骨骼

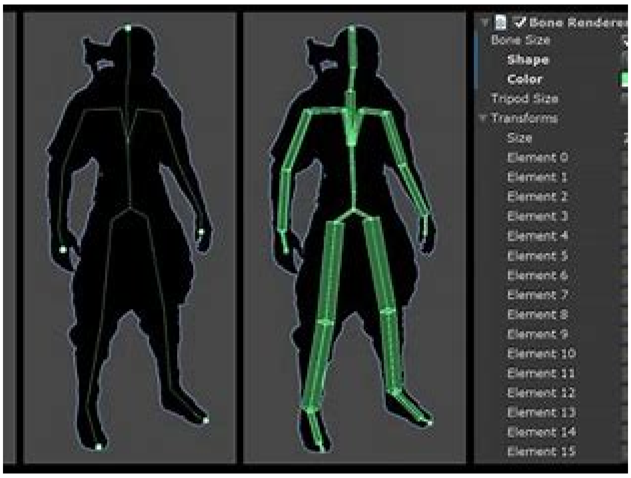
在美妙和谐并狂野的大自然中，人和人的动物朋友们用骨头支撑住身体，给肌肉提供发力点。为了在游戏和影视作品中模拟自然角色的运动和动作，骨骼系统应运而生。

## 模型的基本骨骼架构

一般的模型遵循以下的基本骨骼构造，此构造中不含有物理骨骼：


Eye_R（眼_右）
Head（头）
Eye_L（眼_左）
Neck（脖子）
Shoulder_R（肩膀_右）
Chest（胸口）
Shoulder_L（肩膀_左）
UpperArm_R（上臂_右）
Spine（颈椎）
UpperArm_L（上臂_左）
LowerLeg_R（下臂_右）
Hips（臀部）
LowerArm_L（下臂_左）
Hand_R（手_右）
Armature（骨架）
Hand_L（手_左）
各种右手指头
各种左手指头
UpperLeg_R（大腿_右）
UpperLeg_L（大腿_左）
LowerLeg_R（小腿_右）
LowerLeg_L（小腿_左）
Foot_R（脚_右）
Foot_L（脚_左）
Toe.R（脚趾_右）
Toe.L（脚趾_左）
toe_R_end（脚趾尖_右）
toe_L_end（脚趾尖_左）

Armature 作为骨骼的总集其只是一个载体，它不具有调控某一个骨骼区域的作用，我们模型的骨架是以 Hip 为中心的网状结构，具体架构参考上图，所以不要认为头是起点。


Bone Renderer
Bone Size
Shape
Color
Tripod Size
Transforms
Size
Element 0
Element 1
Element 2
Element 3
Element 4
Element 5
Element 6
Element 7
Element 8
Element 9
Element 10
Element 11
Element 12
Element 13
Element 14
Element 15

在这种架构下，我们的人物会以 Hip 为中心进行运动，也就是说我们在正常控制中Hip会相对于模型位置静止，此外在此构架下我们的模型运作形式将会以正向运动学（FK）和反向运动学（IK）两种形式实现

> 注：在骨骼中没有关节的概念，我们可以使用骨骼的一个端点作为关节，但它毕竟不是关节有偏差是正常的事情

# 物理骨骼

当我们想要在游戏中模拟头发，衣服，布料等属性时，比起真的耗费巨大性能对其进行解算，更加体面也更加被游戏业界所广泛使用的优化方法便是利用带有物理属性的骨骼带动面片运动以模拟头发，衣服，布料等属性。

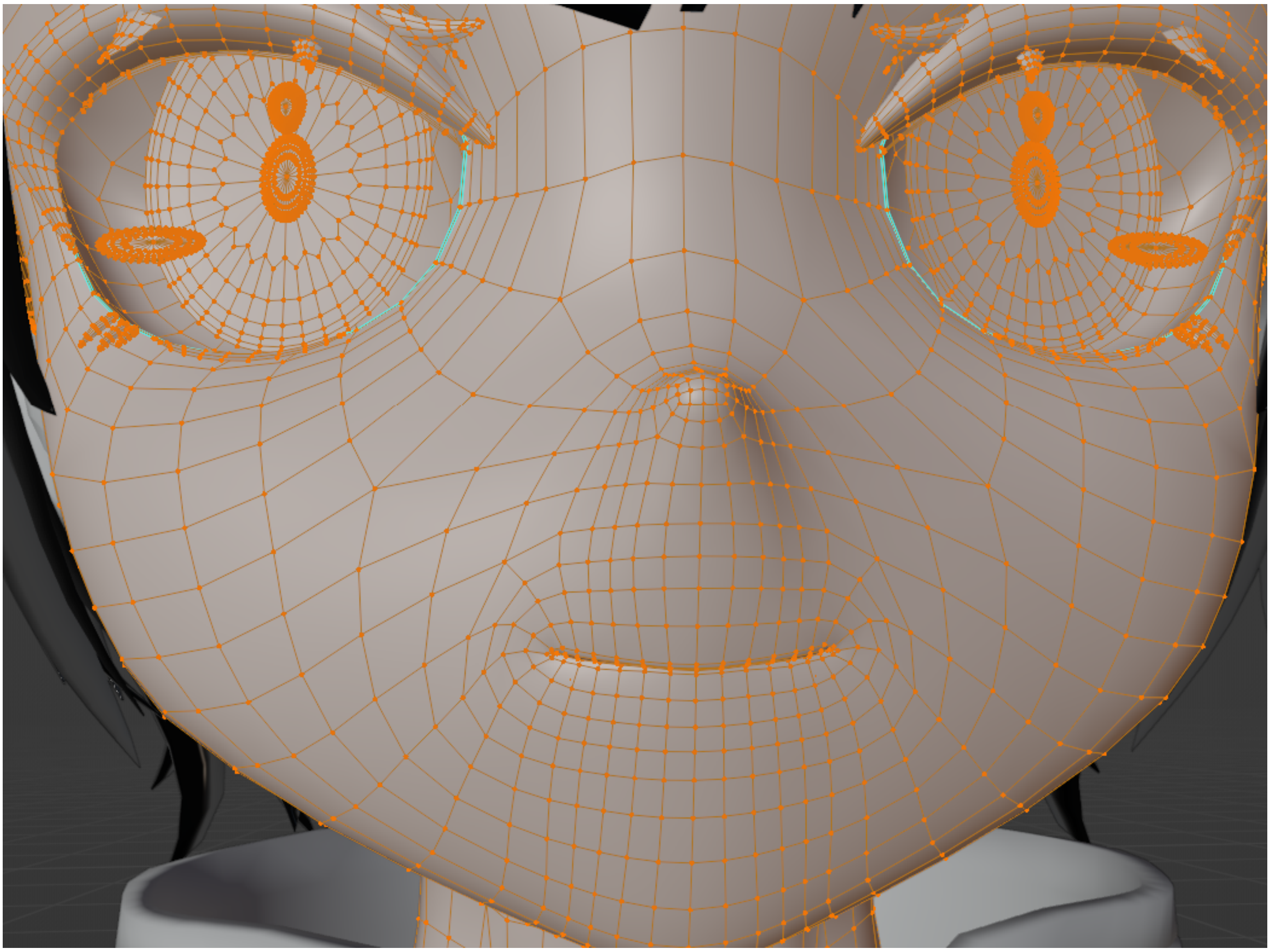
目前 VRChat 使用的物理骨骼 (PhysBone) 是一个简单的基于模拟弹簧振子的算法实现树状柔体的物理模拟插件，VRChat 同时也用它来实现玩家与玩家的交互效果。

> 点击此处查看 VRChat 汉化文档中心 的 PhysBone 文档。

VRChat [illegible]（**Dynamic Bone** [illegible] Phys Bone [illegible] 法支持玩家与玩家之间的交互，就目前来讲

# 三、网络

此处的网络相当于模型没有上色的皮肤，他们是由一个个小的面片拼接而成的集体



从计算机运行本身来讲在计算机图形学中实际上没有体积这一概念，有的只是点组成的不同片面根据相对于摄像机的远近距离决定的覆盖关系在光栅中的渲染结果所以设计程序的人们制作了网格系统，您可以把网格系统理解为在

VIDEO
READ
GAME
ALLDOCUBE

Blende表Maya3DmaxC4DZbrush 等

> 采用3d图形学，优化注意

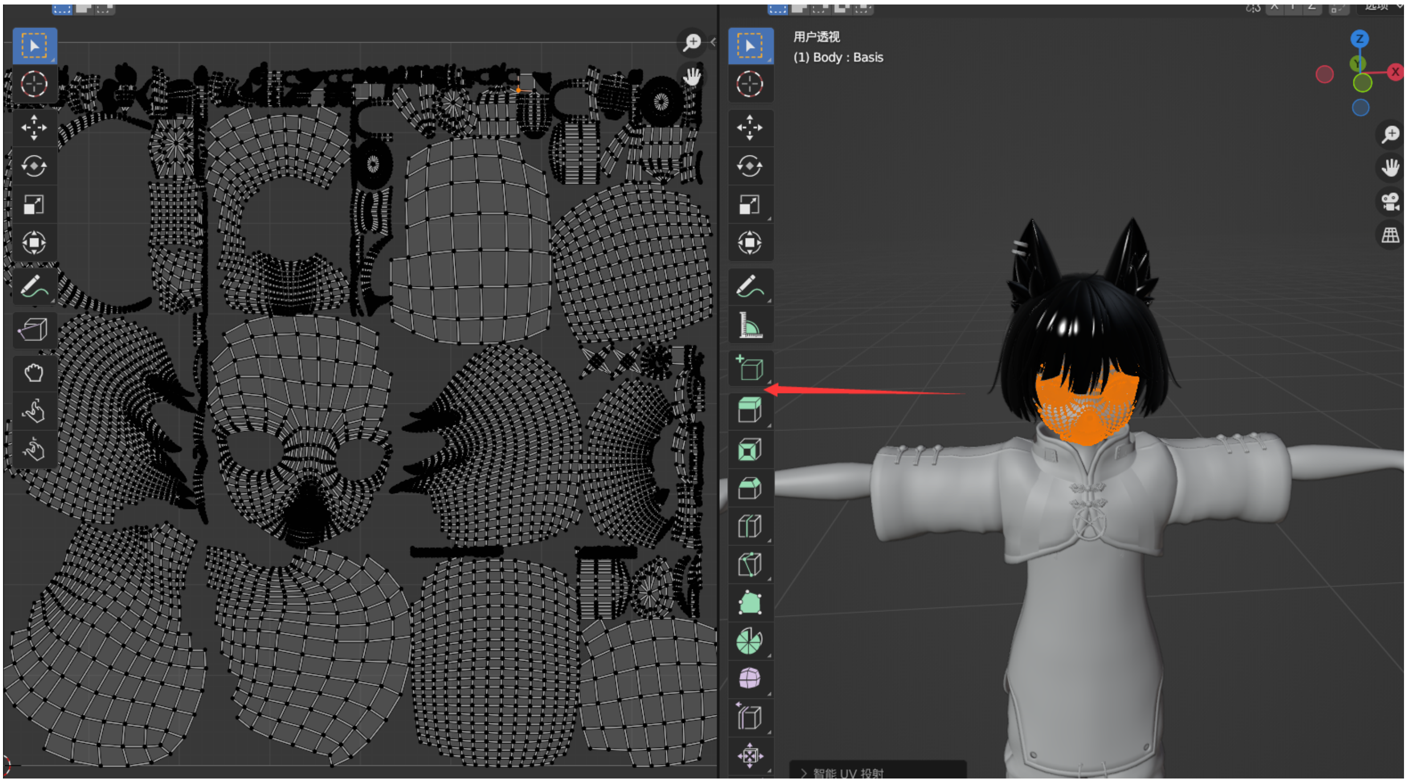
# 四、材质

当我们的模型有了一层未上色的皮后，我们便需要考虑给这层皮上色。然而，我们又不能像涂颜料一样给我们的模型上色，程序只能把图片贴到皮上面。而且，我们的皮是圆圆的且不平整的，这时候我们就需要制作对应的材质来

## UV

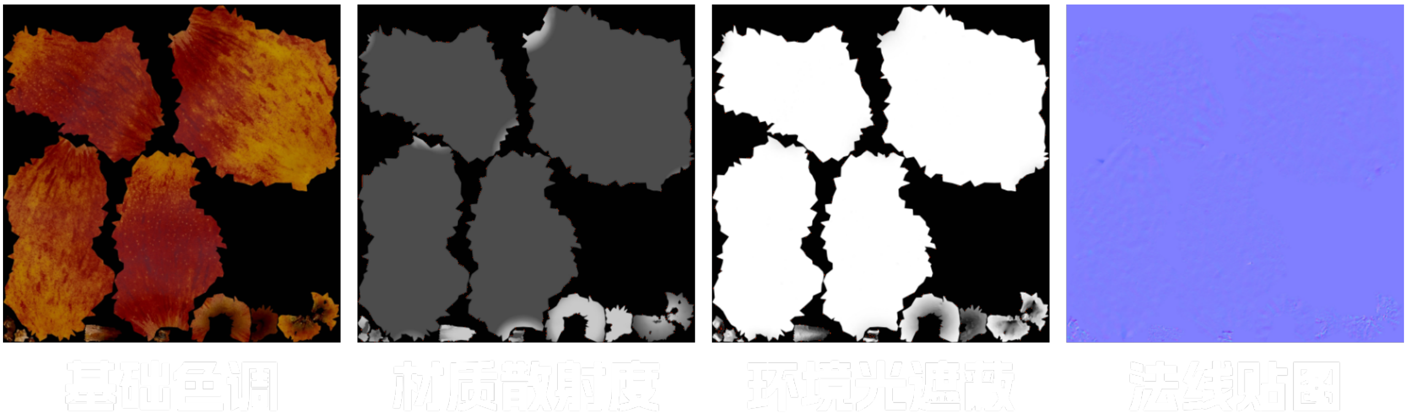
材质记录一个在 3D 世界中存在的物体拥有的纹理，颜色等表面特征，材质可以用来描述物体的纹理、质感、反射、折射等效果为了解决表面不平的问题我们把网络展开成平面，我们用 UV 投射将网络分成几个片域展平

这样我们就可以轻松的为对应区域上色形成贴图，



## 贴图

提贴图有以下几个基础的样式，以一个苹果贴图为例


基础色调
材质散射度
环境光遮蔽
法线贴图

- 基础色调：记录面的颜色/色彩信息，决定面的色彩
- 材质散射度：决定面的粗糙度
- 环境光遮蔽：记录面之间不同材质/网格体间的平滑阴影
- 法线贴图：用于丰富物体在光照下的细节，通过贴图记录高精度模型表面的角度变化，使其能在较低面数的情况下表达更加丰富的表面结构信息
- > 采用3d图形学，优化注意

# 着色器

在上完色之后，我们还不能直接将结果运用到网络上。对于程序来说，你所制作的贴图只是一张图片和一串数据。我们需要一个东西来解释这张贴图如何在网络上呈现和渲染。因此，着色器便应运而生。

着色器是一种用于控制贴图如何在显示中呈现和渲染的程序或代码。它可以让3D模型看起来更加真实或美观，也可以改变模型的光照、阴影、反射、透明等属性。它可以控制各种属性的贴图会以什么样的形式附着到模型上。

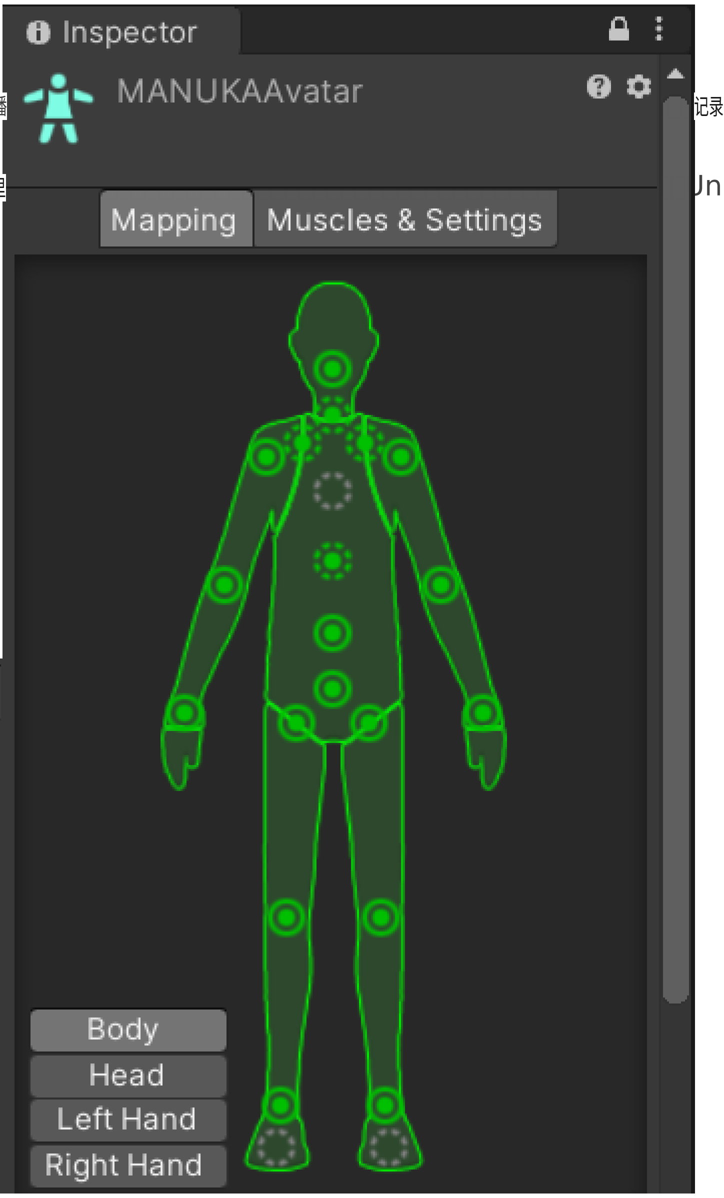
目前，VRChat 中常用的着色器有：

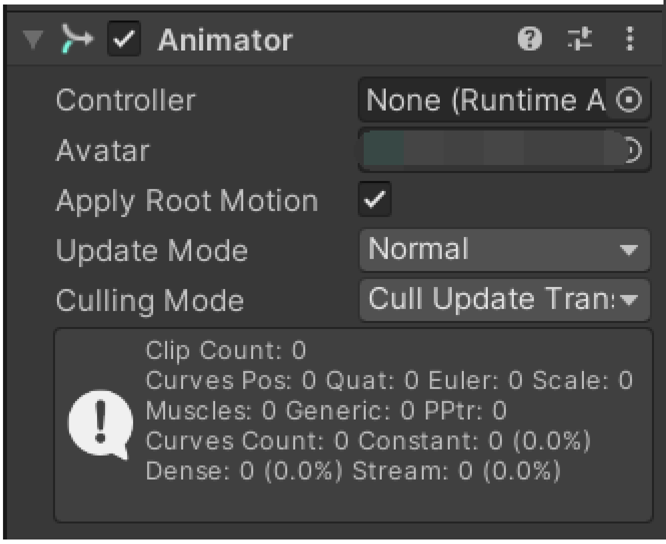
- Liltoon
- Poiyomi(Pro)
- UTS2(Unity Toon Shader 2)
- Arktoon Shader
- Sunao Shader(已停止更新)
- Crystal Reflection
- UnlitWF

# 五、动画器

动画器是一个模型运动的来源，在Unity这个游戏引擎中，模型的一切的自身运动都是由动画控制器播 记录在

我们一般会在我们的模型上看到Animator组件，这就是负责装载动画控制器的组件，当控制栏里 nity


Inspector
MANUKAAvatar
Mapping
Muscles & Settings
Body
Head
Left Hand
Right Hand



Controller None (Runtime A

Avatar

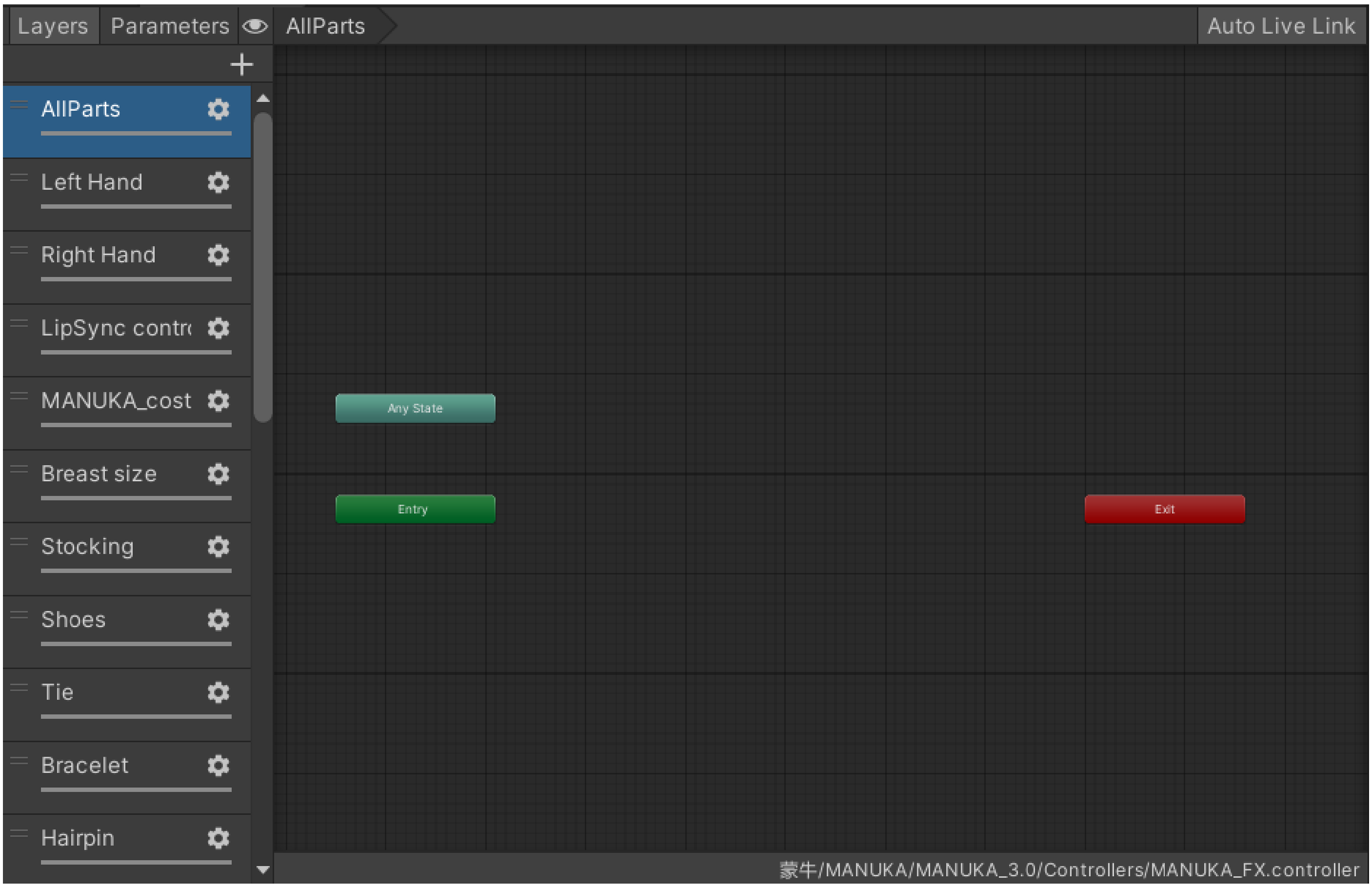
Apply Root Motion ☑

Update Mode Normal

Culling Mode Cull Update Tran:

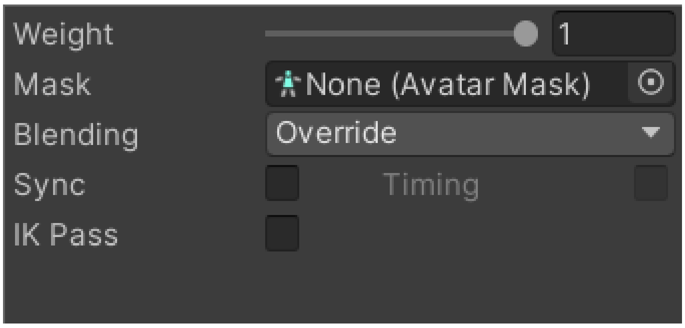
Clip Count: 0
Curves Pos: 0 Quat: 0 Euler: 0 Scale: 0
Muscles: 0 Generic: 0 PPtr: 0
Curves Count: 0 Constant: 0 (0.0%)
Dense: 0 (0.0%) Stream: 0 (0.0%)

在拥有了UnityAvatar后就可以用控制器控制你的模型啦，接下来就是我们该如何去编辑我们的控制器了


Layers
Parameters
AllParts
Auto Live Link
AllParts
Left Hand
Right Hand
LipSync contr
MANUKA_cost
Breast size
Stocking
Shoes
Tie
Bracelet
Hairpin
Any State
Entry
Exit
蒙牛/MANUKA/MANUKA_3.0/Controllers/MANUKA_FX.controller

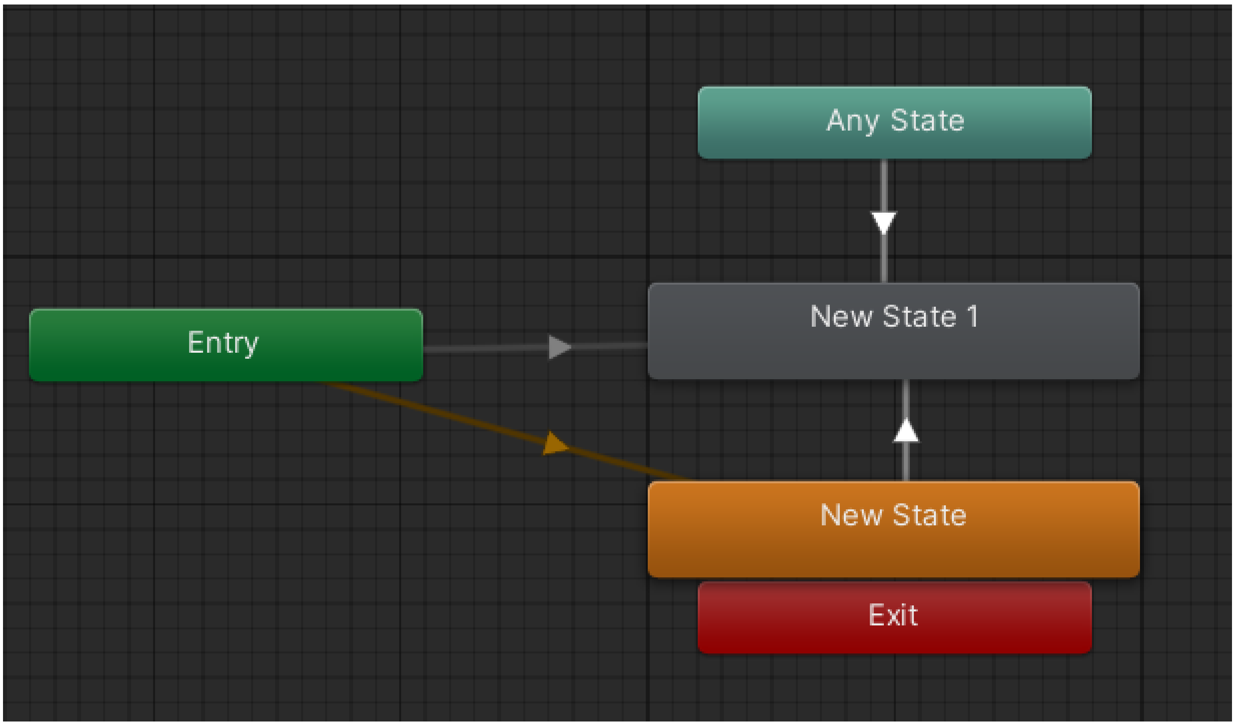
动画控制器分为两个主要区域，左侧有层级和参数，右侧是层级的视图区域

在动画控制器中，我们可以从视图窗口中清楚的看到某一层的逻辑路线，我们点开其中一层的设置窗口



Weight 1
Mask None (Avatar Mask)
Blending Override
Sync ☐ Timing ☐
IK Pass ☐

第一条Weight是我们需要注意的，他控制了这一层与其他层的关系


Any State
Entry
New State 1
New State
Exit

再来看视图窗口的东西，我们把控制器的运行比喻成用电，绿色的方块相当于电流的输出，电会从里面出来，白色的线是电线与开关，黄色的线就是电线，黄色方块是蓄电池，黄色方块与绿色方块之间没有开关限制是直连的，运