本演讲分几章。

1. **系统的目标**
2. **风模拟**
3. **风力马达**
4. **风接收器**
5. **数据和着色器实现，**
6. **我们学到的技巧**
7. **还有一些额外的东西，例如用于植被交互作用的细节和新方法**。

这些章节都试图回答一个简单的问题：

1. **我们为什么要这样做？**
2. **风在哪里？**
3. **是什么原因？**
4. **对此有何反应？**
5. **运作方式如何？**
6. **我们如何才能更好？**
7. **在完成所有这些工作之后，我们如何重用它以获得更多价值**？

首先，让我们了解一下现有技术。

**在基本的风力系统中**，

1. **风向和速度。 如果您保持状态并累积纹理运动，它可能会更改。**
2. **您具有随风滚动的噪波纹理或生成函数**
3. **然后，随着噪波滚动过去，移动网格顶点。**
4. **完成！ 您已经拥有了风力系统！ 装运它**。

这就是当今大多数游戏以及我们正在开发时使用的游戏。它是Crysis技术的简化版本，在**GPU Gems 3**中进行了描述。如果它可以满足您的所有需求，那么就满足您的所有需求。 但是我个人很高兴看到最近的一些游戏在推动更多游戏。《荒野大镖客2》做了一些让我感到卑鄙的事情。在本次演讲中，我们将几次重新访问该系统作为基础参考，然后扩展其概念。

好吧，那么我们打算做什么呢？一个更丰富的系统能为我们带来什么？我们的播放器功能强大，如何将其扩展到我们的环境？

**对于风的形状，我们有三个目标**。

1. **玩家应该能够观察他们的动作是否影响了风。**
2. **玩家应该能够观察风吹动的方式。**
3. **但同时，玩家应该能够观察速度或方向的空间变化。**

提醒您“可以观察”和“必须观察”是不同的目标。我们的目标是微妙，以免分散注意力。**如果我们在大多数情况下都无法实现这三个目标，那么增加系统的复杂性就不值得了**。

在开发资产和工作流程时，我们需要尊重这些目标以及其他一些目标。

1. 作为最低限度的选择，我们可以做得比“先吹风再摆动”要好。我们不仅要改善功能，而且即使在不需要完全动态系统的条件下也要取得更好的效果。
2. 对于所有风速，孤立模型在风吹过时都应该看起来不错
3. 一组模型在风吹过时应该看起来很好。换句话说，相邻模型的行为应具有内聚性有很多限制。 因此，考虑到我们的幻想环境和风格，我们不会将自己约束为现实行为，而是“令人信服”的行为。

好的，我为这次演讲做了最后一部分，但它反映了我在开发过程中的心态。 这从来不是风的PBR。我想认为我们交付的产品可以实现这些目标。让我们回到测试级别，看看它增加了什么……

**什么是风**?

有了这个简单的系统，我们就可以为环境提供单一的风向矢量。

1. 并且所有事物都使用该值，因此无论我们在哪里感兴趣，到处都是风。
2. 我们需要风来改变时空。单独地，这可以在程序上模拟一堆噪音。 如果负担得起，更多的噪音层几乎总是看起来更好。
3. 但是要突出局部效果和玩家动作，我们需要更多控制。 我们通过在一定空间内存储和更新许多风的值来做到这一点。

这是游戏中玩家攻击的特写镜头，在此范围内产生风XYZ中的风强度可视为RGB，可视为体积纹理的水平切片。从理论上讲，您可以使用任何一种流体模拟来更新每个帧– 2D或3D.对于我们的游戏，这是一个简单的3D对流/扩散模拟。渲染风向，进行模糊处理，然后向后采样以使风向前进。

我们的风模拟量涵盖了相当大的区域。

1. 面积为32 x 32米，高度为16米。您可以在那里看到Kratos，以供参考。
2. 3D纹理的分辨率为1米。 我们不仅限于立方体体素，但对于扁平或高体素的测试在我们的用例中并没有取得任何成功。
3. 为了在我们的交互式区域中获得更多的使用，我们将其居中放置在相机前面的一个点上。 我们仍然需要在玩家后面模拟一些东西，因此我们只推进25％。为了清楚起见，我们也是在垂直轴上执行此操作，因此屏幕上的交互式体积部分约为24x24x12m。调整这些数字的目标是功能分辨率并覆盖玩家可以扔到的距离（无论距离尽可能远或尽可能远）。

我们的风几何体称为风力发动机。有几种不同的行为。我们具有三种基本形状：

1. 球体
2. 圆柱体
3. 和圆锥。圆锥体是其中一头半径设置为零的圆柱体。

我们可以为这些形状的强度，位置，方向和比例设置动画。

风被渲染到模拟体积中。

的输出是什么？我们有三个简单的分布：

定向电动机沿主轴发出风。

全向电机从中心点向外散发风。

涡流电机绕主轴发出风。

我们有第四种特殊的风力马达：尾流马达。

1. 它沿行进方向输出风，与风速成正比。
2. 我要在这里暂停一秒钟。这些箭头表示纯粹的行进方向。我们发现更好的方法是使方向展开一点。
3. 因此，我们结合了远离电机中心的方向。在按速度缩放之前，我们以相等的权重执行此操作。
4. 这为唤醒电机提供了一种松果形状。
5. 这是动态输出。使用完整的模拟功能，通过游戏中的示例，这一点将变得更加清晰。
6. 最后一件事：我们使用两帧平均值作为运动输入。这样在紧弯曲的路径（例如斧头挥杆）上会产生更多的向外角度。

\*\*\*您可能不需要此。.老实说，我们也可能不需要此。这不是故意的。在项目后期，我们发现通过计算当前帧位置与上一帧先前位置之间的距离来意外地做到了。我已修复了该错误，并且之前看起来更好。因此，我撤消了修复程序，并重命名了变量。年度最佳游戏。

在该示例中，您可能已经在模拟调试视图中注意到，存在一个与我们看到的行为相匹配的渐变。为什么我们看到这个梯度？嗯，流体正在按预期的速度加速。.**但是包围盒外面没有东西要从边缘推入，所以那边的风比较慢**。这是我们需要解决的问题。

到目前为止，我们一直在尝试使用模拟代替我在介绍中描述的简单风力系统。（1）但是我们可以通过使用简单的系统定义环境风基线来解决此梯度问题。我们使用它来代替大型电动机。 这使风在盒子外面存在，并使动态系统保持整洁。（2）另外，我们可以使用它在系统级别添加一个额外的噪声层，从而创建阵阵和平静的局部区域。

**我们知道风在哪里。风起什么作用（哪些物体对风有响应）**？

**音效**

这很快，因为我不知道我在说什么。

有时音效想知道风速。

我们使用自定义WWISE属性，以便他们可以使用此值进行混合。

**布料**

不为此而激动，因为

对我们不起作用。 但我会告诉您我们尝试了什么。

每个布料对象在其变换中心对风系统进行一次采样

对该力施加到网格上的每个模拟点...

按力方向和布法线的点积缩小。这是为了在布料旋转和拍打时为我们提供各种运动。

和往常一样，我们播放一些音效。

为什么不应用？我们为新战神进行了大修。您可能注意到了。

1. 我们必须做出让步。我认为我们做出了很多正确的决定。 布料系统是一个没有升级的组件。 加风暴露了很多问题。我们希望重新审视布料和整个布料系统的发展。该优先级较低，因为..
2. 我们仍然可以在游戏中移动布料。其中一些是非交互式预模拟元素。这些主要用于电影或具有设计功能的对象，例如此船坞标志。
3. 我们也可以通过使用网状接收器完全跳过衣服系统来获得交互性。我将在短期内谈论这些。这些物体不会碰撞，但它们会响应风，因此也会响应玩家的动作。

**粒子**

每个粒子在其位置采样风速和风向。艺术家控制粒子对此信息的反应程度。

1. 从概念上讲，粒子以与拖动相同的方式进行响应，但以移动的流体进行响应。
2. 因此，缓慢或静止的粒子会加快速度以匹配其周围的流体，
3. 而快速移动的粒子会变慢。
4. 在每一帧中，我们计算每个粒子与风匹配所需的脉冲；并根据粒子的风影响值进行缩放。
5. 目标风速是环境风和动态风的完全组合。

这在早期的测试中显示了很多希望。

我希望自己做些什么？ 我们如何改善前进的道路？ 我想探索两件事：

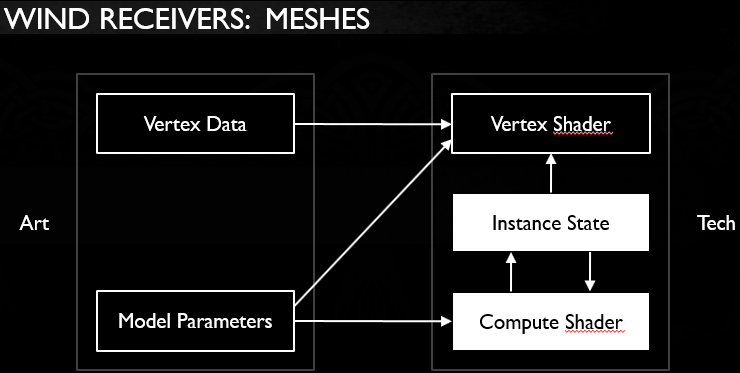
首先，我认为，如果我们对环境风和动态风的影响分别进行控制，我们可以让粒子系统忽略关卡风，而只对动态效果做出响应。这似乎是一个骇客，一个坏主意-这就是我们为什么不这么做的原因-但实际上，fx美术往往会在某种环境下进行烘焙。在完成fx工作之前，关卡并不总是最终确定的。同样，强大的环境风速往往会在粒子甚至消失之前将其推离屏幕。 第二，我们的粒子系统在添加风之前已经使用了噪声场。我们的想法是我们可以使这些组件独立。但是我认为，在平静和多风的情况下，在不同的湍流设置之间进行混合可能会阻止我们在强风条件下运送的某些“滚动粒子”外观。

**Mesh**

对此做了很多改进

在游戏中广泛使用。人物，道具，假布。

此系统包含5个组件。2个是动态风的新功能。

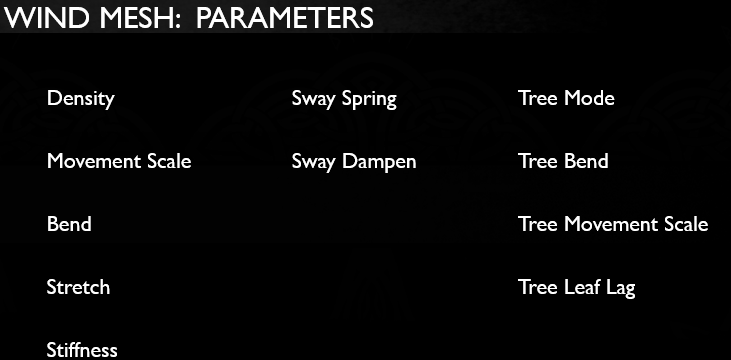


这五个组成部分如何相互关联？艺术作者为每种资产分别创作了两个。

1. 每顶点数据，定义模型的哪些部分移动，哪些部分不移动。工具可以帮助您。
2. 和模型参数，它们描述运动部件的行为。最好建立一个包含这些设置的库。
3. 这些都由技术方面的顶点着色器使用。这就是静态风力系统所需要的。
4. 对于动力学，我们为每个模型实例添加状态。顶点着色器使用它。
5. 我们有一个计算着色器，它捕获每帧的所有状态并更新它们。这主要涉及滚动噪声和平滑原始风信号。一些模型参数用于调整此更新行为。

我们使用四个浮点数来存储两个数据值。

1. 一个浮点数用于风遮罩。 这减弱了由于风引起的运动。通常是一个从零到一的梯度，代表到根顶点的归一化距离：不动的顶点。
2. 其他三个浮点存储该根顶点的相对位置。在此显示XYZ为RGB。我们将其存储为增量而不是绝对位置。这样就可以翻译艺术作品，而无需重新计算颜色（但不能缩放或旋转），并且它意味着数字更小，有助于精度。



模型参数分为三类：

1. 叶参数，用于控制模型的形状和造型。
2. 设置参数，这些参数直接控制多个帧上的更广泛的移动
3. 树参数。对于激活这些控件的模型，我们将第二层次结构定为形状，以对象根为中心，因此我们不需要其他顶点数据或角色技术的外观支持。

这是很多参数！可能会让人不知所措，因此我们谨慎地设置默认值。一个新的模型只需要改变其中的两个或三个，但是了解哪种仍然是艺术家的挑战。建立一个图书馆以便艺术家可以使用相似模型的设置作为起点非常有用。

Movement Scale是最自由的顶点从其静止位置可以移动的总距离。更高的值意味着更多的运动。我们通过风遮罩缩放此值。

垂直运动通过风矢量和噪声矢量的平均值进行。风矢量给我们倾斜，而噪声矢量给我们摆动。

Density控制这些噪声特征的比例。

实际上，我们允许此参数具有两个值，一个用于低速，一个用于高速。随着风速的增加，我们在这些设置之间进行融合，最高可达20-30 m / s。大多数模型对两者都使用相同的值，但是某些模型确实受益于这些单独的控制。

Bend控制元素倾斜时的平直度。

较高的值会将运动隔离到尖端。这是将风遮罩抬高到的指数。基本上是该灰度防风罩上的伽玛斜面。例如，值1表示直件;而值为2表示抛物线弯曲。

Stretch

我们通过平移顶点来获得运动。 [点击]我们不会像我看过某些游戏那样旋转它们。这样可以避免着色器中的三角函数变慢，但这可能会导致拉伸，因此我们需要保留长度。[click]为此，我们需要在移动之前检查到根的距离，[click]，然后朝根或远离根方向缩放以匹配该长度。这有助于防止地面穿透，并在极端作用力下为我们提供受阻旋转的外观。我们不必一直往后缩放，[单击]我们可以使用线性插值来仅校正小部分的拉伸。默认值是0.2.

Stiffness

我们可以通过将顶点纹理坐标朝向元素根部缩放来创建刚性元素的外观。这就像调整噪声密度，但保持效果集中在根部。较高的值可以使单个元素的变化较小，元素之间的差异很大。

Sway Spring

我们有两个摇摆参数可以调整对象的弹簧特性。它们是有状态的，因此在计算着色器中只执行一次计算，然后将结果传递到顶点着色器。前面我说过我们根据风向倾斜时，我撒了谎。它以这种方式开始，但实际上是通过摇摆矢量来倾斜顶点，该摇摆矢量被风力矢量推动并通过弹力拉回到中心。该弹簧的强度就是该摇摆弹簧的参数。

Sway Dampen

另一个摇摆参数控制该弹簧的阻尼，这会影响模型的建立时间。

Tree Mode

返回到顶点着色器。如果激活树模式，则会获得额外的运动层。树围绕物体根旋转。就像叶子运动一样，我们对树的运动进行额外的拉伸固定。

Tree Bend

树木还有一个附加的折弯参数。它的行为与叶子弯曲参数相同：归一化距离上的幂函数–伽玛曲线。笔直的行李箱很容易看到此处显示的1.0线性弯曲和3.0立方弯曲之间的差异。

Tree Movement Scale

树木运动尺度就像叶片运动尺度一样。这是在最高风速下由于树木倾斜而导致顶部顶点移动的距离。

我们的噪声函数是什么？

我们的噪声基础是在游戏启动时计算的三维纹理，其中填充了绑定在单位球体内的随机xyz向量。

\*\*\*我们在访问此纹理时使用三线性采样，因此基本函数很平滑，具有C0连续性。您可以使用任何喜欢的噪声基准，并且我们将要讨论的技术仍应适用。

我们通过使用分形噪声来扩展该基本函数。

分形噪声通常是不同尺度下多个噪声样本的加权和。这些缩放的样本称为八度音阶。最常见的方法是将每个八度音阶减小2的幂。

并且由于按比例缩小了功能，因此我们也按比例缩小了它们的贡献权重。

这是贡献曲线的粗略可视化。这是指数下降。

噪声的这种结合为我们提供了信号丰富的信号，并具有广泛的尺度范围。

通常，缩放噪声非常简单，只需缩放输入坐标即可!

但是如果缩放比例设置为动画会出现问题。我们的是，因为我们根据风速混合缩放值。

63页，有许多不太懂的地方