

# Vizard 引擎简要分析

T-SenseVR 虚拟现实实验室

WorldViz 中国独家授权技术服务中心

2016 年 10 月

## 目 录

一、Vizard 简介.....	3
二、引擎概述.....	5
三、Vizard 的特性.....	6
1、 丰富易用的硬件支撑体系.....	6
2、 内容开发与硬件底层松耦合.....	7
3、 便捷的集群功能.....	9
4、 架构合理使用灵活开放.....	10
5、 一致的开发习惯.....	13
6、 集中的代码管理.....	14
7、 易学易用授人以渔.....	14
四、引擎对比.....	17

## 一、Vizard 简介

VR 技术和其他技术一样，技术转化为产品并推广到市场上都有一定的适用性。VR 市场的一级区分为 B 端市场和 C 端市场。二级区分也就是垂直市场还可以做细分。

首先 VR 的 B 端市场，可以简单概括为行业应用，包含但不限于如下领域：教育科研、军工、工程建筑制造和设计、医疗、体育、市场营销等。Vizard 引擎 1996 年诞生于 MIT 心理学实验室，主要应用于心理学科学研究，后经多个版本迭代改进逐渐被商业化为成熟的 VR 引擎产品并且一直活跃在 B 端领域。Vizard 在 B 端市场的用户据不完全统计为 3000 多家，包含全球顶尖科研机构 and 世界 500 强企业。Vizard 在 B 端市场是涉及行业数量和覆盖面最广的产品。

### **Vizard 在全球的主要客户有：**

- ✓ 波音公司 (Boeing, USA)
- ✓ 洛克希德马丁 (Lockheed Martin, USA)
- ✓ 迪士尼 (Disney, USA)
- ✓ 博世 (The Bosch Group, Germany)
- ✓ 西门子 (Siemens, Germany)
- ✓ 罗克韦尔柯林斯国际公司 (Rockwell Collins, USA)
- ✓ 明尼苏达矿务及制造业公司 (3M, USA)
- ✓ 宝洁 (Procter & Gamble, USA)
- ✓ 罗氏制药 (Roche, Switzerland)
- ✓ 尼桑 (Nissan, Japan)
- ✓ 美国航空航天局 (NASA National Aeronautics and

Space Administration,USA)

✓ 欧洲宇航防务集团 (EADS European Aeronautic

Defence and Space Company)

✓ 欧洲航天局 (ESA European Space Agency)

✓ 北京大学

✓ 清华大学

✓ 南京大学

✓ 浙江大学

✓ 人民大学

✓ 中国科学院

✓ 纽约大学上海分校

✓ 天津天乐建筑设计公司

✓ 墨责国际文化发展有限公司

✓ 百度

✓ 中国东方航空公司



同时，Vizard 作为面向 B 端市场最为出色的专业 VR 引擎，连续多年参加 SIGGRAPH 大会并获奖。SIGGRAPH ( Special Interest Group for Computer GRAPHICS，计算机图形图像特别兴趣小组 ) 成立于 1967 年，一直致力于推广和发展计算机绘图和动画制作的软硬件技术，是世界上影响最广、规模最大，同时也是最权威的集科学、艺术、商业于一身的 CG 展示、学术研讨会。

但是，由于虚拟现实、增强现实等技术在近两年来才被更多中国用户所了解，加之 WorldViz 此前的市场方向主要偏重欧美地区，所以国内市场对于 Vizard 引擎知之甚少。随着 Facebook 收购 Oculus，VR/AR 迅速在国内火热起来，随之而来的自然是对引擎类产品的关注。

## 二、引擎概述

根据应用的领域不同，B 端市场常见的引擎类产品有 Vizard、ICIDO，VR 渲染工具（非引擎类）为 Techviz。C 端市场（主要领域为游戏、电影、社交等）常见的引擎类产品有 Unity、Unreal、CryEngine 等，它们基本都是游戏引擎，为游戏开发而生，为顺应 VR 技术发展趋势在各自引擎基础框架上增加 VR 支持模块或插件。两个市场具有不同的用户群体，因此引擎设计、功能性、营销策略和用户群体完全不同。概括来讲，引擎的两大指标：

（1）性能，即渲染画面效率和质量。在引擎技术和硬件技术快速发展的今天，各个引擎间的渲染能力差别已经不大。尤其 VR 引擎其设计出发点并不像游戏引擎那样追求画面渲染质量的极致，而是在画面渲染质量与渲染效率之间找到最佳平衡点，即实时渲染的效率一定要稳定保持在体验者可以接受的程度之上。

（2）对目标领域支撑是否全面。这主要分为对目标领域特性的支撑、对开

发者学习和使用习惯的支撑、对项目生命周期管理复杂度的支撑。

### 三、Vizard 的特性

Vizard 面向 B 端市场，以科研、教育和企业应用为主，在满足优秀画质的基础上确保极低的延迟和稳定的刷新率，给最终使用者良好的沉浸体验。与此同时针对 B 端市场特点和需求也有很好的支撑。

#### 1、丰富易用的硬件支撑体系

对于科研、教育和大型企业级应用，普遍使用市场主流设备和高端设备。例如：专业投影工程机、数据手套、工业级动作捕捉设备、数据采集设备、力反馈设备等。而 C 端用户普遍使用中低端设备和低价格设备。例如，移动端头盔显示器、手势识别、体感设备等。另外由于实验、教学及应用需求的变化，硬件设备需要经常新增或改变，这就需要引擎对硬件具有良好的兼容性和低成本的硬件管理解决方案。

Vizard 因 VR 而生，诞生之初就已经考虑多硬件支撑，其架构设计从第一个版本就是按照这个目标需求设计。Vizconnect 作为硬件管理层对引擎和开发者提供了统一的硬件界面，开发者只需关心 Vizconnect 的配置和调用方法即可，底层硬件对于开发者是透明的。Vizard 已经支持几乎市面上所有主流的 VR 硬件设备，VizConnect 提供了一个非常简单的，所见即所得的方式连接数量庞大的虚拟现实设备，这样你就不会仅限于使用某一款 VR 设备。

同时，Vizard 植入的多种立体化渲染模式也有助于 VR 设备的扩展，包括垂直/水平分屏显示模式，quad buffering 显示模式（适用于立体眼镜），扫描线

交错模式(line-interlaced)及红蓝立体成像模式 ( anaglyphic ) 等。

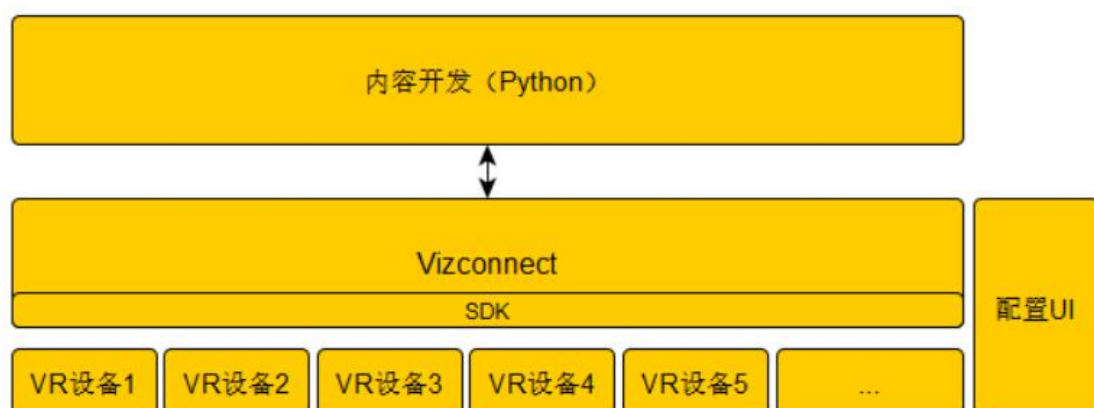
由于 Vizard 比其他引擎支撑更全面的 VR 硬件设备 ,要在极短的时间内完成一系列运算并反馈结果 ( 从周边输入设备 (三维空间追踪、动作捕捉、眼动仪、6DOF 手柄等)获取采集数据->运算获得场景中画面关系->渲染画面->反馈响应数据到终端输出设备 (HMD、CAVE、听觉设备、力回馈设备等)), 因此 Vizard 要有极高的运算效率和良好的架构体系。反观游戏引擎的设计出发点在于精美画面和跨平台产品发布,其架构设计上并未考虑 VR 应用的特殊性。

北京语言大学计算机系：

“选择 Vizard 的原因很简单，可以支持众多 VR 硬件。我们的教学目的之一就是要让学生了解真正的 VR 硬件并且可以互相协同使用。Vizard 引擎像是一款拥有异构虚拟现实环境粘合能力的胶水，将尚未标准化的 VR 设备联系到一起。”

## 2、内容开发与硬件底层松耦合

Vizconnect 作为硬件管理层对引擎和开发者提供了统一的硬件界面，这也有效的将内容开发和硬件使用隔离开，随意替换硬件而不需要重新修改内容代码。



(Vizard 中调用硬件示意图)

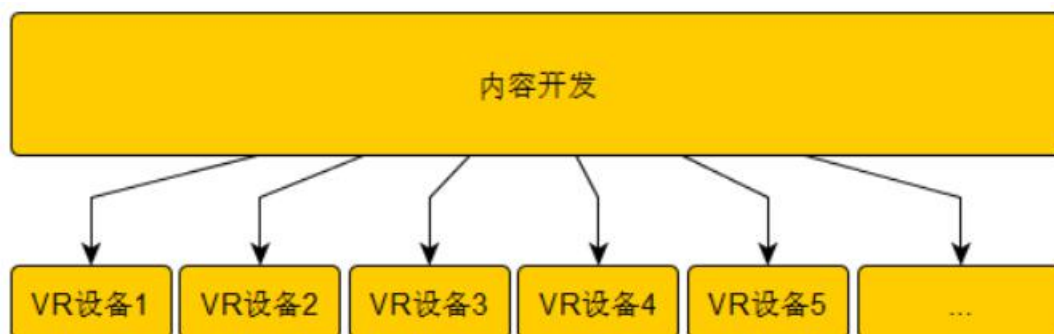
```

ArtShow_MX.py •
1  import viz
2  import vizact
3  import vizshape
4  import vizproximity
5  import vizconnect
6  import vizfx
7
8
9  SCALE = 1.1
10 FOOT_SCALE = 0.8*SCALE
11
12 vizconnect.go('vizconnect_keymouse.py')
13 #vizconnect.go('vizconnect_HTC_VIVE.py')
14 #vizconnect.go('vizconnect_ppt.py')
15 #vizconnect.go('vizconnect_ppt_NoWand.py')
16
17 gallery = vizfx.addChild('art/yanhuang.OSGB')
18 gallery.setScale([SCALE]*3)
19
20 view = vizconnect.getDisplay().getRaw().getView()
21 #viz.MainView.getHeadLight().disable()
22
23 ##选定作品交互=====
24 #春天里

```

该代码片段来自一个艺术类 VR 项目，红框圈定部分就是四组硬件配置：第一行是使用键盘和鼠标代替追踪设备，主要用于项目调试。第二行是使用 HTC VIVE 的头盔、手柄和追踪设备进行作品演示。第三行是使用 WorldViz 大面积主动光学追踪系统及 Oculus DK2 头盔进行作品演示。第四行同上，只是不需要手柄。切换不同硬件时只需要注释掉其他行即可。如此简便完全得益于 Vizconnect 统一的硬件接口。

游戏引擎怎么做呢？







（Unity 使用 HTC VIVE 设备）

由于没有统一的硬件界面，需要针对不同硬件设备进行特定的开发。首先需要导入对应设备的 SDK，然后将对应预设加载到场景中，针对该设备进行定制开发。如果设备更换，则需要重新导入新设备的 SDK 并且修改代码以适配新的设备。游戏制作基本只针对某几款特定 VR 设备开发，如果作为科研教育或企业应用，需要频繁变更设备，那维护成本将大大增加。

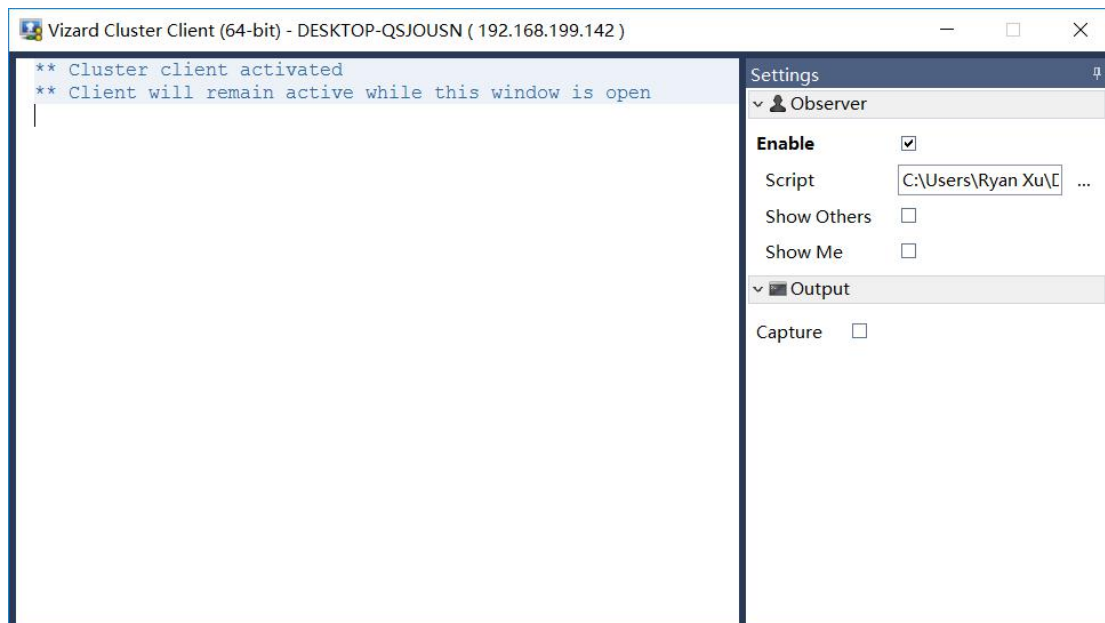
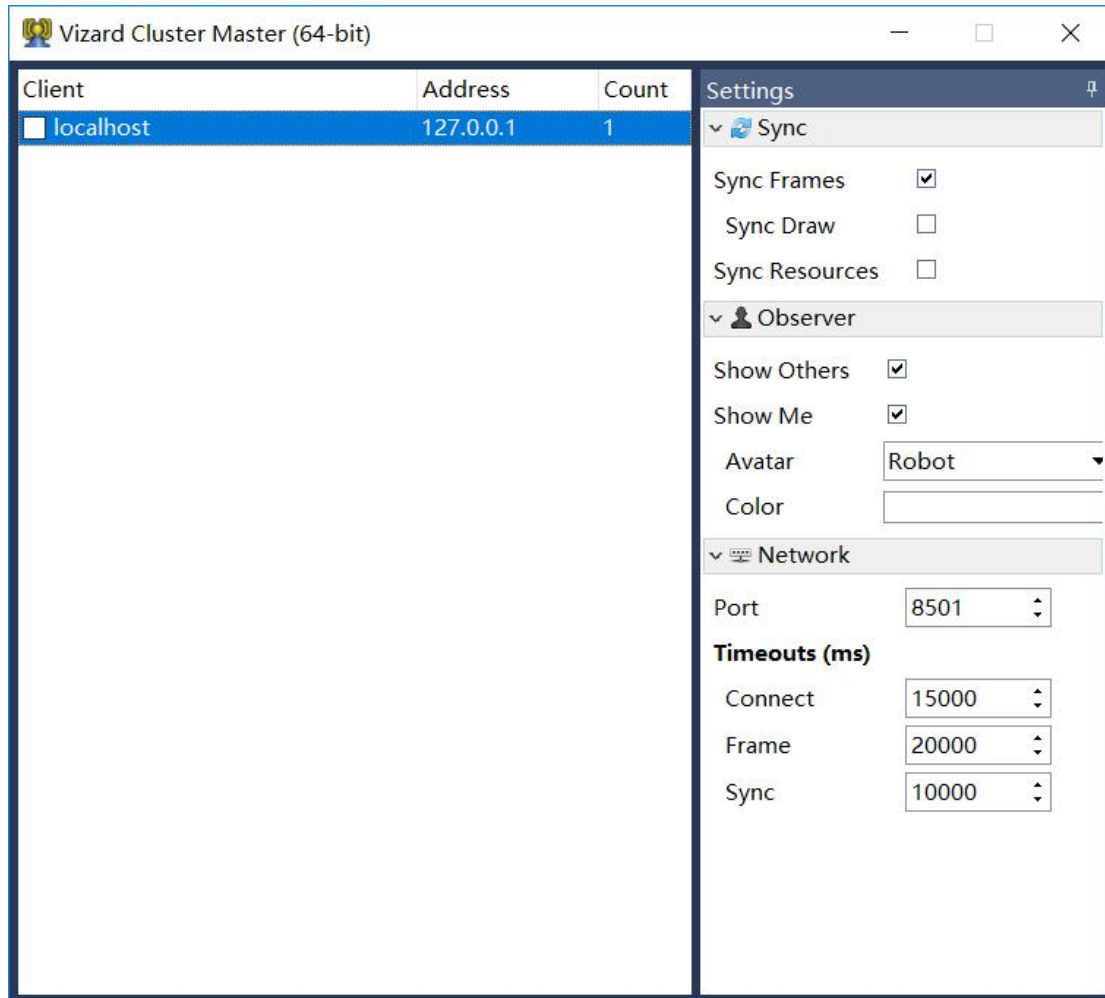
北京师范大学心理系：

“实验室选择 Vizard 的原因除了可以方便的和大面积精确追踪系统进行协同工作外，还可以随着实验目的的调整任意添加眼动仪等设备。”

### 3、便捷的集群功能

Vizard 提供的集群功能使得远程多人交互成为可能。不同地域的体验者可以同时进入同一个虚拟世界进行联机交互。实现这个功能只需要打开 Cluster 插件即可，不用做更多的编码工作即可保证各个终端渲染的同步。

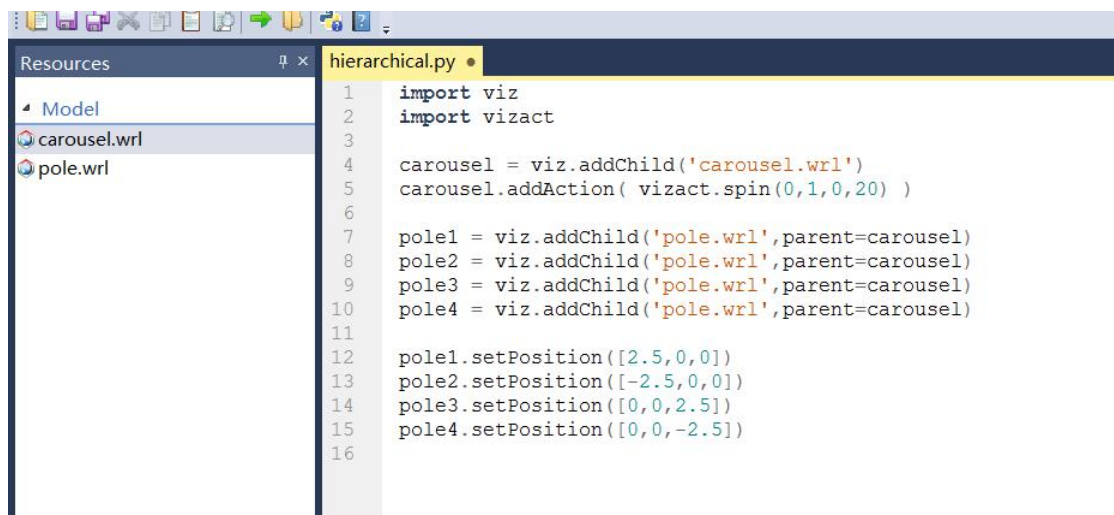
其他引擎若要实现该功能需要单独开发相应网络通信和同步渲染等功能。



#### 4、架构合理使用灵活开放

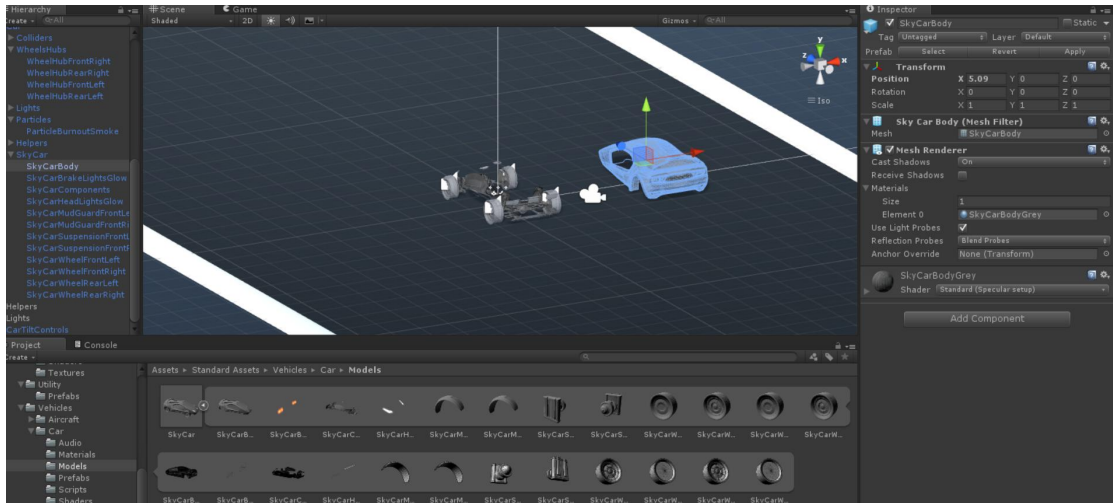
科研、教育和企业应用需要引擎具有足够的灵活度和开放性。一般的游戏引擎为了方便非开发成员或初学者都能够快速使用,会进行图形化操作的封装并且根据游戏开发的特性再进行更高层的横向封装。要求游戏引擎的使用者要参照其指定的 workflow 进行开发创作,在兼顾到非开发成员或初学者能够快速使用的同时也会牺牲引擎的灵活性和开放性。

Vizard 是标准的 IDE 集成开发环境,可以根据自己的习惯自由控制场景中的每一个事物。



```
1 import viz
2 import vizact
3
4 carousel = viz.addChild('carousel.wrl')
5 carousel.addAction( vizact.spin(0,1,0,20) )
6
7 pole1 = viz.addChild('pole.wrl',parent=carousel)
8 pole2 = viz.addChild('pole.wrl',parent=carousel)
9 pole3 = viz.addChild('pole.wrl',parent=carousel)
10 pole4 = viz.addChild('pole.wrl',parent=carousel)
11
12 pole1.setPosition([2.5,0,0])
13 pole2.setPosition([-2.5,0,0])
14 pole3.setPosition([0,0,2.5])
15 pole4.setPosition([0,0,-2.5])
16
```

如图显示 Vizard 引擎里通过极少的代码即可进行三维模型的加载、继承关系的设置以及模型上逻辑动作的控制。



如图显示 Unity 引擎需要按照其图形界面操作要求进行一系列设置。

Vizard 支持一切面向对象的编程思想，一切皆对象。在 Unity 中非常重要的 prefab 概念其实就是对象和实例的具象封装。开发者在使用 Vizard 进行开发时可以根据项目要求使用不同的设计模式来实现 prefab 的概念，同样也可以使用底层提供的 Copy 或 Clone 方法来实现。

再例如，Unity 所有的事件回调都塞进 MonoBehaviour，既不好看也不好。而 Vizard 的事件系统非常灵活，可以根据需要以多种形式编写在任何你想要出现的地方。

知乎社区 Unity 开发者：

*“Prefab 的机制弱爆了。无法嵌套，更新机制不统一。每次让人用 prefab 摆关卡时都提心吊胆，每次更新 prefab 都得全部关卡检查一遍。”*

知乎社区 Unity 开发者：

*“用的 mono 基于 .NET 3.5... 随着微软发布 VS2015 已经把 .NET 升级到 6.0 了，unity 官方自己也不知道怎么跟微软的步伐，跟不跟？还得看 Novell 那帮人对 mono 开源项目的*

步伐。”

知乎社区 Unreal 开发者：

“ 不得不说 UE4 的 Blueprint 要人性化的多，非常方便进行可视化编程。但是相对于真正的编程，比如 Unity 里面的 C#，还是感觉束手束脚，写一个条件判断都要拽上三四个控件连来连去，七八种颜色的 node 表示不同的变量类型也是醉了。同时，查询各种示例也很不方便，看到网上分享的代码，可以拿下来跑，看到分享的 Blueprint 截图？还得一个个对着连，这效率低到姥姥家去了。”

波音公司 Vizard 工程师：

“ 几乎找不到任何一款 VR 引擎能和 Vizard 媲美，很开放，可以做很多与机型研发相关的工作。”

江苏科技大学：

“ Vizard 的灵活开放，便于实验室进行各种实验项目开展，提高了科研效率，为论文发表提供良好的基础支持。”

## 5、一致的开发习惯

科研、教育和企业项目的参与者一般都具备编码能力，大多参与过 IT 项目研发，已经形成了面向对象的思维模式和开发习惯。Vizard 的操作语言 Python 是典型的面向对象语言，Vizard 只针对 VR 底层基础功能进行划分和封装（即提供基础功能的 API）并未规定特有的工作流程，开发人员可以像使用普通的 IDE（集成开发环境，例如 Visual Studio、Eclipse）进行工作而不需要为了使用一

个新的工具而转变固有的思维方式和开发习惯。

大部分游戏引擎为了便于理解和使用，在面向对象编程理论上结合图形化操作形成了独有的引擎操作流程，必须参照其特定操作流程进行内容开发。例如 Unity 使用 GameObject 和 Component 的概念，Unreal 使用蓝图的概念。

对于游戏领域从业者没有任何问题，但是对于科研、教育和企业项目的开发人员为了参与 VR 项目开发而改变已有的思维模式和开发习惯是很痛苦的事情。

## 6、集中的代码管理

游戏引擎基于组件的开发模式，很容易让开发者把渲染和逻辑代码混合到一起，让代码散落在各个角落，不便统一管理和调试。例如 Unity 场景中的 GameObject 都由多个组件构成，GameObject 的逻辑代码也被视为一个组件添加到 GameObject 上。这种方式导致代码分散在各个 GameObject 上，对于企业级大型项目会增加版本管理的难度。

Vizard 是一个标准的 IDE 开发环境，代码的组织完全可以根据项目需要进行架构设计，集中的代码管理极大避免了因为版本冲突导致的项目风险。

知乎社区 Unity 开发者：

*“Unity3d 是基于 3D 模型零件配合碎片化后的程序。注定支撑不了巨大的游戏环境，和最优的运行结果。说白了这个软件是艺术家的翅膀，是程序员的噩梦。”*

## 7、易学易用授人以渔

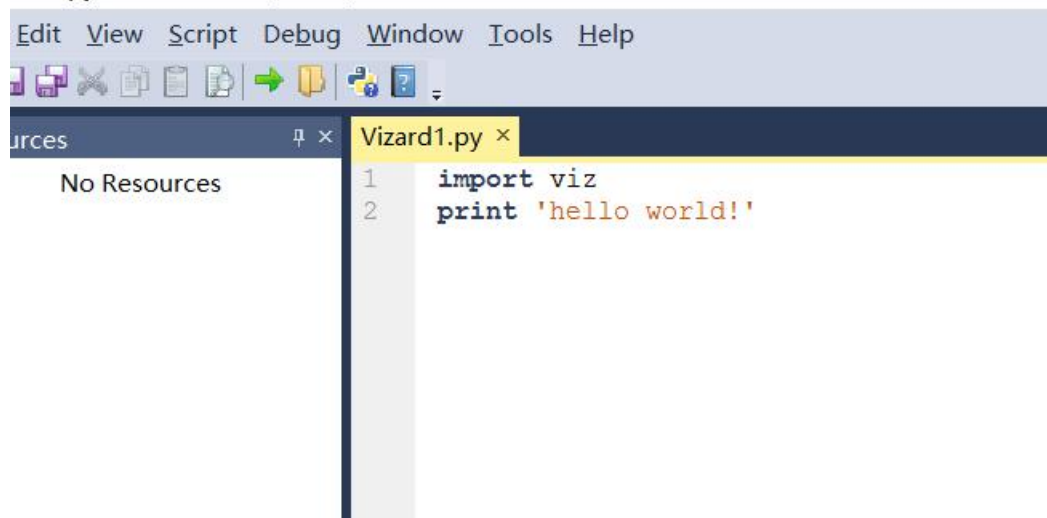
Vizard 的操作语言是 Python，相比其他引擎使用的 C++、C# 同样是标准的面向对象语言但其上手速度之快是其他开发语言无法比拟的。Python 语法简

洁易读,尤其针对非计算机领域的人员相对更容易理解,让人们更多精力关注在 VR 应用而非开发语言本身,因此特别适合科研和教学。Python 在国内的使用人群逐年提高,80%的高校已经开设 Python 课程。同时已经有很多第三方机构和开源社区针对 Python 进行 JIT( just in time,即时编译技术 )性能优化 ,Python 作为动态解释型语言的性能问题已经不再突显。

Vizard 的灵活性和开放性,让 VR 学习和使用更纯粹。入门学习阶段不需要花太多精力去了解特有的操作流程和界面使用方法,直接了解 VR 的运行原理并根据所学原理进行无束缚的开发;对于进阶学习阶段无论是 Vizard 还是具备图形化操作界面的其他引擎都不可避免的涉及深入的编码工作。

对比 Vizard 和 Unity 同样编写打印 “Hello World !”

ard1.py - Vizard 5 Lite (64-bit)

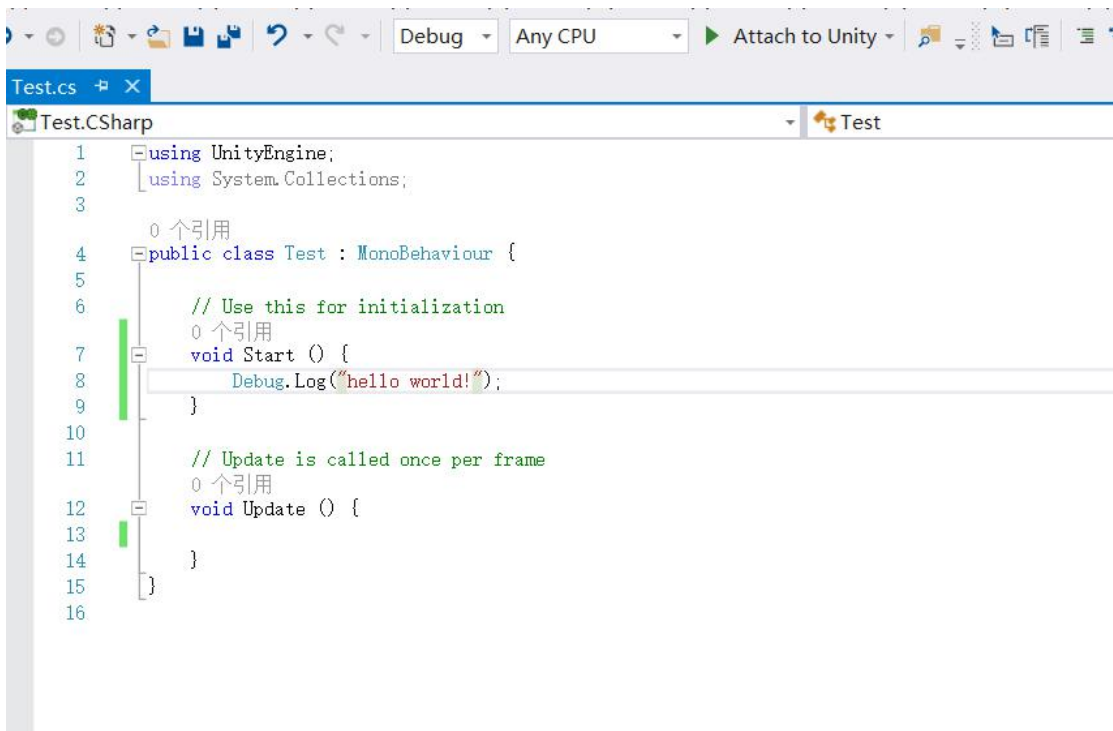
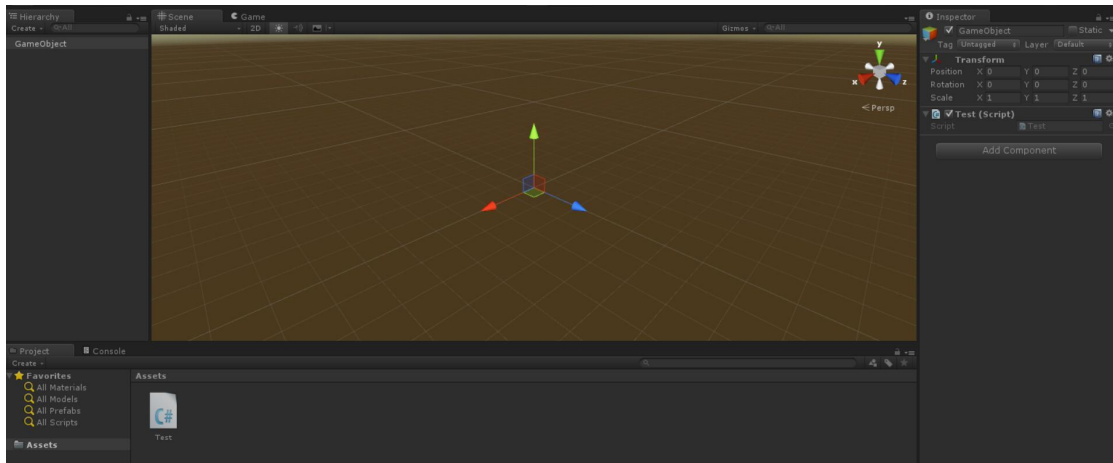


```
ard1.py - Vizard 5 Lite (64-bit)
Edit View Script Debug Window Tools Help
Vizard1.py x
1 import viz
2 print 'hello world!'
```

Vizard 只需两步:

- 1、新建文件
- 2、编写如上图代码
- 3、运行





Unity 引擎里需要开发者参照其制定的开发工作流程进行内容开发：

- 1、新建项目
- 2、创建新的 C#脚本
- 3、创建空的 GameObject
- 4、将脚本拖动到 GameObject 上
- 5、给脚本添加日志命令（如上图）
- 6、运行

另外，一些 VR 引擎更侧重于某几个领域，例如能源化工、房地产等等。把很多东西都封装隐藏了，针对该引擎学成之后只能使用该产品，迁移到其他工具基本思想就变了，不通用了。



不管从事 C 端游戏开发还是 B 端科学研究、企业级项目研发都需要具备良好的编码能力。以 Vizard 作为教学对象，使学员无论在入门阶段还是进阶阶段都学习到 VR 根本性的原理知识而非特定引擎的操作流程和界面使用方法，从而建立完善的知识体系。教育培训授人以鱼不如授人以渔。

北京语言大学实习生：

“因为有了 Vizard 的使用经验，再上手 Unity 入门速度很快，基本上花一个来小时了解 Unity 操作界面和一些特定概念后，就可以制作简单的 Demo。因为在 Vizard 使用过程中会潜移默化的让你了解计算机图形图像的原理及面向对象的编程思想，知道了原理，其他引擎的使用就很快了”

北京师范大学心理系：

“市场发展和科研教学需要，我们也开设了 VR 课程，以 Vizard 为依托是非常好的选择。市面上 Unity 和 Unreal 虽然占有率很高，但是 C++ 和 C# 对于计算机系的学生都很晦涩难懂，更别提我们心理系的学生了，Python 就显得容易很多。另外，Vizard 很灵活开放学生们只要学会简单的编程知识就可以上手操作，不用花费太多精力去学习诸如 Unity 或 Unreal 的操作界面，知道了原理，他们以后完全有能力自学那些引擎。再有，社会上的教学机构有很多开设 Unity、Unreal 的短培班，他们对于工具的培训已经有很成熟的方法，但是他们基本不太注重讲解原理，只告诉你应该这么做而不告诉你为什么。但是对于我们学校老师不太可能花费太多精力自学 Unity 和 Unreal 如何操作，再去教学生。正规学校还是要传授你为什么，和你所学的专业为什么能结合在一起，结合点在哪里”

#### 四、引擎对比

产品名称	介绍	价格	优势	不足
Vizard	专业虚拟现实引擎，可以 让开发者轻松创建虚拟现实应用。兼容众多 VR 硬件设备并且与内容开发松耦合。Vizard 内置的集成开发环境结合 Python 程序语言言让开发更加灵活。主要适用领域：科研、教学、企业级应用。	单机锁定，最高版本单机售价在 10 万人民币以内。在企业级 VR 引擎中性价比最高。  同时提供免费版和学员版供教学使用。	<ul style="list-style-type: none"><li>✓ Vizconnect 图形化界面可做硬件快速配置，硬件更换不影响主程序运行。</li><li>✓ 以 Python 为界面开发语言，开发效率高，复杂程序可维护性好。Python 自有模块和第三方模块也可同时使用，给开发者极大的开发自由度。</li><li>✓ 可支持丰富的 3D 文件格式，可获取丰富的网络 3D 模型资源。</li><li>✓ 提供功能强大的着色器模块。可做照片级别的高级渲染。</li><li>✓ 支持实时灯影。</li><li>✓ 强大的物理引擎可支持刚体动态模拟。</li><li>✓ 可发布 exe 文件做开发内容共享。</li><li>✓ 强大的集群功能，支持远程协同渲染，为 Vizard 独特优势。</li></ul>	<ul style="list-style-type: none"><li>✓ 无图形化开发界面面。</li><li>✓ 需要有编程基础。</li><li>✓ 粒子功能弱（粒子功能主要为游戏引擎的功能，在游戏开发中需要模拟大量烟雾等场景，但粒子渲染会消耗大量资源对于 VR 的实时渲染是个阻碍。游戏引擎开发 VR 场景也很少直接打开粒子功能。vizard 和其他引擎都是通过 3DsMax 的动态贴图或其他建模手段来间接实现）。</li><li>✓ 中文教材少（新版教材编写中，2017 年出版）。</li></ul>
Techviz	该产品是一种可将屏幕画面转为 3D 模式的驱动，不具二次开发性。该驱动仅从底层拦截图形渲染命令，将其转为 3D 模式。	单机锁定，单机软件售价 20 万人民币起。	<ul style="list-style-type: none"><li>✓ 可直接对接 3D 建模软件，将模型做 3D 呈现，流程简单，无无需编。</li></ul>	<ul style="list-style-type: none"><li>✓ 无法做内容开发，3D 呈现效率低。对于复杂模型无法满足实时渲染。</li><li>✓ 受限于主要 3D 软件的版本，若有新版本,Techviz 驱动则无法工作。考虑到软件更新速度越来越快，给 Techviz 的维护带来极大困难，也为用户的正常使用带来麻烦。</li></ul>
ICIDO	工程类、人机工程的专业虚拟现实软件。	单机锁定，单机软件售价 100 万人民币起。	对 CAD 模型支持好，流程简单。	<ul style="list-style-type: none"><li>✓ 画面渲染不够精致，底层优化不好。</li><li>✓ 软件非常封闭，只支持单一硬件设备。</li><li>✓ 价格太高。</li></ul>
Unity	游戏引擎，为迎合 VR 市场，在现有引擎基础上增添支持 VR 的功能模块或插件	功能受限的免费版；企业版按月订阅收费。游戏发布收取版权费用	<ul style="list-style-type: none"><li>✓ 图形化开发界面，虽然功能有限，但对初学者和非开发人员入门简单。</li><li>✓ 复杂的交互内容开发可用 C#做开发</li><li>✓ 有自己的资源商店，可获取各种插件和 3D 资源</li><li>✓ 粒子功能强大</li></ul>	<ul style="list-style-type: none"><li>✓ 由于其本身是游戏引擎，对 VR 硬件设备的支持不够丰富，网络上个人用户开发并分享的免费 plug-in 问题太多，商业插件价格昂贵且技术支持不完善。不支持多屏渲染的立体投影系统（工业界使用较多）</li><li>✓ Unity 的开发语言为 C#，入门比 Python 难。</li><li>✓ 无集群功能，需做复杂的二次开发。</li></ul>
Unreal	游戏引擎，为迎合 VR 市场，在现有引擎基础上增添支持 VR 的功能模块或插件	提供免费版。开发的内容做商业用途时，收取 5%的版权费。	<ul style="list-style-type: none"><li>✓ 渲染画面精致，大部分高质量的游戏都是由 Unreal 引擎制作。</li><li>✓ 占据 80%商业游戏市场份额</li></ul>	<ul style="list-style-type: none"><li>✓ C++较为晦涩难懂，学习曲线较陡，入门到精通的学习成本高。</li><li>✓ 蓝图的图形化界面虽然避免了代码开发，但对大型项目来说维护管理困难并且做深入的功能开发时</li></ul>

				避免不了代码编写 ✓ 作为游戏引擎同样存在 VR 硬件设备支撑不足的问题
--	--	--	--	---

20 年间，Vizard 针对 B 端市场需求不断进行产品优化，紧跟技术前沿。统一的架构体系避免了依赖第三方插件导致的项目运行不稳定，技术服务滞后。Vizard 可以提供统一的售后技术服务、使用培训和认证证书颁发。

引擎是一个大的系统，系统设计最重要的一环是控制和分配。图形学没什么算法是不公开的，关键是引擎的应用领域决定了牺牲哪些地方，注重哪些地方。Vizard 为 VR 而生适合 B 端市场，游戏引擎适合 C 端市场，互相跨界可能都会水土不服。