# N2引擎功能说明

Version 1.0.571 alpha

## 概述

N2引擎(英文名为N2Engine)旨在让任何想设计游戏的人容易并快速地生成可维护性和可拓展性强的2D游戏。目前这个引擎处于早期开发状态。

这是一个开源(Ms-PL)，免费的游戏引擎。目前它基于Win2D图形库，因此在Universal Windows平台上将可以得到广泛应用。

引擎源代码：

<https://github.com/Nukepayload2/Nukepayload2.Graphics.N2Engine>

## 开发门槛

正如概述所写的，这款引擎的入门门槛非常低。开发游戏的全程最消耗脑力的部分是用 Visual Basic编写代码。而这款语言正好是比较接近自然语言的，功能也像Visual C#一样强大。中国的大多数高中的学生生可以轻易自学这款语言，并且完成了UWP开发入门后能通过引擎的设计器和Visual Studio的Xaml设计器少写大量代码。由于本引擎使用Mvvm框架，只要用户模仿引擎的分类规则就能避免很多代码重构。

即使用户不想用开发环境，也能通过场景设计器设计新的拓展关卡（如果游戏适配这个功能）。

## 已设计的主要功能

#### 场景设计器(地图编辑器)

设计器用来生成和编辑你的游戏的各个关卡的内容。只要点击鼠标或触摸屏，拖拽控件，再编辑属性就可以构建游戏场景。你可以保存编辑进度(\*.json 文件)，然后把它发给好友或者传到你的其它设备再打开编辑。编辑完成后可产生程序代码文件(\*.vb)。添加生成的代码到你的游戏的解决方案即可将设计好的场景用到你自己的游戏中。

如果用了这个引擎的游戏对引擎的这个功能做了适配而且你无法使用开发环境，你可以直接解释执行已保存的编辑进度，并且可以让好友玩你设计的场景。你甚至可以把它发布到互联网，让网友也尝试你的场景。但是这样会损失一些灵活性。

#### 刚体碰撞演算

我们制作了Box2D的VB版本实现，因此刚体碰撞直接用引擎中包含的Box2D即可。

这个功能在处理碰撞，爆炸，人物跳跃等功能时很有帮助。

#### 粒子系统

允许用户使用和自定义 火焰，烟雾，轨道炮，火花等粒子系统。有的使用贴图，有的即时演算。

#### 剧情系统和损伤系统

对于大多数游戏，剧情系统和损伤系统是必不可少的。剧情系统能让你更快地写出主线任务，奖励和支线任务。损伤系统则能让玩家损坏场景和伤害怪物, 或者是让玩家操控的人物受伤。

#### 平面光照

我们的引擎支持全局光照，点光照这两种光照。平行光照请用点光照代替。

点光照是带有阴影形状计算功能的。

#### 多媒体

我们的引擎与Xaml深度集成，所以声音和视频都交给Xaml处理。

#### 针对特定游戏类型的拓展库

背包系统，复杂武器系统，寻路系统 等 逻辑是针对特定游戏的。因此在我们的项目拓宽后我们会推出针对特定游戏拓展库。

#### 基础的碰撞检测

提供点击，重叠等几何体碰撞的通知。用于自定义设计器和触发某些事件（比如 遭到攻击, 走进了某个区域）。

#### 存档和读档

我们的引擎暂定使用SQLite数据库进行存档和读档。如果它在ARM处理器有异常行为，则改用json。

#### 场景虚拟化

超出视野范围的场景可以不被绘制。必要时可以阻止它们更新状态。

#### 小地图和滚动视图

充分利用Xaml的控件，我们的引擎可以像RA2一样用小地图和卷动画面，可以像RA3一样放大和缩小画面。

## 框架概述

我们的框架基于Mvvm，并且针对游戏设计的几种情况对Mvvm进行了细微的改变。

Model 层 包含游戏物体的数据模型和相关的逻辑。游戏物体保留了与它关联的视图的引用，因为游戏物体的数据模型和它们的视图几乎是一一对应的。

ViewModel 层 充当地图或沙盘的作用，主要内容是游戏物体和Xaml控件的集合。

View 层 用于呈现游戏内容，响应用户输入。游戏内容的视图有对游戏内容数据模型的引用。

游离在框架之外的是Utilities。这部分代码不仅游戏编程会用到，其它程序编写也会用到。

## 存在的问题

我们在开发维护效率和运行效率中选择了前者。因此这个引擎的游戏运行的效率会低于传统的Direct 2D方式编写的游戏（但性能通常比用GDI+或JPanel的游戏好很多）。为了提升运行效率，我把引擎做成了共享项目，而不是Windows运行时组件。这样可以让.NET Native的优化能力得到充分发挥。