

热更新

- 虚幻引擎交互设计师班 -



火星时代教育

概念分析

- 虚幻引擎交互设计师班 -



火星时代教育

更新目的

更新的主要目的是为了了解决产品包发布后，部分设计存在缺陷或是需要进行升级，通过使用更新技术，将新的内容应用到发布产品上，这个过程也被称作**打补丁**。

软件更新方式

1. **全量更新**：是指软件在更新版本时，需要卸载旧版本，下载完整的新版后才能完成更新。一般当软件发生重大变更时会使用全量更新。
2. **增量更新**：是指软件在更新新的版本时，通过将软件进行二进制比较，将变化部分的数据进行更新，从而达到只更新增加部分的效果。注意，一般增量更新应设计软件版本变更比较，并在用户端设计版本合并。

注意：以上谈到的是软件更新方式，其中增量更新对于用户来说体验最佳。

热更新

什么是热更新？

一般热更新是指在软件**运行过程中**更新部分逻辑，并可直接被应用到当次逻辑中。这样的更新方式称之为热更新。简而言之，即在不关闭应用程序的前提下，更新应用程序逻辑，并被加载。

在手机游戏产品开发中，热更新一直被做为必备模块而存在。主要由于手机端软件更新成本高。例如苹果端，每次做产品更新都需要重新提交审核，审核通过，用户还需要知晓更新才会真的完成更新。但是对于一些产品，可能更新是为了立即解决设计问题，如果走正常的更新流程，就非常耗时。热更新可以很好的解决这一问题。

虚幻引擎热更新

- 虚幻引擎交互设计师班 -



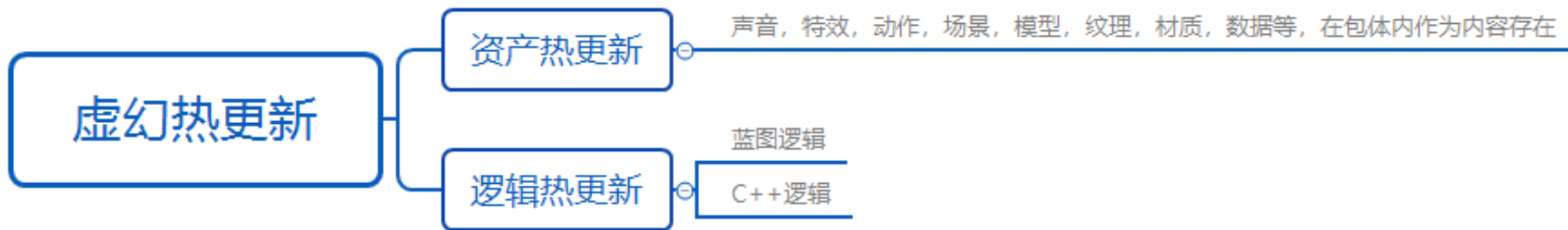
火星时代教育

热更新内容

在虚幻引擎中，需要进行热更新的内容大致分为两种：资产热更新和逻辑热更新。

资产热更新：比较好理解，例如修改模型，贴图，声音，特效，动作等。在不影响用户体验的前提下，完成更新。

逻辑热更新：大致分为两种蓝图逻辑和C++逻辑。



资产更新

- 虚幻引擎交互设计师班 -



火星时代教育

资产补丁

更新过程中，必然需要将新的资产进行拆分，然后将变更资产 进行标记，用作后期更新时使用。这样的资产我们称之为**补丁**。将资产应用到用户端的过程，我们称之为**打补丁**。

对于补丁的创建，我们建议使用创建**与平台无关的补丁数据**，这样对于多平台开发更加方便。不用为每个平台特意创建资产补丁。

添加资产ID

更新过程中，必然需要将新的资产进行拆分，然后将变更资产 进行标记，用作后期更新时使用。这样的资产我们称之为**补丁**。将资产应用到用户端的过程，我们称之为**打补丁**。

对于补丁的创建，我们建议使用创建**与平台无关的补丁数据**，这样对于多平台开发更加方便。不用为每个平台特意创建资产补丁。



Puerts

- 虚幻引擎交互设计师班 -



火星时代教育

Puerts



Puerts是由腾讯设计并开源维护，是游戏引擎下的TypeScript编程解决方案，是允许在运行时态通过使用TS访问宿主引擎的能力。通过使用TS完成业务逻辑设计。

项目开源地址：<https://github.com/Tencent/puerts/>

为什么选用TS

- 语法结构简单易懂，尤其从C++过渡学习成本低
- 带有面向对象编程特性，支持继承
- 带有接口特性，对于抽象行为，设计松耦合非常友好
- 带有类型批注与类型检查，使得编码过程中杜绝类型错误问题，更容易编写健壮的代码
- 编辑器友好，借助VS Code插件可以高效进行生产
- 支持泛型，更容易抽象针对类型设计的程序结构
- 对大型项目构建友好，对于设计模式实施成本较低
- 产品的迭代和后期重构困难度较低


Puerts优势

UnLua可以做什么？

产品开发过程中的逻辑代码会被编译成运行文件，如果过在移动端会被编译到给定app格式文件。如果我们需要修改逻辑就变得十分麻烦，必须要重新编译打包，并要求用户更换新的软件包。Unlua的好处在于，它允许你将一部分逻辑写道Lua中，由于Lua不会被编译，所以可以通过设计，将Lua代码在远端更新。客户端只需要下载新的Lua文件即可应用新的设计逻辑。

Unlua是一套虚幻引擎使用Lua语言做为开发语言的解决方案。



Mars游戏星球 



游戏学院公众号

(教程 + 科普 + 就业 + 优秀作品 + 行业资讯)



游戏学院视频号

(教程 + 优秀作品 + 行业资讯 + 学院风采)



花瓣网账号

(优秀作品 + 知识点长图)

Thanks

-火星时代游戏设计学院-