



(12) 发明专利

(10) 授权公告号 CN 111435303 B

(45) 授权公告日 2023.05.23

(21) 申请号 201910034649.8

CN 106293753 A, 2017.01.04

(22) 申请日 2019.01.15

US 7802243 B1, 2010.09.21

(65) 同一申请的已公布的文献号

申请公布号 CN 111435303 A

C-h-h. “Inno setup 简单打包教程”.

《<https://blog.csdn.net/chh19941125/article/details/47777785>》. 2015, 第1-6页.

(43) 申请公布日 2020.07.21

流波上的舞. “unity3d进程通信利用WM_COPYDARE和HOOK”. 《<https://www.cnblogs.com/sbCat/p/5257521.html>》. 2016, 第1-5页.

(73) 专利权人 福建省天奕网络科技有限公司

地址 350212 福建省福州市长乐区湖南镇大鹤村

军神关云长. “innosetup的静默安装与卸载”. 《<https://blog.csdn.net/qqwangfan/article/details/70159147>》. 2017, 第1页.

(72) 发明人 刘德建 周少毅 林琛

塞北狼烟. “Unity3d发布EXE安装程序”.

(74) 专利代理机构 福州市博深专利事务所(普通合伙) 35214

专利代理师 张明 朱立云

《https://blog.csdn.net/qq_28775437/article/details/79541671》. 2018, 第1-4页.

(51) Int. Cl.

G06F 8/61 (2018.01)

“用User32.dll把UNITY程序的窗口透明化”. 月光水岸. 《https://blog.sina.com.cn/s/blog_c0e6ab9b0102wn2s.html》. 2016, 第1-4页.

(56) 对比文件

CN 106295370 A, 2017.01.04

CN 108897562 A, 2018.11.27

审查员 王为鑫

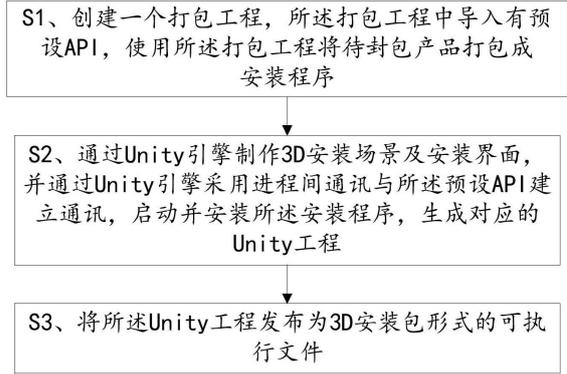
权利要求书1页 说明书5页 附图1页

(54) 发明名称

一种生成3D安装包的方法及终端

(57) 摘要

本发明公开了一种生成3D安装包的方法及终端,通过在创建的打包工程中导入预设API,使用所述打包工程将待封包产品打包成安装程序,通过Unity引擎制作3D安装场景及安装界面,并通过Unity引擎采用进程间通讯与所述预设API建立通讯,启动并安装所述安装程序,生成对应的Unity工程,将所述Unity工程发布为3D安装包形式的可执行文件,可以以较小的代价和较高的效率生成一种高压缩、稳定的3D安装包,高效地实现3D安装包的开发过程,提高工作效率,在所述3D安装包的开发过程中可以通过通讯及时了解情况,方便使用者操作。



1. 一种生成3D安装包的方法,其特征在于,包括步骤:

S1、创建一个打包工程,所述打包工程中导入有预设API,使用所述打包工程将待封包产品打包成安装程序;具体的,在所述打包工程中导入Windows系统的User32.dll的API SendMessage方法,所述SendMessage方法用于与Unity通讯;

S2、通过Unity引擎制作3D安装场景及安装界面,并通过Unity引擎采用进程间通讯与所述预设API建立通讯,启动并安装所述安装程序,生成对应的Unity工程;

所述步骤S2中所述通过Unity引擎采用进程间通讯与所述预设API建立通讯,启动并安装所述安装程序包括:

通过Unity引擎以命令行方式开启所述安装程序,并向所述安装程序传入静默安装命令行参数及安装路径参数以实现所述安装程序的安装;

在所述安装程序的安装过程中通过Unity接收并显示所述预设API发送的安装进度;

S3、将所述Unity工程发布为3D安装包形式的可执行文件;

所述步骤S3包括:

将所述Unity工程发布为一个3D安装包形式的可执行文件和一个data文件夹,采用封包工具将所述data文件夹封包进所述3D安装包形式的可执行文件中;

所述步骤S3还包括:

采用所述预设API将所述可执行文件的窗口去除,并利用Shader将所述可执行文件的纯色背景过滤。

2. 一种生成3D安装包的终端,包括存储器、处理器及存储在存储器上并可在所述处理器上运行的计算机程序,其特征在于,所述处理器执行所述计算机程序时实现以下步骤:

S1、创建一个打包工程,所述打包工程中导入有预设API,使用所述打包工程将待封包产品打包成安装程序;具体的,在所述打包工程中导入Windows系统的User32.dll的API SendMessage方法,所述SendMessage方法用于与Unity通讯;

S2、通过Unity引擎制作3D安装场景及安装界面,并通过Unity引擎采用进程间通讯与所述预设API建立通讯,启动并安装所述安装程序,生成对应的Unity工程;

所述步骤S2中所述通过Unity引擎采用进程间通讯与所述预设API建立通讯,启动并安装所述安装程序包括:

通过Unity引擎以命令行方式开启所述安装程序,并向所述安装程序传入静默安装命令行参数及安装路径参数以实现所述安装程序的安装;

在所述安装程序的安装过程中通过Unity接收并显示所述预设API发送的安装进度;

S3、将所述Unity工程发布为3D安装包形式的可执行文件;

所述步骤S3包括:

将所述Unity工程发布为一个3D安装包形式的可执行文件和一个data文件夹,采用封包工具将所述data文件夹封包进所述3D安装包形式的可执行文件中;

所述步骤S3还包括:

采用所述预设API将所述可执行文件的窗口去除,并利用Shader将所述可执行文件的纯色背景过滤。

一种生成3D安装包的方法及终端

技术领域

[0001] 本发明涉及产品封包技术领域,尤其是涉及一种生成3D安装包的方法及终端。

背景技术

[0002] 目前市场上基本没有使用3D安装包封包的产品,若用现有技术制作3D安装包,通常会使用Unity的3D引擎直接开发,封包方面可借助ZIP等库,而在常用功能方面,例如读写注册表、创建快捷方式和在线更新等功能则需要自行研发,功能完备后还需要解决Windows系统不同版本的兼容性问题。这时,开发一个3D安装包需要耗费大量的开发时间及精力,而且极其容易出现兼容性问题。

发明内容

[0003] 本发明所要解决的技术问题是:提供一种生成3D安装包的方法及终端,能够高效地实现3D安装包开发过程,提高工作效率。

[0004] 为了解决上述技术问题,本发明采用的一种技术方案为:

[0005] 一种生成3D安装包的方法,包括步骤:

[0006] S1、创建一个打包工程,所述打包工程中导入有预设API,使用所述打包工程将待封包产品打包成安装程序;

[0007] S2、通过Unity引擎制作3D安装场景及安装界面,并通过Unity引擎采用进程间通讯与所述预设API建立通讯,启动并安装所述安装程序,生成对应的Unity工程;

[0008] S3、将所述Unity工程发布为3D安装包形式的可执行文件。

[0009] 为了解决上述技术问题,本发明采用的另一种技术方案为:

[0010] 一种生成3D安装包的终端,包括存储器、处理器及存储在存储器上并可在所述处理器上运行的计算机程序,所述处理器执行所述计算机程序时实现以下步骤:

[0011] S1、创建一个打包工程,所述打包工程中导入有预设API,使用所述打包工程将待封包产品打包成安装程序;

[0012] S2、通过Unity引擎制作3D安装场景及安装界面,并通过Unity引擎采用进程间通讯与所述预设API建立通讯,启动并安装所述安装程序,生成对应的Unity工程;

[0013] S3、将所述Unity工程发布为3D安装包形式的可执行文件。

[0014] 本发明的有益效果在于:通过在创建的打包工程中导入预设API,使用所述打包工程将待封包产品打包成安装程序,通过Unity引擎制作3D安装场景及安装界面,并通过Unity引擎采用进程间通讯与所述预设API建立通讯,启动并安装所述安装程序,生成对应的Unity工程,将所述Unity工程发布为3D安装包形式的可执行文件,可以以较小的代价和较高的效率生成一种高压缩、稳定的3D安装包,高效地实现3D安装包的开发过程,提高工作效率,在所述3D安装包的开发过程中可以通过通讯及时了解情况,方便使用者操作。

附图说明

[0015] 图1为本发明实施例的一种生成3D安装包的方法的步骤流程图；

[0016] 图2为本发明实施例的一种生成3D安装包的终端的结构示意图；

[0017] 标号说明：

[0018] 1、一种生成3D安装包的终端； 2、存储器； 3、处理器。

具体实施方式

[0019] 为详细说明本发明的技术内容、所实现目的及效果，以下结合实施方式并配合附图予以说明。

[0020] 本发明最关键的构思在于：在创建的打包工程中导入预设API，并通过Unity引擎采用进程间通讯与所述预设API建立通讯。

[0021] 请参照图1，一种生成3D安装包的方法，包括步骤：

[0022] S1、创建一个打包工程，所述打包工程中导入有预设API，使用所述打包工程将待封包产品打包成安装程序；

[0023] S2、通过Unity引擎制作3D安装场景及安装界面，并通过Unity引擎采用进程间通讯与所述预设API建立通讯，启动并安装所述安装程序，生成对应的Unity工程；

[0024] S3、将所述Unity工程发布为3D安装包形式的可执行文件。

[0025] 从上述描述可知，本发明的有益效果在于：通过在创建的打包工程中导入预设API，使用所述打包工程将待封包产品打包成安装程序，通过Unity引擎制作3D安装场景及安装界面，并通过Unity引擎采用进程间通讯与所述预设API建立通讯，启动并安装所述安装程序，生成对应的Unity工程，将所述Unity工程发布为3D安装包形式的可执行文件，可以以较小的代价和较高的效率生成一种高压缩、稳定的3D安装包，高效地实现3D安装包的开发过程，提高工作效率，在所述3D安装包的开发过程中可以通过通讯及时了解情况，方便使用者操作。

[0026] 进一步的，所述预设API包括Windows系统的User32.dll的API。

[0027] 由上述描述可知，所述预设API包括Windows系统的User32.dll的API，可以为后续生成所述3D安装包的各类操作提供相应的API，方便与所述Unity引擎进行通讯，提高便利性。

[0028] 进一步的，所述步骤S2中所述通过Unity引擎采用进程间通讯与所述预设API建立通讯，启动并安装所述安装程序包括：

[0029] 通过Unity引擎以命令行方式开启所述安装程序，并向所述安装程序传入静默安装命令行参数及安装路径参数以实现所述安装程序的安装；

[0030] 在所述安装程序的安装过程中通过Unity接收并显示所述预设API发送的安装进度。

[0031] 由上述描述可知，通过所述Unity引擎控制所述安装程序的安装，可以方便使用者灵活改变所述程序的安装路径等相关参数，方便使用者操作，同时，通过所述Unity引擎与所述API的通讯可以及时将安装进度显示以供使用者查看，提高便利性。

[0032] 进一步的，所述步骤S3包括：

[0033] 将所述Unity工程发布为一个3D安装包形式的可执行文件和一个data文件夹，采

用封包工具将所述data文件夹封包进所述3D安装包形式的可执行文件中。

[0034] 由上述描述可知,采用封包工具将所述data文件夹封包进所述3D安装包形式的可执行文件中,可以尽可能地压缩数据,大大减少生成的所述3D安装包的占用空间。

[0035] 进一步的,所述步骤S3还包括:

[0036] 采用所述预设API将所述可执行文件的窗口去除,并利用Shader将所述可执行文件的纯色背景过滤。

[0037] 由上述描述可知,通过所述预设的API可以方便将所述可执行文件的窗口去除,通过Unity自带的Shader可以方便将所述可执行文件的纯色背景过滤,方便快捷。

[0038] 请参照图2,一种生成3D安装包的终端,包括存储器、处理器及存储在存储器上并可在所述处理器上运行的计算机程序,所述处理器执行所述计算机程序时实现以下步骤:

[0039] S1、创建一个打包工程,所述打包工程中导入有预设API,使用所述打包工程将待封包产品打包成安装程序;

[0040] S2、通过Unity引擎制作3D安装场景及安装界面,并通过Unity引擎采用进程间通讯与所述预设API建立通讯,启动并安装所述安装程序,生成对应的Unity工程;

[0041] S3、将所述Unity工程发布为3D安装包形式的可执行文件。

[0042] 从上述描述可知,本发明的有益效果在于:通过在创建的打包工程中导入预设API,使用所述打包工程将待封包产品打包成安装程序,通过Unity引擎制作3D安装场景及安装界面,并通过Unity引擎采用进程间通讯与所述预设API建立通讯,启动并安装所述安装程序,生成对应的Unity工程,将所述Unity工程发布为3D安装包形式的可执行文件,可以以较小的代价和较高的效率生成一种高压缩、稳定的3D安装包,高效地实现3D安装包的开发过程,提高工作效率,在所述3D安装包的开发过程中可以通过通讯及时了解情况,方便使用者操作。

[0043] 进一步的,所述预设API包括Windows系统的User32.dll的API。

[0044] 由上述描述可知,所述预设API包括Windows系统的User32.dll的API,可以为后续生成所述3D安装包各类操作提供相应的API,方便与所述Unity引擎进行通讯,提高便利性。

[0045] 进一步的,所述步骤S2中所述通过Unity引擎采用进程间通讯与所述预设API建立通讯,启动并安装所述安装程序包括:

[0046] 通过Unity引擎以命令行方式开启所述安装程序,并向所述安装程序传入静默安装命令行参数及安装路径参数以实现所述安装程序的安装;

[0047] 在所述安装程序的安装过程中通过Unity接收并显示所述预设API发送的安装进度。

[0048] 由上述描述可知,通过所述Unity引擎控制所述安装程序的安装,可以方便使用者灵活改变所述程序的安装路径等及相关参数,方便使用者操作,同时,通过所述Unity引擎与所述API的通讯可以及时将安装进度显示以供使用者查看,提高便利性。

[0049] 进一步的,所述步骤S3包括:

[0050] 将所述Unity工程发布为一个3D安装包形式的可执行文件和一个data文件夹,采用封包工具将所述data文件夹封包进所述3D安装包形式的可执行文件中。

[0051] 由上述描述可知,采用封包工具将所述data文件夹封包进所述3D安装包形式的可

执行文件中,可以尽可能地压缩数据,大大减少生成的所述3D安装包的占用空间。

[0052] 进一步的,所述步骤S3还包括:

[0053] 采用所述预设API将所述可执行文件的窗口去除,并利用Shader将所述可执行文件的纯色背景过滤。

[0054] 由上述描述可知,通过所述预设的API可以方便将所述可执行文件的窗口去除,通过Unity自带的Shader可以方便将所述可执行文件的纯色背景过滤,方便快捷。

[0055] 实施例一

[0056] 请参照图1,一种生成3D安装包的方法,包括步骤:

[0057] S1、创建一个打包工程,所述打包工程中导入有预设API,使用所述打包工程将待封包产品打包成安装程序;

[0058] 其中,所述预设API包括Windows系统的User32.dll的API;

[0059] 具体地,使用Inno Setup软件新建一个打包工程,在所述打包工程中导入所述Windows系统的User32.dll的API——SendMessage,所述SendMessage方法用于与Unity通讯,相关代码为: `function SendMessageWin (Wnd:HWND;Msg:Integer;wParam:Integer;IParam:Longint):Longint;external 'SetMessage@user32.dll stdcall'`,使用Inno Setup安装包软件将待封包产品打包成exe安装程序;

[0060] S2、通过Unity引擎制作3D安装场景及安装界面,并通过Unity引擎采用进程间通讯与所述预设API建立通讯,启动并安装所述安装程序,生成对应的Unity工程;

[0061] 具体地,使用所述Unity引擎制作绚丽的3D安装场景以及安装界面;

[0062] 通过Unity引擎以命令行方式开启所述exe安装程序,并向所述exe安装程序传入静默安装命令行参数及安装路径参数,所述exe安装程序根据所述Unity引擎传入的参数及安装路径执行相应的安装流程以实现所述exe安装程序的安装,在所述exe安装程序的安装过程中通过Unity接收并显示所述预设API中的SendMessage方法发送的安装进度;

[0063] 其中所述静默安装相关命令行包括 `"/verysilent"`、`"/sp-"`、`"/norestart"`、`"/nocancel"`、`"/suppressmsgboxes"`、`"/forcecloseapplications"`和 `"/c"`等等;

[0064] S3、将所述Unity工程发布为3D安装包形式的可执行文件。

[0065] 具体地,将所述Unity工程发布为一个3D安装包形式的exe可执行文件和一个data文件夹,采用封包工具将所述data文件夹封包进所述3D安装包形式的exe可执行文件中;

[0066] 具体地,安装包通常只包括一个exe可执行文件,因此利用WinRAR等封包工具将所述data文件夹封包进所述3D安装包形式的exe可执行文件中,同时减少所述3D安装包的占用空间;

[0067] 采用所述预设API将所述可执行文件的窗口去除,并利用Shader将所述可执行文件的纯色背景过滤;

[0068] 具体地,由于所述Unity工程被发布后带有窗口,需要将所述窗口去除并透明化背景,这时利用所述预设API中的 `SetLayeredWindowAttributes (IntPtr hwnd,uint crKey,byte bAlpha,int dwFlags)` 方法将所述窗口去除,利用Unity的Shader将纯色背景过滤。

[0069] 实施例二

[0070] 请参照图2,一种生成3D安装包的终端1,包括存储器2、处理器3及存储在存储器2上并可在所述处理器3上运行的计算机程序,所述处理器3执行所述计算机程序时实现实施

例一中的各个步骤。

[0071] 综上所述,本发明提供了一种生成3D安装包的方法及终端,通过在创建的打包工程中导入预设API,所述预设API包括Windows系统的User32.dll的API,使用所述打包工程将待封包产品打包成安装程序,通过Unity引擎制作3D安装场景及安装界面,并通过Unity引擎采用进程间通讯与所述预设API建立通讯,启动并安装所述安装程序,生成对应的Unity工程,在所述安装程序的安装过程中通过Unity接收并显示所述预设API发送的安装进度,将所述Unity工程发布为3D安装包形式的可执行文件,可以为生成所述3D安装包的各类操作提供相应的API,方便与所述Unity引擎进行通讯,以较小的代价和较高的效率生成一种高压缩、稳定的3D安装包,高效地实现3D安装包的开发过程,提高工作效率,在所述3D安装包的开发过程中可以通过通讯及时了解情况,方便使用者操作,提高便利性。

[0072] 以上所述仅为本发明的实施例,并非因此限制本发明的专利范围,凡是利用本发明说明书及附图内容所作的等同变换,或直接或间接运用在相关的技术领域,均同理包括在本发明的专利保护范围内。

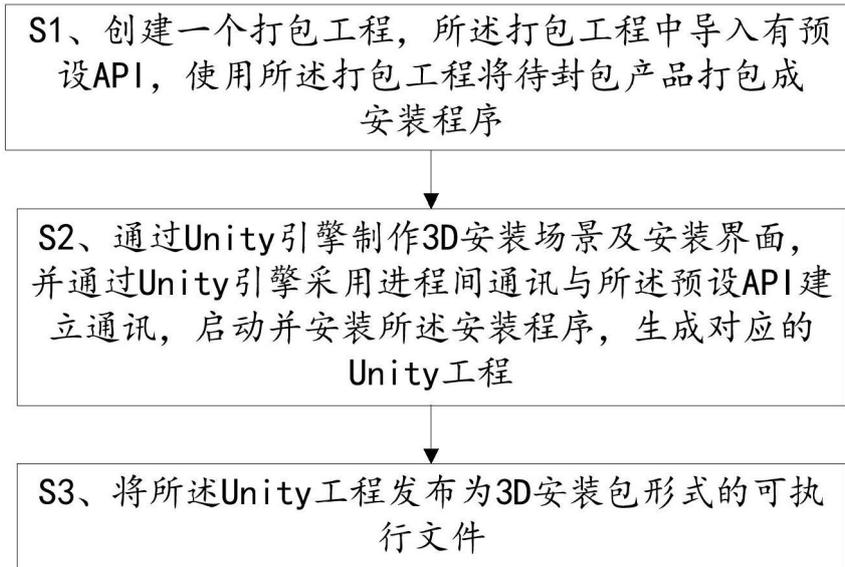


图1

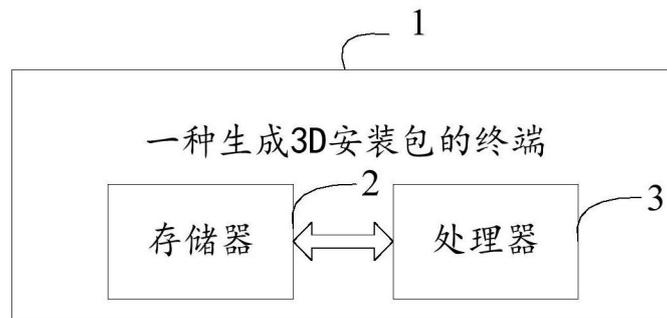


图2