

VR 云渲游平台

SDK 参考

文档版本 02
发布日期 2024-10-28



版权所有 © 华为云计算技术有限公司 2024。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为云计算技术有限公司

地址：贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址：<https://www.huaweicloud.com/>

目录

1 SDK 概述	1
2 Android SDK	2
2.1 SDK 介绍	2
2.2 SDK 下载	2
2.3 SDK 接口参考	3
2.4 客户端开发	9
2.4.1 开发前准备	9
2.4.2 SDK 导入和使用	9
2.4.3 开启“开发者模式”（可选）	13
2.4.4 安装 APK	14
2.4.5 启动应用	14
2.5 附录	14
2.5.1 APP ID 的获取方法	14
2.5.2 JDK 1.8 安装方法	15
2.5.3 Android Studio 3.2.1 安装方法	18
2.5.4 旧版客户端升级	24
2.5.5 错误信息回调	24
2.5.6 修订记录	26
3 Windows SDK	27
3.1 SDK 介绍	27
3.2 SDK 下载	27
3.3 SDK 接口参考	28
3.4 客户端开发	34
3.4.1 开发前准备	34
3.4.2 SDK 导入	34
3.4.3 SDK 使用	37
3.4.4 启动应用	38
3.5 附录	38
3.5.1 Visual Studio 2017 安装方法	38
3.5.2 错误信息回调	39
3.5.3 修订记录	42
4 Web SDK	43

4.1 SDK 介绍.....	43
4.2 SDK 下载.....	43
4.3 Example 示例.....	44
4.4 客户端开发.....	45
4.4.1 开发前准备.....	46
4.4.2 SDK 导入.....	46
4.4.3 SDK 使用.....	46
4.4.4 SDK 接口.....	46
4.4.5 启动应用.....	48
4.4.6 关闭应用.....	48
4.5 附录.....	48
4.5.1 修订记录.....	48

1 SDK 概述

VR云渲游平台服务软件开发工具包（CVR SDK，VR Cloud Rendering & Gaming Platform Software Development Kit）是对CVR服务提供的客户端API进行的封装。用户直接调用SDK提供的接口函数即可实现使用不同类型客户端，使用VR应用或者3D应用的目的。

我们针对不同SDK类型提供了开发指南，开发指南详细介绍了SDK的下载、开发操作、参数、示例和常见问题处理。

SDK类型	开发指南
Android	Android SDK开发指南
Windows	Windows SDK开发指南
Web	Web SDK开发指南

2 Android SDK

2.1 SDK 介绍

华为云VR云渲游平台Android SDK集成华为自研音、视频传输协议及网络优化算法，为您提供低时延、高可靠的云VR和云3D体验。

华为开发者中心提供两类云渲游SDK：Mobile-SDK和VR-SDK，分别面向移动端设备和VR设备，可捕获与预测动作数据（移动端设备：键鼠、触屏等动作数据；VR设备：头显和手柄的动作数据），并将其上传至云端，待云端完成渲染、编码后，将接收到的画面呈现在对应的设备上。此外，SDK还提供获取设备标识符等通用接口，便于您在SDK外层封装自身业务逻辑。

2.2 SDK 下载

约束与限制

华为开发者中心提供的VR云渲游平台（客户端）VR-SDK支持机型包括：

- 6自由度设备（6Dof）：Pico Neo3、Pico Neo2、Oculus Quest、Oculus Quest2
- 3自由度设备（3Dof）：华为VR Glass、小米超级玩家一体机、Pico G2

Mobile-SDK支持机型包括：

- 市面上常见的安卓设备（手机、平板）
- 华为智慧屏

下载 SDK

如需获取VR云渲游平台CVR Android SDK，请执行以下操作：

1. 打开[SDK开发者中心](#)。
2. 选择“VR云渲游平台”的“Android SDK”进行下载。
3. 获取VRClientSDK.zip文件，并解压。

解压后的文件如[表2-1](#)所示。

表 2-1 VRClientSDK.zip 包含的文件

文件名	适应机型
app-Android-Normal-HWVR-debug.aar	华为VR Glass
app-Android-Normal-OVR-debug.aar	Oculus Quest、Oculus Quest2
app-Android-Normal-MiVR-debug.aar	小米超级玩家一体机
assembleAndroidNormalPicoDebug.zip	Pico Neo1、Pico Neo2、Pico G2、Pico Neo3
app-Android-Normal-Mobile-debug.aar	手机、平板、华为智慧屏
version.txt	SDK版本 说明 1.4.0及以上版本的新增文件，1.4.0以下版本暂无识别版本号方式。

其中，assembleAndroidNormalPicoDebug.zip解压后包含的文件如表2-2所示。

表 2-2 assembleAndroidNormalPicoDebug 中包含的文件

文件名	说明	是否必须依赖
app-Android-Normal-Pico-debug.aar	VR云渲游提供的SDK	是
PvrSDK-Native-release.aar	Pico官方提供的SDK	是

2.3 SDK 接口参考

表 2-3 IntentCVR.java 通用接口

接口	描述	是否必选
<code>public IntentCVR(Context context)</code>	【功能说明】 初始化IntentCVR接口。 【请求参数】 context: 新建项目的上下文对象，可以是Activity，也可以是Application。	是

接口	描述	是否必选
public CVRParameter(String userName, String userDomain, String userPassWord, String projectName)	【功能说明】 用户通过该接口配置华为云用户名及密码，客户端根据该信息访问华为云IAM获取Token后再连接至VR云渲游平台进行鉴权，以下简称“ A类 CVRParameter构造函数 ”。 【请求参数】 <ul style="list-style-type: none">• userDomain: 华为云租户名• userName: 华为云用户名• password: 华为云用户密码• projectName: 项目名称 说明 如何获取上述参数对应值，请参考： https://support.huaweicloud.com/sis_faq/sis_04_0032.html	是 CVRParameter构造函数包括A类、B类、C类，请根据需要自选其中一类使用。 三类构造函数的适用场景请参考 SDK导入和使用 。
public CVRParameter(String iamToken)	【功能说明】 用户可自行访问IAM获取Token，通过该接口注入Token进行CVRParameter初始化，以下简称“ B类 CVRParameter构造函数 ”。 【请求参数】 iamToken: 用户Token。	
public CVRParameter(String gpuIp,String gpuToken)	【功能说明】 用户自行获取gpuIp与gpuToken，通过该接口配置gpuIp与gpuToken用于头显直连GPU服务器，以下简称“ C类 CVRParameter构造函数 ”。 【请求参数】 <ul style="list-style-type: none">• gpuIp: GPU服务器公网IP。 gpuIp的获取方式，请参考“渲染节点调度”。• gpuToken: 访问GPU服务器所需鉴权Token。 gpuToken的获取方式，请参考“渲染节点调度”。	
public void setServiceDomain(String serviceDomain)	【功能说明】 设置Service域名，用于指定连接的GPU云服务器所在区域。 【请求参数】 ServiceDomain: Service域名。 ServiceDomain的取值，请参见“ 地区和终端节点 ”中的“终端节点 (Endpoint)”。	是

接口	描述	是否必选
<code>public void setAppId(String appId)</code>	<p>【功能说明】 设置头显要访问的应用。</p> <p>【请求参数】 appId: VR应用对应的应用ID。获取应用ID, 请参考APP ID的获取方法。</p>	是
<code>public void setPort(int port)</code>	<p>【功能说明】 GPU默认监听端口是9944, 如果您更改了GPU监听端口(如使用沙箱模式), 则需要使用该接口修改客户端访问端口号</p> <p>【请求参数】 port: 指定头显发送连接请求的目的端口信息。</p>	否
<code>public String initiateLoad(IntentCVR.CVRParameter customParameter, Class<?> classActivity)</code>	<p>【功能说明】 加载相关参数, 启动云VR流化业务, 并对输入参数进行检验。</p> <p>【请求参数】 customParameter: 已经初始化的鉴权参数。 classActivity: 传入跳转的目标Activity类, 目标Activity的编写建议参考SDK导入和使用章节的Demo项目。</p>	是
<code>public void setAppName(String appName)</code>	<p>【功能说明】 设置客户端连接等待界面展示的应用名称。</p> <p>【请求参数】 appName: 应用名称。</p>	否
<code>public void setLocation(String[] location)</code>	<p>【功能说明】 设置客户端连接的目标云服务器所在区域。</p> <p>【请求参数】 location: 用于设置调度接口的策略, 可以指定一个或多个区域, CVR服务会从指定区域中分配空闲的服务器。支持配置公有云region_id, 如cn-east-3。 如果未设置location, 默认为空数组, 此时CVR服务会根据设备的调度策略进行GPU服务器的分配。 location的取值, 请参见“地区和终端节点”中的“区域”。</p>	否

接口	描述	是否必选
public void setProjectID(String projectID)	<p>【功能说明】 设置访问客户端调度接口使用的项目ID。</p> <p>【请求参数】 projectID: 项目ID, 用于访问调度接口。 获取projectID, 请参见“获取项目ID”。</p>	否 仅使用B类的 CVRParameter 构造函数时 需要传入该 值。
public String getDeviceID(Context context)	<p>【功能说明】 获取设备ID, 访问前需要保证已经获取到了所需的权限。</p> <p>【请求参数】 Context: 请求发起的上下文。所需要获取的权限为: Manifest.permission.READ_EXTERNAL_STORAGE Manifest.permission.WRITE_EXTERNAL_STORAGE</p>	否 仅使用C类的 CVRParameter 构造函数时 需要获取该 值。
public String getBrand()	<p>【功能说明】 获取设备品牌。</p> <p>【请求参数】 无</p>	否, 仅使用C类的 CVRParameter 构造函数时 需要获取该 值。
public static void setErrCallback(CvrErrCallb ack errCallback)	<p>【功能说明】 用于获取SDK内部的错误信息并处理。</p> <p>【请求参数】 errCallback: 回调接口, 支持用户自定义。相关注意事项请参考错误信息回调。</p>	否
public void setTimeLimit(int timeLimit)	<p>【功能说明】 设置应用与GPU服务器建立的连接在后台的最大保活时间。</p> <p>【请求参数】 timeLimit: 此参数代表应用退至后台后, 与GPU服务器建立的连接最大保活时间。 若退至后台到重新打开应用的时间间隔超出该值, 则第二次进入应用后会退出当前VRActivity类, 并触发错误信息回调的onTokenOutdated方法。 取值范围: 0~150s</p>	否

接口	描述	是否必选
<code>public int getTimeLimit()</code>	【功能说明】 获取应用与GPU服务器建立的连接在后台的最大保活时间。 【请求参数】 无	否
<code>public void setIamDomain(String iamDomain)</code>	【功能说明】 设置访问IAM服务鉴权接口的域名。 【请求参数】 iamDomain: 应用访问IAM服务, 获取鉴权的域名。默认为 iam.myhuaweicloud.com	是
<code>public String getIamDomain()</code>	【功能说明】 获取访问IAM服务鉴权接口的域名。 【请求参数】 无	否
<code>public String getDeviceType()</code>	【功能说明】 识别并获取当前设备类型, 包括vr、mobile、ideahub。 用于访问CVR平台的调度接口, 上传当前设备信息。 【请求参数】 无	是
<code>public void setGpuIpType(String gpuInfoType)</code>	【功能说明】 设置gpu服务器ip的类型, 默认为public。 【请求参数】 gpuInfoType: gpu服务器ip的类型, 弹性公有ip填“public”, 私有ip填“private”。	否
<code>public String getGpuIpType()</code>	【功能说明】 获取设置的gpu服务器ip类型。 【请求参数】 无	否

表2 3D客户端IntentCVR.java中额外接口列表

接口	描述	是否必选
public void setApplicationUser(String applicationUser)	【功能说明】 设置访问CVR平台的调度接口使用的用户标识ID。 【请求参数】 applicationUser: 用户标识ID, 用于访问调度接口。	是
public String getApplicationUser()	【功能说明】 获取访问客户端调度接口使用的用户标识ID。 【请求参数】 无	否
public void setAppLaunchArgs(String appLaunchArgs)	【功能说明】 设置应用启动参数, 如应用启动无额外参数, 则不需要设置。 【请求参数】 appLaunchArgs: 应用启动参数。	否
public String getAppLaunchArgs()	【功能说明】 获取设置的应用启动参数。 【请求参数】 无	否
public void setResolution(String resolution)	【功能说明】 设置应用启动的分辨率, 默认为1080p。 【请求参数】 resolution: 应用启动的分辨率。可以设置的参数为“480p”, “540p”, “720p”, “1080p”, “2k”, “4k”	否
public String getResolution()	【功能说明】 获取设置的应用启动分辨率。 【请求参数】 无	否
public static void pauseCVR()	【功能说明】 暂停串流服务, 画面停滞。 【请求参数】 无	否
public static void resumeCVR()	【功能说明】 恢复串流服务, 与pauseCVR组合使用。 【请求参数】 无	否

接口	描述	是否必选
<code>public static void restartCVR()</code>	【功能说明】 重启服务器的应用。 【请求参数】 无	否
<code>public static void setAppClosedCallback(CaPlayer.ICaAppClosedCallback appClosedCallback)</code>	【功能说明】 设置服务端应用被动关闭的反馈接口。 【请求参数】 appClosedCallback: 用户自定义的接口方法。	否
<code>public static CaPlayer.ICaAppClosedCallback getAppClosedCallback()</code>	【功能说明】 获取设置的服务端应用被动关闭的反馈接口。 【请求参数】 无	否

2.4 客户端开发

2.4.1 开发前准备

前提条件

- 已注册华为云，并开通VR云渲游平台公测权限。
- 已按照《[VR云渲游平台快速入门](#)》完成创建集群、创建应用、创建云服务器、配置连接策略（可选）等操作。
- 已获取并记录应用的appID信息，获取appID方法请参见[APP ID的获取方法](#)。
- 已购买VR云渲游平台支持的头显设备。

软硬件环境配置要求

使用VR云渲游平台SDK进行集成开发时需使用Android环境，请进行以下配置：

- 配置Java的编译环境，建议选择JDK版本1.8，安装JDK 1.8的方法请参见[JDK 1.8 安装方法](#)。
- 准备Android Studio集成开发环境，建议版本3.2.1，安装Android Studio方法请参见[Android Studio 3.2.1安装方法](#)。

2.4.2 SDK 导入和使用

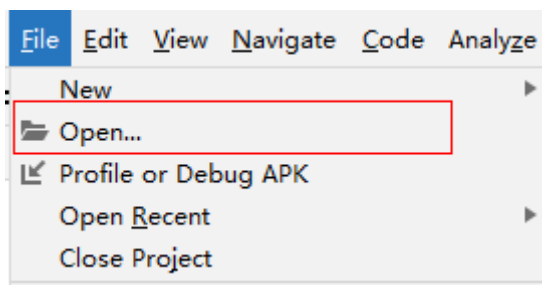
步骤1 根据使用的终端设备下载对应的DemoApplication并导入。

1. 设备若为VR头显，通过<https://cvr-east-3.obs.myhuaweicloud.com/DemoVrApplication.zip>下载DemoVrApplication.zip;

设备若为手机、平板、华为智慧屏，通过<https://cvr-east-3.obs.myhuaweicloud.com/Demo3dApplication.zip>下载 Demo3dApplication.zip

2. 下载后解压，并使用android studio打开。

图 2-1 打开项目



步骤2 配置。

1. 解压下载的SDK.zip文件，根据头显种类获取对应的aar文件。
2. 将所需的aar文件放入模板项目的app\libs中

图 2-2 解压的 SDK 文件

	app-Android-Normal-HWVR-debug...	2021/7/2 14:39	AAR 文件	12,909 KB
	app-Android-Normal-MiVR-debug.aar	2021/7/2 14:39	AAR 文件	12,139 KB
	app-Android-Normal-Mobile-debug...	2021/7/2 14:49	AAR 文件	9,591 KB
	app-Android-Normal-OVR-debug.aar	2021/7/2 14:40	AAR 文件	12,144 KB
	assembleAndroidNormalPicoDebug...	2021/7/2 14:49	压缩(zipped)文件...	13,562 KB
	version.txt	2021/7/2 14:35	文本文档	1 KB

表 2-4 SDK 依赖

头显	依赖命令
华为VR Glass	implementation(name: 'app-Android-Normal-HWVR-debug',ext:'aar')
Pico Neo1、Pico Neo2、Pico G2、Pico Neo3	implementation(name: 'app-Android-Normal-Pico-debug',ext:'aar') implementation(name: 'PvrSDK-Native-release',ext:'aar')
Oculus Quest、Oculus Go、Oculus Quest2	implementation(name: 'app-Android-Normal-OVR-debug',ext:'aar')
小米超级玩家一体机	implementation(name: 'app-Android-Normal-MiVR-debug',ext:'aar')
手机、平板、华为智慧屏	implementation(name: 'app-Android-Normal-Mobile-debug',ext:'aar')

3. 根据引入的sdk名称，在app\build.gradle文件修改配置。

在注释中选用实际使用aar包对应的implementation字段即可。

图 2-3 app\build.gradle 的引用配置

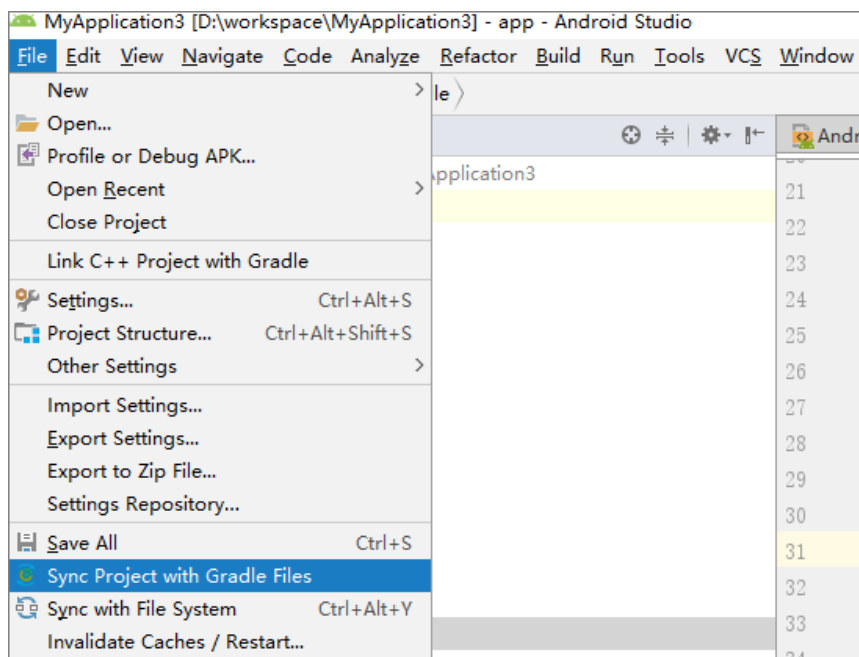
```
dependencies {  
    implementation fileTree(dir: 'libs', include: ['*.jar'])  
    implementation 'com.android.support:appcompat-v7:+'  
    // implementation(name: 'app-Android-Normal-Pico-debug', ext: 'aar')  
    // implementation(name: 'PvrSDK-Native-release', ext: 'aar')  
    // implementation(name: 'app-Android-Normal-0VR-debug', ext: 'aar')  
    // implementation(name: 'app-Android-Normal-HWVR-debug', ext: 'aar')  
    // implementation(name: 'app-Android-Normal-MiVR-debug', ext: 'aar')  
    // implementation(name: 'app-Android-Normal-SVR-debug', ext: 'aar')  
    implementation(name: 'app-Android-Normal-Mobile-debug', ext: 'aar')  
    implementation 'com.android.support.constraint:constraint-layout:2.0.0'  
    testImplementation 'junit:junit:4.12'  
    androidTestImplementation 'com.android.support.test:runner:1.0.2'  
    androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'  
    testImplementation 'org.json:json:20190722'  
}
```

4. 修改app\src\main\java\com\example\myapplication\MainActivity.java。
在MainActivity.java中对参数配置进行修改，两种模板所需修改的参数如下所示。

图 2-4 VR 客户端的模板所需参数

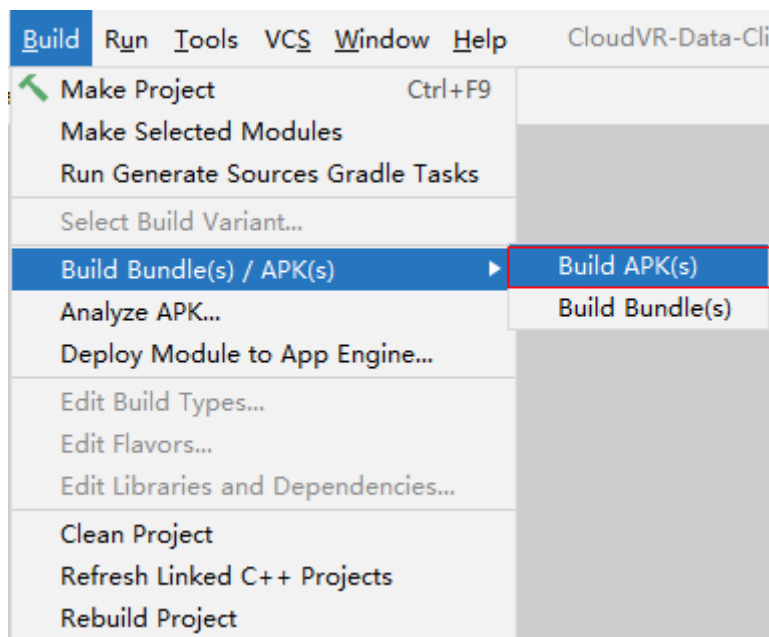
```
//连接类型和连接所需的参数，这里是您注意和修改的参数<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<  
  
private int linkType = TYPE_ACCOUNT;  
private String appID = "appID";  
private String username = "username";  
private String userdomain = "userdomain";  
private String userPassword = "*****";  
private String projectName = "projectName";  
private String serviceDomain = "serviceDomain";  
private String[] location = new String[] {};  
private String iamDomain = "iam.myhuaweicloud.com";  
private String gpuIpType = "";  
  
//连接类型和连接所需的参数，这里是您注意和修改的参数>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
```


图 2-7 选择 “Sync Project With Gradle Files”



2. 选择Build APK(s)，完成APK打包。

图 2-8 选择 Build APK(s)



----结束

2.4.3 开启“开发者模式”（可选）

操作场景

对于“小米超级玩家一体机”用户，需按照本章节操作，开启“开发者模式”。

对于其他用户，请跳过本章节。

操作步骤

使用小米一体机的用户，需要通过如下链接开启“开发者模式”，才能通过adb安装应用。

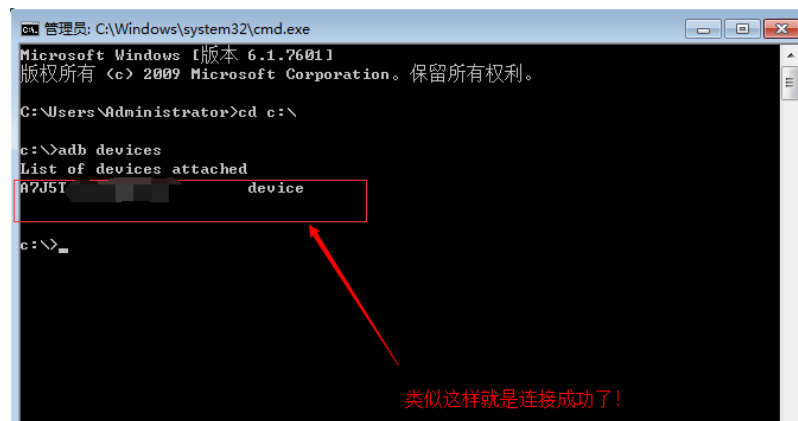
链接：<https://dev.mi.com/console/doc/detail?pld=815>

2.4.4 安装 APK

1. 将设备通过USB连接至PC。
2. 执行以下命令，查看设备状态为“已连接”。

adb devices

图 2-9 设备已连接



3. 执行以下命令，安装APK至设备。

adb install *.apk**

命令行返回success表示安装成功。

2.4.5 启动应用

1. 佩戴头显设备。
2. 在应用列表中找到“VR云渲游客户端”。
3. 打开“VR云渲游客户端”后，允许应用获取相应权限，即可连接至VR云渲游平台开启串流业务。

2.5 附录

2.5.1 APP ID 的获取方法

1. 在服务列表中选择“计算 > VR云渲游平台”。
访问VR云渲游平台。
2. 在左侧导航栏，选择“应用管理”，打开应用管理页面。
3. 查看已创建的应用，红框所示即为appID。

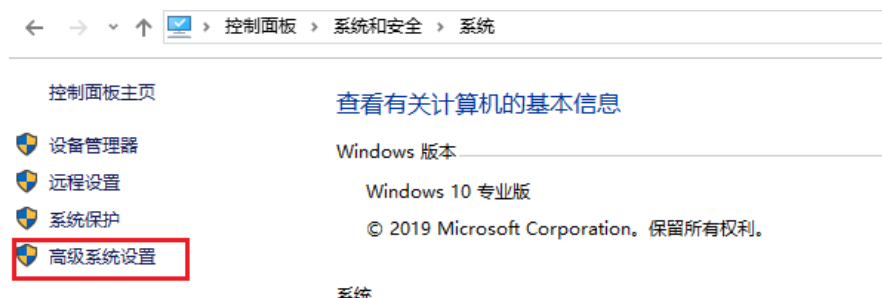


4. SDK1.4.0版本新增getDeviceID(Context context)接口，详情请参见[SDK接口参考](#)。

2.5.2 JDK 1.8 安装方法

1. 访问JDK官网下载链接：<https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html>。
您也可以在搜索引擎中查找JDK1.8，并前往官网下载。
2. 查找最新版本，并选择相应的操作系统进行下载。
示例：
 - 最新版本：8u271
 - 操作系统：选择Windows x64
3. 安装过程可以直接按照默认设置来完成，连续单击“下一步”完成JDK的安装。
4. 安装完成之后，需要配置相应环境变量。
 - a. 右键单击“我的电脑”，选择“属性”，打开“系统”页面。

图 2-10 系统



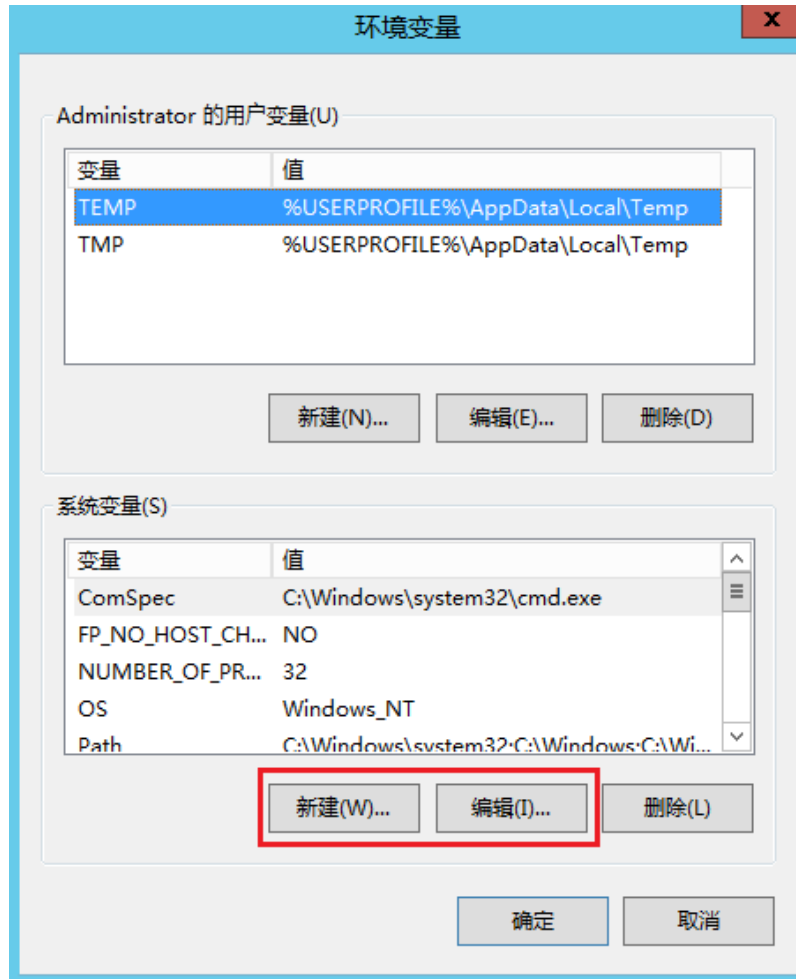
- b. 单击“高级系统设置”，打开“系统属性”页面。

图 2-11 高级系统设置



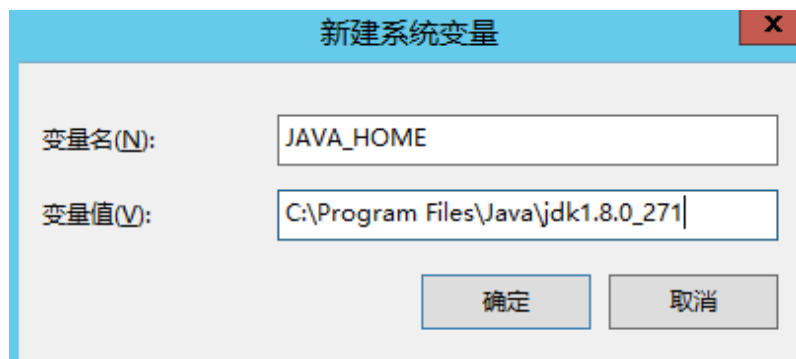
- c. 单击“环境变量”。

图 2-12 环境变量



- d. 在“系统变量”栏单击“新建”，输入如下信息：
变量名：**JAVA_HOME**
变量值：以实际JDK安装时使用的目录为准
- e. 单击“确定”，保存变量。

图 2-13 新建系统变量



- f. 再次单击“新建”新建系统变量：
变量名：**classpath**
变量值：**.;%JAVA_HOME%\lib;**

- g. 在“系统变量”栏查找并选中变量**Path**，然后单击“编辑”，在弹出的对话框中的变量值末尾，添加如下字符串：
`;%JAVA_HOME%\bin;%JAVA_HOME%\jre\bin`

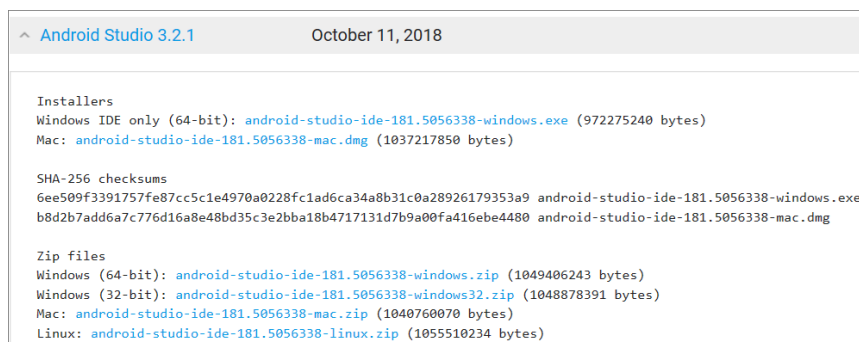
2.5.3 Android Studio 3.2.1 安装方法

1. 下载Android Studio

下载地址：<https://developer.android.google.cn/studio/archive>

在归档版本中找到Android Studio3.2.1并下载系统匹配的版本。

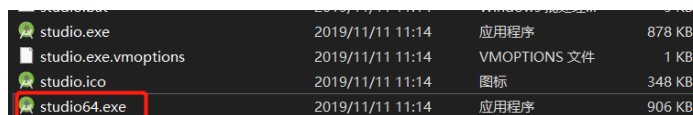
图 2-14 下载 Android Studio3.2.1



2. 安装并配置Android Studio

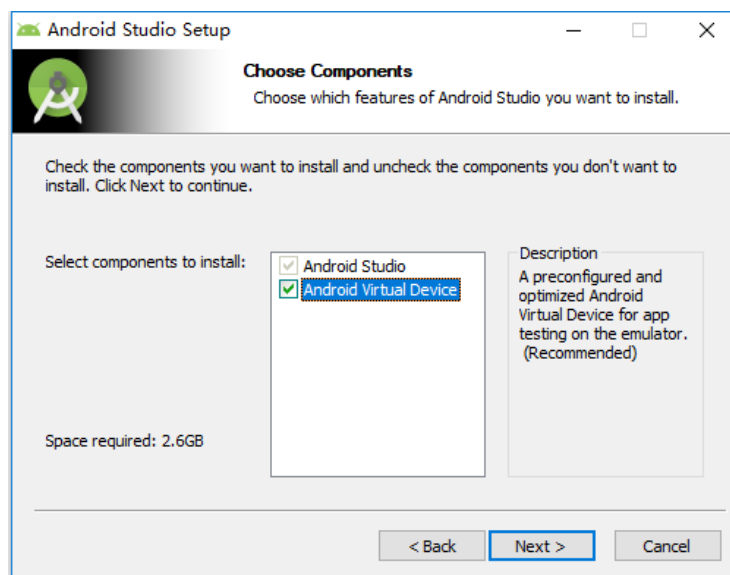
- a. 在下载文件的“bin”目录下找到并双击“Studio64.exe”执行文件进行安装。

图 2-15 双击“Studio64.exe”



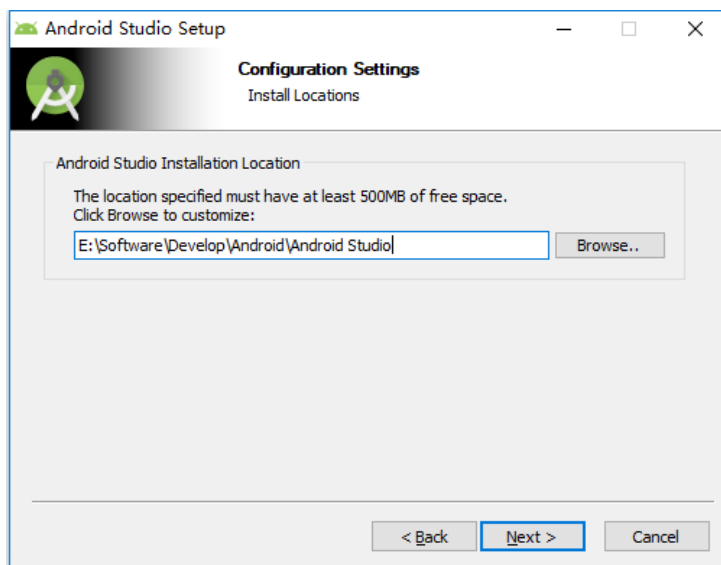
- b. 勾选“Android Virtual Devices”选项，然后单击“Next”。

图 2-16 Choose Components

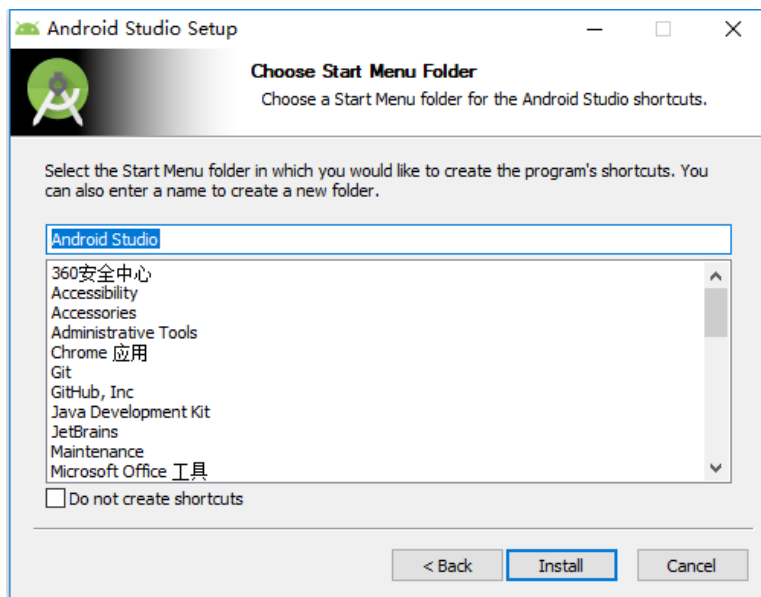


- c. 选择Android Studio安装的本地路径，建议不要安装在C盘。

图 2-17 Install Locations

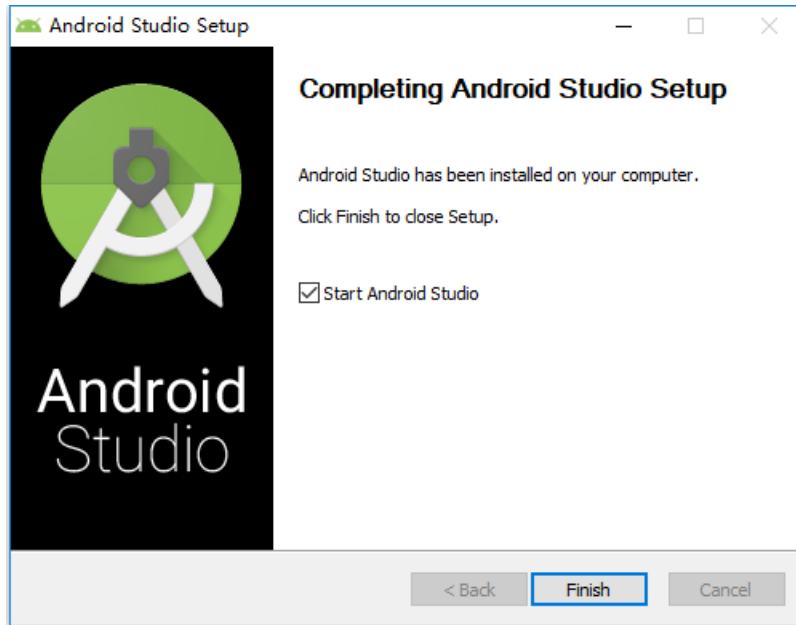


- d. 安装设置完成之后，单击“Install”，等待安装完成。



3. 初步设置

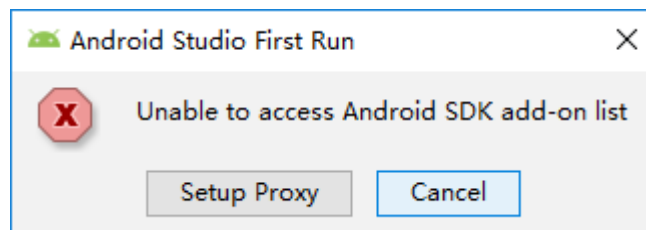
- a. 安装完成后单击“Finish”，启动Android Studio，此时将会进行首次启动前的初步设置。



说明

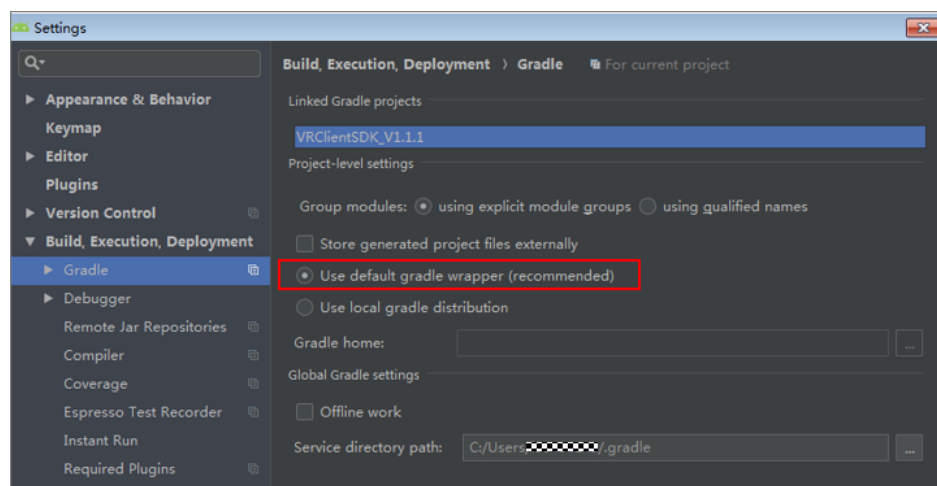
如果系统提示您无法访问SDK加载项列表，此时请单击“Cancel”退出，后续会下载SDK。

图 2-18 Android Studio First Run



- b. 根据个人喜好选择其他配置，并单击“Next”完成初步设置。
4. Gradle配置
根据图2-19所示，完成Gradle配置。

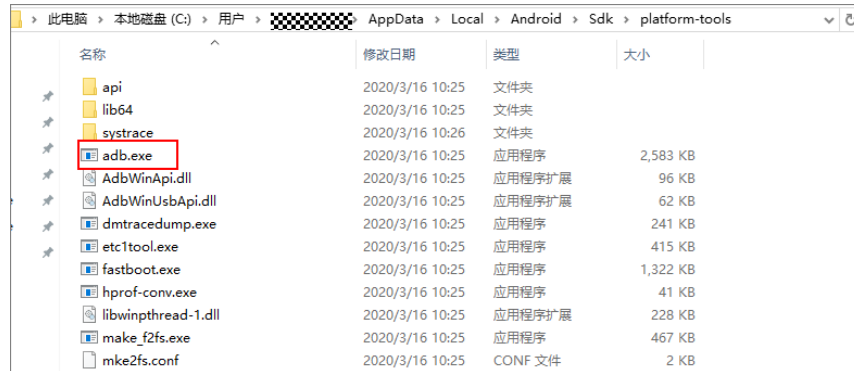
图 2-19 Gradle 配置



5. 配置ADB至环境变量

将adb.exe所在路径（Android Studio SDK安装路径“\platform-tools”）添加至Windows系统变量：Path。

图 2-20 adb 路径



a. 打开“控制面板”，进入“高级系统设置”。

图 2-21 高级系统设置



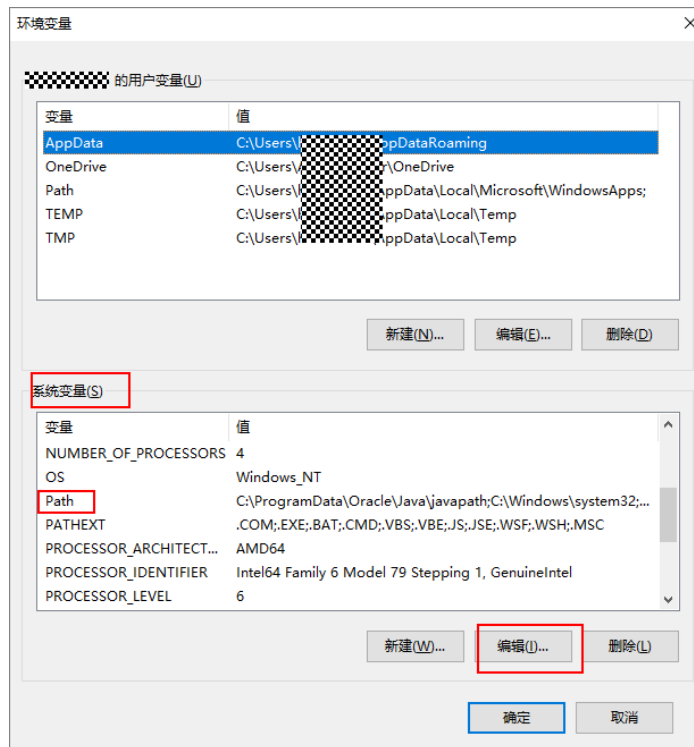
b. 单击“环境变量”。

图 2-22 系统属性



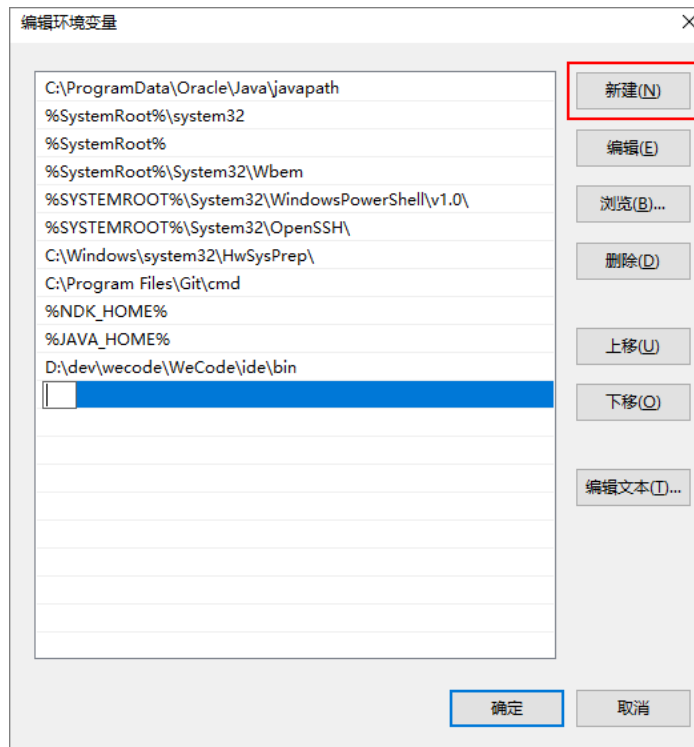
- c. 在“系统变量”栏，选中“Path”所在行，并单击“编辑”。

图 2-23 环境变量



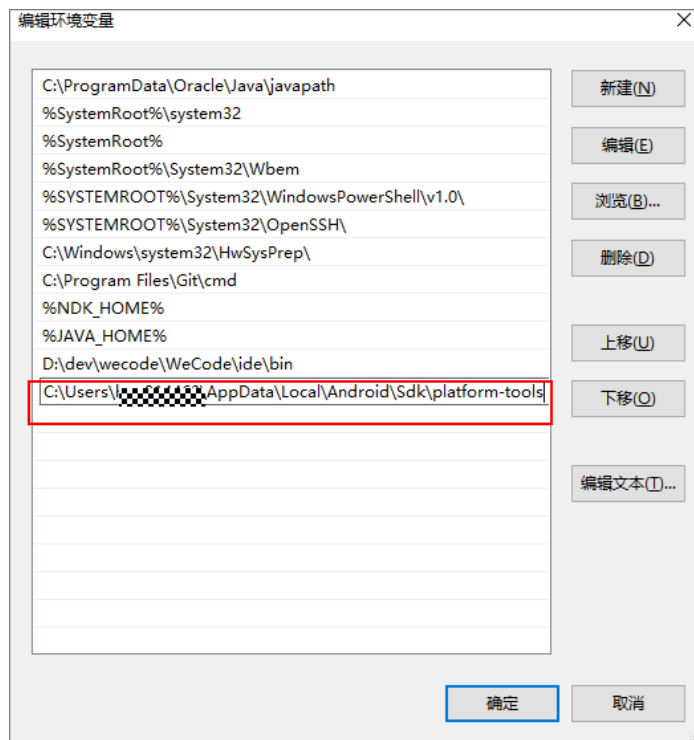
- d. 单击“新建”。

图 2-24 编辑环境变量



- e. 添加adb路径，然后单击“确定”。

图 2-25 粘贴 adb 路径



- f. 打开CMD命令提示窗口，输入命令**adb**，如图2-26所示，adb环境变量配置成功。

图 2-26 adb 命令

```
Microsoft Windows [版本 10.0.17763.1039]
(c) 2018 Microsoft Corporation. 保留所有权利。

C:\Users\>adb
Android Debug Bridge version 1.0.41
Version 29.0.6-6198205
Installed as C:\Users\AppData\Local\Android\Sdk\platform-tools\adb.exe

global options:
-a          listen on all network interfaces, not just localhost
-d          use USB device (error if multiple devices connected)
-e          use TCP/IP device (error if multiple TCP/IP devices available)
-s SERIAL   use device with given serial (overrides $ANDROID_SERIAL)
-t ID       use device with given transport id
-H          name of adb server host [default=localhost]
-P          port of adb server [default=5037]
-L SOCKET   listen on given socket for adb server [default=tcp:localhost:5037]

general commands:
devices [-l]  list connected devices (-l for long output)
help         show this help message
version      show version num
```

至此，Android Studio及adb环境变量配置成功。

2.5.4 旧版客户端升级

客户端由低版本升级到1.4.0版本需要注意以下内容：

- 旧版客户端不支持沙箱型云服务器，1.4.0及以上版本的客户端开始支持沙箱型云服务器。
- “IntentCVR(Context context)”构造函数中，“context”从“Application”类改为“Activity”类。
- “setToken/getToken”的方法名改为“setlamToken/getlamToken”。
- 新增的接口中，“getDeviceID(Context context)”需要先获取存储相关的权限后，然后才能使用，否则会返回空字符串。
- InitiateLoad接口，类型由“void”修改为“string”类型，返回对入参的校验结果。
- 若使用的目标终端设备为VRGlass，“AndroidManifest.xml”中的“<Intent-filter>”配置项需要按如下信息修改：

```
<intent-filter>
  <action android:name="com.huawei.android.vr.action.MAIN" />
  <category android:name="android.intent.category.DEFAULT" />
</intent-filter>
```

2.5.5 错误信息回调

错误信息回调使用的接口为CvrErrCallback，该接口包含成员函数onErrorMessage(String errorCode, String errorMessage)和onTokenOutdated()。

void onErrorMessage(String errorCode, String errorMessage)

注册该方法后，SDK内部识别到错误时，会通过该方法反馈错误信息。入参如下：

- errorCode：错误代号或简明的错误信息。
- errorMessage：具体的错误信息。

表 2-5 错误场景

errCode	errMessage	发生的场景
ERROR_INPUT_INVALID	Context value is null.	调用getDeviceID(Context context)时入参为空。
ERROR_PERMISSION_DENIED	权限未授予，无法获取设备ID	调用getDeviceID前未获取相关存储权限。
TOKEN_NULL	未获取到IAM TOKEN	在访问IAM服务时，未获取到token。可能原因是网络无法访问到IAM服务，或是账号密码错误。
CVR.0001	消息格式体错误	在访问CVR服务的调度接口时出现。
CVR.0005	设备ID格式错误，只允许输入小写英文，数字和-	
CVR.0010	应用ID格式错误，只允许输入小写英文，数字和-	
CVR.0012	指定资源ID错误，未找到（集群ID，应用ID，云服务器ID）	
CVR.0013	指定应用下无闲置云服务器	
CVR.0014	当前集群连接数已到上限，请稍后再连或者升级集群规格	
CVR.0016	指定应用的连接策略禁止您的设备连接	
CVR.0030	监控数据设备品牌格式错误，只允许输入英文大小写，数字和空格_	
CVR.0036	鉴权失败，项目ID错误	
CVR.0040	有闲置云服务器，请安装该应用后重试	
CVR.0041	连接Token生成失败	
CVR.0060	系统错误，请稍后重试或联系客服	
CVR.0194	指定应用下或当前设备的策略配置下无云服务器	
CVR.0195	区域传参类型错误	
CVR.0196	该选择的区域不符合已配置的调度策略	
CVR.0231	sandbox_enable错误，只允许输入[true,false]	

errCode	errMessage	发生的场景
CVR.0232	存在闲置的沙箱型云服务器，请修改sandbox_enable参数以支持沙箱型云服务器	

void onTokenOutdated()

注册该方法后，应用在退至后台后，SDK会记录对应的时间戳，再次打开应用后，计算此时间间隔，并与通过IntentCVR设置的timeLimit值做比较，当大于该值时，当前的VRActivity出栈并触发该方法。

建议在此处加入重连逻辑，保证重新打开应用后，能够快速连接至GPU服务器。

2.5.6 修订记录

发布时间	修订记录
2021-07-30	修改 <ul style="list-style-type: none">• SDK接口参考，SDK1.4.1上线，新增部分SDK接口。• SDK导入和使用，修改SDK使用方式，新增模板。
2021-04-30	第二次正式发布。 新增 <ul style="list-style-type: none">• JDK 1.8安装方法• 旧版客户端升级• 错误信息回调 修改 <ul style="list-style-type: none">• SDK接口参考，SDK1.4.0上线，新增部分SDK接口。• SDK导入和使用，修改SDK使用方式，新增模板。
2020-09-08	第一次正式发布。

3 Windows SDK

3.1 SDK 介绍

华为云VR云渲游平台Windows SDK集成华为自研视频传输协议及网络优化算法，为您提供低时延、高可靠的云3D体验。

SDK可实现鼠标和键盘的动作数据捕获与预测，并将其上传至云端，待云端完成渲染、编码后，将接收到的3D画面呈现在PC客户端上。此外，SDK还提供获取设备标识符等通用接口，便于您在SDK外层封装自身业务逻辑。

3.2 SDK 下载

如需获取VR云渲游平台CVR Windows SDK，请执行以下操作：

1. 打开[SDK开发者中心](#)。
2. 选择“VR云渲游Windows SDK”进行“SDK下载”。
3. 获取WindowsSDK.zip文件，并解压。

解压后的文件如[表3-1](#)所示。

表 3-1 WindowsSDK.zip 包含的文件

文件名	说明
bin	二进制文件
include	包含的头文件
lib	静态数据链接库文件
version.txt	版本号文件

3.3 SDK 接口参考

接口	描述	是否必选
IntentCVR()	【功能说明】 初始化IntentCVR接口。	是
CVRParameter(std::string userName, std::string userDomain, std::string userPassWord, std::string projectName)	【功能说明】 用户通过该接口配置华为云用户名及密码，客户端根据该信息访问华为云IAM获取Token后再连接至VR云渲游平台进行鉴权，以下简称“ A类 CVRParameter构造函数 ”。 【请求参数】 <ul style="list-style-type: none">• userDomain: 华为云租户名• userName: 华为云用户名• password: 华为云用户密码• projectName: 项目名称 说明 如何获取上述参数对应值，请参考： https://support.huaweicloud.com/sis_faq/sis_04_0032.html	是 CVRParameter构造函数包括A类、B类、C类，请根据需要自选其中一类使用。 三类构造函数的适用场景请参考 SDK使用 。
CVRParameter(std::string iamToken)	【功能说明】 用户可自行访问IAM获取Token，通过该接口注入Token进行CVRParameter初始化，以下简称“ B类 CVRParameter构造函数 ”。 【请求参数】 iamToken: 用户Token。	
CVRParameter(std::string gpulp, std::string gpuToken)	【功能说明】 用户自行获取gpuIp与gpuToken，通过该接口配置gpulp与gpuToken用于设备直连GPU服务器，以下简称“ C类 CVRParameter构造函数 ”。 【请求参数】 <ul style="list-style-type: none">• gpuIp: GPU服务器公网IP。 gpuIp的获取方式，请参考“渲染节点调度”。• gpuToken: 访问GPU服务器所需鉴权Token。 gpuToken的获取方式，请参考“渲染节点调度”。	

接口	描述	是否必选
void setServiceDomain(std::string serviceDomain)	【功能说明】 设置Service域名，用于指定连接的GPU云服务器所在区域。 【请求参数】 ServiceDomain: Service域名。 ServiceDomain的取值，请参见“ 地区和终端节点 ”中的“终端节点 (Endpoint)”。	是
void setAppId(std::string appId)	【功能说明】 设置设备要访问的应用。 【请求参数】 appId: 应用对应的应用ID。获取应用ID，请参考 APP ID的获取方法 。	是
void setPort(int port)	【功能说明】 GPU默认监听端口是10020，如果您更改了GPU监听端口（如使用多路会话模式），则需要使用该接口修改客户端访问端口号 【请求参数】 port: 指定设备发送连接请求的目的端口信息。	否
std::string initiateLoad(CVRParameter customParameter)	【功能说明】 加载相关参数，启动云应用流化业务，并对输入参数进行检验。 【请求参数】 customParameter: 已经初始化的鉴权参数。	是
void setLocation(std::string location)	【功能说明】 设置客户端连接的目标云服务器所在区域。 【请求参数】 location: 用于设置调度接口的策略，可以指定一个或多个区域，CVR服务会从指定区域中分配空闲的服务器。支持配置公有云 region_id，如cn-east-3。 如果未设置location，默认为空数组，此时CVR服务会根据设备的 调度策略 进行GPU服务器的分配。 location的取值，请参见“ 地区和终端节点 ”中的“区域”。	否

接口	描述	是否必选
<code>void setProjectID(std::string projectID)</code>	<p>【功能说明】 设置访问客户端调度接口使用的项目ID。</p> <p>【请求参数】 projectID: 项目ID, 用于访问调度接口。获取projectID, 请参见“获取项目ID”。</p>	否 仅使用B类的CVRParameter构造函数时需要传入该值。
<code>void setApplicationUserID(std::string userID)</code>	<p>【功能说明】 设置访问客户端调度接口使用的用户标识ID, 如不存在用户标识ID, 则不需要设置。</p> <p>【请求参数】 userID: 用户标识ID, 用于访问调度接口。</p>	否
<code>void setAppLaunchArgs(std::string appLaunchArgs)</code>	<p>【功能说明】 设置应用启动参数, 如应用启动无额外参数, 则不需要设置。</p> <p>【请求参数】 appLaunchArgs: 应用启动参数。</p>	否
<code>public void setResolution(String resolution)</code>	<p>【功能说明】 设置应用启动的分辨率, 默认为1080p。</p> <p>【请求参数】 resolution: 应用启动的分辨率。可以设置的参数为“480p”, “540p”, “720p”, “1080p”, “2k”, “4k”</p>	否
<code>public String getResolution()</code>	<p>【功能说明】 获取设置的应用启动分辨率。</p> <p>【请求参数】 无</p>	否
<code>public void setGpuIpType(String gpuInfoType)</code>	<p>【功能说明】 设置gpu服务器ip的类型, 默认为public。</p> <p>【请求参数】 gpuInfoType: gpu服务器ip的类型, 弹性公有ip填“public”, 私有ip填“private”。</p>	否
<code>public String getGpuIpType()</code>	<p>【功能说明】 获取设置的gpu服务器ip类型。</p> <p>【请求参数】 无</p>	否

接口	描述	是否必选
std::string getDeviceID()	【功能说明】 获取设备ID。 【请求参数】 无	否 仅使用C类的CVRParameter构造函数时需要获取该值。
std::string getBrand()	【功能说明】 获取设备品牌。 【请求参数】 无	否，仅使用C类的CVRParameter构造函数时需要获取该值。
void setErrCallback(CvrErrCallback errCallback)	【功能说明】 用于获取SDK内部的错误信息并处理。 【请求参数】 errCallback: 回调接口，支持用户自定义。相关注意事项请参考 错误信息回调 。	否
CVRInstance_t* CVRCreatelInstance()	【功能说明】 创建一个云应用客户端实例。 【请求参数】 无	是
CVRCode CVRSetConfig(CVRInstance_t* thiz, CVRConfig_t* config)	【功能说明】 对云应用客户端实例设置属性。 【请求参数】 <ul style="list-style-type: none"> thiz: 由CVRCreatelInstance()创建的云应用客户端实例。 config: CVRConfig_t类型的初始化参数对象。 	是
CVRCode CVRStart(CVRInstance_t* thiz)	【功能说明】 开始显示画面。 【请求参数】 thiz: 由CVRCreatelInstance()创建的云应用客户端实例。 【返回值】 返回CVRCode类型的返回码，如果值为CVR_CODE_OK常量，则为成功。	是
void CVRSetLogLevel(int level);	【功能说明】 设置日志级别。 【请求参数】 level: 范围[0-8]。数值越大，日志越详细。	否

接口	描述	是否必选
CVRCode CVRSetRenderView(CVRInstance_t* this, void* view)	【功能说明】 画面显示在指定窗口句柄。 【请求参数】 <ul style="list-style-type: none"> thiz: 由CVRCreateInstance()创建的云应用客户端实例。 view: 一个窗口句柄。 【返回值】 返回CVRCode类型的返回码, 如果值为CVR_CODE_OK常量, 则为成功。	否
CVRCode CVRSetControlStateChangedCallback(CVRInstance_t* this, CVRControlStateChangedCallback_t callback, void* user_data)	【功能说明】 设置主控模式状态变化的回调通知。 【请求参数】 <ul style="list-style-type: none"> thiz: 由CVRCreateInstance()创建的云应用客户端实例。 callback: 事件触发时的回调函数。 view: 一个窗口句柄。 【返回值】 返回CVRCode类型的返回码, 如果值为CVR_CODE_OK常量, 则为成功。	否
CVRCode CVRSetOnVideoSizeChangedCallback(CVRInstance_t* this, CVROnVideoSizeChangedCallback_t callback, void* user_data)	【功能说明】 设置当前视频大小变化的回调通知。 【请求参数】 <ul style="list-style-type: none"> thiz: 由CVRCreateInstance()创建的云应用客户端实例。 callback: 事件触发时的回调函数。 user_data: 设置回调函数中自定义数据。 【返回值】 返回CVRCode类型的返回码, 如果值为CVR_CODE_OK常量, 则为成功。	否
CVRCode CVRSetShowDebug(CVRInstance_t* this, int show_debug)	【功能说明】 是否显示调试参数。 【请求参数】 thiz: 由CVRCreateInstance()创建的云应用客户端实例。 【返回值】 返回CVRCode类型的返回码, 如果值为CVR_CODE_OK常量, 则为成功。	否

接口	描述	是否必选
CVRCode CVRGetStatistics(CVRInstance_t* thiz, CVRStatistics_t* statistics)	【功能说明】 获取当前串流的统计数据。 【请求参数】 <ul style="list-style-type: none"> thiz: 由CVRCreateInstance()创建的云应用客户端实例。 statistics: 接收相关统计数据的外部变量。 【返回值】 返回CVRCode类型的返回码, 如果值为CVR_CODE_OK常量, 则为成功。	是
CVRCode CVRGetAuth(CVRInstance_t* thiz, CVRAuth* out_val)	【功能说明】 获取当前是主控客户端还是旁观客户端。 【请求参数】 <ul style="list-style-type: none"> thiz: 由CVRCreateInstance()创建的云应用客户端实例。 out_val: 接收当前模式的变量。 【返回值】 返回CVRCode类型的返回码, 如果值为CVR_CODE_OK常量, 则为成功。	否
CVRCode CVRSetOnAppClosedCallback(CVRInstance_t* thiz, CVROnAppClosedCallback_t callback, void* user_data)	【功能说明】 设置应用关闭的回调通知。 【请求参数】 <ul style="list-style-type: none"> thiz: 由CVRCreateInstance()创建的云应用客户端实例。 callback: 事件触发时的回调函数。 user_data: 设置回调函数中自定义数据。 【返回值】 返回CVRCode类型的返回码, 如果值为CVR_CODE_OK常量, 则为成功。	否
CVRCode CVRRestartApp(CVRInstance_t* thiz, bool cancel)	【功能说明】 重新启动应用。 【返回值】 返回CVRCode类型的返回码, 如果值为CVR_CODE_OK常量, 则为成功。	否
CVRCode CVRPause(CVRInstance_t* thiz)	【功能说明】 暂停云应用资源。 【返回值】 返回CVRCode类型的返回码, 如果值为CVR_CODE_OK常量, 则为成功。	否

接口	描述	是否必选
<code>CVRCode</code> <code>CVRResume(CVRInstance_t* thiz)</code>	【功能说明】 恢复云应用资源。 【返回值】 返回CVRCode类型的返回码，如果值为CVR_CODE_OK常量，则为成功。	否
<code>void</code> <code>CVRRelease(CVRInstance_t* thiz)</code>	【功能说明】 获取资源并停止云应用。 【返回值】 返回CVRCode类型的返回码，如果值为CVR_CODE_OK常量，则为成功。	是

3.4 客户端开发

3.4.1 开发前准备

前提条件

- 已注册华为云，并开通VR云渲游平台公测权限。
- 已按照《[VR云渲游平台快速入门](#)》完成创建集群、创建应用、创建云服务器、配置连接策略（可选）等操作。
- 已获取并记录应用的appID信息，获取appID方法请参见[APP ID的获取方法](#)。
- 已购买VR云渲游平台支持的头显设备。

软硬件环境配置要求

使用VR云渲游平台Windows SDK进行集成开发时需使用Windows C++开发环境，请进行以下配置：

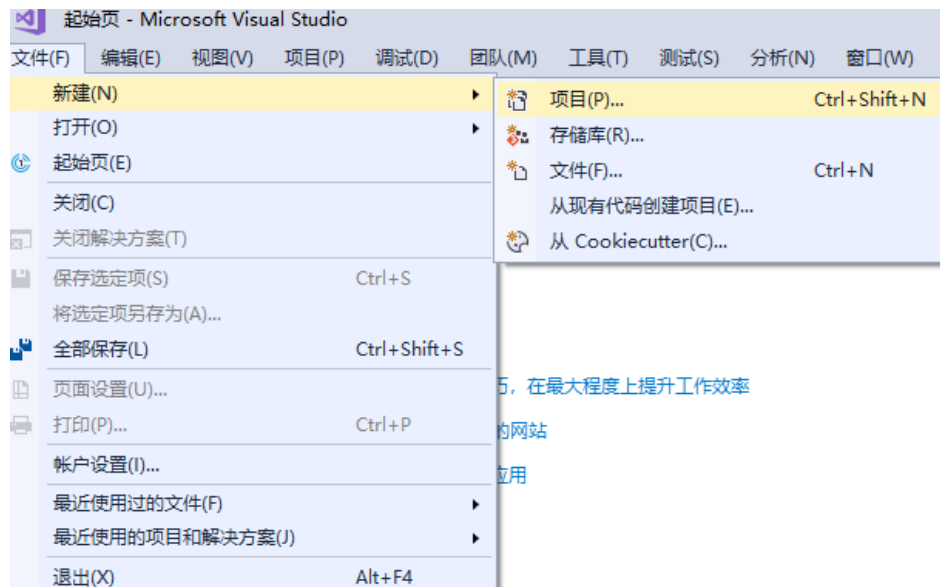
- 准备Visual Studio集成开发环境，建议版本2017，安装Visual Studio方法请参见[Visual Studio 2017安装方法](#)。

3.4.2 SDK 导入

步骤1 创建Visual Studio项目。

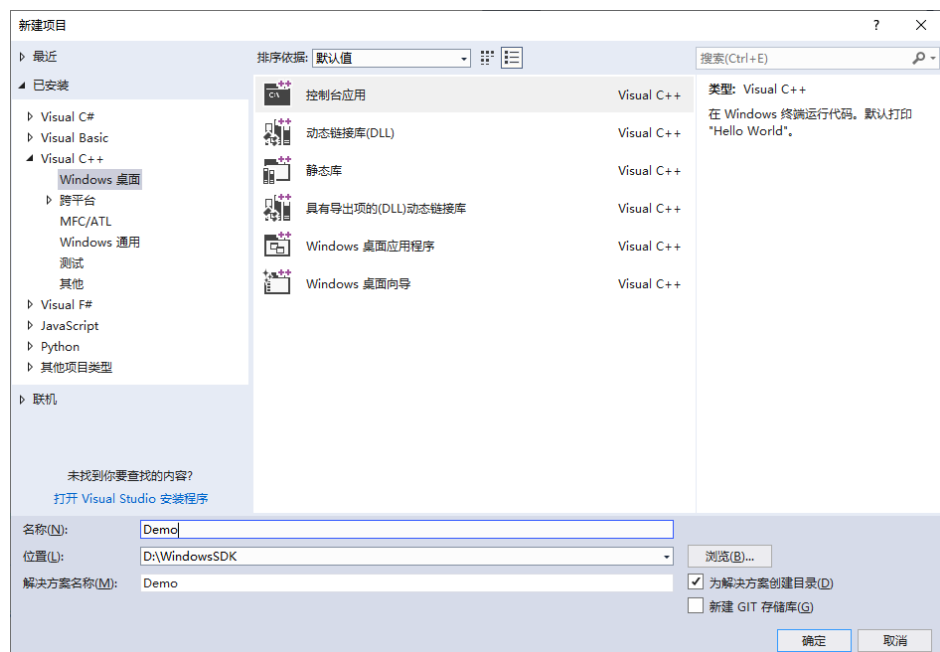
1. 在菜单栏打开“文件”，选择“新建 > 项目”。

图 3-1 新建项目



2. 请根据实际需要，选择项目模板。本节以“控制台应用”为例。

图 3-2 选择项目模板



3. 根据界面提示，配置项目模板。
 - 名称：C++项目名称
 - 位置：项目路径
 - 解决方案名称：工程名称，建议和项目名称保持一致
4. 点击“确定”。
5. 创建Demo.cpp文件，已有则跳过。

图 3-3 创建 cpp 文件

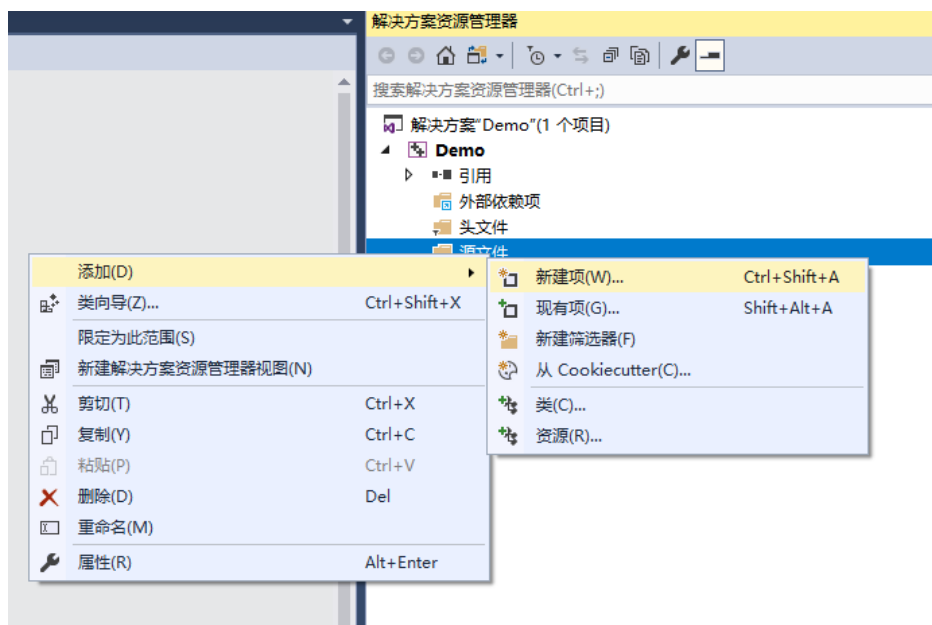
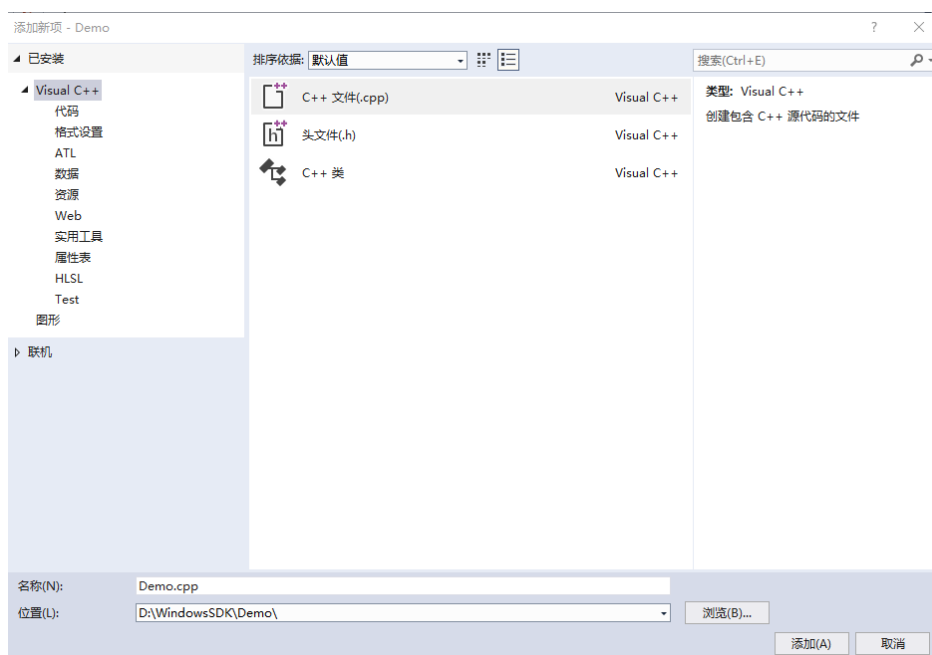
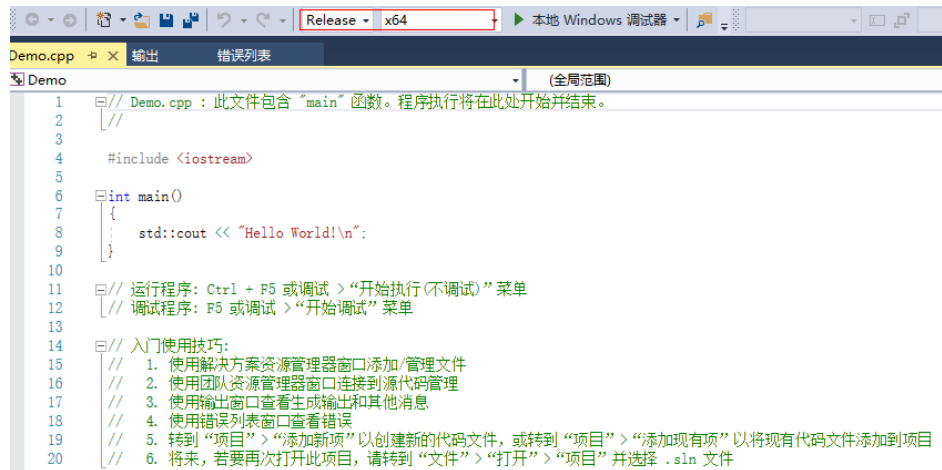


图 3-4 创建 cpp 文件



6. 修改Visual Studio项目配置为“Release”“x64”

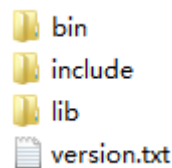
图 3-5 修改项目配置



步骤2 将Windows SDK导入Visual Studio项目。

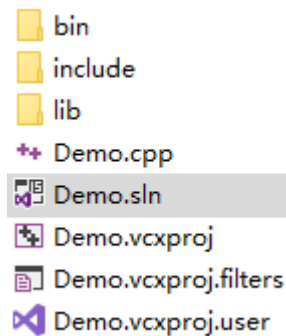
1. 解压下载的Windows SDK.zip文件。

图 3-6 解压 Windows SDK.zip 文件



2. 将bin文件夹、include文件夹、lib文件夹拷贝到项目中。

图 3-7 拷贝 Windows SDK 到项目中



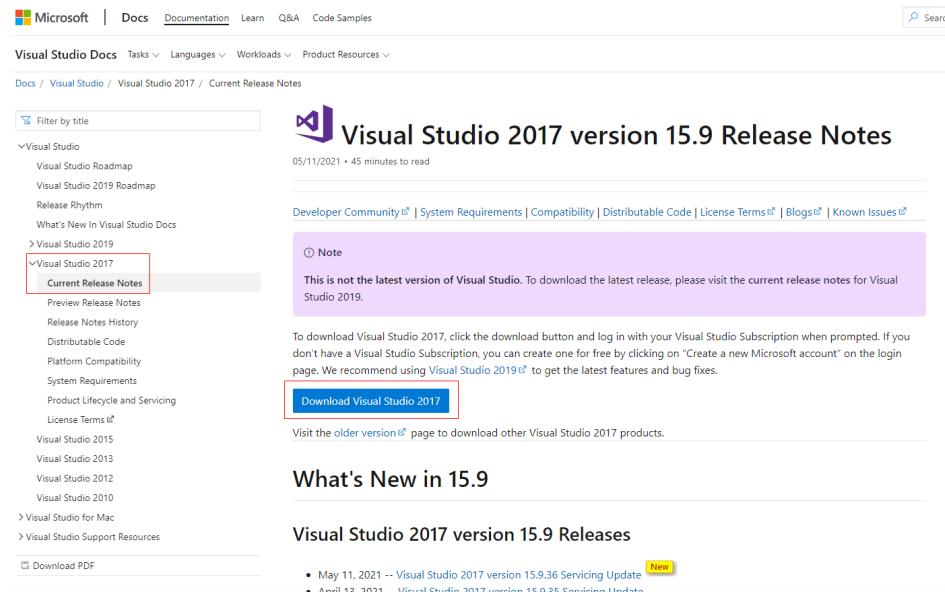
----结束

3.4.3 SDK 使用

集成 SDK

1. 在Demo.cpp文件的main函数中，调用Windows SDK接口完成初始化。
访问<https://cloudvr.obs.cn-east-3.myhuaweicloud.com/example.zip>下载模板example项目。

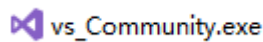
图 3-9 下载 Visual Studio 2017



2. 安装并配置 Visual Studio

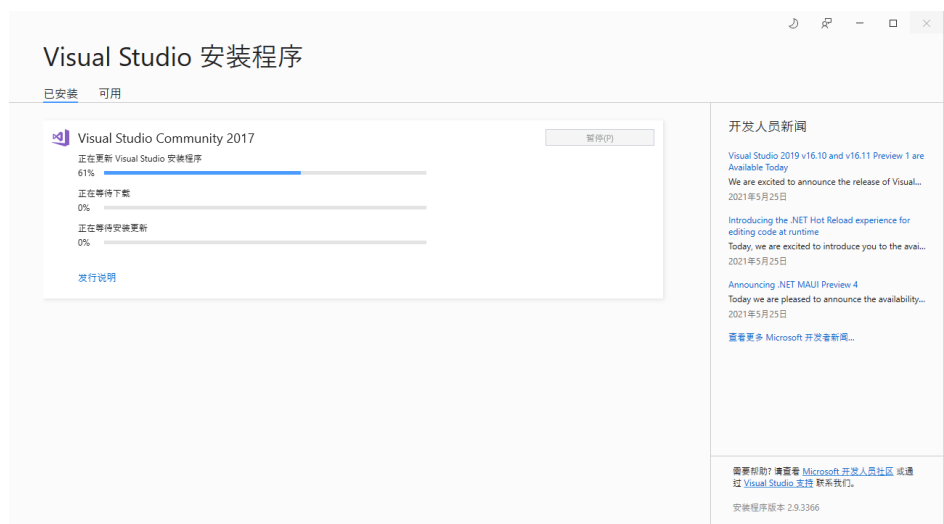
- a. 双击下载的可执行文件进行安装。

图 3-10 双击可执行文件



- b. 等待安装完成。

图 3-11 安装 Visual Studio



3.5.2 错误信息回调

错误信息回调使用的接口为 CvrErrCallback，该接口包含成员函数 onErrorMessage(string errCode, string errorMessage)。

void onErrMsg(string errCode, string errMsg)

注册该方法后，SDK内部识别到错误时，会通过该方法反馈错误信息。入参如下：

- errCode：错误代号或简要的错误信息。
- errMsg：具体的错误信息。

表 3-2 错误场景

errCode	errMsg	描述	处理措施
TOKEN_NULL	IamToken is not obtained.	未获取到IAM TOKEN	请检查网络是否可以访问到IAM服务，且检查账号密码是否正确。
CVR.0005	Device ID is error. It can only contain [a-zA-Z0-9_-].	设备ID格式错误，只允许输入小写英文，数字和-	请更换设备ID
CVR.0009	Project ID is error. It can only contain [a-f0-9-]	项目ID格式错误，只允许输入小写英文a-f，数字和-	请确认传入的项目ID是否正确
CVR.0010	Application ID is error. It can only contain [a-f0-9-].	应用ID格式错误，只允许输入小写英文，数字和-	请确认传入的应用ID是否正确
CVR.0012	The resource id(master id, application id or node id) is not found.	指定资源ID错误，未找到（集群ID，应用ID，云服务器ID）	请确认传入的资源ID是否正确
CVR.0013	The application has no idle servers with plugins installed.	指定应用下无闲置云服务器	当前无可用云服务器，请稍后再试
CVR.0016	This device is forbidden in this application.	指定应用的连接策略禁止您的设备连接	请修改应用的连接策略
CVR.0030	Device brand is error. It can only contain [a-zA-Z0-9_-]	设备品牌格式错误，只允许输入英文大小写，数字和空格_-	请确认设备品牌是否正确
CVR.0036	Forbidden, project_id is error.	鉴权失败，项目ID错误	请确认传入的项目ID是否是API调用区域的

errCode	errMessage	描述	处理措施
CVR.0041	Generate linked token failed.	连接Token生成失败	内部错误，请稍后重试或联系技术支持
CVR.0060	System error, please try again later or contact customer service.	系统错误，请稍后重试或联系客服	请稍后重试或联系技术支持
CVR.0195	Location is not formatted correctly.	调度区域传参错误	请确认传入的调度区域是否正确
CVR.0196	No idle servers exists under regions or available zones of location.	指定location中区域或边缘小站下无闲置云服务器	请确认传入的调度区域是否正确
CVR.0258	Device type is error. It can only be [pc,mobile,vr,ideahub].	设备类型错误，只允许输入 [pc,mobile,vr,ideahub]	请确认传入的设备类型是否正确
CVR.0259	Protocol type is error. It can only be [udp,webrtc,websocket].	协议参数错误，仅支持 [udp,webrtc,websocket]	请确认传入的协议类型是否正确
CVR.0260	Application user is error. It can only contain [a-zA-Z0-9_].	应用使用用户格式错误，只允许输入英文大小写，数字和._-	请确认传入的应用使用用户格式是否正确
CVR.0262	The current user has a render task being dumped under the specified application. Please reconnect after the dump is complete.	当前用户在指定应用下存在一个转储中的渲染任务，请等待转储完成后重新连接	请等待转储完成后重新连接
CVR.0263	The current user or the current device has a render task running under the specified application. Please stop the task and then reconnect. Task id is % (task_id)s.	当前用户或当前设备在指定应用下存在一个渲染任务，请先停止这个任务后重新连接，任务ID为: (task_id)	请停止任务后重新连接

errCode	errMessage	描述	处理措施
CVR.0264	Start 3D application failed. Please reconnect.	启动3D应用失败，请重新连接	请重新连接
CVR.0282	Application launch args is not formatted correctly.	应用启动参数格式错误	请确认传入的应用启动参数格式是否正确

3.5.3 修订记录

发布时间↵	修订记录↵
2021-07-30	修改 <ul style="list-style-type: none">SDK接口参考，SDK1.0.1上线，新增部分SDK接口。
2021-06-09↵	第一次正式发布。

4 Web SDK

4.1 SDK 介绍

华为云VR云渲游平台Web SDK集成华为自研视频传输协议及网络优化算法，为您提供低时延、高可靠的云3D体验。SDK可实现鼠标和键盘的动作数据捕获与预测，并将其上传至云端，待云端完成渲染、编码后，将接收到的3D画面呈现在Web页面上。此外，SDK还提供获取设备标识符等通用接口，便于您在SDK外层封装自身业务逻辑。

4.2 SDK 下载

如需获取VR云渲游平台CVR Web SDK，请执行以下操作：

1. 打开[SDK开发者中心](#)。
2. 选择“VR云渲游Web SDK”进行“SDK下载”。
3. 获取WebSDK.zip文件，并解压。

解压后的文件如表 WebSDK.zip包含的文件所示。

表1 WebSDK.zip包含的文件

文件名	说明
README.md	使用说明
CloudVR-Web-SDK.exe	Example示例服务的启动程序
version.txt	版本号文件
appsettings.json	Example 示例服务程序配置文件(无需修改)
config.yml	Example 示例服务程序配置文件，运行示例前需要配置该文件
wwwroot/	wwwroot为SDK文件夹

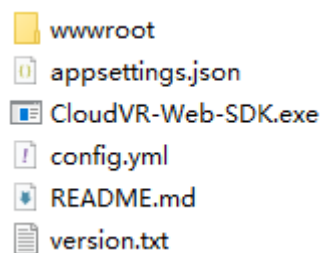
4.3 Example 示例

本章节为测试示例，方便您快速调通整个流程并体验。

具体操作如下：

1. 解压Sdk后的目录。

图 4-1 Sdk 目录



2. 确保环境已经安装 **dotnet 5.0**（若没有安装，运行CloudVR-Web-SDK.exe会出现闪退）。
3. 配置config.yml信息。

表 4-1 参数说明

参数	说明	取值样例
ListenIp	监听ip	0.0.0.0
ListenPort	监听端口	8080
LaunchBrowser	运行CloudVR-Web-SDK.exe后是否自动打开网页	true
AuthCode	web页面访问的验证码	1A2B3c
IAMEndpoint	IAM终端节点	-
SchedulerEndpoint	Scheduler终端节点	-
DomainName	账户名。	-
UserName	IAM用户名。	-
Password	IAM用户登录密码。	-
ProjectName	项目名称。	-
ApplicationId	应用ID，请确保云服务器已经安装该应用。 后续可以根据业务需求自行分配调度。	-

参数	说明	取值样例
ApplicationUser	应用用户名，应用使用者的唯一标识。	TestUser1
ApplicationLaunchArgs	应用启动参数。	-
GpulpType	使用ip类型（公网ip: public, 私网ip: private, 私网ip使用场景: 专线, vpn）	public
Resolution	分辨率	1080p

具体信息获取方式如下：

- 登录管理控制台。
- 鼠标悬停在右上角的用户名，选择下拉列表中的“我的凭证”。
- 在“API凭证”页面，查看IAM用户名、账户名、项目名称信息。
- 在左侧服务列表中选择“VR云渲游平台”，在“应用管理”中选择应用ID。如果没有应用，请先创建应用。
- 在IAM_API终端节点查找区域对应的终端节点信息，选择全局终端节点或资源所在区域的终端节点。

图 4-2 IAM 的终端节点

表1 IAM的终端节点

区域名称	区域	终端节点 (Endpoint)
全局	global	iam.myhuaweicloud.com

- 在地区和终端节点查找CVR服务对应终端节点信息，根据资源所在区域选择相应的终端节点。

图 4-3 终端节点信息

区域名称	区域	终端节点 (Endpoint)	协议类型
global	global	cvr.myhuaweicloud.com	HTTPS
华北-北京四	cn-north-4	cvr.cn-north-4.myhuaweicloud.com	HTTPS
华东-上海一	cn-east-3	cvr.cn-east-3.myhuaweicloud.com	HTTPS
华南-广州	cn-south-1	cvr.cn-south-1.myhuaweicloud.com	HTTPS
西南-贵阳一	cn-southwest-2	cvr.cn-southwest-2.myhuaweicloud.com	HTTPS

- 运行CloudVR-Web-SDK.exe可执行文件。
- 打开浏览器访问：<http://127.0.0.1:8080>。
- 在浏览器页面输入验证码，单击“登录”。

4.4 客户端开发

4.4.1 开发前准备

前提条件

- 已注册华为云，并开通VR云渲游平台公测权限。
- 已按照《VR云渲游平台快速入门》完成创建集群、创建应用、创建云服务器、配置连接策略（可选）等操作。
- 请确认已仔细阅读过 API参考/使用前必读（[如何调用API_VR云渲游平台CVR_API参考_使用前必读](#)）
- 已获取并记录应用的appID信息，获取appID方法请参见[APP ID的获取方法](#)。

4.4.2 SDK 导入

WebSdk需要客户结合自身业务需求搭建web服务器并集成sdk使用。

自行搭建一个web服务器，将wwwroot下的文件拷贝到web服务器的静态资源目录下。

4.4.3 SDK 使用

1. 获取X-Subject-Token和 ProjectId需要用到华为云账号、密码、项目名称，请参考[获取IAM用户Token](#)。
2. 通过wwwroot/js/init.js文件，获取device_id, device_type, device_brand, protocol_type参数信息。具体可参考wwwroot/login.html文件。
3. 获取gpu_ip, port, link_token, task_id。需要用到X-Subject-Token, device_id, device_type, device_brand, protocol_type。X-Subject-Token需要添加在请求头中，请参考[VR云渲游平台节点调度API](#)。
4. 确保安全组配置中已放通所有UDP端口、TCP/10100-10101、TCP/10200-10201、TCP/10300-10301端口。

4.4.4 SDK 接口

接口	描述	是否必选
Init.getDevice_id()	【功能说明】 获取设备id，用于作为渲染调度接口的请求参数。 详细使用请参考wwwroot/login.html文件。	是
Init.getDevice_type()	【功能说明】 获取设备类型，用于作为渲染调度接口的请求参数。 详细使用请参考wwwroot/login.html文件。	是

Init.getDevice_brand()	【功能说明】 获取设备品牌，用于作为渲染调度接口的请求参数。 详细使用请参考wwwroot/login.html文件。	是
Init.getProtocol_type()	【功能说明】 获取协议类型，用于作为渲染调度接口的请求参数。 详细使用请参考wwwroot/login.html文件。	是
cloudDesktopApp.loadApp()	【功能说明】 启动并加载云应用。 【请求参数】 详细使用请参考wwwroot/index.html文件。	是
cloudDesktopApp.on()	【功能说明】 回调函数。 <ul style="list-style-type: none">● ready: 连接就绪。● exit: 退出回调。● streamPause: 串流画面暂停回调。● streamResume: 串流画面恢复回调。● appClosed: 远程app应用关闭回调。	否
cloudDesktopApp.appEnterFullscreen()	【功能说明】 进入全屏。	否
cloudDesktopApp.appExitFullscreen()	【功能说明】 退出全屏。	否
cloudDesktopApp.restartApp()	【功能说明】 重启远程app。	否
cloudDesktopApp.pauseStream()	【功能说明】 暂停串流画面。	否
cloudDesktopApp.resumeStream()	【功能说明】 恢复串流画面。	否

4.4.5 启动应用

启动应用，调用wwwroot/js/cloudapp.js文件中的cloudDesktopApp()函数加载渲染画面，请参考wwwroot/index.html文件。

4.4.6 关闭应用

关闭应用请参考[VR云渲游平台关闭渲染任务API](#)。

4.5 附录

4.5.1 修订记录

发布时间↵	修订记录↵
2021-08-23	修改 <ul style="list-style-type: none">• 4.3 Example示例，更新说明。• 4.4.4 SDK接口，增加回调函数。
2021-06-10↵	第一次正式发布。