**NOLO VR Android SDK**

**インターフェースについて**

NOLO Co., Ltd

2018年3月

目次

[1. 紹介 3](#_Toc510200845)

[1.1 NOLOについて 3](#_Toc510200846)

[1.2 NOLO CV1について 3](#_Toc510200847)

[1.3 NOLO HOMEについて 3](#_Toc510200848)

[1.4 NOLO VR Unity SDKについて 4](#_Toc510200849)

[1.5 モバイル端末のSDK構造 4](#_Toc510200850)

[2. 接続準備 6](#_Toc510200851)

[2.1 DLL及びDLL読込み 6](#_Toc510200852)

[2.2 権限登録 7](#_Toc510200853)

[3. インターフェースについて 7](#_Toc510200854)

[3.1 基本情報 7](#_Toc510200855)

[3.2 位置情報 9](#_Toc510200856)

[4. 注意事項 12](#_Toc510200857)

[4.1 原点位置を設定する。 12](#_Toc510200858)

[4.2 設置 AppKey 13](#_Toc510200859)

[4.3 getInstance NoloVR事例取得 13](#_Toc510200860)

# 紹介

## NOLOについて

NOLOはVRボードゲームの素晴らしい体験及びVR移動の利便性を結びつけ、次世代のVR移動システムのインタラクティブな操作方法を改めて定義する。

現在、NOLOは世界で計8,700万個のVRヘルメットに対応でき、非常に市場の将来性がある。それから、NOLOは国内外で数多くの一体型VRヘルメット、PC VR、AR、サービスロボット及びドローンの関連会社と連携している。

## NOLO CV1について

NOLO CV1はNOLO Inc.自社で開発された世界初の完全没入型VR/AR 製品で、BS1つ、ヘルメットロケーター1つ及びコントローラー2つを含む。

NOLO CV1は現在市販されるVRヘルメット、一部のPC VR、ARのヘルメット及びNibiruOSの一体型に対応でき、空間定位及びインタラクティブの機能を支える。ユーザーは初めて使用する場合、簡単に設置すれば、仮想の現実世界を体験し、コントローラーで仮想世界との多様なインタラクションを体験することができる。

## NOLO HOMEについて

NOLO HOME は NOLO傘下のモバイルVRプラットフォームであり、世界初の携帯電話用6-DoFモバイルVRプラットフォームである。ユーザーはNOLO HOME 、NOLO CV1 及び携帯電話を利用して、時間や場所を問わず、無線で完全な没入型VR体験を享受することができる。一般消費者のために斬新なゲームプレー方法を提供する。現在、Androidのユーザーはダウンロードできる。iOS 版もまもなく使用できる。

## NOLO VR Unity SDKについて

NOLO VR Unity SDKはUnityの開発者がNOLO設備ﾃﾞｰﾀを取得し、Cardboard及びGearVR のSDKを搭載し、モバイル6DoF VRゲームを開発するために、NOLO Inc.により開発されたものである。

## モバイル端末のSDK構造

NOLO SDKはC/S構造を採用する。NOLO HOMEはServeであり、NOLO CV1設備ﾃﾞｰﾀを読込み、処理する。ゲームはClientである。SDKはAIDLによってゲームとNOLOHOMEの間にﾃﾞｰﾀを転送させ、NOLO CV1設備ﾃﾞｰﾀを取得する。ゲームはUSB設備ﾃﾞｰﾀを読み込む必要がなく、NOLO HOMEとの接続のみによってﾃﾞｰﾀの送受信をする。(図 1)

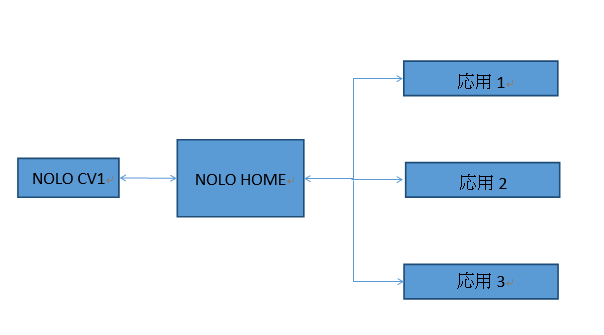


図 1

NOLO VR Android SDKはAIDL及びNOLO HOMEでプロセス間通信を実現することによって、設備ﾃﾞｰﾀを取得する。getInstance()でNOLOVRの実例を取得する。ゲーム開始前にisStallServer()でユーザーがNOLO HOMEをインストールしたかどうかを判断する。インストールを確認された場合、openServer()でServerをバインディングする。ゲーム終了後、closeServer()でバインディングを外す。NOLO VR Unity SDK及びNOLO VR UE4 SDKはNOLO VR Android SDKに対して、部分のインタフェースに対してカプセル化する。開発者はインターフェースを通じて、位置及びボタン値を取得すれば、ゲームを開発することができる。 (図2).

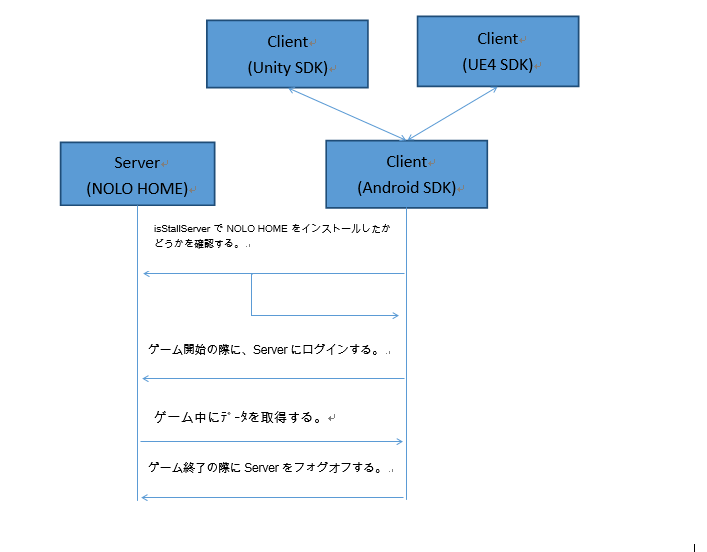


図 2

# 接続準備

## DLL及びDLL読込み

Android直接読込み androidsdkclient-release.aar。

## 権限登録

以下のAndroidManifest.xml権限を登録する。

<uses-permission android:name="android.permission.BROADCAST\_STICKY" />

<uses-permission android:name="android.permission.PACKAGE\_USAGE\_STATS" />

<uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION" />

<uses-permission android:name="android.permission.ACCESS\_WIFI\_STATE" />

<uses-permission android:name="android.permission.INTERNET" />

<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />

<uses-permission android:name="com.android.launcher.permission.WRITE\_SETTINGS" />

<uses-permission android:name="android.permission.WRITE\_APN\_SETTINGS" />

# インターフェースについて

## 基本情報

|  |  |
| --- | --- |
| 関数名 | public static NoloVR getInstance(Context context)； |
| 機能説明 | 取得NoloVR |
| パラメーター | context: Activity対象に伝えなければならない、さもなくば関連ヒントボックスには表示されません |
| 戻り値 | NoloVR instance |
| 先決条件 |  |

|  |  |
| --- | --- |
| 関数名 | public boolean isStallServer()； |
| 機能説明 | インストールNOLO HOME Serverを行うかどうかを判断する。すべてのデータをする必要があるインストールServer。 |
| パラメーター |  |
| 戻り値 | bool |
| 先決条件 | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| 関数名 | public void setAppKey(String appKey)； |
| 機能説明 | 設置appkey |
| パラメーター | String 種類appKey。開発者はプラットフォームから関連パラーメーターを取得する。 |
| 戻り値 |  |
| 先決条件 | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| 関数名 | public void openServer()； |
| 機能説明 | NOLO HOME Serverデータバインディング。Serverデータバインディングができた場合、遠距離からServerにアクセスすることができる。 |
| パラメーター |  |
| 戻り値 |  |
| 先決条件 | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| 関数名 | public void closeServer()； |
| 機能説明 | バインディングを外すServer。ログオフの際にバインディングを外すServer。 |
| パラメーター |  |
| 戻り値 | bool |
| 先決条件 | NoloVR getInstance(Context context) |

## 位置情報

|  |  |
| --- | --- |
| 関数名 | NTrackedDevicePose getPoseByDeviceType(int type); |
| 機能説明 | 設備仕様に基づき位置情報を取得する。 |
| パラメーター | 0:ヘルメット・ロケーター  1:左側コントローラー  2:右側コントローラー  3: BS |
| 戻り値 | NTrackedDevicePose |
| 先決条件 | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| 関数名 | int getElectricityByDeviceType(int type) ; |
| 機能説明 | 設備仕様によって設備の消費電力情報を取得する。 |
| パラメーター | 0:ヘルメット・ロケーター  1:左側コントローラー  2:右側コントローラー  3: BS |
| 戻り値 | int |
| 先決条件 | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| 関数名 | NControllerState getControllerStatesByDeviceType(int type); |
| 機能説明 | ボタン情報 |
| パラメーター | 0:ヘルメット・ロケーター  1:左側コントローラー  2:右側コントローラー  3: BS |
| 戻り値 | NControllerState |
| 先決条件 | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| 関数名 | void SetHmdTrackingCenter(float x,float y,float z); |
| 機能説明 | ヘルメットの回転中心を設定する（廃止） |
| パラメーター | (legacy function, not to be used anymore) |
| 戻り値 |  |
| 先決条件 | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| 関数名 | void triggerHapticPulse(int deviceType,int intensity); |
| 機能説明 | 振動 |
| パラメーター | Deice type:  0:ヘルメット・ロケーター  1:左側コントローラー  2:右側コントローラー  3: BS  Vibration intensity:  0~100 (int) |
| 戻り値 |  |
| 先決条件 | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| 関数名 | void setHmdType(int hmdType); |
| 機能説明 | ヘルメットの種類を設定する |
| パラメーター | 0: Gear VR  1: DayDream  2: CardBoard  3: その他 |
| 戻り値 |  |
| 先決条件 | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| 関数名 | void reportError(String errorLog); |
| 機能説明 | ｹﾞｰﾑの中に収集したエラーログ情報 |
| パラメーター | String errorLog: ｹﾞｰﾑの中に収集したエラーログ情報 |
| 戻り値 |  |
| 先決条件 | NoloVR getInstance(Context context) |

|  |  |
| --- | --- |
| 関数名 | int add(int i,int j) |
| 機能説明 | 遠隔操作サービスの運行をテストする。 |
| パラメーター | 二つのintタイプ |
| 戻り値 | 戻り値が「-1」の場合、遠隔操作のバインディングができていない。戻り値がi+j の場合、遠隔操作のバインディングができた。 |
| 先決条件 | NoloVR getInstance(Context context) |

# 注意事項

## 原点位置を設定する。

NOLO正常に稼動する場合、ヘルメットを床に置き、ロケーターのボタンを押せば、当位置がｹﾞｰﾑの初期位置となる。即ち、ゲームエンジン“”の位置。原点位置を設定すれば、設備に登録される。BSの位置が大きく変更した場合のみ、改めて原点位置を設定する必要がある。

## 設置 AppKey

セッティングappkeyの後、ｹﾞｰﾑプレイができる。appkeyそれは開発者がプラットフォームを利用する際に自動生成される。

ゲームがNOLO HOMEをアップロードしないときは、この公開Appkeyを開発テストに使用することができます。

公開Appkey：4e4f4c4f484f4d457eff82725bc694a5。

## getInstance NoloVR事例取得

は呼び出しを答応するように構成する前に、まず呼び出しisStallServerかどうか判断してインストールしたNOLOホーム、ゲーム開始時、呼び出しopenServer答応するように構成する、ゲーム終瞭時、呼び出しcloseServer終瞭する。