AssociatorPlugin 插件说明

* 插件的接口函数
* Get 插件对象的获取函数

FAssociatorPluginModule::Get()

* 初始化函数

InitAssociatoData(riGameId 游戏的编号Id);

* 具体的功能函数

1. 获得用户的基本信息

GetAssociatoBaseInfo(FString rMacStr,FEventCustomBaseInFoResponse rEventCustomBase);

1. 保存用户游戏数据
2. 直接把游戏数据拼凑成字符串

SaveGameData(FString rMacStr,FString const & rResult);

1. 用字符串数组的形式来组织游戏数据

SaveGameData(FString rMacStr,TArray<FString> const & rArrayResult);

1. 用对象来组织游戏用户的数据(需要继承UGameFightDataBase这个对象)

SaveGameData(FString rMacStr,UGameFightDataBase \*pFightData)

1. 获得用户的游戏数据(http接口还没有定下来暂时不要调用)
2. 如果是用字符串后者字符数组组织的游戏数据获得的方法

GetGameData(FString rMacStr,FEventOnGameDataResponse rEventGameFightInfo)

1. 用对象方式组织的游戏数据获得方法(rOutDataClass 为UGameFightDataBase子类)

GetGameData(FString rMacStr,UClass \*rOutDataClass,FEventOnGameDataResponse rEventGameFightInfo)

1. json 转 结构

JsonStringToUStruct 字符串转 结构

JsonStringToUStruct 结构转json字符串

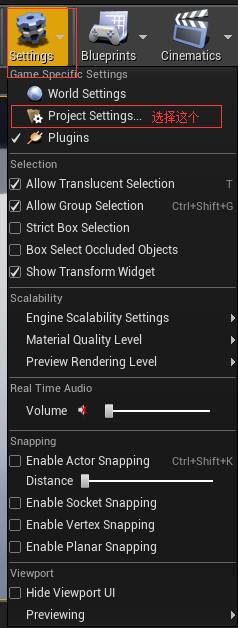
* 事件
* FEventCustomBaseInFoResponse

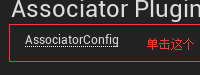
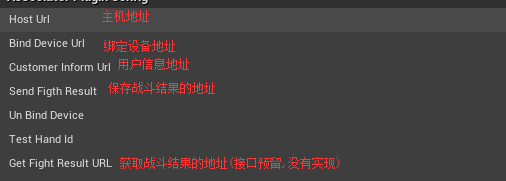
当获得了用户的基本信息的手调用这个事件

* FEventOnGameDataResponse

当获得了游戏数据的时候调用这个事件

* 结构及对象(详细说明见HttpStructDef.h)
* CustomBaseInFo 玩家的基本信息
* GameDataInfo 游戏数据的信息
* UGameFightDataBase 如果用对象来组织游戏数据 需要继承这个对象
* 文件配置
* 编辑器配置



1. 
2. 

* 文本配置

在项目目录下Config\DefaultAssociatorPlugin.ini 这个文件下配置

