

LarkXR DataChannel SDK 3.2.3.1

简述

通过调用本 SDK 与实现网页/移动客户端实现数据交互功能
当前 SDK 支持服务端版 3.2.2.0+

类型定义

XR_SUCCESS	成功
XR_ERROR_INTERFACE_FAILED	无法创建通信接口
XR_ERROR_SERVER_UNSupport	当前应用，服务端不支持 datachannel
XR_ERROR_PARAM	参数错误
XR_ERROR_Opreation	操作错误

接口描述

lr_client_register_taskstatus_callback

```
LARKXR_API void DC_CALL lr_client_register_taskstatus_callback(on_taskstatus taskstatus, void* user_data = NULL);
```

功能描述:

方便应用调用方获取客户端的连接状态, 以及当前应用的 taskid

参数描述:

1. on_taskstatus: 响应通知的回调函数
2. user_data: 传入用户数据指针, 方便面向对象语言(C++/C#)回调

lr_client_start

```
LARKXR_API int DC_CALL lr_client_start(const char* taskid, on_connected cb_connected, on_data cb_data, on_disconnected cb_disconnected, void* user_data);
```

功能描述:

创建通信接口

参数描述:

1. taskid: 为了兼容旧版本 SDK 而保留的参数, 通常可以传 NULL
2. on_connected: 异步回调通知是否连接成功服务端, 禁止为 NULL
3. on_data: 接收到 web 或其他客户端发来的数据, 禁止为 NULL
4. on_error: 链接断开回调函数, 禁止为 NULL, 以及出错
5. user_data: 传入用户数据指针, 方便面向对象语言(C++/C#)回调

返回值参考类型定义

lr_client_send

```
LARKXR_API int DC_CALL lr_client_send(enum data_type type, const char* data, size_t size);
```

功能描述:

发送数据, 支持字符串和二进制数据

参数描述:

1. type: 发送数据类型, 字符串或者二进制数据
2. data: 数据指针
3. size: 数据长度 (字符串建议包含'\0'的长度)

返回值:

小于 0 代表发送失败参考类型定义, 等于发送长度代表发送成功

lr_client_stop

```
LARKXR_API void DC_CALL lr_client_stop();
```

功能描述:

停止通信接口

lr_client_register_aivoice_callback

功能描述:

注册智能语音交互通信回调接口

参数描述:

1. on_aivoice_callback:服务端推送智能语音结构体.

```
struct AiVoicePacket
{
    bool url;                //true :online audio url(mp3) .false: audio pack (pcm)
    unsigned int voice_id;    //语音ID
    const char* online_url;   //如果url为true,该字段为url地址,否则该字段为NULL
    int url_size;             //url长度 包含\0
    const char* nlq;          //当前语音对讲的文本
    int nlq_size;             //对讲文本长度 包含\0
    //如果URL为false 那么下面字段描述每一个pcm包
    unsigned int slice_id;     //一个语音分片ID
    int samples_per_sec;      //eg.16000
    int channels;             //eg.1
    const char* audio;        //数据包指针,如果 url 为true 该字段为空
    int size_byte;            //每一包的字节数
    bool last_packet;         //是否为最后一包
};
```

DEMO

[查看 C++](#) 或 [unity3D](#) 文件夹