

# AnyChat SDK for Mac

## 开发流程指南

(V9.0 版本)



广州佰锐网络科技有限公司

GuangZhouBaiRui Network Technology Co.,Ltd.

<http://www.bairuitech.com> <http://www.anychat.cn>

2021 年 6 月

## 目 录

|     |                              |    |
|-----|------------------------------|----|
| 一、  | 简介.....                      | 3  |
| 1.1 | 面向的读者.....                   | 4  |
| 1.2 | 获取 ANYCHAT SDK FOR MAC.....  | 4  |
| 1.3 | 技术支持.....                    | 5  |
| 二、  | 编写说明.....                    | 6  |
| 三、  | 工程准备.....                    | 7  |
| 3.1 | 添加开发框架.....                  | 7  |
| 3.2 | 导入库文件.....                   | 8  |
| 3.3 | 导入 SDK API 头文件.....          | 8  |
| 四、  | 基本流程.....                    | 10 |
| 4.1 | 准备一个 ANYCHATPLATFORM 对象..... | 10 |
| 4.2 | 监听基本事件.....                  | 10 |
| 4.3 | 初始化 SDK.....                 | 12 |
| 4.4 | 连接、登录服务器.....                | 12 |
| 4.5 | 进入房间.....                    | 12 |
| 五、  | 音视频交互.....                   | 14 |
| 5.1 | 设置必要的参数.....                 | 14 |
| 5.2 | 摄像头硬件初始化.....                | 14 |
| 5.3 | 打开本地音视频.....                 | 15 |
| 5.4 | 关闭本地音视频.....                 | 16 |
| 5.5 | 请求远程音视频.....                 | 16 |
| 5.6 | 关闭远程音视频.....                 | 17 |
| 六、  | 资源释放.....                    | 18 |
| 七、  | 附录.....                      | 19 |

# 一、简介

AnyChat SDK(AnyChat 音视频互动开发平台)是一套跨平台的音视频即时通讯解决方案，基于先进的 H.264 视频编码标准、AAC 音频编码标准与 P2P 技术，支持高清视频，整合了佰锐科技在音视频编码、多媒体通讯领域领先的开发技术和丰富的产品经验而设计的高质量、宽适应性、分布式、模块化的网络音视频互动平台。

基于 Mac 的客户端 SDK 应用于 OS X 10.7 以上版本的桌面 PC（包括 MBP、iMac 等），您可以通过该套 SDK API 接口实现在 Mac 平台快速开发基于音视频通讯交互功能的 Mac 程序，主要提供的功能如下：

- 音视频即时通讯：提供语音、视频一对一、一对多的实时通讯，支持高清视频和高品质音频效果。
- 录像：支持针对单个人的音视频录制、整个视频通话过程内容的合成音视频录制以及集中服务器保存录制
- 抓拍：可对本地视频和正在视频的对象进行抓拍；
- 文字聊天：支持多用户之间的文字交流；
- 透明通道：提供客户端之间、客户端跟服务器之间的数据通讯能力；
- 文件传输：支持客户端直接、客户端跟服务器之间的文件传输功能，支持断点续传；
- 动态设置音视频参数：提供音视频参数设置的接口，可以根据需要动态设置分辨率、码率、帧率等视频参数，满足各种应用场景的需求；
- 外部音视频输入：支持非标准采集设备以外的音视频源输入，满足更多的应用场景；
- 集成第三方外部音视频编解码器：可集成第三方音视频编解码器，满足特殊环境下面的硬件编解码要求；

## 1.1 面向的读者

《AnyChat SDK for Mac 开发流程指南》文档是提供给具有一定的 Mac 编程经验和了解面向对象概念的读者使用，不要求具备音视频开发方面的经验。您在使用遇到任何问题，都可以通过访问 [bbs.anychat.cn](http://bbs.anychat.cn) 反馈给我们。

## 1.2 获取 AnyChat SDK for Mac

您可在 AnyChat 的产品官方网站下载到最新版的 AnyChat for Mac SDK，下载地址为: <http://www.anychat.cn/download.html>，如下图所示：



AnyChat for Mac SDK 包里面提供了 demo 程序的编译程序、开发指南、demo 程序源码和 SDK 文件，其解压之后的目录结构如下所示：

|         |                       |
|---------|-----------------------|
| ----bin | AnyChat SDK 演示程序（安装包） |
| ----doc | 客户端开发指南               |
| ----src | 示例程序源代码               |
| ----sdk | 客户端 SDK 引用文件          |

## 1.3 技术支持

在您使用本 SDK 的过程中，遇到任何困难，请与我们联系，我们将热忱为您提供帮助。

您可以通过如下方式与我们取得联系：

- 1、在线论坛：<http://bbs.anychat.cn/>
- 2、知识中心：<http://www.anychat.cn/faq/>
- 3、官方网站：<http://www.anychat.cn>
- 4、电子邮件：[service@bairuitech.com](mailto:service@bairuitech.com)
- 5、24 小时客服电话：+86 （020） 85276986、38109065、38103410

## 二、编写说明

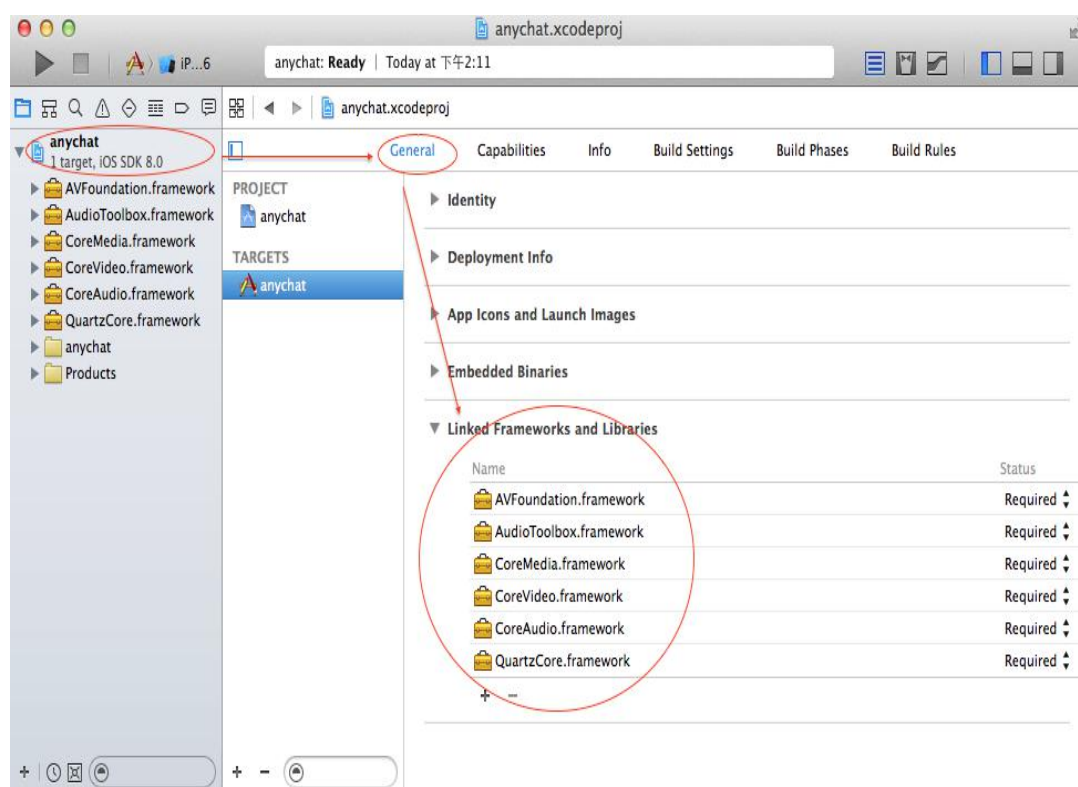
本指南的编写是为了帮助使用 AnyChat for Mac SDK 的用户快速地搭建 SDK 开发环境、熟悉 SDK 开发流程、掌握 SDK 开发功能接口而编写的，其中“工程准备”、“基本流程”、“音视频交互”三章的内容是基于 src\helloAnyChat 目录中提供的 helloAnyChat 工程来编写说明的，涉及到的开发环境配置、以及相关代码说明可以参考 helloAnyChat 工程源码。

## 三、工程准备

Mac 开发需要在 MAC OSX 系统环境下进行，开发工具有很多，开发者可根据自己的喜好进行选择。在此，我们推荐开发者使用 XCode 作为自己的开发工具，本套开发指南也是针对 XCode 开发环境下进行编写的。在 XCode 中新建一个 Mac 工程，对工程进行以下配置，搭建 AnyChat 的开发环境。

### 3.1 添加开发框架

- 1) 点击在 XCode 工程左侧资源管理器中的工程图标。在右侧菜单 TARGETS—>General—>Link Frameworks and Libraries 的路径里，点击 “+” 号增加系统框架。如下图所示：

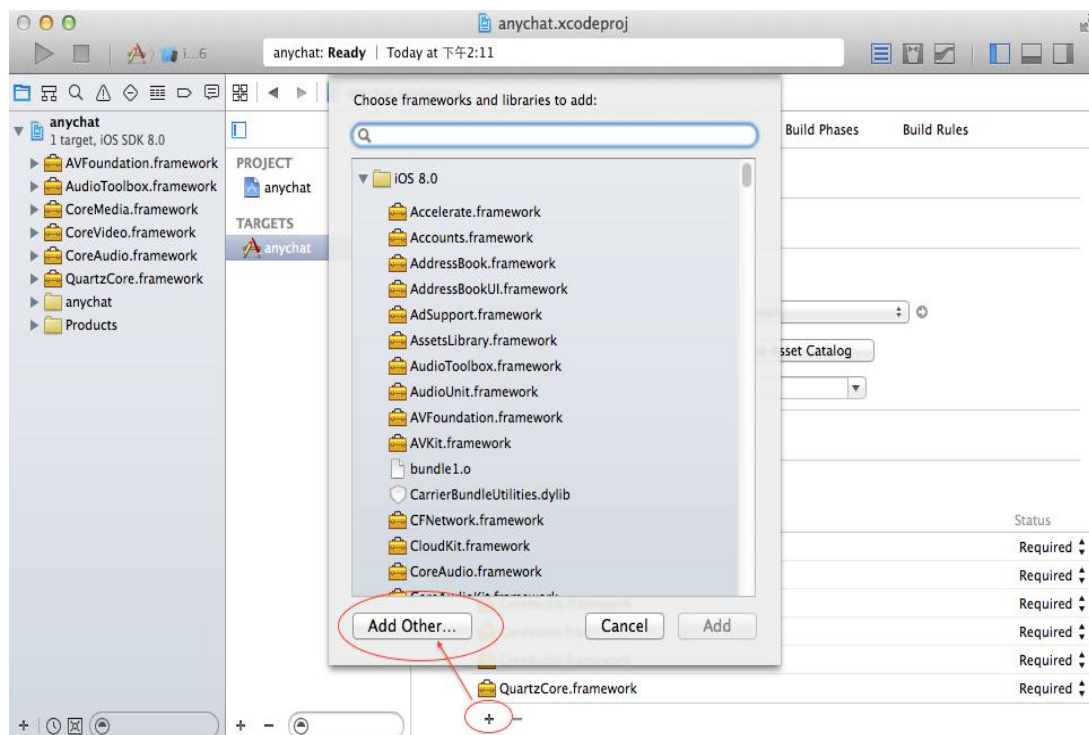


- 2) 工程需要用到的框架包括有：

|                        |                        |
|------------------------|------------------------|
| AVFoundation.framework | AudioToolbox.framework |
| QuartzCore.framework   | CoreMedia.framework    |
| CoreVideo.framework    | CoreAudio.framework    |

## 3.2 导入库文件

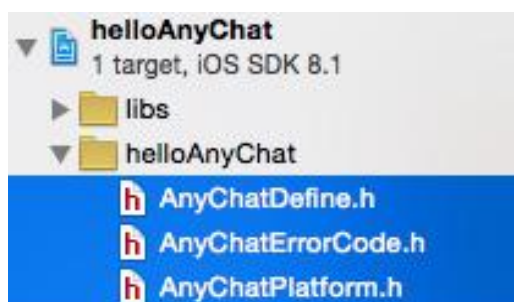
- 1) 在添加开发框架步骤的同一处，点击“Add other”选项导入 AnyChat SDK 库文件。库文件在 SDK 开发包的“sdk\libs\”和“sdk\libs\audioprocess”目录里。如下图所示：



- 2) 导入 C++ 的静态库 libc++.dylib、libstdc++.dylib 和 libstdc++.6.dylib。

## 3.3 导入 SDK API 头文件

- 1) 把开发包“\sdk\include”目录里的 SDK API 头文件添加到工程资源管理器中，操作后的工程目录如下图所示：





- 2) 在工程中使用到 AnyChat API 的代码里，引用 SDK API 头文件，参考代码如下：

```
#import "AnyChatPlatform.h"
#import "AnyChatDefine.h"
#import "AnyChatErrorCode.h"
```

## 四、基本流程

在工程准备好了之后，需要实现以下基本流程，才能调用音视频交互等其他功能接口。

### 4.1 准备一个 AnyChatPlatform 对象

AnyChatPlatform 类是 SDK 的核心类，提供各种功能接口，如连接服务器、登录、进入房间、操作音视频等，在使用这些功能接口构建应用之前，需要准备一个 AnyChatPlatform 对象。

新建一个 AnyChatPlatform 对象，参考代码如下：

```
//新建核心类对象
AnyChatPlatform *anychat = [AnyChatPlatform new];
```

### 4.2 监听基本事件

运用通知中心机制，实现监听“连接服务器、用户登录、进入房间、与服务

器网络连接”等事件。在需要接收的类中实现以下两步。

(1) 在所需要监听的类的.h 头文件里引用 AnyChat 通知消息代理，参考代码如下：

```
//AnyChat通知消息代理
@interfaceAnyChat : UIViewController<AnyChatNotifyMessageDelegate>
```

使用 AnyChat 通知消息<AnyChatNotifyMessageDelegate>异步消息事件协议时，必须实现处理回调信息的 7 个方法，具体方法如下：

```
// 连接服务器消息
- (void) OnAnyChatConnect:(BOOL) bSuccess;
// 用户登陆消息
- (void) OnAnyChatLogin:(int) dwUserId : (int) dwErrorCode;
// 用户进入房间消息
- (void) OnAnyChatEnterRoom:(int) dwRoomId : (int) dwErrorCode;
// 房间在线用户消息
- (void) OnAnyChatOnlineUser:(int) dwUserNum : (int) dwRoomId;
// 用户进入房间消息
- (void) OnAnyChatUserEnterRoom:(int) dwUserId;
// 用户退出房间消息
- (void) OnAnyChatUserLeaveRoom:(int) dwUserId;
// 网络断开消息
- (void) OnAnyChatLinkClose:(int) dwErrorCode;
```

(2) 在监听回调类的-(void)viewDidLoad 方法里注册通知中心, 并实现消息观察者方法和设置消息回调事件接收者, 参考代码如下:

```
@property (strong, nonatomic) AnyChatPlatform *anyChat;
- (void)viewDidLoad
{
    [super viewDidLoad];
    //注册通知中心
    [[NSNotificationCenter defaultCenter] addObserver:self
selector:@selector(AnyChatNotifyHandler:) name:@"ANYCHATNOTIFY"
object:nil];
    //初始化SDK
    anyChat = [[AnyChatPlatform alloc] init];
    //AnyChat通知消息代理(回调事件接收者)
    anyChat.notifyMsgDelegate = self;
}
//消息观察者方法
- (void)AnyChatNotifyHandler:(NSNotification*)notify
{
    NSDictionary* dict = notify.userInfo;
    [anyChat OnRecvAnyChatNotify:dict];
}
```

## 4.3 初始化 SDK

加载资源，应用程序中只需要执行一次，其他的功能接口都必须在初始化之后才能正常使用，参考代码如下：

```
//初始化AnyChatSDK
[AnyChatPlatform InitSDK:0];
```

## 4.4 连接、登录服务器

使用 AnyChat 功能的通讯功能，必须要先连接登录 AnyChat 的通讯服务器，部署 AnyChat 通讯服务器的部署参考链接：<http://bbs.anychat.cn/forum.php?mod=viewthread&tid=8&extra=page%3D1>，这里以我们对外公开测试服务器地址 demo.anychat.cn 为例。连接、登录服务器参考代码如下：

```
//连接服务器,第一个参数为服务器地址,第二参数为端口。
[AnyChatPlatform Connect:@"demo.anychat.cn":@"8906"];
//用户登录(userName 变量: 登录用户名)
[AnyChatPlatform Login:userName:@""];
```

这步流程的执行都是异步的操作，会触发相应的消息回调：

```
// 连接服务器消息
- (void) OnAnyChatConnect:(BOOL)bSuccess;
// 用户登陆消息
- (void) OnAnyChatLogin:(int)dwUserId :(int)dwErrorCode;
```

## 4.5 进入房间

除了音视频交互功能需要本流程之外，没有特殊说明，其他功能都不需要本流程。应用层将 roomid 传入，进入指定的房间，只有在同一个房间内的用户才能进行音视频交互，参考代码如下：

```
//进入房间
[AnyChatPlatform EnterRoom:1 :@""];
```

此流程操作是一个异步的操作，会依次触发 OnAnyChatEnterRoom、OnAnyChatOnlineUser 消息回调，参考代码如下：

```
//用户进入房间消息(dwRoomId表示房间号，dwErrorCode=0表示进入房间成功)
- (void) OnAnyChatEnterRoom:(int) dwRoomId : (int) dwErrorCode{
}

// 房间在线用户消息(在线用户进入房间成功后调用一次。dwUserNum当前房间总人数,包含自己)
- (void) OnAnyChatOnlineUser:(int) dwUserNum : (int) dwRoomId{
}
```

## 五、音视频交互

AnyChat for Mac SDK 为开发者提供了便捷的建立音视频通讯的接口，通过以下几步操作，即可在您的应用中集成音视频交互功能。需要注意的是只有在同一个房间内的用户才能进行音视频通讯。

### 5.1 设置必要的参数

在初始化之后，设置必要的音视频参数，参考代码如下：

```
//设置本地视频采用 Overlay 模式
[AnyChatPlatform SetSDKOptionInt:BRAC_SO_LOCALVIDEO_OVERLAY :1];
//设置本地视频采集随着设备而旋转而处理
[AnyChatPlatform SetSDKOptionInt:BRAC_SO_LOCALVIDEO_ORIENTATION :self.interfaceOrientation];
```

### 5.2 摄像头硬件初始化

这个操作需要在显示视频的类中执行，步骤如下：

(1) 在显示类的.h 头文件中引用 AVFoundationk 框架，参考代码如下：

```
//获取本地视频显示框架
#import <AVFoundation/AVFoundation.h>
```

(2) 在显示类的.m 文件中实现初始化操作方法，代码如下：

```
//创建视频显示层全局变量
AVCaptureVideoPreviewLayer *localVideoSurface;
//AnyChat SDK 自动调用“摄像头硬件初始化”方法
- (void) OnLocalVideoInit:(id)session
{
    //通过 session 控制设备的视频数据输入和输出流向
    localVideoSurface = [AVCaptureVideoPreviewLayerlayerWithSession:
    (AVCaptureSession*)session];
    //视频显示层 UI 设置
    localVideoSurface.frame = CGRectMake(0, 0, 120, 160);
    localVideoSurface.videoGravity = AVLayerVideoGravityResizeAspectFill;
    //视频显示层添加到自定义的 theLocalView 界面显示视图中。
    [self.theLocalView.layeraddSublayer:self.localVideoSurface];
}
```

## 5.3 打开本地音视频

打开本地音视频数据需要在进入房间成功之后才有效，即在收到 OnAnyChatEnterRoom 回调后（参考 4.5）打开本地音视频，其他客户端才能请求到你的音视频数据。调用 UserCameraControl 打开视频，调用 UserSpeakControl 打开音频。参考代码如下：

```
//打开本地音频(参数“-1”表示本地用户，也可以用本地的真实userid)
[AnyChatPlatform UserSpeakControl: -1:YES];
//设置本地视频UI (“0”为默认适配视频显示位置与尺寸大小)
[AnyChatPlatform SetVideoPos:-1 :self :0 :0 :0 :0];
//打开本地视频(参数“-1”表示本地用户，也可以用本地的真实userid)
[AnyChatPlatform UserCameraControl:-1 : YES];
```

执行此操作之后，会自动调用 OnLocalVideoInit 方法，在此方法中可以进行摄像头的初始化操作（参考 5.2）。

## 5.4 关闭本地音视频

打开本地音视频后，可以在音视频交互的过程中选择关闭本地音视频。同时，还可以在关闭之后重新打开本地音视频（参考 5.3）；在音视频交互结束之后需要调用该操作，释放本地摄像头和音频采集设备，参考代码如下：

```
//关闭本地音频
[AnyChatPlatform UserSpeakControl: -1 : NO];
//关闭本地视频
[AnyChatPlatform UserCameraControl: -1 : NO];
```

执行此操作之后，AnyChat SDK 自动调用“摄像头硬件资源释放”方法 `OnLocalVideoRelease`，参考代码如下：

```
- (void) OnLocalVideoRelease:(id)sender
{
    // “localVideoSurface” 表示视频显示层全局变量（参考 4.2.2）
    if(localVideoSurface) {
        localVideoSurface = nil;
    }
}
```

## 5.5 请求远程音视频

在触发 `OnAnyChatOnlineUser` 或者 `OnAnyChatEnterRoom` 并判断通话目标对象已经进入当前房间之后（参考 4.2），该操作才有效。调用 `UserCameraControl` 打开目标对象视频，调用 `UserSpeakControl` 打开目标对象音频；

`SetVideoPos` 绑定指定的 `NSImageView` 用于显示视频。（如要显示多个人的视频，侧要为每一位目标对象用户绑定一个不同指针地址的 `NSImageView` 即可），参考代码如下：



```
//打开当前房间在线目标对象的音视频，需要传入它的userid  
[AnyChatPlatform UserSpeakControl:userid:YES];  
//绑定目标对象视频显示在自定义的remoteVideoSurface  
UIImageView*remoteVideoSurface;  
//“0”参数：目标对象视频显示位置与尺寸大小  
[AnyChatPlatform SetVideoPos:userid:remoteVideoSurface:0:0:0:0];  
//打开目标用户视频  
[AnyChatPlatform UserCameraControl:userid :YES];
```

## 5.6 关闭远程音视频

请求远程音视频后，可以在音视频交互的过程中选择关闭远程音视频。同时，还可以在关闭之后重新请求远程音视频（参考 5.5）；在音视频交互结束之后需要调用该操作，释放远程音视频资源，参考代码如下：

```
//关闭远程音频，userid 为远程目标用户 userid  
[AnyChatPlatform UserSpeakControl: userid : NO];  
//关闭远程视频  
[AnyChatPlatform UserCameraControl: userid: NO];
```

## 六、资源释放

### (1) 离开房间

释放当前房间内的音视频资源。参考代码如下：

```
// “-1” 表示离开当前房间  
[AnyChatPlatform LeaveRoom:-1];
```

在音视频交互结束后，可调用该操作。离开当前房间之后，可再次选择进入指定房间。

### (2) 退出

断开与 AnyChat 通讯服务器连接。参考代码如下：

```
//断开与服务器的连接  
[AnyChatPlatform Logout];
```

在需要断开跟 AnyChat 服务器通讯连接的时候，可调用该操作。退出之后，可以再次调用连接、登录服务器。

### (3) 释放 SDK

释放整个 SDK 资源。参考代码如下：

```
//释放资源  
[AnyChatPlatform Release];
```

建议在退出整个应用程序的时候调用该操作。释放 SDK 之后，需要重新初始化 SDK 之后才能进行连接、登录、进入房间等操作。

## 七、附录

AnyChat for Mac SDK 包里提供的 HelloAnyChat 程序（源码在“src\helloAnyChat”目录下）运行效果如下图所示：

