iVMS-8800 平台 SDK 说明

目录

1 iVMS-8800 平台 SDK 简介.....3
2 iVMS-8800 平台接口详细说明.....4
2.1 初始化和资源释放.....4
2.1.1 初始化资源.....4
2.1.2 释放资源.....4
2.2 用户登录、退出.....4
2.2.1 用户登录.....4
2.2.2 退出登录.....4
2.3 获取错误信息.....5
2.4 数据集查询、遍历（设备列表、录像文件等）.....5
2.4.1 设备查询.....5
2.4.2 获取权限接口.....5
2.4.3 录像文件查询.....6
2.4.4 游标向后移动.....6
2.4.5 获取查询信息字符串属性.....6
2.4.6 获取查询信息整形属性.....7
2.5 浏览实时视频.....7
2.5.1 查询实时流 URL.....7
2.5.2 播放实时视频.....7
2.5.3 停止实时视频.....8
2.6 云镜控制.....9
2.6.1 云镜控制.....9
2.7 视频参数获取、设置.....9
2.7.1 视频参数获取.....9
2.8 打开、关闭视频声音.....10
2.8.1 打开视频声音.....10
2.8.2 关闭视频声音.....10
2.9 获取预览、回放控件取流 XML.....11
2.9.1 查询预览控件取流 xml.....11
2.9.2 查询预览控件取流 xml.....11
2.9.3 查询回放控件取流的 xml.....11
2.9.4 查询回放控件取流的 xml.....12
2.10 录像回放、下载.....12
2.10.1 播放控制.....12
2.10.2 开始文件下载.....13
2.10.3 停止文件下载.....13
2.10.4 获取文件回放、下载百分比.....14
2.11 开启、停止手动录像.....14
2.12 本地抓图、录像.....14
2.12.1 视频抓图.....14
2.12.2 开始本地录像.....15
2.12.3 停止本地录像.....15

2.13 对讲.....15
2.13.1 开始对讲.....15
2.13.2 停止对讲.....15
2.14 报警控制.....16
2.14.1 报警输出控制.....16
2.14.2 获取报警输出当前状态.....16
2.14.3 订阅报警.....16
2.14.4 取消订阅报警.....17
2.15 环境量实时数据.....17
2.15.1 订阅实时数据.....17
2.15.2 取消订阅实时数据.....18
3 iVMS-8800 平台数据定义.....19
3.1 属性定义.....19
3.2 云台控制参数说明.....21
3.3 错误码定义.....21
3.4 权限说明.....22
3.5 结构体定义.....22
3.5.1 资源类型.....22
3.5.2 回放命令定义.....23
3.5.3 巡航结构体定义.....23
4 iVMS-8800 平台编程引导.....24
4.1 登录退出流程.....24
4.2 资源查询流程.....24
4.3 实时预览流程.....26
4.4 回放下载流程.....27

1 iVMS-8800 平台 SDK 简介

iVMS-8800 平台 SDK 与 iVMS-8800 平台配套使用，其功能主要是调用者从 iVMS-8800 平台获取数据资源，然后利用 SDK 提供的外部接口，以一定的调用流程实现实时监控、录像回放下载、云镜控制以及获取环境量数据和报警数据等诸多功能。同时，本 SDK 可以辅助 iVMS-8800 平台的预览和回放控件使用，即第三方平台集成预览和回放控件时，通过本 SDK 从平台中获取某监控点的预览或回放 XML 数据，然后将 XML 数据写入预览和回放控件对应的接口，实现实时预览和录像回放的功能，这种方式尤其适合于 web 开发环境。第三方开发人员使用 iVMS-8800SDK 前请务必仔细阅读本文档，本文档不仅提供了接口以及数据的详细说明，还提供了调用流程和示例代码，将为您的开发提供极大的方便。

2 iVMS-8800 平台接口详细说明

2.1 初始化和资源释放

2.1.1 初始化资源

1.1 Plat_Init

函数名称	int Plat_Init()
功能描述	初始化平台接口
参数说明	无
返回值	成功返回 0，错误时返回-1
备注	初始化接口只需调用一次，在调用任何接口前，必须调用 **Plat_Init** 完成初始化工作。

2.1.2 释放资源

1.2 Plat_Free

函数名称	int Plat_Free()
功能描述	资源释放接口，关闭所有的连接
参数说明	无
返回值	成功时返回 0 错误时返回-1
备注	释放资源，全局只调用一次。

2.2 用户登录、退出

2.2.1 用户登录

1.3

函数名称	int Plat_LoginCMS(const char * csCmsIp, const int iCmsPort, const char * csUserName, const char * csPSW)	
功能描述	登录到指定平台	
参数说明	cscmsIp	iVMS-8800 中心管理服务器地址(如 “172.7.150.2”)
	ncmsPort	iVMS-8800 中心管理服务器端口(默认 80)
	csUserName	用户名
	csPSW	用户密码
返回值	>0 成功，返回一个全局的会话句柄，错误时返回-1	
备注	1.该接口只需调用一次，在调用完 Plat_Init()后调用该接口完成用户登录 iVMS-8800 平台工作； 2. 如果平台 SDK 与 iVMS-8800 平台运行在同一 IP 地址的计算机，cscmsIp 请勿为 127.0.0.1	

2.2.2 退出登录

1.4 Plat_LogoutCMS

函数名称	int Plat_LogoutCMS (int iUserhandle)	
功能描述	退出登录	
参数说明	iUserhandle	Plat_LoginCMS 返回的句柄
返回值	成功时返回 0 错误时返回-1	
备注	该接口只需调用一次，在 Plat_Free 调用前调用该接口完成用户登出 iVMS-8800 平台工作	

2.3 获取错误信息

1.5 Plat_GetLastError

函数名称	int Plat_GetLastError()
功能描述	获取错误信息
参数说明	无
返回值	成功时返回 0 错误时返回错误码
备注	错误码定义见附表

2.4 数据集查询、遍历（设备列表、录像文件等）

2.4.1 设备查询

1.6 Plat_ QueryResource

函数名称	int Plat_QueryResource(int iResourceType, int iUserHandle)	
功能描述	查询用户有权限的所有指定类型的设备信息	
参数说明	iResourceType	设备类型 0x00, 中心 0x01, 区域 0x02, 设备 0x03, 摄像头 0x04, 报警输入 0x05, 报警输出 0x06, 录像文件 0x07, 环境量
	iUserHandle	Plat_LoginCMS 返回的句柄
返回值	成功时返回 0 错误时返回-1	
备注	查询成功后使用 MoveNext()， GetValueStr()， GetValueInt()逐条获取设备信息	

2.4.2 获取权限接口

1.7 Plat_ QueryPrivilege

函数名称	int Plat_QueryPrivilege(const char * csResourceID, const char * csPrivilegeCode, int iUserHandle)

功能描述	查询用户对指定设备的权限	
参数说明	csResourceID	设备 ID
	csPrivilegeCode	权限代码
	iUserHandle	Plat_LoginCMS 返回的句柄
返回值	有权限返回 1　无权限返回 0	
备注	权限代码的定义见附表	

2.4.3 录像文件查询

1.8 Plat_QueryRecordFile

函数名称	int Plat_QueryRecordFile(const char* csCameraID, const long lStartTime, const long lEndtime, const char* csRecordType, const char* csQueryCondition int iUserHandle)	
功能描述	查询指定监控点某一时间段内的录像文件信息	
参数说明	csCameraID	监控点 ID
	lStartTime	查询录像的开始时间（标准时间换算为秒的数值）
	lEndtime	查询录像的结束时间（标准时间换算为秒的数值）
	csRecordType	录像类型 1 计划录像, 2 移动侦测, 4 手动录像, 16 报警录像
	csQueryCondition	查询条件 0 为 IPSAN,　1 为设备,　2 为 PCNVR,　3 为 ENVR
	iUserHandle	Plat_LoginCMS 返回的句柄
返回值	成功时返回 0 错误时返回-1	
备注	查询成功后使用 MoveNext()，GetValueStr()，GetValueInt()逐条获取录像信息(录像片段的 开始时间 结束时间 录像URL)	

2.4.4 游标向后移动

1.9 Plat_MoveNext

函数名称	int Plat_MoveNext(int iUserHandle)	
功能描述	向后移动数据访问游标	
参数说明	iUserHandle	Plat_LoginCMS 返回的句柄
返回值	返回 0 数据未读取完，返回-1 数据读取完成	
备注		

2.4.5 获取查询信息字符串属性

1.10 Plat_GetValueStr

函数名称	const char*　Plat_GetValueStr(const char* propertyName, int iUserHandle)	
功能描述	获取查询信息的字符串属性	
参数说明	propertyName	属性名称
	iUserHandle	Plat_LoginCMS 返回的句柄

返回值	字符串属性值，如果没有返回空字符串
备注	

2.4.6 获取查询信息整形属性

1.11 Plat_GetValueInt

函数名称	int Plat_GetValueInt(const char* propertyName, int iUserHandle)	
功能描述	获取查询信息的整形属性	
参数说明	propertyName	属性名称
	iUserHandle	Plat_LoginCMS 返回的句柄
返回值	整形类型属性，获取属性失败返回 INT_MIN	
备注		

2.5 浏览实时视频

2.5.1 查询实时流 URL

1.12 Plat_ QueryRealStreamURL

函数名称	const char * Plat_QueryRealStreamURL(int iUserHandle, const char* csCameraID, int nStreamType)	
功能描述	查询实时流视频的 URL	
参数说明	iUserHandle	Plat_LoginCMS 返回的句柄
	csCameraID	监控点 ID
	nStreamType	码流类型: 0 主码流，1 子码流，3 根据平台实际配置
返回值	成功时返回 URL 字符串 错误时返回空字符串	
备注		

2.5.2 播放实时视频

1.13 Plat_ PlayVideo

函数名称	int Plat_PlayVideo(const char* URL, long hWnd, int iUserHandle, void(__stdcall*fStreamCallback) (int handle,const char* data,int size,void *pUser), void *pUser)	
功能描述	打开播放视频，同时针对预览和回放	
参数说明	URL	预览或回放的播放路径，由 Plat_QueryRealStreamURL 或 Plat_QueryRecordFile 获得
	hWnd	播放窗口句柄，如果为空，则不播放

	iUserHandle	Plat_LoginCMS 返回的句柄
	fStreamCallback	视频码流接收回调函数指针，如果回调函数为 NULL 则不给码流 回调函数参数： handle Plat_PlayVideo 返回的句柄 data 接收视频码流数据缓冲区指针 size 接收视频码流数据字节数 pUser 用户数据
	pUser	用户数据
返回值	>=0 成功，返回实时视频句柄，错误时返回-1	
备注	fStreamCallback用于调用者额外处理流数据	

1.14 Plat_ PlayRecordVideo

函数名称	int Plat_ PlayRecordVideo (const char* URL, long hWnd, int iUserHandle, const long lStartTime, const long lEndTime, void(__stdcall *fStreamCallback) (int handle,const char* data,int size,void *pUser), void *pUser)	
功能描述	播放视频，只针对回放	
参数说明	URL	回放的播放路径，由 Plat_QueryRecordFile 获得
	hWnd	播放窗口句柄，如果为空，则不播放
	iUserHandle	Plat_LoginCMS 返回的句柄
	lStartTime	录像播放的开始时间
	lEndTime	录像播放的结束时间
	fStreamCallback	视频码流接收回调函数指针，如果回调函数为 NULL 则不给码流 回调函数参数： handle Plat_ PlayRecordVideo 返回的句柄 data 接收视频码流数据缓冲区指针 size 接收视频码流数据字节数 pUser 用户数据
	pUser	用户数据
返回值	>=0 成功，返回实时视频句柄，错误时返回-1	
备注	fStreamCallback回调函数用于调用者额外处理流数据	

2.5.3 停止实时视频

1.15 Plat_ StopVideo

函数名称	int Plat_StopVideo(int hStream)
功能描述	停止视频播放

参数说明	hStream	Plat_PlayVideo 返回的句柄
返回值	成功时返回 0 错误时返回-1	
备注	当调用者调用 Plat_ PlayRecordVideo 按时间进行录像回放时，hStream 为 Plat_ PlayRecordVideo 返回的句柄	

2.6 云镜控制

2.6.1 云镜控制

1.16 Plat_ ControlCamera

函数名称	int Plat_ ControlCamera (const char* cameraID, int dwPTZCommand, int param1, int param2, int param3, int param4, LPVOID *pCruiseInfo, int iUserHandle)	
功能描述	停止视频播放	
参数说明	cameraID	监控点 ID
	dwPTZCommand	控制命令
	param1	1:开始动作 0:停止动作，当 dwPTZCommand 为 8,9,39 时，param1 为预置位序号，最多 255,具体和球机有关,当 dwPTZCommand 为 37,38 时，param1 为巡航轨迹编号
	Param2	云台速度或巡航速度或巡航点停顿时间，从 1 到 7，数字越大，速度越快
	Param3	预留
	Param4	预留
	pCruiseInfo	巡航结构体
	iUserHandle	Plat_LoginCMS 返回的句柄
返回值	成功时返回 0 错误时返回-1	
备注	控制命令见附表	

2.7 视频参数获取、设置

2.7.1 视频参数获取

1.17 Plat_GetVideoParam

函数名称	int Plat_GetVideoParam (int hStream, int &iBValue, int &iCValue, int &iSValue, int &iHValue)
功能描述	播放控制接口

参数说明	hStream	Plat_PlayVideo 返回的句柄
	iBValue	亮度，默认 64； 范围 0-128
	iCValue	对比度，默认 64； 范围 0-128
	iSValue	饱和度，默认 64； 范围 0-128
	iHValue	色调，默认 64； 范围 0-128
返回值	成功时返回 TRUE 错误时返回 FALSE	
备注		

1.18 Plat_SetVideoParam

函数名称	int Plat_SetVideoParam (int hStream, int iBValue, int iCValue, int iSValue, int iHValue)	
功能描述	播放控制接口	
参数说明	hStream	Plat_PlayVideo 返回的句柄
	iBValue	亮度，默认 64； 范围 0-128
	iCValue	对比度，默认 64； 范围 0-128
	iSValue	饱和度，默认 64； 范围 0-128
	iHValue	色调，默认 64； 范围 0-128
返回值	成功时返回 TRUE 错误时返回 FALSE	
备注		

2.8 打开、关闭视频声音

2.8.1 打开视频声音

1.19 Plat_OpenSound

函数名称	int Plat_OpenSound (int hStream)	
功能描述	播放控制接口	
参数说明	hStream	Plat_PlayVideo 返回的句柄
返回值	成功时返回 TRUE 错误时返回 FALSE	
备注		

2.8.2 关闭视频声音

1.20 Plat_CloseSound

函数名称	int Plat_CloseSound (int hStream, BOOL bManual)	
功能描述	播放控制接口	
参数说明	hStream	Plat_PlayVideo 返回的句柄
	bManual	是否手动关闭
返回值	成功时返回 TRUE 错误时返回 FALSE	
备注		

2.9 获取预览、回放控件取流 XML

2.9.1 查询预览控件取流 xml

函数名称	int Plat_QueryPreViewXmlForOCX(const char* csCameraID, int iUserHandle, char *pBuffer, unsigned unBufferSize)	
功能描述	查询预览控件取流 xml	
参数说明	csCameraId	监控点 ID
	iUserHandle	登录 CMS 返回的句柄
	pBuffer	输出，存放 xml 的缓冲区
	unBufferSize	缓冲区长度
返回值	成功返回实际使用缓冲区的长度，错误时返回-1	
备注		

2.9.2 查询预览控件取流 xml

函数名称	int Plat_QueryPreViewXmlForOCXEx(const char*csCameraIndexcode, int iUserHandle, char *pBuffer, unsigned unBufferSize)	
功能描述	查询预览控件取流 xml	
参数说明	csCameraIndexcode	监控点编号
	iUserHandle	登录 CMS 返回的句柄
	pBuffer	输出，存放 xml 的缓冲区
	unBufferSize	缓冲区长度
返回值	成功返回实际使用缓冲区的长度，错误时返回-1	
备注	与 Plat_QueryPreViewXmlForOCX 功能一致，只是传入参数略有不同	

2.9.3 查询回放控件取流的 xml

函数名称	int Plat_QueryPreBackXmlForOCX(const char* csCameraID, int iUserHandle, char *pBuffer, unsigned unBufferSize, int nSelWnd, int nQueryCondition, int nRecordType, const char* strBeginTime, const char* strEndTime)	
功能描述	查询预览控件取流 xml	
参数说明	csCameraId	监控点 ID
	iUserHandle	登录 CMS 返回的句柄
	pBuffer	输出，存放 xml 的缓冲区
	unBufferSize	缓冲区长度

	nSelWnd	焦点窗口号，选中的窗口
	nQueryCondition	查询类型，0 为 IPSAN， 1 为设备，2 为 PCNVR，3 为嵌入式存储(ENVR)
	nRecordType	录像类型，1:计划录像，2:移动录像，4:报警录像，16:手动录像，或以上任意几种类型和
	strBeginTime	查询录像开始时间
	strEndTime	查询录像结束时间
返回值	成功返回实际使用缓冲区的长度，错误时返回-1	
备注		

2.9.4 查询回放控件取流的 xml

函数名称	int Plat_QueryPreBackXmlForOCXEx(const char* csCameraIndexcode, int iUserHandle, char *pBuffer, unsigned unBufferSize, int nSelWnd, int nQueryCondition, int nRecordType, const char* strBeginTime, const char* strEndTime)	
功能描述	查询预览控件取流 xml	
参数说明	csCameraId	监控点 ID
	iUserHandle	登录 CMS 返回的句柄
	pBuffer	输出，存放 xml 的缓冲区
	unBufferSize	缓冲区长度
	nSelWnd	焦点窗口号，选中的窗口
	nQueryCondition	查询类型，0 为 IPSAN， 1 为设备，2 为 PCNVR，3 为嵌入式存储(ENVR)
	nRecordType	录像类型，1:计划录像，2:移动录像，4:报警录像，16:手动录像，或以上任意几种类型和
	strBeginTime	查询录像开始时间
	strEndTime	查询录像结束时间
返回值	成功返回实际使用缓冲区的长度，错误时返回-1	
备注	与 Plat_QueryPreBackXmlForOCX 功能一致，只是传入参数略有不同	

2.10 录像回放、下载

2.10.1 播放控制

1.21 Plat_ PlayControl

函数名称	int Plat_PlayControl(int hStream, int iCommand, int iParam)

功能描述	播放控制接口	
参数说明	hStream	Plat_PlayVideo 返回的句柄
	iCommand	播放控制命令 PLAY_START 开始播放 PLAY_PAUSE 暂停播放 PLAY_FAST 加快播放速度 PLAY_SLOW 减慢播放速度 PLAY_OFFSET 播放定位
	iParam	控制参数
返回值	成功时返回 0 错误时返回-1	
备注	iCommand 为 PLAY_OFFSET 时 iParam 表示文件偏移百分比 0~100，大华录像不支持速度控制。	

2.10.2 开始文件下载

1.22 Plat_ StartDownLoad

函数名称	int Plat_StartDownLoad(const char * csURL, const char * csSaveFilePath, const long lStartTime, const long lEndtime, unsigned __int64 ullMAXFileSize, int iUserHandle)	
功能描述	录像文件下载	
参数说明	csURL	录像文件的 URL
	csSaveFilePath	本地文件保存全路径
	lStartTime	下载录像的开始时间，为 0 则按录像默认开始时间
	lEndtime	下载录像的结束时间，为 0 则按录像默认结束时间
	ullMAXFileSize	分包的大小，如 256*1024*1024,分包为 256MB
	iUserHandle	Plat_LoginCMS 返回的句柄
返回值	>=0 成功，返回文件下载句柄，<0 失败	
备注	查询成功后使用 MoveNext()， GetValueStr()， GetValueInt()逐条获取录像信息(录像片段的 开始时间 结束时间 录像URL)	

2.10.3 停止文件下载

1.23 Plat_ StopDownLoad

函数名称	int Plat_StopDownLoad (int handle)	
功能描述	停止文件下载	
参数说明	handle	Plat_StartDownLoad 返回的句柄
返回值	成功时返回 0 错误时返回-1	
备注		

2.10.4 获取文件回放、下载百分比

1.24 Plat_ GetFilePercent

函数名称	int Plat_GetFilePercent(int handle, long* pVal)	
功能描述	获取文件下载百分比	
参数说明	handle	Plat_StartDownLoad 返回的句柄
	pVal	文件下载百分比指针，范围为 0-100
返回值	成功时返回 0 错误时返回-1	
备注		

2.11开启、停止手动录像

1.25 Plat_ManualRecord

函数名称	int Plat_ManualRecord (int nManualRecordType, const char* csCameraID, int iUserHandle, int nRecordPos, char *pResultBuffer, unsigned unResultBufferSize)	
功能描述	开启、停止手动录像	
参数说明	nManualRecordType	1 表示启动手动录像，0 表示停止手动录像
	csCameraID	监控点 ID
	iUserHandle	Plat_LoginCMS 返回的句柄
	nRecordPos	录像存储位置 0 为 IPSAN 1 为设备 2 为 PCNVR 3 为嵌入式存储(ENVR) 7 为 CVR
	pResultBuffer	启动、停止录像的时间格式 2013-02-28 16：16：16
	unResultBufferSize	字符串 pResultBuffer 长度
返回值	失败返回<0，成功返回>=0，如果=0 说明已经该受到录像存在，>0 表示实际 pResultBuffer 长度	
备注	需要启动、停止手动录像的监控点必须在平台中配置录像计划	

2.12本地抓图、录像

2.12.1 视频抓图

1.26 Plat_ CapPic

函数名称	int Plat_CapPic(int handle,const char* savefilepath)	
功能描述	视频抓图	
参数说明	handle	Plat_PlayVideo 返回的句柄
	savefilepath	本地文件保存全路径
返回值	成功时返回 0 错误时返回-1	

备注	仅支持BMP格式抓图

2.12.2 开始本地录像

1.27 Plat_ StartRecordFile

函数名称	int Plat_StartRecordFile(int handle, const char* savefilepath, unsigned __int64 ullMAXFileSize)	
功能描述	开始本地录像	
参数说明	handle	Plat_PlayVideo 返回的句柄
	savefilepath	本地文件保存全路径
	ullMAXFileSize	分包的大小，如 256*1024*1024,分包为 256MB
返回值	>=0 成功，返回录像句柄，错误时返回-1	
备注		

2.12.3 停止本地录像

1.28 Plat_ StopRecordFile

函数名称	int Plat_StopRecordFile(int handle)	
功能描述	停止本地录像	
参数说明	handle	Plat_StartRecordFile 返回的句柄
返回值	成功时返回 0 错误时返回-1	
备注		

2.13对讲

2.13.1 开始对讲

1.29 Plat_ StartTalk

函数名称	long Plat_StartTalk(const char* DeviceID, int iChannelID, int iUserHandle)	
功能描述	开始对讲	
参数说明	DeviceID	设备 ID
	iChannelID	通道 ID
	iUserHandle	Plat_LoginCMS 返回的句柄
返回值	>=0 成功，返回对讲句柄，错误时返回-1	
备注	全局只能同时有一个对讲	

2.13.2 停止对讲

1.30 Plat_ StopTalk

函数名称	int Plat_StopTalk(long lTalkHandle)
功能描述	停止对讲

参数说明	lTalkHandle	Plat_StartTalk 返回的句柄
返回值	成功时返回 0 错误时返回-1	
备注		

2.14报警控制

2.14.1 报警输出控制

1.31 Plat_ AlarmOut

函数名称	int Plat_AlarmOut(const char* csAlarmOutID, int iCommand, int iUserHandle)	
功能描述	报警输出控制	
参数说明	csAlarmOutID	报警输出 ID
	iCommand	控制命令 OPEN_ALARMOUT 打开报警输出 CLOSE_ALARMOUT 关闭报警输出
	iUserHandle	Plat_LoginCMS 返回的句柄
返回值	成功时返回 0 错误时返回-1	
备注		

2.14.2 获取报警输出当前状态

1.32 Plat_ GetAlarmOutState

函数名称	int Plat_GetAlarmOutState(const char* csAlarmOutID, int iUserHandle)	
功能描述	获取报警输出当前状态	
参数说明	csAlarmOutID	报警输出 ID
	iUserHandle	Plat_LoginCMS 返回的句柄
返回值	1 为打开 0 为关闭 错误时返回-1	
备注		

2.14.3 订阅报警

1.33 Plat_ SubscribeAlarm

函数名称	int Plat_SubscribeAlarm(void(__stdcall *fAlarmCallback) (const char* csResourceid, int alarmtype, const char* csAlarmDetail, int iAlarmLevel, const char* alarmtime, int action, void *pUser), void* pUser,

	int iUserHandle);	
功能描述	订阅报警信息	
参数说明	fAlarmCallback	告警接收回调函数指针 回调函数参数： csResourceid 设备资源 ID alarmtype 告警类型 csAlarmDetail 报警描述 iAlarmLevel 报警等级 alarmtime 告警时间 action 1：告警产生 0：告警消除 pUser 用户数据
	pUser	用户数据
	iUserHandle	Plat_LoginCMS 返回的句柄
返回值	成功时返回 0 错误时返回-1	
备注		

2.14.4 取消订阅报警

1.34 Plat_ UnSubscribeAlarm

函数名称	int Plat_UnSubscribeAlarm(int iUserHandle)	
功能描述	取消订阅报警	
参数说明	iUserHandle	Plat_LoginCMS 返回的句柄
返回值	成功时返回 0 错误时返回-1	
备注		

2.15 环境量实时数据

2.15.1 订阅实时数据

1.35 Plat_SubscribeRealTimeData

函数名称	Int Plat_SubscribeRealTimeData(void(__stdcall *fRealTimeDataCallback) (int iResourceId, int iTypeValve, const char* csTypeName, float fDataValve, const char* csAlarmTime, void * pUser), void * pUser, int iUserHandle);	
功能描述	订阅实时数据	
参数说明	fRealTimeDataCallback	实时数据值接收回调函数指针 函数参数： iResourceId 资源 id 号

		iTypeValue 资源类型值 csTypeName 资源类型名称 fDataValue 实时数据值 csAlarmTime 发生时间 pUser 用户数据
	pUser	用户数据
	iUserHandle	Plat_LoginCMS 返回的句柄
返回值	返回值：成功返回 0，其它失败。	
备注	资源类型值： 1--温度 2--湿度 3--浓度 4--红外 5—风速 9--客流量统计	

2.15.2 取消订阅实时数据

1.36 Plat_UnSubscribeRealTimeData

函数名称	int Plat_UnSubscribeRealTimeData(int iUserHandle)	
功能描述	取消订阅实时数据	
	iUserHandle	Plat_LoginCMS 返回的句柄
返回值	返回值：成功返回 0，其它失败。	
备注		

3 iVMS-8800 平台数据定义

3.1 属性定义

控制中心的属性名

Cell	描述
ControlUnitID	控制中心 ID
ControlUnitName	控制中心名称
UpControlUnitID	上级控制中心 ID

区域的属性名

Region	描述
region_id	区域 ID
region_name	区域名称
region_high	上级区域 ID
region_level	区域等级
region_cellLsh	区域所属控制中心 ID

设备的属性名

Device	描述
device_id	设备 ID
device_indexcode	设备 indexcode
device_name	设备名称
device_type	设备类型 海康 1xxxx 大华 2xxxx
device_state	设备状态 0-在线 1-断线
device_talk	对讲通道数
device_chan	通道个数
ip_address	设备 IP 地址
cell_id	所属区域 ID
device_port	设备端口
device_alarmin	报警输入个数
device_alarmout	报警输出个数

摄像头设备的属性名

Camera	描述
category_id	类别 ID（0-枪机/1-半球/2-快球/3-云台）
cam_indexcode	监控点 indexcode
device_id	摄像头 ID
device_name	摄像头名称

device_type	设备类型，摄像头固定为 0x03
inport	通道号
ConnectType	连接类型（0-TCP/1-UDP/2-MCAST/3-RTP）
controlunit_id	所属控制中心 ID
parent_device_id	父设备 ID，即设备 ID
region_id	所属区域 ID

报警输入的属性名

Alarmin	**描述**
IO_id	IO 信息
device_id	IO ID
device_name	IO 名称
device_type	设备类型，报警输入固定为 0x04
iochanindex	通道号
parent_device_id	父设备 ID
region_id	所属区域 ID

报警输出的属性名

Alarmout	**描述**
IO_id	IO 信息
device_id	IO ID
device_name	IO 名称
device_type	设备类型，报警输出固定为 0x05
iochanindex	通道号
parent_device_id	父设备 ID
region_id	所属区域 ID

录像文件的属性名

RecordFile	**描述**
starttime	开始时间
endtime	结束时间
type	片段类型
filename	文件名
size	文件大小（字节）

环境量的属性名

EnvDevice	**描述**
starttime	开始时间
endtime	结束时间
type	片段类型
filename	文件名

size	文件大小（字节）
envdev_channel	环境量设备通道号
region_id	所属区域 ID
envdev_maxvalue	模拟环境量最大值(485 环境量量程上限)
envdev_minvalue	模拟环境量最小值(485 环境量量程下限)
envdev_boundtype1	模拟环境量报警线类型 1(485 环境量没有该属性)
envdev_boundtype2	模拟环境量报警线类型 2(485 环境量没有该属性)
envdev_value1	环境量量程报警线 1
envdev_value2	环境量量程报警线 2
envdev_value3	485 环境量量程报警线 3(模拟环境量没有该属性)
envdev_value4	485 环境量量程报警线 4(模拟环境量没有该属性)
envdev_unit	环境量单位

3.2 云台控制参数说明

值	描述
8	设置预置位
9	删除预置位
11	镜头拉近
12	镜头移远
13	焦点前调
14	焦点后调
15	光圈扩大
16	光圈缩小
21	镜头向上
22	镜头向下
23	镜头向左
24	镜头向右
30	将预置点加入巡航序列
31	设置巡航点停顿时间
32	设置巡航速度
33	将预置点从巡航序列中删除
37	开始巡航
38	停止巡航
39	到达预置位

3.3 错误码定义

定义	值	描述
PLAT_NOERROR	0	无错误
PLAT_LOGIN_USERNOEXIST	1	用户不存在
PLAT_LOGIN_PSWERROR	2	密码错误

PLAT_PARAERROR	3	参数错误
PLAT_DEVNORESPONSE	4	设备无响应
PLAT_NETWORK_EXCEPTION	5	网络异常
PLAT_LOGIN_ALREADYLOGIN	6	用户已经登录
PLAT_UNKNOWNREASON	7	未知原因
PLAT_LOGIN_NOT_EXCLUSIVE	8	用户名称不唯一
PLAT_LOGIN_EXPIRED	9	用户已经过期
PLAT_LOGIN_NOACTIVE	10	用户未激活
PLAT_NORIGHT	11	用户无权限
PLAT_NOINIT	12	没有初始化
PLAT_STREAMSTART_FAIL	13	流媒体库启动失败
PLAT_NOUSERHANDLE	14	无此用户会话 ID
PLAT_GETCELLINFO_FAIL	15	获取中心信息失败
PLAT_GETREGIONINFO_FAIL	16	获取区域信息失败
PLAT_GETDEVICEINFO_FAIL	17	获取设备信息失败
PLAT_GETCAMINFO_FAIL	18	获取摄像头信息失败
PLAT_GETIOINFO_FAIL	19	获取 IO 设备信息失败

3.4 权限说明

定义	值	描述
PRIVILEGE_CODE_MANAGER_ALL	1	拥有所有的管理\操作权限
PRIVILEGE_CODE_OPERATOR_ALL	2	拥有所有的操作权限
PRIVILEGE_CODE_PREVIEW	10001	实时预览
PRIVILEGE_CODE_LOCAL_RECORD	10003	本地录像
PRIVILEGE_CODE_PTZ	10004	云台控制
PRIVILEGE_CODE_SET_PTZ	10005	云台配置
PRIVILEGE_CODE_PLAYBACK	10006	录像回放
PRIVILEGE_CODE_RECORD_LOCK	10007	录像锁定
PRIVILEGE_CODE_RECORD_BACKUP	10008	录像备份
PRIVILEGE_CODE_ALARM_LOG	10010	查看报警日志
PRIVILEGE_CODE_DEVICE_TALK	10013	设备对讲
PRIVILEGE_CODE_DEVICE_BROADCAST	10014	设备广播
PRIVILEGE_CODE_ALARM_OUT	10019	报警输出控制
PRIVILEGE_CODE_RECV_ALARM	10020	接收报警

3.5 结构体定义

3.5.1 资源类型

```
#define _CONTROLUNIT        0x00            // 控制中心
```

```
#define _REGION          0x01        // 区域
#define _DEVICE          0x02        // 编码器
#define _CAMERA          0x03        // 摄像头
#define _ALARMIN         0x04        // 报警输入
#define _ALARMOUT        0x05        // 报警输出
#define _RECORDFILE      0x06        // 录像文件
#define _ENVDEVICE       0x07        // 环境量外设
#define _485ENVDEVICE    0x08        // 485 环境量外设
```

3.5.2 回放命令定义

```
#define PLAY_START          10001 //开始播放
#define PLAY_PAUSE          10002 //暂停播放
#define PLAY_FAST           10003 //加快播放速度
#define PLAY_SLOW           10004 //减慢播放速度
#define PLAY_OFFSET         10005 //播放定位
```

3.5.3 巡航结构体定义

```
struct CRUISEPOINT
{
    BYTE PresetNum; //预置点
    BYTE Dwell;     //停留时间
    BYTE Speed;     //速度
    BYTE Reserve;   //保留
}CRUISEPOINT, *LPCRUISEPOINT;

struct NET_MULTI_SDK_CRUISE_RET{
    CRUISEPOINT struCruisePoint[32];    //最大支持 32 个巡航点
}NET_MULTI_SDK_CRUISE_RET, *LPNET_MULTI_SDK_CRUISE_RET;
```

4 iVMS-8800 平台编程引导

4.1 登录退出流程

1. 初始化平台（Plat_Init 接口）：对整个系统的初始化，初始化 ClientService 取流库播放库网络 SDK 库等操作。

```
示例
Plat_Init();
```

2. 用户登录（Plat_LoginCMS 接口）：用户输入登录的 CMS 中心的地址，用户名，密码，登录成功后会获取用户会话句柄，初始化用户数据，包括用户心跳机制，报警 MQ 启动。

```
示例
int handle
=Plat_LoginCMS("172.7.79.80",80,"admin","12345");
```

4.用户注销（Plat_LogoutCMS 接口）：注销成功后会删除用户数据，关闭心跳和 MQ 等。

```
示例
 Plat_LogoutCMS(handle);
```

5.平台释放资源（Plat_Free 接口）：退出时需要释放 ClientService 资源，取流库，释放播放库，封装网络 SDK 库。

```
示例
 Plat_Free();
```

4.2 资源查询流程

1. 资源查询（Plat_QueryResource 接口）：用户成功登录后必须先获取平台系统资源，1）获取监控点资源时应按照中心->区域->设备->摄像头的顺序进行，否则会出现错误；

```
示例
if (Plat_QueryResource(0,handle) == 0)
    {
        while (Plat_MoveNext(handle) != -1)
        {
            int Id = atoi(Plat_GetValueStr(Cell::ControlUnitID,
handle));
            CString Name =
Plat_GetValueStr(Cell::ControlUnitName, handle);
            int high =
atoi(Plat_GetValueStr(Cell::UpControlUnitID, handle));
        }
    }

    if (Plat_QueryResource(1,handle) == 0)
```

```
    {
        while (Plat_MoveNext(handle) != -1)
        {
            int Id = atoi(Plat_GetValueStr(Region::region_id,
handle));
            CString Name = Plat_GetValueStr(Region::region_name,
handle);
            int high=atoi(Plat_GetValueStr(Region::region_high,
handle));
        }
    }

    if (Plat_QueryResource(2,handle) == 0)
    {
        while (Plat_MoveNext(handle) != -1)
        {
            int Id = Plat_GetValueInt(Device::device_id,
handle);
            CString Name = Plat_GetValueStr(Device::device_name,
handle);
        }
    }

    if (Plat_QueryResource(3,handle) == 0)
    {
        while (Plat_MoveNext(handle) != -1)
        {
            int Id = atoi(Plat_GetValueStr(Camera::device_id,
handle));
            CString Name =Plat_GetValueStr(Camera::device_name,
handle);
            int high = atoi(Plat_GetValueStr(Camera::region_id,
handle));
        }
    }
...
```

2）获取环境量实时数据资源应按照中心->区域->环境量

示例

```
if (Plat_QueryResource(0,handle) == 0)
    {
        while (Plat_MoveNext(handle) != -1)
        {
            int Id = atoi(Plat_GetValueStr(Cell::ControlUnitID,
```

```
handle));
          CString Name =
Plat_GetValueStr(Cell::ControlUnitName, handle);
          int high =
atoi(Plat_GetValueStr(Cell::UpControlUnitID, handle));
       }
    }

    if (Plat_QueryResource(1,handle) == 0)
    {
       while (Plat_MoveNext(handle) != -1)
       {
          int Id = atoi(Plat_GetValueStr(Region::region_id,
handle));
          CString Name = Plat_GetValueStr(Region::region_name,
handle);
          int high=atoi(Plat_GetValueStr(Region::region_high,
handle));
       }
    }

    if (Plat_QueryResource(7,handle) == 0)
    {
       while (Plat_MoveNext(handle) != -1)
       {
          //环境量全部属性请参考头文件 EnvDevice 定义
          int Id = Plat_GetValueInt(EnvDevice::envdev_id,
handle);
         CString Name=Plat_GetValueStr(EnvDevice::envdev_nam,
handle);
       }
    }
     ...
```

3）其它资源请按照类似流程处理。

4.3 实时预览流程

1.实时预览（Plat_QueryRealStreamURL Plat_PlayVideo 接口）。先使用 Plat_QueryRealStreamURL 获取摄像头实时预览的 URL，再调用 Plat_PlayVideo 进行播放。

```
   示例
    const char *CameraUrl = Plat_QueryRealStreamURL(cId,
   handle);
```

```
    int streamhandle =
   Plat_PlayVideo(CameraUrl,(long)hwnd,handle,StreamCallback,
   this);
```

4.4 回放下载流程

1.回放和下载（Plat_QueryRecordFile Plat_PlayVideo 接口）。先使用 Plat_QueryRecordFile 查询摄像头录像文件，用游标函数获取 URL，再调用 Plat_PlayVideo 进行播放，下载同样使用该 URL。

```
示例
Plat_QueryRecordFile(cId,absTimeFrom.GetTime(),absTimeTo.GetT
ime(), QueryCondition, handle);
    while (Plat_MoveNext(handle) != -1)
    {
        CString size = Plat_GetValueStr(RecordFile::size,
handle);
       CString filename=Plat_GetValueStr(RecordFile::filename,
handle);
    CString starttime=Plat_GetValueStr(RecordFile::starttime,
 handle);
        CString endtime=Plat_GetValueStr(RecordFile::endtime,
handle);
    }
int streamhandle =
Plat_PlayVideo(filename,(long)hwnd, handle, StreamCallback);

streamhandle =
Plat_StartDownLoad(filename,strFileName,
    absTimeFrom.GetTime(),absTimeTo.GetTime(),256*1024*1024,
    handle);
```