杭州海康威视数字技术股份有限公司

定制外部接口说明文档

密级级别：[外部公开]

生效时间： 2016年9月13日

保密期：无

杭州海康威视数字技术股份有限公司 版权所有

**目 录**

[1.1 [扩展]获取物理磁盘信息 4](#_Toc465328246)

[1.2 [修改]上传数据文件 5](#_Toc465328247)

[1.2.1 [复用]接口 5](#_Toc465328248)

[1.2.2 接口参数详细定义 6](#_Toc465328249)

[1.2.3 宏定义及结构体 6](#_Toc465328250)

[1.2.4 备注(remark) 7](#_Toc465328251)

[1.3 [修改]下载数据文件 7](#_Toc465328252)

[1.3.1 [复用]接口 7](#_Toc465328253)

[1.3.2 接口参数详细定义 8](#_Toc465328254)

[1.3.3 宏定义及结构体 8](#_Toc465328255)

[1.3.4 备注(remark) 9](#_Toc465328256)

[1.4 [新增]异地备份备机信息配置 9](#_Toc465328257)

[1.4.1 [新增]异地备份备机信息配置 9](#_Toc465328258)

[1.4.2 [新增]异地备份备机信息配置能力 10](#_Toc465328259)

[1.5 [新增]异地备份任务配置 11](#_Toc465328260)

[1.5.1 [新增]添加异地备份任务 11](#_Toc465328261)

[1.5.2 [新增]添加异地备份任务能力 13](#_Toc465328262)

[1.5.3 [新增]获取异地备份任务 13](#_Toc465328263)

[1.5.4 [新增]获取异地备份任务的能力 15](#_Toc465328264)

[1.5.5 [新增]删除异地备份任务 16](#_Toc465328265)

[1.5.6 [新增]删除异地备份任务能力 17](#_Toc465328266)

[1.6 [新增]数据文件传输速率配置 18](#_Toc465328267)

[1.6.1 [新增]数据文件传输速率配置 18](#_Toc465328268)

[1.6.2 [新增]数据文件传输速率配置能力 19](#_Toc465328269)

[1.7 [新增]数据文件锁定配置 20](#_Toc465328270)

[1.7.1 [新增]数据文件锁定配置 20](#_Toc465328271)

[1.7.2 [新增]数据文件传输速率配置能力 21](#_Toc465328272)

[1.8 [新增]获取数据文件信息 22](#_Toc465328273)

[1.8.1 [新增]获取数据文件信息 22](#_Toc465328274)

[1.8.2 [新增]获取数据文件信息能力 23](#_Toc465328275)

## [扩展]获取物理磁盘信息

**命令号**

#define NET\_DVR\_GET\_PHY\_DISK\_INFO 6306 //获取物理磁盘信息

**数据结构**

typedef struct tagNET\_DVR\_PHY\_DISK\_INFO

{

WORD wPhySlot; // 硬盘id号 对应Array的索引，从0开始

BYTE byType; // 硬盘smart信息；0 普通，1全局热备，

2-阵列热备 3-阵列盘

BYTE byStatus; // 硬盘状态； 1-正常

BYTE byMode[40]; // 硬盘类型 字符串

DWORD dwHCapacity; // 磁盘总量高32位 单位kb

DWORD dwLCapacity; // 磁盘总量低32位

BYTE byArrayName[MAX\_NAMELEN];

WORD wArrayID; // 所属阵列ID

BYTE byFunctionType; //功能类型：0-视频，1-文件，2-图片，3-存档卷

BYTE byRes1; // 保留字节

DWORD dwHRemainCapacity; // 磁盘剩余总量高32位 单位kb

DWORD dwLRemainCapacity; // 磁盘剩余总量低32位

BYTE byRes[92]; // 保留字节

}NET\_DVR\_PHY\_DISK\_INFO, \*LPNET\_DVR\_PHY\_DISK\_INFO;

#define MAX\_NAMELEN 16;

**接口函数**

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数: NET\_DVR\_StartRemoteConfig

描述: 长连接通用接口

输入: iUserID： NET\_DVR\_Logon\_V30返回值

dwCommand 长连接配置命令 NET\_DVR\_GET\_PHY\_DISK\_INFO

lpInBuffer: 输入缓冲，指向NET\_DVR\_PHY\_DISK\_INFO

dwInBufferLen: 输入缓冲的大小，sizeof（NET\_DVR\_PHY\_DISK\_INFO）

cbStateCallback： 状态回调函数，这里为fRemoteConfigCallback

pUserData: 用户数据指针

输出: 无

返回值: -1表示失败，其他值作为句柄

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

NET\_DVR\_API HPR\_INT32 CALLBACK NET\_DVR\_StartRemoteConfig(HPR\_INT32 lUserID, HPR\_UINT32 dwCommand, HPR\_VOIDPTR lpInBuffer, HPR\_UINT32 dwInBufferLen, fRemoteConfigCallback cbCallback, HPR\_VOIDPTR pUserData)

//物理磁盘信息是通过回调函数回调出来的

typedef void(CALLBACK \*fRemoteConfigCallback)( DWORD dwType, void \*lpBuffer,

DWORD dwBufLen, void \*pUserData);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_StopRemoteConfig

函数描述: 关闭长连接配置接口所创建的句柄,释放资源

输入参数: iHandle: 长连接配置句柄

输出参数:

返回值: HPR\_TRUE表示成功,HPR\_FALSE表示失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

NET\_DVR\_API HPR\_BOOL CALLBACK NET\_DVR\_StopRemoteConfig(HPR\_INT32 lHandle)

## [修改]上传数据文件

### [复用]接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_UploadFile\_V40

函数描述: 上传文件

输入参数: lUserID: NET\_DVR\_Login\_V30的返回值

dwUploadType:上传文件类型，见下面表格

lpInBuffer：输入参数。不同的dwUploadType，输入参数不同，具体参见下文列表

dwInBufferSize：输入缓冲区大小

sFileName：上传文件的路径（包括文件名）

dwOutBufferSize：输出缓冲区大小

输出参数: lpOutBuffer：输出参数。不同的dwUploadType，输出参数不同，具体参见下文列表

返回值: -1表示失败，其他值作为NET\_DVR\_UploadClose和NET\_DVR\_GetUploadState等函数的参数

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

LONG NET\_DVR\_UploadFile\_V40(LONG lUserID, DWORD dwUploadType, LPVOID lpInBuffer, DWORD dwInBufferSize, char const \* sFileName, LPVOID lpOutBuffer, DWORD dwOutBufferSize);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_UploadSend

函数描述: 发送数据

输入参数: lUploadHandle: NET\_DVR\_UploadFile\_V40的返回值

pstruSendParamIN：上传的数据内容（NET\_DVR\_SEND\_PARAM\_IN结构体指针）

输出参数: lpOutBuffer：输出参数，保留，置为NULL

返回值: -1表示失败，大于等于0表示本次发送成功的字节数，未发送成功的剩余字节数可以再次调用该接口进行发送

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

LONG NET\_DVR\_UploadSend(LONG lUploadHandle, const NET\_DVR\_SEND\_PARAM\_IN \*pstruSendParamIN, void\* lpOutBuffer);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_GetUploadState

函数描述: 获取上传状态

输入参数: lUploadHandle: NET\_DVR\_UploadFile\_V40的返回值

输出参数: pProgress：输出参数，上传进度

返回值: -1表示失败，其他为上传的状态值，1-上传成功；2- 正在上传；3- 上传失败；4- 网络断开 6-硬盘错误 7-无审讯文件存放盘 8-容量不足 9-设备资源不足 10-文件个数超过最大值，

15-文件类型错误

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

LONG NET\_DVR\_GetUploadState(LONG lUploadHandle, LPDWORD pProgress);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_GetUploadResult

函数描述: 获取当前数据上传的URL

输入参数: lUploadHandle: NET\_DVR\_UploadFile\_V40的返回值

dwOutBufferSize：lpOutBuffer大小

输出参数: lpOutBuffer：结构体NET\_DVR\_UPLOAD\_FILE\_RET指针

返回值: TRUE表示成功，FALSE表示失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_GetUploadResult(LONG lUploadHandle, LPVOID lpOutBuffer, DWORD dwOutBufferSize);

### 接口参数详细定义

|  |  |  |
| --- | --- | --- |
| **dwUploadType宏定义** | **lpInBuffer对应值** | **lpOutBuffer对应值** |
| UPLOAD\_DATA\_FILE | NET\_DVR\_UPLOAD\_DATA\_PRARAM | NULL |

### 宏定义及结构体

* 宏定义

#define UPLOAD\_DATA\_FILE 31 //上传数据文件

* 结构体

typedef struct tagNET\_DVR\_UPLOAD\_DATA\_PRARAM\_

{

DWORD dwSize;

BYTE byFileType; //文件类型 0-录像，1-音频，2-其他

BYTE byStoragePoolID; //存储池ID

BYTE byRes1[2];

DWORD dwFileSize; //文件大小

char szFileName[MAX\_FILE\_NAME\_LEN/\*100\*/]; //文件名称

BYTE byRes[300];

} NET\_DVR\_UPLOAD\_DATA\_PRARAM,\*LPNET\_DVR\_UPLOAD\_DATA\_PRARAM;

typedef struct tagNET\_DVR\_UPLOAD\_FILE\_RET

{

BYTE sUrl[MAX\_UPLOADFILE\_URL\_LEN/\*240\*/] ; //url

BYTE byRes[260];

} NET\_DVR\_UPLOAD\_FILE\_RET,\*LPNET\_DVR\_UPLOAD\_FILE\_RET;

//本功能只需用pSendData、dwSendDataLen字段

typedef struct tagNET\_DVR\_SEND\_PARAM\_IN

{

BYTE \*pSendData; //发送的缓冲区

DWORD dwSendDataLen; //发送数据长度

NET\_DVR\_TIME\_V30 struTime; //图片时间

BYTE byPicType; //图片格式,1-jpg,2-bmp,3-png,4-SWF,5-GIF

BYTE byRes1[3];

DWORD dwPicMangeNo; //图片管理号

BYTE sPicName[NAME\_LEN]; //图片名称

DWORD dwPicDisplayTime; //图片播放时长，单位秒

BYTE byRes[200];

}NET\_DVR\_SEND\_PARAM\_IN, \*LPNET\_DVR\_SEND\_PARAM\_IN;

### 备注(remark)

当上传类型为UPLOAD\_DATA\_FILE时，lpInBuffer对应结构体NET\_DVR\_UPLOAD\_DATA\_PRARAM，上层使用时可选用调用NET\_DVR\_UploadSend接口或路径的方式二选一进行上传。

调用顺序：

1. 调用接口NET\_DVR\_UploadFile\_V40，命令UPLOAD\_DATA\_FILE，lpInBuffer对应结构体NET\_DVR\_UPLOAD\_DATA\_PRARAM。
2. 若传入文件路径为NULL，则通过循环调用NET\_DVR\_UploadSend接口来传入文件数据，否则，忽略本步骤
3. 循环调用接口NET\_DVR\_GetUploadState获取上传状态
4. 当接口NET\_DVR\_GetUploadState返回1即上传成功时。调用接口NET\_DVR\_GetUploadResult，lpOutBuffer对应结构体NET\_DVR\_UPLOAD\_FILE\_RET
5. 调用接口NET\_DVR\_UploadClose停止上传

## [修改]下载数据文件

### [复用]接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_StartDownload

函数描述: 下载文件

输入参数: lUserID:NET\_DVR\_Login\_V30的返回值

dwDownloadType:下载文件类型，见下文列表

lpInBuffer：输入参数。不同的dwDownloadType，输入参数不同，见下文列表

dwInBufferSize：输入缓冲区大小

sFileName：下载文件的保存路径（绝对路径，包括文件名）

输出参数:

返回值: -1表示失败，其他值作为NET\_DVR\_StopDownload和NET\_DVR\_GetDownloadState等函数的参数

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

LONG NET\_DVR\_StartDownload(LONG lUserID, DWORD dwDownloadType, LPVOID lpInBuffer, DWORD dwInBufferSize, const char \*sFileName);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_GetDownloadState

函数描述: 获取下载状态

输入参数: lDownloadHandle: NET\_DVR\_StartDownload的返回值

输出参数: pProgress：返回的进度值，0~100

返回值: -1表示函数调用失败，其他为下载的状态值：1- 下载成功；2- 正在下载；3- 下载失败；4- 网络断开，状态未知

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

LONG NET\_DVR\_GetDownloadState(LONG lDownloadHandle, LPDWORD pProgress);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_StopDownload

函数描述: 停止下载文件

输入参数:

输出参数: lHandle: NET\_DVR\_StartDownload的返回值

返回值: TRUE表示成功，FALSE表示失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_StopDownload(LONG lHandle);

### 接口参数详细定义

|  |  |
| --- | --- |
| **dwCommand宏定义** | **lpInBuffer对应值** |
| NET\_SDK\_DOWNLOAD\_DATA\_FILE | NET\_DVR\_DOWNLOAD\_DATA\_PARAM |
|  |  |

### 宏定义及结构体

#define NET\_SDK\_DOWNLOAD\_DATA\_FILE 18 //下载数据文件

typedef BOOL (\*DOWNLOAD\_DATA\_CB)( LONG lHandle, NET\_DOWNLOAD\_CB\_INFO \*pCBInfo, void \*pUserData);

typedef struct tagNET\_DOWNLOAD\_CB\_INFO

{

DWORD dwType; //类型, 0-数据信息,1-单个图片信息(云存储批量下载),2-图片总体信息(云存储批量下载)

BYTE \*pData; //数据指针

DWORD dwDataLen; //数据长度

void\* pFileInfo; //文件信息结构体(云存储批量下载时有效)

DWORD dwFileInfoLen; //文件信息结构体长度(云存储批量下载时有效)

BYTE byRes[120]; //保留

}NET\_DOWNLOAD\_CB\_INFO, \*LPNET\_DOWNLOAD\_CB\_INFO;

typedef struct tagNET\_DVR\_DOWNLOAD\_DATA\_PARAM\_

{

DWORD dwSize;

BYTE sUrl[MAX\_UPLOADFILE\_URL\_LEN/\*240\*/] ; //url

BYTE byRes[248];

DOWNLOAD\_DATA\_CB fnDownloadDataCB; //下载数据回调函数

void \*pUserData; //用户参数, 在fnPreviewDataCB回调出来

} NET\_DVR\_DOWNLOAD\_DATA\_PARAM, \*LPNET\_DVR\_DOWNLOAD\_DATA\_PARAM;

### 备注(remark)

当下载类型为NET\_SDK\_DOWNLOAD\_DATA\_FILE时，lpInBuffer对应结构体NET\_DVR\_DOWNLOAD\_DATA\_PARAM，上层使用时可选用数据回调或路径的方式二选一进行下载，fnDownloadDataCB、sFileName同时只能有一个参数生效

调用顺序：

1. 调用接口NET\_DVR\_StartDownload，

命令NET\_SDK\_DOWNLOAD\_DATA\_FILE，

lpInBuffer对应结构体NET\_DVR\_DOWNLOAD\_DATA\_PARAM，

1. 调用接口NET\_DVR\_GetDownloadState，获取状态和进度
2. 重复步骤2，获取进度
3. 调用接口NET\_DVR\_StopDownload停止下载

## [新增]异地备份备机信息配置

### [新增]异地备份备机信息配置

* + - 1. 接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_STDXMLConfig

函数描述: SDK-ISAPI功能透传接口

输入参数: lUserID: NET\_DVR\_Login\_V30()的返回值

lpInputParam: 输入参数指针

lpOutputParam：输出参数指针

输入输出参数

返回值: TRUE-成功,FALSE-失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_STDXMLConfig(LONG lUserID, const NET\_DVR\_XML\_CONFIG\_INPUT\* lpInputParam, NET\_DVR\_XML\_CONFIG\_OUTPUT\* lpOutputParam);

获取移动设备刻录配置：

lpInputParam->lpRequestUrl为：GET /ISAPI/ContentMgmt/RemoteBackup/BackupMachineInfo?security=&iv=\r\n；

lpInputParam->lpInBuffer为NULL；

lpOutputParam->lpOutBuffer为BackupMachineInfo；

lpOutputParam->lpStatusBuffer为ResponseStatus，失败时返回；

设置移动设备刻录配置：

lpInputParam->lpRequestUrl为：PUT /ISAPI/ContentMgmt/RemoteBackup/BackupMachineInfo\r\n；

lpInputParam->lpInBuffer为BackupMachineInfo；

lpOutputParam->lpOutBuffer为NULL；

lpOutputParam->lpStatusBuffer为ResponseStatus，成功、失败均返回；

* + - 1. XML及结构体

**BackupMachineInfo XML Block**

|  |
| --- |
| <BackupMachineInfo version="2.0" xmlns="http://www.isapi.org/ver20/XMLSchema">  <ipVersion> <!-- req, xs:string, “v4,v6,dual” --></ipVersion>  <ipAddress> <!-- dep, xs:string--> </ipAddress>  <ipv6Address><!-- dep, xs:string--></ipv6Address>  <userName><!--req, xs:string, 用户名，min="" max=""--></userName>  <password><!--req, xs:string, 密码，min="" max=""--></password>  //当输入userName 或password 节点时，SDK内部会将整个报文进行加密  </BackupMachineInfo> |

* + - 1. 备注

### [新增]异地备份备机信息配置能力

* + - 1. 接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_STDXMLConfig

函数描述: SDK-ISAPI功能透传接口

输入参数: lUserID: NET\_DVR\_Login\_V30()的返回值

lpInputParam: 输入参数指针

lpOutputParam：输出参数指针

输入输出参数

返回值: TRUE-成功,FALSE-失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_STDXMLConfig(LONG lUserID, const NET\_DVR\_XML\_CONFIG\_INPUT\* lpInputParam, NET\_DVR\_XML\_CONFIG\_OUTPUT\* lpOutputParam);

lpInputParam->lpRequestUrl为：GET /ISAPI/ContentMgmt/RemoteBackup/BackupMachineInfo/capabilities\r\n；

lpInputParam->lpInBuffer为NULL；

lpOutputParam->lpOutBuffer为BackupMachineInfo；

lpOutputParam->lpStatusBuffer为ResponseStatus，失败时返回；

* + - 1. XML及结构体

**BackupMachineInfo XML Block**

|  |
| --- |
| <BackupMachineInfo version="2.0" xmlns="http://www.isapi.org/ver20/XMLSchema">  <ipVersion opt=“v4,v6,dual”> <!-- req, xs:string--></ipVersion>  <ipAddress> <!-- dep, xs:string--> </ipAddress>  <ipv6Address><!-- dep, xs:string--></ipv6Address>  <userName min="" max=""><!--req, xs:string, 用户名--></userName>  <password min="" max=""><!--req, xs:string, 密码--></password>  //当输入userName 或password 节点时，SDK内部会将整个报文进行加密  </BackupMachineInfo> |

* + - 1. 备注

无

## [新增]异地备份任务配置

### [新增]添加异地备份任务

* + - 1. 接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_STDXMLConfig

函数描述: SDK-ISAPI功能透传接口

输入参数: lUserID: NET\_DVR\_Login\_V30()的返回值

lpInputParam: 输入参数指针

lpOutputParam：输出参数指针

输入输出参数

返回值: TRUE-成功,FALSE-失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_STDXMLConfig(LONG lUserID, const NET\_DVR\_XML\_CONFIG\_INPUT\* lpInputParam, NET\_DVR\_XML\_CONFIG\_OUTPUT\* lpOutputParam);

添加异地备份任务：

lpInputParam->lpRequestUrl为：POST /ISAPI/ContentMgmt/RemoteBackup/addTask\r\n；

lpInputParam->lpInBuffer为BackupData；

lpOutputParam->lpOutBuffer为TaskData；

lpOutputParam->lpStatusBuffer为ResponseStatus，失败时返回；

* + - 1. XML及结构体

**BackupData XML Block**

|  |
| --- |
| <BackupData version="2.0" xmlns="http://www.isapi.org/ver20/XMLSchema">  <BackupList size=”50”>  <BackupInfo>  <fileIndex><!--req, xs:string, min=”0”,max=”64”--></fileIndex>  <url><!--req, xs:string, min=”0”,max=”240”--></url>  </BackupInfo>  </BackupList>  </BackupData> |

**TaskData XML Block**

|  |
| --- |
| <TaskData version=“2.0” xmlns=“http://www.isapi.org/ver20/XMLSchema”>  <TaskList size=”50”>  <TaskInfo>  <taskID><!-- opt, xs: string min=”” max=”” --></taskID>//返回字符串为-1表明添加任务失败  <errorDescription><!-- opt, xs: string min=”” max=”” --></errorDescription>//出错时返回错误原因  </TaskInfo>  </TaskList>  </TaskData> |

* + - 1. 备注

### [新增]添加异地备份任务能力

* + - 1. 接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_STDXMLConfig

函数描述: SDK-ISAPI功能透传接口

输入参数: lUserID: NET\_DVR\_Login\_V30()的返回值

lpInputParam: 输入参数指针

lpOutputParam：输出参数指针

输入输出参数

返回值: TRUE-成功,FALSE-失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_STDXMLConfig(LONG lUserID, const NET\_DVR\_XML\_CONFIG\_INPUT\* lpInputParam, NET\_DVR\_XML\_CONFIG\_OUTPUT\* lpOutputParam);

lpInputParam->lpRequestUrl为：GET /ISAPI/ContentMgmt/RemoteBackup/addTask/capabilities\r\n；

lpInputParam->lpInBuffer为NULL；

lpOutputParam->lpOutBuffer为BackupData；

lpOutputParam->lpStatusBuffer为ResponseStatus，失败时返回；

* + - 1. XML及结构体

**BackupData XML Block**

|  |
| --- |
| <BackupData version="2.0" xmlns="http://www.isapi.org/ver20/XMLSchema">  <BackupList size=”50”>  <BackupInfo>  <fileIndex min=”0”,max=”64”><!--req, xs:string--></fileIndex>  <url min=”0”,max=”240”><!--req, xs:string--></url>  </BackupInfo>  </BackupList>  </BackupData> |

* + - 1. 备注

无

### [新增]获取异地备份任务

* + - 1. [复用]接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数: NET\_DVR\_StartRemoteConfig

描述: 长连接通用接口

输入: iUserID： NET\_DVR\_Logon\_V30返回值

dwCommand 长连接配置命令

lpInBuffer: 输入缓冲

dwInBufferLen: 输入缓冲的大小

cbStateCallback： 状态回调函数

pUserData: 用户数据指针

输出: 无

返回值: -1表示失败，其他值作为句柄

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

NET\_DVR\_API LONG \_\_stdcall NET\_DVR\_StartRemoteConfig(LONG lUserID, DWORD dwCommand, LPVOID lpInBuffer, DWORD dwInBufferLen, fRemoteConfigCallback cbStateCallback, LPVOID pUserData);

typedef void(CALLBACK \*fRemoteConfigCallback)(DWORD dwType, void\* lpBuffer, DWORD dwBufLen, void\* pUserData);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_StopRemoteConfig

函数描述: 关闭长连接配置接口所创建的句柄,释放资源

输入参数: iHandle: 长连接配置句柄

输出参数:

返回值: TRUE表示成功,FALSE表示失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

NET\_DVR\_API BOOL CALLBACK NET\_DVR\_StopRemoteConfig(LONG lHandle)

* + - 1. 接口参数详细定义

#define NET\_DVR\_GET\_REMOTE\_BACKUP\_TASK 26007 //获取异地备份任务

* + - 1. 宏定义及结构体

#define NET\_SDK\_TASK\_LEN 32 //任务ID长度

#define NET\_SDK\_FILE\_INDEX 64 文件索引长度

typedef struct tagNET\_DVR\_REMOTE\_BACKUP\_TASK\_COND

{

DWORD dwSize;

BYTE byType; //获取方式：0-根据任务ID获取，1-获取所有未执行任务，2-获取已执行任务，3-获取全部异地备份任务

BYTE byRes1[3];

BYTE byTaskID[NET\_SDK\_TASK\_LEN /\*32\*/];//任务ID，仅当byType为0时有效

BYTE byRes[320];

}NET\_DVR\_REMOTE\_BACKUP\_TASK\_COND,\*LPNET\_DVR\_REMOTE\_BACKUP\_TASK\_COND;

typedef struct tagNET\_DVR\_REMOTE\_BACKUP\_TASK\_CFG

{

DWORD dwSize;

BYTE byTaskID[NET\_SDK\_TASK\_LEN /\*32\*/];//任务ID

BYTE byFileIndex[NET\_SDK\_FILE\_INDEX /\*64\*/];//文件索引

BYTE sUrl[MAX\_UPLOADFILE\_URL\_LEN/\*240\*/] ; //url

BYTE byStatus;//异地备份状态：0-未执行，1-备份中，2-完成备份，3-备份失败

BYTE byRes1[3];

BYTE sBackupMachineUrl[MAX\_UPLOADFILE\_URL\_LEN/\*240\*/] ; //完成备份之后，文件在备机上的URL（仅当备份状态为2-完成备份时有效）

BYTE byRes[128];

}NET\_DVR\_REMOTE\_BACKUP\_TASK\_CFG,\*LPNET\_DVR\_REMOTE\_BACKUP\_TASK\_CFG;

* + - 1. 备注

// 长连接回调类型

typedef enum

{

NET\_SDK\_CALLBACK\_TYPE\_STATUS = 0, // 回调状态值

NET\_SDK\_CALLBACK\_TYPE\_PROGRESS, // 回调进度值

NET\_SDK\_CALLBACK\_TYPE\_DATA // 回调数据内容

}NET\_SDK\_CALLBACK\_TYPE;

当回调类型为 NET\_SDK\_CALLBACK\_TYPE\_DATA 表示有实际数据值(NET\_DVR\_XXXX\_CFG)；

回调类型为NET\_SDK\_CALLBACK\_TYPE\_STATUS时的状态值

typedef enum

{

NET\_SDK\_CALLBACK\_STATUS\_SUCCESS = 1000, // 成功

NET\_SDK\_CALLBACK\_STATUS\_PROCESSING, // 处理中

NET\_SDK\_CALLBACK\_STATUS\_FAILED // 失败

}NET\_SDK\_CALLBACK\_STATUS\_NORMAL;

### [新增]获取异地备份任务的能力

* + - 1. 接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_STDXMLConfig

函数描述: SDK-ISAPI功能透传接口

输入参数: lUserID: NET\_DVR\_Login\_V30()的返回值

lpInputParam: 输入参数指针

lpOutputParam：输出参数指针

输入输出参数

返回值: TRUE-成功,FALSE-失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_STDXMLConfig(LONG lUserID, const NET\_DVR\_XML\_CONFIG\_INPUT\* lpInputParam, NET\_DVR\_XML\_CONFIG\_OUTPUT\* lpOutputParam);

lpInputParam->lpRequestUrl为GET /ISAPI/ContentMgmt/RemoteBackup/searchTask/capabilities\r\n；

lpInputParam->lpInBuffer为NULL；

lpOutputParam->lpOutBuffer为SearchTask；

lpOutputParam->lpStatusBuffer为ResponseStatus，失败时返回；

* + - 1. XML及结构体

**SearchTask XML Block**

|  |
| --- |
| <SearchTask version="2.0" xmlns="http://www.std-cgi.org/ver20/XMLSchema">  <searchID min="" max=""><!--req,xs:string; 搜索记录唯一标识，用来确认上层客户端是否为同一个(倘若是同一个，则设备记录内存，下次搜索加快速度) --></searchID>  <searchResultPosition><!-- req, xs: integer--></searchResultPosition>  <maxResults min="" max=""><!-- req, xs: integer --></maxResults>  <type opt="byTaskID, non-executionTask, executionTask, allTask"><!--req, xs: string--></type>//获取方式：根据任务ID获取，获取所有未执行任务，获取已执行任务，获取全部异地备份任务  </SearchTask> |

### [新增]删除异地备份任务

* + - 1. 接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_STDXMLConfig

函数描述: SDK-ISAPI功能透传接口

输入参数: lUserID: NET\_DVR\_Login\_V30()的返回值

lpInputParam: 输入参数指针

lpOutputParam：输出参数指针

输入输出参数

返回值: TRUE-成功,FALSE-失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_STDXMLConfig(LONG lUserID, const NET\_DVR\_XML\_CONFIG\_INPUT\* lpInputParam, NET\_DVR\_XML\_CONFIG\_OUTPUT\* lpOutputParam);

删除异地备份任务：

lpInputParam->lpRequestUrl为：Delete /ISAPI/ContentMgmt/RemoteBackup/deleteTask\r\n；

lpInputParam->lpInBuffer为DeleteTask；

lpOutputParam->lpOutBuffer为NULL；

lpOutputParam->lpStatusBuffer为ResponseStatus，失败时返回；

* + - 1. XML及结构体

**DeleteTask XML Block**

|  |
| --- |
| <DeleteTask version=“2.0” xmlns=“http://www.isapi.org/ver20/XMLSchema”>  <deleteType><!--req, xs: string, opt="deleteByTaskID,deleteAll"--></deleteType>//按任务ID删除，删除所有  <taskID> <!-- dep, xs: string min=””, max=””, 仅当deleteType为deleteByTaskID有效--> </taskID>  </DeleteTask> |

* + - 1. 备注

### [新增]删除异地备份任务能力

* + - 1. 接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_STDXMLConfig

函数描述: SDK-ISAPI功能透传接口

输入参数: lUserID: NET\_DVR\_Login\_V30()的返回值

lpInputParam: 输入参数指针

lpOutputParam：输出参数指针

输入输出参数

返回值: TRUE-成功,FALSE-失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_STDXMLConfig(LONG lUserID, const NET\_DVR\_XML\_CONFIG\_INPUT\* lpInputParam, NET\_DVR\_XML\_CONFIG\_OUTPUT\* lpOutputParam);

lpInputParam->lpRequestUrl为：Delete /ISAPI/ContentMgmt/RemoteBackup/deleteTask/capabilities\r\n；

lpInputParam->lpInBuffer为NULL；

lpOutputParam->lpOutBuffer为DeleteTask；

lpOutputParam->lpStatusBuffer为ResponseStatus，失败时返回；

* + - 1. XML及结构体

**DeleteTask XML Block**

|  |
| --- |
| <DeleteTask version=“2.0” xmlns=“http://www.isapi.org/ver20/XMLSchema”>  <deleteType opt="deleteByTaskID,deleteAll"><!--req, xs: string, --></deleteType>//按任务ID删除，删除所有  <taskID min=””, max=””> <!-- dep, xs: string, 仅当deleteType为deleteByTaskID有效--> </taskID>  </DeleteTask> |

* + - 1. 备注

无

## [新增]数据文件传输速率配置

### [新增]数据文件传输速率配置

* + - 1. 接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_STDXMLConfig

函数描述: SDK-ISAPI功能透传接口

输入参数: lUserID: NET\_DVR\_Login\_V30()的返回值

lpInputParam: 输入参数指针

lpOutputParam：输出参数指针

输入输出参数

返回值: TRUE-成功,FALSE-失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_STDXMLConfig(LONG lUserID, const NET\_DVR\_XML\_CONFIG\_INPUT\* lpInputParam, NET\_DVR\_XML\_CONFIG\_OUTPUT\* lpOutputParam);

获取数据文件传输速率配置：

lpInputParam->lpRequestUrl为：GET /ISAPI/ContentMgmt/FileData/TransferSpeedConfiguration\r\n；

lpInputParam->lpInBuffer为NULL；

lpOutputParam->lpOutBuffer为TransferSpeedConfiguration；

lpOutputParam->lpStatusBuffer为ResponseStatus，失败时返回；

设置数据文件传输速率配置：

lpInputParam->lpRequestUrl为：PUT /ISAPI/ContentMgmt/FileData/TransferSpeedConfiguration\r\n；

lpInputParam->lpInBuffer为TransferSpeedConfiguration；

lpOutputParam->lpOutBuffer为NULL；

lpOutputParam->lpStatusBuffer为ResponseStatus，成功、失败均返回；

* + - 1. XML及结构体

**TransferSpeedConfiguration XML Block**

|  |
| --- |
| <TransferSpeedConfiguration version="2.0" xmlns="http://www.isapi.org/ver20/XMLSchema">  <upload> <!-- req, xs:string, opt=”low,mediumLow,medium,mediumHigh,high,full” --></upload>//低速，中低速，中速，中高速，高速，全速  <download> <!--req, xs:string, opt=”low,mediumLow,medium,mediumHigh,high,full” --> </download>//低速，中低速，中速，中高速，高速，全速  </TransferSpeedConfiguration> |

* + - 1. 备注

### [新增]数据文件传输速率配置能力

* + - 1. 接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_STDXMLConfig

函数描述: SDK-ISAPI功能透传接口

输入参数: lUserID: NET\_DVR\_Login\_V30()的返回值

lpInputParam: 输入参数指针

lpOutputParam：输出参数指针

输入输出参数

返回值: TRUE-成功,FALSE-失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_STDXMLConfig(LONG lUserID, const NET\_DVR\_XML\_CONFIG\_INPUT\* lpInputParam, NET\_DVR\_XML\_CONFIG\_OUTPUT\* lpOutputParam);

lpInputParam->lpRequestUrl为：GET /ISAPI/ContentMgmt/FileData/TransferSpeedConfiguration/capabilities\r\n；

lpInputParam->lpInBuffer为NULL；

lpOutputParam->lpOutBuffer为TransferSpeedConfiguration；

lpOutputParam->lpStatusBuffer为ResponseStatus，失败时返回；

* + - 1. XML及结构体

**TransferSpeedConfiguration XML Block**

|  |
| --- |
| <TransferSpeedConfiguration version="2.0" xmlns="http://www.isapi.org/ver20/XMLSchema">  <upload opt=”low,mediumLow,medium,mediumHigh,high,full”> <!-- req, xs:string--></upload>//低速，中低速，中速，中高速，高速，全速  <download opt=”low,mediumLow,medium,mediumHigh,high,full”> <!--req, xs:string--> </download>//低速，中低速，中速，中高速，高速，全速  </TransferSpeedConfiguration> |

* + - 1. 备注

无

## [新增]数据文件锁定配置

### [新增]数据文件锁定配置

* + - 1. 接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_STDXMLConfig

函数描述: SDK-ISAPI功能透传接口

输入参数: lUserID: NET\_DVR\_Login\_V30()的返回值

lpInputParam: 输入参数指针

lpOutputParam：输出参数指针

输入输出参数

返回值: TRUE-成功,FALSE-失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_STDXMLConfig(LONG lUserID, const NET\_DVR\_XML\_CONFIG\_INPUT\* lpInputParam, NET\_DVR\_XML\_CONFIG\_OUTPUT\* lpOutputParam);

设置数据文件锁定配置：

lpInputParam->lpRequestUrl为：PUT /ISAPI/ContentMgmt/FileData/LockConfiguration\r\n；

lpInputParam->lpInBuffer为LockConfiguration；

lpOutputParam->lpOutBuffer为NULL；

lpOutputParam->lpStatusBuffer为ResponseStatus，成功、失败均返回；

* + - 1. XML及结构体

**LockConfiguration XML Block**

|  |
| --- |
| <LockConfiguration version="2.0" xmlns="http://www.isapi.org/ver20/XMLSchema">  <LockList size=”50”>  <LockInfo>  <id> <!-- req, xs:string; id --> </id>  <url><!--req, xs:string, min=”0”,max=”240”--></url>  <operateType><!--req, xs:string, opt=”lock,unlock”--></operateType>  <lockTime><!--dep, xs:string, min=”0”,max=”32”, unit:s, 0xffffffff as Permanent lock--></lockTime>//加锁时长，单位秒，0xffffffff表示永久锁，该字段仅当operateType为lock时有效  </LockInfo>  </LockList>  </LockConfiguration> |

**ResponseStatus XML Block**

|  |
| --- |
| <ResponseStatus version="1.0" xmlns="http://www.std-cgi.org/ver20/XMLSchema">  <requestURL><!-- req, ro,xs:string --> </requestURL>  <statusCode><!-- req, ro,xs:integer --></statusCode>  <!—0和1-OK, 2-Device Busy, 3-Device Error, 4-Invalid Operation, 5-Invalid XML Format, 6-Invalid XML Content; 7-Reboot Required -->  <statusString><!-- req, ro,xs:string --></statusString>  <subStatusCode><!-- req, ro,xs:string --></subStatusCode>  <id> <!-- opt, xs:string; id --> </id>//配置状态, 按位表示, 从第一位开始, 每一位的值表示为相应索引值锁定配置的成功状态, 0-失败, 1-成功  </ResponseStatus> |

* + - 1. 备注

### [新增]数据文件传输速率配置能力

* + - 1. 接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_STDXMLConfig

函数描述: SDK-ISAPI功能透传接口

输入参数: lUserID: NET\_DVR\_Login\_V30()的返回值

lpInputParam: 输入参数指针

lpOutputParam：输出参数指针

输入输出参数

返回值: TRUE-成功,FALSE-失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_STDXMLConfig(LONG lUserID, const NET\_DVR\_XML\_CONFIG\_INPUT\* lpInputParam, NET\_DVR\_XML\_CONFIG\_OUTPUT\* lpOutputParam);

lpInputParam->lpRequestUrl为：GET /ISAPI/ContentMgmt/FileData/LockConfiguration/capabilities\r\n；

lpInputParam->lpInBuffer为NULL；

lpOutputParam->lpOutBuffer为LockConfiguration；

lpOutputParam->lpStatusBuffer为ResponseStatus，失败时返回；

* + - 1. XML及结构体

**LockConfiguration XML Block**

|  |
| --- |
| <LockConfiguration version="2.0" xmlns="http://www.isapi.org/ver20/XMLSchema">  <LockList size=”50”>  <LockInfo>  <id> <!-- req, xs:string; id --> </id>  <url min=”0”,max=”240”><!--req, xs:string--></url>  <operateType opt=”lock,unlock”><!--req, xs:string--></operateType>  <lockTime min=”0”,max=”32”><!--dep, xs:string, unit:s, 0xffffffff as Permanent lock--></lockTime>//加锁时长，单位秒，0xffffffff表示永久锁，该字段仅当operateType为lock时有效  </LockInfo>  </LockList>  </LockConfiguration> |

* + - 1. 备注

无

## [新增]获取数据文件信息

### [新增]获取数据文件信息

* + - 1. 接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_STDXMLConfig

函数描述: SDK-ISAPI功能透传接口

输入参数: lUserID: NET\_DVR\_Login\_V30()的返回值

lpInputParam: 输入参数指针

lpOutputParam：输出参数指针

输入输出参数

返回值: TRUE-成功,FALSE-失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_STDXMLConfig(LONG lUserID, const NET\_DVR\_XML\_CONFIG\_INPUT\* lpInputParam, NET\_DVR\_XML\_CONFIG\_OUTPUT\* lpOutputParam);

获取数据文件信息：

lpInputParam->lpRequestUrl为：POST /ISAPI/ContentMgmt/FileData/SearchFile\r\n；

lpInputParam->lpInBuffer为SearchFileCondition；

lpOutputParam->lpOutBuffer为SearchFileResult；

lpOutputParam->lpStatusBuffer为ResponseStatus，失败时返回；

* + - 1. XML及结构体

**SearchFileCondition XML Block**

|  |
| --- |
| <SearchFileCondition version="2.0" xmlns="http://www.isapi.org/ver20/XMLSchema">  <SearchList size=”50”>  <SearchInfo>  <url><!--req, xs:string, min=”0”,max=”240”--></url>  </SearchInfo>  </SearchList>  </SearchFileCondition> |

**SearchFileResult XML Block**

|  |
| --- |
| <SearchFileResult version=“2.0” xmlns=“http://www.isapi.org/ver20/XMLSchema”>  <FileList size=”50”>  <FileInfo>  <fileExist><!--req,xs:boolean--></fileExist>//文件是否存在  <fileSize><!-- dep, xs: string min=”” max=”” --></fileSize>//文件大小，仅当fileExist为true时才有效  <fileLock><!--dep,xs:boolean--></fileLock>//文件是否加锁，仅当fileExist为true时才有效  <unlockTime><!-- dep, xs: string min=”” max=”” --></unlockTime>//解锁文件时间，仅当fileLock为true时才有效  </FileInfo>  </FileList>  </SearchFileResult> |

* + - 1. 备注

### [新增]获取数据文件信息能力

* + - 1. 接口

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数名: NET\_DVR\_STDXMLConfig

函数描述: SDK-ISAPI功能透传接口

输入参数: lUserID: NET\_DVR\_Login\_V30()的返回值

lpInputParam: 输入参数指针

lpOutputParam：输出参数指针

输入输出参数

返回值: TRUE-成功,FALSE-失败

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

BOOL NET\_DVR\_STDXMLConfig(LONG lUserID, const NET\_DVR\_XML\_CONFIG\_INPUT\* lpInputParam, NET\_DVR\_XML\_CONFIG\_OUTPUT\* lpOutputParam);

lpInputParam->lpRequestUrl为：GET /ISAPI/ContentMgmt/FileData/SearchFile/capabilities\r\n；

lpInputParam->lpInBuffer为NULL；

lpOutputParam->lpOutBuffer为SearchFileCondition；

lpOutputParam->lpStatusBuffer为ResponseStatus，失败时返回；

* + - 1. XML及结构体

**SearchFileCondition XML Block**

|  |
| --- |
| <SearchFileCondition version="2.0" xmlns="http://www.isapi.org/ver20/XMLSchema">  <SearchList size=”50”>  <SearchInfo>  <url min=”0”,max=”240”><!--req, xs:string--></url>  </SearchInfo>  </SearchList>  </SearchFileCondition> |

* + - 1. 备注

无