

# Cloud Storage SDK 使用说明

## CSCloudServer

成员变量

变量名	变量类型	说明
serverName	NSString	云端名称
isWebLogin	BOOL	判断是否为网页登陆
isLogin	BOOL	判断是否登入
serverType	CSCloudServerType	云端对应的枚举值
userName	NSString	登入云端的用户名
password	NSString	登入云端的用户密码，只有非网页登陆的云端能获得用户密码
hostName	NSString	云端（FTP 和 webdave）的主机名称
portID	NSString	FTP 的端口号
fileExtension	NSString	通过文件后缀来获取文件
fileExtensions	NSArray	文件后缀数组
delegate	id<CSCloudServerDelegate>	云端对象的代理
hasDownloadProgress	BOOL	是否有下载进度条
isAutoUpload	BOOL	是否设置自动上传

使用

云端登陆：

初始化一个云端对象，通过云端对应的枚举值来初始化一个 autorelease 类型的云端对象。由于使用云端时有大量的回调信息，所以初始化时最好将 delegate 设置好。当对象被释放时将 delegate 设为 nil。

```
CSCloudServer *cloudServer = [CSCloudServer
cloudServerForType:serverType];

cloudServer.delegate = self;
```

安全登陆云端，通过 `server` 的 `isWeblogin` 来判断是否是网页登陆方式从而确定登陆方法的传参，如果为 YES 则要传一个父视图对象且父视图是 `push` 进去的。如果为 NO 则传 `nil`，但调用登陆方法前必须对 `server` 登陆需要的参数赋值（详见下表）。登陆后返回登陆成功或是失败回调，对于成功登陆的云端会将用户信息保存方便下次登陆。支持多帐号的云端有：Box、DropBox、googleDrive、GoogleDocs、FTP、Mydisk、WebDav、SugarSync，支持单帐号的云端有：EverNote、skyDrive。对于支持单帐号云端，假如已经登陆，如果再次登陆，则直接返回登陆成功回调。

```
if (cloudServer.isWebLogin) {
    [cloudServer loginFromController:self];
}else{
    //获取到登陆云端需要的参数
    [cloudServer loginFromController:nil];
}
```

serverType	cloudServer 参数设置			
	userName	password	hostName	portID
Box	网页登陆无需参数			
DropBox	网页登陆无需参数			
GoogleDrive	网页登陆无需参数			
EverNote	网页登陆无需参数			
skyDrive	网页登陆无需参数			
MyDisk	√	√		
WebDav	√	√	√	
GoogleDocs	√	√		
sugarSync	√	√		
FTP	√	√	√	√

云端退出：

安全退出云端帐号，并返回退出成功或是失败回调。退出成功时，会将系统纪录

的用户帐号信息和登陆标识删除，达到安全退出的状态；对于退出失败，系统保存的帐号信息和登陆标识不会被删除。

```
[cloudServer logout];
```

获取文件列表：

获取根目录：

获取云端根目录下面所有的文件和文件夹信息。获取成功时，返回获取到的文件数组和文件夹数组信息，即两个 cloudFile 对象数组；获取失败，返回云端对象和错误信息。

```
[cloudServer getRootListAndFolder];
```

获取文件目录：

获取指定文件夹里面的全部文件和文件夹信息。成功和失败回调信息同获取根目录。

```
[cloudServer getListAndFolderByFolder:folder];
```

下载文件：

将云端文件下载到指定路径下，并可以通过 [CSDownloadManager sharedManager] 的代理方法获取文件的下载进度以及下载状态等功能。同时，调用添加到下载列表函数将任务添加到下载类表中。下载成功时，返回下载成功回调；下载失败，返回下载文件云端类型和下载失败错误信息。对于 FTP、Evernote 两个云端里面的文件下载时没有下载进度显示，只有下载状态显示。

```
[CSDownloadManager sharedManager].delegate = self;
```

```
[cloudServer downloadFile:fileObject  
toPath:savePath];
```

添加到下载列表：

将云端要下载的任务添加到下载列表中，并显示任务的下载状态，添加到下载列表中的任务可以控制其下载进度。注意事项：GoogleDrive、FTP、Evernote 没有添加到下载列表操作。

```
[cloudServer addToDownloadList:file  
withSavePath:savePath];
```

上传文件：

将本地文件上传到指定云端的指定目录下面，并返回上传成功或是失败回调。包括上传单个文件和上传多个文件功能。上传注意事项：对于 skyDrive、SugarSync、FTP 上传同名文件时，会覆盖原来的文件；DropBox 同名文件他会自动添加后缀，GoogleDrive、GoogleDocs 允许同名文件存在；Box、MyDisk、WebDav 上传同名文件失败；EverNote 不能上传内容一模一样的文件，尽管文件名字不一样，上传也会失败。

```
[cloudServer uploadFilePath:localFilePath
```

```
toFolder:destinationFolder];
```

```
[cloudServer uploadFiles:localFilePathArr
```

```
toFolder:destinationFolder];
```

新建文件夹:

在当前云端当前目录下面，创建一个新的文件夹，返回新建成功或是失败回调。对于云端 SugarSync 没有新建文件夹功能；对于云端 EverNote，只能在根目录下面创建 note 文件夹，不能在 note 下面创建 note 文件夹。另外，skyDrive、WebDav、MyDisk、FTP、Box、DropBox 不允许新建同名文件夹；EverNote、GoogleDrive、GoogleDocs 允许新建同名文件夹。

```
[cloudServer createNewFolder:folderName
```

```
inFolder:destinationFolder];
```

删除文件:

删除指定文件对象，返回删除成功或是失败回调。删除文件功能注意事项：Box、DropBox、GoogleDrive、GoogleDocs、skyDrive 同时支持删除文件和文件夹功能；FTP 只能删除文件，不能删除文件夹；Evernote 只能删除文件夹，不能删除文件；MyDisk、WebDav、SugarSync 没有删除功能。

```
[cloudServer deleteFile:fileObject];
```

文件重命名:

对指定的文件对象进行重命名操作，传入文件的新名字不需要文件的后缀，并返回重命名成功或失败回调。注意事项：Box、DropBox、skyDrive 不允许同名重命名，GoogleDrive、EverNote 允许同名重命名，GoogleDocs、FTP、MyDisk、WebDav、SugarSync 没有重命名功能。

```
[cloudServer renameFile:fileObject
```

```
withName:newName];
```

在线编辑:

在线编辑功能为 GoogleDrive 特有功能，可以对部分类型文件进行在线编辑，保存。

```
[cloudServer getOnlineEditorURL: file];
```

查看文件权限拥有者:

查看文件权限拥有者功能，目前只支持 GoogleDrive 文件，查看哪些用户对这个文件拥有访问权限。

```
[cloudServer getPermissionList:file];
```

添加文件权限访问者

添加其他用户对本文件的访问权限（读或是写）。

```
[cloudServer insertPermission:userName  
wihtRole:@"reader" toFile:file];
```

删除某个用户对文件的访问权限

删除某个用户对文件的访问权限，不能删除文件创建者。

```
[cloudServer deletePermission:file  
atPermissionID:permission];
```

## CloudFile

成员变量

变量名	变量类型	说明
fileName	NSString	文件的名字
fileObject	id	文件对象
serverType	CSCloudServerType	属于哪个云端的文件
canOnline	BOOL	判断文件是否能在线编辑
fileState	CSCloudFileState	文件的状态

使用：

文件状态枚举类型：

```
typedef enum{  
    CSFileNone = 0, //无操作  
    CSFileDelete, //文件正在删除  
    CSFileDownloading, //文件正在下载  
    CSFileRenam //文件正在重命名  
}CSCloudFileState;
```

## CSDownloadTask

成员变量

变量名	变量类型	说明
mAddress	NSString	下载文件源地址
mFileName	NSString	下载文件的名字
mReceivedSize	SInt64	下载时接收到的数据大小
mFileSize	SInt64	下载文件的大小
mDownloadState	CSDOWNLOAD_STATE	文件下载的状态
serverType	CSCloudServerType	下载文件所属的云端
delegate	Id<CSTaskDelegate>	代理
token	NSString	云端登陆唯一标识
savepath	NSString	文件存放路径
loadData	NSData	存放已经下载数据

代理方法

**@protocol** CSTaskDelegate <NSObject>

**@optional**

任务的下载状态回调

当任务的下载状态改变时会调用，任务下载状态有：正在下载、暂停下载、完成下载、下载失败、待操作状态。

– **(void)**taskStateChange:(CSDownloadTask \*) **task**;

任务的下载进度回调

当任务正在下载时且下载的数据有改变时调用，EverNote、FTP没有返回下载进度功能。

– **(void)**taskDownloadProgress:(CSDownloadTask \*)  
**task**;

任务下载成功回调

– **(void)**taskDownloadSuccess:(NSString \*)name;

任务下载失败回调

```
- (void)taskDownloadFail:(NSString *)name  
withError:(NSError *)error;  
@end
```

使用:

开始下载任务

开始下载本任务，或是继续下载本任务。当继续下载暂停任务时，下载未完成的数据，而不是重新下载。

```
[task startRunning] ;
```

暂停下载

将当前任务暂停，此时，下载状态将会有缓存，以便下次继续下载。

云端Evernote，FTP、GoogleDrive不能控制下载进度。

```
[task stopRunning] ;
```

删除任务

删除当前任务，包括正在下载、下载失败等状态的任务，同时会将下载的缓存数据清空。

```
[task deleteTask] ;
```

判断是否为同一个任务

```
- (BOOL)isEqualToTask:(CSDownloadTask *)task;
```

## CSDownloadManager

成员变量

变量名	变量类型	说明
taskArray	NSMutableArray	保存下载任务数组

下载任务的状态枚举

```
typedef enum {  
    kCSDownloadStateNone = 0, //任务无状态  
    kCSDownloadStateRunning , //正在下载  
    kCSDownloadStateStopped , //暂停下载  
    kCSDownloadStateFinished , //下载完成  
    kCSDownloadStateFailed //下载失败  
} CSDOWNLOAD_STATE;
```

删除指定下载任务

可以删除任何一种下载状态的任务，包括正在下载、下载完成等状态的任务。

```
-(void) deleteTask:(CSDownloadTask *)task;
```

删除所有已完成任务

删除已经下载完成的任务。

```
-(void)deleteFinishTask;
```

开始下载

开始下载所有任务，不包括已经下载完成、下载失败和正在下载的任务。

```
-(void)startDownload;
```

改变任务下载状态

```
-(void)changeTaskState:(CSDownloadTask *)task;
```



保存历史下载列表

```
-(void)saveDownloadTask;
```

## CSPermission

成员变量

变量名	变量类型	说明
permissionName	NSString	权限者名字
permissionRole	NSString	拥有权限
permissionnObject	id	拥有权限者对象