5 视频下载

`PLVVodDownloadManager` 是点播 SDK 中的一个单例类，主要负责视频的下载、下载信息回调和下载队列的管理。下载信息封装在类 `PLVVodDownlaodInfo` 中，每一个下载任务对应一个 `PLVVodDownlaodInfo` 对象。

下载器的使用详见 demo 中 `PLVDownloadManagerViewController` 的实现。

5.1 下载设置

在进行下载之前，我们需要先进行一些设置。

5.1.1 AppDelegate

首先，在 `AppDelegate` 中，我们需要进行以下配置：

```
- (void)application:(UIApplication *)application handleEventsForBackgroundURLSession:(NSString *)identifier
    completionHandler:(void (^)(void))completionHandler {
    // 处理后台下载完成回调的事件
    [[PLVVodDownloadManager sharedManager] handleEventsForBackgroundURLSession:identifier
completionHandler:completionHandler];
}
- (void)applicationDidEnterBackground:(UIApplication *)application {
    // app 退到后台时进行相关设置
    [[PLVVodDownloadManager sharedManager] applicationDidEnterBackground];
}

- (void)applicationWillEnterForeground:(UIApplication *)application {
    // app 回到前台时进行相关设置
    [[PLVVodDownloadManager sharedManager] applicationWillEnterForeground];
}

- (void)applicationWillTerminate:(UIApplication *)application {
    // 程序即将结束时，修改并保存下载中视频的下载状态，用于程序下次启动后，恢复视频下载状态
    [[PLVVodDownloadManager sharedManager] applicationWillTerminate];
}
```

5.1.2 下载器设置

`PLVVodDownloadManager` 包含以下下载属性，开发者可以根据需要进行设置，也可以保持默认设置。

5.1.2.1 设置下载目录

下载目录默认为沙盒文件夹 "Library/Caches/PolyvVodCache"。注意，修改下载目录后，上一个目录的离线视频将无法索引。修改下载目录可以通过修改 `PLVVodDownloadManager` 的属性 `downloadDir`，也可以通过方法 `-setDownloadDir:skipBackup:error:` 进行修改。

```
// 下载目录，默认为 Library/Caches/PolyvVodCache
@property (nonatomic, copy) NSString *downloadDir;

/**
 设置下载目录
 @param downloadDir 设置的下载目录
 @param skipBackup 是否忽略 iCloud 的备份
 @param error 错误回调
 @return 是否成功
 */
- (BOOL)setDownloadDir:(NSString *)downloadDir skipBackup:(BOOL)skipBackup error:(NSError **)error;
```

5.1.2.2 是否允许使用蜂窝移动网络进行下载

默认允许，如需禁止使用蜂窝移动网络进行下载，需要修改属性值 `allowsCellularAccess` 为 NO。

```
// 是否允许使用蜂窝移动网络进行下载，默认 YES
@property (nonatomic, assign) BOOL allowsCellularAccess;
```

5.1.2.3 下载最大并发数

`PLVVodDownloadManager` 的下载并发数通过 `maxRuningCount` 设置，默认是 1，最大可设为 3。

```
// 允许同时下载的最大任务数，默认为1，最大为3
@property (nonatomic, assign) NSUInteger maxRuningCount;
```

5.2 下载信息模型

`PLVVodDownloadInfo` 是下载信息模型类，包含如下只读属性：

```
// 唯一标识
@property (nonatomic, copy, readonly) NSString *identifier;

// 队列ID
@property (nonatomic, assign, readonly) NSInteger downloadId;

// PLVVodVideo 对象
@property (nonatomic, strong, readonly) PLVVodVideo *video;

// vid 便捷属性
@property (nonatomic, copy, readonly) NSString *vid;

// 清晰度
@property (nonatomic, assign, readonly) PLVVodQuality quality;
// 文件类型,默认为视频
@property (nonatomic, assign, readonly) PLVDownloadFileType fileType;


// 下载相关
@property (nonatomic, assign, readonly) PLVVodDownloadState state; // 下载状态
@property (nonatomic, assign, readonly) double bytesPerSeconds; // 下载速率（单位：byte/s）
@property (nonatomic, assign, readonly) double progress; // 下载进度（0-1）
@property (nonatomic, assign, readonly) double unzipProgress; // 解压进度（0-1）
@property (nonatomic, strong, readonly) NSError *error; // 下载错误


// UI展示
@property (nonatomic, copy, readonly) NSString *snapshot; // 封面
@property (nonatomic, copy, readonly) NSString *title;    // 视频名称
@property (nonatomic, assign, readonly) NSUInteger filesize; // 文件大小
```

其中，清晰度的枚举值定义参见点播文档 4.2.1，下载状态、文件类型的枚举定义如下：

```
// 下载状态
typedef NS_ENUM(NSInteger, PLVVodDownloadState) {
    PLVVodDownloadStatePreparing = 0,        // 准备，默认值
    PLVVodDownloadStatePreparingStart = 7, // 准备中，正在创建任务
    PLVVodDownloadStateReady = 1,            // 就绪，下载任务已创建
    PLVVodDownloadStateRunning = 2,                  // 正在下载
    PLVVodDownloadStateStopping = 3,         // 正在停止
    PLVVodDownloadStateStopped = 4,                  // 停止下载
    PLVVodDownloadStateSuccess = 5,                  // 下载成功
    PLVVodDownloadStateFailed = 6            // 下载失败
};

// 下载文件类型
typedef NS_ENUM(NSUInteger, PLVDownloadFileType) {
    PLVDownloadFileTypeVideo = 1 << 0,  // 视频文件
    PLVDownloadFileTypeAudio = 1 << 1   // 音频文件
};
```

5.3 视频下载

5.3.1 新增下载任务

使用方法 `-downloadVideo:` 将视频对象 `PLVVodVideo` 添加到下载队列中，如果属性 `autoStart` 为默认值 NO，那么需要调用方法 `-startDownload` 启动下载，否则则不需要。相关属性、方法声明如下：

```
// 添加任务后是否自动启动，默认 NO
@property (nonatomic, assign) BOOL autoStart;

/**
 使用后台设置的默认画质添加至下载队列
 @param video PLVVodVideo 视频对象
 @return 下载信息
 */
- (PLVVodDownloadInfo *)downloadVideo:(PLVVodVideo *)video;
/**
 开始队列下载
 */
- (void)startDownload;

/**
 停止队列下载
 */
- (void)stopDownload;
```

添加视频并开始/停止下载代码示例如下：

```
[[PLVVodDownloadManager sharedManager] downloadVideo:video]; // 将视频 video 添加到下载队列
[[PLVVodDownloadManager sharedManager] startDownload]; // 开始下载
[[PLVVodDownloadManager sharedManager] stopDownload]; // 停止下载
```

使用如下方法还可以启动、停止指定 `vid` 的视频：

```
/**
 从指定视频开始下载
 @param vid 视频vid
 */
- (void)startDownloadWithVid:(NSString *)vid;

/**
 停止下载指定视频
 @param vid 视频vid
 */
- (void)stopDownloadWithVid:(NSString *)vid;
```

5.3.2 移除下载任务

移除下载任务方法声明如下：

```
/**
 移除下载任务，并删除对应文件
 @param vid vid
 @param error 错误回调
 */
- (void)removeDownloadWithVid:(NSString *)vid error:(NSError **)error;

/**
 移除所有下载任务，并删除对应文件，与已完成下载的视频无关
 @param completion 删除任务后的回调
 */
- (void)removeAllDownloadWithComplete:(void(^)(void *result))completion;
```

注意，任务移除之后，对应的视频文件也会被删除。移除所有下载任务不包括已成功下载的。

5.3.3 读取下载任务

使用如下方法，可以获取所有已成功下载的视频信息/本地视频，也可以根据 `vid` 获取指定的视频下载信息。

```
/**
 获取所有已缓存成功视频信息
 @return 下载完成的视频数组
 */
- (NSArray<PLVVodDownloadInfo *> *)requestDownloadCompleteList;

/**
 根据vid 获取视频下载信息
 @param vid 视频vid
 */
- (PLVVodDownloadInfo *)requestDownloadInfoWithVid:(NSString *)vid;

/**
 获取已下载的本地视频
 @return 一组 PLVVodLocalVideo 对象
 */
- (NSArray<PLVVodLocalVideo *> *)localVideos;
```

5.4 下载回调

5.4.1 下载器回调

`PLVVodDownloadManager` 使用以下回调对视频下载进行监听：

```
// 完成所有下载回调
@property (nonatomic, copy) void(^completeBlock)(void);

// 单个视频下载完成回调
@property (nonatomic, copy) void(^downloadCompleteBlock)(PLVVodDownloadInfo *info);

// 下载错误回调
@property (nonatomic, copy) void (^downloadErrorHandler)(PLVVodVideo *video, NSError *error);
```

示例代码如下：

```
PLVVodDownloadManager *downloadManager = [PLVVodDownloadManager sharedManager];

downloadManager.downloadCompleteBlock = ^(PLVVodDownloadInfo *info) {
    NSLog(@"video vid %@ download success", info.vid);
};

downloadManager.downloadErrorHandler = ^(PLVVodVideo *video, NSError *error) {
    NSLog(@"video vid %@ download error: %@", video.vid, error);
};
downloadManager.completeBlock = ^{
    NSLog(@"all video download success");
};
```

5.4.2 下载任务回调

每一个视频下载信息对象 `PLVVodDownloadInfo` 也有自己的回调，开发者通过实现这些回调，可以进行 UI 的更新：

```
// 下载状态变更回调
@property (nonatomic, copy) void (^stateDidChangeBlock)(PLVVodDownloadInfo *info);

// 下载速率（单位：byte/s）变更回调
@property (nonatomic, copy) void (^bytesPerSecondsDidChangeBlock)(PLVVodDownloadInfo *info);

// 下载进度（0-1）变更回调
@property (nonatomic, copy) void (^progressDidChangeBlock)(PLVVodDownloadInfo *info);

// 解压进度（0-1）变更回调
@property (nonatomic, copy) void (^unzipProgressDidChangeBlock)(PLVVodDownloadInfo *info);
```