

- [1 功能概述](#)
- [2 使用演示](#)
- [3 实现介绍](#)
 - [3.1 PLVStreamerPPTView](#)

1 功能概述

PPT&白板模块包括白板、PPT展示、画笔标注等功能。PPT&白板展示界面被封装在类 `PLVLSDocumentAreaView` 中。

2 使用演示

使用PPT&白板模块需在 `PLVLSStreamerViewController` 中添加 `PLVLSStatusAreaView` 控件：

```
self.documentAreaView = [[PLVLSDocumentAreaView alloc] init];
self.documentAreaView.delegate = self;
[self.view addSubview:self.documentAreaView];
```

同时 `PLVLSStreamerViewController` 需遵循协议 `PLVLSDocumentAreaViewDelegate`，并实现以下回调：

```
#pragma mark - PLVSDocumentAreaView Delegate

- (void)documentAreaView:(PLVLSDocumentAreaView *)documentAreaView openBrush:(BOOL)isOpen {
    // isOpen - YES: 打开画笔隐藏聊天区域
    // isOpen - NO: 打开画笔显示聊天区域
}

- (void)documentAreaView:(PLVLSDocumentAreaView *)documentAreaView changeFullScreen:(BOOL)isFullScreen {
    self.fullscreen = isFullScreen; // 布尔值 fullscreen 表示当前页面是否处于 ppt 全屏状态
    if (fullscreen) {
        [self.view insertSubview:self.documentAreaView aboveSubview:self.statusAreaView];
    } else { // 全屏时将 documentAreaView 控件插入到子视图的最顶部
        [self.view insertSubview:self.documentAreaView atIndex:0];
    }
}
```

并在 `-viewWillLayoutSubviews` 方法中，对 `PLVLSStatusAreaView` 进行布局：

```
if (self.isFullscreen) {
    self.documentAreaView.frame = self.view.bounds;
} else {
    self.documentAreaView.frame = CGRectMake(PLVLSUtils.safeSidePad, 44, documentAreaViewWidth,
    documentViewHeight);
}
```

3 实现介绍

PPT&白板视图 `PLVLSStatusAreaView` 主要由以下控件组成，其中最重要的，就是负责白板&PPT展示的 `PLVStreamerPPTView` 类：

```
@property (nonatomic, strong) PLVStreamerPPTView *pptView;           // PPT 功能模块视图
@property (nonatomic, strong) PLVLSDocumentNumView *pageNum;        // 页码
@property (nonatomic, strong) PLVLSDocumentToolView *toolView;      // 控制条视图
@property (nonatomic, strong) PLVLSDocumentBrushView *brushView;    // 画笔条视图
```

3.1 PLVStreamerPPTView

`PLVStreamerPPTView` 位于 `PolyvLiveCommonModule` 文件夹下，UI 方面由一个 `WKWebView` 的实例对象组成，依赖 SDK 的 `PLVWebViewBridge` 类进行原生与 H5 的通信，实现各种交互功能。

`PLVStreamerPPTView` 对外提供以下只读属性，获取PPT&白板视图的状态属性：

```
@property (nonatomic, assign, readonly) NSInteger autoId;          // ppt id, 0是白板
@property (nonatomic, assign, readonly) NSInteger currPageNum;      // 当前页码
@property (nonatomic, assign, readonly) NSInteger totalPageNum;     // 总页码
@property (nonatomic, assign, readonly) NSInteger pptStep;         // 当前文档所处于动画步数
```

提供以下方法，用于实现原生与 H5 的通信：

```

/// 设置画板是否处于可绘制状态
/// @param open 打开或关闭画板
- (void)setPaintStatus:(BOOL)open;

/// 设置画笔类型
/// @param type line - 自由笔; text - 文字; arrowLine - 箭头
- (void)setDrawType:(PLVWebViewBrushPenType)type;

/// 完成文本输入
- (void)changeTextContent:(NSString *)content;

/// 修改画笔颜色
/// @param hexString RGB色值, 如红色为“#FF0000”
- (void)changeColor:(NSString *)hexString;

/// 进入画笔删除状态
- (void)toDelete;

/// 删除所有画笔
- (void)deleteAllPaint;

/// 告诉 h5 现在开始上课, h5 会清空画板
/// @param jsonDict 开始推流时发送的 socket 消息
- (void)setSliceStart:(NSDictionary *)jsonDict;

```

提供以下方法，用于完成白板、文档的切换与翻页等功能：

```

/// 白板、文档间的切换方法
/// 如果切换的是文档，切换成功后触发回调 '-jsbridge_documentChangeWithAutoId:imageUrls:'
/// @param autoId 切换的文档的 autoId, 如果是白板 autoId 为 0
/// @param pageNumber 切换到文档的第几页
- (void)changePPTWithAutoId:(NSUInteger)autoId pageNumber:(NSInteger)pageNumber;

/// 当前白板/文档下的翻页方法
- (void)turnPage:(BOOL)isNextPage;

/// 新增白板方法
- (void)addWhiteboard;

```

定义了协议 `PLVStreamerPPTViewDelegate`，提供以下回调，用于页面 UI 的更新：

```
@protocol PLVStreamerPPTViewDelegate <NSObject>

/// PPT加载完成准备就绪回调
///
/// @param streamerPPTView PPT视图对象
- (void)streamerPPTViewDidLoad:(PLVStreamerPPTView *)streamerPPTView;

/// PPT加载失败回调
///
/// @param streamerPPTView PPT视图对象
- (void)streamerPPTView:(PLVStreamerPPTView *)streamerPPTView loadFailWithError:(NSError *)error;

/// webView 上准备输入文字时回调
/// @param streamerPPTView PPT视图对象
/// @param inputText 输入文本
/// @param textColor 文本颜色字符串 格式: #FFFFFF
- (void)streamerPPTView:(PLVStreamerPPTView *)streamerPPTView inputWithText:(NSString *)inputText textColor:
(NSString *)textColor;

/// 切换到新文档时的回调
/// 用于切换文档时更新文档缩略图
/// @param streamerPPTView PPT视图对象
/// @param autoId 文档autoId
/// @param imageUrls 文档页面缩略图
- (void)streamerPPTView:(PLVStreamerPPTView *)streamerPPTView changeWithAutoId:(NSUInteger)autoId imageUrls:
(NSArray *)imageUrls;

/// 文档、白板页码变化的回调、
///
/// @param streamerPPTView PPT视图对象
/// @param autoId 文档autoId
/// @param pageNumber 文档当前页码
- (void)streamerPPTView:(PLVStreamerPPTView *)streamerPPTView pageStatusChangeWithAutoId:(NSUInteger)autoId
pageNumber:(NSUInteger)pageNumber totalPage:(NSUInteger)totalPage pptStep:(NSUInteger)step;

@end
```