

- 1 功能概述
- 2 接收商品实时消息
- 3 商品推送
 - 3.1 PLVECCommodityPushView 类介绍
 - 3.2 PLVECCommodityPushView 类使用
- 4 商品列表
- 5 核心Common和SDK
- 6 商品详情

1 功能概述

商品模块是在直播带货中可以根据直播实时情况，由后台配置操作新增、删除、修改信息、排序商品，并有商品推送功能实时同步展示到观看端上，主要三个页面分别有：商品推送 PLVECCommodityPushView、商品列表 PLVECCommodityViewController 和商品详情 PLVECGoodsDetailViewController，三个页面展示和商品实时消息功能是在 PLVECHomePageView 中实现的。

2 接收商品实时消息

接收商品实时消息能够及时的同步后台商品配置操作到观看端，已保证观众看到的商品数据是最新的。商品实时消息目前包括两大类：配置操作商品和推送商品。

PLVECHomePageView.m

```
- (void)productMessageEvent:(NSDictionary *)jsonDict {
    NSString *subEvent = PLV_SafeStringForDictKey(jsonDict, @"EVENT");
    if (![subEvent isEqualToString:@"PRODUCT_MESSAGE"]) {
        return;
    }

    if (self.shoppingCardButton.isHidden) {
        return; // 未开启商品功能
    }

    NSInteger status = PLV_SafeIntegerForDictKey(jsonDict, @"status");

    // 配置操作商品
    if (self.commodityVC) {
        [self.commodityVC receiveProductMessage:status content:jsonDict[@"content"]];
    }

    // 处理推送商品
    if (9 == status) {
        NSDictionary *content = PLV_SafeDictionaryForDictKey(jsonDict, @"content");
        PLVCommodityModel *model = [PLVCommodityModel commodityModelWithDict:content];
        [self.pushView setModel:model];
    }
}
```

3 商品推送

商品推送是保利威云直播后台中“观看页管理-商品库-操作”列“推送”按钮功能，点击“推送”按钮直播带货场景会接收到商品实时消息，弹出单个商品视图 `PLVECCommodityPushView`，5秒后自动消失。

3.1 `PLVECCommodityPushView` 类介绍

商品视图 `PLVECCommodityPushView` 类是UI布局类，类定义如下：

```

@protocol PLVECCommodityPushViewDelegate <NSObject>

/// 点击跳转详情页回调
- (void)jumpToGoodsDetail:(NSURL *)goodsURL;

@end

@interface PLVECCommodityPushView : UIView

/// 商品图片
@property (nonatomic, strong) UIImageView *coverImageView;

/// 序号Id
@property (nonatomic, strong) UILabel *showIdLabel;

/// 商品名称
@property (nonatomic, strong) UILabel *nameLabel;

/// 商品价格
@property (nonatomic, strong) UILabel *realPriceLabel;

/// 购买价格
@property (nonatomic, strong) UILabel *priceLabel;

/// 关闭按钮
@property (nonatomic, strong) UIButton *closeButton;

/// 跳转详情按钮
@property (nonatomic, strong) UIButton *jumpButton;

/// 商品数据模型
@property (nonatomic, strong) PLVCommodityModel *model;

@property (nonatomic, weak) id<PLVECCommodityPushViewDelegate> delegate;

/// 销毁
- (void)destroy;

@end

```

3.2 PLVECCommodityPushView 类使用

PLVECCommodityPushView 类初始化、显示、销毁等操作是在 PLVECHomePageView 中实现管理的，具体使用代码可以查看Demo中 PLVECHomePageView 类。

4 商品列表

商品列表功能以 PLVECCommodityViewController 为入口控制商品列表视图 PLVECCommodityPushView 类和数据管理 PLVECCommodityModelsManager 类。

- 类定义介绍

```

@protocol PLVECommodityDelegate <NSObject>

/// 商品跳转代理方法
- (void)jumpToGoodsDetail:(NSURL *)goodsURL;

@end

@interface PLVECommodityViewController : UIViewController

@property (nonatomic, weak) id<PLVECommodityDelegate> delegate;

/// 处理socket商品信息
///
/// @param status 商品操作信息类型
/// @param content 商品信息
- (void)receiveProductMessage:(NSInteger)status content:(id)content;

@end

```

- 主要功能介绍
 - 初始化商品列表视图类并显示；
 - 初始化数据管理类，加载接口数据并绑定数据到商品列表；
 - 传递配置操作商品实时到数据管理类；
 - 将跳转商品详情事件传递到 `PLVEHomePageView`；

`PLVECommodityViewController` 类的使用代码可以查看Demo中 `PLVEHomePageView` 类。

5 核心Common和SDK

商品列表数据是调用Common层 [6_2 核心common-数据](#)中 `PLVRoomData` 定义的中间接口，Common调用SDK中 `PLVLiveVideoAPI` 定义接口来获取的。

- 商品列表数据实现接口

PLVECCommodityModelsManager.m

```
- (void)requestCommodityInfo:(NSInteger)rank completion:(void (^)(NSError *))completion {
    if (self.loading || self.totalItems == self.models.count) {
        return;
    }

    _loading = YES;

    PLVRoomData *roomData = [PLVRoomDataManager sharedManager].roomData;
    if (rank == 0) {
        [self.models removeAllObjects];
    }

    __weak typeof(self)weakSelf = self;
    [roomData requestCommodityList:roomData.channelId.integerValue
                        rank:rank
                        count:20
                        completion:^(NSUInteger total, NSArray<PLVCommodityModel *> *commoditys) {
        weakSelf.loading = NO;
        weakSelf.totalItems = total;

        [weakSelf.models addObjectsFromArray:commoditys];

        if (completion) {
            completion(nil);
        }
    } failure:^(NSError * _Nonnull error) {
        weakSelf.loading = NO;
        NSLog(@"requestCommodityInfo error description: %@",error.localizedDescription);
        if (completion) {
            completion(error);
        }
    }];
}
```

- Common层接口定义

```

PLVRoomData.h

/**
 获取商品列表

@param channelId 频道号
@param rank 排序号 （加载序号以前的商品）
@param count 获取列表数据大小，默认10条，最大20
@param completion 加载成功 total: 商品总数, commoditys: 当前拉取的数据列表
(返回商品列表顺序与后台显示顺序一致)
@param failure 加载失败
*/
- (void)requestCommodityList:(NSInteger)channelId rank:(NSInteger)rank count:(NSInteger)count completion:
(void (^)(NSInteger total, NSArray<PLVCommodityModel *> *commoditys))completion failure:(void (^)(NSError
*))failure;

```

- SDK层接口定义

```

/**
PLVLiveVideoAPI.h

 获取商品列表

@param channelId 频道号
@param rank 排序号 （加载序号以前的商品）
@param count 获取列表数据大小，默认10条，最大20
@param completion 加载成功 total: 商品总数, commoditys: 当前拉取的数据列表
(返回商品列表顺序与后台显示顺序一致)
@param failure 加载失败
*/
+ (void)loadCommodityList:(NSInteger)channelId rank:(NSInteger)rank count:(NSInteger)count completion:(void
(^)(NSInteger, NSArray<PLVCommodityModel *> *))completion failure:(void (^)(NSError *))failure;

```

6 商品详情

商品详情功能是点击推送页或者商品列表购买按钮，打开保利威后台配置的商品通用链接Url地址。

商品详情页是 `PLVECGoodsDetailViewController` 类来实现的，主要是WKWebView加载商品通用链接地址。

类定义如下：

```

@interface PLVECGoodsDetailViewController : UIViewController

- (instancetype)initWithGoodsURL:(NSURL *)URL;

@end

```

具体实现代码可以查看Demo中 `PLVECGoodsDetailViewController`