

乘客端接口说明（iOS）

MAPassengerRouteManager

状态枚举

```
///行程状态
typedef NS_ENUM(NSInteger, MASCTXRouteStatus)
{
    MASCTXRouteStatusUnspecified = 0, ///<未指定状态
    MASCTXRouteStatusPickupPassenger = 1, ///<去接乘客
    MASCTXRouteStatusWaitPassenger = 2, ///<等待乘客上车
    MASCTXRouteStatusPassengerOnBoard = 3, ///<乘客已上车
    MASCTXRouteStatusOrderComplete = 4, ///<订单已结束
    MASCTXRouteStatusOffline = 5, ///<离线状态，只走降级策略获取司机位置，内部不做其他请求
};
```

乘客端控制器

```
///司乘同显乘客端Map控制器
@interface MAPassengerRouteManager : NSObject

///司机位置过期时间 默认120s
@property (nonatomic, assign) NSTimeInterval driverPositionExpiredTime;

///路况刷新时间间隔，默认60s
@property (nonatomic, assign) NSTimeInterval trafficRefreshInterval;

///车辆刷新时间间隔，默认10s
@property (nonatomic, assign) NSTimeInterval carRefreshInterval;

///起点标注
@property (nonatomic, strong, readonly) MAPointAnnotation *beginAnnotation;

///终点标注
@property (nonatomic, strong, readonly) MAPointAnnotation *endAnnotation;

///车辆标注
@property (nonatomic, strong, readonly) MAAnimatedAnnotation *carAnnotation;

///是否自动调整地图显示区域，默认为YES。默认行为：在司机位置回调后，调整地图可视范围以显示carAnnotation和终点annotation。注意，当前版本内部自动调整不在选中annotation。如果要自定义此行为，请设置此属性为NO，且在routeStatusChangeForManager回调中对地图进行操作，获取添加timer进行自定义操作。
@property (nonatomic, assign) BOOL automaticAdjustMapRegion;

///调整地图显示区域时间间隔，默认10s
@property (nonatomic, assign) NSTimeInterval autoAdjustMapTimeInterval;

///地图显示区域改变后，自动调整地图时间间隔，默认10s
@property (nonatomic, assign) NSTimeInterval autoAdjustMapAfterRegionChangedTimeInterval;

///默认调整地图时需要设置的padding，默认为(30, 30, 80, 30)
@property (nonatomic, assign) UIEdgeInsets automaticAdjustPadding;

///小车不进行动画的移动距离，单位米。默认为800。
@property (nonatomic, assign) double ignoreCarAnimationDistance;

///是否开启log输出。默认为NO。
```

```
/// 是否显示步行线，默认为NO。
```

```
@property (nonatomic, assign) BOOL enableDebugLog;
```

```
///是否显示步行线，默认为NO。
```

```
@property (nonatomic, assign) BOOL showsWalkPolyline;
```

```
///小车图片
```

```
@property (nonatomic, strong) UIImage *carImage;
```

```
///起点图片
```

```
@property (nonatomic, strong) UIImage *startImage;
```

```
///终点图片
```

```
@property (nonatomic, strong) UIImage *endImage;
```

```
///路况纹理 - 拥堵
```

```
@property (nonatomic, strong) UIImage *blockedImage;
```

```
///路况纹理 - 非常缓慢
```

```
@property (nonatomic, strong) UIImage *verySlowImage;
```

```
///路况纹理 - 缓慢
```

```
@property (nonatomic, strong) UIImage *slowImage;
```

```
///路况纹理 - 畅通
```

```
@property (nonatomic, strong) UIImage *goodImage;
```

```
///路况纹理 - 未知
```

```
@property (nonatomic, strong) UIImage *unknownImage;
```

```
///路线默认纹理
```

```
@property (nonatomic, strong) UIImage *defaultImage;
```

```
///步行路线纹理
```

```
@property (nonatomic, strong) UIImage *walkImage;
```

```
///线宽，默认 18
```

```
@property (nonatomic, assign) CGFloat lineWidth;
```

```
///行程状态，设置行程状态会改变manager的行为
```

```
@property (nonatomic, assign) MASCTXRouteStatus status;
```

```
/**
```

```
 * @brief 初始化方法
```

```
 * @param mapView 地图对象，必填
```

```
 * @param delegate 代理，必填
```

```
 * @return MAPPassengerRouteManager实例
```

```
 */
```

```
- (id)initWithMapView:(MAMapView *)mapView delegate:(id<MAPPassengerRouteManagerDelegate>)delegate;
```

```
/**
```

```
 @brief 设置订单信息，如果没有经纬度，请传 kCLLocationCoordinate2DInvalid。注意：如果endPosition传invalid，则送乘客阶段不显示路线。
```

```
 @param orderId 订单id
```

```
 @param begin 起点坐标
```

```
 @param end 终点坐标
```

```
 */
```

```
- (void)setOrderInfoWith:(NSString *)orderId
```

```
        beginPosition:(CLLocationCoordinate2D)begin
```

```
        endPosition:(CLLocationCoordinate2D)end;
```

```
/**
```

```
 @brief 设置小车弹出框，如果执行了此接口，则不显示默认callout。如果传入nil，则恢复显示默认callout。
```

```
 @param calloutView 气泡显示内容
```

```
 */
```

```

    /
- (void)updateDriverInfoWith:(UIView *)calloutView;

/**
 @brief 设置起点annotation弹出框
 @param calloutView 气泡显示内容
 */
- (void)updateStartAnnotationInfoWith:(UIView *)calloutView;

/**
 @brief 设置终点annotation弹出框
 @param calloutView 气泡显示内容
 */
- (void)updateEndAnnotationInfoWith:(UIView *)calloutView;

/**
 @brief 设置订单信息，只在默认calloutView时有效果，如果设置了自定义calloutView，则无效。
 @param distance 剩余距离，单位米
 @param time 预估时间，单位秒
 @param price 预估价格，默认只在送驾时有效，单位元
 */
- (void)updateDriverInfoWithDistance:(double)distance time:(NSTimeInterval)time price:(double)price;

///判断是否是内部overlay
- (BOOL)isInternalOverlay:(id<MAOverlay>)overlay;

#pragma mark - MapView代理相关方法

/**
 * @brief 在mapView对应的delegate方法内调用此方法
 * @param mapView 地图View
 * @param animated 是否动画
 */
- (void)mapView:(MAMapView *)mapView regionWillChangeAnimated:(BOOL)animated;

/**
 * @brief 在mapView对应的delegate方法内调用此方法
 * @param mapView 地图View
 * @param animated 是否动画
 */
- (void)mapView:(MAMapView *)mapView regionDidChangeAnimated:(BOOL)animated;

/**
 * @brief 在mapView对应的delegate方法内调用此方法
 * @param mapView 地图view
 * @param wasUserAction 标识是否是用户动作
 */
- (void)mapView:(MAMapView *)mapView mapWillMoveByUser:(BOOL)wasUserAction;

/**
 * @brief 在mapView对应的delegate方法内调用此方法
 * @param mapView 地图view
 * @param wasUserAction 标识是否是用户动作
 */
- (void)mapView:(MAMapView *)mapView mapDidMoveByUser:(BOOL)wasUserAction;

/**
 * @brief 在mapView对应的delegate方法内调用此方法
 * @param mapView 地图view
 * @param wasUserAction 标识是否是用户动作
 */
- (void)mapView:(MAMapView *)mapView mapWillZoomByUser:(BOOL)wasUserAction;

/**
 * @brief 在mapView对应的delegate方法内调用此方法

```

```

    @brief 在mapView对应的delegate方法内调用此方法
    * @param mapView 地图view
    * @param wasUserAction 标识是否是用户动作
    */
- (void)mapView:(MAMapView *)mapView mapDidZoomByUser:(BOOL)wasUserAction;

/**
    @brief 在mapView对应的delegate方法内调用此方法
    @param mapView 地图对象
    @param overlay overlay
    @return 对于属于当前controller的overlay，返回相应的MAOverlayRenderer对象，否则返回nil
    */
- (MAOverlayRenderer *)mapView:(MAMapView *)mapView rendererForOverlay:(id <MAOverlay>)overlay;

/**
    * @brief 在对应的地图delegate方法中调用此方法
    * @param mapView 地图
    * @param annotation 标注
    * @return 对于属于当前controller的annotation，返回相应的annotationview，否则返回nil
    */
- (MAAnnotationView *)mapView:(MAMapView *)mapView viewForAnnotation:(id <MAAnnotation>)annotation;

/**
    * @brief 在对应的地图delegate方法中调用此方法
    * @param mapView 地图View
    * @param userLocation 用户定位信息(包括位置与设备方向等数据)
    * @param updatingLocation 标示是否是location数据更新，YES:location数据更新 NO:heading数据更新
    */
- (void)mapView:(MAMapView *)mapView didUpdateUserLocation:(MAUserLocation *)userLocation updatingLocation:(BOOL)updatingLocation;

@end

```

MAPassengerRouteManagerDelegate

乘客端控制器代理

```

@protocol MAPassengerRouteManagerDelegate <NSObject>
@optional

///进入降级模式，采用本地路径规划
- (void)didDowngradeModeStarted:(MAPassengerRouteManager *)manager;

///结束降级模式，采用司乘同显路线
- (void)didDowngradeModeEnded:(MAPassengerRouteManager *)manager;

///发生请求错误时会回调
- (void)passengerRouteManger:(MAPassengerRouteManager *)manager requestFailedWithError:(NSError *)error;

///remainingDistance:剩余距离 estimatedTime:预计时间
- (void)routeStatusChangeForManager:(MAPassengerRouteManager *)manager
    remainingDistance:(double)remainingDistance
    estimatedTime:(NSTimeInterval)estimatedTime;

///当内部获取不到司机位置时，会调用该代理方法；
- (CLLocationCoordinate2D)extraDriverPositionForManager:(MAPassengerRouteManager *)manager;

@end

```

