

[SCA] Support offline mode for Driver App - Pickup, Return and On-hold modules tech solution

- Revision history
- Ownership
- Resources
- Summary
 - Background
 - Goal
 - Overall Design
 - 一、Pickup支持离线方案整体设计
 - 1.1 方案一、容灾方式(同现有delivery成功离线方式)
 - 1.2 方案二、缓存优先离线方式
 - 1.3 方案三、骑手操作全程离线，数据流在APP后台按照顺序上传同步
 - 1.4 离线预置数据同步
 - 1.4.1 接口合并
 - 1.4.2 缓存离线依赖数据数据同步
 - 二、Delivery、Return支持离线整体设计
- Detailed Design
 - 一、Pickup详细设计
 - 1.1 Task从Accepted扭转为Arrived状态
 - 1.2 扫描
 - 1.2.1 Task本地存储
 - 1.2.2 Task中order本地存储
 - 1.2.3 Order扫描APP本地校验
 - 1.3 order On-hold
 - 1.4 Task on hold
 - 1.5 Task HandOver
 - 1.6 Proof of On-hold和Proof of Pickup配置
 - 1.7 签名提交
 - 1.7.1 PickupStart Task扭转到PickupDone Task
 - 1.7.2 Accepted Task扭转到OnHold Task
 - 二、Delivery离线方案设计
 - 2.1 现状
 - 2.1.1 Delivery首页
 - 2.1.2 扫描
 - 2.1.3 Delivery成功/Delivery on-hold
 - 2.1.4 on-hold reason
 - 2.1.5 Proof of Delivery配置获取
 - 2.2 Delivery离线改动
 - 2.2.1 放开on-hold离线
 - 2.2.2 调整To-do页面缓存订单数量上限
 - 2.2.3 Delivery成功/On-hold时间调整
 - 2.2.4 支持多包裹离线(建议不支持离线)
 - 2.2.5 Proof of Delivery配置获取
 - 2.2.6 Delivery成功放开无需remark图片信息的派送流程
 - 三、Return离线方案
- Storage Design (Optional)
 - 一、新增APP本地存储离线数据表
 - 新增表1 pickup_offline_task_status_data
 - 新增表2 pickup_offline_scan_order_list_data
 - 新增表3 pickup_offline_order_onhold_data
 - 新增表4 pickup_offline_task_sign_data
 - 二、APP本地已有存储表
 - 1、Task列表(现有表)
 - 2、Order列表(现有表)
- Data track
- Monitoring
- Legacy & Risk Appraisal (Optional)
- Tickets

Revision history

Version	Revision date	Revisor	Revision content
v1	2022.08.03	Zhi.Xiao	初稿

Ownership

Product Manager	Mengke.Zhang
Project Manager	Qingchan.Li
Native Dev	Zhi.Xiao
RN Dev	NA
Server Dev	FM/LM: Xiaoyang.Huang, Xigun.Yan
Designer	Xiaojun.Gu、Wenchi.Qiu
QA	Xin.Hu, Zijian.Fang

Resources

PRD	[SEA + BR] Support offline mode for Driver App - Pickup, Return and On-hold modes
Figma	
Transify	https://transify.sea.com/resources/252
Git Repo(Optional)	https://git.garena.com/shopee/ssz-client/android/supplychain/fms-android
Dependent Service Doc (Optional)	NA
Project Schedule Page (Optional)	https://docs.google.com/spreadsheets/d/1BvhhbFLjncPxajJPkkgoeKHlo5nyyTe6X8a_ee1wMc/edit#gid=730435512
Feasibility Study Doc (Optional)	NA

Summary

Background

骑手在实际作业时，可能面临无网络或者网络较差情况，无离线模式带来了两个最大痛点：

- 1、影响骑手的操作效率
- 2、骑手只能在网络情况良好时再操作，不利于后续SOP的管控

而当前Driver App离线模式只支持了delivery成功的流程，没有支delivery on-hold、pickup和return；

Goal

支持delivery on-hold、pickup和return等场景的离线模式；

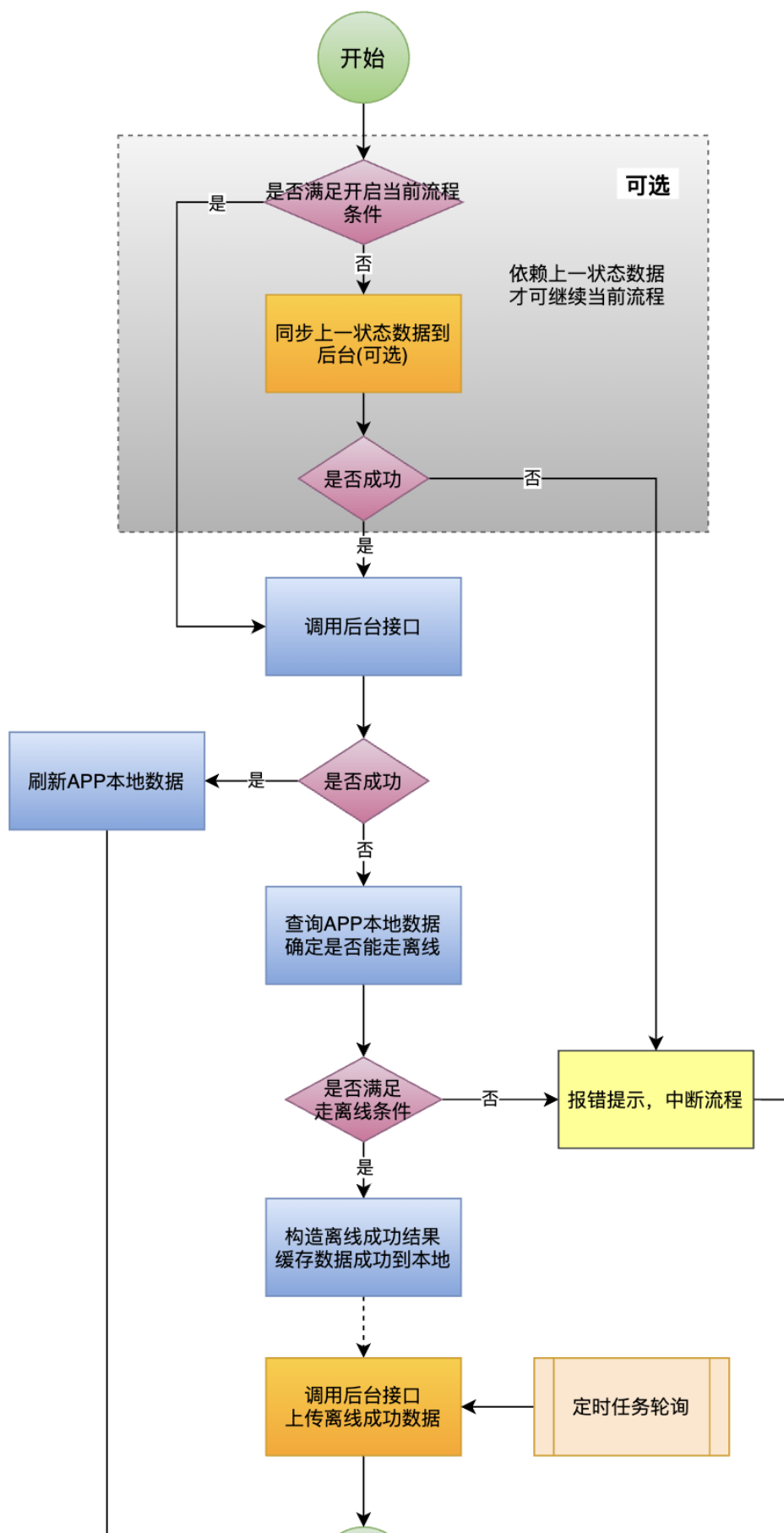
Overall Design

一、Pickup支持离线方案整体设计

Pickup离线涉及接口改造如下：

https://docs.google.com/spreadsheets/d/1-Dlc1D8y0SthBSOqUDdkoIR9NwXd8T3_iKD80eb_UR4/edit#gid=0

1.1 方案一、容灾方式(同现有delivery成功离线方式)

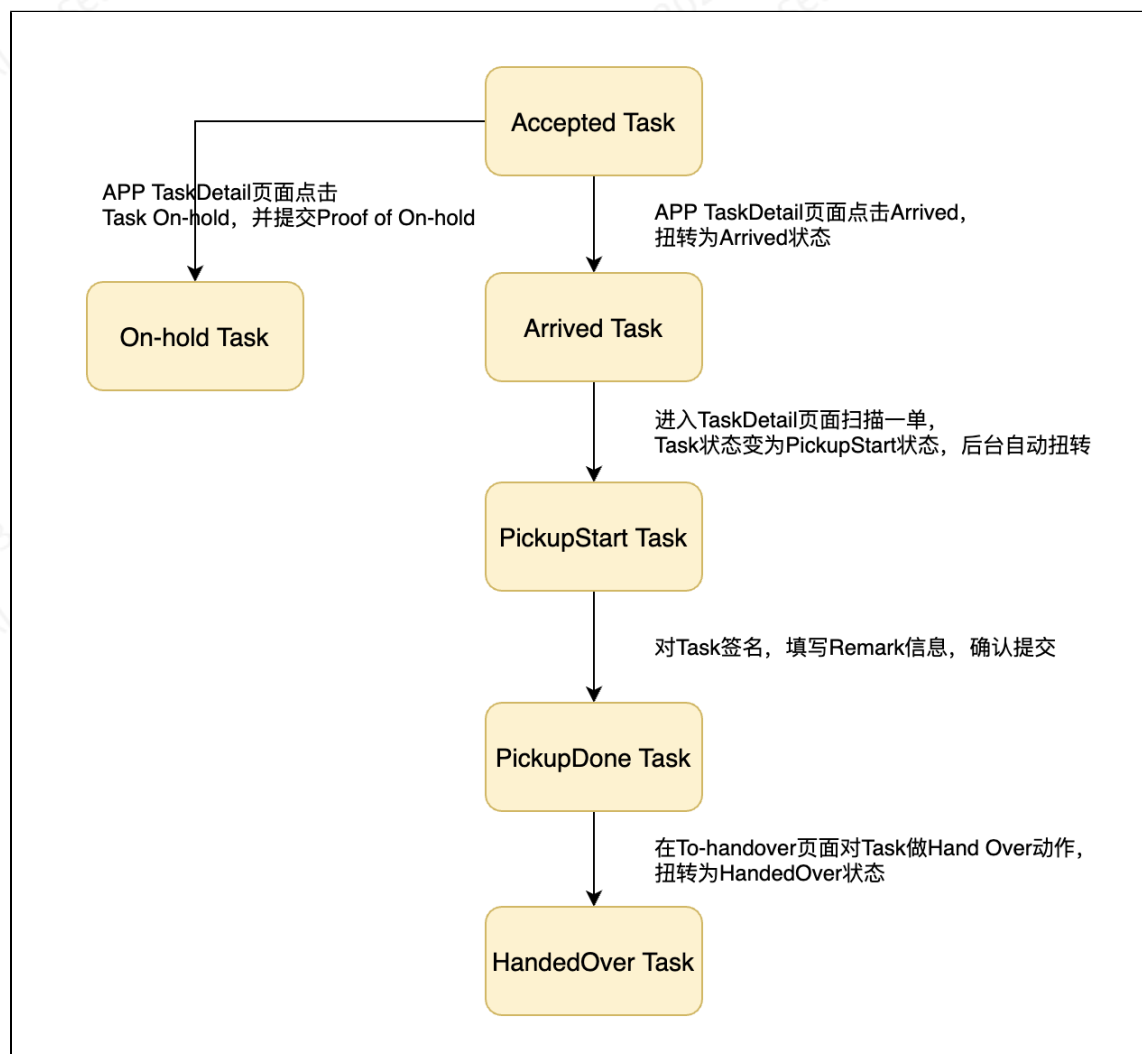




主要思路是先调用后端接口，如果成功，则走后台接口调用，同现有非离线流程；如果失败则走APP离线，依赖APP本地缓存数据；如果APP离线也失败，则报错，流程中断。

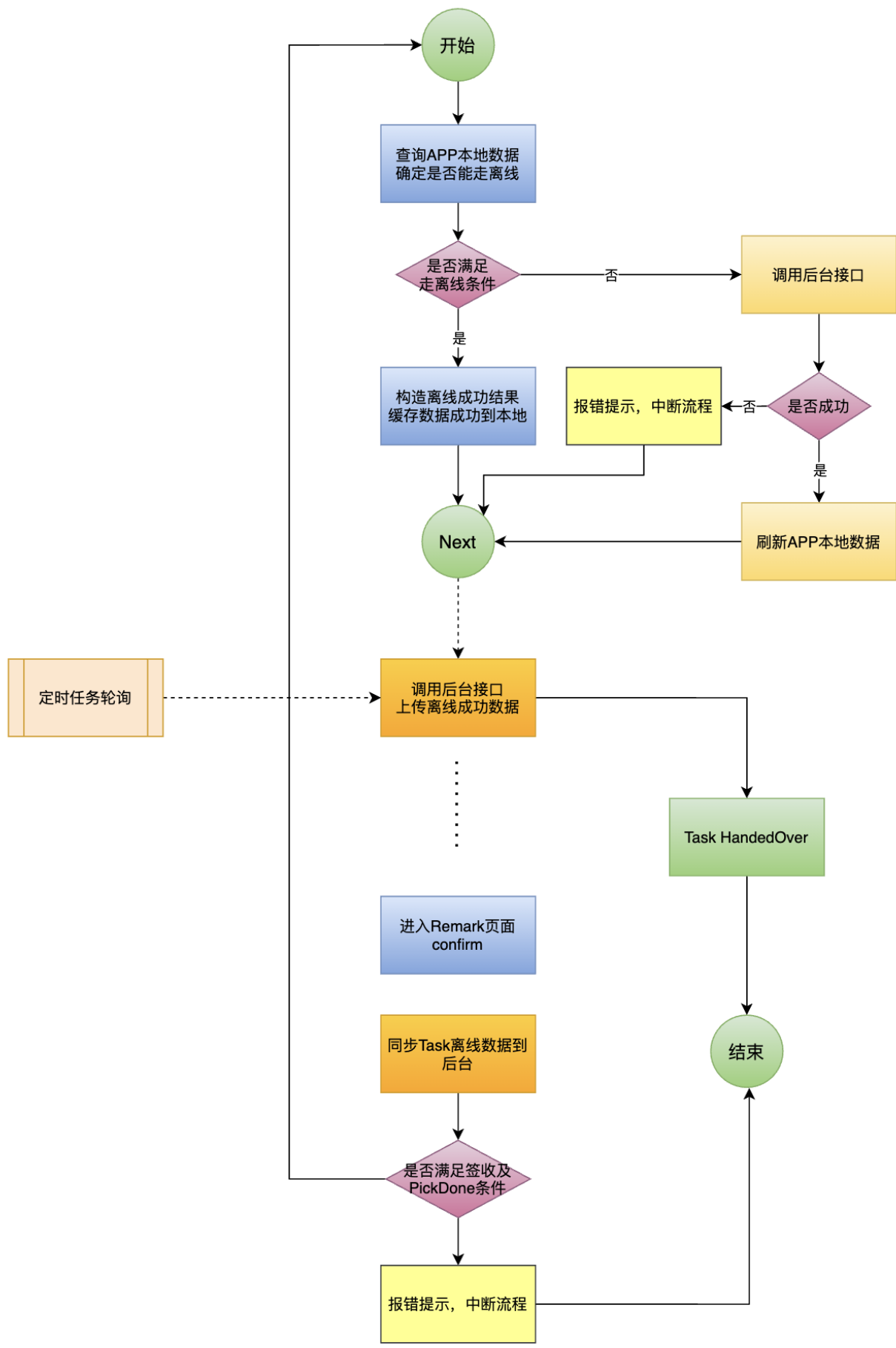
APP离线流程：构造数据成功结果，保存相关状态、时间等信息在本地，置数据为待上传状态，待网络就绪后(周期任务，使用WorkManager定时轮询是否可上传同步数据)，执行原本调用后台接口完成数据同步，置数据为已上传，删除本地成功数据

Driver App操作Task状态扭转流程如下，参考[pickup order/task 状态扭转图](#)



容灾方式离线保证网络OK下走后台，和现有流程一致，只有在网络故障时才考虑走离线，此方案大多数情况下走后台，能保证APP和后台数据一致性，不容易出问题。

1.2 方案二、缓存优先离线方式



缓存优先离线方式，在支持离线场景每个需要调用后台接口时，优先查询本地缓存数据是否满足走离线条件，如果满足构造成功结果，继续执行后续流程；如果不满足走离线，则调用后台接口走后台，同现有流程；

离线数据的上传在APP后台自动完成，在整个揽件流程末端，进入到Remark页面时，规定必须预先经过离线数据同步，以保证Task数据可正常完成签名和Task PickupDone动作，否则报错中断流程。

签名提交前依赖Task校验，APP在与后台同步数据后再执行提交流程；

此方案大多数流程都是依赖离线数据，优先走离线，对骑手来说，在揽件前前面流程比较顺畅，没有阻塞；在remark页面统一强制同步离线数据(这里可能在前端流程已经完成了同步，这里起到最后一关校验)，

如果满足签收及PickupDone条件，则可以继续签收(这个动作可以走离线)；否则无法签收，报错提示中断流程。

无论采用哪一种方案，Pickup后台都需要处理不同时机APP同步的离线数据，这些数据可能不是按照骑手操作时间先后到达；因此，这里从Task状态扭转流程出发，设定签名提交流程为关键步骤，并且由于

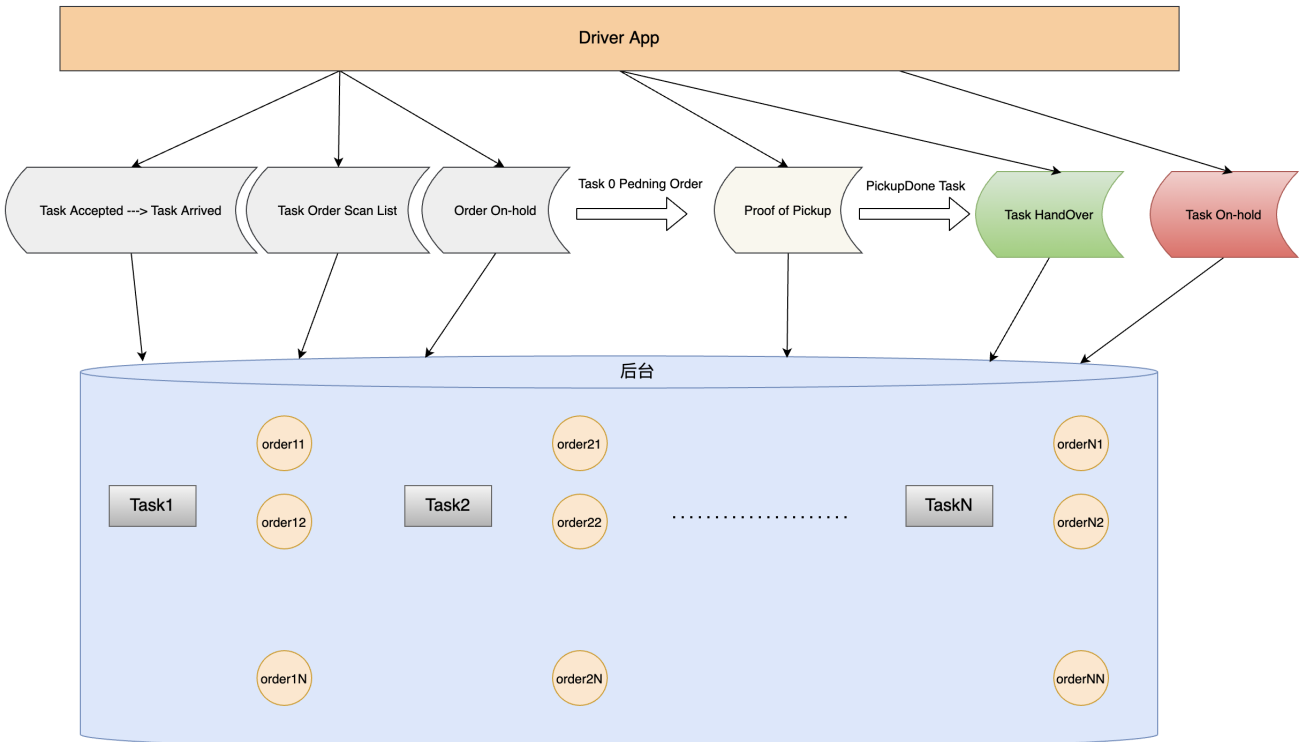
Task签名提交从PickupStart状态扭转为PickupDone状态，要满足Task中Pending Orders数量为0，否则无视这些校准要求执行后续Task状态扭转，可能带来一系列问题（Pickup后台评估 xiaoyang.huang@shopee.com）

问题1：Task HandOver调用接口离线数据处于失败，接口重试，实际是Task状态不对，后台返回Task状态不对，删除本地Task离线数据？

问题2：签名提交接口失败，接口重试，实际是Task有待Pickup订单，后台返回Task存在待Pickup订单，无法提交扭转状态，删除签名提交离线数据？

问题3：如果是使用容灾方式，这里签名提交先调用后台接口，如果前面步骤离线数据未同步(并非有后台校验失败订单)，这里也无法继续提交，如何处理？

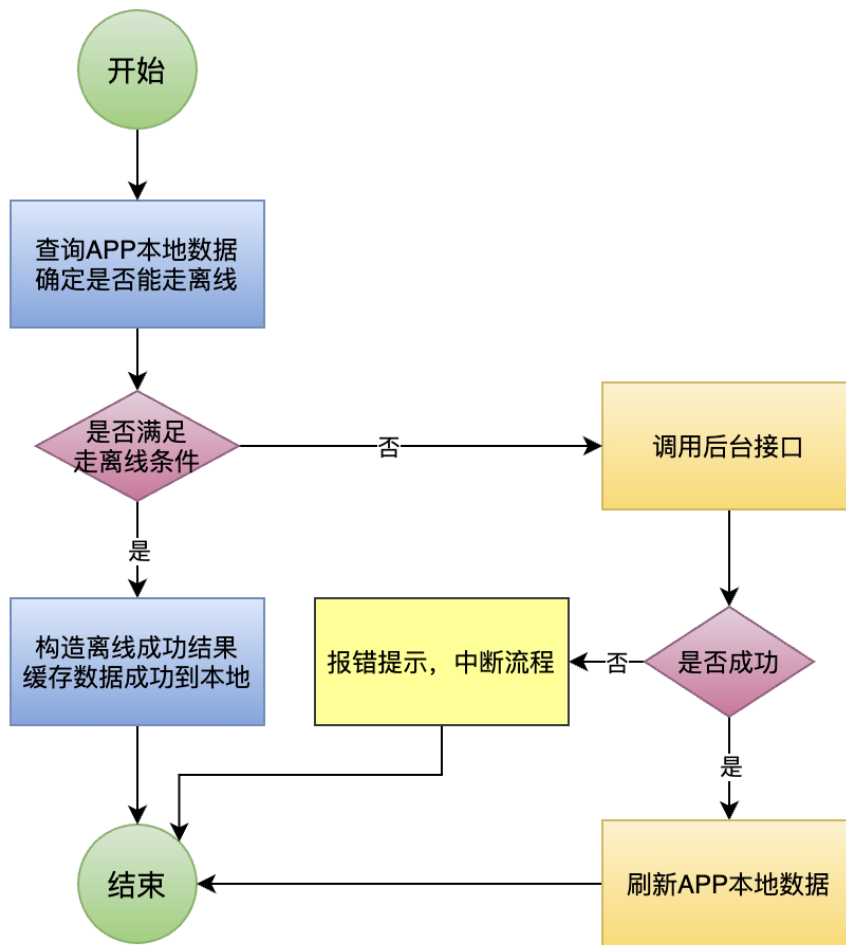
如果是使用缓存离线优先方式，直接构造签名提交成功结果，在离线数据上传到后台后存在校验失败订单或者Task状态还停留在Accepted，如何处理？



方案一和方案二步骤依赖见方案三说明

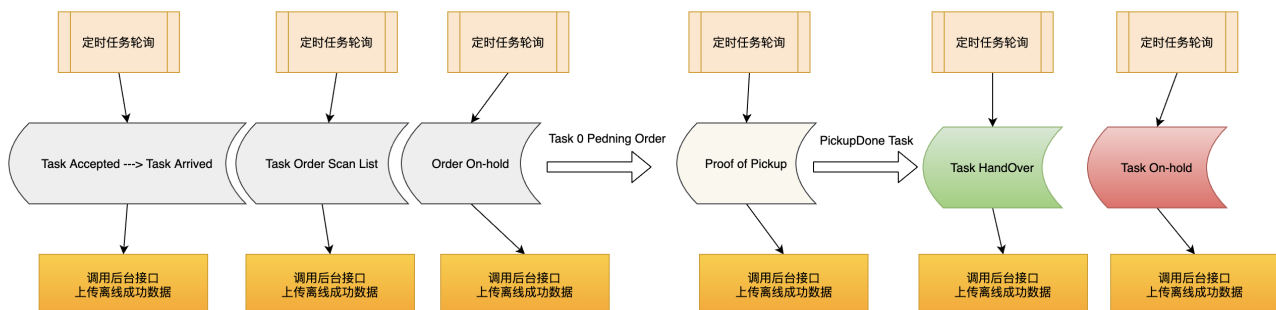
1.3 方案三、骑手操作全程离线，数据流在APP后台按照顺序上传同步

每个步骤处理流程如下所示，和方案二一致，只是这里不再签名提交步骤强制同步数据；



每个步骤离线后上传离线成功数据都遵循Task和Order实际数据流转流程上传，APP通过多个定时任务轮询离线数据是否满足上传后台条件，满足则调用后台接口上传同步数据，否则等待前置步骤上传成功，如果前置上传失败超过重试次数，则清除后置

步骤离线数据，恢复最新离线成功状态；



APP本地离线数据定时轮询同步到后台依赖说明：

(1) 上传Task本地为Arrived Status数据，理论上前面没有依赖离线数据上传，简单判断Task后台更新状态是否为Accepted状态，上传成功则更新APP存储Task状态；

失败则重试(限定三次)，三次都失败则删除Task离线数据(后续骑手操作离线数据都无效，APP本地清除)

(2) 上传扫描订单列表到后台校验, 依赖Task状态为Arrived Status, 如果Task后台更新状态还未到Arrived状态, 则不调用后台接口上传, 上传成功则更新APP存储Task和Order状态;

校验成功订单, 更新Order存储后台状态, 校验失败本地存储后台Order状态仍为非Picked Up;

(3) 上传On-hold订单列表到后台, 依赖Task状态为Arrived Status, 如果Task后台更新状态还未到Arrived状态, 则不调用后台接口上传, 上传成功则更新Order存储Task状态;

校验成功订单, 更新Order存储后台状态, 校验失败本地存储后台Order状态仍为非原本状态;

(4) 上传签名提交离线数据到后台, 依赖Task状态为PickupStart, 如果Task后台更新状态还未到Start状态, 则不调用后台接口上传, 上传成功则更新APP存储Task状态;

失败则重试(限定三次), 三次都失败则删除Task离线数据(后续骑手操作离线数据都无效, APP本地清除);

(5) 上传Task On-hold离线数据到后台, 理论上前面没有依赖离线数据上传, 简单判断Task后台更新状态是否为Accepted状态, 上传成功则更新APP存储Task状态;

失败则重试(限定三次), 三次都失败则删除本步骤Task离线数据;

(6) 上传HandOver离线数据到后台, 依赖Task状态为PickupDone, 如果Task后台更新状态还未到PickupDone状态, 则不调用后台接口上传, 上传成功则更新APP存储Task状态;

失败则重试(限定三次), 三次都失败则删除本步骤Task离线数据;

1.4 离线预置数据同步

1.4.1 接口合并

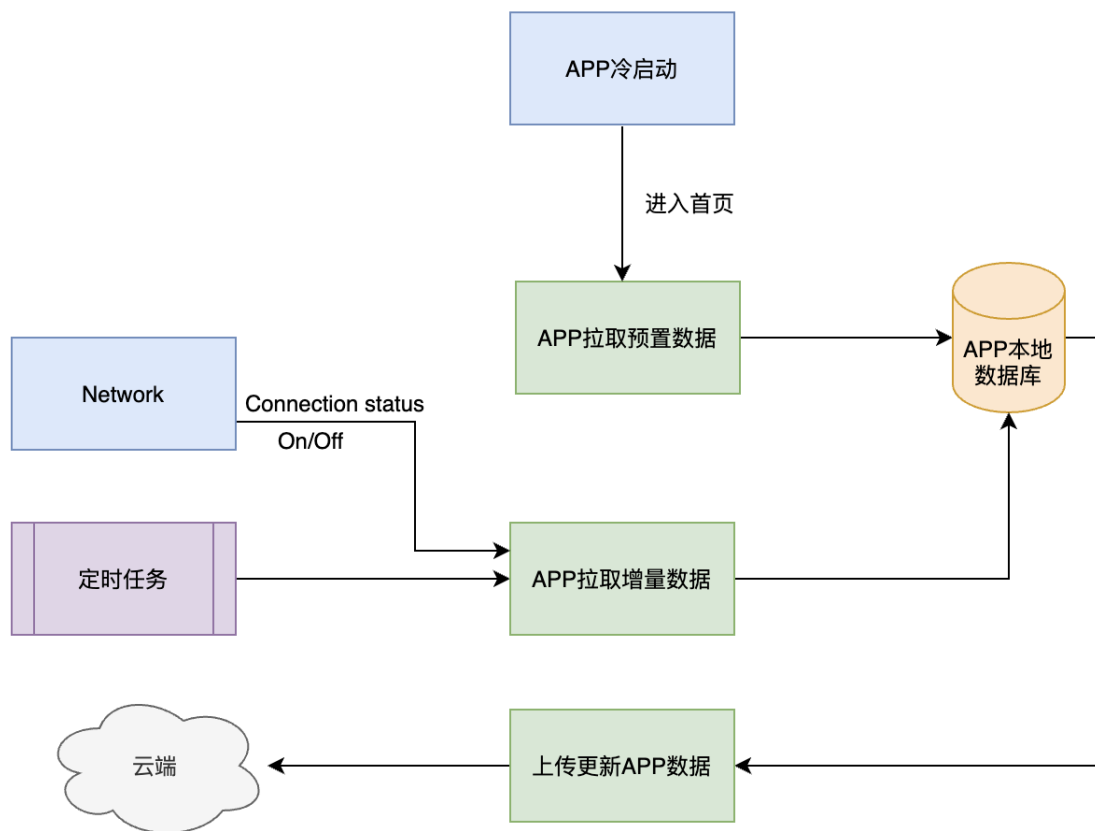
Pickup主页和详情页接口合并

首页接口: api/v4/pickup/pickup_task/search

详情页接口: api/v4/pickup/pickup_task/detail

1.4.2 缓存离线依赖数据数据同步

离线预置数据包含: Task列表、Task详情、Shop列表、Order列表



同步任务采用单任务方式（即同时只有一个同步任务在执行）减少多线程带来的并发问题维护成本，数据的一致性得到保证。

基本的同步过程：

1. 移动端读取本地最新同步时间，根据时间请求服务端
2. 服务端下发增量数据
3. 客户端对比本地时间和增量数据的时间，判断是否已拉取最新数据
4. 如果本地已是最新数据，停止同步；否则继续增量拉取数据，重复步骤1

二、Delivery、Return支持离线整体设计

离线主要思路：

- 2.1 Delivery在已有离线基础上使用容灾方式，放开On-hold离线
- 2.2 Proof of Delivery配置和On-hold Reason配置可以改用缓存优先
- 2.3 Return离线参考Delivery离线做法

Detailed Design

一、Pickup详细设计

1.1 Task从Accepted扭转为Arrived状态

走APP离线，增加Task状态效应，只有在Task状态为TASK_STATUS_ACCEPTED时才满足离线条件；构造离线成功数据，需将Task本地状态置为Arrived Status；

接口：api/v3/pickup/pickup_task/batch_update/ POST请求

上传数据结构包含字段：

字段名	字段类型	含义
pickup_task_id_list	List<String>	task列表，这里为单个taskId
action_type	int	扭转目标状态type，这里为Arrived Status 值为8
lat	double	纬度
lng	double	经度
arrived_operate_time(新增)	long	骑手在APP操作Task扭转到Arrived时间戳（单位：秒）

APP本地存储离线数据结构

字段名	字段类型	含义
pickup_task_id_list	List<String>	task列表，这里为单个taskId
action_type	int	扭转目标状态type，这里为Arrived Status 值为8
lat	double	纬度
lng	double	经度
arrived_operate_time	long	骑手在APP操作Task扭转到Arrived时间戳（单位：秒）
upload_status	int	上传状态，用于标示离线数据上传状态，pending：0、success：1、freeze：2
driver_id	String	骑手ID

1.2 扫描

现状：

1.2.1 Task本地存储

在To-pickup页面缓存，同时每间隔3s拉取Task Detail和shop信息，一个task最多拉取3次

Task及Detail本地数据每隔一天删除，shop信息数据半天删除

1.2.2 Task中order本地存储

在TaskDetail页面拉取task中三种状态（ORDER_TYPE_PENDING、ORDER_TYPE_PICKED_UP_DROP_OFF、ORDER_TYPE_PICKUP_ON_HOLD）的订单，每种订单状态最多1000i订单

order信息本地数据最多保存7天、在签名提交成功后清楚已提交的Order缓存

1.2.3 Order扫描APP本地校验

(1) 扫描结果在APP本地有对应订单信息

(2) Order状态和本地设置状态不为ORDER_STATUS_PICKED_UP

(3) Order为shopee单，且是非DropOff单

(4) Order状态为以下几种：

ORDER_STATUS_WAVING、ORDER_STATUS_ALLOCATED、ORDER_STATUS_PICKUP_RETRY、ORDER_STATUS_PENDING_FAILED、ORDER_STATUS_ON_HOLD

APP扫描订单离线依赖扫描结果在本地是否有对应订单信息，因此，对于首次无网络进入TaskDetail页是无法对扫描离线处理；

[\[SCA\] Pickup 离线扫描设计文档](#) [\[SCA\] Offline mode for Driver App - Detailed design of pickup/delivery cache priority pre pull](#)

后台扫描订单校验逻辑：

- (1) 判断扫描结果是否为合法订单
- (2) 判断订单是否属于当前骑手(可能是当前Task或者其他Task中的订单)
- (3) 判断订单状态是否可扭转为Picked Up

APP扫描改动(实时扫描)，下线批扫模式？

方案一：Finish Scanning时直接跳转到Submit Result页面，由于离线校验在扫描时APP本地已经校验，这里submit只是个动作，没有实际逻辑，Submit Result结果页预期都是APP离线扫描成功结果，这里考虑界面是否给予离线结果提示？

待网络就绪后，在上传离线订单列表到后台校验，校验结果(后台校验成功订单列表和失败列表，失败原因等)如果给用户展示或者提示？

方案二：Finish Scanning时优先走后台接口效应，和现有流程一致，如果无网络，则直接跳过进入Submit Result结果页，展示离线成功结果页，同时提醒骑手打开网络，这块需要PD看看如何设计

这里如果允许提交给后台扫描校验这个动作无感知上传，在后台校验完成结果应该以何种方式告知骑手，弹对话框展示校验结果还是怎么如何展示？ Wench i Qiu

提交后台校验接口：api/v3/pickup/pickup_order/pickdup

字段名	字段类型	含义
pickup_task_id	String	taskId
pickup_order_id_list	List<String>	提交待效应order列表
pickup_order_list	List<PickupCalculateFeeRequestBean>	带有计算称重运费的order列表，其中包含扫描时间
lat	double	纬度
lng	double	经度
support_shopeepay	int	支持ShopeePay Pickup 收款功能，TH市场特有

PickupCalculateFeeRequestBean数据结构字段定义

字段名	字段类型	含义
pickup_order_id	String	orderId
sender_weight	Double	下单重量尺寸信息-重量
sender_length	Integer	下单重量尺寸信息-长度
sender_width	Integer	下单重量尺寸信息-宽度
sender_height	Integer	下单重量尺寸信息-高度
driver_weight	Double	driver 输入的重量尺寸信息-重量
driver_length	Integer	driver 输入的重量尺寸信息-长度
driver_width	Integer	driver 输入的重量尺寸信息-宽度
driver_height	Integer	driver 输入的重量尺寸信息-高度
shipping_fee	Double	运费
scan_time	Long	扫描时间（后台获取订单扫描时间字段）

APP本地存储离线扫描待提交效应数据结构定义

字段名	字段类型	含义
pickup_task_id	String	taskId
pickup_order_id_list	List<String>	提交待效应order列表
lat	double	纬度
lng	double	经度
support_shopeepay	int	支持ShopeePay Pickup 收款功能, TH市场特有
scan_operate_time	long	骑手APP操作扫描订单时间(单位: 秒)
upload_status	int	上传状态, 用于标示离线数据上传状态, pending: 0、success: 1、freeze: 2
driver_id	String	骑手ID

1.3 order On-hold

方案一：容灾方式，后台接口不可用时走APP离线，APP本地缓存order on-hold数据，本地扭转订单状态为on-hold后，需刷新本地存储order数据为on-hold状态，待网络就绪后上传同步离线数据；

方案二：优先走APP离线，调用后台接口on-hold动作对用户无感知；

APP增加本地校验：对于某些状态下的订单才可on-hold (ORDER_STATUS_WAVING、ORDER_STATUS_ALLOCATED、ORDER_STATUS_PICKUP_RETRY、ORDER_STATUS_PENDING_FAILED)

接口：api/v3/pickup/pickup_order/onhold/

上传后台数据结构定义

字段名	字段类型	含义
pickup_task_id	String	taskId
pickup_order_id_list	List<String>	正选使用 orderIdList
ignore_pickup_order_id_list	List<String>	反选使用 ignoreOrderIdList
on_hold_reason	int	on-hold原因
lat	double	纬度
lng	double	经度
on_hold_operate_time(新增)	long	骑手在APP对订单on-hold操作时间, 单位: 秒

APP本地存储on-hold离线数据结构定义

字段名	字段类型	含义
pickup_task_id	String	taskId
pickup_order_id_list	List<String>	正选使用 orderIdList
ignore_pickup_order_id_list	List<String>	反选使用 ignoreOrderIdList
on_hold_reason	int	on-hold原因
lat	double	纬度
lng	double	经度
on_hold_operate_time	long	骑手在APP对订单on-hold操作时间, 单位: 秒

upload_status	int	上传状态，用于标示离线数据上传状态，pending: 0、success: 1、freeze: 2
driver_id	String	骑手ID

1.4 Task on hold

此处不是对Task状态做扭转，最终扭转Task状态为on-hold是在签名提交流程，这里涉及接口是获取on-hold reason配置；目前是第一次Task On-hold直接从后台拉取配置，后面从内存读取配置(与APP进程存在关联)

离线改进方式：APP进入To-Pickup主页时拉取该配置，存储在本地数据(现有流程支持)，每次进pickup主页都刷新该配置；在Task On-hold时优先从本地数据库读取配置，读取不到则从后台获取；

接口：/api/on_hold_reason/search type为9

本地存储数据表：on_onhold_reason

1.5 Task HandOver

走APP离线，增加Task状态效应，只有在Task状态为TASK_STATUS_PICKUP_DONE时才满足离线条件；构造离线成功数据，需将Task本地状态置为Handed Over Status；

接口：api/v3/pickup/pickup_task/batch_update/ POST请求

上传数据结构包含字段：

字段名	字段类型	含义
pickup_task_id_list	List<String>	task列表，这里为单个taskId
action_type	int	扭转目标状态type，这里为Handed Over 值为10
lat	double	纬度
lng	double	经度
arrived_operate_time(新增)	long	骑手在APP操作Task扭转到Handed Over时间戳（单位：秒）

APP本地存储离线数据结构

字段名	字段类型	含义
pickup_task_id_list	List<String>	task列表，这里为单个taskId
action_type	int	扭转目标状态type，这里为Handed Over 值为10
lat	double	纬度
lng	double	经度
arrived_operate_time	long	骑手在APP操作Task扭转到Handed Over时间戳（单位：秒）
upload_status	int	上传状态，用于标示离线数据上传状态，pending: 0、success: 1、freeze: 2
driver_id	String	骑手ID

1.6 Proof of On-hold和Proof of Pickup配置

对Task On-hold配置是Proof of On-hold，对应On-hold reason配置，在上面1.4说明了获取流程；

对order On-hold对应Proof of Pickup配置，接口：/api/order_settings/proof_rule/list type: 2, entity_type:4 表名：proof_rule

现状：目前Proof of Pickup配置在进入Pickup主页获取，在进入Proof of Pickup时默认读取本地存储配置，如果读取不到，则给定默认配置

这里如果本地没有配置，是否需要再从后台拉取一次配置？

1.7 签名提交

方案一：容灾方式

Task去做签名提交动作，必须满足Task中没有Pending Order才可能状态扭转到PickupDone；因此，这里无法仅依靠APP本地Task Pending数量判断

(1) 本地校验通过扭转order状态为Picked Up，实际到后台可能校验失败

(2) On-hold失败，Task仍存在Pending订单

如果这里做成不阻塞骑手签名提交流程，后台怎么处理这些冲突数据？ xiaoyang.huang@shopee.com

需保证该Task相关离线数据全部同步到后台才能执行下一步，否则提示骑手网络异常；

Task相关离线数据包含本章节1.2和1.3数据，在对订单校验与后台同步后再执行签名提交流程；

1.7.1 PickupStart Task扭转到PickupDone Task

现状：

(1) 调用接口 /api/batch/image/upload/ 上传seller和driver签名 、调用 /api/batch/image/upload/ 上传remark第一张图片

(2) 调用 api/v3/pickup/pickup_task/sign 完成task状态扭转，创建凭证

(3) 异步动作：如果remark只有一张图片，则调用attachment/updateattachment/update更新凭证并推送epop给SLS；如果remark存在超过1张图片，上传其余1张或者2张图片，成功后调用 attachment/update更新凭证并推送epop给SLS

离线方案：参考Delivery离线容灾方式

(1) 调用api/v3/pickup/pickup_task/sign接口，接口要携带seller、driver签名 xiaoyang.huang@shopee.com 否则构造离线成功数据

(2) 异步上传remark三张图片，第一张图片不受网络设置开关影响，其他2张跟随WLAN设置

APP本地构造离线成功数据结构定义

字段名	字段类型	含义
pickup_task_id_list	List<String>	task列表
seller_signature_path	String	卖家签名图片url
driver_signature_path	String	骑手签名图片url
proof_rule_id	int	配置Proof RuleId
rule_update_time	int	Rule配置修改更新时间
remark	String	remark备注信息
photo_path_list	List<String>	remark页面上传额外图片url列表
signatory	String	署名
task_done_operate_time	long	骑手在APP扭转Task状态操作时间，单位：秒
upload_status	int	上传状态，用于标示离线数据上传状态，pending：0、success：1、freeze：2
driver_id	String	骑手ID

1.7.2 Accepted Task扭转到OnHold Task

现状：

(1) 调用接口 /api/batch/image/upload/ 上传remark三张图片

(2) 调用api/v3/pickup/pickup_task/batch_update/ 完成task状态扭转

离线方案：参考Delivery离线容灾方式

(1) 调用api/v3/pickup/pickup_task/batch_update/ 接口，网络异常则构造离线成功数据，走APP离线

(2) 异步上传remark三张图片，第一张图片不受网络设置开关影响，其他2张跟随WLAN设置

(3) 如何更新凭证？

APP本地存储离线数据结构

字段名	字段类型	含义
pickup_task_id_list	List<String>	task列表，这里为单个taskId
action_type	int	扭转目标状态type，这里为On hold Status 值为15
lat	double	纬度
lng	double	经度
onhold_operate_time	long	骑手在APP操作Task扭转到On-hold时间戳（单位：秒）
upload_status	int	上传状态，用于标示离线数据上传状态，pending：0、success：1、freeze：2
driver_id	String	骑手ID

二、Delivery离线方案设计

在现有Delivery成功场景的离线流程上继续支持更多场景的离线

2.1 现状

2.1.1 Delivery首页

To-do：根据页面总单数，不足20单，缓存全部订单数；超过20单，缓存订单数量为：最低20单，最多50单；

Delivered：最多缓存10单

On-Hold：最多缓存10单

2.1.2 扫描

本地缓存了对应订单，扫描时可走离线，支持离线扫描，否则报错提示；

2.1.3 Delivery成功/Delivery on-hold

Proof of on-hold、Proof of Delivery支持离线，本地缓存了该订单，构造成功数据结果，待网络就绪后同步上传给后台，更新状态及凭证；

但是，APP目前on-hold在remark页允许离线是关闭，即不支持走离线，代码流程实际是支持的；

2.1.4 on-hold reason

在Driver进入首页拉取on-hold原因配置，保存这个配置数据到本地数据库，在对订单on-hold时先拉取on-hold配置，如果成功则继续否则从本地数据获取配置，否则从本地数据库拿；

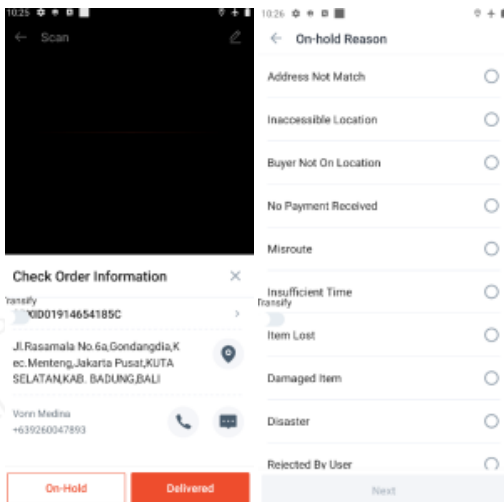
2.1.5 Proof of Delivery配置获取

在Driver进入首页拉取，在进入SelectRecipient页面时读取ProofRuleItem配置，这里是直接从内存读取，不是从数据库读取，依赖首页拉取配置，如果首页拉取失败，在扫描订单进入SelectRecipient页面时无法拉取ProofRule配置

2.2 Delivery离线改动

2.2.1 放开on-hold离线

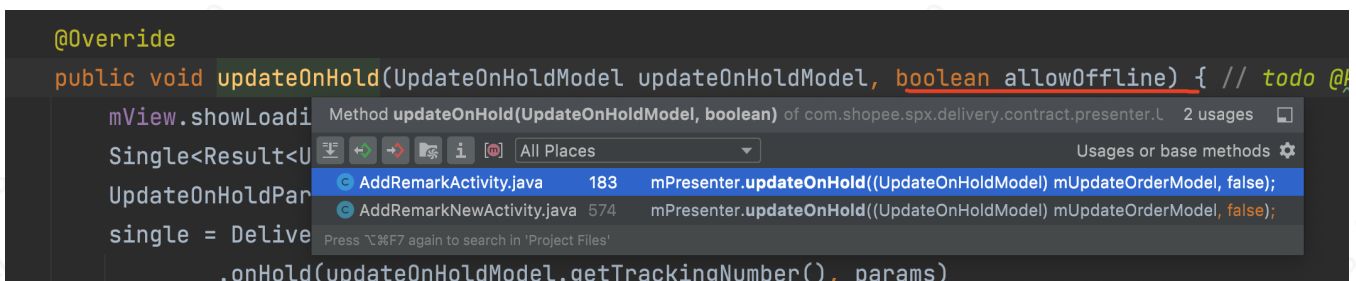
(1) 离线扫描订单on-hold放开，当前on-hold无法跳转到On-hold Reason页面



(2) on-hold读取配置使用缓存优先策略

由于on-hold reason配置更改频率较低，采用缓存优先方案，目前在Driver进入首页拉取on-hold原因配置，保存这个配置数据到本地；在对订单on-hold时优先从缓存拿，如果拿不到，再从后台拉取；

(3) 对Proof of on-hold离线放开



2.2.2 调整To-do页面缓存订单数量上限

根据PM要求最多支持100单，基于22Q2 Delivery离线基础上调整缓存todo订单数量上限

2.2.3 Delivery成功/On-hold时间调整

派送接口：/delivery/api/v1/delivery_order/deliver

增加字段：APP派送时间戳

On-hold接口：/delivery/api/v1/delivery_order/on_hold

增加字段：APP on-hold时间戳

xigun.yan@shopee.com

2.2.4 支持多包裹离线(建议不支持离线)

容灾方式支持离线(同扫描离线方式)

先请求后台接口，如果失败则走APP离线；APP从本地缓存订单列表查询是否有相同收件信息的订单，如果有则走多包裹流程，Delivered/On-hold都需改动；

相同收件信息判断：receiver_name、receiver_address、receiver_phone

2.2.5 Proof of Delivery配置获取

与on-hold reason配置一样，在Driver进入首页拉取ProofRule配置，保存这个配置数据到本地，在扫描订单进入SelectRecipient页面时优先从本地数据库读取，获取不到再从后台拉取，都拉取不到，则报错提示中断操作；

2.2.6 Delivery成功放开无需remark图片信息的派送流程

```

        case RecipientModel.STATUS_DONE:
            mRecipientModel = recipientModel;
            mPresenter.updateSignature(recipientModel, allowOffline: false);
            break;
        default:
    
```

三、Return离线方案

同Delivery改造方案

Storage Design (Optional)

一、新增APP本地存储离线数据表

新增4个离线数据表，如果上传成功则删除，否则保存3天有效期，重试次数沿用已有1000次（参数可配置？）

新增表1 pickup_offline_task_status_data

表名: pickup_offline_task_status_data 含义: Task离线状态数据表, Task Arrived、Task On-hold、Task HandOver时使用

字段名	字段类型	含义
pickup_task_id	String	taskId 作为表主键
action_type	int	扭转目标状态type
lat	double	纬度
lng	double	经度
arrived_operate_time	long	骑手在APP操作Task扭转到Arrived时间戳（单位：秒）
upload_status	int	上传状态, 用于标示离线数据上传状态, pending: 0、success: 1、freeze: 2
driver_id	String	骑手ID
failCount	int	重试次数

新增表2 pickup_offline_scan_order_list_data

表名: pickup_offline_scan_order_list_data 含义: Task中扫描order list离线数据表

字段名	字段类型	含义
id	long	表主键, 自动递增
pickup_task_id	String	taskId
pickup_order_id_list	List<String>	提交待效应order列表
lat	double	纬度
lng	double	经度
support_shopeepay	int	支持ShopeePay Pickup 收款功能, TH市场特有
scan_operate_time	long	骑手APP操作扫描订单时间(单位: 秒)
upload_status	int	上传状态, 用于标示离线数据上传状态, pending: 0、success: 1、freeze: 2

driver_id	String	骑手ID
failCount	int	重试次数

新增表3 pickup_offline_order_onhold_data

表名: pickup_offline_order_onhold_data 含义: order on-hold时本地离线成功数据

字段名	字段类型	含义
id	long	表主键, 自动递增
pickup_task_id	String	taskId
pickup_order_id_list	List<String>	正选使用 orderIdList
ignore_pickup_order_id_list	List<String>	反选使用 ignoreOrderIdList
on_hold_reason	int	on-hold原因
lat	double	纬度
lng	double	经度
on_hold_operate_time	long	骑手在APP对订单on-hold操作时间, 单位: 秒
upload_status	int	上传状态, 用于标示离线数据上传状态, pending: 0、success: 1、freeze: 2
driver_id	String	骑手ID
failCount	int	重试次数

新增表4 pickup_offline_task_sign_data

表名: pickup_offline_task_sign_data 含义: Pickup of Proof签名提交离线数据

字段名	字段类型	含义
pickup_task_id	String	taskId, 表主键
seller_signature_path	String	卖家签名图片url
driver_signature_path	String	骑手签名图片url
proof_rule_id	int	配置Proof RuleId
rule_update_time	int	Rule配置修改更新时间
remark	String	remark备注信息
photo_path_list	List<String>	remark页面上额外图片url列表
signatory	String	署名
task_done_operate_time	long	骑手在APP扭转Task状态操作时间, 单位: 秒
upload_status	int	上传状态, 用于标示离线数据上传状态, pending: 0、success: 1、freeze: 2
driver_id	String	骑手ID
failCount	int	重试次数

二、APP本地已有存储表

1、Task列表(现有表)

task缓存数量限制15个 APP本地保存一天清除

表名: pickup_task_cache_data

字段	字段类型	含义	备注
pickup_task_id	String	TaskId, 表主键	
pickup_point_name	String	pup名称	
quick_pickup_task_id	String	orderId	可不存储
status	int	task状态	
task_local_status	int	task本地状态	新增字段
over_time	long	drop off订单标识	
pickup_task_label	int	Pickup Task/Shop flag	
pending_order_quantity	int	pending order	
pickedup_order_quantity	int	v3/v4 v4 To-handover tab drop off order	
dropoff_order_quantity	int	picked up drop off	
pickup_complete_order_quantity	int	v2接口返回, pending order数量	
handover_onhold_order_quantity	int	Handover Onhold	
rejected_time	long	task rejected时间	
disabled_time	long	task disabled 时间	
shop_id	String	shop信息id	
seller_name	String	卖家名称	
seller_address	String	卖家地址	
seller_contact	String		
handedover_dropoff_order_quantity	int	v3/v4	
inbounded_dropoff_order_quantity	int	HandedOver Inbounded	
total_cash	double		
total_settlement	double		
pending_weight	double	pending重量	
pending_volume	double	pending体积	
pickup_weight	double	重量	
pickup_volume	double	总体积	
task_version	int	v0 shopv3	
pickup_point_address	String	pup地址	

customer_quantity	int	shop数量, v3接口没有该值	
total_order_quantity	int	task中总订单数量	
total_paid_asf_amount	double	totalPaidAsfAmount	
task_visible_time	long	task主页面task item可见最新时间	
on_hold_order_quantity	int	on-hold order数量	
pickedup_dropoff_order_quantity	int	picked up	
handover_order_onhold_quantity	int	以order维度进行提交handover on-hold order数量	
handover_order_quantity	int	orderhandover order	
pickup_task_quantity	int	task数量, normal pickup这里为1	
pickup_task_detail_cache_time	long	task 数据最新缓存时间	
driver_id	int	骑手Id	
task_detail_data_flag	boolean	是否包含Task Detail数据	
task_shop_data_flag	boolean	是否包含Task shop数据	
failCount	int	拉取Task Detail和shop信息失败次数, 最多3次	
to_data	PickupShopBean.ToInfo	TO信息	
asf_info_list	List<PickupShopBean.AsfInfo>	ASF列表信息	

2、Order列表（现有表）

Order数量限制1000个(这个是指1个task order数量限制1000还是全部order数量限制1000?)

表名: pickup_order

数据字段定义:

字段	字段类型	含义
id	long	表主键, 自动递增
pickup_order_id	String	orderId
status	int	订单状态
isDropOff	int	drop off订单标识
isFromShopee	boolean	shopee订单标识
third_party_tracking_num	String	spxi订单对应三方订单号
pointLineFlag	int	点线flag
sls_tracking_number	String	spxi订单对应sls订单号
driver_pickedup_time	long	后端/本地记录的 pickup 时间, 单位 s
shop_id	String	订单的shop信息标识id
pickup_order_local_status	int	本地修改的Pickup的状态
pickup_task_id	String	本地设置的taskId

pickup_task_cache_time	long	用于记录获取task 缓存刷新的时间，单位 ms
------------------------	------	--------------------------

Data track

1、对于APP本次涉及离线接口，骑手在操作时走了APP离线增加记录上报到MDAP

pickup/delivery/return设计离线接口：https://docs.google.com/spreadsheets/d/1-Dlc1D8y0StHBSOqUDdKoIR9NwXd8T3_iKD80eb_UR4/edit#gid=0

2、针对骑手每日在线时的离线时长记录埋点，统计上报到MDAP

3、对于在签名提交前同步离线数据校验失败情况统计原因

4、每个节点离线数据从产生到成功同步到后台的耗时及重试次数

Monitoring

1、骑手手机存储空间不足100M上报到Firebase

2、骑手数据库占用存储空间占用超过500M或者1G上报到Firebase

Legacy & Risk Appraisal (Optional)

1、pickup离线模式批扫是否保留

2、在签名提交前同步离线数据，这样是否可能导致较多用户操作时卡在这一步骤，在签名提交步骤时后台校验后Task仍存在pending order如何处理？（APP校验成功，后台校验失败该如何提示用户）

3、本地校验通过，回传时校验失败的单，不同场景（还在扫描中，已经回到主页）需要怎么提示 driver？在pickup主页如何展示离线Task数据？

4、主页和详情页Task如何展示？比如对Task A做了离线状态扭转；在首页To-Pickup不展示这个Task，有网络后下拉刷新又出现这个Task，骑手是否有疑惑？

Tickets

Task content	PIC	JIRA Ticket
[SEA + BR] Support offline mode for Driver App - Pickup, Return and On-hold modes	Zhi.Xiao	SPXFM-33924