

编辑导语：很多人都有记账的习惯，但是记到后面却发现自己的帐算不清楚，记账不能只靠着单方面的账单，还要进行对账才能确保无误；本文将会从产品设计的业务知识点出发，详细介绍对账业务流程，并列举会出现的常见问题和解决方法。



业务背景：

对账模块是支付系统的核心能力之一，是信息流和资金流关联的重要依据，平台如果只使用渠道的单边账单或者平台流水订单，出现差错或渠道恶意扣单的风险极高。

为提高资金账务的正确性和保障平台的利益，需要通过平台系统对账能力与上游渠道对账单逐笔勾兑确认，如有差异能及时解决或归档。

用户画像：

- 1) 清结算专员：负责发起清分的操作者，首先确保信息流对平，然后确认资金流应收款和信息流平账账单金额一致。希望能及时发现长短款问题，并解决，保障资金清算给商户（平台可收款用户）的时效性。
- 2) 对账异常订单处理专员：负责核算异常订单原因，并在平台操作将异常订单执行修正、平账。

一、必须知道的业务知识点

1. 对账

在会计上概念：指为了保证账簿记录的正确性而进行的有关账项的核对工作；做到账证相符、账账相符、账实相符。

在支付系统上的体现：

1) 账证核对：是将账簿记录与记账凭证进行核对。这里是记账凭证是指第三方上游提供的渠道对账单，第三方渠道会根据对账单金额实际结算资金，也就是常说的信息流对账。

（有的支付公司或银行只要收到对账单了且平账，即使资金未实际到账，业务也允许发起清分以提高清算时效性）

2) 账账核对：是把有相互关系的多个账簿记录进行核对，有相互关系的账簿记录，包括总分类账簿间核对，明细账簿间核对等多种类型。整个支付系统可以被拆分成了多个子系统，如交易系统、账户系统、会计系统、账户系统，每个子系统在处理各自的业务并记录，其实就相当于会计理论中的账簿。系统间的对账，主要用于修正内部系统的数据不一致。

3) 账实核对：是各项资产物资的记录数值与实际真实数额间的核对。确认第三方汇款到银行账户资金和平账对账单结算金额是否匹配，也就是常说的资金流对账。

2. 轧帐

对账系统主要做的是信息流的对账，若对账中发现有差异的订单归类记入对账异常订单表，可称为轧帐。

3. 平帐

对账异常订单进入差错流程，可以通过人工或者自动的方式，按照事先设计好的规则处理这些异常差错，可称为平帐。

4. 渠道对账单

上游渠道会按照平台在其申请的渠道账户维度推送对账单，渠道账户也就是常说的支付通道。

如果是第三方支付公司或银行，上游渠道是微信、支付宝、银联二维码（云闪付）等等。
例如：支付平台申请有微信 2 通道和微信 6 通道，则微信侧会生成 2 份对账单文件。

每份对账单会包括支付成功订单和退款成功订单。第三方会以对账单中的结算金额（支付订单金额-支付订单手续费）-（退款订单金额-退款订单手续费）结算汇款给到平台资金账户。

5、银联二维码（难点）

银联二维码是银联平台自主推出的支付产品，C 端使用云闪付、各手机银行 APP 支付，订单底层走的都是银联二维码通道。

为什么银联二维码需要重点说呢？

因为它不同于微信、支付宝通道统一费率的原则，银联会根据 C 端用户支付时使用的银行卡借贷性质和交易金额是否大于 1000 作为费率规则，并且还会收取额外的品牌服务费，详情参见下图。

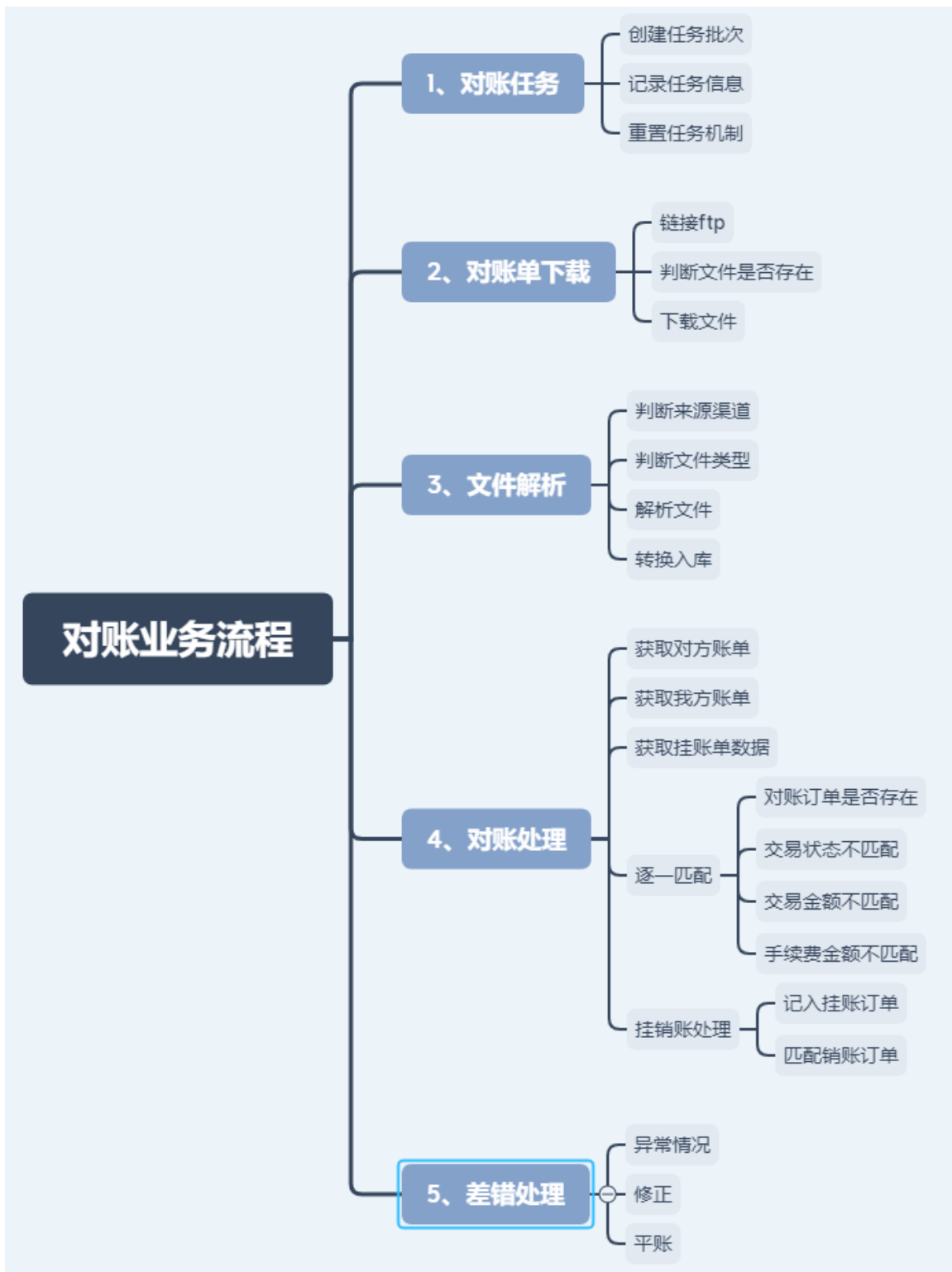
	A	B	C	D	E	F
1	本行卡支付费率					
2	卡种	交换费（发卡行收取）		转接费（银联收取）		品牌费（银联收取）
3		交易金额≥1000元	交易金额<1000元	交易金额≥1000元	交易金额<1000元	
4	借记卡	0.35%（13元封顶）	0.14%（5.2元封顶）	—	—	—
5	贷记卡	0.45%	0.18%	—	—	—
6						
7	他行卡支付费率					
8	卡种	交换费（发卡行收取）		转接费（银联收取）		品牌费（银联收取）
9		交易金额≥1000元	交易金额<1000元	交易金额≥1000元	交易金额<1000元	
10	借记卡	0.35%（13元封顶）	0.14%（5.2元封顶）	0.0325%（3.25元封顶）	0.013%（1.3元封顶）	—
11	贷记卡	0.45%	0.18%	—	—	0.02%
12						
13	备注：房地产、汽车销售商户品牌费封顶5000元，批发类商户2000元					
14						

所以设计银联二维码通道对账时，还需考虑到多费率及品牌服务费的场景。

二、对账流程

1. 业务流程

对账业务可以插解为 5 个业务环节，本文主要说明每个环节的能力职责、常见问题和通用解决方法，具体的产品方案还需要结合读者平台自身的业务特点和系统架构设计。



三、对账任务

对账是一个日常操作，正常情况下上游渠道都会以 D+1 的周期生成渠道对账单。

每天系统可以默认生成定时对账任务，每个上游渠道生成时间不一样，可以事前和上游确认，并结合平台对账的处理时效和商户到账需求，设计一个合理的时间执行。

对账任务设计前需确认，渠道对账单推送方式、解析方法、匹配字段，并提前做好联调适配工作；例如渠道有可能会需要申请白名单权限或提供 SFTP 地址信息，要谨防上线后才发现系统无法正常获取对账单的情况。

1. 创建任务批次

创建批次一方面是为了防止重复对账，另一方面需要在对账结束的时候将对账的结果信息存储到批次中。

2. 记录任务信息

对账任务信息，例如：通道名称、通道编号、渠道商户号、对账任务批次、对账任务状态、交易时间、任务创建时间、下载开始时间、下载结束时间、下载状态、对账开始时间、对账结束时间、对账结果、对账方式；

对账方式为对账处理时的对账规则，可以根据业务实际情况分为：无需对账、以渠道为准、以平台为准。

以渠道为准，则若对账订单平台交易状态为支付中或支付失败，但渠道为支付成功，则平台状态改为支付成功。以平台为准，则是若出现上述情况，对账订单记录为异常订单。

3. 重置任务机制

考虑到对账过程中可能会遇到的来自上游渠道的问题或平台系统自身问题，需要设计重置机制。

上游渠道对账单错误，需求二次或多次推送，所以需要设计重新下载渠道对账单或重新上传渠道对账单；有可能平台自身数据错误导致出现大量的差异订单，修复后需要重新对账。

4. 对账任务详情示例

对账信息：记录对账任务基本情况；**对账结果信息：**显示关键对账字段值；**挂销账信息：**显示是否有挂销账订单及金额；**对账异常信息：**显示是否有对账异常订单及金额；**备注：**将系统自动处理的过程记录，也可以手动修改。

对账信息

交易时间: 2019.08.08

支付通道:  支付宝2通道

对账结果: 对平

是否失效: 正常

对账结果信息

平台成功金额: 5,414,207.98

第三方成功金额: 5,414,207.98

平台退款金额: 55,457.60

第三方退款金额: 55,457.60

平台手续费: 55,457.60

第三方手续费: 55,457.60

平台交易净额: 55,457.60

银联品牌费: 55,457.60

平台应收金额: 55,457.60

挂销账信息

挂账金额: 0

挂账笔数: 0

销账金额: 0

销账笔数: 0

对账异常信息

成功异常金额: 0

成功异常笔数: 0

退款异常金额: 0

退款笔数: 0

备注

1、对平，无系统操作时显示：该批对账任务订单金额、状态均匹配，无异常
2、对平，系统自动挂销账提示：系统自动操作行为，例：平台支付订单：123456，已挂账。
3、对平，系统自动状态同步：平台订单号：1234567890，同步时间：2019.08.08 11:12:12，原状态：未支付，同步后状态：支付成功。
4、差异，平台订单状态为已支付第三方为未支付，例：平台订单号：123456，平台订单状态为已支付，第三方为未支付
差异，第三方订单号：1234，平台无该订单记录

四、对账单下载

1. 获取文件

渠道对账单获取方式，一般提前作为任务规则写死。

大多数银行都要求接入方提供 ftp 服务，银行定时将对账单推送到接入方提供的 ftp 服务器上面；

还有一部分银行会提供对账单的下载服务，通过 ftp/http 的都有，ftp 方式居多；

另外网银的对账单比较特殊，一般都需要结算登录网银的后台管理系统中，手动下载，结算下载完对账单后在导入到对账系统。

2. 判断文件是否存在

任务自动获取文件的情况下需要判断任务是否存在：

自动获取渠道对账单：不存在需要设置轮询，每间隔一段时间重新获取。重试次数和间隔的设置需要小心，重试太频繁，容易把服务器打死；时间间隔太大，又会阻塞后续处理步骤。5 ~ 10 分钟是一个合适的重试间隔区间。

手动导入渠道对账单：要设计导入入口，导入成功后任务状态也要做相应的变更。

3. 下载文件

技术实现上可以做成工厂模式，不同的支付渠道有不同的下载类，如果是 http 接口将文件写入到对账单，如果是 ftp 服务器，将服务器中的对账单下载到本地带解析的目录中。主要涉及的代码 ftp 工具类、http(s)工具类，相关 IO 读写。

4. 判断来源渠道

获取到上游对账单文件后，很有可能多个渠道的对账单在同一个 SFTP 地址，根据文件名匹配到对应的对账任务，文件名一般会包含账单时间、渠道商户号，然后再执行下一步。

五、文件解析

1. 解析文件

解析文件主要是将下载的对账文件解析成我们可以对账的数据类型并且入库。

解析的文件不同渠道有不同的类型，因此也可以设计成不同的解析模板，使用工厂模式将不同格式的文件解析成可以对账的统一数据类型。

解析的文件类型一般包括：json、text、csv、excel 等，另外部分银行会对账单做加密或者提供 zip 打包的格式，这里就需要额外开发 zip 工具类和加解密工具类进行处理。

对账文件中包含的主要信息有：商户订单号、交易流水号、交易时间、支付时间、付款方、交易金额、交易类型、交易状态这些字段。

2. 转换入库

每个渠道的账单格式都不尽相同，在得到账单后，下一步是对账单做标准化处理，这样轧帐以及后续工作就可以统一处理了。

标准化后的账单数据可以放在文件系统或者数据库中，这取决于交易数据量；每天百万以上的量，还是使用文件系统，比较合适，数据库操作相对比较慢，也浪费资源。

基于文件系统的标准化涉及如下内容：

文件格式标准化统一使用 csv 或者 json 或者 xml 格式，如果是使用 hadoop 或者 spark 来对账，也可以使用 csv。文件存储统一化文件目录，文件名都需要遵循统一命名规范。

六、对账单处理

1. 获取对方对账单

将事前准备的上游对账单标准文件放入缓冲对账池。

2. 获取我方对账单

本地交易记录的准备，总的来说有如下方法：

啥都不做，直接用订单表的原始数据：鉴于大部分系统使用的是 MySQL，这也意味着在 MySQL 上做对账。对账时需要大量的数据查找工作，必然会影响线上业务。在数据规模较大，比如超过 100 万时，就不太合适了。

使用备库来执行对账：这样既简单，也不影响线上业务，这是典型的空间换时间的做法。

采用分表分库对账：如果业务大到需要分表分库才能处理，那对账数据准备也不一样。

3. 逐一匹配

前文有提到对账方式有三种，不对账、以渠道为准、以平台为准，大部分的情况下的对账方式都以渠道为准，信息流的传递方向，支付成功结果是由上游渠道通知平台的，平台有可能会因网络或系统问题而没有收到通知。

一般按照交易金额、交易状态、手续费金额逐一匹配，对账方式选择以渠道为准的处理逻辑为例：

- 1) 交易金额不匹配：记入异常订单。
- 2) 交易状态不匹配：若上游为支付成功，我方为未支付或支付失败，则以上游为准。

若我方订单为支付成功，上游只会推送支付成功的订单为对账单，上游对账单不存在的情况，将我方订单记入为挂账订单；若上游对账单存在，我方不存在的情况，记入异常订单。

- 3) 手续费金额不匹配：记入异常订单。

4. 挂销账处理

常会存在因日切时间点不一致或网络延时等情况，导致我方平台订单时间与上游渠道时间不一致，同一个订单在渠道交易时间是1月1号，但在平台是1月2号。

挂账，对账时若上游无我方有，订单放入挂账订单中。销账，若匹配中挂账订单匹配上了，记录为当日平台账单，挂销账状态改为已销账。

七、差错处理

关于差错流程，每个系统的业务特性、运营团队流程、公司财务管理办法不一样，不能生搬硬套，但原则是一样的，所有的异常订单的报销账必须有理有据，多重审核。

1. 异常原因

(1) 若出现订单金额不一致的情况，一般是平台调用上游交易接口时，双方字段的定义不匹配导致的。

(2) 若出现手续费金额不一致的情况，一般是平台手续费计算规则和上游不匹配导致的，差额不会太大，可以设计阈值若在可以接受的范围类不计为异常。

(3) 若出现上游对账单存在，平台订单不存在的情况，通常是有第三方绕过平台系统与上游系统发生支付或退款交易。需要及时核查是否存在交易密钥泄露或被绕过商户去上游系统平台操作。

2. 修正

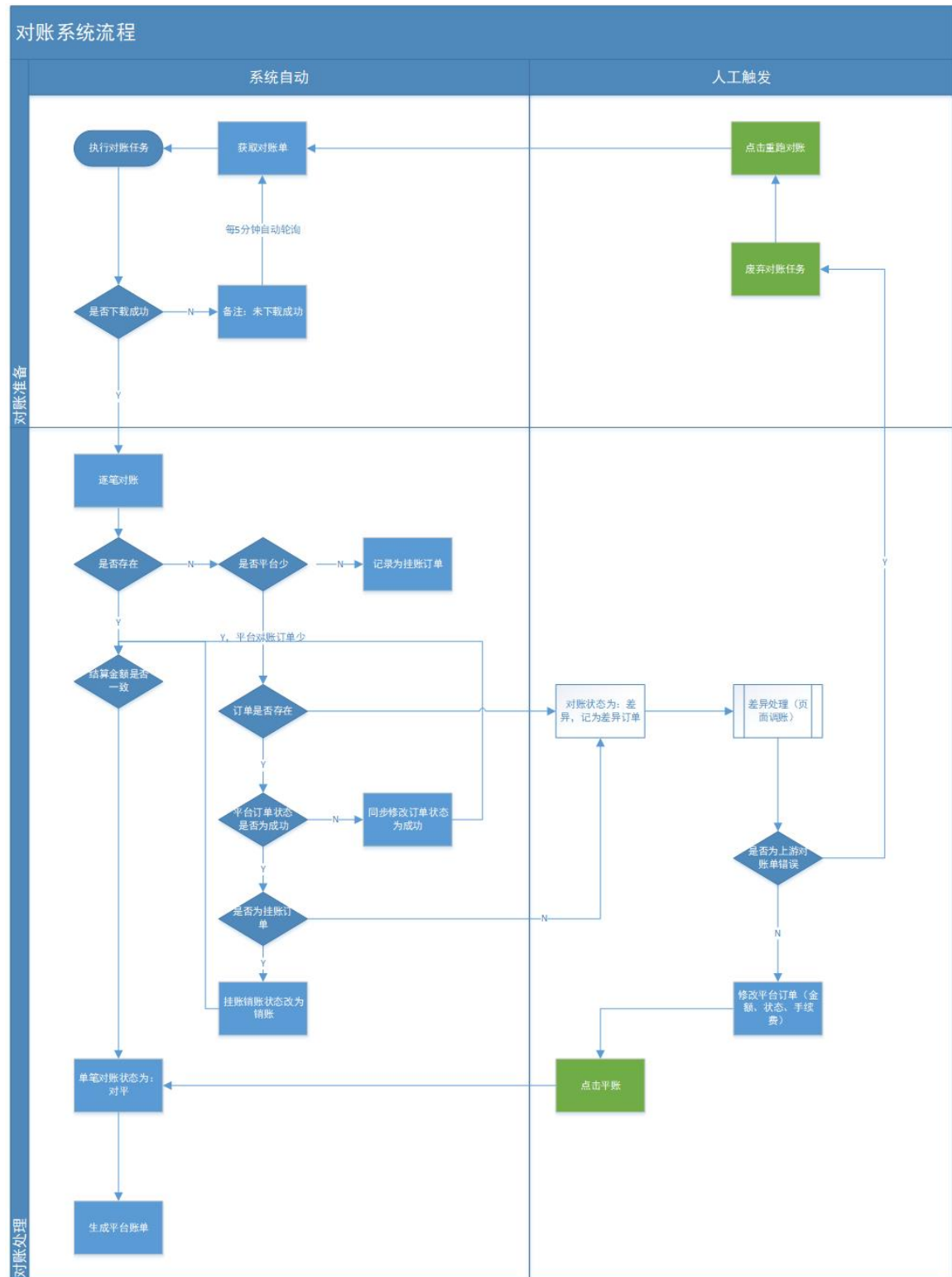
选择以平台为准或以上游为准，修改订单金额、订单状态、手续费金额。此时写入修正原因很重要，便于后期业务追踪考求。

3. 平账

将平台异常订单状态改为平账，并将平账时的时间作为账单时间，合入至平台账单。平台会根据平台账单计算渠道分润金额和商户结算金额。

八、业务规则系统化

最后，每个平台产品细节都会有其特定的业务规则，就不具体说明平台页面和平台操作功能，笔者不做详情说明。以上业务规则系统化，系统流程图如下，读者可以结合自己平台业务情况提取细化。



题图来自 Unsplash，基于 CC0 协议