

D05 计量与计费域 · 设计文档

本域总览见《账号计费中心 · 结构大纲》。本文为该域完整设计：定位与边界 / 角色与依赖 / 数据模型 / 核心流程 / 接口契约 / 关键约束与验收标准。

1. 定位与边界

定位：用量与金额的事实底座。回答四个问题——「一次调用产生了多少用量」（量）、「这些用量归谁」（归属维度）、「这些用量应付多少钱」（金额）、「上报的量与实扣的钱对得上吗」（对账）。

本域是唯一的用量落库方与对账判定方：所有通道的用量汇聚到同一张日聚合底座，所有对账结论由本域给出。按「先事实结构、再运行路径、后对外投影」的次序组织：第 3 节给出底座结构，第 4 节给出结构之上的写入/查询/对账路径，第 5 节给出这些路径的对外投影。

核心口径

口径	内容
量价分离	应付金额由计量方按自身计价规则即算即传；本侧只入账不计价，不读计价规则、不做二次换算
覆盖式	同一维度组合同一统计日重复上报 = 覆盖，不是累加；量可重算，对账重跑不双计
归属六元组	组织 / 组 / 成员 / 模型 / 密钥展示前缀 / 统计日期
日粒度	按统计日期聚合的日粒度底座，非实时明细；日期一律按 Asia/Shanghai 归日
金额口径	元、6 位小数、字符串传输；列级精度与资金侧口径对齐
对账容差	默认 0，即精确比对（两侧同源同值）
差异方向	差异 = 计费额 - 计量额：为正即多扣（计费额大于计量额），为负即缺扣
债务口径	对账差异只落处置台账与告警，不构成欠费判定依据（欠费只认现金余额为负，判定在钱包与账务域）

边界（不管什么）

- 不管理访问密钥：密钥的签发、轮换、失效属订购与支付域；本域只接收一个可读的展示前缀作为归属列，不建跨域外键。
- 不执行扣减：钱包余额的变动属钱包与账务域；本域上报应付金额，扣减与否、如何拆分赠金与现金由资金侧决定。
- 不定价：计价规则的发布与公示是平台能力，供计量方定价参考；本域运行时既不消费也不反向拉取它。
- 不判定准入：账户状态与准入属账户状态与准入域；本域既不读也不写准入状态，也不因对账异常改变任何账户状态。
- 不判定债务：对账发现的量钱差异只做台账与告警，欠费判定只看现金余额为负，不在本域。
- 不开票：账期汇总可作发票数据源，开票处理本身属发票域。

本域在多域中的位置

- 上游：调用产生的用量由各通道上报，本域既不做采集也不做反向拉取——推送制是唯一入口，因此「采集口径 vs 上报口径」这类分歧不会在本域出现。
- 中游：本域同时服务两类读取方——资金侧（账期账单与发票数据源）与运营侧（对账差异报表）。两者都读同一张底座，因此资金侧看到的金额不会与运营侧看到的金额产生口径分歧。
- 下游：处置动作不落在本域。本域只产生台账与告警，补扣、核销、人工调整等动作由资金侧与运营面执行。

2. 角色与依赖

参与角色

角色	身份	在本域的能力
实例计量侧	服务间调用方	按窗口聚合后推送用量行（覆盖式），随行带应付金额
数据面网关计量侧	服务间调用方	推送账户密钥通道的用量行（密钥展示前缀与实例侧不同源）
计费路由计量侧	服务间调用方	推送中转通道的用量行；组标识与成员标识可能来自另一命名空间
运营面	运营人员（持运营范围令牌）	查看对账差异报表、手动触发对账扫描
组织管理者	工作区 Owner / Admin	查看组织透视用量（任意维度组合、导出）
个人用户	门户用户	查看本人透视用量、账期账单

能力矩阵

操作	实例 / 网关 / 路由计量侧	运营面	组织管理者	个人用户
推送用量行	✓	✗	✗	✗
账期汇总读取	✗	✓	✗	✗
对账差异报表	✗	✓	✗	✗
手动触发对账扫描	✗	✓	✗	✗
组织透视用量	✗	✗	✓ (Owner / Admin)	✗
本人透视用量 / 账期账单	✗	✗	✗	✓

- 计量侧的令牌范围只覆盖写入，不覆盖账期汇总与对账读取——写入方无法通过本域读取其他组织的用量与金额。

上游依赖

- 租户与组织域：工作区、组、成员标识（作为归属维度的取值域，只读）。
- 订购与支付域：账户访问密钥的展示前缀（归属列取值，不建外键）。
- 钱包与账务域：扣减流水（对账的钱侧输入）；对账不读余额、不写余额。

下游被引用（谁引用本域）

引用方	引用内容	语义
钱包与账务域	用量日聚合	账期账单的用量侧数据源
发票域	账期汇总	开票数据源（按周期聚合后的用量与金额）
风控与合规域	用量日聚合	用量突增的阈值与环比判定输入；台账表由该域拥有
通知与触达域	站内信记录	对账异常与量级突增的送达（本域只借写站内信行）
用户自助与门户	聚合函数与口径	门户用量页、组织透视页、账单页复用同一聚合入口与分页口径

访问边界

- 内部路径不经任何公网入口：服务间调用一律走集群内地址，应用层令牌是第二道防线。
 - 令牌按调用方范围隔离：计量写链路 & 运营读链路分属不同范围，两侧严格互斥；未按调用方拆分时回落统一令牌。
 - 门户路径按用户维度隔离（会话令牌），组织透视另需组织级管理角色。
-

3. 数据模型

3.1 域内关联总览



图 1

三条弱引用线均为非同一域内的归属关系：本域刻意不为它们建外键（见 3.4 第 4 条）。

3.2 域间引用

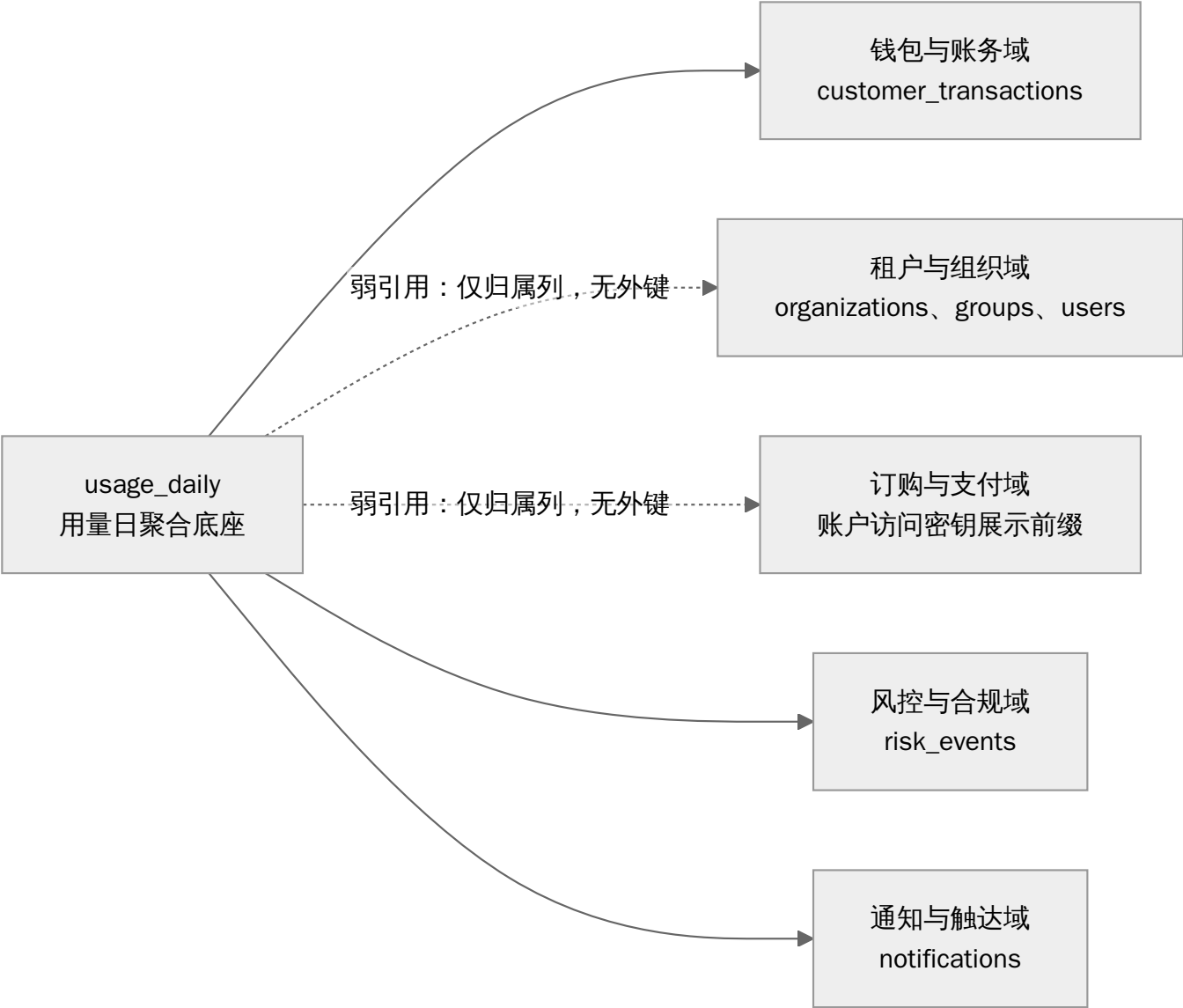


图 2

- 实线表示本域写入或强相关的读写路径，虚线表示只承载归属列、不建约束的弱引用。

3.3 表清单（自有 1 表 + 借写 2 表）

表	归属	职责	关键字段
usage_daily	本域自有	维度化用量日聚合底座（覆盖式）	organization_id / group_id / user_id / model / key_display / stat_date / requests / input_tokens / output_tokens / amount / quoted_amount / updated_at
risk_events	风控与合规域，本域借写	对账差异与量级突增的处置台账	rule_id / organization_id / group_id / customer_id / trigger_tokens / threshold_tokens / action / triggered_at
notifications	通知与触达域，本域借写	对账异常与量级突增的站内信送达	customer_id / type / group_id / title / content / read_at

- 归属口径：`usage_daily` 是唯一自有表；台账与站内信两表的表结构属他域，本域仅在对账扫描时写入行，故列为「借写」，不重复展开其设计。

3.4 设计约束

1. 六元组唯一：`uq_usage_daily_dims` 覆盖（工作区、组、成员、模型、密钥展示前缀、统计日期）；一切写入路径都必须命中该约束做覆盖，不得追加。
 2. 成员维度可空但不写空：`user_id` 列可空，但写入方在无法归因到人时一律归一为哨兵标识——空值在唯一约束下互不相等，直接写空会破坏覆盖式语义，使同一维度同一天分裂成多行。
 3. 密钥展示前缀不做跨域外键：`key_display` 只是可读标识，仅承载展示与归属切分；实例通道与账户通道的前缀来源不同，天然分属不同唯一行，从而两写者互斥、不双计。
 4. 归属一律弱引用：工作区、组、成员三列不建外键——用量是高频写入事实表，外键约束会拖住写入路径，且归属主体可能先于用量行被关闭。
 5. 金额列可空、精度固定：`amount` 与 `quoted_amount` 均为元、固定 6 位小数；未计价时为空，覆盖写入按「新值优先、空值不清旧值」处理。
 6. 日期窗口受限：`stat_date` 只接受 [今天 - N, 今天] (Asia/Shanghai)，N 由 `usage_stat_date_backfill_days` 给定、默认 90 天；越界整批拒绝。
 7. 只增不改不删：`usage_daily` 不提供删除路径，覆盖是唯一更新方式；`updated_at` 记录最后一次覆盖时刻，可据此判断窗口是否被重推。
 8. 查询索引：`ix_usage_daily_stat_date` 支撑按日期区间的全量聚合；`ix_usage_daily_group_date` 支撑对账的按组 × 统计日聚合。
 9. 借写最小化：本域只向台账与站内信写入必要列，不扩列、不改其结构；金额类信息在台账中以既有列类型可承载的形式落（见 6.3 开放项 2）。
-

4. 核心流程

4.1 用量上报：覆盖式 UPSERT

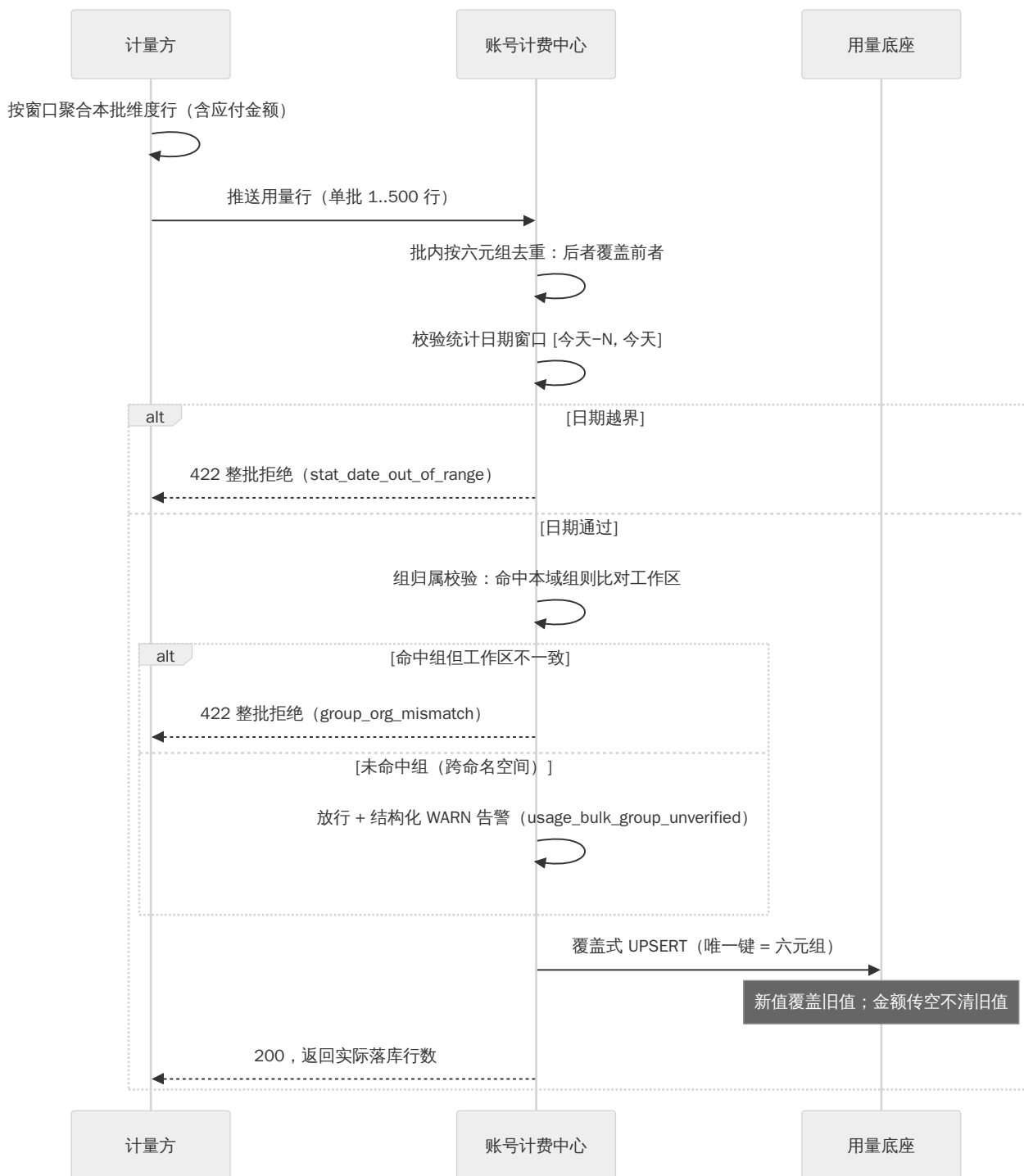


图 3

- 单批行数上限 500；批内重复维度先收敛为一行，避免单语句二次命中同一唯一行导致的整批失败。
- 「实际落库行数」是批内去重后的行数，不是请求行数——推送方据此可发现自身聚合异常。
- 覆盖式是唯一写入语义：同一窗口重推多少次，底座里都只有一行，对账重跑安全。
- 整批同事务：一批的校验与写入在同一事务内完成，全部成功或全部不产生——不存在「批内部分行已落库」的中间态，重推即从头覆盖。

4.2 写入侧校验与告警

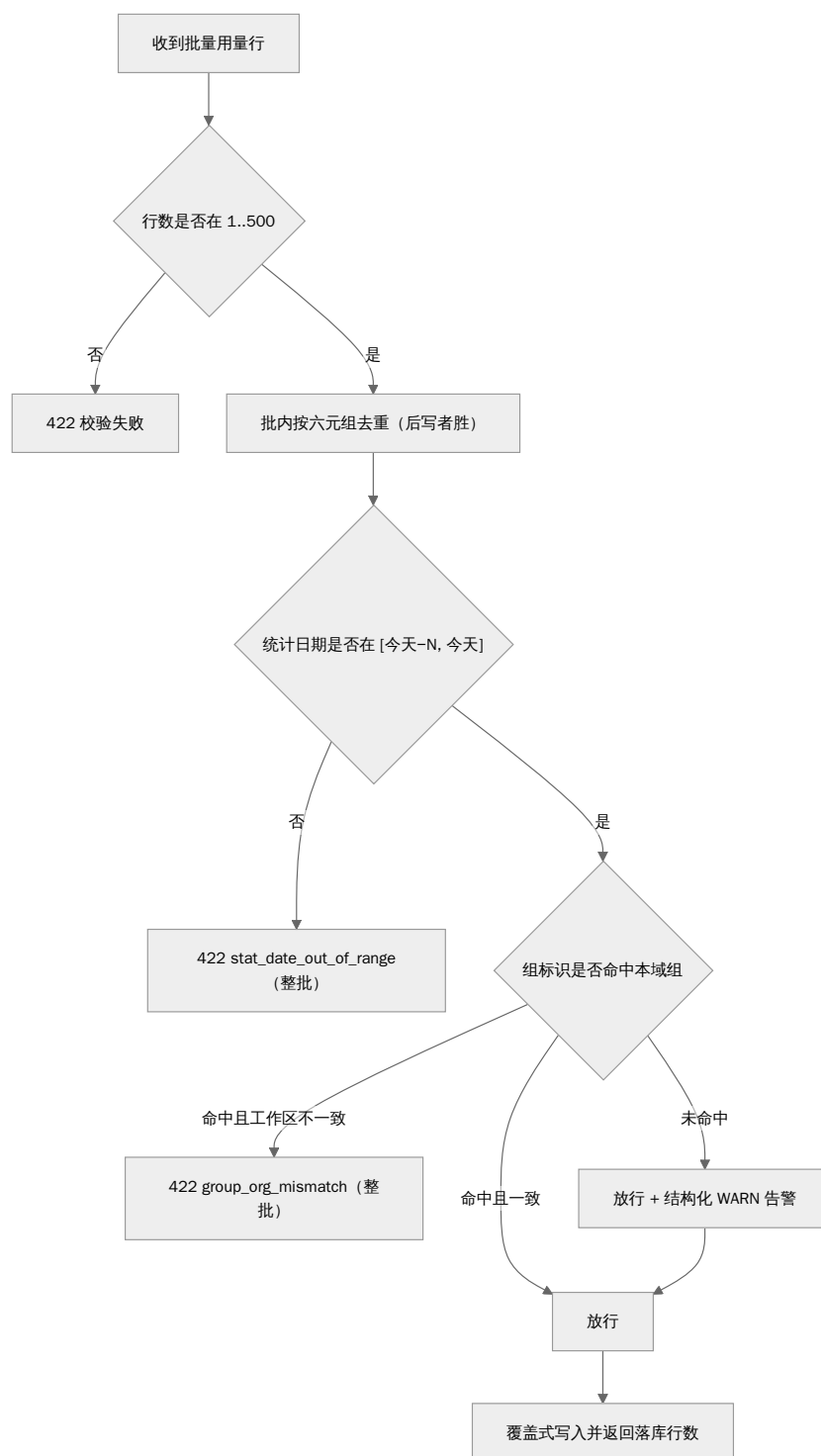


图 4

- 告警为结构化 WARN 日志，每批一条，载荷含未校验行数、总行数、组标识样例、工作区标识样例；日志聚合侧按该标记建规则即可持续观测「未被校验的流量」。
- 未命中的组标识不等于伪造：中转通道推送的组标识来自另一命名空间，与本域组标识不同源，恒查不到；若按「必须存在」拒绝，该通道每批失败并中止整轮推送。故口径为「放行 + 告警」，真伪鉴别交给对账与告警量级突变监控。
- `usage_stat_date_backfill_days` 取值需覆盖计量采集的迟到重推与对账重跑跨度；越界拒绝的理由是：未来日期污染长期报表，任意历史日期污染已封账账期。

4.3 归属维度与不可归因处理

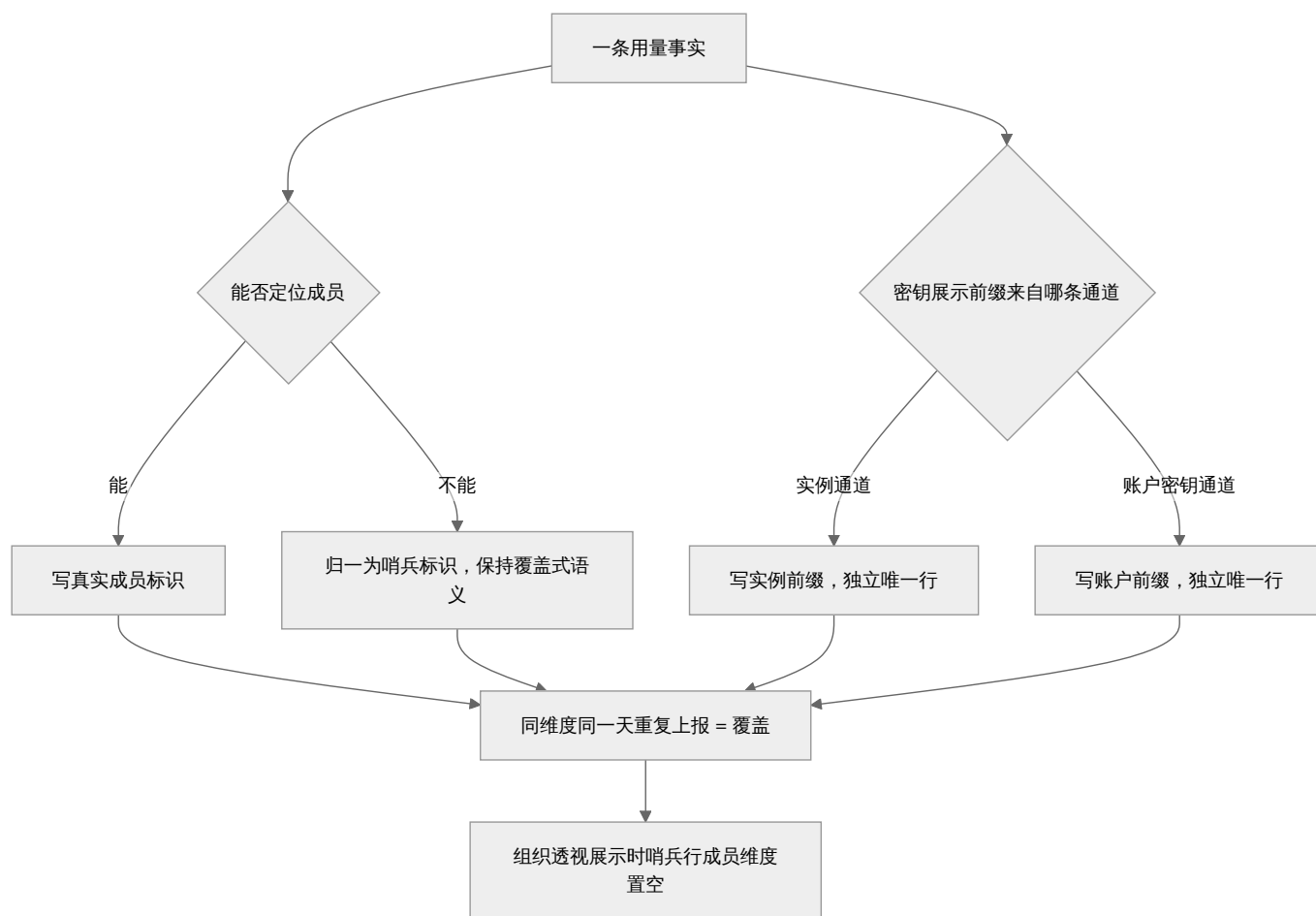


图 5

- 哨兵行在组织透视里展示为空，不把内部哨兵标识透给用户。
- 两条通道前缀来源不同，「同一时刻两方各写一行」是设计内行为而非双计——它们描述的是不同密钥的用量。
- 量与金额同行走同一行，量价同期覆盖回填；金额晚到（首次上报不带金额、后续重推带金额）也能收敛。

4.4 用量查询：维度聚合、分页与汇总口径

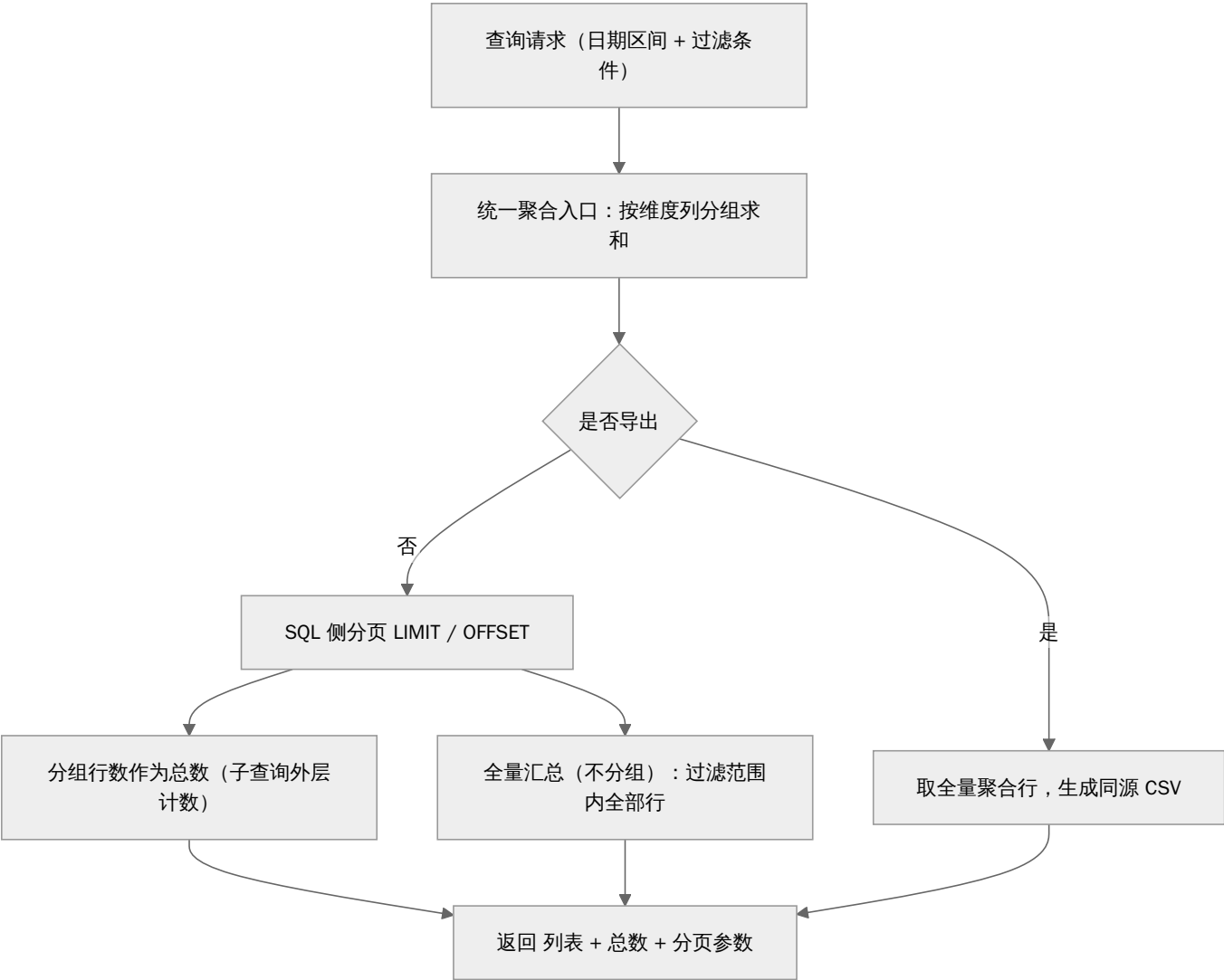


图 6

- 单聚合入口：页面查询、CSV 导出、组织透视、账期账单、对账全部走同一聚合实现，只有分组维度不同；口径不因入口而异。
- 个人透视分组维度为（统计日期、模型、密钥展示前缀），并按本人成员标识过滤；默认区间为最近 30 天。
- 组织透视分组维度为日期 / 组 / 成员 / 模型 / 密钥展示前缀的任意子集，默认「组 × 日期」；非法维度立即拒绝。
- 总数口径为「聚合后的分组行数」，用分组子查询外层计数，不把全量行拉进内存再数。
- 汇总口径为过滤范围内全量的合计（请求数、输入/输出 token、金额），不随当前页变化——翻页不会改变汇总卡数字。
- 导出为全量聚合行：导出取的是聚合后的分组行（行数受维度组合约束），不是原始明细全量，因此导出不会随底座行数线性膨胀。

4.5 对账：量与钱逐组逐日精确比对

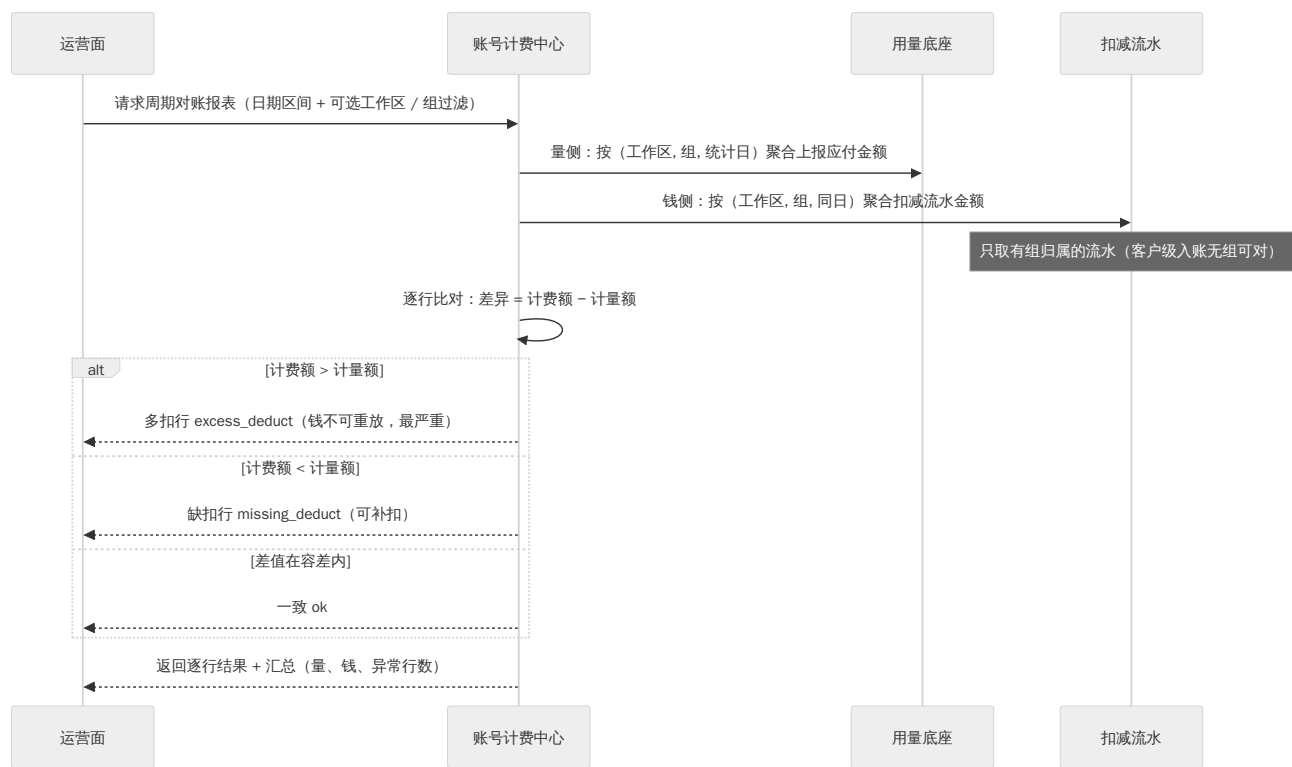


图 7

- 比对键 (组织, 组, 统计日期)：量与钱都按同一归日口径聚合 (Asia/Shanghai)，不随数据库会话时区漂移。
- 容差 reconcile_tolerance_amount 默认 0，即精确比对；两侧同源同值，无例外。
- 金额口径一致：上报应付金额与扣减流水金额都是元、6 位小数；减免话单的留痕金额恒为 0，只计数、不进量钱等式。
- 报表只输出「有量或有钱」的行；纯入账日（只有充值或赠金）不产生核对行。
- 每行附客户标识与工作区名称，供运营面直接进入人工核对。
- 处置语义：多扣为最严重项（钱不可重放，需人工介入）；缺扣通常是对端幂等重试积压，可补扣。两者都只出报表与台账，本域不执行任何补扣或冲销。

4.6 对账异常扫描：周期循环、幂等与多副本安全

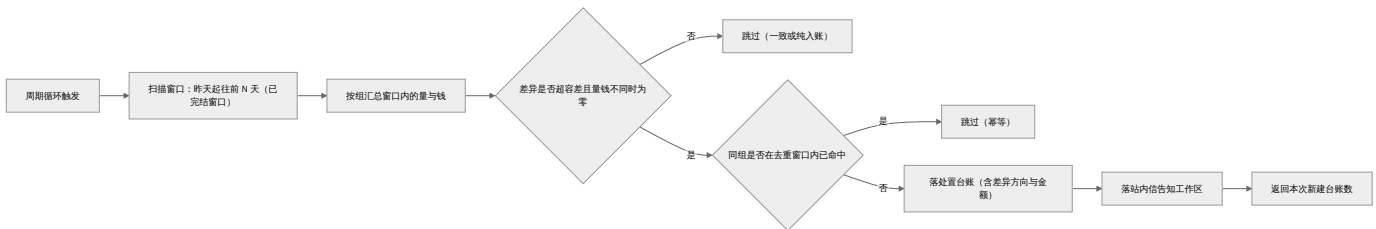


图 8

- 扫描窗口：reconcile_scan_window_days 默认 1，即只扫「昨天」这一已完结窗口，避免在途数据误报。
- 容差与去重：差异绝对值需超过 reconcile_tolerance_amount（默认 0）才判异常；同组在 reconcile_dedupe_hours（默认 24 小时）内不重复命中。

- 台账规则惰性内置：本域内置对账异常规则（规则类型 `reconcile`），平台级、首次需要时创建；台账行以该规则为外键，动作固定为通知类。
- 多副本安全：循环为进程内周期循环，开关与间隔由 `account_reconcile_scan_enabled`、`account_reconcile_scan_interval_seconds` 控制；多副本同时运行不产生重复台账——去重键为「同规则 × 组织 × 组」，重复扫描不重复落行。
- 手动触发等价：运营面手动触发与周期循环共用同一实现，返回本次新建台账数，并写运营操作留痕。
- 差异只作处置线索：台账与告警都不改变余额、不改变账户状态、不驱动欠费判定。
- 循环健壮性：单轮扫描异常不退出循环，只记录错误后继续下一轮，避免一次瞬时故障导致对账长期停摆。

4.7 抵消类流水（免计费留痕）量级突增扫描

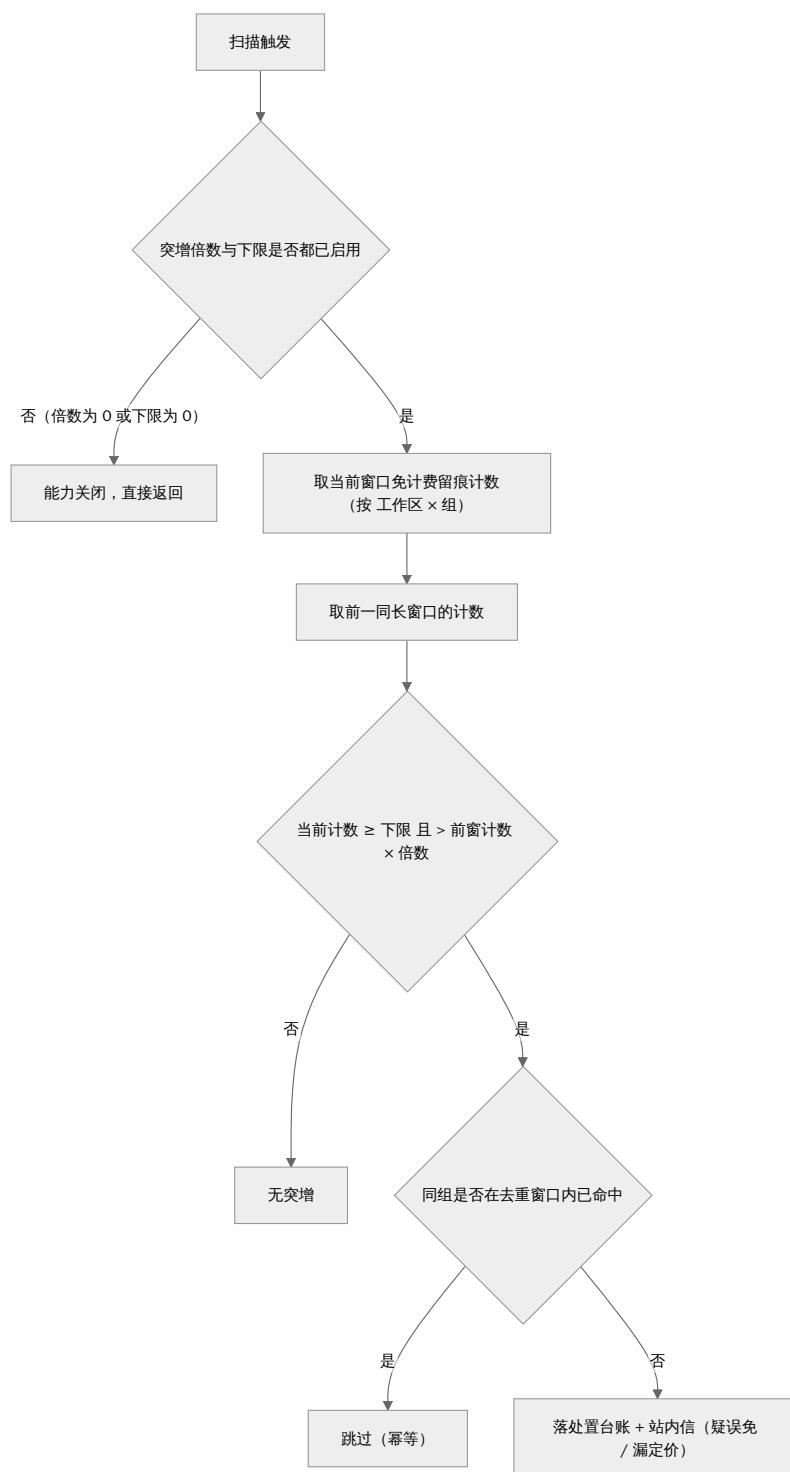


图 9

- 判据：当前窗口免计费留痕计数 $\geq$ `reconcile_waiver_spike_min_count`（默认 10，小样本豁免）且 $>$ 前一窗口计数 $\times$ `reconcile_waiver_spike_ratio`（默认 5）；倍数设为 0 或下限设为 0 即关闭该项。
- 免计费留痕金额恒为 0，不计入对账等式，只按（工作区、组、日）计数；两侧计数的不等由对端每日比对，本域只做自身量级监控。
- 内置规则同为平台级、惰性创建，规则类型为 `waiver_spike`。

4.8 量价分离的边界

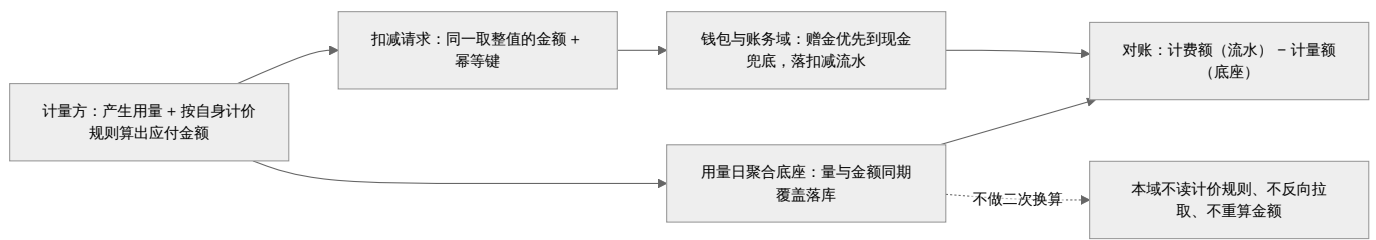


图 10

- 上报金额与扣减金额取同一取整值，两侧才能对上；本域只把金额原样落库，不做任何换算。
- 计价规则保留为平台「发布与公示」能力，供计量方参考定价；本域运行时不消费它。
- 对账侧的「钱」来自流水（不可重放、契约幂等），「量」来自底座（可重算、覆盖式）——两者幂等性质不同，这正是本域用覆盖式兜住重跑、用容差 0 约束钱的根因。

5. 接口契约

5.1 用量写入 (/internal/usage)

方法	路径	语义
POST	/internal/usage/bulk	批量推送用量行 (1..500 行)：覆盖式 UPSERT，返回实际落库行数；日期越界返回 stat_date_out_of_range，命中本域组但工作区不一致返回 group_org_mismatch

- 鉴权：内部服务令牌（计量写范围）；调用方不携带用户会话。

5.2 账期汇总与对账 (/internal/usage、 /internal/reconcile)

方法	路径	语义
GET	/internal/usage/summary	账期汇总：按 (组, 统计日) 聚合的用量 + 流水按类型/组的聚合 (发票与账单数据源)；结束日期早于开始日期返回 end_date_before_start
GET	/internal/reconcile/check	对账差异报表：周期内每 (组 × 统计日) 的量/钱 + 交叉校验结果 + 汇总
POST	/internal/reconcile/scan	手动触发对账异常检测 (幂等)：量钱差异与免计费量级突增一并扫描，返回新建台账数

5.3 运营面对账 (/api/account/ops)

方法	路径	语义
GET	/api/account/ops/reconcile/check	对账差异报表 (运营面入口，与内部实现共用同一实现，不做双轨)
POST	/api/account/ops/reconcile/scan	手动触发对账扫描 (幂等)，并写运营操作留痕

- 鉴权：门户令牌 + 运营范围细粒度校验；运营面端点索引统一登记在 D11，语义细节以本域为准。

5.4 门户用量查询 (/api/account)

方法	路径	语义
GET	/api/account/usage	本人透视用量：按（统计日期、模型、密钥展示前缀）聚合分页，默认最近 30 天；返回总数与分页参数
GET	/api/account/org/usage	组织透视用量：维度组合可配（默认组 × 日期），返回列表 + 全量汇总 + 总数；导出时返回同源 CSV
GET	/api/account/bills	账期账单：用量日汇总 + 按模型聚合 + 流水按类型聚合

- 组织透视需组织级管理角色；维度非法返回 invalid_dimensions，日期反序返回 end_date_before_start。
- 门户编排与页面能力清单详见 D12，本域负责其用量口径。
- 三个门户端点的返回结构对齐：金额一律元、6 位小数字符串（无计价时为 空），请求数与 token 数为整数。

6. 关键约束与验收标准

6.1 约束

- 覆盖式是唯一写入语义：任何通道、任何重推，同一六元组窗口在底座里恒为一行；不存在累加路径，也不存在删除路径。
- 维度完备且非空：工作区、组、模型、密钥展示前缀、统计日期必填；成员维度可空，但写入时归一为哨兵，保证唯一约束恒可命中。
- 写入即校验、校验即整批：批内去重、日期窗口、组归属三类校验任一不通过则整批拒绝，不产生部分写入。
- 归属弱引用：所有归属列都不建跨域外键，避免约束拖住高频写入路径，也避免归属主体关闭时拖累已落库的用量事实。
- 金额字符串口径：对外一律元、6 位小数字符串；列级精度固定，禁止浮点表示。
- 对账容差为 0：量与钱同源同值；差异只有多扣与缺扣两个方向，判定以「计费额 - 计量额」符号为准。
- 差异落地即留痕：每一处超容差差异都落处置台账并送达站内信，不静默吞掉；台账行不改变任何资金状态。
- 扫描幂等且多副本安全：周期扫描与手动触发共用同一实现；以「同规则 × 组织 × 组 × 去重窗口」判重，重复运行不产生重复台账。
- 债务口径外置：对账异常与免计费突增都不构成欠费判定依据；欠费判定只认现金余额为负，在钱包与账务域。
- 同一聚合入口：页面、导出、账单、对账只走一个聚合实现，差异只体现在分组维度与过滤条件上。
- 访问范围互斥：计量写链路 & 运营读链路分属不同令牌范围，一个范围不可执行另一范围的端点；内部路径不经公网入口。

6.2 验收标准

#	验收口径
1	同一六元组连续推送两次且量不同：底座只有一行，取后一次的值（覆盖而非累加）

#	验收口径
2	同一批内含重复维度：请求成功，落库行数等于去重后的行数，不出现整批失败
3	成员维度为空的上报：写入归一为哨兵，重复上报仍覆盖同一行（不分裂成多行）
4	实例通道与账户密钥通道各推一行、其余维度相同：底座两行并存、互不覆盖，合计等于两行之和
5	统计日期超出 [今天 - N, 今天]：整批拒绝并返回越界原因，底座无任何新增行
6	组标识命中本域组但工作区不一致：整批拒绝；组标识未命中：放行且产生一条结构化告警
7	覆盖写入时金额传空：原有金额保留（不被抹成空）；首次上报即带金额：正常落值
8	个人透视查询：只返回本人成员维度的行，总数与分页正确，翻页不改变汇总数字
9	组织透视任意维度组合：总数等于聚合后分组行数，汇总等于过滤范围内全量合计（不随当前页变化）
10	量与钱完全一致时报表全行为一致；仅钱多时标为多扣、仅钱少时标为缺扣，差异额等于两者之差
11	无组归属的入账流水不出现在对账报表行内；纯入账日不产生核对行
12	连续两次扫描同一窗口：第二次不产生新台账；多副本同时扫描亦不产生重复台账
13	免计费留痕金额恒为 0 且不计入量钱等式；当前窗计数相对前一窗口超过倍数且不低于下限时产生台账与站内信，低于下限时不产生
14	突增倍数设为 0 或下限设为 0 时，免计费突增扫描不产生任何台账
15	对账异常落台账后，工作区余额与账户状态均不发生变化
16	内部写入与汇总端点无内部令牌不可访问；运营面对账端点无运营范围不可访问

6.3 待后续设计细化的开放项

- 1. 跨命名空间组标识的映射收敛：中转通道推送的组标识与本域组标识不同源，现为放行加告警；是否由对端改推本域组标识、或引入映射表，需与对端一并定案。
- 2. 台账金额列的精度收敛：处置台账的异常金额现以最小货币单位的整数落（受既有列类型约束），随台账结构重写收敛为元小数口径。
- 3. 迟到重推与已封账账期的边界：回溯窗口目前由参数给定；是否引入按账期封账的硬边界，需与账期结算流程一并设计。
- 4. 减免额的可核对性：原应有费用已可承载「原价 - 实收」求差；减免原因的分类统计与运营可核对口径待细化。
- 5. 用量突增与对账差异的联动：用量突增由风控与合规域扫描、对账差异由本域扫描，两者在台账层是否合并成一类处置线索待定。

6. 口径迁移边界日的例外：两侧同源同值、容差为 0；历史上因计量单位切换产生的存量行是否需要一次性容差窗口，待与资金侧确认后收口。