

福建水务营收系统-营收业务模块正文

系统设计团队

2024年12月19日

- 1 [福建水务营收系统详细设计-营收业务模块正文](#)
 - 1.1 [章节导航 \(精简\)](#)
 - 1.2 [文档定位](#)
- 2 [营收业务详细设计](#)
 - 2.1 [营收模块统一约束](#)
 - 2.2 [接口与数据追溯矩阵](#)
 - 2.3 [REV-001 客户资料管理](#)
 - 2.3.1 [功能说明](#)
 - 2.3.2 [关键设计](#)
 - 2.3.3 [核心数据](#)
 - 2.3.4 [迁移补充 \(旧系统承接\)](#)
 - 2.3.4.1 [定额共享](#)
 - 2.3.4.2 [定额核定](#)
 - 2.3.4.3 [批量修改](#)
 - 2.3.5 [接口映射](#)
 - 2.3.6 [落地边界](#)
 - 2.4 [REV-002 抄表开账](#)
 - 2.4.1 [功能说明](#)
 - 2.4.2 [业务流程](#)
 - 2.4.3 [关键规则](#)
 - 2.4.4 [开账触发与结果表达](#)
 - 2.4.5 [核心数据](#)
 - 2.4.6 [迁移补充 \(旧系统承接\)](#)
 - 2.4.6.1 [特殊开账](#)
 - 2.4.6.2 [开账记录迁移](#)
 - 2.4.7 [接口映射](#)
 - 2.4.8 [落地边界](#)
 - 2.5 [REV-003 营业收费](#)
 - 2.5.1 [功能说明](#)
 - 2.5.2 [业务流程](#)
 - 2.5.3 [关键规则](#)
 - 2.5.4 [核心数据](#)
 - 2.5.5 [迁移补充 \(旧系统承接\)](#)

- [2.5.5.1 柜台结账](#)
- [2.5.5.2 账单打印服务](#)
- [2.5.5.3 红冲记录](#)
- [2.5.6 接口映射](#)
- [2.5.7 落地边界](#)
- [2.6 REV-004 账务处理](#)
 - [2.6.1 功能说明](#)
 - [2.6.2 业务流程](#)
 - [2.6.3 关键规则](#)
 - [2.6.4 核心数据](#)
 - [2.6.5 主要场景](#)
 - [2.6.6 迁移补充（旧系统承接）](#)
 - [2.6.7 接口映射](#)
 - [2.6.8 落地边界](#)
- [2.7 REV-005 发票与税务处理](#)
 - [2.7.1 功能说明](#)
 - [2.7.2 业务流程](#)
 - [2.7.3 状态说明](#)
 - [2.7.4 关键规则](#)
 - [2.7.5 后台申请入口与校验补充](#)
 - [2.7.6 SYS-008 异步协同与查询补偿](#)
 - [2.7.7 终态保护与异常核查](#)
 - [2.7.8 结果回写、账单关联与客户侧消费](#)
 - [2.7.9 核心数据](#)
 - [2.7.10 迁移补充（旧系统承接）](#)
 - [2.7.11 接口映射](#)
 - [2.7.12 落地边界](#)
- [2.8 REV-006 催缴与通知](#)
 - [2.8.1 功能说明](#)
 - [2.8.2 业务流程](#)
 - [2.8.3 关键规则](#)
 - [2.8.3.1 催缴对象筛选、排除与频控边界](#)
 - [2.8.4 核心数据](#)
 - [2.8.4.1 催缴对象与规则摘要](#)
 - [2.8.4.2 四态与人工核查边界](#)
 - [2.8.5 迁移补充（旧系统承接）](#)
 - [2.8.5.1 催缴记录](#)

- [2.8.5.2 停水记录](#)
 - [2.8.5.3 预存短信](#)
- [2.8.6 接口映射](#)
- [2.8.7 落地边界](#)
- [2.9 REV-007 统计分析](#)
 - [2.9.1 功能说明](#)
 - [2.9.2 关键设计](#)
 - [2.9.3 核心数据](#)
 - [2.9.3.1 统计主题与口径摘要](#)
 - [2.9.4 接口映射](#)
 - [2.9.5 落地边界](#)
- [2.10 REV-008 代收与银行业务](#)
 - [2.10.1 功能说明](#)
 - [2.10.2 业务流程](#)
 - [2.10.3 关键规则](#)
 - [2.10.4 核心数据](#)
 - [2.10.4.1 文件传输配置与审计补充](#)
 - [2.10.5 迁移补充（旧系统承接）](#)
 - [2.10.5.1 银行托收](#)
 - [2.10.5.2 实时收费查询与对账](#)
 - [2.10.5.3 当前实现对齐说明](#)
 - [2.10.6 接口映射](#)
 - [2.10.7 落地边界](#)
- [2.11 REV-009 业务参数配置](#)
 - [2.11.1 功能说明](#)
 - [2.11.2 关键设计](#)
 - [2.11.3 核心数据](#)
 - [2.11.4 迁移补充（旧系统承接）](#)
 - [2.11.5 接口映射](#)
 - [2.11.6 落地边界](#)

1 福建水务营收系统详细设计-营收业务模块正文

1.1 章节导航（精简）

- [文档定位](#)
- [营收业务详细设计正文](#)

- [营收模块统一约束](#)
- [接口与数据追溯矩阵](#)
- [REV-001 客户资料管理](#)
- [REV-002 抄表开账](#)
- [REV-003 营业收费](#)
- [REV-004 账务处理](#)
- [REV-005 发票与税务处理](#)
- [REV-006 催缴与通知](#)
- [REV-007 统计分析](#)
- [REV-008 代收与银行业务](#)
- [REV-009 业务参数配置](#)

1.2 文档定位

本文档为 `01_Detailed_Design.md` 中“营收业务详细设计”章节的模块正文拆分稿，便于按模块独立维护。正式交付口径以主详设为准。

2 营收业务详细设计

2.1 营收模块统一约束

- `SYS-002` 负责营收主流程，发票、支付结算、消息触达分别通过 `SYS-008`、`SYS-009`、`SYS-010` 协同完成，外部系统仅回写结果状态。
- 账单、收费、发票、代扣等关键状态变更必须通过业务流程驱动，不允许绕过业务校验直接改写主业务对象。
- 幂等控制遵循接口主键约束：支付以业务订单号为主，发票以申请单号或账单组合为主，代扣以批次号为主，消息以业务事件号为主。
- 账务调整、发票申请、催缴触达、银行批次下发等关键动作必须写入操作留痕，满足审计与问题追踪要求。
- 数据口径统一采用 `biz_*` 与 `bk_*` 命名体系，历史命名仅用于追溯，不作为本章节正式设计口径。

2.2 接口与数据追溯矩阵

说明：详细字段与报文以 `../03_Technical_Design/03_Interface_Design.md` 为准，数据库字段口径以 `../03_Technical_Design/01_Database_Design.md` 为准。

REV 模块	关键接口	核心数据域（摘要）	主要协同 对象
REV-001 客户资料管理	IF-REV-001	biz_cust、biz_account、biz_cust_*	客户服务模块、报装模块
REV-002 抄表开账	IF-REV-004、IF-REV-005	biz_meter_book、biz_meter_read、biz_reading_*、biz_charge*	抄表APP、物联网集抄
REV-003 营业收费	IF-REV-006	biz_charge*、biz_collection、bk_transaction*	SYS-009
REV-004 账务处理	IF-REV-007	biz_charge*、biz_operat_log*	财务与营业人员
REV-005 发票与税务处理	IF-REV-008	biz_invoice*、biz_cust_invoice	SYS-008
REV-006 催缴与通知	IF-REV-013	biz_charge*、催缴结果留痕	SYS-010
REV-007 统计分析	IF-REV-010	客户、抄表、收费、渠道聚合对象	统计分析端
REV-008 代收与银行业务	IF-REV-011	bk_withholding_*、bk_reconcile_*、bk_settlement_*	SYS-009、银行
REV-009 业务参数配置	IF-REV-012	biz_parameter_settings、biz_page_settings*、biz_price_*	统一平台、营收模块

2.3 REV-001 客户资料管理

2.3.1 功能说明

负责客户主档、账户主档、联系人、客户分组、客户与水表关系、客户开票信息、客户渠道绑定及托收/代扣关系维护，是抄表、收费、发票、代扣等后续业务的主数据基础。

2.3.2 关键设计

1. 客户建档覆盖立户、变更、更名、过户、销户、报停等全生命周期处理。
2. 客户资料支持按客户类型、用水性质、片区、集团客户、重点客户等维度管理。

3. 客户与水表、开票主体、托收/代扣签约关系按关联表维护，避免在主表中堆叠多类属性。
4. 客户编号、集收标记、计划用水方案等规则类数据由统一配置与客户关系表共同支撑。

2.3.3 核心数据

- `biz_cust`：客户主档。
- `biz_account`：客户账户与账户状态。
- `biz_cust_contact`：联系人及联系方式。
- `biz_cust_group`：客户分组。
- `biz_cust_meter`：客户与水表绑定关系。
- `biz_cust_invoice`：客户开票信息。
- `biz_cust_app_binds`：渠道绑定关系。
- `biz_cust_collection_rel`：客户托收关系。
- `biz_cust_withholding_rel`：客户代扣关系。
- `biz_cust_water_use_scheme`、`biz_cust_water_scheme_rel`：客户计划用水方案关系。
- `biz_cust_no_rule`：客户编号规则。
- `biz_cust_hub_marks`：集收号/集收标记关系。

2.3.4 迁移补充（旧系统承接）

2.3.4.1 定额共享

- 旧系统在“客户资料”下提供定额主客户、子客户绑定与共享清单管理能力。
- 当前正式设计不新增并行主模型，统一归入客户与计划用水方案关系对象承接。
- 迁移时必须至少保留主客户、子客户、绑定状态、生效时间、解除时间、备注与操作留痕，确保共享清单可查询、可解绑、可追溯。

2.3.4.2 定额核定

- 旧系统支持对已建立共享关系的客户执行定额核定、撤销核定和共享清单联查。
- 当前建议以客户关系对象和计划用水方案关系为主承接核定结果，不额外发明独立账务主表。
- 核定结果迁移后应支持按客户、站点、核定状态、核定时间查询，并保留核定依据、操作人和变更留痕。

2.3.4.3 批量修改

- 旧系统已形成“手工修改 + 模板导入修改”的双模式客户资料维护能力。
- 当前正式设计应将其视为客户主数据治理能力的一部分，而不是临时导入工具。
- 迁移时需明确三类口径：可批量修改字段范围、导入模板校验规则、批量修改审计留痕；历史批量修改结果至少需保留任务来源、修改字段、执行结果和失败原因。

2.3.5 接口映射

- `IF-REV-001`：客户档案、账户状态、联系人与水表绑定关系查询。
- `IF-CS-001`、`IF-CS-002`：客户服务侧账户绑定与信息查询场景复用客户域数据。

2.3.6 落地边界

- **已落地**：客户主档、账户、联系人、分组、绑定、开票、托收/代扣关系。
- **部分落地**：客户扩展关系、集收与规则类对象已见明确关系表，但具体业务场景仍需结合流程进一步细化。
- **文档先行**：主副卡、部分复杂客户关系对象在当前 backend 中未见完全独立表族，文档中仅保留业务对象表述。

2.4 REV-002 抄表开账

2.4.1 功能说明

负责抄表计划、册本管理、抄表录入、抄表状态跟踪、异常复核、计费计算与账单生成，是营收核心处理链路的起点。

2.4.2 业务流程

```
flowchart TD
    A[制定抄表计划] --> B[生成抄表册本]
    B --> C[分配抄表任务]
    C --> D[人工/远传/自报抄表]
    D --> E[数据校验]
    E --> F{是否异常}
    F -->|是| G[异常复核处理]
    F -->|否| H[生成开账数据]
    G --> H
    H --> I[按价格模板与费用组成计费]
    I --> J[生成营业账与明细]
    J --> K[账单审核确认]
    K --> L[进入收费/催缴/开票]
```

2.4.3 关键规则

- 抄表数据同时校验本次读数、上次读数、用量波动和抄表状态，只有通过校验或完成异常复核的数据才能进入开账。
- 异常抄表支持估抄、补抄、重录、人工复核；异常处理的目标是形成可继续开账或明确阻断的统一结论，而不是绕开开账规则单独落账。
- `IF-REV-005` 的正式边界是“抄表校验完成后的账单生成”，不负责收费核销、发票开具、催缴执行和统计分析。
- 开账计费按价格归属、价格模板、费用组成、阶梯规则、计划用水方案综合计算；价格配置缺失、费用组成不完整或规则冲突时，必须阻断生成并返回失败原因。
- 账单生成结果统一由 `biz_charge` 与 `biz_charge_detail` 承接：主表表达客户、账期、应收日期、账单总金额和主状态，明细表表达费用组成、用量、单价和明细金额。
- 特殊开账、无码客户开账、罚款类开账等非标准来源，仍纳入同一营业账主明细模型承接，通过来源类型、业务类型、依据说明和操作留痕区分，不单独扩展平行账表。
- 审核通过后的营业账方可进入收费、催缴和发票流程；后续链路只能消费已生成账单结果，不反向改变本接口的生成边界。
- 远传抄表数据可由 `SYS-006` / IoT 能力提供采集支撑，但账单生成仍归属 SYS-002。

2.4.4 开账触发与结果表达

- 触发前提：正式口径下可按抄表批次、指定客户范围或指定抄表任务范围组织生成；当前 backend 已落地的入口为按 `readingDataIds` 批量复核并开账，对应 `ChargeController.generateCheckChargeBatch`、`ChargeServiceImpl.generateCheckChargeBatch` 与 `ReadingDataServiceImpl.batchReCheckReadingData`。
- 规则来源：价格归属决定客户适用的价格口径；价格模板、阶梯规则、费用组成和计划用水方案共同决定营业账主表金额与明细拆分方式。
- 当前承接证据：`ChargeServiceImpl.generateSingleChargeWithCache` 成功路径会写入 `biz_charge` 主表、循环写入 `biz_charge_detail` 明细，并回写抄表数据开账状态，说明现有实现已具备“按抄表数据 ID 生成营业账主明细”的 backend 基础。
- 结果表达：正式 `IF-REV-005` 应返回成功清单、失败清单、生成汇总及主明细级结果；当前 backend 返回仍为“本次复核成功X条 / 本次开账成功Y条”的字符串拼接，尚未形成结构化成功/失败结果对象。
- 阻断与限制：价格模板不存在、费用调整配置缺失、结算方式非 `ACTUAL_USAGE` 等场景当前会直接阻断单条生成；其中固定水量、按人口数、最低消费等非实际水量结算方式仍未纳入当前实现。
- 下游边界：`REV-002` 只负责生成营业账结果并交由后续审核/收费链路消费，不在本章节扩展收费核销、发票申请或催缴执行细节。

2.4.5 核心数据

- `biz_meter_book`：册本与抄表计划。
- `biz_meter_read`：抄表任务状态/执行状态。
- `biz_reading_data`：抄表数据。
- `biz_last_reading`：上次抄表结果。
- `biz_reading_logs`：抄表日志与过程留痕。
- `biz_meter`：计量水表主档引用。
- `biz_charge`：营业账主表。
- `biz_charge_detail`：营业账明细。
- `biz_price_category`：价格归属。
- `biz_price_template`：价格模板。
- `biz_price_adjustment_snap`：调价快照。
- `biz_price_tier_adjustment`：阶梯规则。
- `biz_cost_component`：费用组成。
- `biz_water_use_scheme`、`biz_water_use_scheme_tier`：计划用水方案与阶梯。

2.4.6 迁移补充（旧系统承接）

2.4.6.1 特殊开账

- 旧系统支持在非标准客户、无码客户或罚款类场景下直接开账。
- 新系统仍以 `biz_charge`、`biz_charge_detail` 作为开账结果承载对象，不建议额外平行建设“特殊开账账表”。
- 迁移与新建场景均需保留特殊开账来源、业务类型、经办人、依据说明、打印状态与后续收费关联，避免与普通抄表开账混淆。

2.4.6.2 开账记录迁移

- 旧系统“开账记录”是历史查询与账务核对的核心入口，必须纳入迁移最小保留集。

- 迁移后的开账记录应至少支持按站点、账务年月、册本、客户、抄表员、欠费状态查询，并支持统计水量、总金额及费用构成。
- 对已收费、已作废、销户拆表、特殊开账等旧状态，不要求照搬旧表结构，但必须保留可对照的新状态映射关系。

2.4.7 接口映射

- IF-REV-004：抄表数据提交与异常标记。
- IF-REV-005：账单生成与开账结果返回。
- IF-METER-004：远传抄表数据接收后进入开账流程。

2.4.8 落地边界

- **已落地**：册本、抄表数据、上次抄表、抄表日志、营业账主明细、价格模板与阶梯规则。
- **部分落地**：部分异常场景对象可能仍通过状态字段和日志表承载，而非全部拆成独立业务表。
- **文档先行**：个别精细稽查、轨迹、下载同步对象当前未在本轮映射中确认为独立表。

2.5 REV-003 营业收费

2.5.1 功能说明

支持柜台收费、预存款/余额抵扣、线上缴费回写、柜面扫码、营业网点收费及收费凭证管理，统一承接营收账单的核销处理。

2.5.2 业务流程

```
flowchart TD
    A[查询客户及待缴账单] --> B[选择账单与核销方式]
    B --> C[选择支付渠道]
    C --> D[支付方式]
    D --> E[柜台现金/POS/扫码]
    D --> F[微信/支付宝/聚合支付]
    D --> G[预存款/余额抵扣]
    E --> H[更新营业账状态]
    F --> I[等待异步回调确认]
    G --> I
    I --> H
    H --> J[生成收费记录与凭证]
    J --> K[进入发票/对账流程]
```

2.5.3 关键规则

1. 一次缴费可对应多个账单或账单明细的组合核销。
2. 收费记录必须保留渠道、流水号、网点、操作员、终端信息。
3. 线上支付必须以回调或查询确认结果为准，不得以发起状态直接记账。
4. 支付能力由 SYS-009 提供，SYS-002 负责账单核销与业务状态回写。
5. 当前实现侧已确认 PayCeb 的欠费查询、缴费处理基础闭环可用，但代理收费对账仍为预留能力；正式文档不得将实时收费对账写成已闭环能力。

2.5.4 核心数据

- `biz_charge`、`biz_charge_detail`：待缴与已缴账单主明细。
- `biz_collection`：托收/代收主表。
- `biz_withholding`：代扣/托收主表。
- `bk_transaction`：渠道交易流水。
- `bk_transaction_callback`：支付回调记录。
- `bk_transaction_exception`：支付异常记录。

2.5.5 迁移补充（旧系统承接）

2.5.5.1 柜台结账

- 旧系统将“柜台收费”和“柜台结账”拆分为两个菜单，结账阶段包含未结/已结查询、结账红冲、追加抄表和打印动作。
- 当前设计可继续采用统一收费核销模型，但必须补出“收费记录 → 班结结果 → 打印/红冲/查询”的业务闭环，避免柜面日终处理缺口。
- 迁移时需保留结账时间、结账人、网点、收费汇总口径和结账后红冲痕迹，保证财务对账与审计连续。

2.5.5.2 账单打印服务

- 旧系统存在账单打印、补打、打印次数控制与打印记录查询能力。
- 新系统不要求单独建立打印主模块，但必须明确打印模板、补打权限、打印状态与打印留痕的承接关系。
- 对历史账单迁移，应允许基于账单主明细和打印配置恢复打印视图，避免割接后只能查账不能补打。

2.5.5.3 红冲记录

- 旧系统支持红冲记录查询、导出与明细展开，是收费差错追溯的重要入口。
- 当前设计可将红冲视为收费核销后的修正场景，不强制要求独立实体表，但必须提供历史只读查询口径。
- 红冲迁移最小保留信息应包括原收费记录、红冲时间、红冲金额、原因、经办人、关联账单和后续账务状态。

2.5.6 接口映射

- `IF-REV-006`：创建收费记录、执行账单核销并回写状态。
- `IF-CS-003`、`IF-CS-007`：客户渠道与柜面扫码支付场景复用收费核销链路。

2.5.7 落地边界

- **已落地**：营业账主明细、交易流水、回调、异常、托收/代扣主对象。
- **部分落地**：柜台班结、部分收费汇总类对象可能通过业务流程与报表实现，不一定存在独立表。
- **文档先行**：部分红冲、实时收费汇总类台账暂不表述为已确认独立实体表。

2.6 REV-004 账务处理

2.6.1 功能说明

REV-004 一期仅覆盖水量调整、金额调整、退款、冲正、坏账申请五类场景，统一挂靠 `IF-REV-007` 作为账务处理入口，目标是在既有正式文档体系内先收敛范围、承接口径、留痕要求与审批边界。

本阶段按“共性能力先统一、场景能力再分批”组织：先统一账单承接、原交易校验、结果表达、操作留痕与审批边界，再分别展开五类场景。违约金减免、分账调整、价差调整、跨周期水量、预存退款细表等内容仅作为旧系统迁移语义或后续扩展参考，不作为一期新增独立范围。

2.6.2 业务流程

```
flowchart TD
    A[发起账务调整申请] --> B[校验账单状态与权限]
    B --> C{是否通过}
    C -->|否| D[驳回并记录原因]
    C -->|是| E[执行重算或退款冲正]
    E --> F[更新账单与明细状态]
    F --> G[写入操作日志与审批留痕]
    G --> H[返回处理结果]
```

2.6.3 关键规则

1. 一期场景严格限定为水量调整、金额调整、退款、冲正、坏账申请，不扩展到其他接口族或独立账务台账重构。
2. 所有场景均以 `biz_charge` / `biz_charge_detail` 为主承接对象，并通过 `biz_operat_log` / `biz_operat_log_detail` 记录处理依据、前后变化和责任归属。
3. 退款、冲正必须联动 `bk_transaction`、`bk_transaction_callback`、`bk_transaction_exception` 等原支付流水及渠道状态校验，不允许仅依据账单状态直接处理。
4. 接口结果统一返回 `resultStatus`、`writeBackStatus`，其中 `resultStatus` 表示处理结论，`writeBackStatus` 表示账单状态回写结论，两者不得混用。
5. 审批相关内容一期仅保留 `approvalRequired`、`PENDING_APPROVAL` 与审批边界说明，不展开完整 BPM 流程、节点、流转规则或审批回写实现细节。
6. 对于当前未见明确独立实体表的特账、跨周期水量、退款账等对象，文档以“业务处理场景”表述，不强行落为已实现表。

2.6.4 核心数据

- `biz_charge`、`biz_charge_detail`：账务调整的核心对象，承接调整前后账单主明细状态。
- `bk_transaction`、`bk_transaction_callback`、`bk_transaction_exception`：退款、冲正场景的原交易校验与异常追溯对象。
- 价格调整/优惠相关表：用于重算账单或差额追溯。
- `biz_operat_log`、`biz_operat_log_detail`：操作与变更留痕，记录字段差异、处理说明、附件依据与责任归属。

2.6.5 主要场景

场景	说明	控制要点
水量调整	更正异常水量	需复核原因、附件和原抄表依据
金额调整	更正账单金额	需记录依据、差异金额和审批边界
退款	退回客户支付资金或预存款	需校验原交易、退款余额与幂等性
冲正	修正误收/误核销记录	需关联原交易与账单状态
坏账申请	对长期欠费进行分类处理	需结合账龄、客户状态与审批边界

2.6.6 迁移补充（旧系统承接）

旧账务对象	当前承接方式	迁移口径
预存退款 / 预存退款详情	作为账务处理场景承接	保留申请单、原支付引用、退款结果与审批留痕
已销调整汇总 / 明细	作为已收费后修正场景承接	保留原账单、调整原因、前后差异、处理结果
价差调整汇总 / 明细	作为重算与差额修正场景承接	保留原价格口径、新价格口径、差额和生效时间
分账调整汇总 / 明细	作为费用组成重分摊场景承接	保留原分摊结果、调整后结果、责任人和审批链
账单-违约金减免	作为滞纳金修正场景承接	保留减免原因、减免金额、审批结果和生效时间
账单-呆坏账	作为坏账申请与生效场景承接	保留账龄、申请原因、审批结果、核销状态

1. P0 阶段不要求为每一类旧账务台账都新增独立实体表，但必须在业务对象和历史查询层形成可追溯闭环。
2. 旧系统精细台账迁移后至少保留：原单据标识、原账单标识、处理类型、处理原因、处理前后金额/水量、申请/审批/生效时间、经办人与附件依据。
3. 与支付、发票、渠道回调强关联的处理场景，必须保留与原交易、原发票、原收费记录的关联关系，避免后续对账和审计断链。

2.6.7 接口映射

- IF-REV-007：账务调整、退款、冲正、坏账等处理入口。
- IF-REV-006：与收费核销状态联动，确保调账后账单状态一致。

2.6.8 落地边界

- **已落地**：营业账主明细、操作日志、价格/方案相关重算支撑。
- **部分落地**：精细化账务对象更多表现为流程与场景，并未在 backend 中全部体现为独立表族。
- **文档先行**：特账、退款账、跨周期水量等对象保留业务语义，不宣称为已实现独立表。

2.7 REV-005 发票与税务处理

2.7.1 功能说明

负责发票业务闭环的业务接入与状态落账，覆盖后台发票申请、开票校验、`SYS-008` 异步协同、查询兜底、结果回写、账单关联、客户侧已开票电子发票查询/下载/推送，以及后台作废与红冲处理。

2.7.2 业务流程

```
flowchart TD
    A[后台提交发票申请] --> B[校验账单、客户开票信息、税率与限额]
    B --> C{是否满足开票条件}
    C -->|否| D[返回不可开票原因]
    C -->|是| E[生成申请单号并写入SUBMITTED]
    E --> F[调用SYS-008发起异步开票]
    F --> G[记录受理号并转为PENDING]
    G --> H[系统轮询或后台查询兜底]
    H --> I{开票结果}
    I -->|成功| J[回写SUCCESS并更新账单-发票关联]
    I -->|失败| K[回写FAIL并记录失败原因]
    J --> L[客户侧查询/下载/推送电子发票]
    K --> L
```

2.7.3 状态说明

- `SUBMITTED`：后台申请已受理，已完成本地校验并生成申请单。
- `PENDING`：已提交 `SYS-008`，等待异步结果或查询补偿结果。
- `SUCCESS`：已取得有效发票代码、号码或电子票地址，且账单关联已完成更新。
- `FAIL`：开票失败，需保留失败原因、最近查询结果与后续人工核查依据。
- `INVALID`：发票已作废，作为后续能力预留状态。
- `RED_INK`：发票已红冲，作为后续能力预留状态。

2.7.4 关键规则

- 一期采用“后台申请开票 + 客户侧查询下载推送”的入口模式，客户侧不直接发起开票申请。
- 发票申请以客户信息、已收费未开票账单、税率配置和开票限额为基础；原始单账单不支持直接任意部分金额开票。
- 个人与企业开票均通过客户开票信息与税率表完成合法性校验；如需多张发票，沿用拆账/分账后的账单分别开票口径。
- `SYS-008` 采用“异步申请 + 查询兜底”模式，成功状态不得被后续失败查询结果覆盖。
- 电子发票仅在 `SUCCESS` 且存在票据文件地址时允许客户侧下载或推送。
- 发票作废、红冲仍由 `SYS-008` 统一承接税控侧处理，`SYS-002` 负责后台触发入口、状态校验、查询补偿、结果落账与日志留痕；当前轮次按二期范围补齐 backend 实现入口。

2.7.5 后台申请入口与校验补充

- 后台支持营业收费员、财务人员按单笔或批量已收费账单发起开票申请。
- 申请时至少校验：账单收费状态、开票状态、客户开票信息完整性、票种适配性、开票限额、账单集合是否属于同一客户。
- 企业抬头场景重点校验纳税人识别号；电子发票场景重点校验邮箱/手机号；不满足条件时直接返回不可开票原因。

- 申请成功后生成申请单号并进入 `SUBMITTED/PENDING` 状态流转，失败校验场景不进入外部协同。
- 幂等控制以 `applicationNo` 或 `custId + chargeIds` 为主，避免相同账单组合重复申请。

2.7.6 SYS-008 异步协同与查询补偿

- 本地申请校验通过后，`SYS-002` 先写入 `biz_invoice` 申请记录，再向 `SYS-008` 发起异步开票请求，并记录 `sysRequestNo` 作为后续查询与回写的协同主键。
- `SYS-008` 返回“已受理”后，发票状态转为 `PENDING`；若仅完成本地受理但尚未拿到受理号，则保留 `SUBMITTED` 并进入待补偿查询状态。
- 查询补偿采用“回写优先、主动查询兜底”原则：`IF-EXT-007` 回写为首选结果来源，后台人工查询与系统定时补偿查询共用同一结果落账逻辑。
- 系统补偿查询至少保留最近查询时间、下次计划查询时间、累计查询次数、最近一次返回结果摘要与异常原因，便于问题追踪和人工核查。
- 后台按申请单号或受理号触发查询时，应同步刷新查询上下文；查询仍未取得终态时，仅更新补偿上下文，不得伪造成功或失败结论。

2.7.7 终态保护与异常核查

- `SUCCESS` 属于正常开票闭环终态，一旦已取得有效发票代码、发票号码或电子票据地址，后续失败查询结果不得覆盖成功状态。
- `FAIL` 仅在 `SYS-008` 明确返回失败结论或多次补偿查询确认失败时写入，并同步保留失败原因、最近查询结果与人工核查依据。
- 当后台人工查询、系统补偿查询与外部回写结果不一致时，应以最新有效外部凭据为准，并记录差异说明，不允许直接覆盖既有成功票据关键信息。
- 每次提交协同、主动查询、状态变更和异常分支均应写入操作留痕，确保能够追溯责任人、触发来源、状态前后值与异常说明。

2.7.8 结果回写、账单关联与客户侧消费

- `IF-EXT-007` 回写成功结果时，除更新 `biz_invoice.invoice_status`、`invoice_code`、`invoice_number`、`file_url` 外，还应同步刷新账单快照、账单关联状态与推送状态初值。
- 账单关联以 `biz_invoice.charge_id + charge_ids_snapshot` 记录本次开票覆盖账单集合，并同步把对应 `biz_charge.invoice_state` 更新为“开票完成”，保留失败原因与开票时间，便于账单明细、收费记录和客户侧结果统一展示。
- 客户侧查询仅允许按本人 `custId` 访问已存在的发票记录；可通过 `invoiceId`、`applicationNo` 或 `sysRequestNo` 定位，但都必须命中同一客户名下记录。
- 客户侧下载与推送前必须校验发票状态为 `SUCCESS` 且 `fileUrl` 非空；不满足条件时仅返回不可下载/推送原因，不得伪造文件地址。
- 推送动作应记录推送渠道、目标邮箱/手机号与结果状态；成功后更新 `pushStatus=PUSHED`，失败则写入 `FAIL` 并保留失败原因供人工处理。

2.7.9 核心数据

- `biz_invoice`：发票主记录。
- `biz_invoice_taxrate`：税率配置。
- `biz_cust_invoice`：客户开票信息。

2.7.10 迁移补充（旧系统承接）

- 旧系统数据字典中存在“发票明细表、营业账开票表、开票配置表”等更细粒度对象。
- 当前正式设计已明确主承接对象为 `biz_invoice`、`biz_invoice_taxrate`、`biz_cust_invoice`，但迁移时不能忽略旧开票明细和账单关联关系。
- P0 阶段建议先补三类迁移口径：账单与发票的关联关系、发票申请与结果回写记录、开票配置与税率的有效期；未确认已落地的细表对象仍按“历史只读或辅助映射”处理。

2.7.11 接口映射

- `IF-REV-008`：后台发票申请接口，负责单笔/批量申请、幂等控制与受理号生成。
- `IF-REV-009`：发票结果查询接口，负责后台按申请单号/受理号查询以及系统补偿查询。
- `IF-CS-004`：客户侧电子发票消费接口，负责已开票结果查看、下载、推送。
- `IF-EXT-007`：发票结果回写协同接口（由发票服务侧回传）。

2.7.12 落地边界

- 已落地**：发票主记录、税率配置、客户开票信息，以及一期正常开票闭环所需的后台申请、查询兜底、结果回写、账单关联与客户侧电子发票消费能力。
- 部分落地**：发票修改、开票过程留痕在后端中已有相关对象，但整套发票明细/批次类对象尚未全部确认。
- 二期补齐**：发票作废、红冲及其查询补偿仍由 `SYS-008` 统一承接，但当前轮次补齐后台触发入口、状态流转、结果回写与日志留痕，不再仅停留于文档预留。
- 文档先行**：发票明细、营业账开票关系等对象仍按设计能力描述，不表述为本轮已确认独立表。

2.8 REV-006 催缴与通知

2.8.1 功能说明

针对欠费账单按账龄、金额、客户类别等规则生成催缴任务，通过短信、微信、站内通知等方式触达客户，并回写催缴结果；本模块同时定义催缴与停复水/工单处置之间的联动边界与追溯关系，但不展开停复水内部处置流程。

2.8.2 业务流程

```
flowchart TD
    A[生成欠费客户清单] --> B[按策略分组催缴任务]
    B --> C[触发IF-REV-013生成催缴任务]
    C --> D[调用IF-EXT-008协同SYS-010]
    D --> E[接收发送结果回写]
    E --> F[更新催缴状态与后续策略]
    F --> G[按联动边界挂接停复水/工单处置]
```

2.8.3 关键规则

- 催缴策略以营业账状态、欠费金额、账龄分布、客户类别和渠道偏好为基础，支持按策略编码进行任务分组与频控。
- 自动催缴与人工催缴可并行；自动任务用于常规批量催缴，人工任务用于补发、核查或例外处置。

3. `SYS-002` 负责催缴对象筛选、任务生成、业务事件编号、结果承接与历史查询；`SYS-010` 负责短信、微信公众号、站内信等触达执行与结果回传。
4. `REV-006` 正式结果状态固定为 `PENDING`、`SUCCESS`、`FAIL`、`MANUAL_VERIFIED` 四态，其中 `MANUAL_VERIFIED` 仅用于外部结果未定或需人工核查补记的场景。
5. 停复水在本模块中仅定义联动触发条件、处置引用与追溯关系，不展开停复水内部审批、派工或现场执行流程。
6. 当前后端中部分催缴汇总、停水明细对象未确认独立落表，文档中保持保守描述，不误写为已确认在线主表。

2.8.3.1 催缴对象筛选、排除与频控边界

- `IF-REV-013` 任务生成前必须完成候选筛选，筛选最小维度为：欠费状态、欠费金额、账龄分组、客户类别、渠道偏好和策略编码。
- 候选对象必须以有效欠费账单为前提；以下场景不得进入正式催缴任务：已收费核销、已作废、已进入不允许催缴的处置流程、策略规则未命中。
- 触发类型按 `triggerType` 区分自动与人工；自动用于批量触发，人工用于补发、核查和例外补记，不改变正式接口编号与状态语义。
- 频控以“同客户/同策略/同渠道/同账期窗口”为最小拦截单元；命中频控时允许部分阻断，并返回被跳过对象及原因摘要。

2.8.4 核心数据

- `biz_charge`、`biz_charge_detail`：催缴对象来源。
- 催缴结果与通知日志：通过业务状态与消息结果联动留痕。

2.8.4.1 催缴对象与规则摘要

- `Reminder Candidate`：由欠费账单、客户类别、账龄分组、欠费金额、联系方式集合和命中策略编码组成，是催缴任务的输入对象。
- `Reminder Strategy`：定义账龄规则、金额规则、客户类别规则、渠道优先级、重复触达拦截窗口和是否触发后续处置关注。
- `Reminder Task`：一次正式催缴执行单元，至少包含 `taskNo`、`eventNo`、`strategyCode`、`channelType`、`triggerType`、`status` 和关联账单信息；正式业务接口编号固定为 `IF-REV-013`。
- `Reminder Result`：承接 `IF-EXT-008` 回传结果后由业务侧映射的正式四态结果，最少记录 `status`、`lastCallbackTime`、`failReason` 与回传摘要。
- `Disposal Link`：用于记录催缴结果与停水、复水、工单或人工跟进之间的关联引用，只承担追溯职责，不替代下游业务对象。

2.8.4.2 四态与人工核查边界

- `PENDING`：已生成任务并完成外部受理或等待外部终态回传，尚未形成业务终态。
- `SUCCESS`：外部触达结果明确成功，且业务侧已完成结果承接。
- `FAIL`：外部返回明确失败或业务判定失败，必须记录失败原因。
- `MANUAL_VERIFIED`：仅用于外部结果长期未定、人工核查补记或例外核销说明场景；必须留存核查说明与核查人。
- 人工核查是状态收口手段，不得用于绕过候选筛选、排除条件或频控约束。

2.8.5 迁移补充（旧系统承接）

2.8.5.1 催缴记录

- 旧系统支持催缴记录查询、导出和明细展开，记录中包含推送内容、号码、方式、结果等信息。
- 新系统可继续以消息协同结果和账单状态联动承接，但必须明确催缴记录查询口径，而不能仅保留“已发送/未发送”状态。
- 历史查询最少保留客户号、账期、催缴方式、发送对象、发送时间、执行结果、关联账单、关联处置引用等字段，并兼容四态结果或其历史映射值。
- 历史催缴记录按只读口径承接，作为查询与追溯来源，不反推为已确认在线主表。

2.8.5.2 停水记录

- 停水记录不是孤立账务对象，应由催缴结果、业务处置和现场执行工单共同形成闭环。
- 迁移后需支持按客户、站点、停水原因、停水时间、复水状态查询，并能追溯到对应欠费账单和工单执行结果。
- 正式设计只定义“何时建立联动、如何保存处置引用、如何追溯关联结果”，不在 `REV-006` 中展开停复水内部流程设计。
- 停复水关联以 `Disposal Link` 的处置引用承接，最少包含任务号、处置类型、处置引用号和建联时间。

2.8.5.3 预存短信

- 旧系统对预存款余额不足客户提供短信推送和发送记录查询。
- 新系统建议将其纳入催缴与通知统一策略，不再单建平行模型，但必须保留触发条件、发送内容、发送结果和补发记录。
- 该类记录与催缴记录一样，按历史只读口径承接，不表述为新增同名在线主表。

2.8.6 接口映射

- `IF-REV-013`：催缴任务生成、任务查询与结果承接接口，负责业务侧任务生成、四态状态维护和历史查询挂接。
- `IF-EXT-008`：消息协同结果回写接口（由 `SYS-010` 协同）。

2.8.7 落地边界

- **已落地**：以营业账为基础的欠费识别前提数据、催缴对象来源字段和消息协同边界约束。
- **部分落地**：催缴登记汇总、停水汇总等对象暂未在 backend 中确认独立表，当前以业务事件、操作留痕和历史查询口径承接。
- **文档先行**：复杂催缴台账、停复水统计和人工核查界面仅作为业务场景保留，不表述为 backend 已完成能力。

2.9 REV-007 统计分析

2.9.1 功能说明

提供营收、抄表、收费、欠费、渠道、客户等多维度统计查询能力，为经营分析、业务监管和迁移核查提供统一的数据口径支撑；本模块以经营查询为主，不扩展到预测分析、专题大屏或独立 BI 平台实现。

2.9.2 关键设计

1. 统计查询按“主题 + 维度 + 指标”三层口径组织，避免仅以报表名称堆砌需求。
2. 主题范围至少包括营收汇总、收费与实收统计、欠费规模与账龄统计、客户结构统计、渠道交易统计、抄表完成率统计以及营收相关业务概览类摘要。
3. 查询维度至少包括时间区间、账期、营业所/片区、客户类别、渠道、客户/账户、状态等，并支持必要的分组汇总。
4. 指标口径需明确区分应收金额、实收金额、欠费余额、账单数、客户数、交易笔数、渠道占比、抄表完成率等相近但不等价的统计概念。
5. 导出与查询结果受数据权限控制；导出属于查询扩展能力，但不在本轮展开具体导出实现细节。
6. 重点查询可按聚合视图、汇总口径或预聚合结果承接，但不将未确认存在的统计表、专题分析表或离线数仓对象写成已实现事实。

2.9.3 核心数据

- 客户维度：biz_cust、biz_account。
- 抄表维度：biz_meter_book、biz_reading_data、biz_last_reading。
- 账务与收费维度：biz_charge、biz_charge_detail。
- 收费与交易维度：biz_collection、bk_transaction。
- 渠道维度：bk_transaction、bk_payment_channel。
- 组织与权限维度：system_dept、数据权限控制结果。

2.9.3.1 统计主题与口径摘要

- Statistics Theme：按经营主题组织查询，至少覆盖营收、收费、欠费、客户、渠道、抄表完成率和必要的业务概览。
- Statistics Dimension：按时间、账期、营业所/片区、客户类别、渠道、客户/账户、状态等条件筛选或分组。
- Statistics Indicator：至少明确应收金额、实收金额、欠费余额、账单数、客户数、交易笔数、渠道占比、完成率等指标含义和单位。
- Aggregation Source：统计结果以现有在线主数据聚合、视图或汇总口径承接，不反推为已存在独立统计表族。

2.9.4 接口映射

- IF-REV-010：营收、收费、欠费、渠道、客户等统计查询接口，承接主题查询、维度筛选、指标汇总和权限/导出边界。

2.9.5 落地边界

- 已落地：主要统计源数据在客户、抄表、账单、收费、交易和渠道等领域均已具备，足以支撑经营统计口径设计。
- 部分落地：部分统计结果可能依赖聚合视图、汇总查询或报表层实现，当前未见明确的 backend 统计控制器或独立统计模型入口。
- 文档先行：预测类、专题分析类深度模型、BI 大屏和独立数仓能力暂不写成后端已实现能力。

2.10 REV-008 代收与银行业务

2.10.1 功能说明

支持银行代收、银行代扣、实时收费、夜间批量扣款、对账与结算处理，是 SYS-002 面向 `SYS-009` 支付与银行结算能力的业务承接模块。

2.10.2 业务流程

```
flowchart TD
    A[生成代扣批次] --> B[校验签约与待扣账单]
    B --> C[调用IF-REV-011下发批次]
    C --> D[SYS-009对接银行处理]
    D --> E[回写扣款结果]
    E --> F[执行对账与差异识别]
    F --> G{差异是否已处理}
    G -->|否| H[进入人工补偿]
    G -->|是| I[确认结算并更新状态]
```

2.10.3 关键规则

1. 渠道、路由、接口配置、签约、交易、回调、异常、对账、结算形成完整银行业务链条。
2. 实时收费场景由渠道交易流水驱动账单核销，批量代扣场景由签约关系与批次处理驱动。
3. 对账结果区分一致、长款、短款、失败待处理等状态，支持差异追踪与人工补偿。
4. 国密报文、批量文件、标准 API 等技术细节由 `SYS-009` 承载，SYS-002 保留业务规则与状态协同。
5. 当前 backend 已确认 `BankWithholding` 六条银行入口（客户状态查询、送盘、送盘状态查询、取消送盘、回盘、回盘状态查询）已形成最小实现态闭环；`BankCollection` 平行链路、对账与结算协同仍以“部分实现或文档先行”表述，不得统一写成已闭环能力。
6. 银行代扣文件传输配置按“默认规则 + 银行通道覆盖 + 租户覆盖 + 租户-银行通道覆盖”建模，命中优先级固定为 `TENANT_CHANNEL > TENANT > CHANNEL > DEFAULT`。
7. 目录字段至少区分 `send/back/reconcile/archive/localTemp` 五类阶段；上层覆盖未完整定义时按字段级回退，不允许生成空路径。
8. 路径模板仅允许 `{tenantId}`、`{companyId}`、`{channelCode}`、`{yyyyMMdd}`、`{yyyyMM}`、`{batchNo}`、`{fileName}` 七个固定变量；命中未声明变量或缺少变量取值时立即阻断文件动作。
9. `BankWithholding` 在送盘创建时固化 `sendProtocol/sendDir/sendFilePath` 与 `backProtocol/backDir`，配置切换仅影响新发起文件动作，已落库批次继续沿用原解析结果。

2.10.4 核心数据

- `bk_payment_channel`：支付渠道。
- `bk_channel_api_config`：渠道接口配置。
- `bk_channel_route_rule`：渠道路由规则。
- `bk_withholding_agreement`：代扣签约。
- `bk_withholding_batch`、`bk_withholding_item`：代扣批次与明细。
- `bk_reconcile_batch`、`bk_reconcile_diff`：对账批次与差异。
- `bk_settlement_batch`：结算批次。
- `bk_transaction`、`bk_transaction_callback`、`bk_transaction_exception`：交易、回调、异常。
- `biz_collection`、`biz_withholding`：代收/代扣业务主对象。

2.10.4.1 文件传输配置与审计补充

- 环境默认规则通过 Spring profile + Nacos 承接，不在仓库样例中写入真实密码、私钥或证书。
- `bk_channel_api_config` 使用专用 `apiType=FILE_TRANSFER_CONFIG` 承接文件传输覆盖配置，`extParams` 记录作用域、业务类型、协议、连接引用和五类目录字段。
- `bk_withholding_batch` 固化送盘/回盘目录与协议，`bk_reconcile_batch` 固化对账阶段最终 `protocol/dir/filePath/fileName`，用于审计与问题回放。

2.10.5 迁移补充（旧系统承接）

2.10.5.1 银行托收

- 旧系统“银行托收”菜单重点承接托收送盘、托收信息查询和托收结果回看。
- 当前设计已形成 `biz_collection` + `bk *` 渠道模型，迁移时应补出“旧托收菜单 → 托收批次/交易/回盘结果”的映射，而不是按旧菜单名平移建模。
- 旧托收历史记录应至少保留送盘批次、客户范围、送盘结果、回盘结果和账单核销结果。

2.10.5.2 实时收费查询与对账

- 旧系统“实时收费”更偏运营查询和渠道对账入口，不只是支付成功回写。
- 当前建议以 `bk_transaction*` 作为主承接对象，并补充按结算日期、银行/渠道、收费结果、差异状态查询和导出能力说明。
- 对旧“实时收费汇总/日志/明细”对象，P0 阶段先按历史只读查询口径保留，不误写为当前已落地的独立主模型。

2.10.5.3 当前实现对齐说明

- `PayCeb` 路径已具备欠费查询、缴费处理、流水唯一性校验和交易日志留痕，可作为实时收费基础闭环的实现证据。
- `BankWithholding` 路径已具备签约、解约、客户状态查询、送盘、送盘状态查询、取消送盘、回盘、回盘状态查询及对应交易留痕的实现证据，可作为代扣最小实现态闭环依据。
- `BankCollection` 路径当前仍仅能确认签约、解约与协议/交易日志处理具备实现证据。
- 对账、结算、真实银行文件解析、SFTP/文件通道联调和运行态样本补证当前仍未闭环，正式文档应继续保留为后续完善项。

2.10.6 接口映射

- `IF-REV-011`：代扣批次、对账与结算协同入口。
- `IF-EXT-001`：银行代扣批次下发与回盘协同。
- `IF-EXT-003`：银行实时收费查询、缴费与结果确认协同。

2.10.7 落地边界

- **已落地**：渠道、路由、交易、回调、异常、代扣/托收签约、解约，以及 `BankWithholding` 的客户状态查询、送盘、送盘状态查询、取消送盘、回盘、回盘状态查询和对应日志留痕等主对象已具备明确实现证据。
- **部分落地**：`BankCollection` 批次、明细、送盘、回盘、状态查询、差异台账和后台资源管理入口已具备对象或骨架，但不等同于银行协同闭环全部完成；`BankWithholding` 的真实文件解析、异常补偿和运行态联调证据仍待补齐。

- **文档先行**：夜间批量代扣调度、完整对账处理、结算确认、扩展银行台账等内容不得在当前阶段写成已完成能力。

2.11 REV-009 业务参数配置

2.11.1 功能说明

负责营收域的价格参数、客户编号规则、页面配置、打印与渠道相关业务参数配置，为客户、开账、收费、发票、催缴等模块提供统一配置支撑。

2.11.2 关键设计

1. 业务参数按租户、单位、片区、业务类别分层管理。
2. 价格体系、客户编号规则、页面字段配置、打印与通知参数统一归口维护。
3. 配置变更应具备版本化、操作留痕与生效范围控制。

2.11.3 核心数据

- `biz_parameter_settings`：业务参数配置。
- `biz_page_settings`、`biz_page_settings_detail`：页面配置。
- `biz_price_category`、`biz_price_template`、`biz_template_dept_rel`：价格归属与模板站点关系。
- `biz_cust_no_rule`：客户编号规则。
- `sys_wechat_app_settings`：微信/微网厅基础配置。

2.11.4 迁移补充（旧系统承接）

- 旧系统后台存在“页面配置、业务字段、微信参数、打印维护”等运营配置入口。
- 当前建议统一按业务参数、页面配置、渠道参数与打印参数归口承接，不新增“微客服后台配置”并行主文档。
- 迁移时需明确三类配置映射：客户/业务办理字段展示与校验规则、微信/微网厅基础参数、打印模板与补打策略；未确认已实现的高级灰度能力继续按“文档先行”处理。

2.11.5 接口映射

- `IF-REV-012`：查询与维护价格模板、业务参数、页面参数配置。
- `IF-UP-004`：统一平台参数字典能力协同，为营收域参数提供基础字典支撑。

2.11.6 落地边界

- **已落地**：业务参数、页面配置、价格归属与模板站点关系、客户编号规则等核心配置对象。
- **部分落地**：部分打印模板、通知策略等参数由统一平台或外部渠道参数共同承载，营收域仅保留业务侧映射。
- **文档先行**：参数灰度发布、多版本并行生效等高级治理能力当前仅保留设计语义，不宣称为独立实现模块。