

AgentSQL

智盾

使用手册

v4.3.0

agentsql.cn

v4.3.0 使用文档

目录

v3.9.0 Oracle 与 SQL Server 受控读取

v3.8.0 ClickHouse 预算与 Milvus 租户过滤

v3.7.0 MongoDB 与 Sentinel 受控读取

v3.2.0 SQLBot 接入 /sqlbot

能力目录与交付诊断

v2.9.0 策略包治理与外部审计见证

v2.8.0 Milvus 受控向量检索

v2.7.0 ClickHouse 受控分析查询

v2.6.0 Redis / Valkey 受控键读取

1. 总览 /

接入向导 /onboarding（v1.9.0）

v2.4.0：openGauss 7.0.0 LTS 指定实例受控只读

v2.3.0：SQL Server 2025 指定实例受控只读验证

v2.2.0：Oracle Free 指定实例受控只读验证

v2.1.0：崖山指定实例受控只读验证

v2.0.0：固定双网关与 PostgreSQL 列策略登记

2. 审计 /audit

- 筛选参数
- 导出

3. 演示台 /playground

- 静态评估：所有普通部署都有
- Live Demo：仅 Demo 模式

4. Agent /agents

5. 数据源 /datasources

- YashanDB 客户端配置

6. 权限 /policies

- 对象与动作
- 列级白名单

row_filter
临时只读授权的使用边界 /task-read-grants（v1.0.0）
任务授权的申请、审批与领取（v1.0.0）
MCP 用户身份委托（v1.0.0 可选）
7. 规则 /rules
内置规则目录
当前可编辑能力
阈值与动态依赖
8. 审批 /approvals
9. 脱敏 /mask-rules
类型与算法能力矩阵
三档作用域与选择建议
配置 range 数值分桶
配置 range 日期截断
API 字段与完整更新语义
精确匹配与 fail-closed 安全兜底
敏感发现工作流
校验与冲突报错
range 行为与边界
脱敏密钥轮换与等值边界
已知限制与建议
10. 配置参考
server
store
defaults
theme
demo
redaction
环境变量
11. 运维操作
agentsqlctl 公开子命令
health

升级、回滚与备份

性能口径

12. 当前版本能力边界

在线安装包升级预演（与 schema 检查不同）

v4.2.0 离线交付与诊断

v4.2.0 原生问数

v4.3.0 图表、仪表板与嵌入应用

AgentSQL 使用手册

语言 / Language: 简体中文 | English

> 现行指南：适用于 v4.3.0 及兼容的 4.3.x 维护版本。历史引入版本与验收记录保留原始事实；日常入口见文档索引。

> 本文说明 AgentSQL v4.3.0 的功能与操作。安装或拉取 v4.3.0 前，先在 GitHub Releases 核对正式资产与签名，并确认对应 GHCR 镜像可以匿名拉取；源码仓库为 `github.com/cuipengdba/agentsql`。

本手册按控制台主要菜单说明 v4.3.0 的能力和操作。首次使用请先完成 快速上手；MCP 客户端配置见 接入指南；版本范围与验收状态见 v4.3.0 发布说明和设计与验收记录；「设置与集成 → 通知设置」的 Webhook、Syslog 与安全边界见 通知外发指南。

控制台登录页和登录后右上角的语言按钮可在中文和英文之间切换，选择会保存在当前浏览器；`?lang=en` 或 `?lang=zh` 可指定首次打开语言。左侧菜单在有效视口高度不足时可以独立滚动，聚焦菜单后也可用键盘访问底部选项。语言切换仅改变界面文案，不修改 SQL、查询结果和审计记录。完整英文文档可通过文首语言链接访问；版本能力以对应验收记录为准。

v3.9.0 Oracle 与 SQL Server 受控读取

- 对指定 Oracle 26ai Free 选择 `verify-full`、填写绝对路径 CA 文件，并使用与证书名称匹配的连接主机名。网关每次新建连接均验证证书。Oracle 钱包、mTLS 与 RAC 不在本次验收范围。
- 仅在专用非系统只读账号与已验证的 `PLAN_TABLE` 就绪后，在 `read_control_json` 启用 `{"oracle":{"explain":true}}`。获准的窄 SELECT 随后进行有界计划预检；无法取得计划或计划无效时失败关闭。既有 Oracle 数据源默认仍不启用计划预检。
- Oracle 与 SQL Server 都使用 MCP `query`。SQL Server 保持严格证书校验和限定 SELECT/SHOWPLAN 权限，轮换及恢复后分别核对允许、拒绝、脱敏和持久审计 ID。公网 Demo 仍为 PostgreSQL/MySQL 合成数据。见发布说明、升级指南和验收记录。

v3.8.0 ClickHouse 预算与 Milvus 租户过滤

- 在数据源页面选择 ClickHouse 或 Milvus，填写可选的 `read_control_json`。ClickHouse 预算可收紧读取行、字节和内存；启用前由 DBA 为只读角色配置受限的逐查询设置。Milvus 指定租户字段和允许等值过滤的字段，集合中的租户值须与 AgentSQL 租户 ID 一致。

- 为 Agent 精确授权表或 collection，以及输出、过滤和 Milvus 租户字段。使用 MCP `query_analytics` 或 `search_vectors` 验证允许、拒绝、脱敏、审计和跨租户隔离；公网 Demo 不包含这些原生数据库。配置与限制见设计、升级指南和验收记录。

v3.7.0 MongoDB 与 Sentinel 受控读取

- 在数据源页面设置仅含拓扑和批准索引元数据的 `read_control_json`。MongoDB 副本集须明确列出三个受信成员与集合索引；Redis/Valkey Sentinel 须列出三个 Sentinel、允许主节点及独立 ACL 用户名。Sentinel 密码由详情页显示的 `sentinel_password_env` 对应环境变量注入每个网关，不写入 JSON。
- 为 Agent 精确授权 MongoDB 集合和所需字段，或 Redis/Valkey 逻辑库、Hash 键与每个字段。通过 MCP `find_documents` 或 `read_hash_fields` 调用，核对允许、拒绝、脱敏与审计 ID；错误证书、未知成员、索引缺失、ACL 撤销及审计故障必须拒绝。公开 Demo 不含这些原生拓扑，配置格式和已测边界见NoSQL 支持范围、升级指南与验收记录。

v3.2.0 SQLBot 接入 `/sqlbot`

在「设置与集成 → SQLBot 接入」配置普通登录用户、工作空间、数据源名称到既有 AgentSQL 人员、只读 Agent 和 PostgreSQL 或 MySQL 数据源的精确映射。MySQL 发现 schema 固定为该数据源的数据库名。工作空间及用户 ID 按字符串填写；绑定创建、修改和连接凭据轮换要求同时拥有策略、人员和数据源管理权限。人员必须保有查询执行权限，Agent 必须处于有效只读状态。

连接凭据仅在创建或轮换时显示一次。将它和独立的 Agent Key 分别存入 SQLBot 服务端环境变量；导出的 JSON 只含变量名。修改、禁用、删除或轮换绑定会使先前签发的 SQLBot 人员令牌在后续请求失效；停止该 Agent 的全部访问还需撤销 Agent Key。配置诊断不代表已执行真实业务查询。

安装固定 SQLBot v1.10.2 的适配器后，普通问数先过滤模型可见的表与字段，再通过网关执行。拒绝不触发自动改写重试；用户可在 SQLBot 的 `/api/v1/agentsql/console.html` 页面明确提交只读审批、查看结果并单次执行批准的封存 SQL。完整配置、支持的 SQL 子集和信任边界见 SQLBot 接入指南。

能力目录与交付诊断

控制台「数据源」页按类型显示 Agent 能力层级、入口工具、限定范围和验收文档。`/api/v1/datasource-types` 的 `support` 字段来自同一服务端目录；MCP `list_datasources` 只为当前 Agent 可见的数据源附带同一说明。层级依次为 `controlled_sql`、`bounded_sql`、`controlled_read_tool` 和 `connection_preview`。最后一层仅表示连接或管理预览，不能作为 Agent 读权限使用。任何层级都不自动授权真实请求，仍需逐次通过身份、对象及列策略、执行、脱敏和持久审计。

升级前在隔离环境以目标配置运行 `agentsqlctl doctor --config /etc/agentsql/config.yaml --format json`。退出码 0 表示已配置存储的迁移历史就绪，2 表示有待执行迁移，1 表示配置或存储检查失败。该命令不迁移 schema、不创建缺失 SQLite 库，也不输出 DSN、密码或密钥；迁移历史就绪不等于业务数据完整或生产可用，仍要按部署指南备份与恢复。可用 `--format text` 查看简短的人读结果。

v2.9.0 策略包治理与外部审计见证

“策略包治理”页面导出当前租户策略包与摘要。编辑 JSON 后生成差异计划，可以填写最多 32 个精确 Agent/数据源/表案例查看离线表策略候选变化。发起人提交提案后，把提案 ID 交给另一位具有“策略管理”和“审计查看”权限的管理员审批；只有审批人可应用。应用时若源策略已变更，需重新导出、生成计划和审批。批准后直接改动策略会显示为 `drifted`。本版可变更的范围和 128 条策略、32 项变更上限见设计。

独立审计见证的简例（密钥和见证文件应放在数据库及网关数据卷之外）：

```
agentsqlctl audit checkpoint-keygen --private-seed-out /secure/audit-checkpoint-seed.hex --public-key-out /secure/audit-checkpoint-public.hex
agentsqlctl chain verify --domain traffic --config /etc/agentsql/config.yaml
agentsqlctl chain status --domain traffic --json --config /etc/agentsql/config.yaml
agentsqlctl audit report --format archive --out /secure/audit-export.zip --config /etc/agentsql/config.yaml
agentsqlctl audit checkpoint-create --in /secure/audit-export.zip --out /secure/audit-checkpoint.json --private-seed-file /secure/audit-checkpoint-seed.hex --chain-id traffic --head-seq "$HEAD_SEQ" --head-hash "$HEAD_HASH"
agentsqlctl audit checkpoint-verify --in /secure/audit-export.zip --checkpoint /secure/audit-checkpoint.json --public-key-file /secure/audit-checkpoint-public.hex --max-age 24h
```

先从 `chain status --json` 核对并设置 `HEAD_SEQ` 和 `HEAD_HASH`；不要直接复制未核实的值。后续见证需把上一份 `checkpoint-create` 输出的 `sha256` 作为 `--previous-sha256`；核验时从独立可信记录提供同一值给 `--expected-previous-sha256`。归档可包含筛选结果，因此链头值是经过 `chain verify` 单独核对后的**声明**，工具不会从筛选归档重算完整链头。该文件不是 WORM 或可信时间戳。

v2.8.0 Milvus 受控向量检索

在数据源中选择 Milvus，填写 `default` 数据库、2.6.25 实例的 gRPC TLS 端口、网关可读取的 CA 绝对路径、匹配证书的主机名及非 `root` 账号密码。数据库账号应只获目标 collection 的 Search 权限。为 Agent 建立精确 `default.collection` 表授权和同一对象的列白名单，白名单须包含向量字段及每个返回字段。MCP `search_vectors` 提交一个有限 float32 向量、显式标量字段和 1–20 的 `top_k`；网关核对维度与 COSINE 索引，并在脱敏、持久预检和终态审计成功后交付。原有管理预览与通用 SQL `query` 均不是此能力入口，参见v2.8.0 设计和候选验收。

v2.7.0 ClickHouse 受控分析查询

在数据源中选择 ClickHouse，填写指定数据库、原生 TLS 端口、可验证 CA、证书主机名及只读账号。数据库角色需固定 `readonly=1` 并限制查询时间、读行/字节、内存、输出行和线程。为 Agent 建立精确 `database.table` 表授权及同一对象的列白名单，白名单须包括返回列和过滤列。MCP `query_analytics` 只接受表名、显式列、String 列的等值条件与 1–100 行上限；网关仅查询 MergeTree 单表、逐次重验权限并在脱敏和持久预检/终态审计成功后交付。通用 `query` SQL 工具不是此能力入口。支持范围以v2.7.0 设计及完成后的验收为准。

v2.6.0 Redis / Valkey 受控键读取

在数据源中选 Redis 或 Valkey，配置固定 DB、严格 TLS、可验证 CA 和最小 ACL 账号；为普通 Agent 建立 `key` 策略，范围写作 `dbN:exact-key` 或有界 `dbN:prefix*`。MCP `read_key` 只填写 `datasource_id` 和精确 `key`。策略拒绝优先，SQL 表策略不能代替键授权。服务端固定 `TYPE` 和最多 64 KiB 的 `GETRANGE`，只接受 UTF-8 字符串；脱敏及持久预检、终态审计成功后交付。拒绝或错误时可用响应审计 ID 定位，不在审计记录保存原始键名或值。仅 Redis 8.2.9 / Valkey 8.1.10 TLS/ACL 单实例合成环境已验证，详见本版验收。

1. 总览 `/`

总览把近期决策、趋势和实时事件集中在一页：

- `allow`：规则与策略允许，执行类请求可以进入受控执行。
- `deny`：被身份、权限或规则拦截，不触达业务 SQL 执行。
- `warn`：允许执行，但附带需要关注的告警。
- `approve`：转人工审批，SQL 不执行。
- `error`：解析、评估、存储或执行链路发生错误。

```
请求 → 鉴权 → 解析 → 静态规则 → 必要的动态阶段 → 决策
                                                  ├─ allow   → 执行 → 脱敏 → 审计
                                                  ├─ warn    → 执行 → 脱敏 → 审计
                                                  ├─ deny    → 不执行       → 审计
                                                  ├─ approve → 不执行 → pending 审批 + 审计
                                                  └─ error   → 失败         → 尝试审计
```

趋势图用于观察决策数量随时间变化；实时事件流通过 `GET /api/v1/stream` 使用 SSE 推送新审计事件。反向代理该路径时必须关闭缓存、响应压缩和 buffering，并把空闲超时设为大于 25 秒。`server.event_stream_max_connections` 控制并发 SSE 连接上限。

注意：工具层前置拒绝可能尚未进入流水线，因此并非每一个失败都会出现在事件流或审计页。

接入向导 `/onboarding`（v1.9.0）

选择当前租户中的 Agent 和数据源，运行「连接与权限预检」。页面检查 Agent 是否启用或过期，使用已保存的数据库账号测试连接，并检查该 Agent 与数据源之间是否存在策略；不会执行 Agent SQL。策略存在时只显示“需验证”，因为策略数量不能证明某条 SQL 获得授权。

v1.9.0 在同一页面增加 Dify 1.17.1 / MCP Inspector 0.16.8 的接入说明、目标 SQL 离线解释和审计 ID 定位。先填写客户端可达的 AgentSQL 根地址，复制包含 `<Agent API Key>` 占位符的配置说明，再到客户端安全填写真实 Key；控制台不接收 Key。远端明文 HTTP 会传输 Key，应使用 HTTPS 或限定在隔离内网。Dify 需要显式 `MCP-Protocol-Version: 2025-06-18`，Inspector 0.16.8 初始化后自动添加，不能重复手工设置。Dify 容器里的 `127.0.0.1` 指向 Dify 自己，需使用容器可达地址。

离线解释只返回“明确拒绝”或“尚不能确认允许”，并可列出对象、匹配策略 ID 与静态规则；它不连接业务库，也不执行 SQL。最后在真实客户端分别发起允许与拒绝请求，将响应的正整数审计 ID 填入向导直达审计详情，核对 Agent、数据源、决策、规则及脱敏。没有审计 ID 时按错误码先查认证、协议或依赖故障。向导不能替代逐对象授权、执行链路或生产支持验收。

v2.4.0：openGauss 7.0.0 LTS 指定实例受控只读

创建 `opengauss` 数据源时必须填写能解析到目标服务器且与证书 SAN 一致的主机名，并配置 `verify-full` 及可信 CA；自签证书豁免、明文与其他 TLS 模式均被拒绝。数据库账号仅授予目标普通表的 SELECT，并保持默认事务只读。网关在每次新建物理连接时核验指定 LTS build 和只读状态。

MCP SQL 仅接受明确 schema、`FROM ONLY` 单张普通物理表、直接列投影、非恒真 WHERE 和 1–10000 整数 LIMIT 的 SELECT。视图、外表、JOIN、函数、聚合、系统目录、写入、B2 列级绑定和 B5 会话均不开放。先在接入向导预检，再用真实 MCP Inspector 或 Dify 发起允许与无权表拒绝调用，以审计 ID 核对脱敏与持久记录。指定实例的边界和故障测试见候选验收。

v2.3.0：SQL Server 2025 指定实例受控只读验证

本版在 SQL Server 2025 Developer x86_64 指定合成实例上，用校验本地测试 CA 的严格 TLS、只获目标表 `SELECT` 与数据库级 `SHOWPLAN` 的临时账号，验证真实 HTTP MCP、表策略 R010、手机号脱敏与持久审计。Inspector 与 Dify 完成允许/拒绝调用；撤销 SELECT、撤销 SHOWPLAN、业务库和独立审计库停机均按不交付原始结果验收。创建数据源时请使用可验证的服务器证书和专用最小权限账号；无权表的静态 R010 在网关终止，不以数据库权限错误替代。此结果不覆盖 Windows、Always On、其他版本、生产证书轮换、复杂 SQL、写入、B2/B5 或完整生产支持。详见候选验收。

v2.2.0：Oracle Free 指定实例受控只读验证

本版在 Oracle AI Database 26ai Free、`FREEPDB1`、服务端 `23.26.3.0.0` 的隔离实例中，用只获 `CREATE SESSION` 和目标表 `SELECT` 的临时账号验证真实 HTTP MCP、表策略 R010、手机号脱敏与持久审计。Inspector 与 Dify 的允许/拒绝调用、数据库 SELECT 撤权、业务库及独立审计库停机均用合成数据复验。Oracle 运行账号未验证 `PLAN_TABLE` 写权限，网关不预执行 `EXPLAIN PLAN`；商业版、TCPS、主备、复杂 SQL、写入与完整生产支持不在本版声明中。详见候选验收。

v2.1.0：崖山指定实例受控只读验证

本版在 YashanDB Enterprise Edition 23.4.1.109、`COMPAT_VECTOR=yashan` 的隔离实例中，使用临时最小权限账号验证真实 HTTP MCP、表策略 R010 拒绝、手机号脱敏和持久审计。MCP Inspector 与 Dify 的允许/拒绝调用、数据库 SELECT 撤权、业务库及独立审计库停机也经过合成数据测试。该结论限指定版本、模式和窄 SELECT 子集；写入、事务、EXPLAIN、服务端 TLS、主备、客户端取消/超时仍未验证或保持关闭。详见候选验收 与 YashanDB 方言边界。

已认证 Agent 对不可见数据源调用 `query` 或 `execute_write` 时，网关先写入无原始 SQL 的预检拒绝审计，再返回审计 ID；审计库故障时返回错误且不交付数据。`list_schema` 等非 SQL 工具与身份认证失败不属于该交通审计路径。

v2.0.0：固定双网关与 PostgreSQL 列策略登记

双网关 Kubernetes 部署只承载无状态 MCP 只读请求。两个固定副本共享 PostgreSQL metadata 和独立 audit 库，每个副本使用自己的 B2 实例标识；B5 跨请求事务、HTTP 状态会话与事件续传在此配置中关闭。运维人员须按双网关操作指南执行迁移检查、维护窗口切换、故障演练与回退。公开 Live Demo 为单网关合成数据环境。

具有策略管理权限的管理员可先创建 PostgreSQL 单张普通表的 `object_type: column`、`action: allow` 策略，使用该策略当前 `ETag` 调用 `POST /api/v1/policies/{id}/bind-postgres`，请求体为一条投影精确列清单的 `SELECT`。登记过程只分析 SQL 和数据库目录，不执行该 SELECT；成功后须在真实 MCP 客户端验证允许、拒绝、脱敏及持久审计。对象不符、额外引用列、多语句、视图、跨表、通配投影或原生扩展模式将拒绝；旧 `ETag` 返回 412。此 API 只覆盖本版闭合绑定模式的 PostgreSQL 单表范围。

2. 审计 `/audit`

审计页记录经过应用流水线的 SQL、对象、规则命中、决策、估算行数、返回行数、耗时与错误信息。它用于工程排障和行为追溯，不是法规级 WORM。

筛选参数

参数	含义
`time_start` / `time_end`	RFC3339 时间范围
`agent_id`	Agent ID
`datasource_id`	数据源 ID
`session_id`	调用会话 ID
`mcp_tool`	MCP 工具名
`decisions`	决策列表，逗号分隔
`stmt_types`	语句类型列表，逗号分隔
`risk_min` / `risk_max`	风险范围
`keyword`	SQL 或错误信息关键字
`object`	对象名

分页默认是 `page=1&page_size=20`，`page_size` 有效范围为 1–100。详情抽屉展示请求身份、SQL、对象、规则命中、各阶段耗时和最终决策，构成一次判断的证据链。

导出

导出沿用当前页面筛选条件，支持 NDJSON（JSONL）、CSV、PDF 与合规归档（ZIP），单次上限均为 10000 条；超过上限返回 HTTP 422，页面提示缩小时间范围或增加筛选条件。导出响应带 `Cache-Control: no-store` 与 `X-Content-Type-Options: nosniff`。

NDJSON（JSONL，面向机器）

- `Content-Type: application/x-ndjson`，文件名 `agentsql-audit.jsonl`，无 BOM，每行一条 JSON（snake_case）。
- 适合导入 SIEM、日志平台，或用脚本 / `jq` 进一步分析；字段超长（很长的 SQL、详情 JSON）时不受表格软件单元格长度限制。

CSV（面向人工合规报表）

- `Content-Type: text/csv; charset=utf-8`，文件名 `agentsql-audit.csv`，文件头带 UTF-8 BOM，Microsoft Excel / WPS 可直接双击打开而不乱码。
- 首行为固定 25 列中文表头：时间、审计ID、决策、风险等级、Agent ID、数据源、数据库类型、会话ID、对话ID、MCP工具、语句类型、命中对象、命中规则、原始SQL、归一化SQL、预估行数、返回行数、耗时毫秒、客户端IP、模型、动作、执行者类型、执行者ID、错误信息、详情JSON。
- 时间列统一为 UTC 的 RFC3339（带纳秒，末尾 `Z`）；数字列为纯数字，空值留空；含逗号、引号、换行的字段（如原始 SQL、详情 JSON）按 CSV 规范用英文双引号包裹，字段内引号转义为两个双引号，多行内容仍位于同一个单元格内。

- 公式注入防护：为避免在 Excel / WPS 中把以 `=`、`+`、`-`、`@`（含全角 `＝` `＋` `－` `＠`）开头、或开头就是 Tab / 回车 / 换行的文本误当公式执行，这类文本单元格会被前置一个英文单引号。这是降低表格软件公式注入风险的措施，并不对所有办公软件构成绝对保证；如需对原文做二次处理，请以 JSONL 导出为准，不要在表格软件中删除前导单引号后另存覆盖。
- 一致性与边界：导出按分页顺序读取，并非事务快照；导出期间审计数据若持续写入，极少数情况下可能重复或遗漏，需要严格一致的取证请直接查询审计库。CSV 属高敏数据，请按内部合规要求保存与传递，不要随意外传。
- 单元格长度：个别表格软件单个单元格上限约 32767 字符，超长 SQL / 详情在表格中可能显示异常，此类场景请改用 JSONL。

PDF 与合规归档

- PDF 由后端纯 Go 生成，不依赖浏览器或外部命令；包含筛选元数据、决策/动作分布与逐条事件摘要，长字段在 PDF 中有界截断，精确原文仍以归档内 JSONL 为准。
- 合规归档文件名为 `agentsql-audit-archive.zip`，固定包含 `audit.csv`、`audit.jsonl`、`report.pdf`、`summary.json` 和 `manifest.json`。manifest 为四个载荷记录媒体类型、字节数和 SHA-256。
- 可用 `agentsqlctl audit verify-archive --in <file.zip>` 离线校验成员全集、大小和摘要。校验通过不等于数字签名、来源认证、可信时间戳、WORM 保留或“这是最新副本”；攻击者若可同时重写载荷与 manifest，仍可构造自洽的新包。

导出行为留痕（audit.export）：每一次成功的审计导出都会在审计库追加一条管理操作记录，动作为 `audit.export`、执行者类型为 `admin`，用于追溯“谁在什么时间、从哪个来源 IP、以何种格式（JSONL/CSV/PDF/archive）、在什么筛选条件下、导出了多少条”。该记录只包含导出行为的元信息（格式、条数、筛选维度、来源 IP），不包含被导出的审计内容；使用关键字检索时只记录是否使用了关键字（`has_keyword`），不保存关键字原文。来源 IP 取服务器直连对端地址，不信任 `X-Forwarded-For` 等可伪造请求头。留痕为尽力而为：即使留痕写入失败也不影响已经成功的导出，仅在服务端记录告警，且不触发实时通知。`audit.export` 记录本身会出现在后续审计列表与导出中，并计入单次 10000 条导出上限（每次成功导出增加一行）。

3. 演示台 `/playground`

演示台有两套不同能力，不能混为一谈。

静态评估：所有普通部署都有

`POST /api/v1/playground/assess` 只解析 SQL 并运行静态规则，不连接业务库、不执行 SQL、不写审计。六条内置样例是：

场景	预期
正常点查	`allow`，无规则命中

场景	预期
无 `WHERE` 全表更新	`deny`，命中 R002
堆叠注入	`deny`，至少命中 R001、R006
`pg_sleep` 危险函数	`deny`，命中 R007
有 `WHERE`、无 `LIMIT` 的 MySQL 批量写	`approve`，命中 R202
残缺 SQL	`error`，解析阶段 fail-closed

界面中的 `stageSamples` 仅用于展示 allow/deny/approve/warn/error 的流程样本，不是第三套可执行场景。

Live Demo：仅 Demo 模式

`POST /api/v1/playground/run` 只在专用 Demo 模式注册。它使用固定合成数据源与演示身份，前五张卡会真实进入受控流水线，第六张是审计回看：

1. PostgreSQL 只读查询放行，返回不超过 5 行并写审计。
2. MySQL 无 `WHERE` 更新被 R002 与 Demo 只读屏障拒绝，不触库。
3. MySQL `phone`、`email` 结果脱敏。
4. PostgreSQL 大结果在 Demo 阈值下命中 R005，截断为最多 20 行。
5. 未授权 `internal_notes` 被 R010 拒绝。
6. 用最近 `audit_id` 查看审计详情，并跳到总览实时流。

普通生产部署没有 `/playground/run` 和 Live 控件。完整启动方法见 DEMO.md。

4. Agent `/agents`

Agent 是调用 AgentSQL 的独立身份。

字段	说明
`id`	唯一 ID
`name`	名称
`owner`	负责人
`level`	`readonly`、`dml` 或 `ddl`
`status`	`active` 或 `disabled`
`expires_at`	可选过期时间

能力档位：

- `readonly`：只允许查询。
- `dml`：允许查询与 `INSERT`、`UPDATE`、`DELETE`。
- `ddl`：包含 DDL 能力；具体 SQL 仍受策略与规则约束。

创建 Agent 时生成的 API Key 只展示一次。轮换 Key 后旧 Key 立即失效，不存在并行有效期。Agent 被禁用、过期、删除或轮换时，原 Key 都不能再通过认证。

不要把 `qps_per_agent` 理解为每个 Agent 的可编辑字段：它是全局默认值。连接上限 `conn_limit` 配在数据源上，也不是 Agent 字段。

控制台管理员登录 token 有效期为 12 小时；它与 Agent API Key 用途不同。

5. 数据源 `/datasources`

数据源保存业务库连接与执行上限。必须为 AgentSQL 创建独立最小权限数据库账号，禁止使用 superuser、owner 或 root。

字段	约束与默认值
`id`	必填，创建后不可修改
`name`	显示名称
`db_type`	从控制台已注册类型中选择；`highgo` 为 9.0.10 企业版 PG 模式限定只读类型
`host`	数据库主机
`port`	1–65535；PostgreSQL 默认 5432，MySQL 默认 3306，HighGo 默认 5866
`database`	数据库名
`username`	运行账号
`password`	新建必填；编辑留空保留原值；API 永不回显
`conn_limit`	默认 5，最小 1
`stmt_timeout_ms`	默认 5000
`row_limit`	默认 1000
`tls_mode`	PostgreSQL 协议路径和指定 Vastbase G100 Build 9 PG 模式实例可选择 `verify-full`（推荐）或 `disable`（仅限隔离测试）；SQL Server 使用其独立 TLS 模式；其他数据库按适配器实际能力判断
`tls_ca_file`	PostgreSQL `verify-full` 必填，为网关服务端可读取的 CA 文件绝对路径
`tls_server_name`	PostgreSQL `verify-full` 可选；不填时校验连接主机名，填写时校验证书中的该名称

保存后用「测试连接」验证。MySQL 驱动关闭 multi-statements；AgentSQL 本身也只接受单条 SQL。

选择 `highgo` 时，使用独立的最小 SELECT 权限账号。当前仅验证 9.0.10 企业版 PG 模式、单实例合成数据环境；连接逐次校验厂商版本、模式和只读事务状态，只接受显式 schema、单表直接列、非恒真 WHERE 与 LIMIT 的 SELECT。元数据发现可用，样本采集与 PostgreSQL B2/B5 能力均不开放；详见 v1.7.0 验收记录。

> **TLS 边界：**v1.7.0 在 KingbaseES V9R1C10 PG 模式隔离单实例验证了 PostgreSQL 协议路径的 `verify-full`、错误 CA/主机名拒绝和证书轮换；v1.8.0 在指定 Vastbase G100 Build 9 PG 模式实例完成独立验收。Vastbase 服务端需要启用 `ssl=on`，并为只读账号设置优先于普通规则的 `hostnossl ... reject`；该 Build 的 `require_ssl=on` 不兼容当前 pgx 协议 3.0 启动包。既有数据源空模式保留旧行为，不视为已验证 TLS。HighGo 的显式 TLS 选项仍被拒绝；v3.3.0 起仅指定 DM8 Pack3 构建可显式选择 `verify-full` 并提供网关本地绝对 CA 路径和证书主机名，错误证书或加密降级拒绝；其他数据库不能继承这些验证结论。证书部署见部署指南及v1.8.0 设计说明。

数据源密码使用 `AGENTSQL_SECRET` 加密。丢失或更换 SECRET 会导致已有密码无法解密；数据库/控制面备份与 SECRET 必须成对保存，详见 部署指南。

YashanDB 客户端配置

崖山 C 客户端再分发已获厂家口头授权；仓库无书面授权文件。当前 Linux 发行包和 GHCR 镜像包含独立客户端 23.4.7.100，正式发行验收核对 amd64/arm64 架构、库文件和哈希。自行从源码构建时，需取得同版客户端，使用 `CGO_ENABLED=1` 与 `-tags yashan` 编入 Go 驱动 v1.4.4，并配置客户端运行库目录。详见部署指南。

自行安装客户端后，在启动 AgentSQL 前设置客户端库目录，例如：

```
export LD_LIBRARY_PATH="/path/to/yashandb-client/lib${LD_LIBRARY_PATH:+:$LD_LIBRARY_PATH}"
./agentsql --version
./agentsql serve -c ./deploy/systemd/config.yaml
```

Linux 发行包将客户端放在 `lib/yashandb`，安装器写入 `/usr/local/lib/agentsql/yashandb`；GHCR 镜像中的路径为 `/opt/yashandb-client/lib`。使用前按正式资产的清单、签名与独立可信公钥核对文件和架构。

客户端缺失或动态加载失败时，YashanDB 连接失败关闭；其他数据源的使用不依赖该客户端。v2.1.0 已在 Enterprise Edition 23.4.1.109、`COMPAT_VECTOR=yashan`、x86_64 单实例合成数据上验证最小权限 HTTP MCP、窄 SELECT、R010 拒绝、手机号脱敏、Inspector/Dify、持久审计、SELECT 撤权，以及业务库和独立审计库故障关闭与恢复。参见指定实例验收与正式发布证据。

TLS、客户端取消、SQL 超时、主备、持续生产负载与 aarch64 崖山服务端仍未验证；写入、事务、EXPLAIN 和未知输出继续拒绝，不据此声明完整生产支持。出现 `libyascli.so: cannot open shared object file` 时检查实际进程的 `LD_LIBRARY_PATH`；出现未定义符号、架构错误或 `YAS-02143` 时，核对 Go 驱动、独立客户端、服务器版本与 CPU 架构。

6. 权限 `/policies`

策略绑定一个 Agent 与一个数据源。默认无授权即拒绝；创建数据源和 Agent 并不自动授予任何表。

对象与动作

- 对象类型：`database`、`schema`、`table`、`column`。
- 动作：`allow`、`deny`。
- 合法对象名：`*`、`schema.*`、`table`、`schema.table`。
- 非法对象名：空值、首尾空格、多于一个点、空片段、`*.x`、`*.*`。

> **高危配置警告：`*` 与 `schema.*` 都是合法的整表全列授权，会让匹配表的列级白名单失效。不要把它们当作普通示例或便捷默认值。只有 `*.*` 与 `*.x` 属于非法写法。**

`database`、`schema`、`table` 三种对象类型当前走同一套表模式匹配，没有独立的数据库层级或 schema 层级解析语义。

列级白名单

列级策略必须满足：

- 对象名精确到 `table` 或 `schema.table`。
- 不允许任何通配符。
- 至少填写一列。
- 动作只能是 `allow`。

v1.0.0 起，有列策略的 SELECT 不再退回仅校验表名的路径。PostgreSQL 必须由已启用且可用的 B2 绑定器核对投影、谓词、JOIN 和自连接中的列引用；B2 不可用会在执行前失败关闭。HighGo、Vastbase 等没有等价列绑定证明的数据源不能依赖列策略；MySQL 仅在其独立受限 B2 子集启用后可用。权限页会提示这一边界；离线预检只证明明确拒绝或未知，不代表真实执行许可。

v3.4.0 仅在指定 KingbaseES V9R1C10 PG 模式单实例验证了 PostgreSQL 协议路径的受限单表 `SELECT` B2 列授权。接入时先检查实例版本、B2 就绪和实时列允许/拒绝；视图、JOIN、函数及无法证明列来源的语句仍拒绝。该内核的 B5 写入行数合同未通过，网关写入与会话入口关闭。Vastbase G100 Build 9 的 B2 探测未通过，列策略继续失败关闭；两者均不得套用“完整 PostgreSQL 支持”的声明，见真库验收。

`row_filter`

`row_filter` 尚未实现行级执行。当前候选版拒绝新增含该字段的策略；遇到旧数据中命中的行筛选策略，策略解析和 B2 列授权均失败关闭，不能按未筛选结果执行。它不是已支持的行级安全，不要配置、宣传或用它替代数据库 RLS 或安全视图。

临时只读授权的使用边界 `/task-read-grants`（v1.0.0）

具备 `query.execute` 的申请人提交一个 Agent、一个数据源和一条完全匹配的简单 `SELECT`，有效期最长 24 小时；另一位同租户且同时具备 `strategy.manage` 与 `query.execute` 的用户核对 SQL 并审批，批准后原申请人一次性领取 Key。列表只显示 SQL 哈希与状态，审批详情仅向有权审批者显示 SQL 原文。任务 Key 在旧 `/mcp` 和新 `/mcp/2026-07-28` 都只注册 `query`，不提供写入、审批或 B5 工具。它只收窄原 Agent 权限，原有对象策略、规则、脱敏和数据库只读账号约束继续生效。

调用时 SQL 原始字节必须与签发时一致；空格、大小写、参数或注释变化都可能使请求被拒绝。当前仅接受单条、以 `SELECT` 开头且不含圆括号的保守子集，函数、子查询、`WITH`、`EXPLAIN`、锁行和写入均不适用。完成任务后在同页撤销；撤销后的新请求被拒绝，已在途结果仍有并发窗口。进入 SQL 流水线的任务请求会记录授权 ID 和签发者，不把任务 Key 或 SQL 原文写入流量审计；工具层前置拒绝可能没有持久记录。完整边界见 任务级临时只读授权。

任务授权的申请、审批与领取（v1.0.0）

申请人需要 `query.execute` 权限，提交一条精确的简单 `SELECT`、Agent、数据源、最长 24 小时的到期时间和最多 1–10 次的结果预算；省略预算时默认为 1 次。列表展示 SQL SHA-256、状态、申请人、审批人和授权 ID，不展示 SQL 原文或任务 Key。另一位同租户且同时拥有 `strategy.manage` 与 `query.execute` 的用户在申请详情中核对 SQL，批准或拒绝。获批后只能由原申请人领取一次任务 Key；复制并安全保存这次响应，刷新页面不能再次获取。旧版直接签发接口返回 HTTP 410。

授权列表显示 `已用 / 上限`；预算在持久审计后、结果交付前原子扣减。旧版空预算授权保持原有效期与撤销约束。授权 ID 可以带到「审计」页筛选相关管理与 SQL 流水线记录。需要立即停用时由申请人或管理员撤销；审计记录中只保留授权 ID、申请 ID、SQL 哈希和操作者。升级或恢复 metadata 库时必须保留原 `AGENTSQL_SECRET`，否则未完成申请中的加密 SQL 无法供审批人核对。更多条件与限制见 任务级临时只读授权。

MCP 用户身份委托（v1.0.0 可选）

启用 `mcp.delegation.require_human_token: true` 且保持控制台开启后，拥有 `query.execute` 的人员可在本页按 Agent ID 签发短时委托令牌。页面只展示这一次；令牌最长有效 30 分钟，仅限指定 Agent 的 HTTP MCP 调用，不能访问控制台 API，也不能代替 Agent Key。客户端须分别发送 Agent Key 和 `X-AgentSQL-User-Authorization` 请求头。用户退出登录、停用或失去查询权限后令牌立即失效；有状态 MCP 会话换人后需重新初始化。签发写管理审计，审计不可用时不交付令牌。双身份模式暂不提供 B5、`request_approval` 与 `execute_approved`。配置示例和请求头见 MCP 接入指南。

7. 规则 `/rules`

内置规则目录

规则页的严重度使用 1（低）到 5（严重）；`dynamic` 表示需要 EXPLAIN、索引/表元数据或事务运行态等动态信息。

ID	标题	数据库	严重度	dynamic
R001	多语句堆叠防护	通用	5	否
R002	无条件批量写防护	通用	5	否
R003	只读 Agent 写操作防护	通用	5	否
R004	大范围扫描审批	通用	4	是
R005	大结果集限制提醒	通用	3	是
R006	注释与未知语句注入防护	通用	5	否
R007	危险函数黑名单	通用	5	否
R008	请求速率与并发限制	通用	5	否
R009	SQL 复杂度限制	通用	4	否
R010	越权表访问防护	通用	5	否
R101	高危 DROP 防护	PostgreSQL	5	否
R102	持重锁操作审批	PostgreSQL	4	否
R103	管理与文件函数防护	PostgreSQL	5	否
R104	COPY 外部程序防护	PostgreSQL	5	否
R105	无索引批量写审批	PostgreSQL	4	是
R106	大表结构变更审批	PostgreSQL	4	是
R107	长事务与空闲事务告警	PostgreSQL	3	是
R201	MySQL 文件读写防护	MySQL	5	否
R202	MySQL 危险批量写审批	MySQL	4	否
R203	MySQL 高危管理命令	MySQL	5	否
R204	MySQL 大事务审批	MySQL	4	是

标题、严重度和 `dynamic` 属性以 `web/src/constants/ruleMeta.ts` 为准。

当前可编辑能力

- 内置规则运行时只读取 `enabled`。控制台只能启用或停用，修改在下一次真实请求生效。
- 内置规则不能删除。

- 当前不提供运行时阈值、动作或风险分的编辑；“恢复默认”只是重新启用规则。
- 自定义规则支持 CRUD，字段为 `id`、`db_type`、`title`、`risk_level`、`pattern_type=ast`、`definition`、`enabled`。
- **自定义规则 ID 不会被装配进执行引擎。** 这只是配置记录能力，当前没有实际拦截效果。

阈值与动态依赖

内部阈值键包括：

- 通用：`max_scan_rows`、`row_limit`、`qps_per_agent`、`max_conns_per_datasource`、`max_sql_length`、`max_nesting_depth`、`max_union_count`。
- PostgreSQL：`large_table_rows`、`long_transaction_ms`、`idle_transaction_ms`。
- MySQL：`mysql_transaction_age_ms`、`mysql_transaction_affected_rows`。

这些阈值属于配置/运行时层，不在规则页编辑。当前公开严格 YAML 的 `defaults` 仅暴露 `row_limit`、`qps_per_agent`、`max_conns_per_datasource`，以及执行超时 `statement_timeout_ms`；其余名字虽然是引擎内部阈值键，但 `internal/config/config.go` 尚未暴露相应 YAML 字段。不要把未暴露键直接加入配置，否则严格解析会拒绝启动；公开配置入口需主控确认后再补。

动态规则依赖：R004 与 R005 依赖 EXPLAIN；R105 依赖索引元数据；R106 依赖表行数；R107 与 R204 依赖事务运行态。

> **R005 特别说明：**普通生产流水线在动态阶段使用 EXPLAIN 估算结果规模；估算超过数据源 `row_limit` 的无界查询命中 R005。通用执行路径与 PostgreSQL 列级授权路径还会在实际 `row_limit` 截断时补充 R005 命中，并写入同一条持久审计。EXPLAIN 或结果规模事实不可可靠取得时保持 fail-closed。

页面目录的 1–5 是规则严重度。流水线内部另有 `1=deny`、`2=approve`、`3=warn`、`4=info` 的决策优先级；两者不是同一个统一分值，不得混写。

8. 审批 `/approvals`

普通 SQL 审批状态包括 `pending`、`approved`、`claimed`、`completed`、`failed`、`unknown`、`rejected`、`expired`。新申请有 24 小时有效期；过期后即使状态仍显示 `approved` 也不能认领。旧申请缺少封存字段，不可执行。

管理员只能决定 `pending` 单。决定请求体是：

```
{
  "decision": "approve",
  "comment": "可选"
}
```

`decision` 只允许 `approve` 或 `reject`，保存后状态变为 `approved` 或 `rejected`。

> **核心语义：** 管理员批准不会自动执行 SQL。原 Agent 可在有效期内调用 `execute_approved(approval_id)` 一次；执行时重新核验身份、数据源、当前策略与规则。人工批准不能绕过拒绝规则。认领后的记录不得自动重试。

工具层有两条路径：

- `execute_write` 返回 `decision:"approve"` 时，系统已经自动创建 `pending` 单并返回 `approval_id`，不要再调用 `request_approval`。
- 调用方主动希望先送审时，才使用 `request_approval`；它同样只建单、不执行。

调用方传入的 `reason` 保存为申请说明；规则评估原因、审批备注分别保存。普通 SQL 审批当前只接受字面 SQL，不提供参数数组输入。

9. 脱敏 `/mask-rules`

脱敏按列在查询结果层执行，不修改数据库原值。当前有四种可运行算法，分别面向保留可读片段、生成可关联指纹、整值阻断和保留粗粒度分布：

算法	支持类型	保留原文片段	可等值关联	需要密钥	典型场景
打码 `mask`	手机号 `phone`、邮箱 `email`、身份证 `idcard`、银行卡 `bankcard`、IP 地址 `ip`、出生日期 `birthdate` 六类	是，按格式部分遮蔽	否	否	客服核对、运维排障等需要保留少量可读信息的列
哈希指纹 `hash`	九类	否；v1 输出 34 字符 `h.<32hex>`，v2+ 输出 `h.<版本>.<32hex>`	仅同一版本与同一密钥下，相同字符串字节得到相同指纹	是；单 key 或严格多版本 manifest，每份材料至少 32 字节	同版本窗口内的下游等值关联、去重和分组
阻断 `block`	九类	否；每个非空值固定输出 `***`，不保留原文长度	否	否	完整证件、密码/密钥、薪资、健康或违法细节等最高敏感列
分桶/截断 `range`	数值 `number`、日期 `date`	否；输出数值区间或年/季/月	是；同桶值可关联	否；使用类型专属参数	年龄分段、金额区间，以及出生日期按年/季/月统计

选择算法时，可读性核对优先使用 `mask`，需要稳定关联且不能返回原文时使用 `hash`，不应返回任何非空原值片段时使用 `block`，需要观察数值或日期的粗粒度分布时使用 `range`。`range` 不是匿名化：同桶值仍可关联，敏感场景应改用 `block` 或纳入审批。

类型与算法能力矩阵

敏感类型	mask	hash	block	range
手机号 phone	✅	✅	✅	❌
邮箱 email	✅	✅	✅	❌
身份证 idcard	✅	✅	✅	❌
银行卡 bankcard	✅	✅	✅	❌
IP 地址 ip	✅	✅	✅	❌
出生日期 birthdate	✅	✅	✅	❌
通用敏感值 generic	❌	✅	✅	❌
数值 number	❌	✅	✅	✅
日期 date	❌	✅	✅	✅

`number` / `date` 表示列语义，`range` 只对这两类开放。`birthdate` 仍使用生日专用 `mask`，即使原值可解析为日期也不能选择 `range`；手机、邮箱、身份证、银行卡、IP、生日六类以及 `generic` 同样不能选择 `range`。`hash` / `block` 可处理九类，管理员在 `range`、`hash`、`block` 之间切换时不需要把 `number` / `date` 改标为 `generic`。

`generic` 不依赖手机号、邮箱等格式识别，可对命中列的完整原始字符串做 `hash` 或 `block`；它不能搭配 `mask`，`mask+generic` 属于非法组合。除空值哨兵外，哈希前不会 trim，也不会做大小写、Unicode、日期、时区或 decimal 正规化，因此表示不同的值会得到不同指纹。`block` 同样不识别格式且没有参数，任何非空值（包括畸形值、首尾带空格的值和原值恰为 `***` 的值）都输出固定 `***`。

> `***` 是 `block` 的正常成功输出；`[REDACTED]` 是 `mask` 或 `range` 无法安全处理非空值时的 fail-closed 兜底。两者的字面量和算法语义不同。规则管理界面和规则配置可通过 `algo=block` 识别算法，但查询审计、SSE、通知不记录具体脱敏算法，结果投影本身也不能可靠反推出 `***` 的产生原因；本版不扩展审计协议。

空字符串、trim 后为空的字符串，以及忽略大小写的 `NULL` / `<nil>` 哨兵会原样返回且不计入 `MaskedCells`；因此 `block` 会暴露该结果单元为空/NULL 的状态。SQL NULL 与数据库空字符串在进入脱敏前都已字符串化为 `""`，当前结果模型无法区分。

敏感列发现始终只识别前六类并生成 `algo=mask` 的 disabled 草稿；它不会产出 `hash`、`block` 或 `range` 规则，也不会发现 `generic`、`number` 或 `date`。`number` / `date` 的 `range` 规则需要在「脱敏规则」页手工新建；发现流程的完整说明见《敏感列发现指南》。

控制台的算法下拉有「打码（mask）」「哈希指纹（hash）」「阻断（block）」「分桶/截断（range）」四项。配置 `block` 时：

1. 进入「脱敏规则」并新建或编辑规则，算法选择「阻断（block）」。
2. 选择九类中的任一支持类型，填写目标结果列和作用域；页面说明和示例固定显示 `***`，不显示哈希密钥警示。
3. 保存并启用规则；`block` 不需要密钥或其他参数，不参与 hash key 启动 fail-fast，也不会返回 `HASH_REDACTION_UNAVAILABLE` / HTTP `503`。

未配置哈希密钥时，`mask`、`block` 与 `range` 可正常创建、启用和运行；只有 `hash` 受密钥门禁约束。此时 `hash` 可保存为停用规则，但默认启用的新建规则或启用操作会被拒绝，页面对应的服务端响应为 HTTP `503`、`HASH_REDACTION_UNAVAILABLE`，且规则不会写入。通过 `AGENTSQL_REDACTION_HASH_KEY` 或 `redaction.hash_key` 配置至少 32 字节密钥并重启后，再启用 `hash`。类型/算法组合不合法时服务端返回 HTTP `422`、`INVALID_MASK_RULE`。

三档作用域与选择建议

作用域	schema_name	table_name	何时使用
全局列规则	`""`	`""`	同名列需要跨所有表统一保护时；例如任何查询结果中的 `phone` 都应打码
表.列规则	`""`	真实表名	仅保护某张表的列；**默认推荐**，敏感发现生成的 table-only 草稿即此档
模式.表.列规则	真实模式名	真实表名	PostgreSQL 等场景中，跨模式存在同名表且必须区分时

这三档是列规则的关系作用域，与数据源作用域正交。不选数据源时，规则对所有数据源生效；选择数据源时，只在该数据源中生效。`schema_name` 不能脱离 `table_name` 单独填写。

配置全局列规则

1. 进入「脱敏规则」，选择「新建规则」。
2. 按需选择数据源；「模式 Schema」和「表名」均留空。
3. 填写列名，选择敏感类型、算法和启用状态后保存。

配置表.列规则

1. 进入「脱敏规则」，选择「新建规则」并选定数据源。
2. 「模式 Schema」留空，「表名」填写真实表名，例如 `customers`。
3. 填写列名、敏感类型与算法；手工新建可直接启用，发现生成的草稿则应先审阅再启用。

配置模式.表.列规则

1. 进入「脱敏规则」，选择「新建规则」并选定数据源。
2. 同时填写「模式 Schema」与「表名」，例如 `tenant_a` 和 `customers`。
3. 填写列名、敏感类型、算法和启用状态后保存。查询中只有显式写出且精确匹配的 schema 才能命中此档。

PostgreSQL 的表.列规则留空 schema 时匹配任意模式，不会自动填入 `public`。MySQL 的表名按大小写敏感的精确字符串比较，列名大小写不敏感；留空 schema 即当前库的 table-only 语义。PostgreSQL 的 `customers` 与 `"Customers"` 也是两个不同的表键。

列名会去除首尾空白和单层引号、反引号或方括号，再统一小写；表名与 schema 则保留原始大小写并精确比较。同一物理作用域与归一化列名只能有一条规则，disabled 草稿也占用该唯一键。

配置 `range` 数值分桶

数值规则选择敏感类型 `number` 和算法 `range`。控制台表单会显示「桶宽」与「偏移」两个整数输入：

- `range_bucket_width`：桶宽，必填，必须是 `1..1,000,000,000` 的整数。
- `range_bucket_offset`：偏移，可留空，默认 `0`，必须是 `-1,000,000,000..1,000,000,000` 的整数。

直观上，偏移决定桶边界的起点，桶宽决定每段覆盖多少数值。公式为 `lower = floor((v - offset) / width) * width + offset`、`upper = lower + width`，其中 `floor` 向负无穷取整。结果使用左闭右开的 `[lower,upper)`：`width=10`、`offset=0` 时，`42` 输出 `[40,50)`，`-3` 输出 `[-10,0)`。

配置 `range` 日期截断

日期规则选择敏感类型 `date` 和算法 `range`，不使用桶宽或偏移。`range_granularity` 可留空，默认 `year`；显式值只能是年 `year`、季 `quarter` 或月 `month`。例如 `1990-08-21` 按年输出 `1990`，按季输出 `1990Q3`，按月输出 `1990-08`。

完整日期、日期时间、RFC3339 以及 `YYYY` / `YYYY-MM` 等常见日期写法均可处理；输入精度不足以生成所选粒度、日期非法或无法识别时，会安全地显示 `[REDACTED]`。

API 字段与完整更新语义

新建三档规则均使用 `POST /api/v1/mask_rules`。以下是一条全局列规则：

```
{
  "id": "global-phone-mask",
  "datasource_id": "ds-pg",
  "schema_name": "",
  "table_name": "",
  "column_name": "phone",
  "sensitive_type": "phone",
  "algo": "mask",
  "enabled": true
}
```

表.列规则的 `schema_name` 留空，例如只保护 `customers.phone`：

```
{
  "id": "customers-phone-mask",
  "datasource_id": "ds-mysql",
  "schema_name": "",
  "table_name": "customers",
  "column_name": "phone",
  "sensitive_type": "phone",
  "algo": "mask",
  "enabled": true
}
```

模式.表.列规则同时填写两个作用域字段：

```
{
  "id": "tenant-a-customers-phone",
  "datasource_id": "ds-pg",
  "schema_name": "tenant_a",
  "table_name": "customers",
  "column_name": "phone",
  "sensitive_type": "phone",
  "algo": "mask",
  "enabled": true
}
```

更新使用 `PUT /api/v1/mask_rules/{id}`。PUT 是完整更新，应每次发送希望保留的 `datasource_id`、`schema_name`、`table_name`、`column_name`、`sensitive_type`、`algo` 和 `enabled`；路径中的 `{id}` 是目标规则 ID，请求体不需重复传 `id`。例如：

```
PUT /api/v1/mask_rules/customers-phone-mask
Content-Type: application/json
```

```
{
  "datasource_id": "ds-mysql",
  "schema_name": "",
  "table_name": "customers",
  "column_name": "phone",
  "sensitive_type": "phone",
  "algo": "block",
  "enabled": false
}
```

对 `schema_name` 和 `table_name` 这两个作用域字段，缺省、JSON `null` 与空串都按空值处理；为避免 PUT 时意外改变作用域，建议始终显式传空串。`datasource_id` 缺省、`null` 或空串表示所有数据源。

在脱敏规则创建/更新 JSON 中，`range` 额外使用 `range_bucket_width`、`range_bucket_offset`、`range_granularity`。数值规则不能携带 `range_granularity`，日期规则不能携带 `range_bucket_width` 或 `range_bucket_offset`。一个最小数值规则请求体如下：

```
{
  "id": "amount-range",
  "schema_name": "",
  "table_name": "orders",
  "column_name": "amount",
  "sensitive_type": "number",
  "algo": "range",
  "range_bucket_width": 10,
  "range_bucket_offset": 0,
  "enabled": true
}
```

一个最小日期规则请求体如下：

```
{
  "id": "birth-date-range",
  "schema_name": "",
  "table_name": "customers",
  "column_name": "birth_date",
  "sensitive_type": "date",
  "algo": "range",
  "range_granularity": "month",
  "enabled": true
}
```

规则创建和更新采用完整 PUT 语义：请求中缺失的 `range` 参数会写回 `NULL`；从 `range` 切换到其他算法，或在 `number` / `date` 之间切换时，不适用的参数会被清空。数值偏移 `0` 是合法值，API 会保留它；省略偏移时则规范化为 `0`。

精确匹配与 fail-closed 安全兜底

顶层直接投影列能唯一归属到物理关系时，表级规则按来源表和来源列精确匹配；有 schema 精确规则时先用精确规则，再尝试 `schema_name=""` 的表.列规则。裸列只在顶层 `FROM` 作用域恰好有一个可见来源，且该来源是物理关系时才能唯一归属。表起了别名后必须使用别名限定；原表名已被遮蔽。

JOIN 裸列、`SELECT *` 和 CTE 外层列可能无法确定物理来源。如果某个未解析结果列命中表级规则，且该受保护表出现在整条语句的可能关系集合中，AgentSQL 不会因“无法证明来源”而返回原文，也不会拒绝整条查询；它会对该列执行不可关闭的 fail-closed 安全兜底，将所有非空单元格固定阻断为 `***`。这个兜底不使用某张表配置的 `mask` / `hash` / `range` 业务算法，因为来源尚未确定。

消除误伤的首选方式是精确化 SQL：为列加上表名或表别名限定符，并避免使用 `SELECT *` 投影受保护列。如果同名列本来就应跨所有表统一保护，可把该列改配为全局列规则。

PostgreSQL JOIN 示例（假设 `orders` 没有 `phone` 列，SQL 在数据库中合法，但 parser 不查询 schema 来猜列归属）：

```
-- 修改前：多来源裸列无法唯一归属，可能触发 *** 兜底
SELECT phone
FROM public.customers AS c
JOIN public.orders AS o ON o.customer_id = c.id;

-- 修改后：用表别名限定，精确归属到 public.customers.phone
SELECT c.phone
FROM public.customers AS c
JOIN public.orders AS o ON o.customer_id = c.id;
```

MySQL `SELECT *` 示例：

```
-- 修改前：star 不产生可精确归属的顶层直接投影
SELECT *
FROM customers;

-- 修改后：显式投影并限定受保护列
SELECT customers.id, customers.full_name, customers.phone
FROM customers;
```

PostgreSQL CTE 示例：

```
-- 修改前：外层只看到非物理来源 x，phone 可能触发 *** 兜底
WITH x AS (
  SELECT phone FROM public.customers
)
SELECT phone FROM x;

-- 修改后：直接限定物理关系
SELECT customers.phone
FROM public.customers;
```

在 MCP `query` 响应中，实际改变了非空单元格的兜底列会记入 `redact.unresolved_scoped_columns`。当前 Playground REST 响应只投影 `touched_columns` 和 `masked_cells`，Playground 与审计 UI 都不展示 `unresolved_scoped_columns` 计数。在这些页面看到结果为 `***` 时，先检查查询是否使用了 JOIN 裸列、`SELECT *` 或 CTE 外层列，再按上述方式精确化 SQL。

敏感发现工作流

1. 在「数据源」页进入敏感发现，一次选择 1–20 张可见基础表，按需开启每表 1–20 行（默认 10 行）的受控样本扫描。
2. 审阅候选列、类型、置信度与证据计数；发现只识别手机、邮箱、身份证、银行卡、IP 和出生日期六类。
3. 选中后应用候选。系统按物理来源 `(datasource, schema, table, column)` 识别候选，并生成 `schema_name=""`、`table_name=<真实表名>`、`algo=mask`、`enabled=false` 的 table-only 草稿。
4. 前往「脱敏规则」页，逐条核对表名、列名、敏感类型与算法，确认后再启用。草稿默认停用，创建后不会立即影响查询。

不同表中的同名列会生成多条独立的 table-only 草稿。如果某候选列已被启用的全局列规则覆盖，应用结果会标记 `CoveredByGlobal` 并跳过新建；disabled 全局草稿不会阻挡新的 table-only 草稿。

校验与冲突报错

HTTP / 错误码	含义	处理方式
`422 INVALID_MASK_RULE`	`schema_name` 非空但 `table_name` 为空，作用域标识符非法，或类型/算法/参数组合非法	同时填写 schema 和 table，去除首尾空白、通配符、控制字符或单字段内的 `.`，并核对类型与算法
`409 MASK_RULE_CONFLICT`	同一物理键 `(datasource_scope,schema_name,table_name,归一化 column_name)` 已存在；disabled 行也算冲突	编辑既有规则或删除其一；仅停用无法释放物理唯一键
`409 MASK_RULE_SCOPE_CONFLICT`	同一 datasource 有效范围内，同列名的启用全局列规则与表级规则算法不一致	对齐算法或删除其一；可先停用待调整规则，再修改另一条并按顺序重新启用

`MASK_RULE_SCOPE_CONFLICT` 只比较启用规则，disabled 草稿豁免。对 `range` 规则，“算法一致”还要求敏感类型相同；数值规则比较桶宽和规范化后的偏移，日期规则比较规范化后的截断粒度。

`range` 行为与边界

- `range` 无需配置密钥；即使服务没有 hash 密钥也可创建、启用和运行。
- 空串、trim 后为空、忽略大小写的 `NULL` / `<nil>` 等空值哨兵原样返回且不计入 `MaskedCells`；其他非空值无论成功泛化还是输出 `[REDACTED]` 都计入。
- 泛化在网关取回查询结果后于内存中完成，不改写数据库数据、不减少数据库查询，也不阻止数据库按原值执行 `WHERE`、`JOIN` 或 `GROUP BY`。
- 泛化结果是字符串，不保证兼容原数值或日期 schema，消费方应按文本处理。
- `range` 不承诺 k-匿名；同桶值仍可关联，数值区间、日期时段和空值状态仍然可见。小桶、月粒度、小样本、多偏移或辅助查询都可能增加重识别风险，敏感场景应使用 `block` 或审批。

确定性哈希指纹会暴露相等关系与频率，也可能在共享同一 key 的库或表之间形成关联追踪。不可逆不等于匿名：key 泄漏、hash oracle、低熵字典猜测或辅助数据仍可能重识别原值。互不应关联的环境或租户应使用不同 key。完整密钥管理与轮换限制见部署指南的“脱敏哈希密钥管理”。

脱敏密钥轮换与等值边界

v0.4 支持多版本密钥 manifest、登记对账、历史指纹核验和**重启式计划切换**，不支持热加载、蓝绿混版本服务、自动重算或自动双写。v1 指纹形如 `h.<32 位小写十六进制>`；v2 及以后形如 `h.<版本>.<32 位小写十六进制>`。

标准切换窗口是：公告维护 → 停止或拒绝新写入 → 等待旧实例在途请求结束 → 在登记表中把目标版本切为 active → 修改所有实例使用的 manifest `active_version` → 重启全部实例 → 执行强对账并确认 `/readyz` 返回 HTTP 200 → 恢复流量。旧实例必须完全退出后才能让新实例服务；同一服务组内所有实例的 `active_version`、对应 key commitment 与 revision 必须一致。v0.4 不提供跨实例协调器，不能让不同 active 版本同时对外服务。

轮换会发生**等值分裂**：同一明文在 v1 与 v2 下的指纹不同。依赖指纹列做 JOIN、去重、缓存键、CDC 比对或报表关联的下游，必须在维护窗口重 hash，或由下游自行实施双版本过渡；v0.4 不提供自动化。长期需要跨轮换稳定关联时，应设计稳定 token 或双版本 token，不能假定当前指纹跨版本相等。

回滚按相反顺序执行：再次停写和 drain → 把旧版本 CAS 回 active → 恢复旧 `active_version` 并重启全部实例 → 强对账与 `/readyz` 验证 → 恢复流量。只要旧 key 材料未改变，同一明文会恢复为逐字节相同的旧指纹；切换期间产生的新版本历史指纹仍可用对应 legacy key 核验。确认所有历史依赖和回滚窗口结束前不要退役 key；退役是终态，管理核验入口不再接受该版本。

空字符串、trim 后为空的字符串，以及忽略大小写的 `NULL` / `<nil>` 是现有结果模型的空值哨兵：它们原样返回、不生成指纹；SQL NULL 与数据库空字符串在进入脱敏前都字符串化为 `""`，无法区分。字面量 `"NULL"` 因此也不会被 hash，不能把它当作普通业务字符串参与等值关联。

`block` 不输出原值字符、长度或等值关系，但仍保留结果集行列形状、行数、列名和是否有结果，并泄漏上述空值状态；这与 `mask` / `hash` / `range` 的结果层泄漏等级一致，不是匿名化或“零信息”。四种算法都在数据库执行、结果返回 AgentSQL 后处理，不减少数据库读取，也不阻止数据库侧按原值执行 `WHERE`、`JOIN` 或 `GROUP BY`。被 `block` 的列是不透明字符串；固定 `***` 不保证数值、日期、JSON、UUID 等下游 schema 兼容，消费方应按不透明文本处理。

已知限制与建议

- PostgreSQL/MySQL 已验证的函数包裹、别名、聚合、`CAST`、CTE 和集合运算会跟踪受保护列；无法唯一证明安全来源且可能涉及受保护列时，结果固定阻断为 `***`。`CONCAT(phone,'') AS x`、`lower(phone) AS y` 等已纳入负向用例。新的数据库函数、视图内部语义或多步统计查询仍可能超出证明能力，不可据此宣称完整 DLP。
- `RETURNING` 走写屏障，只返回 `RowCount`，不进入查询结果脱敏。不得把它当作一条可返回脱敏后行数据的通道。
- 视图引用名可按已配置规则匹配；未知视图或函数内部的真实来源不由 AgentSQL 自动证明。数据库最小权限、安全视图和列级授权仍需独立配置与验证。

10. 配置参考

配置文件使用严格 YAML：未知字段会导致启动失败。生成基础配置可使用 `agentsqlctl init-config`，生产细节见 部署指南。

`server`

字段	说明
`http_listen`	HTTP 监听地址，例如 `127.0.0.1:7780`
`console_enabled`	是否启用管理控制台；未显式配置时默认 `true`
`event_stream`	是否启用 SSE；未显式配置时默认 `true`
`event_stream_max_connections`	SSE 最大连接数，默认 100，有效范围 1–1000

`store`

`store.sqlite_path` 是 SQLite 简写，不能与 `store.metadata` 同时使用。

字段	说明
`sqlite_path`	默认 SQLite 控制面路径
`auto_migrate`	默认 `true`；`false` 时只校验 schema 版本，不执行 DDL
`metadata.driver`	`sqlite` 或 `postgres`，默认 `sqlite`
`metadata.sqlite_path`	metadata 使用 SQLite 时的路径

字段	说明
`metadata.dsn`	metadata 使用 PostgreSQL 时的 DSN
`metadata.max_open_conns`	默认 10
`metadata.max_idle_conns`	默认 5，不能超过 open
`metadata.conn_max_lifetime`	默认 `30m`，使用 Go duration 字符串
`audit.separate`	是否使用独立 audit 存储，默认 `false`
`audit.driver`	独立 audit 当前只支持 `postgres`
`audit.dsn`	独立 audit PostgreSQL DSN
`audit.max_open_conns`	默认 10
`audit.max_idle_conns`	默认 5，不能超过 open

`defaults`

字段	说明
`statement_timeout_ms`	全局默认语句超时，不能为负；模板值 5000
`row_limit`	全局默认结果行数上限；模板值 1000
`max_conns_per_datasource`	全局默认数据源连接/并发上限；模板值 5
`qps_per_agent`	全局默认每 Agent QPS 与突发上限；模板值 20

规则内部还识别 `max_scan_rows`、`max_sql_length`、`max_nesting_depth`、`max_union_count`、`large_table_rows`、`long_transaction_ms`、`idle_transaction_ms`、`mysql_transaction_age_ms`、`mysql_transaction_affected_rows` 等阈值键，但当前公开 `Config` 结构没有对应 YAML 字段，状态为**待确认/未公开配置入口**。它们不能在规则页修改，也不要直接加入严格 YAML。

`theme`

字段	说明
`default`	`light` 或 `dark`

`demo`

字段	说明
`enabled`	是否启用专用 Live Demo
`banner`	Demo 强制横幅；空值使用固定默认文案
`allowed_datasource_ids`	启用时必须恰好是 `ds-demo-pg` 与 `ds-demo-mysql`
`qps_per_agent`	Demo QPS，默认 2，有效范围 1–5
`anchor_date`	可选，`YYYY-MM-DD` 真实日期

生产环境不要启用 `demo`。

`redaction`

字段	说明
`hash_key`	可选的 HMAC 哈希指纹密钥，非空时至少 32 字节；为空或未配置时 `mask`、`block` 与 `range` 仍可运行，enabled `hash` 会导致启动 fail-fast
`hash_keys.active_version`	多版本 manifest 的活动版本，范围 1–9999；必须恰好匹配一个 `keys[].id`，变更后需重启装配
`hash_keys.revision`	可选的非秘密配置 revision；多实例应保持一致，省略时由 active 版本与 commitments 派生
`hash_keys.keys[]`	版本化 key 列表；每项只能使用 `key_file` 或 canonical `key_b64` 之一，材料至少 32 字节并满足多样性检查

`hash_key` 是兼容的单 key 配置；它不能与 `hash_keys` 或 `AGENTSQL_REDACTION_HASH_KEYS_FILE` 混用。推荐让 `AGENTSQL_REDACTION_HASH_KEYS_FILE` 指向受权限保护的绝对路径 manifest。密钥不得与 `AGENTSQL_SECRET` 复用或相互派生，也不得进入 Git、镜像、日志、审计或配置回显。

环境变量

变量	用途
`AGENTSQL_SECRET`	必填，恰好 32 字节；加密数据源密码并派生管理员 token 签名密钥
`AGENTSQL_REDACTION_HASH_KEY`	可选，至少 32 字节；仅供 `hash` 生成不可逆 HMAC 指纹，存在即覆盖 YAML（含空串）；未配置时仍可使用 `mask`、`block` 与 `range`
`AGENTSQL_REDACTION_HASH_KEYS_FILE`	可选，多版本密钥 manifest 的绝对路径；与 inline `hash_keys`、`hash_key`、`AGENTSQL_REDACTION_HASH_KEY` 互斥
`AGENTSQL_ADMIN_USER`	管理员用户名，默认 `admin`
`AGENTSQL_ADMIN_PASSWORD`	控制台启用时必填，至少 12 字符且不能使用弱口令/公开示例
`AGENTSQL_API_KEY`	stdio 的 `--api-key` 替代；优先用环境变量，避免进入进程参数
`AGENTSQL_STORE_METADATA_DSN`	存在即覆盖 YAML metadata DSN，包括显式空值
`AGENTSQL_STORE_AUDIT_DSN`	存在即覆盖 YAML audit DSN，包括显式空值
`AGENTSQL_INSECURE`	仅本地测试允许公开示例凭据；不关闭认证、安全规则或 fail-closed
`AGENTSQL_DEMO`	值为 `1` 时启用 Demo 模式
`AGENTSQL_DEMO_RO_KEY`	Demo 只读 Agent Key
`AGENTSQL_DEMO_DML_KEY`	Demo DML 拦截 Agent Key
`LOG_LEVEL`	日志级别
`LOG_FORMAT`	`json`（默认）或 `console`

`AGENTSQL_DOWNLOAD_BASE` 仅供安装器下载源覆盖；`AGENTSQL_BENCH_DSN` 仅供 benchmark。两者都不是生产运行配置项。

11. 运维操作

`agentsqlctl` 公开子命令

子命令	用途
`version`	输出版本
`init-config`	生成默认配置
`check-config`	严格校验配置，不创建 SQLite 目录
`migrate`	初始化或升级 metadata/audit schema；v0.7.0 起提供只读 `--check`
`migrate-sqlite-to-postgres`	把组合 SQLite 控制面迁移到 PostgreSQL
`health`	检查 HTTP 健康或直接检查控制面存储
`demo-seed`	仅演示环境维护，用固定 manifest 写入 Demo 数据
`redaction-key`	管理脱敏密钥登记、核验与审计发件箱 relay；子命令为 `reconcile`、`registry-mark-active`、`registry-mark-retired`、`verify`、`relay`
`chain`	管理审计完整性链；子命令为 `status`、`verify`、`provision`

不存在 `run-check-config-and-health` 子命令；该字符串只是迁移输出中的步骤文案。

`health`

两种模式互斥：

- `--url`：默认检查 `http://127.0.0.1:7780/healthz`，要求 HTTP 200 且 JSON `status:"ok"`。
- `--config` / `-c`：直接检查 metadata/audit 存储，输出 `metadata_driver`、`audit_driver`、`audit_separate`。
- `--timeout`：默认 3 秒。

```
# 适用前提：服务已在本机运行
agentsqlctl health --url http://127.0.0.1:7780/healthz --timeout 3s
```

```
# 适用前提：运行用户可读配置、SECRET 和对应控制面存储
agentsqlctl health --config /etc/agentsql/config.yaml --timeout 3s
```

升级、回滚与备份

升级器会做预检、备份和失败回滚；具体命令、服务状态检查和离线包流程见 DEPLOY.md。不要跳过外部备份确认。

v1.0.0 可在迁移窗口前使用 `agentsqlctl migrate --check --config /etc/agentsql/config.yaml` 读取已配置的 metadata/audit schema 版本。退出码 `0` 为全部当前版本，`2` 为待迁移，`1` 为配置、连接或迁移历史错误。`--check` 不执行 DDL，不创建缺失的 SQLite 库，也不验证全部数据库结构或跨库一致性；待迁移状态不能直接启动新版服务。完成成套备份并停止写入后，由 migration owner 运行 `agentsqlctl migrate`，再用运行账号复核 `--check` 为 `0`。PostgreSQL 分库要成对备份 metadata、audit 和对应 `AGENTSQL_SECRET`；详细顺序见 部署指南和 当前版本运维手册。v0.6.0 安装包不含 `migrate --check`。

备份铁律：**控制面数据库与生成其数据源密码密文的 `AGENTSQL_SECRET` 必须成对备份、成对恢复。** 仅有数据库或仅有 SECRET 都不能恢复已有数据源密码。

SQLite → PostgreSQL 控制面迁移要点：停止旧服务写入；备份 SQLite 与 SECRET；准备 PostgreSQL metadata/audit 目标；运行 `migrate-sqlite-to-postgres` 并核对行数/哈希；切换运行账号 DSN；保留原 SECRET；运行 `check-config` 与 `health`；验证 `/readyz`、读、拒绝、审批和新审计 ID。完整步骤见 DEPLOY.md。

性能口径

当前归档的唯一对外基准口径是：内存 fake executor、20 worker、20 万样本，不包含数据库网络与执行时间，网关 CPU 路径 P99≈1.81ms。它不是端到端查询延迟；真实耗时还包括连接、EXPLAIN、数据库执行与审计存储。

12. 当前版本能力边界

- 默认无授权即拒绝；必须显式创建 Agent × 数据源策略。
- 缺 `reason`、用 `query` 发写语句、SQL 在工具层解析失败等前置拒绝发生在进入流水线之前，不保证每条失败都有审计。
- 自定义规则只做配置记录，不进入执行引擎；`row_filter` 不执行，当前候选版拒绝新增并对命中的旧策略失败关闭。
- 数据源没有 SSL mode、CA、客户端证书、连接超时或 DSN 附加参数字段。
- R005 在普通生产动态阶段按 EXPLAIN 估算告警，并在通用执行与 PostgreSQL 列级授权路径实际触发 `row_limit` 截断时补充告警。
- 审批不会自动执行 SQL；原 Agent 可凭未过期审批单号单次执行封存请求，执行时重新核验当前权限和规则。
- 脱敏是结果层按列名与可唯一解析的物理来源匹配，不是完整 DLP。全局列、表.列、模式.表.列规则均已支持；JOIN 裸列、`SELECT *` 和 CTE 外层等未解析投影在可能涉及受保护表时会 fail-closed 固定阻断为 `***`。函数包裹并改名、聚合、`CAST`、`UNION` 和视图重命名仍可能无法溯源；JOIN / 自连接的授权仍只到表级；`*` / `schema.*` 是整表全列授权。
- `RETURNING` 走写屏障并只返回 `RowCount`，不经过结果脱敏。

- `hash` 只处理业务库返回后的结果值，不参与数据库内的 JOIN/WHERE；确定性指纹泄漏相等关系和频率。v0.4 支持版本化历史核验与重启式计划切换，但不提供热加载、双写、在线轮换或自动重算；跨版本同明文指纹不相等。
- `block` 同样只处理数据库返回后的结果值；它以不透明字符串 `***` 阻止结果单元外发原文，但保留结果形状、行数、列名、结果存在性和空值状态，且不保证下游 schema 兼容。它不是匿名化，也不限制数据库侧按原值过滤、关联或分组。
- `range` 只对 `number` / `date` 开放并在结果层输出字符串；同桶值仍可关联，不承诺 k-匿名，也不减少数据库读取或限制数据库侧按原值过滤、关联和分组。
- 审计是应用层记录，不是法规级 WORM，也不能防止 DBA 或其他高权限账号直连数据库。
- PostgreSQL 的 B2 列级授权、B5 跨请求逻辑会话与计划事务均出厂默认开启；MySQL 不进入 B2 PostgreSQL 路径，且不支持 B5 跨请求事务。每个 operation 仍只允许一条顶层 SQL，并受预检计划和会话安全边界约束。
- 完整原生路径仍以 PostgreSQL 14–18 与 MySQL 8 为主。DM、Oracle、YashanDB 的独立最小 dialect，以及 OpenTenBase、PolarDB 等协议兼容路径均受严格边界约束；达梦 DM8 Pack3、金仓 V9R1C10 PG 模式、Vastbase G100 Build 9 PG 模式在 v1.5.0 完成指定实例的真实客户端、权限撤销和断库恢复复验，v1.6.0 在相同范围内通过独立审计库故障、真实锁等待超时、客户端取消、数据库端终止和恢复后脱敏复验。v1.7.0 的独立 HighGo 类型仅限 9.0.10 企业版 PG 模式有界 SELECT；金仓 TLS 仅限 V9R1C10 PG 模式指定实例。范围见瀚高验收与金仓 TLS 验收，不得视为完整方言支持或厂商认证。
- v1.0.0 的 `/mcp/2026-07-28` 是独立无状态 HTTP 入口：每次 POST 都重新验证 Agent Key、由 Key 确定租户并与旧入口共用 Agent QPS；版本 Header、`Mcp-Method`、适用时的 `Mcp-Name` 必须与请求体匹配。旧 `/mcp` 与 stdio 继续服务旧协议；新入口没有 B5 跨请求逻辑会话或计划事务工具。客户端配置和请求示例见 MCP 接入指南。
- v0.4.0 起提供 linux/arm64 原生 glibc 包（`aarch64` / `arm64`），并继续提供 linux/amd64 包（`x86_64` / `amd64`）；v2.4.0 GHCR tag 须在公开发布并通过双架构 manifest 与匿名拉取验收后使用。v0.5.0 已提供可选的 TOTP MFA、OIDC 和 LDAP/AD 人员认证；配置与限制见 身份认证指南。v2.4.0 的 Kubernetes 清单限固定双网关与指定无状态只读路径；PostgreSQL 自身 HA、法规级 WORM、SIEM、musl / Alpine 或 CentOS 7 原生一键安装仍不支持，RBAC / 多租户能力仍是 MVP。

在线安装包升级预演（与 schema 检查不同）

准备可信渠道提供的 HTTPS 清单，JSON 字段为 `version`、同源 HTTPS `url` 和下载内容的 `sha256`。当前仅支持版本检查和只读预演：

```
agentsqlctl upgrade check --manifest-url https://example.org/releases/manifest.json
agentsqlctl upgrade apply --manifest-url https://example.org/releases/manifest.json --backup-dir
/new/backup/path --config config.yaml --dry-run
```

第二条命令要求备份目标是尚不存在的目录，并读取下载内容校验 SHA-256，但不会创建目录、备份文件、停止服务或安装程序。`--yes` 不能开启安装。清单及摘要须经独立可信渠道核对；详见 在线升级安全边界。

v4.2.0 离线交付与诊断

控制台「版本与升级」提供当前版本的离线验签命令和诊断导出命令。先用独立受信公钥验证 15 项正式资产，再安装；`support-bundle` 只导出固定 schema 检查结果和产品证据，拒绝覆盖已有文件。完整操作、返回码和跨版本恢复步骤见交付指南。

v4.2.0 原生问数

通过 `install.sh install --with-chatbi` 部署 Go 问数服务，配置工作空间模型与个人网关映射后，打开其独立入口。中英文界面展示 SQL、表格结果与持久审计编号；v4.2.0 支持最多三轮成功问数语境、术语与推荐问题、受控 CSV/XLSX 文件数据源和当前权限下的 CSV/XLSX 导出。原版 SQLBot 页面只在已记录的合成流程中对照通过。身份导入、离线安装、恢复和待完成的兼容范围见问数指南。

v4.3.0 图表、仪表板与嵌入应用

问数页可将当前授权结果呈现为表格、柱状图、折线图或饼图；历史图表重新查询当前权限并写审计。自有仪表板支持文字、记录图表、布局、Tab、搜索、预览和不含结果行的配置导入导出。嵌入应用管理页提供创建、编辑、启停、签名密钥轮换和删除。应用令牌绑定现有非管理员用户、工作空间与应用状态，不同应用的会话隔离。原版 SQLBot v1.10.2 页面和 SDK 仅在指定合成流程完成对照；复杂 SQL、全部页面、全部接入方式及真实业务零改动替换仍待验证。详见本版发布说明和验收记录。