

RockPipe 工具介绍

文档属性	内容
文档版本	V1.0.0-2026-08-01
对应软件版本	RockPipe V1.0.0-SNAPSHOT
适用对象	业务方、技术负责人、实施与运维、系统对接开发
文档类型	产品介绍 + 使用说明
配套说明	管理端「帮助文档」（应用内速查）；部署启停细则见发布包 / 安装说明

本文用[真实界面截图](#)说明 RockPipe 是什么、能做什么、如何上手。详细启停与升级步骤见[安装部署说明](#)。

目录

1. 一句话定位	
2. 为什么选 RockPipe	
3. 产品形态	
4. 五步上手 （须先完成 §7.0 首次安装 ）	
5. 功能亮点 （含界面）	
◦ 5.1 数据源	
◦ 5.2 消息队列	
◦ 5.3 任务编排	
◦ 5.4 执行与运维	
◦ 5.5 安全与治理	
6. 给业务系统对接 （MQ）	
◦ 6.1 消费时找不到任务 / 异常怎么处理	
7. 交付与部署 （摘要）	
◦ 7.0 首次安装与主密钥	
◦ 7.1 Windows 部署	
◦ 7.2 Linux 部署	
◦ 7.3 进阶：禁用向导 / CLI 初始化	
8. 备份与迁移	

- 9. [日常使用建议](#)
 - 10. [常见问题（速查）](#)
 - 11. [技术交流与支持](#)
 - 12. [术语](#)
-

1. 一句话定位

RockPipe 是一款**轻量消息驱动型数据管道工具**：在 Web 管理端把「跨库同步 / 批处理 / 调存储过程」配成可复用任务，通过 MQ / **定时** / **手动** 触发执行，并提供完整的日志、脏数据、死信与审计能力。

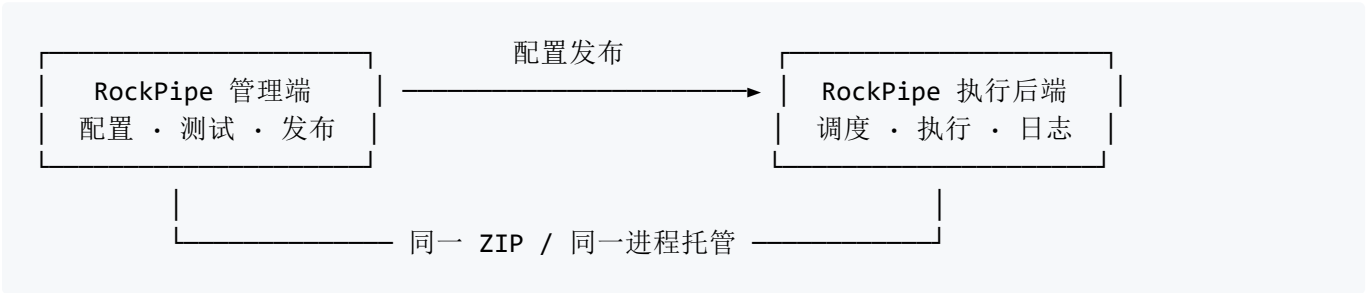
它不是大数据集群，也不是全功能 ETL / 实时流平台——而是企业数据流转的**轻量调度中枢**：支持 Windows / Linux 跨平台部署（运行仅需 JRE 17+），**ZIP 开箱即用**——解压启动后通过浏览器安装向导配置密钥与管理员即可；管理端与执行后端同包单进程托管，**无需**现场安装 Node、Maven 或单独部署 Nginx。

2. 为什么选 RockPipe

价值点	说明
跨平台·开箱即用	支持 Windows、Linux；ZIP 含 jar、预构建前端与启停脚本（Windows <code>startup.bat</code> / Linux <code>startup.sh</code> ，可选 systemd）；现场只需 JRE 17+，无需 Node / Maven / 独立前端部署
配置即可用	数据源、任务节点可视化编排；SQL 占位符助手降低编写成本； 消息队列按需配置 （仅 MQ 触发时需要）
多种触发	可选 RocketMQ 消息、定时 Cron、运维「立即执行」，可组合；不强制部署 MQ
可测可发	流程测试与正式发布分离，改完先测再上线
运维闭环	执行日志 → 节点详情 → 脏数据 / 死信 → 审计 / 健康检查
安全可控	登录认证、三角色权限、IP 白名单、密码策略、强制改密

典型场景： 订单状态变更后同步库表、日终批处理、跨库汇总、CALL 存储过程、MQ 驱动的轻量集成。

3. 产品形态



部分	面向谁	职责
管理端（Web）	运维 / 开发	数据源、MQ、任务编排、测试、发布、运维查询
执行后端	系统自动运行	消费触发、按节点执行、写日志与容错

浏览器访问管理端即可完成全流程配置；未登录不可操作。

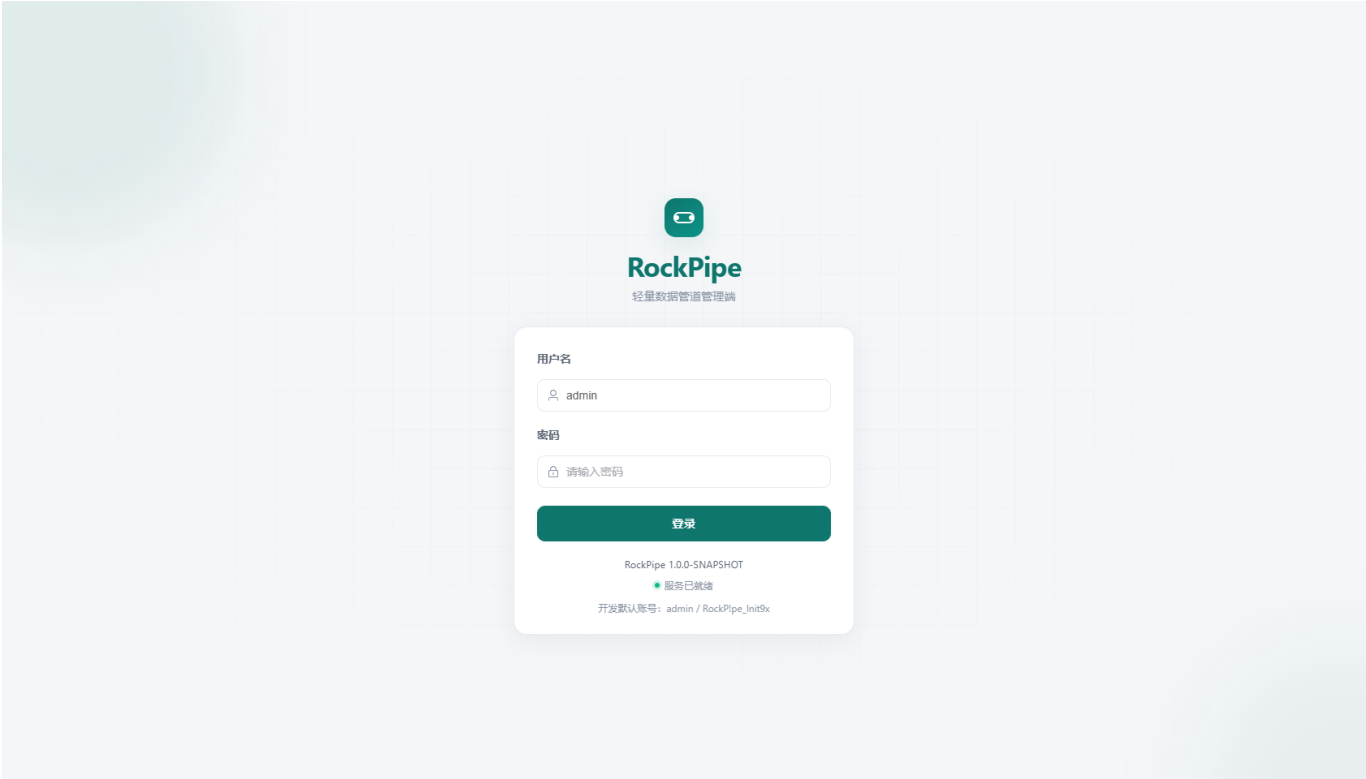


图 3-1 管理端登录（开发环境可提示默认账号；生产由现场下发账号）

4. 五步上手

配置数据源 → （可选）配置 MQ → 新建并编排任务 → 流程测试 → 发布启用 → 看执行日志

步骤	菜单	要做什么
1	数据源	配源库 / 目标库，测试连接后保存；可导出 / 导入
2	消息队列	仅 MQ 触发时需要；测通 NameServer / ACL
3	任务设置	新建 taskCode → 编排节点与触发方式 / 超时
4	流程测试 / 发布	先试跑（流式节点结果），再发布并启用
5	执行日志	核对结果；异常看脏数据 / 死信

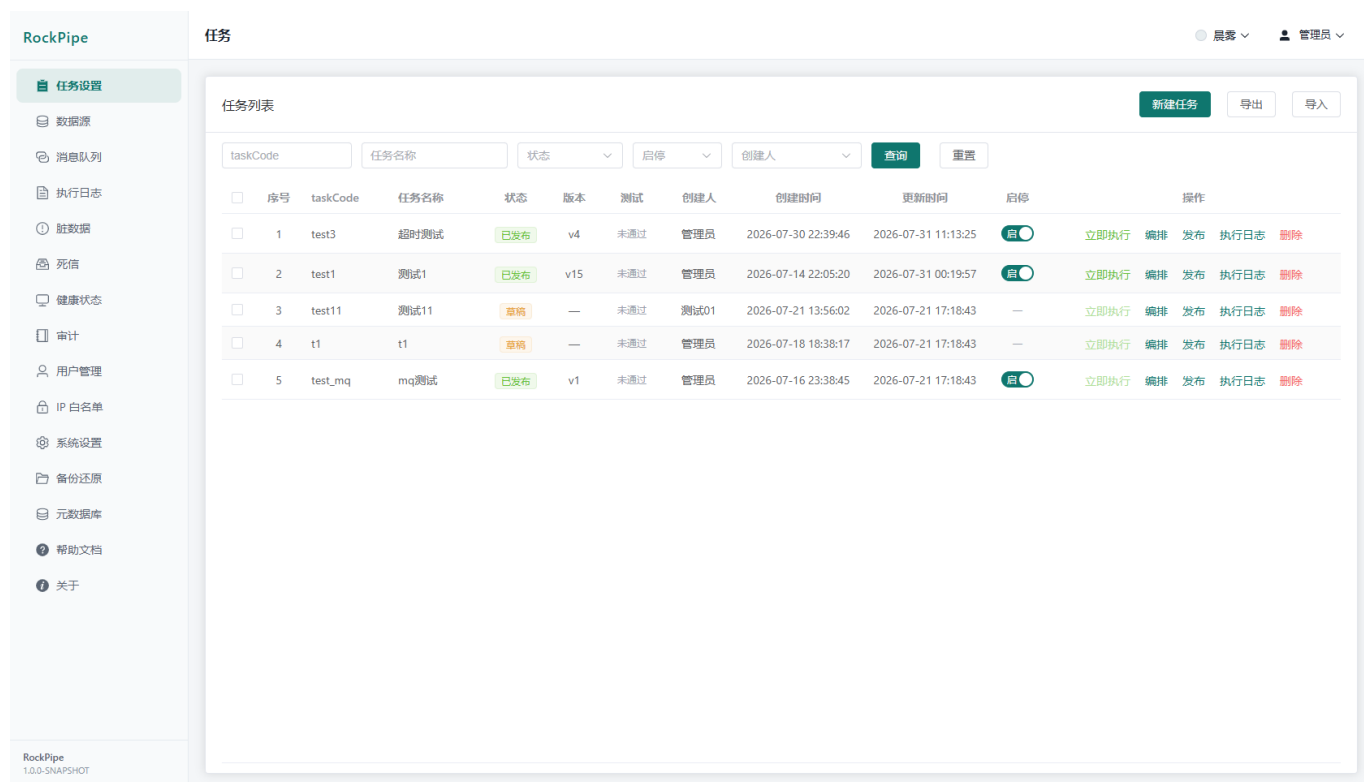


图 4-1 任务列表：工具栏「新建 → 导出 → 导入」；行内立即执行 / 编排 / 发布 / 执行日志

5. 功能亮点（含界面）

5.1 数据源——连上库再说

按库类型选 JDBC 模板，填写账号密码与连接池；**保存前可测试连接**。密码加密存储，编辑时留空表示不改。

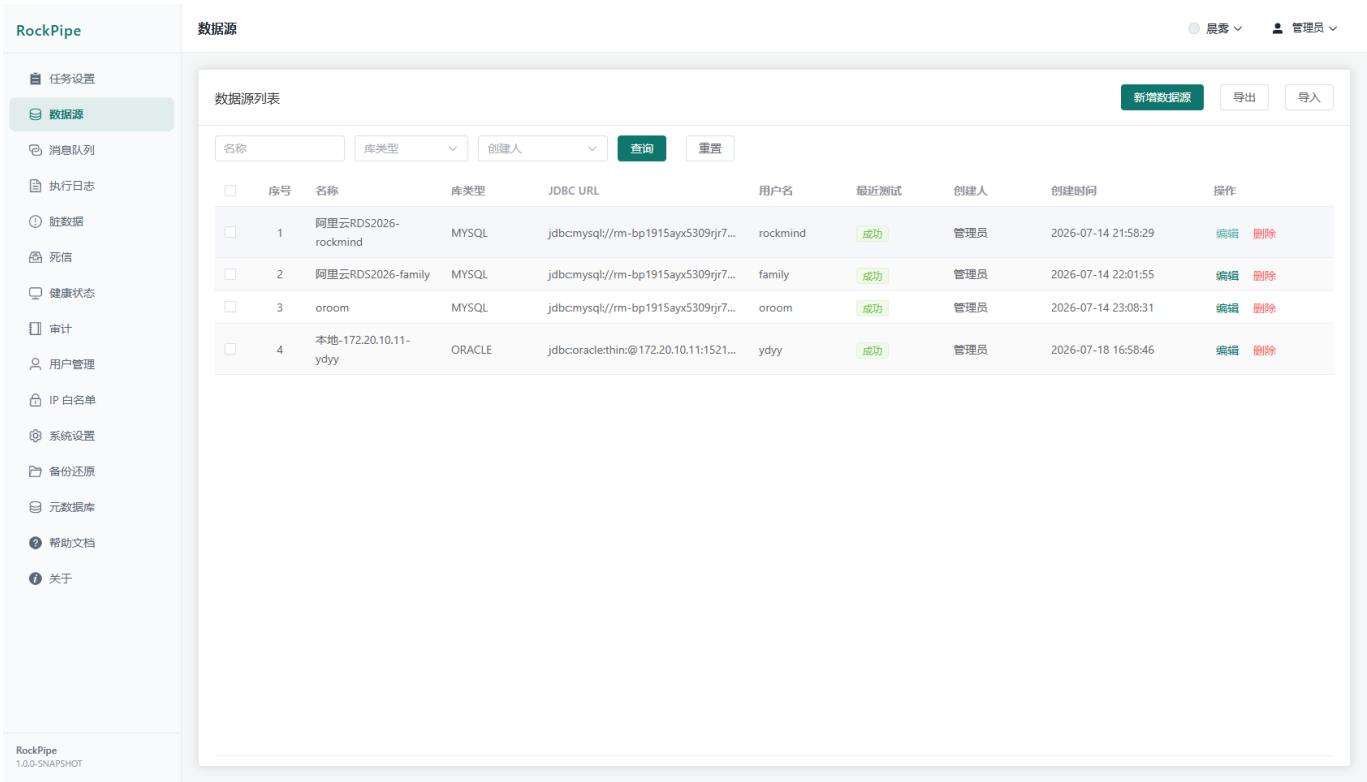


图 5-1 数据源列表

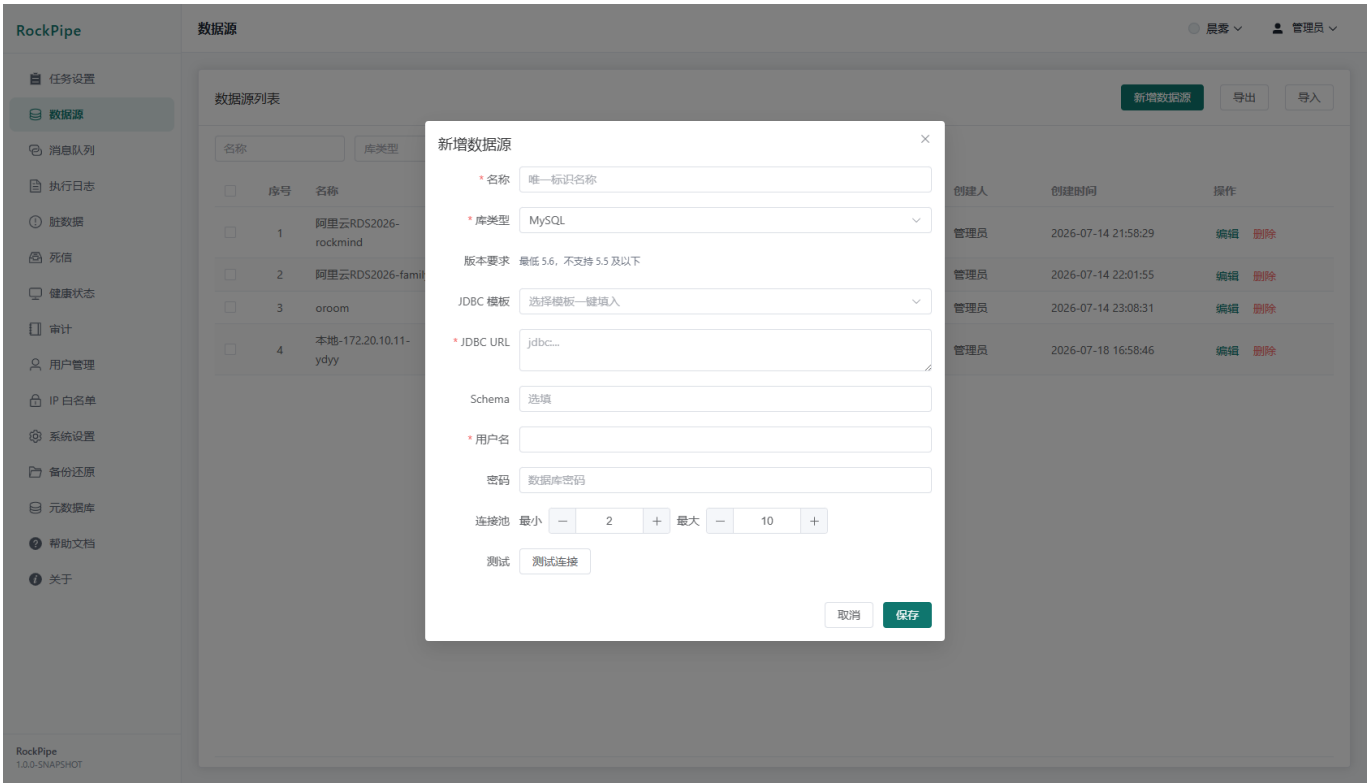


图 5-2 新增数据源：模板填 URL + 测试连接

5.2 消息队列——按需接入 RocketMQ（可选）

消息队列为可选项：仅当任务启用「MQ 触发」时需要配置。仅使用定时 / 手动触发时，可不配置、不部署 RocketMQ。

配置 NameServer；云 MQ 一般开启 ACL。任务编排中只展示实例名称，降低误配。

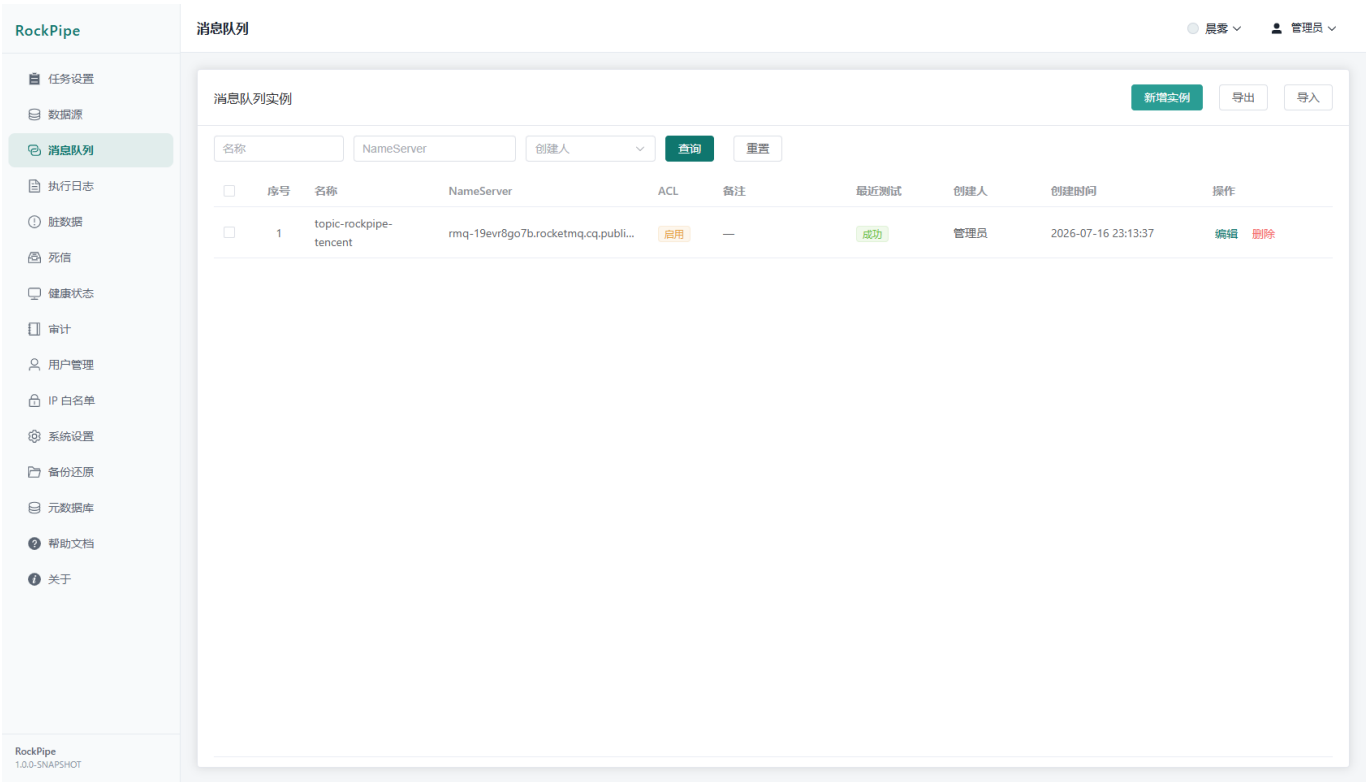


图 5-3 消息队列实例

5.3 任务编排——依赖条件 / 抽取 / 加载 / 执行 SQL

- **触发：**MQ（可选） / 定时 / 手动，可多选，至少一种；**仅勾选 MQ 时须配置实例与 Topic。**
- **节点：**依赖条件（可选） → 数据抽取 → 数据加载（可多个） → 执行 SQL（含 CALL/EXEC）。
- **依赖条件：**运行模式「依赖探测」（默认）或「条件判断」。探测：契约 `depend / ispass / msg` + 「按依赖重复执行」；条件：至少 `ispass`，未过为 `SKIPPED_CONDITION`。模式显式配置，不按节点位置自动切换。
- **执行超时：**任务级默认 30 分钟；超时取消当前 SQL 并以失败结束（正式任务可按重试配置重跑；流程测试不重试）。
- **草稿保存：**名称即可落库；完整校验在测试 / 发布时进行。
- **流程测试：**流式展示各节点结果与耗时，按钮旁有运行秒表；通过后可「去发布」。
- **占位符助手：**写 SQL 时预览 `${message.xxx}`、`${row.xxx}` 等替换效果。
- **列表工具栏：**新建 / 新增 → 导出 → 导入（数据源、消息队列、任务一致）。

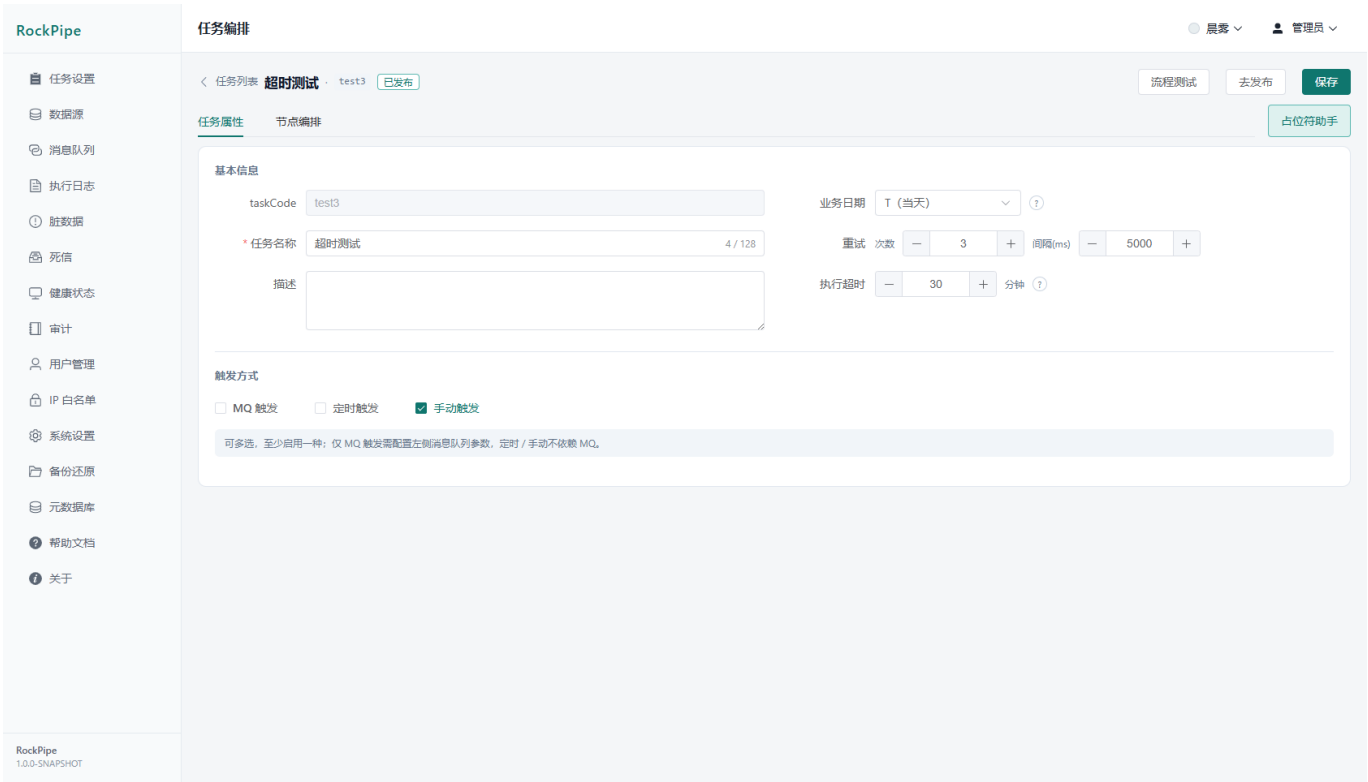


图 5-4 任务编排：任务属性（含执行超时）与节点编排；顶栏「流程测试 / 去发布 / 保存」

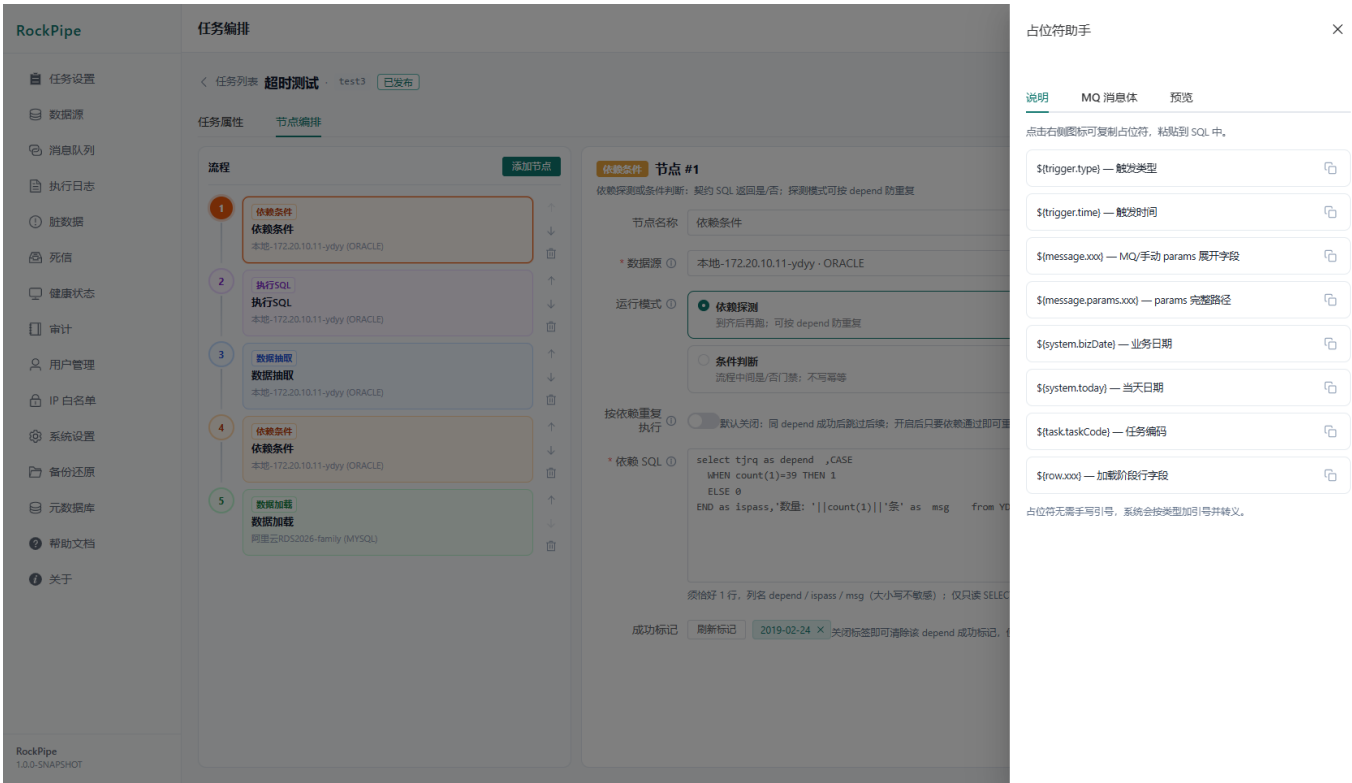


图 5-5 占位符助手：说明 / 消息体示例 / 预览

常用占位符（勿手写外层引号）：

占位符	含义
<code>\${trigger.type}</code> / <code>\${trigger.time}</code>	触发类型 / 触发时间
<code>\${message.xxx}</code>	MQ / 手动 params 字段
<code>\${message.params.xxx}</code>	params 完整路径
<code>\${system.bizDate}</code>	业务日期（按任务规则 T / T-1）
<code>\${system.today}</code> / <code>\${system.yesterday}</code>	当天 / 昨天
<code>\${row.xxx}</code>	加载节点当前行字段
<code>\${task.taskCode}</code>	任务编码

5.4 执行与运维——看得见、查得清

从任务行可直接打开**执行日志**抽屉（运行中会自动刷新）；运维菜单提供全局查询、脏数据、死信、健康与审计。

RockPipe

任务设置

数据源

消息队列

执行日志

脏数据

死信

健康状态

审计

用户管理

IP 白名单

系统设置

备份还原

元数据库

帮助文档

关于

RockPipe
1.0.0-SNAPSHOT

执行日志 · 超时测试 (test3)

状态

触发类型

查询

序号	taskCode	任务名称	版本	触发	状态	重试	时长(秒)	开始时间	结束时间	操作
1	test3	超时测试	v4	手动	成功	0	180	2026-07-31 00:14:16	2026-07-31 00:17:16	节点
2	test3	超时测试	v3	手动	成功	0	181	2026-07-31 00:07:09	2026-07-31 00:10:10	节点
3	test3	超时测试	v3	手动	成功	0	180	2026-07-31 00:00:06	2026-07-31 00:03:06	节点

Total 315/page<1>

图 5-6 任务侧「执行日志」抽屉（分页浏览；运行中自动刷新）

RockPipe

任务设置

数据源

消息队列

执行日志

脏数据

死信

健康状态

审计

用户管理

IP 白名单

系统设置

备份还原

元数据库

帮助文档

关于

RockPipe
1.0.0-SNAPSHOT

执行日志

晨露

管理员

执行日志

taskCode

任务名称

状态

触发类型

查询

序号	taskCode	任务名称	版本	触发	状态	重试	时长(秒)	开始时间	结束时间	操作
1	test3	超时测试	v4	手动	成功	0	180	2026-07-31 00:14:16	2026-07-31 00:17:16	节点
2	test1	测试1	v13	手动	成功	0	55	2026-07-31 00:07:57	2026-07-31 00:08:52	节点
3	test3	超时测试	v3	手动	成功	0	181	2026-07-31 00:07:09	2026-07-31 00:10:10	节点
4	test3	超时测试	v3	手动	成功	0	180	2026-07-31 00:00:06	2026-07-31 00:03:06	节点
5	test1	测试1	v13	手动	成功	0	97	2026-07-23 09:30:15	2026-07-23 09:31:52	节点
6	test1	测试1	v13	手动	失败	3	87	2026-07-23 08:52:32	2026-07-23 08:53:59	节点
7	test1	测试1	v13	手动	失败	2	86	2026-07-23 08:51:01	2026-07-23 08:52:27	节点
8	test1	测试1	v13	手动	失败	1	87	2026-07-23 08:49:29	2026-07-23 08:50:56	节点
9	test1	测试1	v13	手动	失败	0	87	2026-07-23 08:47:57	2026-07-23 08:49:24	节点
10	test1	测试1	v13	手动	成功	0	52	2026-07-21 13:46:06	2026-07-21 13:46:58	节点
11	test1	测试1	v13	手动	成功	0	51	2026-07-21 13:43:11	2026-07-21 13:44:02	节点
12	test1	测试1	v12	手动	成功	0	12	2026-07-18 18:25:36	2026-07-18 18:25:48	节点
13	test1	测试1	v12	手动	成功	0	12	2026-07-18 18:15:35	2026-07-18 18:15:47	节点
14	test_mq	mq测试	v1	MQ	成功	0	0	2026-07-16 23:48:02	2026-07-16 23:48:02	节点
15	test_mq	mq测试	v1	MQ	成功	0	0	2026-07-16 23:46:59	2026-07-16 23:46:59	节点

Total 33

15/page

< 1 2 3 >

图 5-7 全局执行日志：状态、触发类型、节点详情

RockPipe

任务设置

数据源

消息队列

执行日志

脏数据

死信

健康状态

审计

用户管理

IP 白名单

系统设置

备份还原

元数据库

帮助文档

关于

RockPipe
1.0.0-SNAPSHOT

脏数据

晨露

管理员

脏数据

taskCode

executionId

节点名称

查询

导出 CSV

序号	taskCode	执行 ID	节点	错误原因	记录时间	数据
No Data						

图 5-8 脏数据：加载批次失败可导出排查

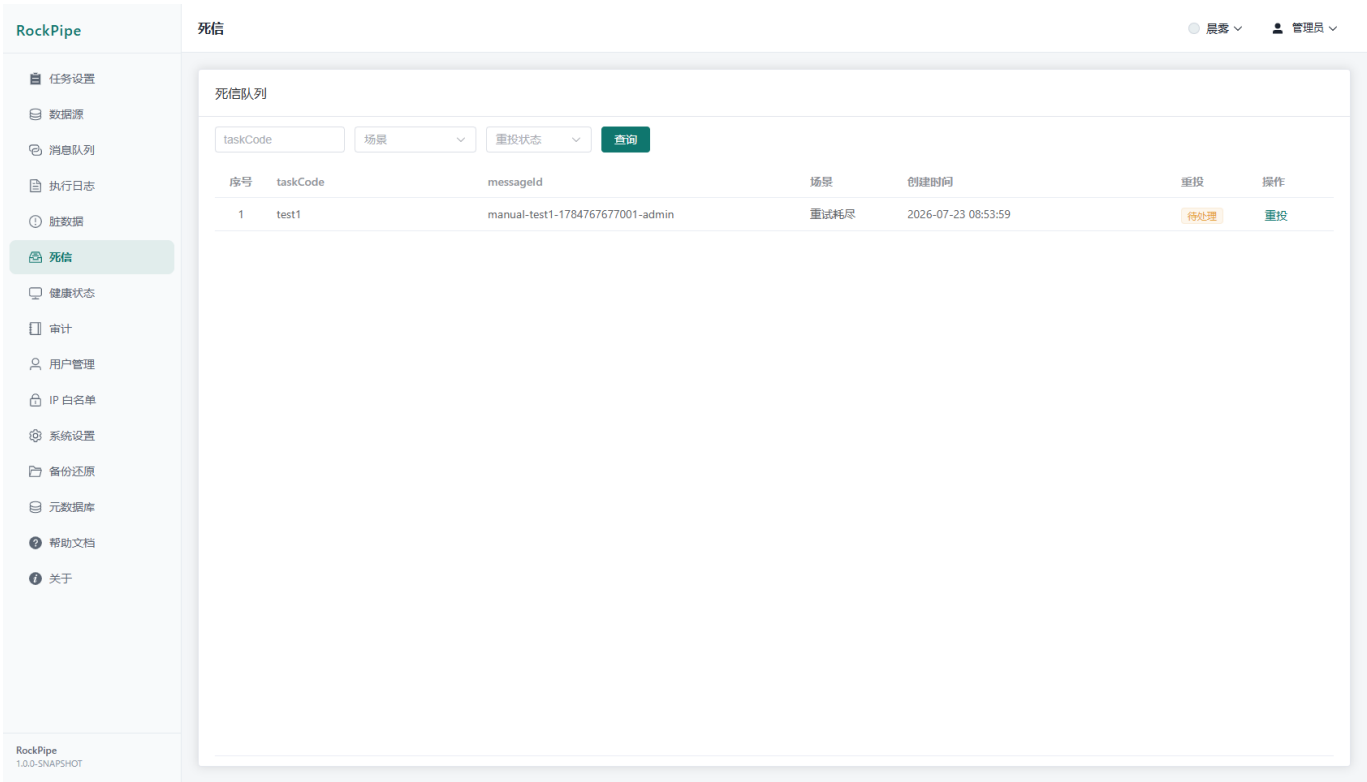


图 5-9 死信：排队超限 / 重试耗尽等，可按策略重投

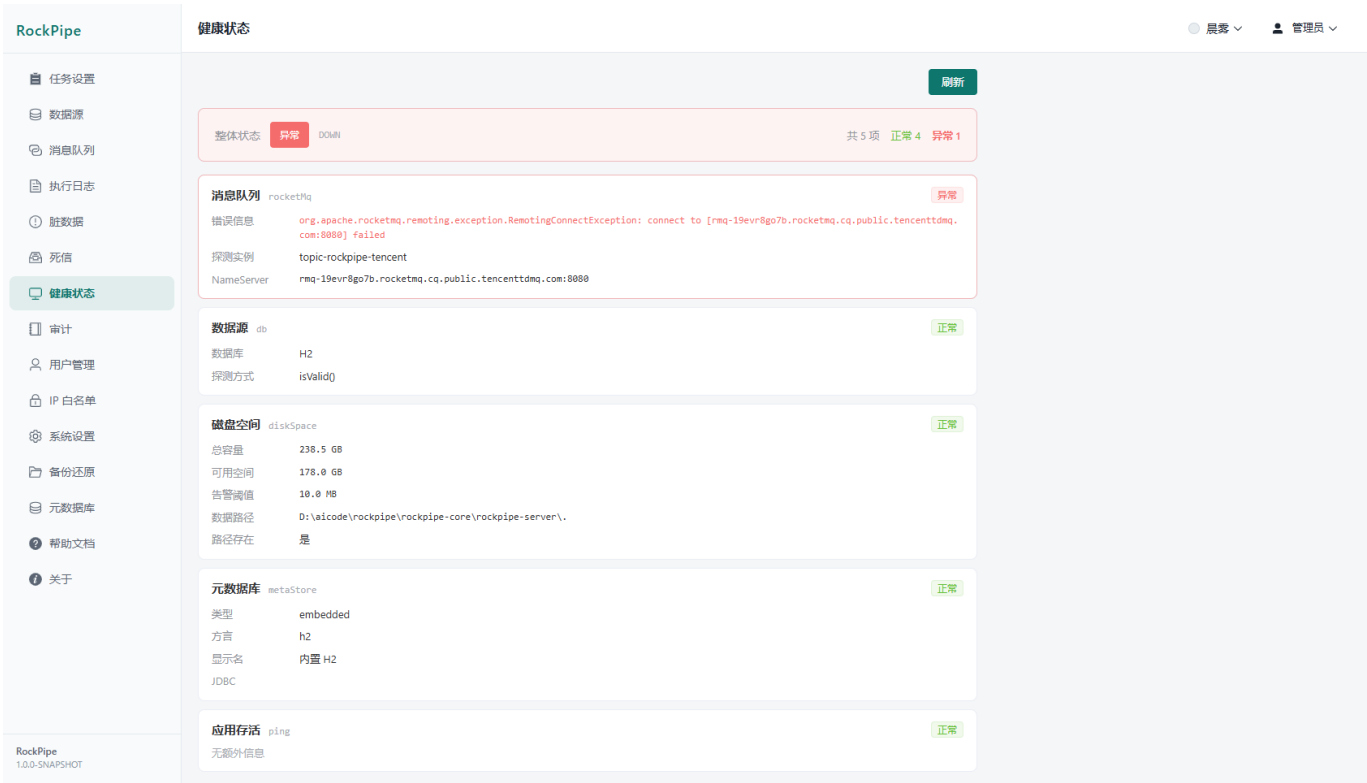


图 5-10 健康状态：进程 / H2 / MQ 探活

RockPipe

任务设置

数据源

消息队列

执行日志

脏数据

死信

健康状态

审计

用户管理

IP 白名单

系统设置

备份还原

元数据库

帮助文档

关于

RockPipe
1.0.0-SNAPSHOT

审计

晨露 管理员

审计日志

操作人

操作类型

查询

序号	时间	操作人	操作类型	目标	客户端 IP	结果	详情
1	2026-08-04 17:13:45	admin	LOGIN	admin	127.0.0.1	SUCCESS	
2	2026-08-04 17:11:22	admin	LOGIN	admin	127.0.0.1	SUCCESS	
3	2026-07-31 11:13:25	admin	TASK_NODES_SAVE	test3	127.0.0.1	SUCCESS	{"nodeCount":5}
4	2026-07-31 11:13:25	admin	TASK_UPDATE	test3	127.0.0.1	SUCCESS	{"id":7}
5	2026-07-31 11:13:04	admin	TASK_NODES_SAVE	test3	127.0.0.1	SUCCESS	{"nodeCount":5}
6	2026-07-31 11:13:04	admin	TASK_UPDATE	test3	127.0.0.1	SUCCESS	{"id":7}
7	2026-07-31 11:07:23	admin	LOGIN	admin	127.0.0.1	SUCCESS	
8	2026-07-31 00:19:57	admin	TASK_PUBLISH	test1	127.0.0.1	SUCCESS	{"versionNo":15,"status":"PUBLISHED"}
9	2026-07-31 00:19:39	admin	TASK_NODES_SAVE	test1	127.0.0.1	SUCCESS	{"nodeCount":7}
10	2026-07-31 00:19:39	admin	TASK_UPDATE	test1	127.0.0.1	SUCCESS	{"id":1}
11	2026-07-31 00:19:31	admin	TASK_PUBLISH	test1	127.0.0.1	SUCCESS	{"versionNo":14,"status":"PUBLISHED"}
12	2026-07-31 00:19:02	admin	LOGIN	admin	127.0.0.1	SUCCESS	
13	2026-07-31 00:14:16	admin	TASK_MANUAL_TRIGGER	test3	127.0.0.1	SUCCESS	{"forceDependRepeat":false,"messageId":...
14	2026-07-31 00:14:02	admin	TASK_PUBLISH	test3	127.0.0.1	SUCCESS	{"versionNo":4,"status":"PUBLISHED"}
15	2026-07-31 00:07:57	admin	TASK_MANUAL_TRIGGER	test1	127.0.0.1	SUCCESS	{"forceDependRepeat":false,"messageId":...
16	2026-07-31 00:07:09	admin	TASK_MANUAL_TRIGGER	test3	127.0.0.1	SUCCESS	{"forceDependRepeat":true,"messageId":...

Total 320 < 1 2 3 4 5 6 ... 16 >

图 5-11 审计：关键配置与触发操作留痕

5.5 安全与治理——可管、可控

三角色（超管 / 管理员 / 普通用户）、IP 白名单、系统标题配置；右上角支持个人信息与修改密码（弹窗交互）。

RockPipe

任务设置

数据源

消息队列

执行日志

脏数据

死信

健康状态

审计

用户管理

IP 白名单

系统设置

备份还原

元数据库

帮助文档

关于

RockPipe
1.0.0-SNAPSHOT

用户管理

晨露 管理员

用户列表

新建用户

序号	账号	姓名	角色	部门	手机	邮箱	内置	操作
1	admin	管理员	超级管理员	—	13527437510	3711@11.com	是	—
2	test	test01	普通用户	测试	—	—	—	编辑 设置密码 删除
3	test01	测试01	普通用户	—	—	—	—	编辑 设置密码 删除

图 5-12 用户管理

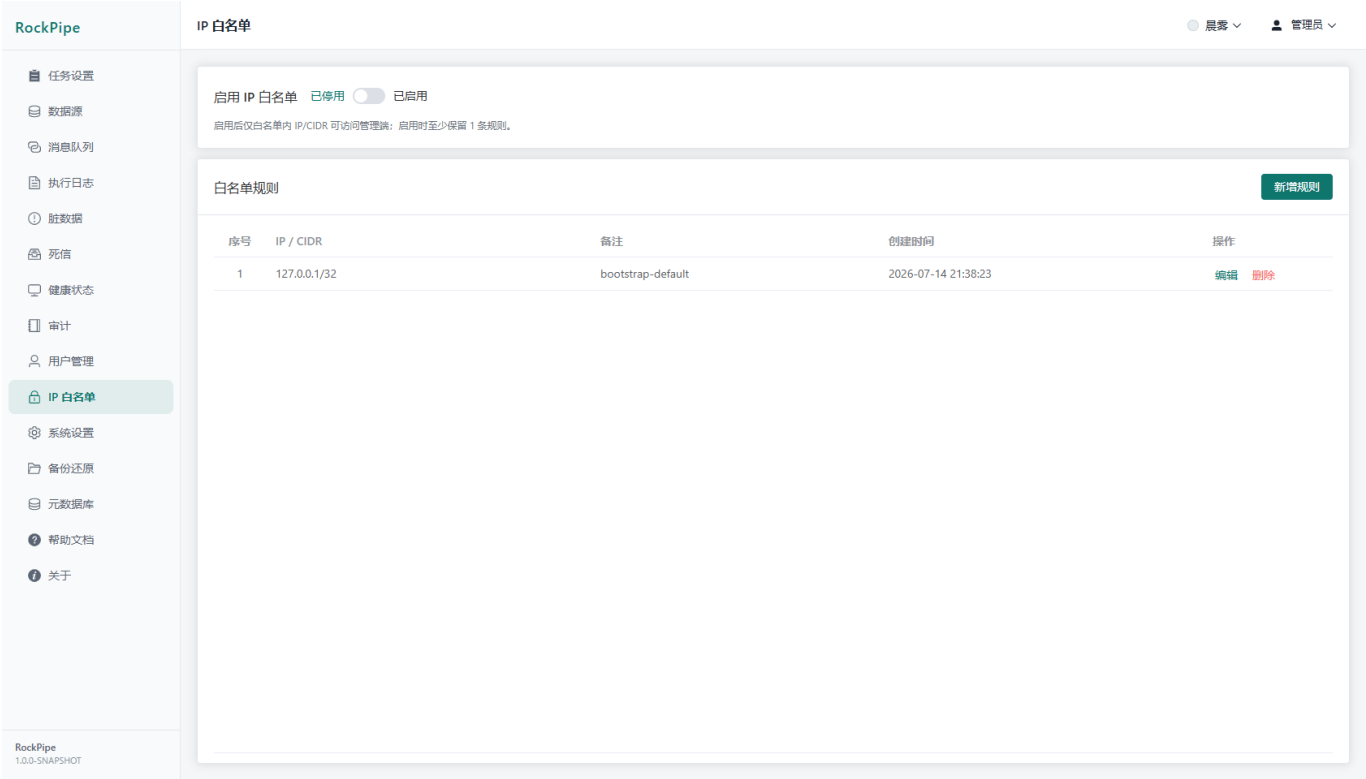


图 5-13 IP 白名单

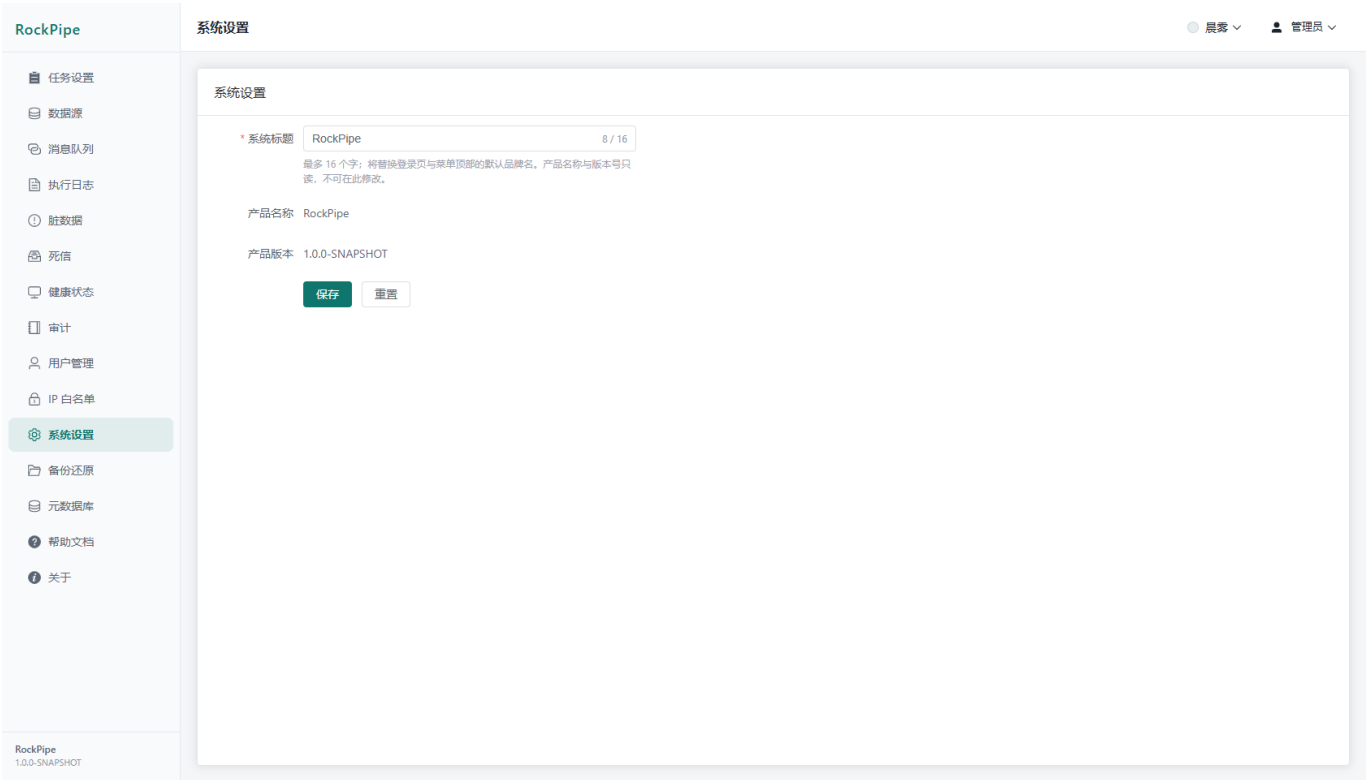


图 5-14 系统设置（超管）

能力	说明
个人信息	姓名、部门、手机、邮箱可改；账号与角色只读
修改密码	实时规则校验；成功后会话失效须重新登录
强制改密	新建 / 重置密码后，对方须先改密再进业务

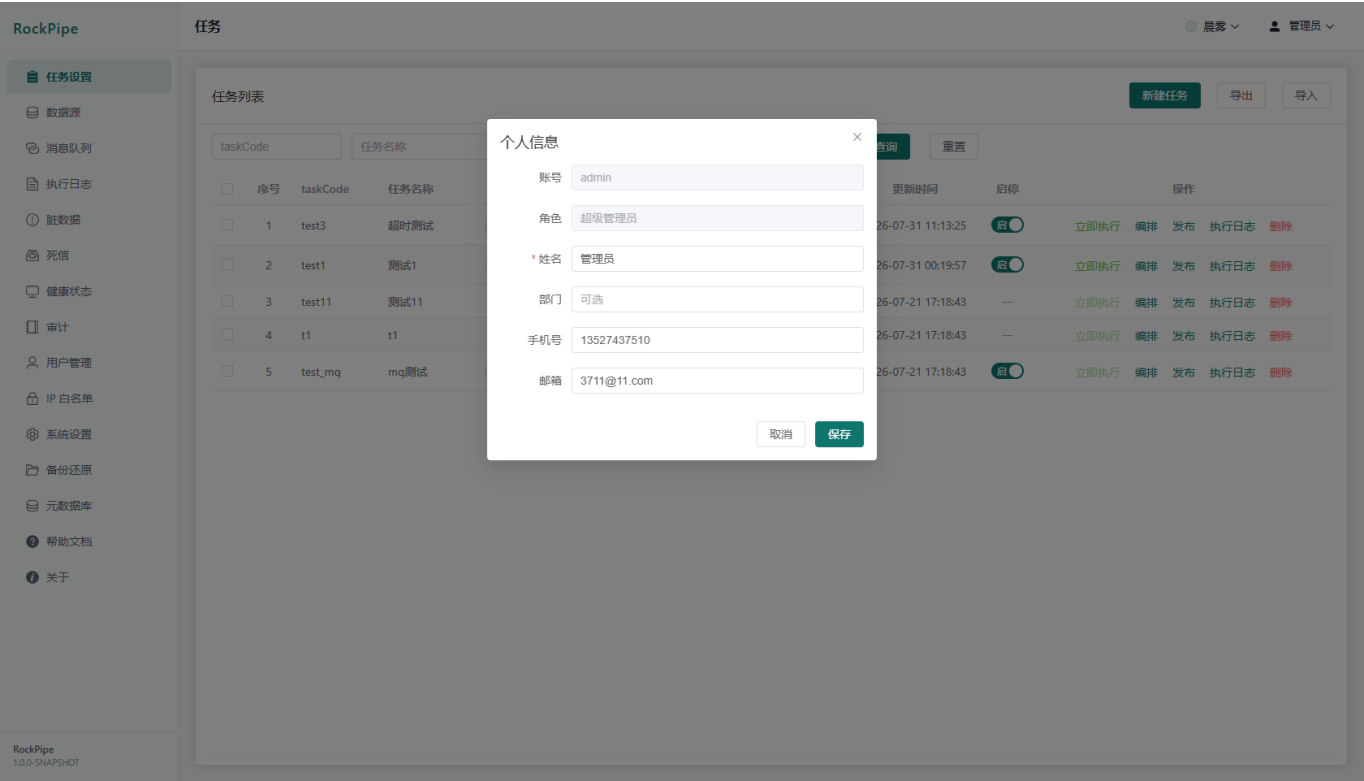


图 5-15 个人信息弹窗

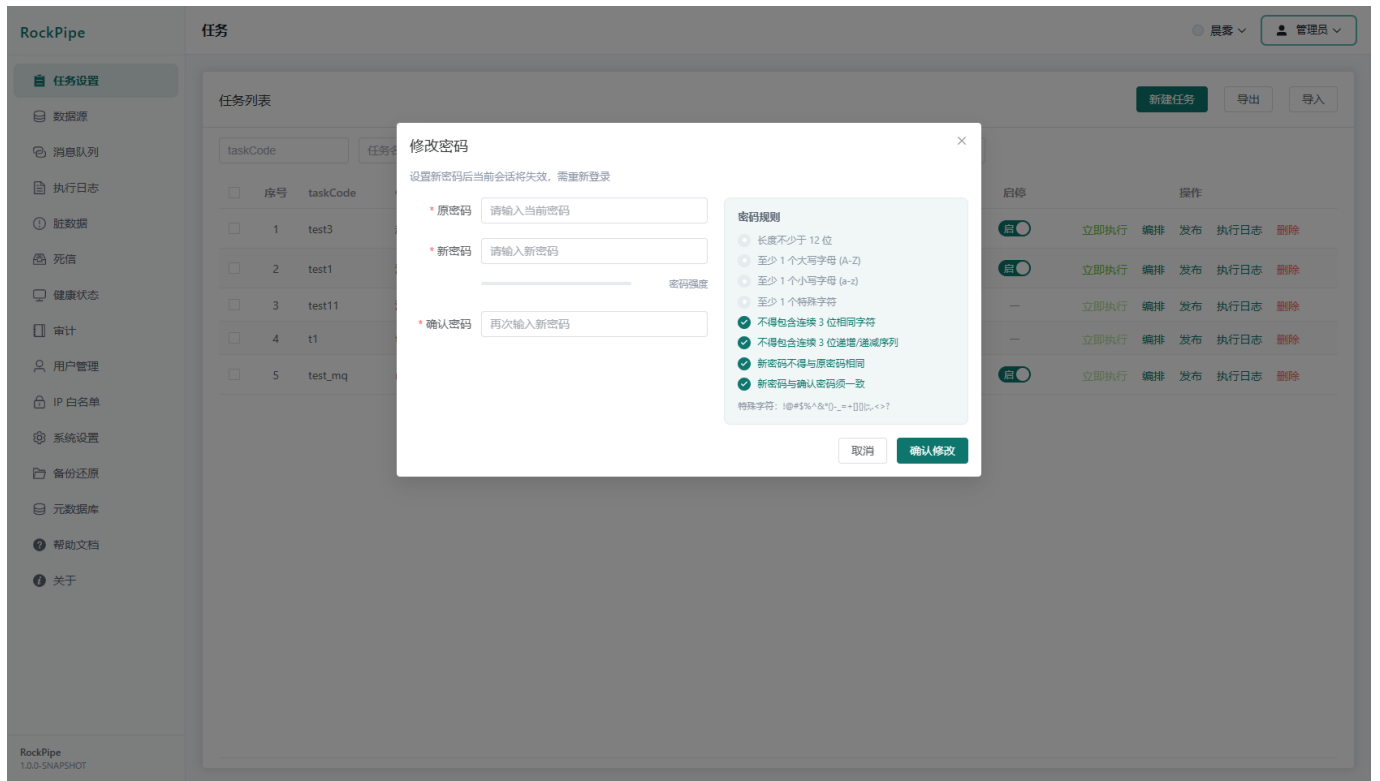


图 5-16 修改密码弹窗（规则实时反馈）

6. 给业务系统对接（MQ）

向任务 Topic 发送 JSON，taskCode 必须与已发布且启用的任务一致：

```
{
  "messageType": "TASK_EXECUTE",
  "taskCode": "your_task_code",
  "triggerSource": "MQ",
  "testMode": false,
  "params": {
    "orderId": 12345,
    "status": "PAID"
  },
  "sourceSystem": "order-service"
}
```

- SQL 中用 `${message.orderId}` 引用 `params` 字段；
- **禁止**在消息体中传递数据库密码、MQ SecretKey 等凭证；
- 编排页「占位符助手 → MQ 消息体」可一键复制带当前 taskCode 的示例。

仅定时 / 手动触发时，可不配置 MQ。

6.1 消费时找不到任务 / 异常怎么处理

每个已发布且启用了 MQ 触发的任务会单独注册 Consumer（订阅该任务的 Topic + Consumer Group）。处理规则如下：

情况	系统行为	是否 ACK（确认消费）
任务已发布、已启用，消息正常	入队执行；成功或进死信后再 ACK	延后 ACK
任务已发布但已停用	丢弃、不执行、不入队，记日志	是（避免积压）
本任务 Consumer 仍在，但本地无有效发布快照（短暂不一致等）	记 warn，返回稍后重试	否（ RECONSUME_LATER ）
消息 JSON 解析失败	记死信摘要，稍后重试	否
消息体未带 taskCode	用该 Consumer 绑定的 taskCode 补全	按后续逻辑继续
没有对应已发布 MQ 任务（未发版 / 未开 MQ 触发）	RockPipe 不会订阅该 Topic，本系统不拉消息	不适用（消息留在 Broker）

说明：业务侧应把消息发到目标任务自己的 Topic，且 taskCode 与该任务一致。未配置消费者的 Topic 上的消息不会被 RockPipe「当成功消费掉」。

7. 交付与部署（摘要）

项	说明
交付物	ZIP 发布包 (jar + 预构建前端 + Windows / Linux 启停脚本)
支持系统	Windows、Linux (跨平台; 运行依赖 JRE 17+)
运行依赖	JRE 17+; 业务源/目标库; RocketMQ 可选 (仅 MQ 触发时需要 NameServer / 兼容云 MQ)
访问地址	默认 <code>http://<主机>:8088</code>
业务数据	安装目录 <code>data/</code> (含 H2 与 <code>master.key</code>)
外置配置	<code>backend/config/application.yml</code>

推荐开箱路径 (两端相同) :

1. 安装 JRE 17+, 解压 ZIP, 执行 `startup.bat` / `startup.sh` ;
2. 浏览器打开 `http://主机:8088` (启动脚本打印的入口; 远程把 localhost 换成主机 IP) ;
3. 未完成安装时会自动进入向导: 主密钥 (可生成)、管理员密码 (用户名固定 `admin`) 、IP 白名单 (默认启用并带上当前访问 IP) ;
4. 用刚设的密码登录; 请妥善备份 `data/master.key` (或环境变量中的密钥) 。

升级 / 跨机还原时主密钥不可随意更换。CLI `encrypt` + `password-enc` 仅作进阶/自动化装机, 见安装目录 README。

7.0 首次安装与主密钥

何时出现: 生产 ZIP 首次启动后, 若安装目录尚无 `data/master.key` (且未设置 `ROCKPIPE_CRYPTO_KEY`) 或尚无管理员, 浏览器打开 `http://<主机>:8088` 会**自动进入** Web 安装向导 (`/setup`), 无法进入任务等业务页。



1 主密钥 ①

* 主密钥 (Base64)

AA=

①

自动生成

复制

☒ 我已安全备份主密钥 ①

2 管理员账号 ①

用户名

admin

①

* 密码

.....



①

* 确认密码

.....



3 IP 白名单 ①

启用白名单 ①



* 白名单规则

10.5.50.32

①

删除

新增规则

完成安装并进入登录

安装遇到问题? 扫码寻求技术支持



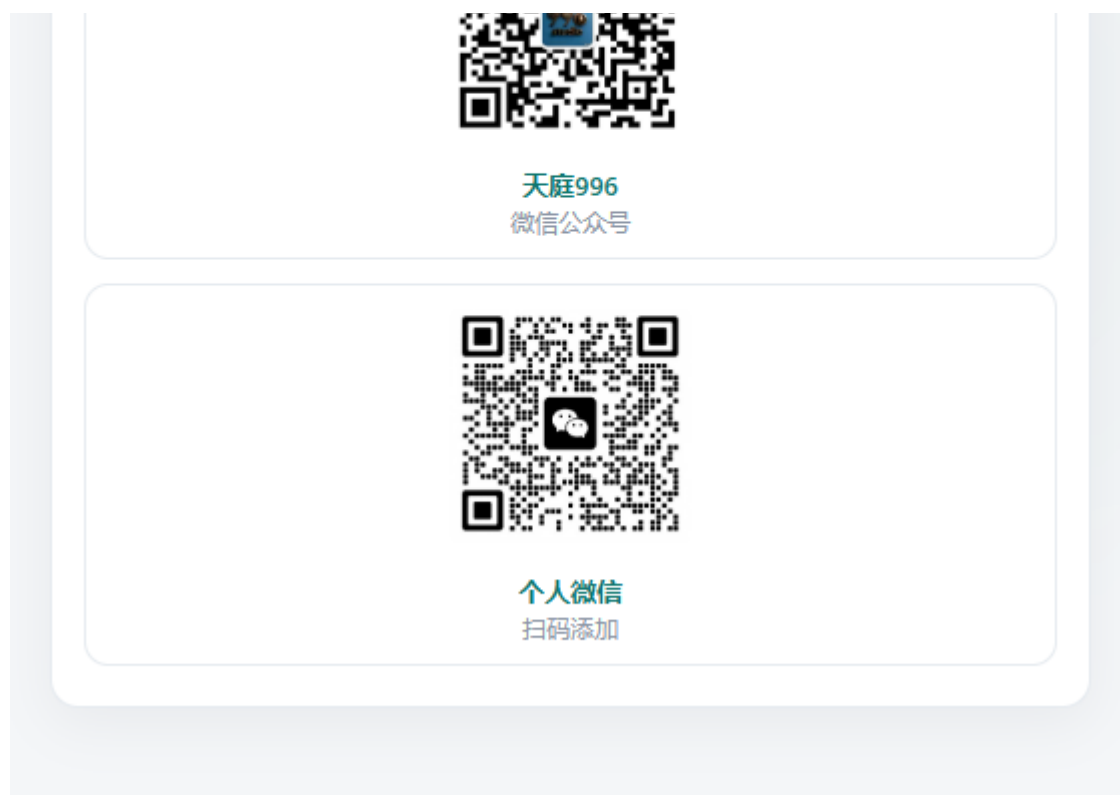


图 7-0 安装向导：① 主密钥 → ② 管理员 → ③ *IP* 白名单 → ④ 安装确认（须同意《安装与使用声明》）

向导步骤	界面要点	运维须知
① 主密钥	「自动生成」或粘贴 Base64；勾选「我已安全备份主密钥」	AES-256-GCM 主密钥（Base64 编码的 32 字节）。加密数据源口令、MQ Secret、备份 ZIP 内密文； 丢失后无法解密历史数据
② 管理员	用户名固定 <code>admin</code> ，设置密码与确认密码	超级管理员，用于首次登录；与主密钥不同（登录密码为 BCrypt，不可逆）
③ IP 白名单	默认启用，预填当前访问 IP，可增删 CIDR	远程安装务必包含运维出口 IP，否则安装后可能被拦在登录外；完成后可在「IP 白名单」修改
④ 安装确认	勾选《安装与使用声明》	开源友好表述：按「现状」提供，自行负责部署与密钥/数据安全；适用法律允许范围内项目贡献者不承担误用或环境因素导致的赔偿责任；文档/Issue 等可提供技术支持协助，非 SLA 承诺

安装完成后密钥落盘：

方式	说明
<code>data/master.key</code>	向导默认写入安装目录 <code>data/</code> （单行 Base64）， 请与 H2 数据目录一并备份
<code>ROCKPIPE_CRYPTO_KEY</code>	环境变量注入同一 Base64 密钥； 优先级高于 <code>master.key</code> ，适合 systemd / 容器
密钥指纹	「备份还原」导入前会比对指纹；目标环境密钥必须与导出环境 一致

主密钥使用边界（易混点）：

- **需要同一密钥：** 跨机迁移、还原系统备份、复制含密文的配置包、升级后沿用原 `data/`。

- **不要随意更换**：换密钥后旧 `ENC(...)` 密文无法解密，须专业「双钥重加密」流程（MVP 无一键轮换）。
- **与 JDBC 执行习惯**：节点 SQL（含 TRUNCATE）经 JDBC 执行，**句末不要加分号**；PL/SQL 客户端能跑不代表编排里可带 `;`（Oracle 常见 `ORA-00911`）。

禁用向导（进阶）：设置 `ROCKPIPE_DISABLE_SETUP=true` 时须自行准备 `ROCKPIPE_CRYPT0_KEY` 或 `master.key`，并用 `encrypt` 生成 `application.yml` 中的 `password-enc`，否则无法启动。详见 [7.3](#)。

7.1 Windows 部署

解压后进入安装根目录。多 JDK 时可设：

```
$env:JAVA_HOME = 'C:\Program Files\Java\jdk-17'
# 或：$env:ROCKPIPE_JAVA = 'C:\Program Files\Java\jdk-17\bin\java.exe'
.\startup.bat
# 成功后访问：http://localhost:8088（未安装则自动进入向导）
.\shutdown.bat
```

脚本会等待健康检查（默认最长 90s，`ROCKPIPE_START_WAIT` 可调）。

7.2 Linux 部署

```
export JAVA_HOME=/usr/lib/jvm/java-17-openjdk # 按实际路径；须为 17+
chmod +x startup.sh shutdown.sh
./startup.sh
# 成功后访问：http://localhost:8088（远程换成主机 IP；未安装则自动进入向导）
./shutdown.sh
```

若 `./startup.sh` 报 `No such file or directory`，多为 CRLF：`sed -i 's/\r$//' startup.sh shutdown.sh rockpipe.service`。

systemd（长期运行）：安装到 `/opt/rockpipe` 后复制 `rockpipe.service`，确认 `WorkingDirectory` 与 Java 路径；**首次建议先用脚本启动并完成 Web 安装**，再改用 `systemd`（密钥已在 `data/master.key`）。同一目录勿混用脚本与 `systemd`。

```
sudo systemctl enable --now rockpipe
journalctl -u rockpipe -f
```

8. 备份与迁移

RockPipe 提供两套配置迁移能力（均通过 Web 管理端操作，包格式为 ZIP： `manifest.json` + `payload.json`）。

8.1 系统备份 / 还原（仅超管）

- 入口：侧栏「备份还原」。
- 默认包含：用户、数据源、MQ、任务（草稿 + 最新发布快照）、 `taskCode` 登记、系统设置、IP 白名单。
- **不含**：执行日志、脏数据、死信、审计、会话。
- **密钥要求**：包内保留 `passwordEnc` / `secretKeyEnc` 等密文；目标环境必须使用**同一** `ROCKPIPE_CRYPTO_KEY`。还原前会校验密钥指纹，不一致则失败且不写入。
- **冲突策略**：用户按 `username` 存在则跳过；数据源 / MQ 按 `name` 覆盖；任务按 `taskCode` 整任务替换（草稿节点 + 最新发布 + registry）。
- **IP 白名单**：默认**合并**（保留本地规则，仅追加备份中缺失的 CIDR；启用状态取本地或备份任一侧开启），并自动补上当前操作者访问 IP，避免跨机还原锁出。可选「完全替换」（仍会自动补当前 IP）。关闭「还原 IP 白名单」则完全不改本地规则。

8.2 资源导入 / 导出（普通用户）

- 入口：数据源 / 消息队列 / 任务列表工具栏「导出」「导入」。
- 仅能操作自己 `createdBy` 的资源；导出勾选行，或未勾选时导出本人全部。
- 导出任务时：依赖的数据源 / MQ **仅当也是自己创建**才打进包，否则只写引用名；导入时若目标环境缺少对应 `name` 则失败并列出缺失项。
- 导入任务包内依赖时：目标环境**尚无**该 `name` → 创建并归属当前用户；**本人已有** → 覆盖配置；**他人已有同名** → **按名称复用**（不覆盖、不报错），便于导入超管导出的任务并挂到已有公共数据源 / MQ 上。
- 导入任务时：若同 `taskCode` 曾被逻辑删除，会**复活该行并覆盖配置**（归属改为当前导入用户），避免唯一约束冲突；未删除且属他人的 `taskCode` 仍会拒绝。

8.3 外置元数据库（仅超管）

- 入口：侧栏「元数据库」。
- 默认：内置 H2（`data/`）。可将整库迁移至 MySQL 8 / MariaDB 或 PostgreSQL 14+。
- 流程：测试连接 → 预检 → 维护模式全量迁移 → 写入 `backend/config/meta-store.yml` → **重启生效**。
- 迁移成功后页面以 `meta-store.yml` 为准展示外置库（待重启 / 已外置），并禁止再次迁移；本进程在重启前仍连原库。

- 一期不支持一键切回 H2；回退请用迁移前 `data/` 备份，并将指向改为 `type: embedded`（或删除外置配置）后重启。
 - 外置后请用 `mysqldump` / `pg_dump` 备份目标库；仍建议单实例调度。
-

9. 日常使用建议

1. **先测后发**：改 SQL 或节点顺序后，先流程测试，再发布。
 2. **改连接先测通**：数据源 / MQ 保存前点「测试连接」。
 3. **停用再删**：下线任务先停用，确认无运行实例再删除；taskCode **手工新建**不可复用，但可通过「任务导入」复活已删除的同编码任务。
 4. **对账三板斧**：执行日志（整体）→ 节点详情（哪一步）→ 脏数据 / 死信（残留）。
 5. **单实例访问 H2**：同一套 `data/` 不要多开后端进程。
 6. **迁移带密钥**：跨环境还原必须同步 `ROCKPIPE_CRYPT0_KEY`，否则密文无法解密。
-

10. 常见问题（速查）

问题	处理思路
MQ 测试报 <code>request type 106 not supported</code>	多为云 MQ 不支持管理接口；确认 NameServer / ACL，使用当前版本的路由探测
测试成功但「最近测试」仍未测试	新建时先测试再保存，保存前勿改连接字段
任务一直不执行	是否发布且启用；触发方式 / Topic / Group / taskCode；定时是否在生效窗口
MQ 消息发了但任务不动 / 一直重试	见 6.1：无消费者则消息留 Broker；无快照或解析失败不 ACK 会重投；停用则 ACK 丢弃不执行
<code>Database may be already in use</code>	结束占用同一 H2 的旧进程后再启
生产启动报缺少密钥 / 解密失败	优先用 <code>/setup</code> 安装向导；或确认 <code>data/master.key</code> / <code>ROCKPIPE_CRYPT0_KEY</code> 与 <code>password-enc</code> 匹配
打开页面一直进安装向导	完成向导后勿删除 <code>data/master.key</code> ；确认已成功创建管理员账号
<code>./startup.sh</code> 提示已启动但实际起不来	新版脚本会等待健康检查；看 <code>logs/stdout.log</code> 尾部
admin 忘密	安装目录执行 <code>reset-admin.bat</code> / <code>reset-admin.sh</code> ，重启后登录（见 README）
还原提示密钥指纹不匹配	确认目标环境 <code>ROCKPIPE_CRYPT0_KEY</code> 与导出环境一致后再还原

11. 技术交流与支持

欢迎通过以下渠道进行问题收集与技术交流。

渠道	说明
天庭996 微信公众号	产品动态与交流
个人微信	扫码添加，便于一对一沟通



天庭996 微信公众号 | 个人微信（扫码添加）

12. 术语

术语	含义
taskCode	任务唯一编码，匹配消息与追溯的依据
发布	将配置固化为执行引擎可加载的版本
内存管道	抽取结果在进程内传给后续加载节点
脏数据	加载某批失败时落库的行级错误
死信	无法正常完成调度 / 重试后的问题消息
占位符	SQL 中 <code>\${...}</code> 形式的参数引用
主密钥	AES 安装级密钥，Base64 32 字节；存于 <code>data/master.key</code> 或 <code>ROCKPIPE_CRYPT0_KEY</code>

文档版本 *V1.0.0-2026-08-01*，对应软件 *RockPipe V1.0.0-SNAPSHOT*。界面截图来自管理端实际运行环境；主题皮肤可能因用户选择略有差异，功能一致。如现场流程与本文不一致，以当前软件版本为准。