

连接器操作手册

1. 简介

连接器管理模块主要用于配置和管理系统间的 API 接口集成。通过可视化的路由配置、灵活的字段映射（支持 AI 智能生成）以及数据转换规则，实现异构系统间的数据对接与流转。此外，模块还提供了在线路由测试与全链路请求日志功能，便于快速调试与排查问题。

The screenshot shows the 'Router Configuration' page under the 'Connector' management module. The page includes a sidebar with navigation links: 'Router Management' (selected), 'Request Log', and 'Router Test'. The main content area has tabs: 'Router Configuration' (selected), 'Request Log', and 'Router Test'. It features a search bar with fields for 'Router Name' and 'Status' (set to 'All'), and buttons for 'Search' and 'Reset'. A table lists two router configurations:

Router Name	Channel	Source Path	Target Address	Method	Status	Created Time	Operations
Test	60	/externalApi/warehouseTransferInNotify	http://localhost:8084/warehouse.wms.outward.a...	POST	Enabled	2026-01-23 11:18:18	Edit Copy History Delete
Warehouse Transfer In Log System	60	/externalApi/warehouseTransferInNotify	http://localhost:8084/warehouse.wms.outward.a...	POST	Enabled	2026-01-22 19:24:09	Edit Copy History Delete

At the bottom, there are pagination controls: '2 items' (共2条), a page number '1' (1), and a '10 items/page' dropdown.

2. 路由配置

路由配置是连接器的核心功能，用于定义从“源系统”到“目标系统”的请求转发规则。

2.1 路由列表

进入 [连接器管理] -> [路由配置] 页面，可以看到所有已配置的路由信息。

查询：支持通过路由名称、状态等条件进行筛选。

列表信息：展示路由名称、渠道、源路径、目标地址、请求方法、状态及创建时间。

快捷操作：

编辑：修改现有路由配置。

复制：快速克隆一个现有的路由配置（适用于相似接口的快速创建）。

历史：查看该路由的配置变更记录及版本对比。

删除：删除不再使用的路由。

The screenshot shows a web-based route configuration interface. At the top, there's a header with the title '路由配置' (Route Configuration) and a navigation bar with links like '首页' (Home), '帮助文档' (Help Document), and a user icon. Below the header is a search/filter bar with fields for '路由名称' (Route Name), '状态' (Status), and buttons for '查询' (Search) and '重置' (Reset). The main area is a table listing routes. The columns are: 路由名称 (Route Name), 渠道 (Channel), 源路径 (Source Path), 目标地址 (Target Address), 方法 (Method), 状态 (Status), 创建时间 (Creation Time), and 操作 (Operations). There are two rows of data: one for '测试' (Test) and one for '仓库过户入库通知系统' (Warehouse Transfer In-Notice System). Each row includes a timestamp watermark 'lcb 2026-01-23 11:31'. The operations column for each row contains buttons for '编辑' (Edit), '复制' (Copy), '历史' (History), and '删除' (Delete). At the bottom right of the table area, there are pagination controls showing '共2条' (2 pages), a page number '1', and a dropdown for '10条/页' (10 items per page).

2.2 新建/编辑路由

点击“新建路由”按钮或列表中的“编辑”按钮，侧边栏将打开路由配置表单。

(1) 基本信息

配置路由的基础属性：

路由名称：接口的业务名称。

渠道：接口所属的渠道标识（如：淘宝、京东、ERP等）。

HTTP 方法：支持 GET, POST, PUT, DELETE。

源路径 (连接器)：连接器监听的相对路径（例如`/api/v1/orders`）。

目标地址 (云仓)：数据最终转发到的完整 URL（例如`https://cloud-wms.com/api/orders`）。

状态：启用或停用该路由。

新建路由

基本信息 字段映射 安全配置 请求头 聚合配置

* 路由名称

渠道

请输入路由名称

例如：淘宝、京东

* HTTP 方法

状态

POST

启用

* 源路径 (连接器)

/api/v1/orders

连接器监听的请求路径

* 目标地址 (云仓)

https://cloud-wms.com/api/orders

(2) 字段映射 (核心功能)

配置源数据到目标数据的转换逻辑。

手动映射：

源字段：输入源 JSON 中的字段路径（如 `order.id` 或 `items[*].price`）。

目标字段：输入目标 JSON 中的字段名。

类型：指定目标字段的数据类型（字符串、整数、小数、布尔、数组、对象）。

默认值：当源字段为空时使用的默认值。

转换规则：点击 “**转换规则**” 按钮，可为特定字段添加处理逻辑。支持链式处理（上一步的输出作为下一步的输入）：

字符串操作：转大写、转小写、去空格、截取 (Substring)、拼接 (Concat)、替换 (Replace)。

数组/其它操作：分割 (Split)、连接 (Join) 等。

新建路由

基本信息 字段映射 安全配置 请求头 聚合配置

+ 添加映射

AI 智能映射

清空映射

hello



world

字符串



默认值

转换规则 (0)



AI 智能映射：

点击“AI 智能映射”按钮，打开智能助手。

1. 在左侧粘贴 源数据 JSON 示例。
2. 在右侧粘贴 目标数据 JSON 示例。
3. 点击“开始生成”，AI 将自动分析语义并生成映射规则。
4. 确认无误后点击“应用映射”自动填充到配置中。

i 使用说明

请在左侧粘贴【源数据 JSON】，在右侧粘贴【目标数据 JSON】。AI 将自动分析字段含义并生成映射规则。

源数据示例 (Source JSON)

```
        "specificationValue": "T0000001",  
        "calculateType": 11,  
        "quantity": 20,  
        "weight": 300.80,  
        "packageNo": "PKG20260122003",  
        "qualityDesc": "合格",  
        "theoreticalWeight": 300.20,  
        "remark": "明细3-品材规产不一致"  
    }  
]  
}
```

美化

目标数据示例 (Target JSON)

```
        "calculateType": 11,  
        "quantity": 20,  
        "weight": 300.80,  
        "packageNo": "PKG20260122003",  
        "qualityDesc": "合格",  
        "theoreticalWeight": 300.20,  
        "remark": "明细3-品材规产不一致"  
    }  
]  
}
```

美化

选择模型:

Doubao Seed 1.6 Flash

) 开始生成映射

AI 正在思考...

AI 思考过程:

我现在需要分析源数据和目标数据的字段映射规则。首先，我要仔细对比这两个JSON的结构和内容，看看每个字段是如何对应起来的。

首先看顶层字段，源数据和目标数据的顶层结构几乎完全一样。源数据的字段有：outsideBillNo、warehouseCode、oldOwner、newOwner、plateNumber、receiptTime、receiptByName、receiptPhone、remark、receiptType、detail。目标数据的字段也完全相同，所以这些顶层字段应该是——对应的，每个字段名和含义都一样，所以它们的映射应该是直接对应。

取消

正在生成...

选择模型:

Doubaor Seed 1.6 Flash

开始生成映射

plateNumber、receiptTime、receiptByName、receiptPhone、remark、receiptType、detail。目标数据的字段也完全相同，所以这些顶层字段应该是一对一对应的，每个字段名和含义都一样，所以它们的映射应该是直接对应。

接下来查看detail数组部分。源数据的detail是一个包含三个对象的数组，目标数据的detail也是同样的结构和三个对象。每个对象里的字段也完全一致，比如outsideDetailId、skuCode、categoryName等等。所以明细数组中的每个字段都应该是——对应的，不需要调整，只是需要在映射规则里加上数组占位符“[]”。

现在主要确认每个字段的类型。比如outsideBillNo是字符串类型、warehouseCode也是字符串、oldOwner、newOwner都是字符串、plateNumber字符串。

已生成 24 条映射规则 (点击“应用映射”按钮保存)

源字段	目标字段	类型	默认值
outsideBillNo	outsideBillNo	string	
warehouseCode	warehouseCode	string	
oldOwner	oldOwner	string	
newOwner	newOwner	string	
plateNumber	plateNumber	string	

取消

应用映射

(3) 安全配置

用于对转发的数据进行加密处理。

加密方式：目前支持 RSA 非对称加密。

公钥：输入目标系统的 RSA 公钥。

加密后字段名：指定加密后的数据存放在哪个字段（默认为 `data`）。

新建路由

基本信息 字段映射 安全配置 请求头 聚合配置

加密方式

RSA 非对称加密

公钥 (Public Key)

-----BEGIN PUBLIC KEY-----...

加密后字段名

默认为 data

(4) 请求头配置

可以添加自定义的 HTTP Header，随请求一起发送给目标系统。

基本信息 字段映射 安全配置 请求头 聚合配置

+ 添加 Header

Key

Value

描述



(5) 聚合配置

适用于列表数据的聚合处理场景。

启用聚合：勾选后开启。

数组字段：指定需要聚合的明细数组字段名。

分组字段：按哪些字段进行分组。

求和字段：需要进行数值累加的字段。

计数保存字段：用于存储记录条数的字段名。

启用聚合

数组字段

detail

分组字段

categoryName ✕ materialName ✕ factoryName ✕ specificationName ✕

求和字段

quantity ✕ weight ✕

计数保存字段

例如: count

3. 路由测试

在配置完成后，可以通过 [路由测试] 页面直接验证接口的连通性和转换逻辑。

1. **选择路由**：在下拉框中选择一个已配置的路由，系统会自动填充 Method 和 URL。

2. **生成模拟数据 (AI)**：

如果不想手写复杂的请求体，可以点击输入框上方的 AI 机器人图标。

* 系统会根据路由的映射配置和字段定义，智能生成符合规范的 Mock JSON 数据。

3. **发送请求**：点击 “发送” 按钮。

4. **查看结果**：右侧将展示响应状态码、耗时以及响应体数据。

首页 路由配置 请求日志 路由测试

提示：选择已有路由可自动填充请求参数

选择路由 (可选)

【POST】仓库过户入库通知系统 (/externalApi/warehouseTransferInNotify)

请求地址 (URL) POST /connector/proxy/externalApi/warehouseTransferInNotify 操作 发送请求

Query Params Headers Body

Doubao Seed 1.6 Flash AI 自动生成测试数据 JSON 美化

```
{  
  "mock": true,  
  "timestamp": 1769139411555  
}
```

响应结果

现在，我需要根据这些配置，生成一个符合要求的JSON请求体。首先，确定需要包含哪些字段。根据mappings里的target字段，需要生成这些target对应的字段。比如：
> receiptTime: source是datetime, target是string，所以要生成一个ISO8601格式的时间字符串，比如“2023-10-01T12:00:00+08:00”这样的格式。
>
> receiptType3: 因为source是receiptType，类型是integer，target是receiptType3，所以receiptType3是integer，比如1或2等。
>
> oldOwner: 默认值是“原货主不能为空”，但有transformations替换“不能”为“可以”，所以处理后应该是“原货主可以空”？或者可能defaultValue是当source不存在时的默认值？这里可能需要先取source的值，如果不存在则用defaultValue，然后应用transformations。不过可能这里的配置是说如果source字段存在，就处理它，否则用defaultValue并处理。但用户可能希望生成一个有效的值，所以可能oldOwner的source是某个字段，但这里的target是oldOwner，可能直接设置为一个值，比如“原货主”，然后替换为“原货主可以空”？或者可能用户希望我们直接生成target字段的值，不管source？比如oldOwner的target是oldOwner，所以直接写oldOwner的值，而transformations是replace，所以比如原value是“不能”，替换成“可以”。但可能这里的default

选择路由 (可选)

【POST】仓库过户入库通知系统 (/externalApi/warehouseTransferInNotify)

请求地址 (URL)

POST /connector/proxy/externalApi/warehouseTransferInNotify

操作

发送请求

Query Params Headers Body

Douba Seed 1.6 Flash AI 自动生成测试数据 JSON 美化

```
"categoryName": "化工产品",
"packageNo": "PKG-003",
"materialName": "塑料粒子",
"outsideDetailId": 10003,
"detailSpecificationName": "PP-001",
"skuCode": "PLS-P001",
"specificationName": "聚丙烯"
},
{
"plateNumber": "沪B67890",
"warehouseCode": "WH-TR-001"
}
```

响应结果

Status: 200 Time: 350ms

```
{
  code: 0,
  data: null,
  message: "成功"
}
```

copy

4. 请求日志

[请求日志] 页面记录了所有经过连接器的请求流水，是排查问题的利器。

日志列表：展示请求时间、路由名称、状态码、耗时等概览信息。

日志详情：点击“查看”可打开详情弹窗，包含三个重要视图：

原始入参：连接器收到的原始数据。

转换后入参：经过映射和转换规则处理后，发给目标系统的数据。

响应数据：目标系统返回的数据。

支持一键复制 JSON 内容，便于调试。

路由名称:	路由名称	渠道:	渠道名称	请求URL:	请求URL	状态:	全部	查询	重置
lcb 2026-01-23 11:38									
lcb 2026-01-23 11:38									
请求时间	路由名称	渠道	请求路径	状态码	耗时	错误信息	操作		
2026-01-23 11:37:55	仓库过户入库通知系统	60	/externalApi/warehouseTransferInN...	200	314 ms	-		详情	
2026-01-23 11:19:45	测试	60	/externalApi/warehouseTransferInN...	400	54 ms	-		详情	
2026-01-23 10:31:19	仓库过户入库通知系统	60	/externalApi/warehouseTransferInN... /externalApi/warehouseTransferInNotify2	200	1238 ms	-		详情	
2026-01-23 10:10:50	仓库过户入库通知系统	60	/externalApi/warehouseTransferInN...	500	4146 ms	Connect to http://localhost:8084 [localhost/127.0.0.1, localhost/0:0:0:0:0:1] fa...		详情	
2026-01-23 09:36:51	仓库过户入库通知系统	60	/externalApi/warehouseTransferInN...	500	4463 ms	Connect to http://localhost:8084 [localhost/127.0.0.1, localhost/0:0:0:0:0:1] fa...		详情	
2026-01-22 20:26:25	仓库过户入库通知系统	60	/externalApi/warehouseTransferInN...	200	487 ms	计量方式不能为空: 计量方式不能为空:		详情	
2026-01-22 20:06:31	仓库过户入库通知系统	60	/externalApi/warehouseTransferInN...	200	38 ms	-		详情	
2026-01-22 19:39:53	仓库过户入库通知系统	60	/externalApi/warehouseTransferInN...	200	112 ms	-		详情	
2026-01-22 19:39:39	仓库过户入库通知系统	60	/externalApi/warehouseTransferInN...	200	262 ms	-		详情	
2026-01-22 19:39:02	仓库过户入库通知系统	60	/externalApi/warehouseTransferInN...	200	496 ms	-		详情	
2026-01-22 19:34:25	仓库过户入库通知系统	60	/externalApi/warehouseTransferInN...	200	661 ms	-		详情	

共11条 < 1 > 20 条/页

请求日志详情

X

请求时间	2026-01-22 20:26:25
路由名称	仓库过户入库通知系统
渠道	60
请求路径	/externalApi/warehouseTransferInNotify
状态码	200
耗时	487 ms
错误信息	计量方式不能为空; 计量方式不能为空;

原始入参 转换后入参 响应数据

复制JSON

```
{  
  "outsideBillNo": "RK20260122001",  
  "warehouseCode": "WH001",  
  "oldOwner": "OWNER_OLD001",  
  "newOwner": "OWNER_NEW001",  
  "plateNumber": "京A12345",  
  "receiptTime": "2026-01-22 10:30:00",  
  "receiptByName": "张三",  
  "receiptPhone": "13800138000",  
  "remark": "过户入库测试",  
  "detail": [  
    {  
      "outsideDetailId": 10001,  
      "skuCode": "SKU001",  
      "categoryName": "高粱谷物酒"  
    }  
  ]  
}
```

5. 版本历史

在路由列表点击“历史”，可以查看该路由的所有修改记录。

系统会高亮显示旧版本与新版本的配置差异（Diff视图），方便回溯配置变更。

配置变更历史

X

2026-01-23 12:58:56

修改人: lcb

2026-01-23 12:58:46

修改人: lcb

配置对比

行内对比 左右对比 回滚至此版本

```
@@ -1,9 +1,9 @@
1  {
2 - "name": "仓库过户入库通知系统",
3   "method": "POST",
4   "status": "active",
5 - "channel": "70",
6   "targetUrl": "http://localhost:8084/warehouse.wms.
7   "sourcePath": "/externalApi/warehouseTransferInNot
8   "mappingConfig": {
9     "headers": [
```

```
1  {
2 + "name": "仓库过户入库通知系统3",
3   "method": "POST",
4   "status": "active",
5 + "channel": "60",
6   "targetUrl": "http://localhost:8084/warehouse.wms.
7   "sourcePath": "/externalApi/warehouseTransferInNot
8   "mappingConfig": {
9     "headers": [
```