

庖丁解牛：华为云微服务工具解放开发者

部门：应用平台服务
作者：王启军
日期：20190920



目录

1. 微服务拆分工具
2. 契约管理工具

微服务架构的优势

因素	单体架构	微服务架构
交付速度	较慢	较快
故障隔离范围	线程级	进程级
整体可用性	较低	更高
架构持续演进	困难	简单
沟通效率	低	高
技术栈选择	受限	灵活
可扩展性	受限	灵活
可重用性	低	高
实现业务复杂性分解难度	困难	容易
产品创新复杂度	困难	容易
一致性实现成本	低	高
时延	低	高
资源成本	低	高
关联查询复杂度	简单	复杂
远程调用	不涉及	涉及
服务治理	不涉及	涉及
对开发人员的要求	低	高
对工具的依赖	较低	较高
运维复杂度	低	高

微服务架构的第一大问题——如何拆分微服务？

- 影响因素

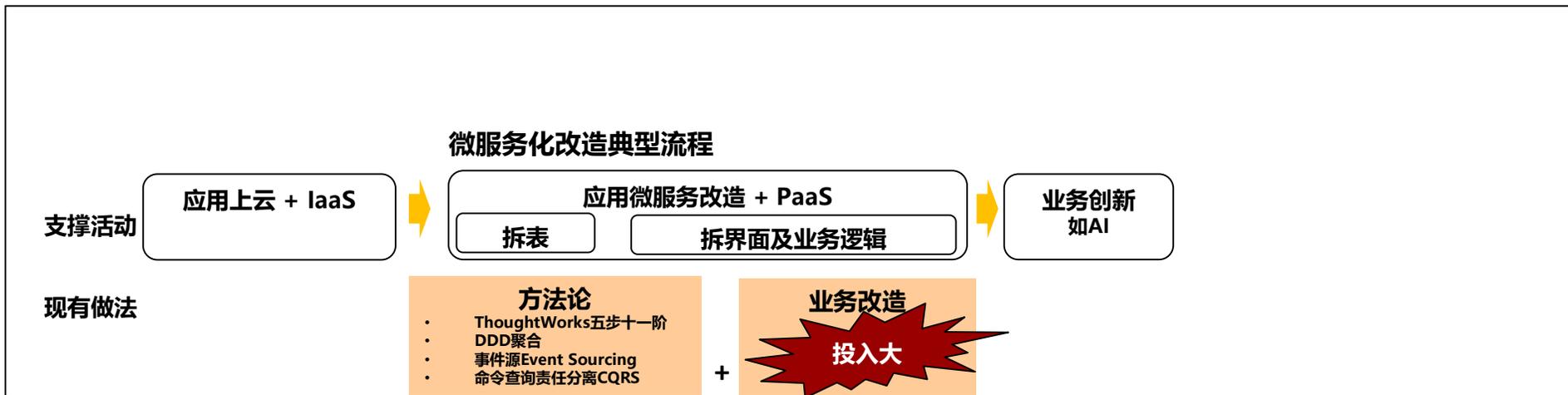
- › 团队大小
- › 交付周期
- › 业务方向
- › 故障范围
- › 现有数据规模
- › 吞吐量
- › 一致性
- ›

拆分不好，可能带来的问题

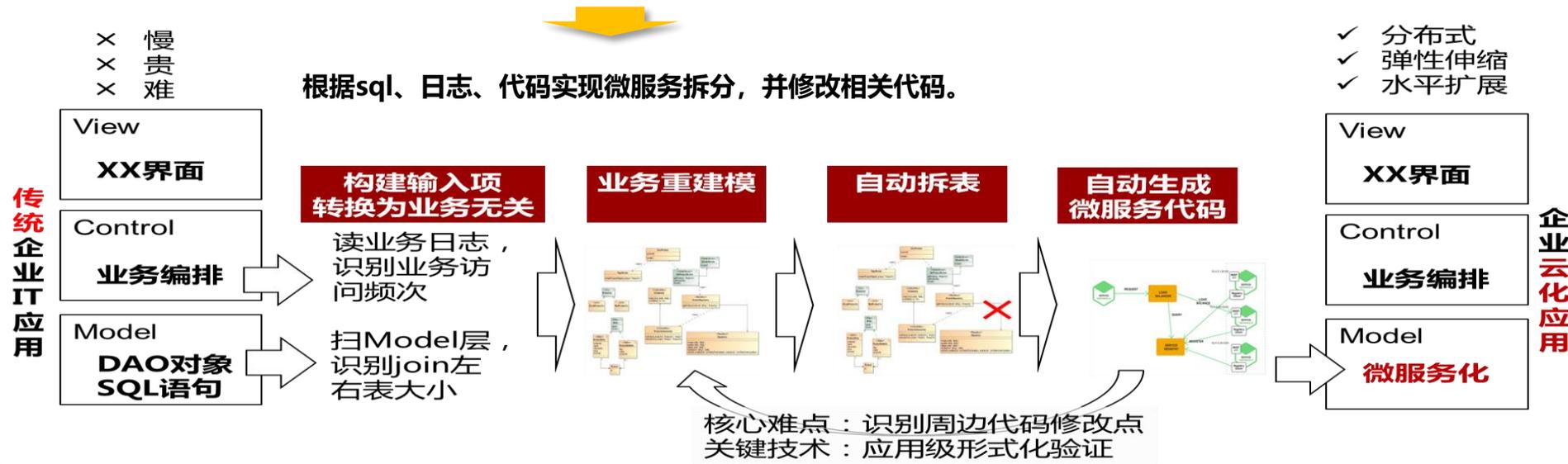
- 拆分不好，可能带来的问题

- › 服务数量爆炸，运维复杂
- › 服务数量太少，不够灵活
- › 一个user story影响众多服务
- › 接口频繁变更
- › 大量关联查询
- › 系统架构复杂度提升

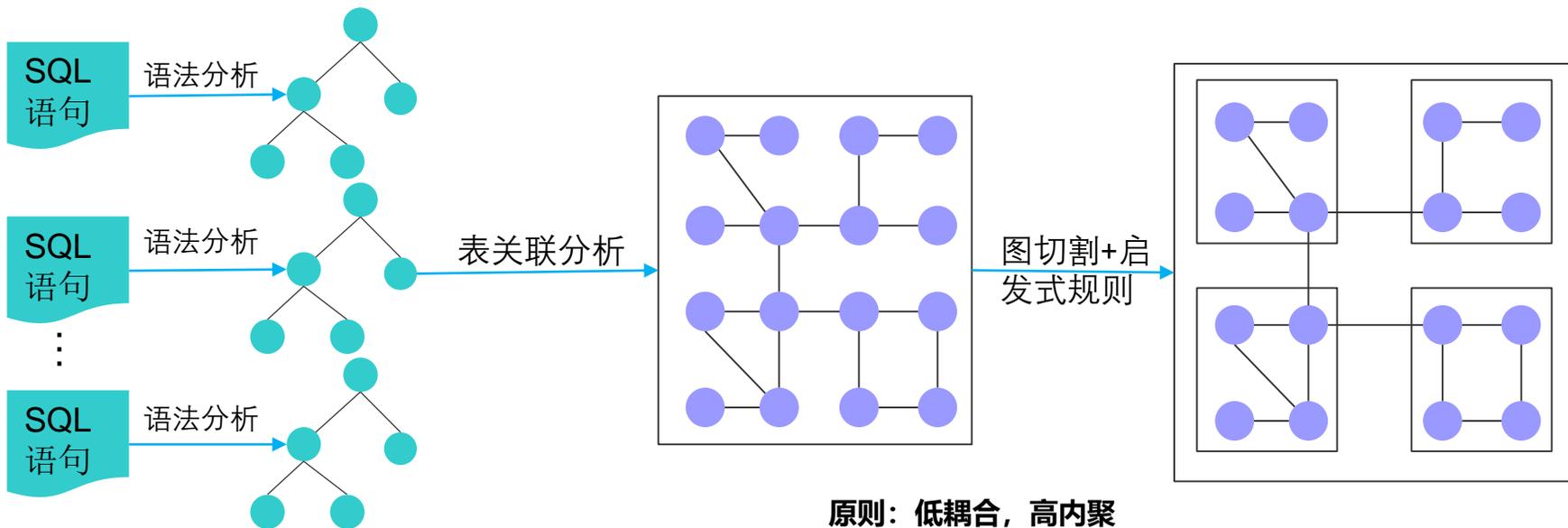
通过微服务拆分工具解放开发者



挑战：强业务相关，必须业务+微服务专家介入，投入大、时间长；
关键技术：应用微服务化改造工具套件



微服务拆分工具-表自动分库



Step1: 提取系统所有SQL

Step2: 逐条根据SQL创建语法树

Step3: 在每棵语法树上分析表的关联关系并汇集
成表关联关系加权图

Step6: 采用图切割算法并配合启发式规则将表分
割到不同库中

原则：低耦合，高内聚

耦合度：微服务划分后，微服务与微服务连接的边的权重之和

内聚度：微服务划分后，微服务内部表与表之间的边权重之和

采用图搜索算法划分微服务的启发式规则(规则可自定义):

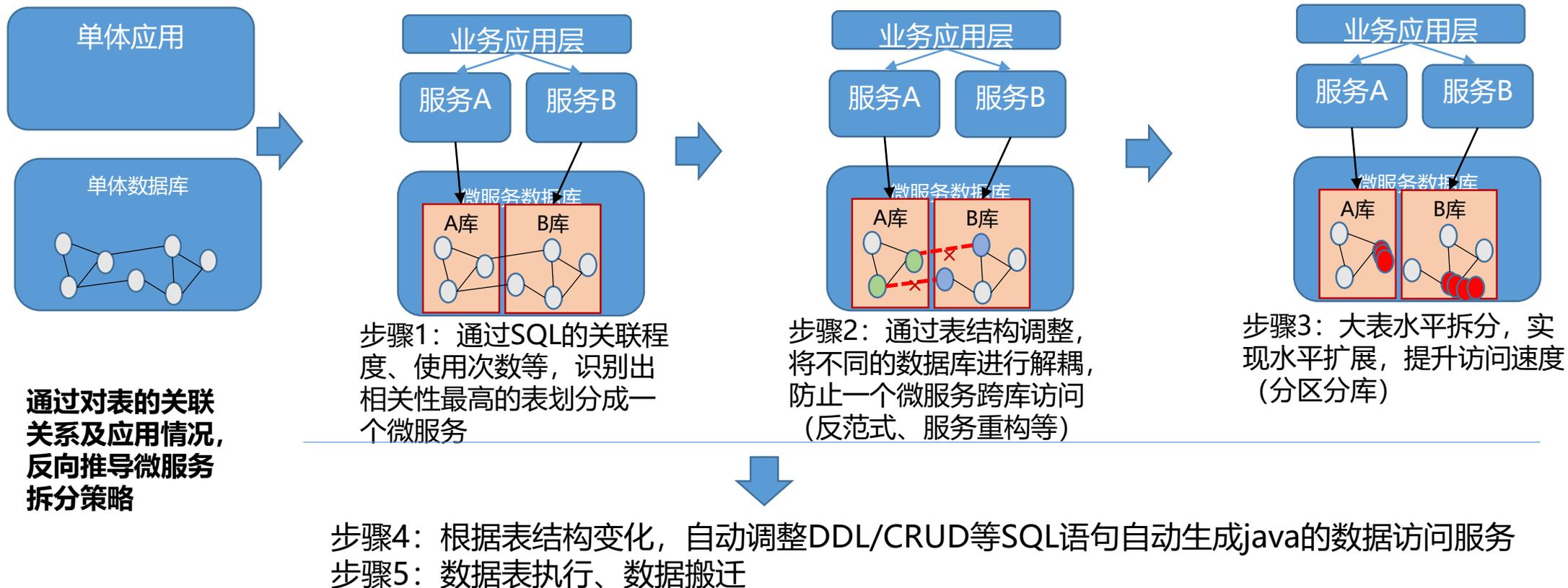
规则1：贪婪原则

规则2：完全独立的表可归属任意微服务，也可独立成微服务

规则3：归属同一微服务的表数量不低于5（作为参数可配置）

规则4：关联度大于10（作为参数可配置）的表归属同一微服务

微服务拆分工具



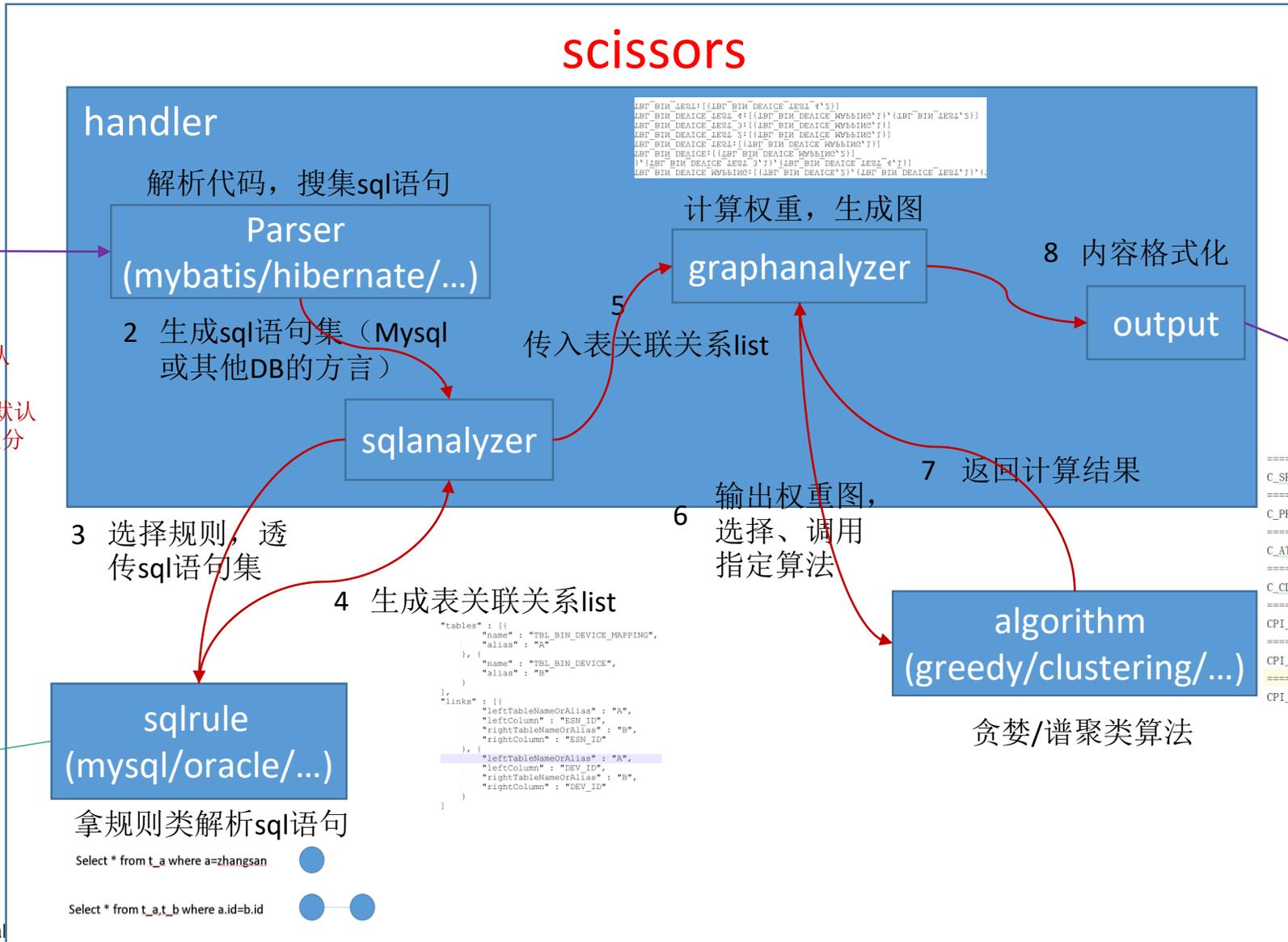
拆分原理



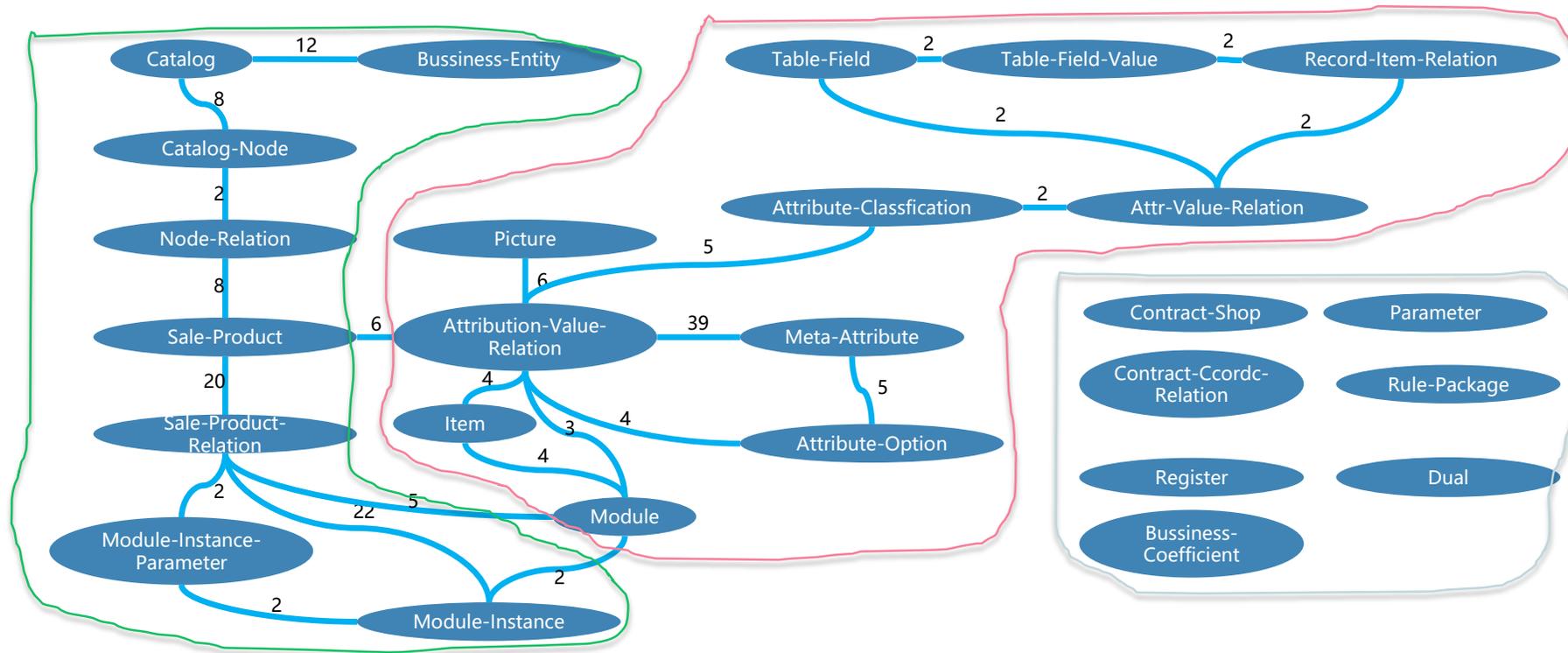
1. 指定代码路径
/home/xxx/project
2. 指定算法 (目前默认 greedy)
3. 指定sql规则 (目前默认 mysql, 后续做成智能分析)

开发时预处理: 通过 Antlr4 工具生成 java 可识别的解析规则的对象, 埋入源码

mysql/oracle 的规则文件

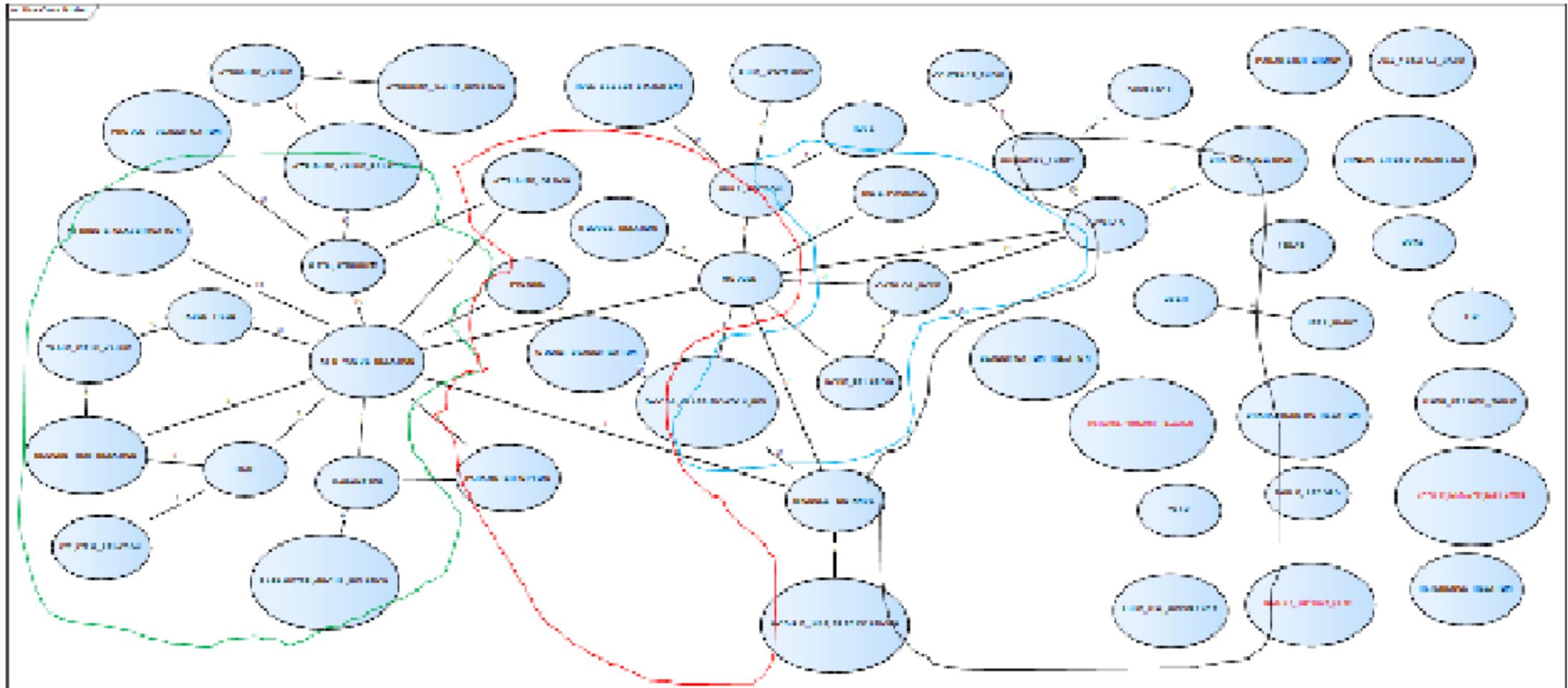


案例1：存量系统1微服务拆分分析-图搜索拆分微服务



从左到右分别是拆分出来的商品库、模板库，最后一些离散表是系统为了关联外部系统而引入的表，独立成库

案例2：存量系统2微服务拆分分析-图搜索拆分微服务



从左到右分别是拆分出扩展属性服务、模型/模型实例服务、目录服务以及离散表独立成的服务，其中离散表中标红部分概念上应该划分到模型/模型实例服务中，但实际的sql，却并未体现其与模型/模型实例的关联关系，因此被自动划分到独立服务中

微服务架构下，API管控的挑战

- API在微服务架构下爆炸性增长
- 移动互联网、物联网
- API是技术leader和架构师的控制力，API==契约
- API First，解耦各个服务的开发者

ServiceStage 契约管理工具—管控分离

* 应用设计名称

* 版本

描述

* 契约地址 

授权信息

用户名/组织 仓库名称

* 契约文件

契约文件	契约地址
petstore	petstore.yaml

新增契约

```
1
2 swagger: "2.0"
3 info:
4   description: "This is a sample server Petstore server. You can find out
5   version: "1.0.0"
6   title: "Swagger Petstore"
7   termsOfService: "http://swagger.io/terms/"
8   contact:
9     email: "apiteam@swagger.io"
10  license:
11    name: "Apache 2.0"
12    url: "http://www.apache.org/licenses/LICENSE-2.0.html"
13  host: "petstore.swagger.io"
14  basePath: "/v2"
15  tags:
16  - name: "pet"
17    description: "Everything about your Pets"
18    externalDocs:
19      description: "Find out more"
20      url: "http://swagger.io"
21  - name: "store"
22    description: "Access to Petstore orders"
23  - name: "user"
24    description: "Operations about user"
25    externalDocs:
26      description: "Find out more about our store"
27      url: "http://swagger.io"
28  schemes:
29  - "https"
30  - "http"
31  paths:
32  /pet:
33    post:
34      tags:
35      - "pet"
36      summary: "Add a new pet to the store"
37      description: ""
```

* 契约文件

* 契约地址

Swagger Petstore 1.0.0

[Base URL: petstore.swagger.io/v2]

This is a sample server Petstore server. You can find out more about Swagger at <http://swagger.io> or on irc.freenode.net, #swagger. For this sample, you can use the api key `special-key` to test the authorization filters.

[Terms of service](#)

[Contact the developer](#)

[Apache 2.0](#)

[Find out more about Swagger](#)

pet Everything about your Pets Find out more: <http://swagger.io>

POST /pet Add a new pet to the store

PUT /pet Update an existing pet

GET /pet/findByStatus Finds Pets by status

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home and
organization for a fully connected,
intelligent world.

**Copyright©2018 Huawei Technologies Co., Ltd.
All Rights Reserved.**

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.

