**文档编号：JDYP\_SD002**

**版本号：V1.0**

**禁毒情报研判系统**

**详细设计说明书**

**四川科瑞软件有限责任公司**

|  |  |
| --- | --- |
| 编制：唐鑫 | 生效日期： |
| 审核：张亮 | 批准： |

---------------------------------------------------------------------

四川科瑞软件有限责任公司对本文件资料享受著作权及其它专属权利，未经书面许可，不得将该等文件资料（其全部或任何部分）披露予任何第三方，或进行修改后使用。

文件更改摘要：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 日期 | 版本号 | 修订说明 | 修订人 | 审核人 | 批准人 |
| 2020.7.1 | V0.1 | 创建 | 唐鑫 | 严义兵 | 张亮 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

目录

[1 引言 7](#_Toc44939645)

[1.1 编写目的 7](#_Toc44939646)

[1.2 背景 7](#_Toc44939647)

[1.3 定义 7](#_Toc44939648)

[1.4 参考资料 7](#_Toc44939649)

[2 总体设计 7](#_Toc44939650)

[2.1 运行环境 7](#_Toc44939651)

[2.2 软件功能描述 8](#_Toc44939652)

[2.3 软件结构 10](#_Toc44939653)

[3 模块设计 11](#_Toc44939654)

[3.1 界面设计 11](#_Toc44939655)

[3.2 涉毒档案 11](#_Toc44939656)

[3.2.1 添加档案 11](#_Toc44939657)

[3.2.2 导入档案 13](#_Toc44939658)

[3.2.3 信息补全 16](#_Toc44939659)

[3.2.4 全息档案 19](#_Toc44939660)

[3.2.5 关系图谱 22](#_Toc44939661)

[3.2.6 活动轨迹 28](#_Toc44939662)

[3.2.7 关系扩展 29](#_Toc44939663)

[3.2.8 编辑标签 31](#_Toc44939664)

[3.3 研判应用 32](#_Toc44939665)

[3.3.1 关系路径分析 32](#_Toc44939666)

[3.3.2 特征交易资金分析 36](#_Toc44939667)

[3.3.3 团伙间关系分析 41](#_Toc44939668)

[3.4 毒情作战室 44](#_Toc44939669)

[3.4.1 多维毒情分析 44](#_Toc44939670)

[3.4.2 数据资源统计 46](#_Toc44939671)

[3.4.3 业务数据统计 46](#_Toc44939672)

[3.5 系统管理 47](#_Toc44939673)

[3.5.1 积分维度配置 47](#_Toc44939674)

[3.5.2 账号管理 50](#_Toc44939675)

[3.5.3 角色管理 52](#_Toc44939676)

[3.5.4 权限管理 53](#_Toc44939677)

[3.5.5 数据字典管理 56](#_Toc44939678)

[3.5.6 消息中心 57](#_Toc44939679)

[3.5.7 关注对象 60](#_Toc44939680)

[3.5.8 积分配置 61](#_Toc44939681)

[3.5.9 高危地区配置 63](#_Toc44939682)

[3.5.10 组织架构 65](#_Toc44939683)

[3.5.11 人员考核 68](#_Toc44939684)

[3.6 实战应用 70](#_Toc44939685)

[3.6.1 话单导入 70](#_Toc44939686)

[3.6.2 话单导出 73](#_Toc44939687)

[3.6.3 话单分析 77](#_Toc44939688)

[3.6.4 话单积分 79](#_Toc44939689)

[3.6.5 话单团伙重要对象分析 83](#_Toc44939690)

[3.6.6 共同陌生电话分析 87](#_Toc44939691)

[3.6.7 共同通话人分析 89](#_Toc44939692)

[3.6.8 互通电话分析 91](#_Toc44939693)

[3.6.9 共同位置分析 94](#_Toc44939694)

[3.6.10 基站积分 96](#_Toc44939695)

[3.6.11 基站下通话详情 99](#_Toc44939696)

[3.6.12 基站数据导出 102](#_Toc44939697)

[3.6.13 基站查询 103](#_Toc44939698)

[3.6.14 导入账单 106](#_Toc44939699)

[3.6.15 导出账单 109](#_Toc44939700)

[3.6.16 账单分析 112](#_Toc44939701)

[3.6.17 账单积分 114](#_Toc44939702)

[3.6.18 资金汇总分析 118](#_Toc44939703)

[3.6.19 资金流向分析 122](#_Toc44939704)

[3.6.20 共同交易账号分析 124](#_Toc44939705)

[3.6.21 互相交易分析 127](#_Toc44939706)

[3.6.22 通讯录导出 129](#_Toc44939707)

[3.6.23 bcp文件导入 131](#_Toc44939708)

[3.6.24 通讯录分析 134](#_Toc44939709)

[3.6.25 共同联系人分析 137](#_Toc44939710)

[3.6.26 好友归属地分析 140](#_Toc44939711)

[3.6.27 通讯录团伙分析 144](#_Toc44939712)

[3.6.28 新建任务 146](#_Toc44939713)

[3.6.29 任务筛选 148](#_Toc44939714)

[3.6.30 任务详情 150](#_Toc44939715)

[3.6.31 任务对象管理 154](#_Toc44939716)

[3.6.32 任务话单管理 158](#_Toc44939717)

[3.6.33 任务账单管理 161](#_Toc44939718)

[3.6.34 任务通讯录管理 164](#_Toc44939719)

[3.7 自动研判 167](#_Toc44939720)

[3.7.1 隐毒对象预警 167](#_Toc44939721)

[3.8 涉毒云搜 171](#_Toc44939722)

[3.8.1 搜索账号 171](#_Toc44939723)

[3.8.2 搜索电话 173](#_Toc44939724)

[3.8.3 搜索人员 175](#_Toc44939725)

[3.9 证书登录 179](#_Toc44939726)

[3.9.1 PKI登录 179](#_Toc44939727)

[3.10 管控预警 182](#_Toc44939728)

[3.10.1 涉毒人员活跃度预警 182](#_Toc44939729)

[4 接口设计 186](#_Toc44939730)

[4.1 外部接口设计 186](#_Toc44939731)

[4.2 内部接口设计 186](#_Toc44939732)

[5 性能设计 186](#_Toc44939733)

[5.1 性能瓶颈分析 186](#_Toc44939734)

[5.2 性能设计方案 187](#_Toc44939735)

[5.2.1 缓存 187](#_Toc44939736)

[5.2.2 预处理 187](#_Toc44939737)

# 引言

## 编写目的

根据相关需求文档以及项目章程，以概要设计说明书作为指导编写软件详细说明书，为相关人员具体编码提供参考，提升开发效率。

## 背景

1. 软件系统名称暂定为【信息分析系统】。
2. 本文档主题包括系统的总体设计、详细的模块设计以及接口设计，本文档定义了每个功能模块的输入输出以及处理过程等，为编码提供了指引。
3. 本文档的读者对象包含软件开发人员以及测试人员，以上人员可以在项目实施过程中以本文档作参考作用。

## 定义

无。

## 参考资料

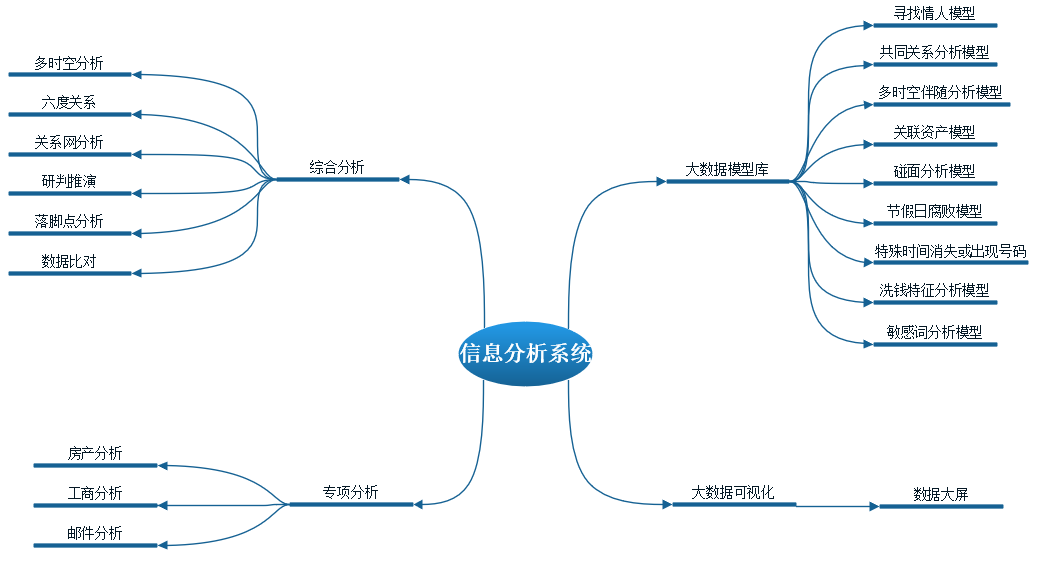
《IAS\_RD003\_软件需求规格说明书》

# 总体设计

## 软件功能描述

|  |  |
| --- | --- |
| 功能模块 | 需求名称 |
|  |
| 综合分析 | 多时空分析 |  |
| 六度关系 |  |
| 关系网分析 |  |
| 研判推演 |  |
| 落脚点分析 |  |
| 数据比对 |  |
| 大数据模型库 | 寻找情人模型 |  |
| 共同关系分析模型 |  |
| 多时空伴随分析模型 |  |
| 数据共振模型 |  |
| 关联资产模型 |  |
| 碰面分析模型 |  |
| 节假日腐败模型 |  |
| 特殊时段消失或出现号码 |  |
| 洗钱特征分析模型 |  |
| 敏感词分析模型 |  |
| 数据大屏 | 数据大屏 |  |
| 专项分析 | 房产分析 |  |
| 工商分析 |  |
| 邮件分析 |  |

## 软件结构

以图表形式给出软件子系统结构和模块组成结构及数据关联。

# 模块设计

## 界面设计

参考原型

## 综合分析

### 多时空分析

#### 时空轴

##### 功能描述

对人员事件，按照时间顺序对事件依次展示。

##### 设计描述

##### 输入

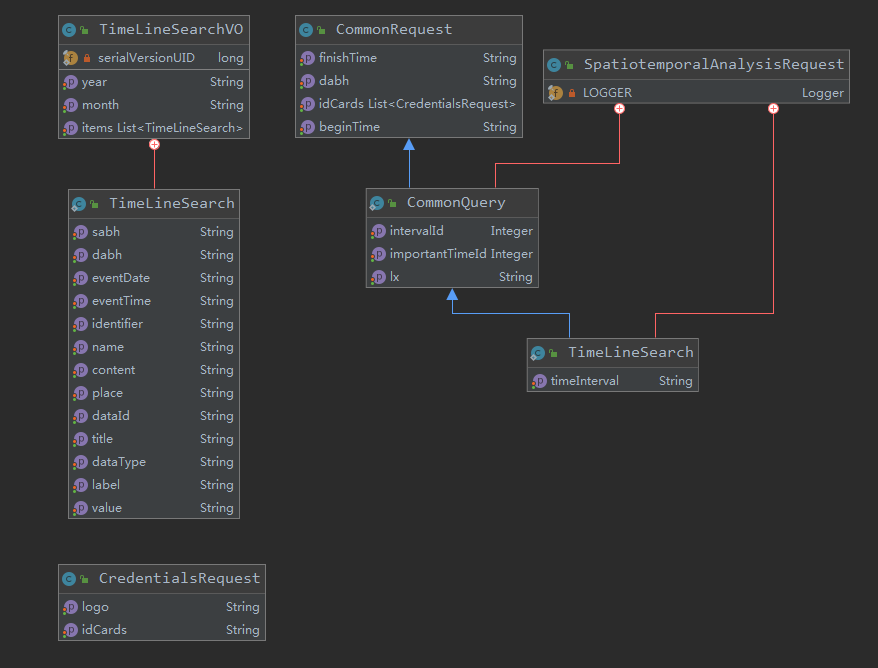
档案编号、起止时间、人员证件号、数据维度、重要时间。

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **字段名称** | **数据类型** | **是否必传** |
| id | id | int(32) | 是 |
| age | 年龄 | int(8) | 否 |

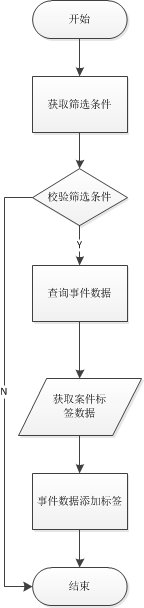
##### 输出

事件记录。

##### 类图



##### 流程图



##### 处理过程

1. 校验处理参数；
2. 获取事件数据；
3. 获取当前案件标签数据；
4. 事件数据添加标签、添加年月信息；
5. 返回给终端。

##### 算法

##### 质量要求

案件编号不能为空，数据维度类型与提供数据相对应。

#### 地图轨迹

##### 描述

对人员事件，根据事件位置信息，在地图上进行展示。

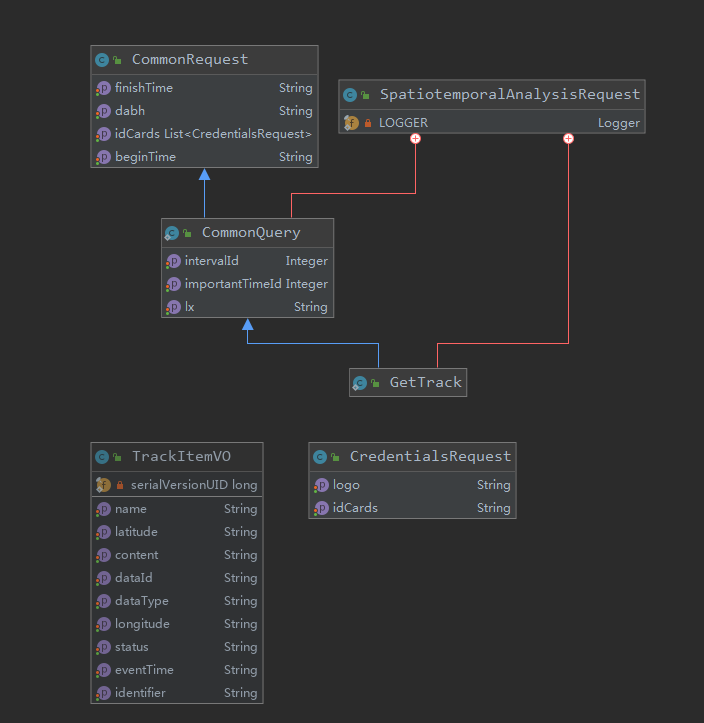
##### 输入

档案编号、起止时间、人员证件号、数据维度、轨迹类型。

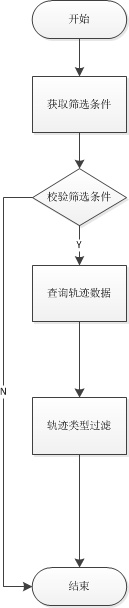
##### 输出

包含位置的事件信息。

##### 类图



##### 流程图



##### 处理过程

1. 校验处理参数；
2. 获取轨迹数据；
3. 根据轨迹类型进行过滤；
4. 返回给终端。

##### 质量要求

档案编号不能为空，轨迹数据包含经纬度。

#### 特殊时刻分析

##### 描述

##### 输入

##### 输出

##### 类图

##### 流程图

##### 处理过程

##### 质量要求

### 六度关系

#### 描述

选择两个涉案人员，设置开始结束日期，各统计维度筛选条件，选择关系层级，点击开始分析，即可显示两个涉案人员之间的关系。

点击关系列表，展示具体的路径层级、路径详情。

点击“关系分析图”中的边或者“路径详情”中的维度显示具体关系明细。

#### 输入

{

"airplaneCount": 22,

"busCount": 22,

"callAmount": 22,

"callCount": 22,

"dabh": "string",

"dimensionality": "01,02",

"endTime": "string",

"idCards": [

{

"idCard": "5112232323244343z",

"logo": "00"

}

],

"keyword": "string",

"liveCount": 22,

"relationLevel": 2,

"startTime": "string",

"trainCount": 22,

"transferAmount": 22.22,

"transferCount": 22

}

#### 输出

两个涉案人员的关系图

#### 类图

#### 流程图



#### 处理过程

1. 对传参校验
2. 维度类型解析
3. 根据传参kylin查询关系统计结果
4. 根据传参对关系统计结果过滤
5. 广度遍历求路径信息
6. 路径信息结果集封装

#### 质量要求

两个涉案人为必选项，能够根据档案编码，及多个筛选项，准确展示两个涉案人之间层级关系。

### 关系网分析

#### 关系网分析

##### 描述

根据案件内对象资源信息，在画布工具上展示案件内所有对象之间的关系（直接关系或间接关系），并且用户可以查看所有关系的详情。

##### 输入

{

dabh string 档案编号

dimensionality string 维度：01 同户 02 同事 03 银行卡转账 04 同住 05 同行-飞机 06 同行-大巴 07 同行-火车 08 投资 09 通话，多个维度之间以逗号分隔

endTime string 结束时间 yyyyMMdd

idCards [

证件号集合

Credentials{

idCard string 证件号

logo string 证件标识 00 个人 01 企业

}

]

intimacyRange string 亲密度范围,例如：20-40

keyword string 关键字查询

startTime string 开始时间 yyyyMMdd

}

##### 输出

{

nodeInfos [

{

age integer($int32) 年龄

idCard string 身份证号码或者企业社会代码

labels [...标签列表...]

nodeId string 节点对象id（身份证号码或者企业社会代码或电话、银行卡）

nodeType string 节点类型 00 个人 01 企业

objectType integer($int32) 节点对象类型：1 当前分析人员；2 本案人员；3 非本案人员； 4 陌生电话；5 陌生卡号

organization string 所在单位

position string 职务

realName string 姓名

sex string 性别： 男 女

}

]

relationVOS [

{

intimacy integer($int32) 亲密度 0-100

relationTypeVOS [

{

relationType string 关系类型

score number 该维度的分数

}

]

source string 源节点

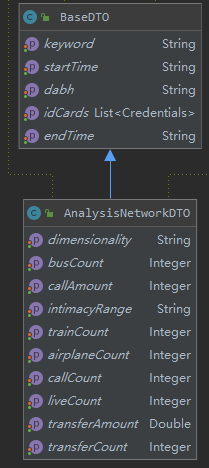
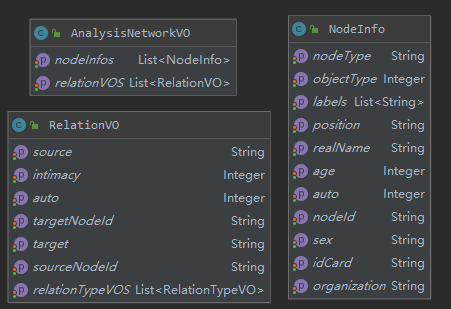
target string 目标节点

}

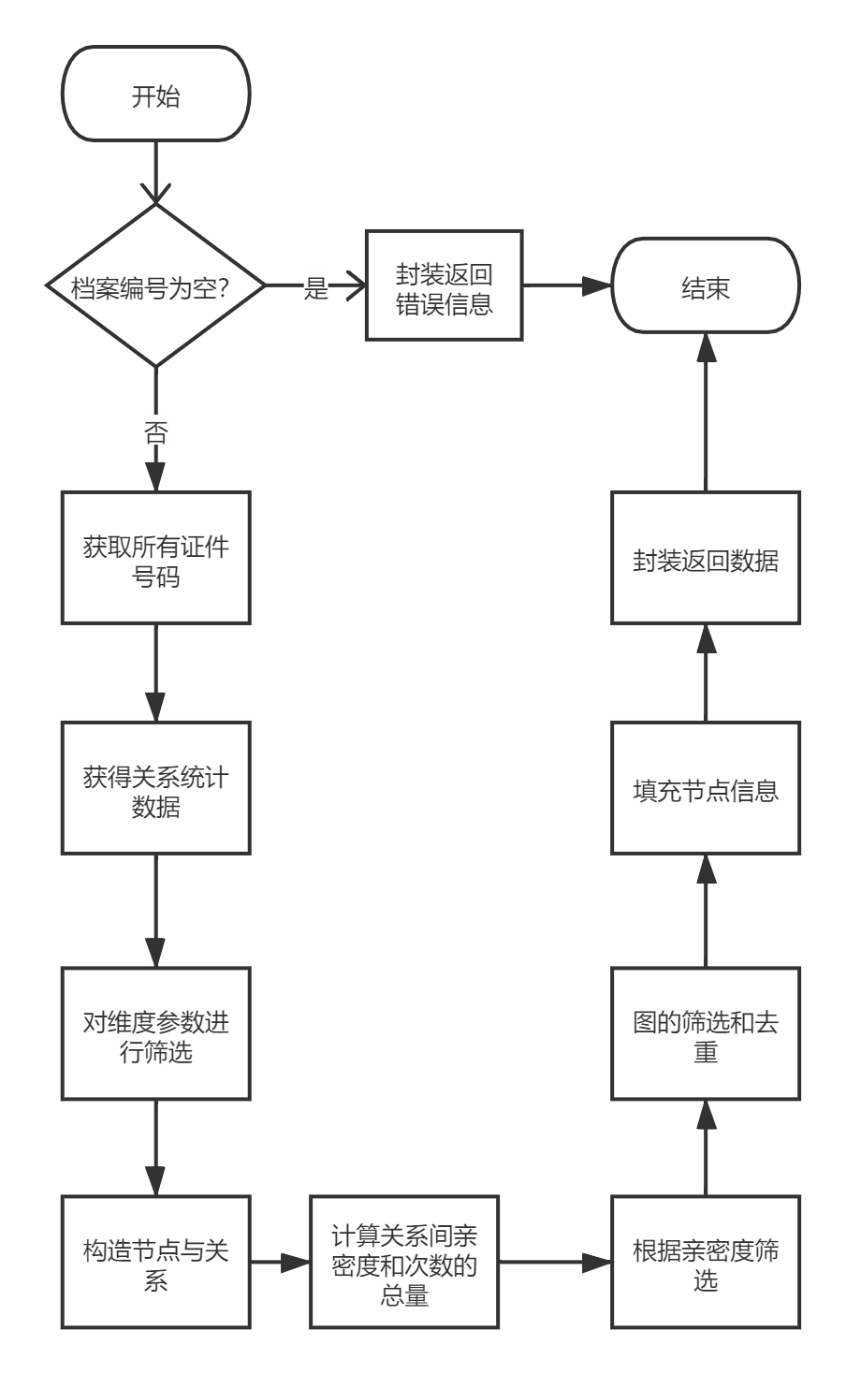
]

}

##### 类图

##### 流程图



##### 处理过程

1. 判断档案编号是否为空，如果没有直接结束
2. 获取传输的所有证件号码，若无则获取案件内所有证件号码
3. 获得案件内所有关系统计数据
4. 根据维度参数要求进行筛选
5. 构造节点关系图
6. 聚合计算关系间亲密度和次数总量
7. 亲密度筛选
8. 图筛选去重
9. 填充节点信息，返回最终结果

##### 质量要求

要求传递的档案编号字段不为空

#### 刷新关系网分析

##### 描述

在进行关系网分析或者加载历史分析记录之后，可手动添加节点，点击刷新会对当前分析结果进行数据更新，并保留手动添加的节点。

##### 输入

{

关系网刷新分析传输类

airplaneCount integer($int32) 飞机同行次数

busCount integer($int32) 大巴同行次数

callAmount integer($int32) 通话时长

callCount integer($int32) 通话次数

【必须】dabh string 档案编号

dimensionality string 维度：01 同户 02 企业主要人员 03 银行卡转账 04 同住 05 同行-飞机 06 同行-大巴 07 同行-火车 08 投资 09 通话，多个维度之间以逗号分隔

endTime string 结束时间 yyyyMMdd

idCards [

证件号集合

Credentials{

idCard string 证件号

logo string 证件标识 00 个人 01 企业

}

]

intimacyRange string 亲密度范围,例如：20-40

keyword string 关键字查询

liveCount integer($int32) 同住次数

nodeInfos [

{

节点信息

age integer($int32) 年龄

auto integer($int32) 0 系统节点 1 手动节点

idCard string 身份证号码或者企业社会代码

labels [标签列表]

nodeId string 节点对象id（身份证号码或者企业社会代码或电话、银行卡）

nodeType string 节点类型 00 个人 01 企业

objectType integer($int32) 节点对象类型：1 当前分析人员；2 本案人员；3 非本案人员； 4 陌生电话；5 陌生卡号

organization string 所在单位

position string 职务

realName string 姓名

sex string 性别： 男 女

}

]

relations [

{

关系信息

auto integer($int32) 0 系统关系 1 手动关系

intimacy integer($int32) 亲密度 0-100

relationTypeVOS [

{

关系类型列表

relationType string 关系类型

score number 该维度的分数

}

]

source string 源节点证件号码

sourceNodeId string 源节点nodeId

target string 目标节点证件号码

targetNodeId string 目标节点证件号码

}

]

startTime string 开始时间 yyyyMMdd

trainCount integer($int32) 火车同行次数

transferAmount number($double) 交易金额transferCount integer($int32) 交易次数

}

##### 输出

{

nodeInfos [

{

age integer($int32) 年龄

idCard string 身份证号码或者企业社会代码

labels [...标签列表...]

nodeId string 节点对象id（身份证号码或者企业社会代码或电话、银行卡）

nodeType string 节点类型 00 个人 01 企业

objectType integer($int32) 节点对象类型：1 当前分析人员；2 本案人员；3 非本案人员； 4 陌生电话；5 陌生卡号

organization string 所在单位

position string 职务

realName string 姓名

sex string 性别： 男 女

}

]

relationVOS [

{

intimacy integer($int32) 亲密度 0-100

relationTypeVOS [

{

relationType string 关系类型

score number 该维度的分数

}

]

source string 源节点

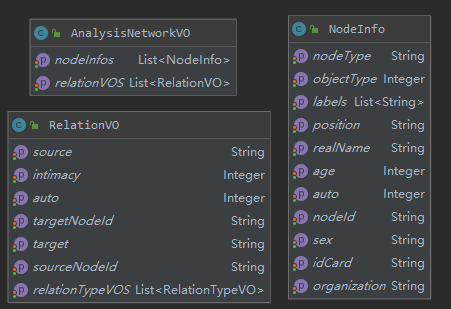
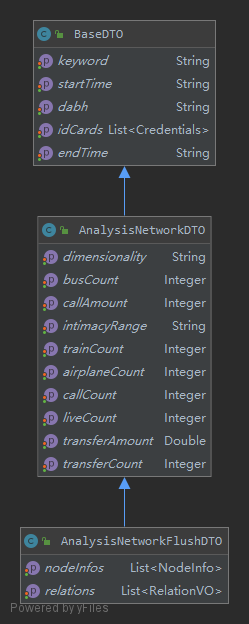
target string 目标节点

}

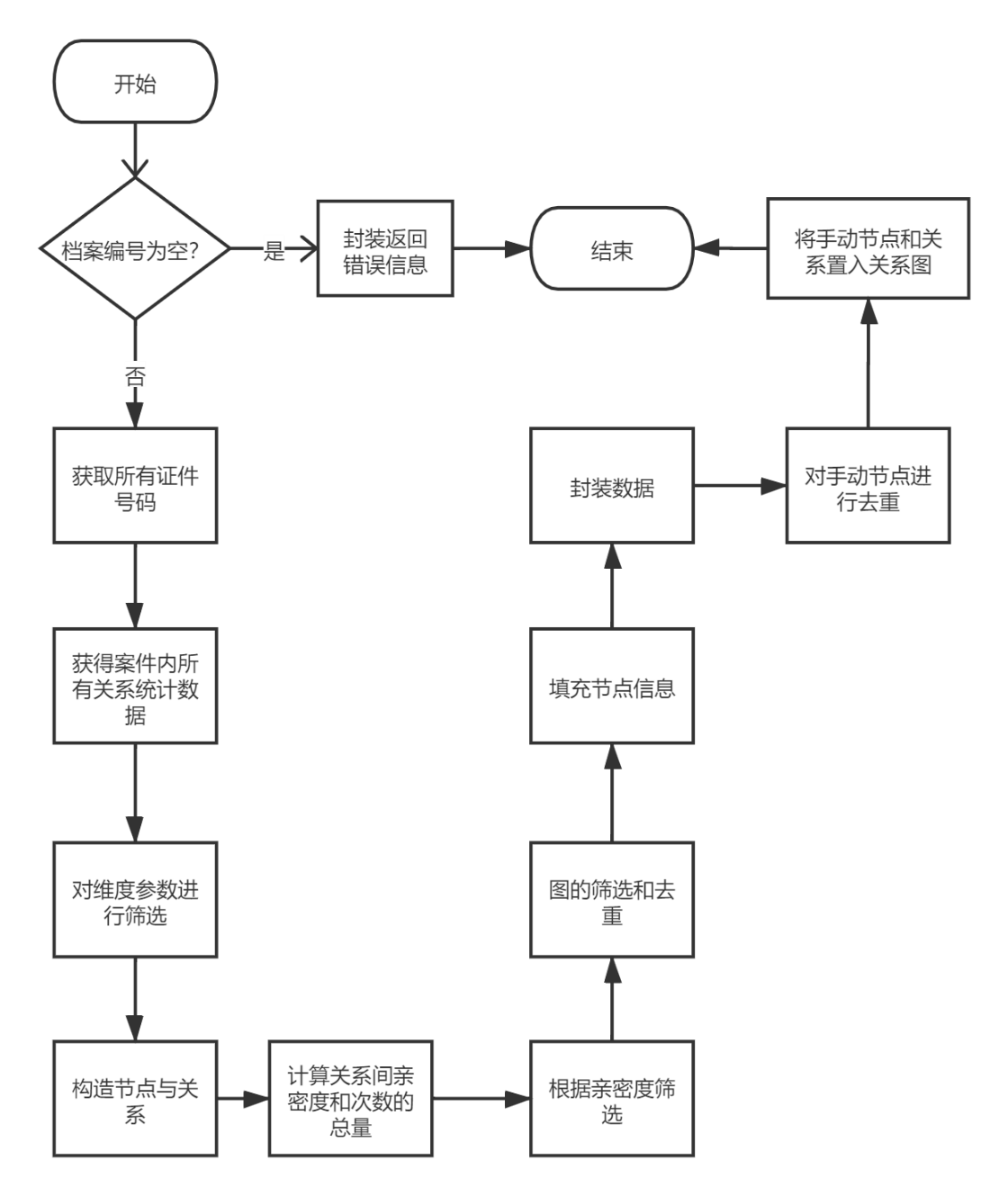
]

}

##### 类图



##### 流程图



##### 处理过程

1. 判断档案编号是否为空，如果没有直接结束
2. 获取传输的所有证件号码，若无则获取案件内所有证件号码
3. 获得案件内所有关系统计数据
4. 根据维度参数要求进行筛选
5. 构造节点关系图
6. 聚合计算关系间亲密度和次数总量
7. 亲密度筛选
8. 图筛选去重
9. 填充节点信息
10. 对传输的手动节点列表去重
11. 将手动节点和关系置入新的关系图完成更新并返回

##### 质量要求

要求传递的档案编号字段不为空，若存在手动添加的节点应将节点与关系信息一并传递

#### 保存分析结果

##### 描述

进行关系网分析之后，点击“保存分析”可对分析结果进行保存

##### 输入

{

【必须】caseId string 案件id

【必须】content string 分析内容

desc string 记录描述

queryCriteria string 查询条件

【必须】querySaveType integer($int32) 分析记录类型，关系网分析为1

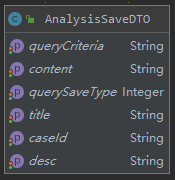
title string 记录标题

}

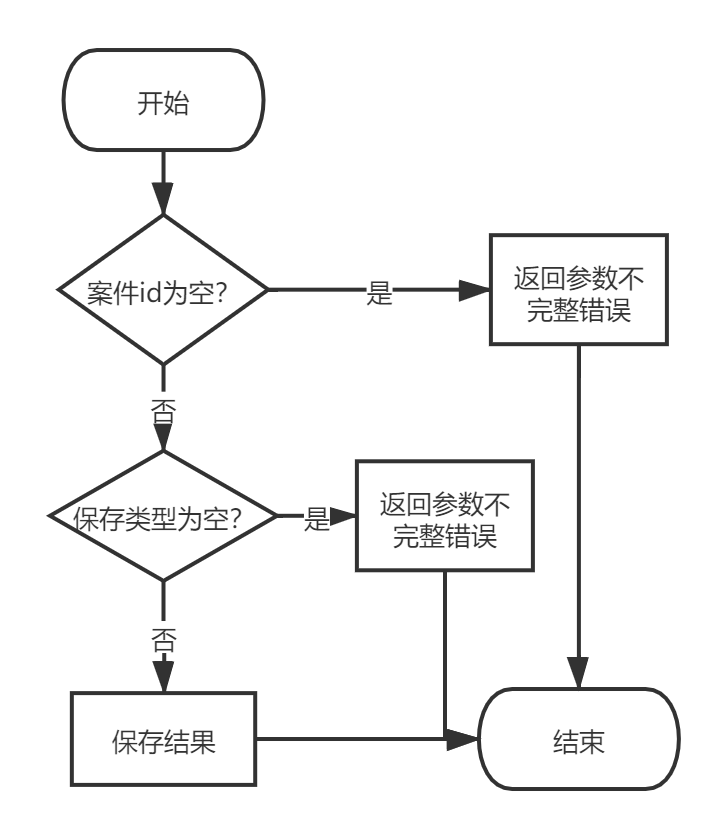
##### 输出

保存成功提示

##### 类图



##### 流程图



##### 处理过程

1. 判断案件id是否为空
2. 判断保存类型是否为空
3. 保存结果并返回成功信息

##### 质量要求

需要传递案件id和保存类型，保存类型应为1

#### 查看历史分析记录

##### 描述

查看某个案件的所有分析记录

##### 输入

caseId string 案件id

querySaveType Integer 记录类型,关系网为1

##### 输出

{

caseId string 案件id

content string 分析内容

createTime string($date-time) 保存时间

desc string 记录描述

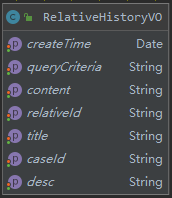
id string 分析记录id

queryCriteria string 查询条件,前端解析

title string 记录标题

}

##### 类图



##### 流程图



##### 处理过程

1. 根据案件id和查询类型查询出案件所有关系网分析记录
2. 根据分析时间倒序排序并分页
3. 封装结果集返回

##### 质量要求

需要传递案件id和保存类型，保存类型应为1

#### 删除分析结果

##### 描述

在分析历史记录表单页面可选择对某一记录进行删除

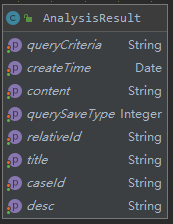
##### 输入

relativeId string 分析内容id

##### 输出

删除成功信息

##### 类图



##### 流程图



##### 处理过程

1. 判断分析记录是否存在
2. 根据分析记录唯一主键删除分析记录

##### 质量要求

需要传递分析记录唯一主键

#### 同行-大巴详情

##### 描述

在列表详情页面查看各个对象之间的关系表单，点击“查看详情”时可调用本接口查看同行-大巴维度的详细信息

##### 输入

{

dabh string档案编号

endTime string结束时间 yyyyMMdd

idCards [

证件号集合

Credentials{

idCard string 证件号

logo string 证件标识 00 个人 01 企业

}

]

keyword string关键字查询

pageNum integer($int32)页码

pageSize integer($int32)每页条数

sortName string排序字段

sortRule string排序规则(ASC,DESC)

sourceIdCard stringsource证件

startTime string开始时间 yyyyMMdd

targetIdCard stringtarget证件

}

##### 输出

{

同行-大巴

bccx string 班车车型

bch string 班次号

bclx string 班次类型

ccrxm string 乘车人姓名

ccrzjlx string 乘车人证件类型

cph string 车牌号

cpjg number 车票价格

dabh string 档案编号

dfccrxm string 对方乘车人姓名

dfccrzjlx string 对方乘车人证件类

dfcpjg number 对方车票价格

dfmddmc string 对方目的地名称

dfqpsj string 对方取票时间

dfsabh string 对方涉案编号

dfzjhm string 对方证件号码

dfzwh string 对方座位号

fcrq string 发车日期yyyyMMdd

mddmc string 目的地名称

qpsj string 取票时间

sabh string 涉案编号

sfczbm string 始发车站编码

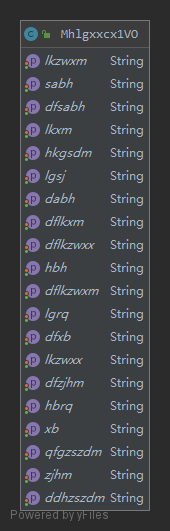
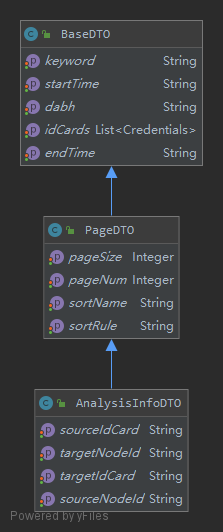
sfczmc string 始发车站名称

zjhm string 证件号码

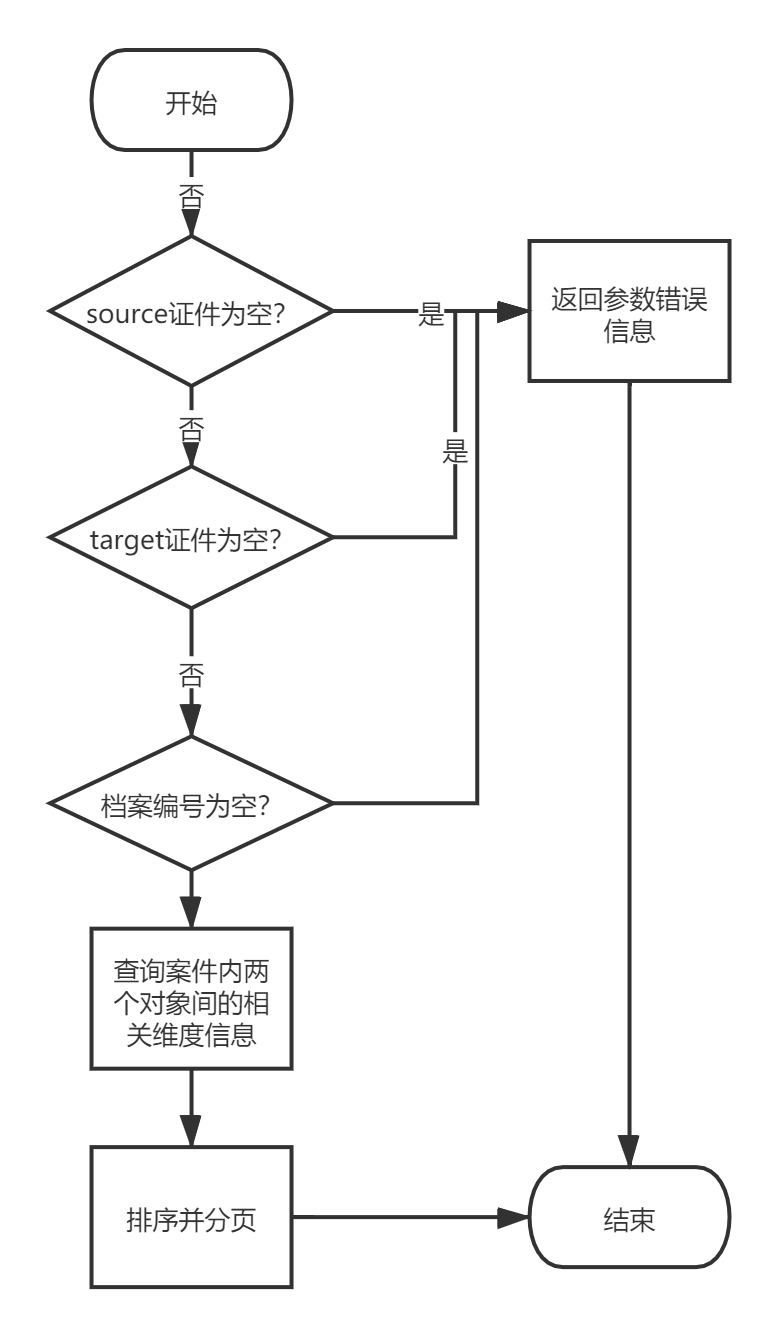
zwh string 座位号

}

##### 类图



##### 流程图



##### 处理过程

1. 判断source证件、target证件和档案编号是否为空
2. 查询案件内两个对象间的相关维度信息
3. 排序并分页，封装分页信息。

##### 质量要求

source证件、target证件和档案编号必须传递

#### 同行-飞行详情

##### 描述

在列表详情页面查看各个对象之间的关系表单，点击“查看详情”时可调用本接口查看同行-飞行维度的详细信息

##### 输入

{

dabh string 档案编号

endTime string 结束时间 yyyyMMdd

idCards [

证件号集合

Credentials{

idCard string 证件号

logo string 证件标识 00 个人 01 企业

}

]

keyword string 关键字查询

pageNum integer($int32) 页码

pageSize integer($int32) 每页条数

sortName string 排序字段

sortRule string 排序规则(ASC,DESC)

sourceIdCard string source证件

startTime string 开始时间 yyyyMMdd

targetIdCard string target证件

}

##### 输出

{

同行-飞机

dabh string 档案编号

ddhzszdm string 到达航站三字代码

dflkxm string 对方旅客姓名

dflkzwxm string 对方旅客中文姓名

dflkzwxx string 对方旅客座位信息

dfsabh string 对方涉案编号

dfxb string 对方性别

dfzjhm string 对方证件号码

hbh string 航班号

hbrq string 航班日期

hkgsdm string 航空公司代码

lgrq string 离港日期yyyyMMdd

lgsj string 离港时间

lkxm string 旅客姓名

lkzwxm string 旅客中文姓名

lkzwxx string 旅客座位信息

qfgzszdm string 起飞港站三字代码

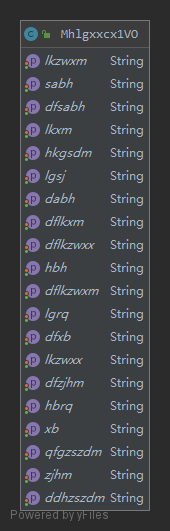
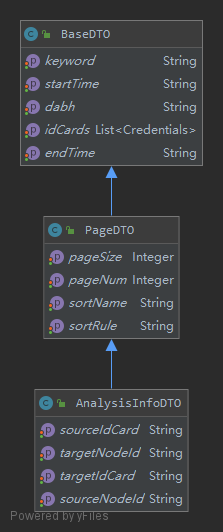
sabh string 涉案编号

xb string 性别

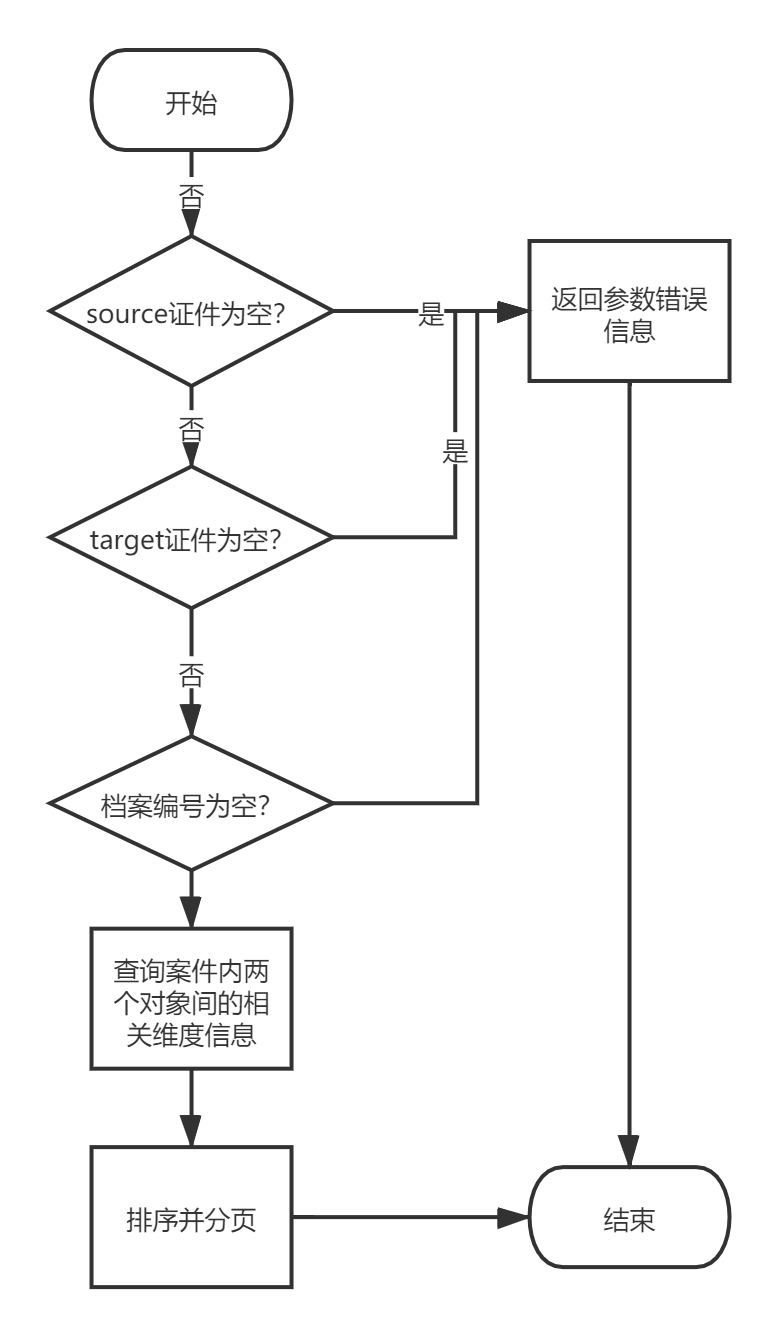
zjhm string 证件号码

}

##### 类图



##### 流程图



##### 处理过程

1. 判断source证件、target证件和档案编号是否为空
2. 查询案件内两个对象间的相关维度信息
3. 排序并分页，封装分页信息。

##### 质量要求

source证件、target证件和档案编号必须传递

#### 同行-火车详情

##### 描述

在列表详情页面查看各个对象之间的关系表单，点击“查看详情”时可调用本接口查看同行-火车维度的详细信息

##### 输入

{

dabh string 档案编号

endTime string 结束时间 yyyyMMdd

idCards [

证件号集合

Credentials{

idCard string 证件号

logo string 证件标识 00 个人 01 企业

}

]

keyword string 关键字查询

pageNum integer($int32) 页码

pageSize integer($int32) 每页条数

sortName string 排序字段

sortRule string 排序规则(ASC,DESC)

sourceIdCard string source证件

startTime string 开始时间 yyyyMMdd

targetIdCard string target证件

}

##### 输出

{

同行-火车

ch string 车号

cxh string 档案编号

dabh string 档案编号

dfcxh string 对方车厢号

dfdz string 对方到站

dffz string 对方发站

dfsabh string 对方证件号码

dfxm string 对方姓名

dfzjhm string 对方证件号码

dfzwh string 对方座位号

dz string 档案编号

fcrq string 发车日期

fz string 档案编号

rq string 发车日期yyyyMMdd

sabh string 档案编号

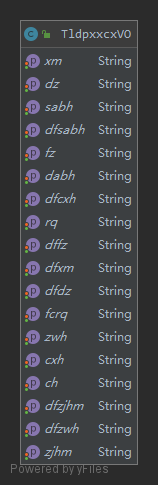
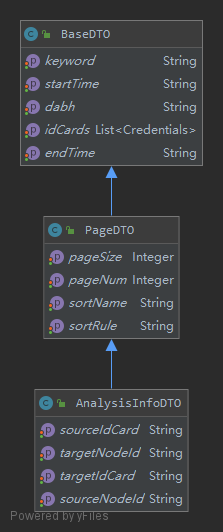
xm string 档案编号

zjhm string 档案编号

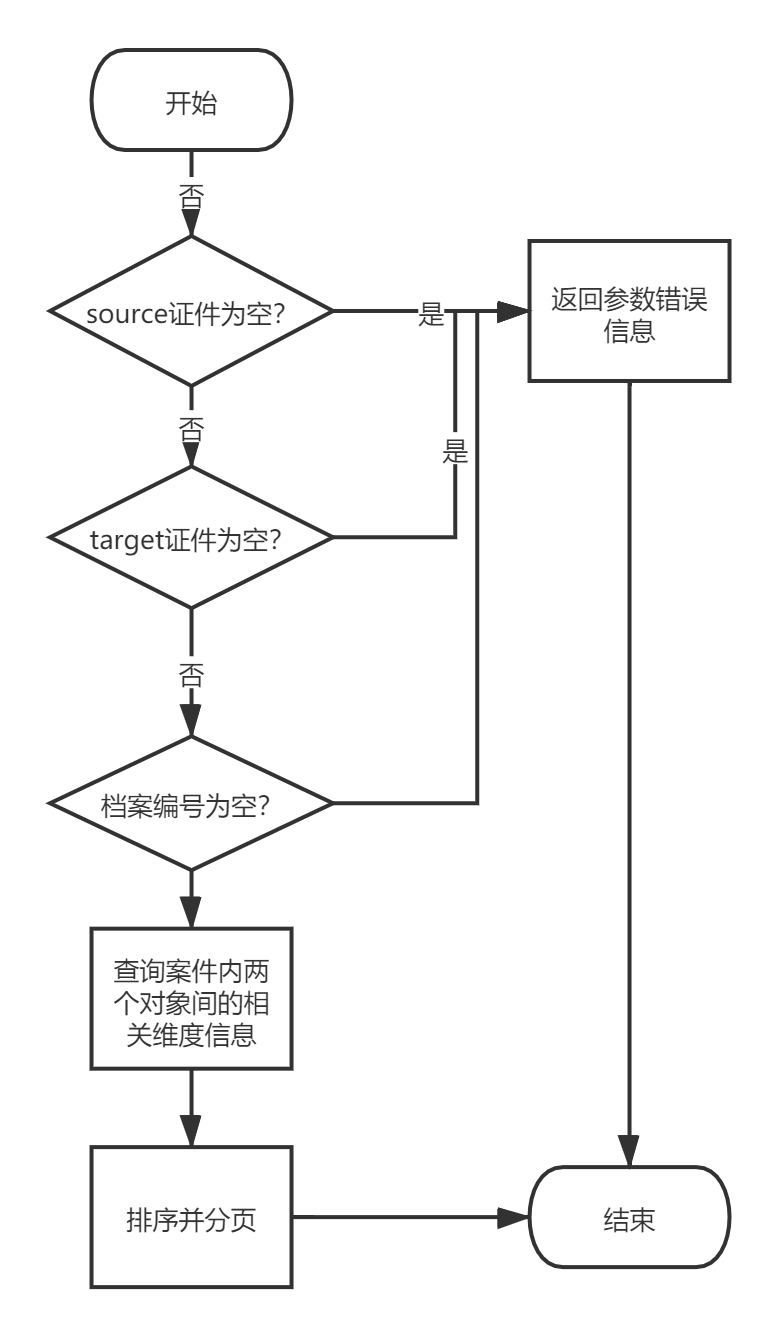
zwh string 档案编号

}

##### 类图



##### 流程图



##### 处理过程

1. 判断source证件、target证件和档案编号是否为空
2. 查询案件内两个对象间的相关维度信息
3. 排序并分页，封装分页信息。

##### 质量要求

source证件、target证件和档案编号必须传递

#### 投资详情

##### 描述

在列表详情页面查看各个对象之间的关系表单，点击“查看详情”时可调用本接口查看投资的详细信息

##### 输入

{

dabh string 档案编号

endTime string 结束时间 yyyyMMdd

idCards [

证件号集合

Credentials{

idCard string 证件号

logo string 证件标识 00 个人 01 企业

}

]

keyword string 关键字查询

pageNum integer($int32) 页码

pageSize integer($int32) 每页条数

sortName string 排序字段

sortRule string 排序规则(ASC,DESC)

sourceIdCard string source证件

startTime string 开始时间 yyyyMMdd

targetIdCard string target证件

}

##### 输出

{

投资

dabh string 档案编号

gd\_sabh string 股东涉案编号

gd\_sfsa string 股东是否涉案：是、否

gdgb string 股东国别

gdlx string 股东类型

gdxm string 股东姓名

qymc string 企业名称

rjczbl string 认缴出资比例

rjczbz string 认缴出资币种

rjczfs string 认缴出资方式

rjczjewy number 认缴出资金额（万元）

rjczrq string 认缴出资日期

rjczsj string 认缴出资时间

sabh string 涉案编号

tyshxydm string 统一社会信用代码

zjhm string 证件号码

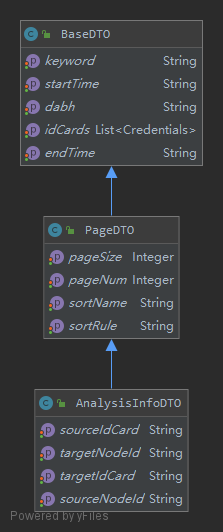
zjlx string 证件类型

zzhm string 证照号码

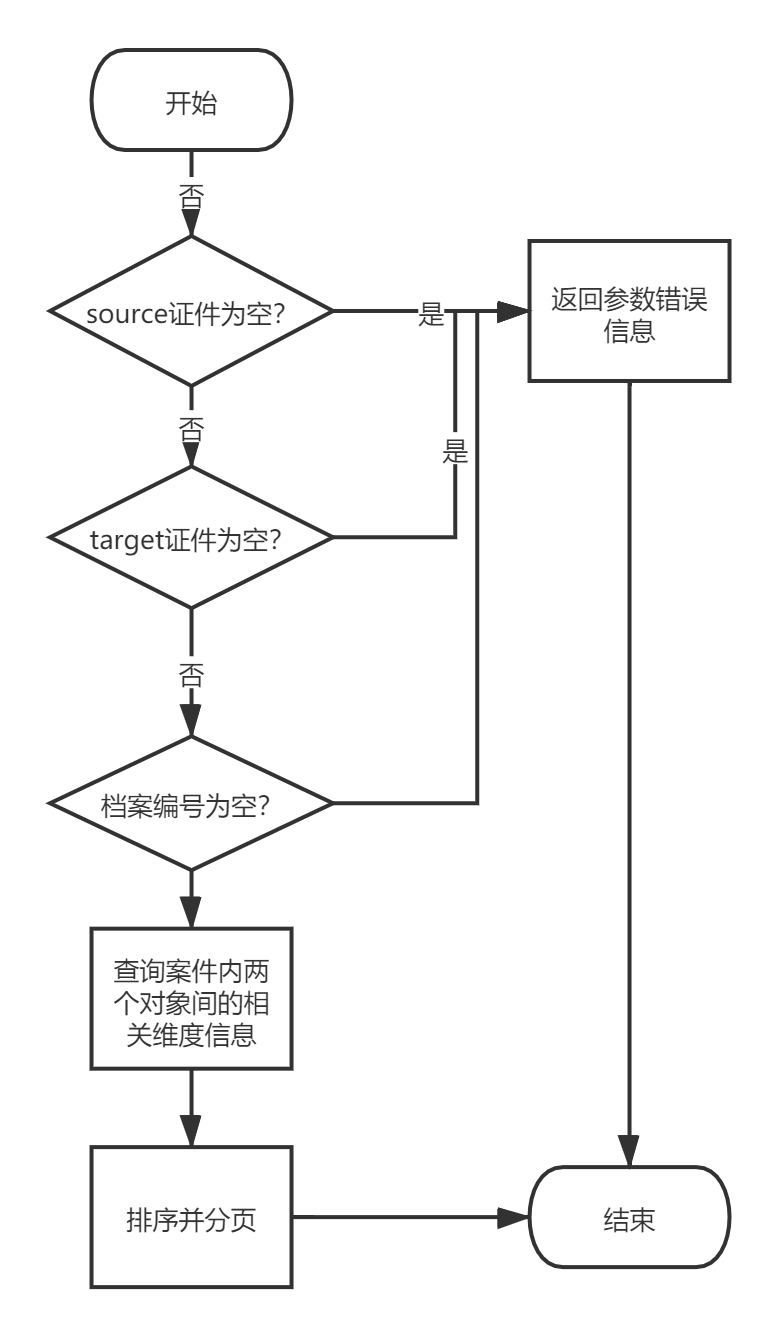
zzlx string 证照类型

}

##### 类图



##### 流程图



##### 处理过程

1. 判断source证件、target证件和档案编号是否为空
2. 查询案件内两个对象间的相关维度信息
3. 排序并分页，封装分页信息。

##### 质量要求

source证件、target证件和档案编号必须传递

#### 同户详情

##### 描述

在列表详情页面查看各个对象之间的关系表单，点击“查看详情”时可调用本接口查看维度的详细信息

##### 输入

{

dabh string 档案编号

endTime string 结束时间 yyyyMMdd

idCards [

证件号集合

Credentials{

idCard string 证件号

logo string 证件标识 00 个人 01 企业

}

]

keyword string 关键字查询

pageNum integer($int32) 页码

pageSize integer($int32) 每页条数

sortName string 排序字段

sortRule string 排序规则(ASC,DESC)

sourceIdCard string source证件

startTime string 开始时间 yyyyMMdd

targetIdCard string target证件

}

##### 输出

{

同户

csdgjdq string 出生地国家

csdssx string 出生地省市县

csrq string 出生日期

cym string 曾用名

dabh string 档案编号

dfcsdgjdq string 对方出生地国家

dfcsdssx string 对方出生地省市县

dfcsrq string 对方出生日期

dfcym string 对方曾用名

dffwsc string 对方服务所处

dfjggjdq string 对方籍贯国家

dfjgssxq string 对方籍贯省市县

dfmz string 对方民族

dfsabh string 对方涉案编号

dfsfsa string 对方是否涉案

dfsg string 对方身高

dfsspcsmc string 对方所属派出所名称

dfssssxq string 对方所属省市县

dfxb string 对方性别

dfxm string 对方姓名

dfyhzgx string 对方与户主关系

dfzjhm string 对方身份证号

dfzzxz string 对方住址详址

fwsc string 服务所处

jggjdq string 籍贯国家

jgssxq string 籍贯省市县

mz string 民族

sabh string 涉案编号

sfsa string 是否涉案

sg string 身高

sspcsmc string 所属派出所名称

ssssxq string 所属省市县

xb string 性别

xm string 姓名

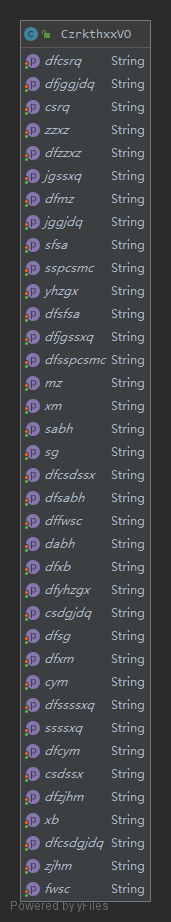
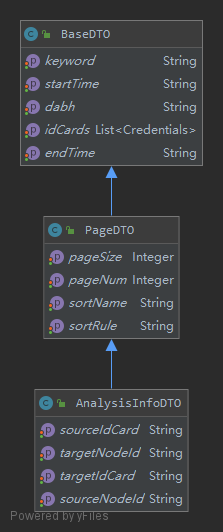
yhzgx string 与户主关系

zjhm string 身份证号

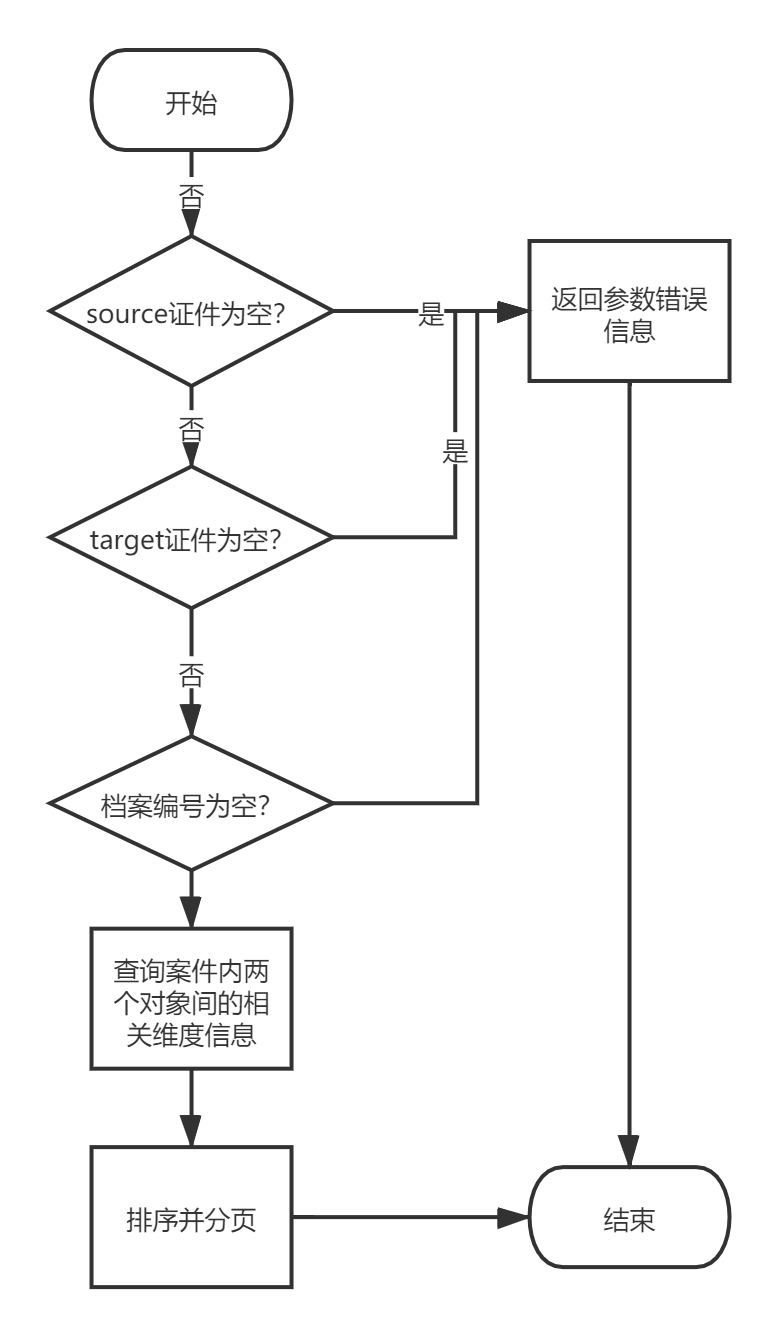
zzxz string 住址详址

}

##### 类图



##### 流程图



##### 处理过程

1. 判断source证件、target证件和档案编号是否为空
2. 查询案件内两个对象间的相关维度信息
3. 排序并分页，封装分页信息。

##### 质量要求

source证件、target证件和档案编号必须传递

#### 企业主要人员详情

##### 描述

在列表详情页面查看各个对象之间的关系表单，点击“查看详情”时可调用本接口查看企业主要人员维度的详细信息

##### 输入

{

dabh string 档案编号

endTime string 结束时间 yyyyMMdd

idCards [

证件号集合

Credentials{

idCard string 证件号

logo string 证件标识 00 个人 01 企业

}

]

keyword string 关键字查询

pageNum integer($int32) 页码

pageSize integer($int32) 每页条数

sortName string 排序字段

sortRule string 排序规则(ASC,DESC)

sourceIdCard string source证件

startTime string 开始时间 yyyyMMdd

targetIdCard string target证件

}

##### 输出

{

企业主要人员

dabh string 档案编号

gr\_sabh string 个人涉案编号

gr\_sfsa string 个人是否涉案：是、否

sabh string 涉案编号

tyshxydm string 统一社会信用代码

xm string 姓名

zjhm string 证件号码

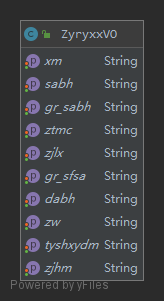
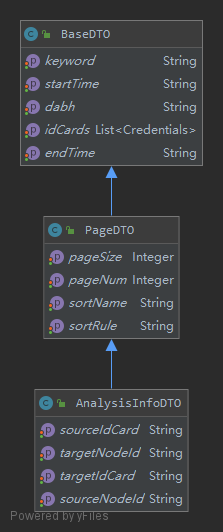
zjlx string 证件类型

ztmc string 主体名称

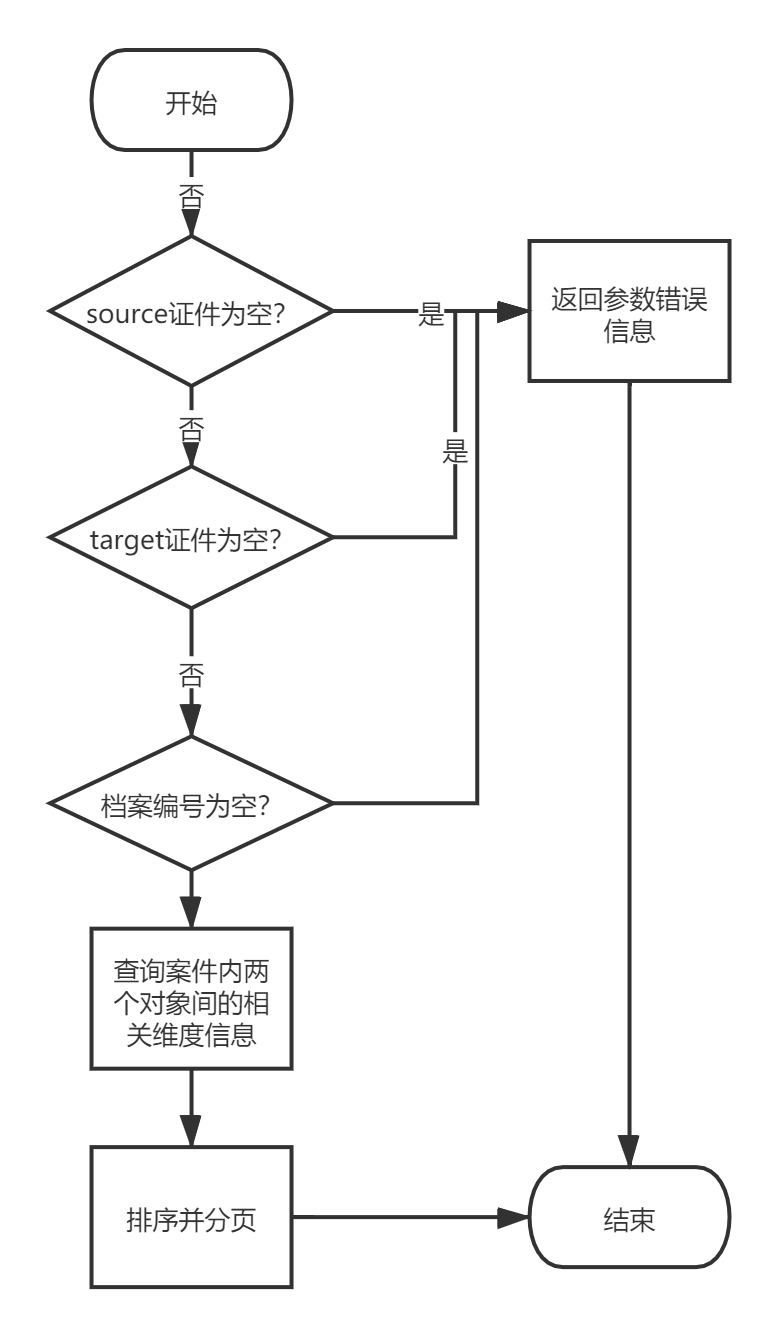
zw string 职务

}

##### 类图



##### 流程图



##### 处理过程

1. 判断source证件、target证件和档案编号是否为空
2. 查询案件内两个对象间的相关维度信息
3. 排序并分页，封装分页信息。

##### 质量要求

source证件、target证件和档案编号必须传递

#### 同住详情

##### 描述

在列表详情页面查看各个对象之间的关系表单，点击“查看详情”时可调用本接口查看同住维度的详细信息

##### 输入

{

dabh string 档案编号

endTime string 结束时间 yyyyMMdd

idCards [

证件号集合

Credentials{

idCard string 证件号

logo string 证件标识 00 个人 01 企业

}

]

keyword string 关键字查询

pageNum integer($int32) 页码

pageSize integer($int32) 每页条数

sortName string 排序字段

sortRule string 排序规则(ASC,DESC)

sourceIdCard string source证件

startTime string 开始时间 yyyyMMdd

targetIdCard string target证件

}

##### 输出

{

旅馆

dabh string 档案编号

dfduration number 对方住宿时长

dfrzfh string 对方入住房号

dfrzsj string 对方入住时间

dfsabh string 对方涉案编号

dfsfzmhm string 对方证件号码

dfsfzmmc string 对方身份证明名称

dftfsj string 对方退房时间

dfxb string 对方性别

dfxm string 对方名称

duration number 住宿时长

ldmc string 旅店名称

ldxxdz string 旅店详细地址

rzfh string 入住房号

rzrq string 本方入住日期yyyyMMdd

rzsj string 入住时间

sabh string 涉案编号

sfzmhm string 证件号码

sfzmmc string 身份证明名称

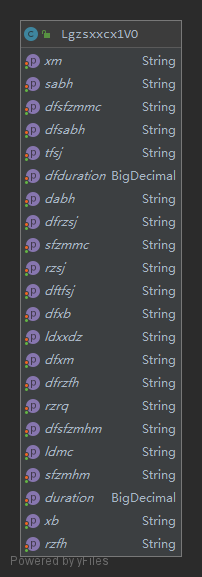
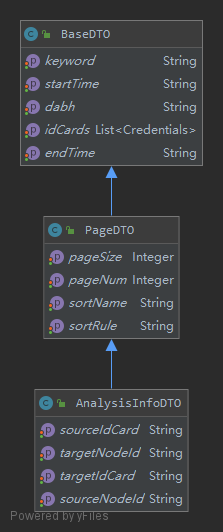
tfsj string 退房时间

xb string 性别

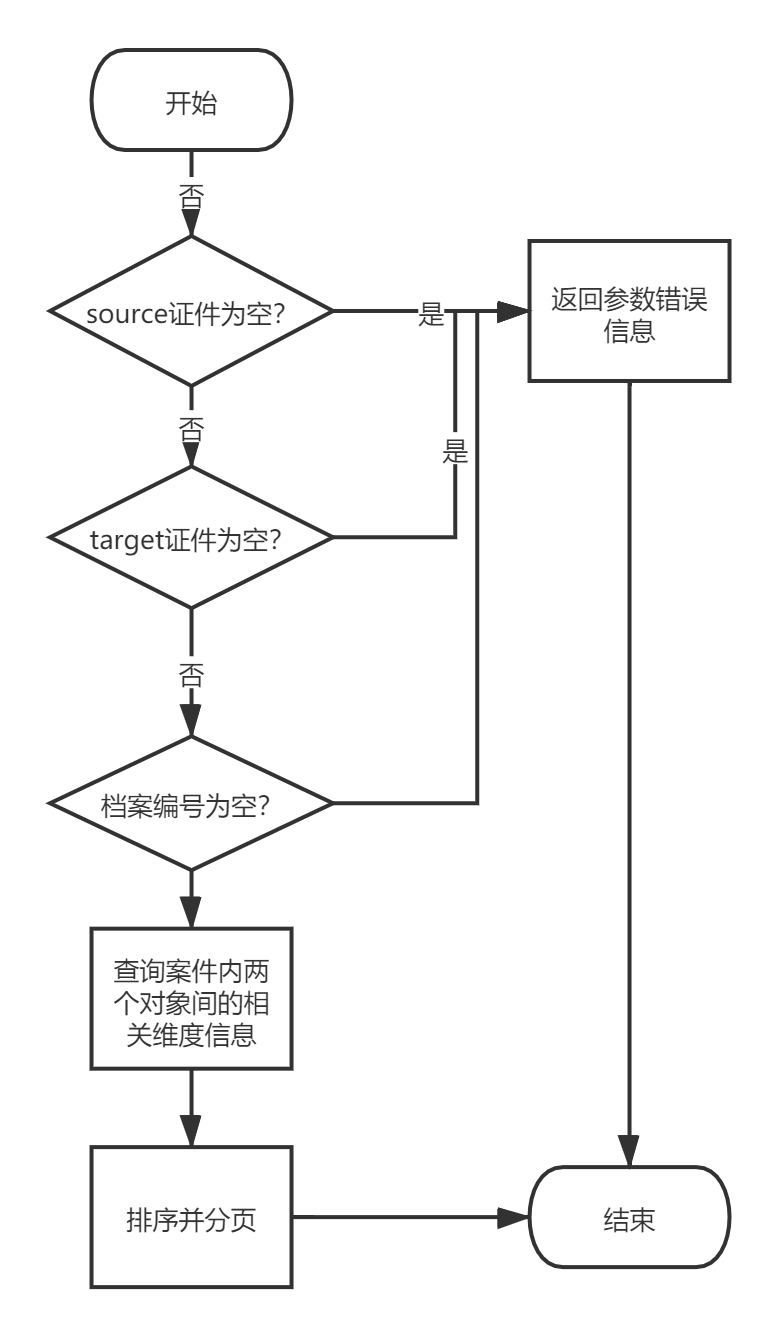
xm string 姓名

}

##### 类图



##### 流程图



##### 处理过程

1. 判断source证件、target证件和档案编号是否为空
2. 查询案件内两个对象间的相关维度信息
3. 排序并分页，封装分页信息。

##### 质量要求

source证件、target证件和档案编号必须传递

#### 交易详情

##### 描述

在列表详情页面查看各个对象之间的关系表单，点击“查看详情”时可调用本接口查看交易维度的详细信息

##### 输入

{

交易传参类

amount string 交易金额

dabh\* string 档案编号

endPeriod string 交易时段 结束

endTime string 结束时间 yyyyMMdd

idCards [

证件号集合

Credentials{

idCard string 证件号

logo string 证件标识 00 个人 01 企业

}

]

keyword string 关键字查询

pageNum integer($int32) 页码

pageSize integer($int32) 每页条数

sortName string 排序字段

sortName1 string 排序字段

sortRule string 排序规则(ASC,DESC)

sortRule1 string 排序规则(ASC,DESC)

sourceIdCard string source证件

sourceNodeId string source NodeID

startPeriod string 交易时段 开始

startTime string 开始时间 yyyyMMdd

targetIdCard string target证件

targetNodeId string target NodeID

transferType string 交易类型

}

##### 输出

{

账户交易明细

bz string 币种

cxkh string 查询卡号

dabh string 档案编号

jdbz string 借贷标志

je number 交易金额

jydfkh string 对方卡号

jydfsabh string 交易对方涉案编号

jydfsalx string 交易对方涉案类型

jydfxm string 交易对方名称

jydfzjhm string 对方证件号

jydsye number 交易对手余额

jyfsd string 交易发生地

jylsh string 交易流水号

jylx string 交易类型

jyrq string 交易日期yyyyMMdd

jysj string 交易时间

jywdmc string 交易网点名称

mc string 名称

sabh string 涉案编号

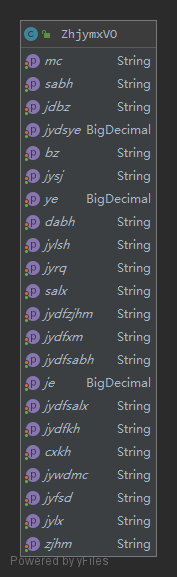
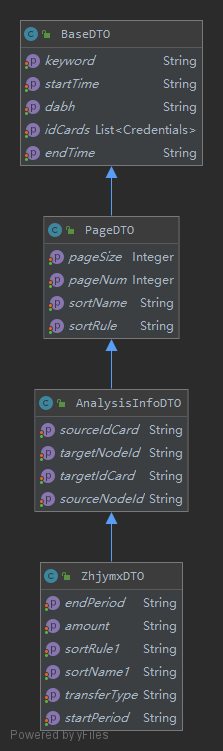
salx string 涉案类型

ye number 交易余额

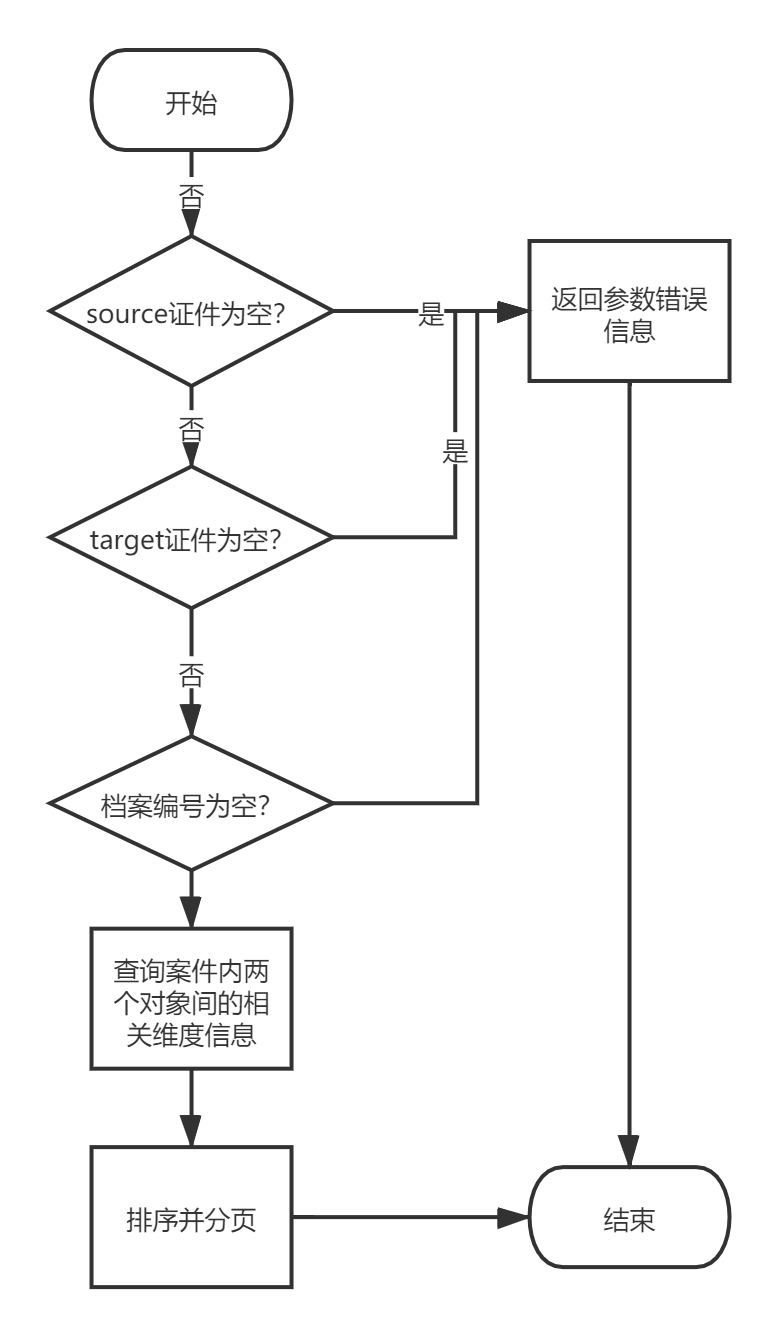
zjhm string 证件号码

}

##### 类图



##### 流程图



##### 处理过程

1. 判断source证件、target证件和档案编号是否为空
2. 查询案件内两个对象间的相关维度信息
3. 排序并分页，封装分页信息。

##### 质量要求

source证件、target证件和档案编号必须传递

### 研判推演

#### 查询各维度数据条数接口

##### 描述

用于综合分析-研判推演功能中，进行自动扩展的分析维度选择时，显示该节点对象案件内各个维度上存在直接关系的对象数。

##### 输入

{

各维度关联对象数-参数类

【必须】dabh\* string 档案编号

endTime string 结束时间 yyyyMMdd

【必须，存在就传】idCard string 证件id

【必须，存在就传】nodeId string 节点对象id（身份证号码或者企业社会代码或电话、银行卡）

startTime string 开始时间 yyyyMMdd

}

##### 输出

[{

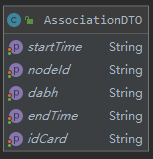
各维度关联对象数-展示类

dimensionality string 维度：01 同户 02 同事 03 银行卡转账 04 同住 05 同行-飞机 06 同行-大巴 07 同行-火车 08 投资 09 通话

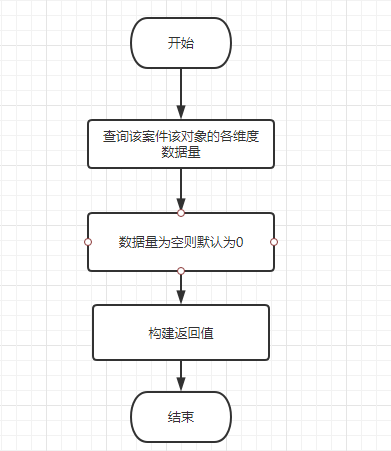
num integer($int32) 关联对象数量

}]

##### 类图



##### 流程图



##### 处理过程

1. 查询该案件该案件对象的各维度数据量
2. 维度上无数据则置0
3. 构建返回值

##### 质量要求

要求传递案件编号

#### 研判推演

##### 描述

用于综合分析-研判推演功能中，进行自动扩展。在右侧“对象选择”列表中单选一个对象，或者在已有节点上选择自动扩展，然后选择的维度和维度筛选信息。点击“确认扩展”调用该接口，分析案件内该与对象直接关联的所有对象，并返回下一级相关节点的信息。

##### 输入

{

研判推演传参类

airplaneCount integer($int32) 飞机同行次数

busCount integer($int32) 大巴同行次数

callAmount integer($int32) 通话时长

callCount integer($int32) 通话次数

【必须】dabh string 档案编号

dimensionality string 维度：01 同户 02 同事 03 银行卡转账 04 同住 05 同行-飞机 06 同行-大巴 07 同行-火车 08 投资 09 通话，多个维度之间以逗号分隔

endTime string 结束时间 yyyyMMdd

【必须，存在就传】idCard string 证件id

intimacyRange string 亲密度范围,例如：20-40

【必须】level integer($int32) 节点层级，根节点为0

liveCount integer($int32) 同住次数

【必须，存在就传】nodeId string 节点对象id（身份证号码或者企业社会代码或电话、银行卡）

startTime string 开始时间 yyyyMMdd

trainCount integer($int32) 火车同行次数

transferAmount number($double) 交易金额

transferCount integer($int32) 交易次数

}

##### 输出

{

description:

关系网分析前端展示类

nodeInfos [

{

节点信息

age integer($int32) 年龄

auto integer($int32) 0 系统节点 1 手动节点

idCard string 身份证号码或者企业社会代码

labels [标签列表]

nodeId string 节点对象id（身份证号码或者企业社会代码或电话、银行卡）

nodeType string 节点类型 00 个人 01 企业

objectType integer($int32) 节点对象类型：1 当前分析人员；2 本案人员；3 非本案人员； 4 陌生电话；5 陌生卡号

organization string 所在单位

position string 职务

realName string 姓名

sex string 性别： 男 女

}

]

relations [

{

关系信息

auto integer($int32) 0 系统关系 1 手动关系

intimacy integer($int32) 亲密度 0-100

relationTypeVOS [

{

关系类型列表

relationType string 关系类型

score number 该维度的分数

}

]

source string 源节点证件号码

sourceNodeId string 源节点nodeId

target string 目标节点证件号码

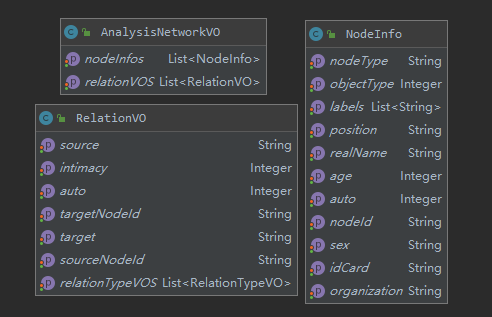
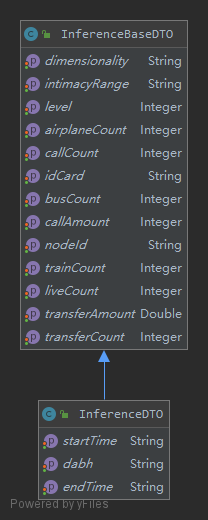
targetNodeId string 目标节点证件号码

}

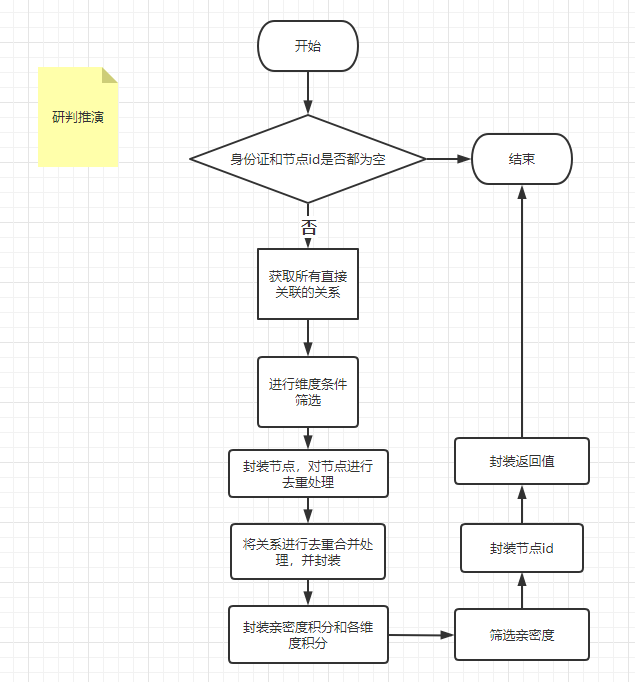
]

}

##### 类图



##### 流程图



##### 处理过程

1. 判断身份证和节点id为空
2. 获取所有直接关联的关系
3. 进行维度条件筛选
4. 封装节点并去重
5. 关系去重、合并、封装
6. 封装亲密度积分和各维度积分
7. 亲密度筛选
8. 节点id封装
9. 封装信息并返回

##### 质量要求

要求传递档案编号，需要选择当前案件中一个对象作为推演起点，idCards和nodeIds只要存在就应传递

#### 刷新推演

##### 描述

用于综合分析-研判推演功能中，已经读取某个分析记录，欲对该记录的数据进行更新。点击“更新”按钮，调用该接口将手动扩展节点信息与自动扩展的条件信息发往后端，后端根据自动扩展条件重新分析推演，并将结果与手动扩展节点拼接起来，再返回。

##### 输入

{

研判推演刷新传参类

【必须】dabh string 档案编号

endTime string 结束时间 yyyyMMdd

idCards [

证件号集合

Credentials{

idCard string 证件号

logo string 证件标识 00 个人 01 企业

}

]

keyword string 关键字查询

nodeInfos [

{

研判推演返回的节点信息

age integer($int32) 年龄

auto integer($int32) 0 系统节点 1 手动节点

【必须，存在就传】idCard string 身份证号码或者企业社会代码

labels [标签列表]

【必须】level integer($int32) 节点层级，根节点为0

【必须，存在就传】nodeId string 节点对象id（身份证号码或者企业社会代码或电话、银行卡）

nodeType string 节点类型 00 个人 01 企业

objectType integer($int32) 节点对象类型：1 当前分析人员；2 本案人员；3 非本案人员； 4 陌生电话；5 陌生卡号

organization string 所在单位

position string 职务

realName string 姓名

relation InferenceRelationVO

{研判推演关系信息

auto integer($int32) 0 系统关系 1 手动关系

intimacy integer($int32) 亲密度 0-100

relationTypeVOS

[

{关系类型列表

relationType string 关系类型

score number 该维度的分数

}

]

}

sex string 性别： 男 女

}

]

params [

{

研判推演基础传参类

airplaneCount integer($int32) example: 9

飞机同行次数

busCount integer($int32) 大巴同行次数

callAmount integer($int32) 通话时长

callCount integer($int32) 通话次数

dimensionality string 维度：01 同户 02 同事 03 银行卡转账 04 同住 05 同行-飞机 06 同行-大巴 07 同行-火车 08 投资 09 通话，多个维度之间以逗号分隔

idCard string 证件id

intimacyRange string 亲密度范围,例如：20-40

【必须】level\* integer($int32) 节点层级，根节点为0

liveCount integer($int32) 同住次数

【必须】nodeId string 节点对象id（身份证号码或者企业社会代码或电话、银行卡）

trainCount integer($int32) 火车同行次数

transferAmount number($double) 交易金额

transferCount integer($int32) 交易次数

}

]

startTime string 开始时间 yyyyMMdd

}

##### 输出

{

刷新研判推演前端展示类

childNodes [

{

研判推演返回的节点信息

age integer($int32) 年龄

auto integer($int32) 0 系统节点 1 手动节点

idCard string 身份证号码或者企业社会代码

labels [...]

【必须】level integer($int32) 节点层级，根节点为0

nodeId string 节点对象id（身份证号码或者企业社会代码或电话、银行卡）

nodeType string 节点类型 00 个人 01 企业

objectType integer($int32) 节点对象类型：1 当前分析人员；2 本案人员；3 非本案人员； 4 陌生电话；5 陌生卡号

organization string 所在单位

position string 职务

realName string 姓名

relation InferenceRelationVO{

研判推演关系信息

auto integer($int32) 0 系统关系 1 手动关系

intimacy integer($int32) 亲密度 0-100

relationTypeVOS [

{

关系类型列表

relationType string 关系类型

score number 该维度的分数

}]

}

sex string 性别： 男 女

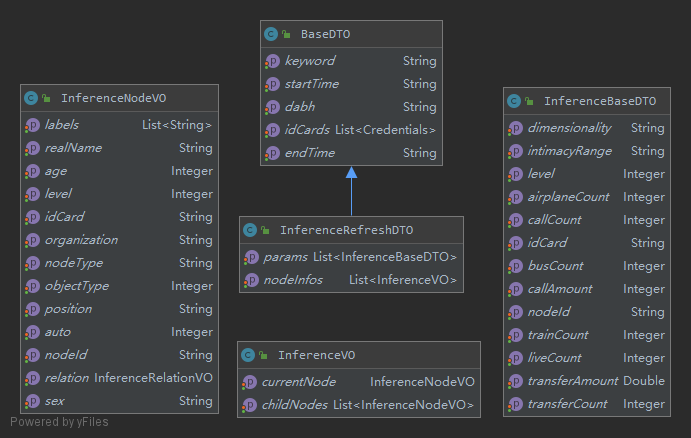
}

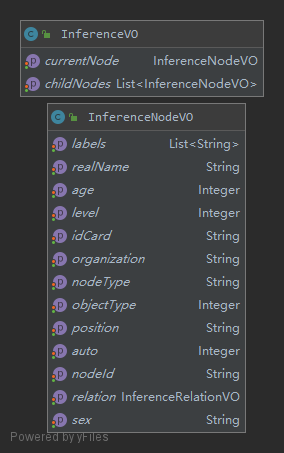
]

currentNode InferenceNodeVO{同childNodes}

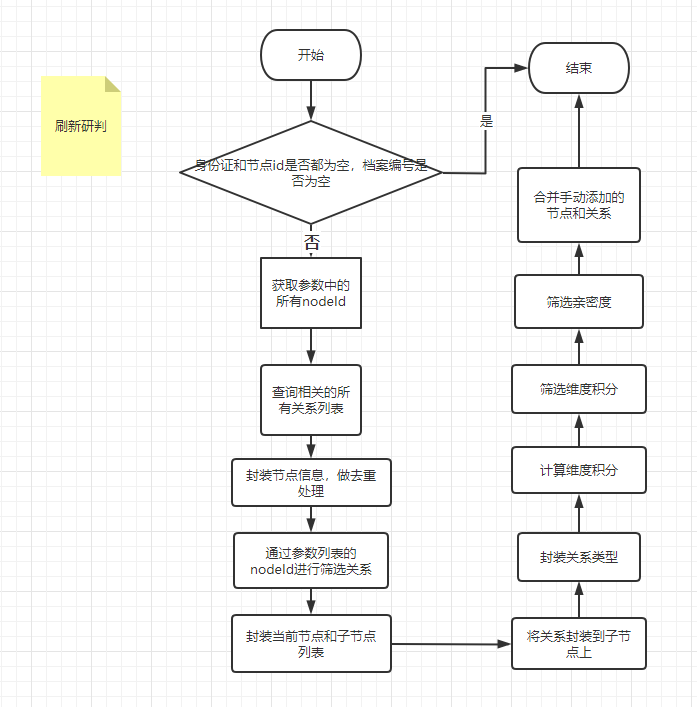
}

##### 类图





##### 流程图



##### 处理过程

1. 判断身份证和节点id为空
2. 获取参数中的nodeId
3. 获取相关的所有关系列表
4. 封装节点并去重
5. 通过参数列表的nodeId进行关系筛选
6. 封装当前节点和子节点列表
7. 将关系封装到子节点上
8. 封装关系类型
9. 计算维度积分
10. 亲密度和维度积分筛选
11. 合并手动添加的节点和关系
12. 封装信息并返回

##### 质量要求

要求传递档案编号，idCards和nodeIds只要存在就应传递

### 落脚点分析

#### 活动轨迹

##### 描述

在地图上展示人员的地理位置信息，并统计出现次数。

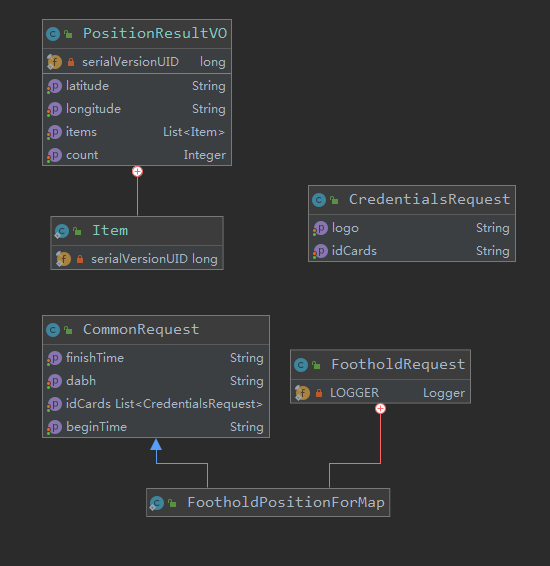
##### 输入

档案编号、起止时间、常用时间段、数据维度。

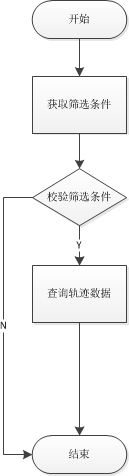
##### 输出

落脚点信息，落脚点相关事件信息。

##### 类图



##### 流程图



##### 处理过程

1. 校验参数；
2. 查询轨迹；
3. 返回数据给终端。

##### 质量要求

无。

#### 事件统计

##### 描述

列表展示人员相同地点的事件频次。可以查看频次对应的详情。

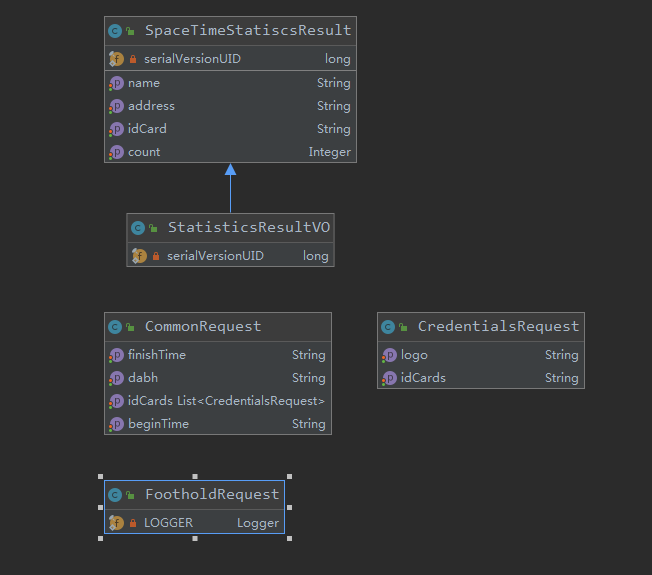
##### 输入

档案编号、起止时间、常用时间段、数据维度。

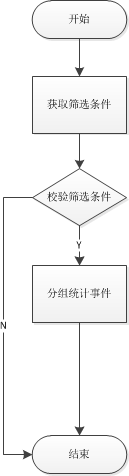
##### 输出

频次统计信息。

##### 类图



##### 流程图



##### 处理过程

1. 校验筛选条件；
2. 分组统计事件；
3. 返回统计结果给终端；
4. 根据筛选条件和统计结果分别查询不同的明细表；
5. 返回明细结果给终端。

##### 质量要求

无。

### 数据比对

#### 描述

审查人员可以在画板上拖拽两个数据源，包括系统数据源、另存为数据源、上传数据源。编辑数据源的查询条件。选择比对类型，包括交、并、差，并编辑关联字段、关联结果字段。点击开始按钮，进行离线的数据比对操作，可以预览或导出查看。

#### 输入

|  |
| --- |
| 数据源编码a |
| 模版编码a |
| 数据源编码b |
| 模版编码b |
| 查询条件a |
| 查询条件b |
| 比对类型 |
| 关联字段a |
| 关联字段b |
| 展示字段a |
| 展示字段b |

#### 输出

比对结果集

#### 类图



#### 流程图



#### 处理过程

1. 选择数据源
2. 输入查询条件
3. 选择比对类型
4. 编辑关联字段、关联结果字段
5. 点击开始进行比对
6. 查看比对结果
7. 保存模型

#### 质量要求

要求用户合理选择数据源及比对字段，服务准确返回比对结果。

## 大数据模型库

### 寻找情人模型

#### 描述

在当前案件中对某涉案人员进行分析，寻找疑似情人的对象。

#### 输入

{

查找潜在情人查询类

【必须】dabh\* string 档案编号

【必须】idCard\* string 身份证号

pageNum integer($int32) 分页页码

pageSize integer($int32) 分页大小

startTime string 开始时间 yyyyMMdd

endTime string 结束时间 yyyyMMdd

}

#### 输出

{

查找潜在情人前端展示类

together integer($int32) 同行同住人数

transfer integer($int32) 交易异常人数

call integer($int32) 通话异常人数

togetherAndCall integer($int32) 同行同住且通话异常人数

togetherAndTransfer integer($int32) 同行同住且交易异常人数

transferAndCall integer($int32) 交易异常且通话异常人数

allCount integer($int32) 三个维度都存在异常人数

suspiciousData [

{

可疑关系列表前端展示类

idCard string 对方身份证号

liveNum string 同住次数

name string 对方姓名

sex string 对方性别

suspiciousLevel integer($int32) 可疑程度

callNum string 通话异常次数

transferNum string 交易异常次数

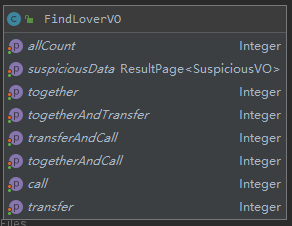
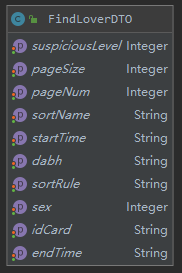
travelNum string 同行次数

}

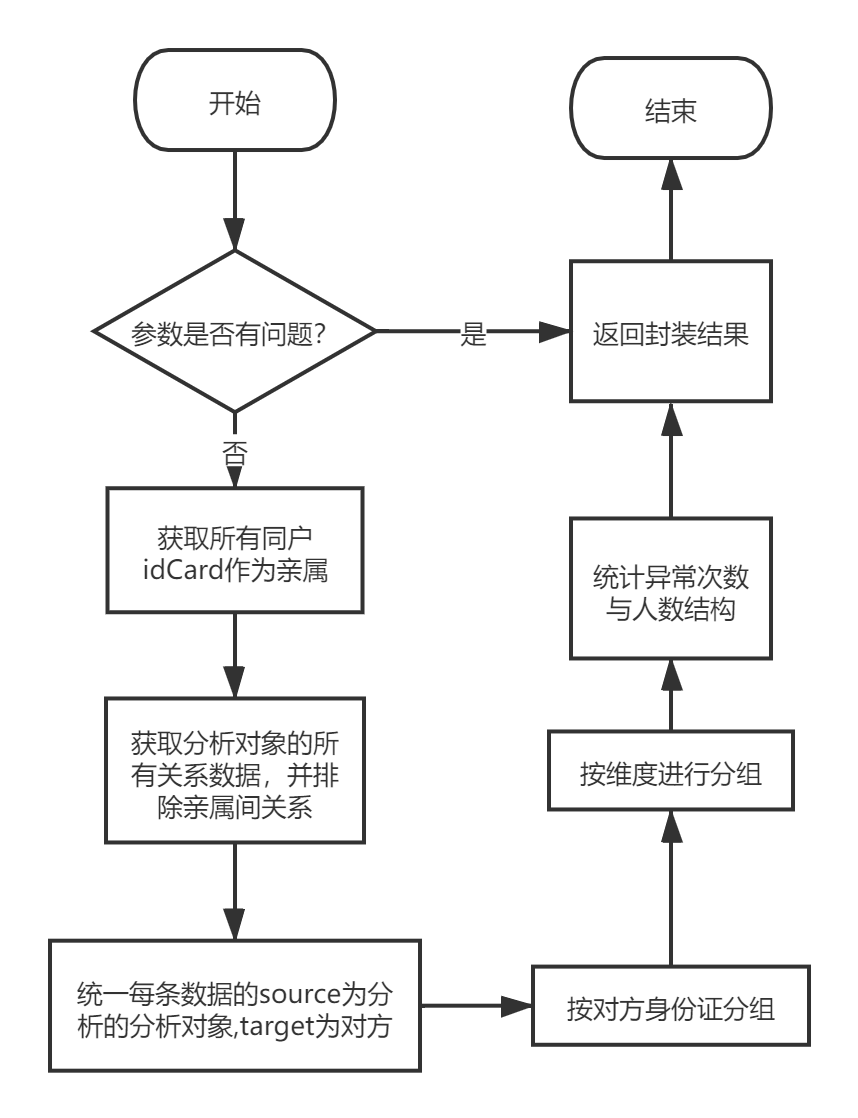
]

}

#### 类图



#### 流程图



#### 处理过程

1. 判断参数传递是否完整规范

2. 根据关系汇总表单查询出该分析对象相关所有关系并去除亲属关系，按对方身份证号分类

3. 统一每条数据的source为分析的分析对象,target为对方

4. 按维度统计异常数

5. 计算异常人数封装返回

#### 质量要求

要求传递的档案编号字段不为空，选择的对象必须为人，只有人才能拥有情人，所以要求需要传递idCard。

### 共同关系分析模型

#### 描述

选中多个分析对象，查看该案件与这些分析对象有共同关系的人员情况

图表和详情表格可共用此接口

#### 输入

|  |  |
| --- | --- |
| **callAmount** | **integer($int32) *example: 5* 通话时长** |
| **callCount** | **integer($int32) *example: 6* 通话次数** |
| **dabh\*** | **String 档案编号** |
| **dimensionality** | **String *example: 01,02***  **维度：01 同户 02 企业主要人员 03 银行卡转账 04 同住 05 同行-飞机 06 同行-大巴 07 同行-火车 08 投资 09 通话 10 通讯录，多个维度之间以逗号分隔** |
| **endTime** | **String 结束时间 yyyyMMdd** |
| **idCard** | **String 证件id** |
| **num\*** | **integer($int32) 关联人数** |
| **startTime** | **String 开始时间 yyyyMMdd** |
| **transferAmount** | **number($double) *example: 22.22* 交易金额** |
| **transferCount** | **integer($int32) *example: 4* 交易次数** |

#### 输出

[

共同关系分析模型前端展示类

CommonRelationVO{

age integer($int32) 年龄

commonRelationVOS [

{

"age": 0,

"dimensionalityList": [

"通话",

"交易"

],

"idCard": "string",

"nodeId": "string",

"nodeType": "string",

"objectType": 0,

"realName": "string",

"sex": "string",

"total": 0

}

]

dimensionalityList [] 维度列表 example: [ "通话", "交易" ]

idCard string身份证号码或者企业社会代码

nodeId string节点对象id（身份证号码或者企业社会代码或电话、银行卡）

nodeType string 节点类型 00 涉案人员 01 涉案企业 02 非涉案人员 03 非涉案企业 04 不确定归属的银行卡 05 不确定归属的电话卡

objectType integer($int32) 节点对象类型：1 当前分析人员；2 本案人员；3 非本案人员； 4 陌生电话；5 陌生卡号

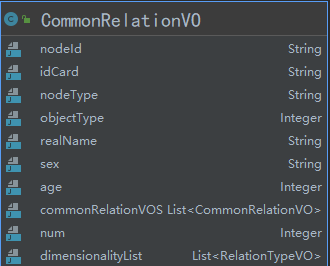
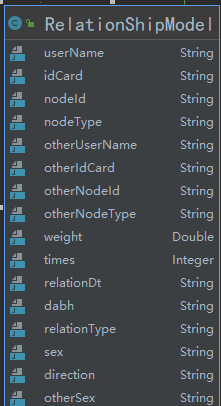
realName string姓名

sex string性别： 男 女

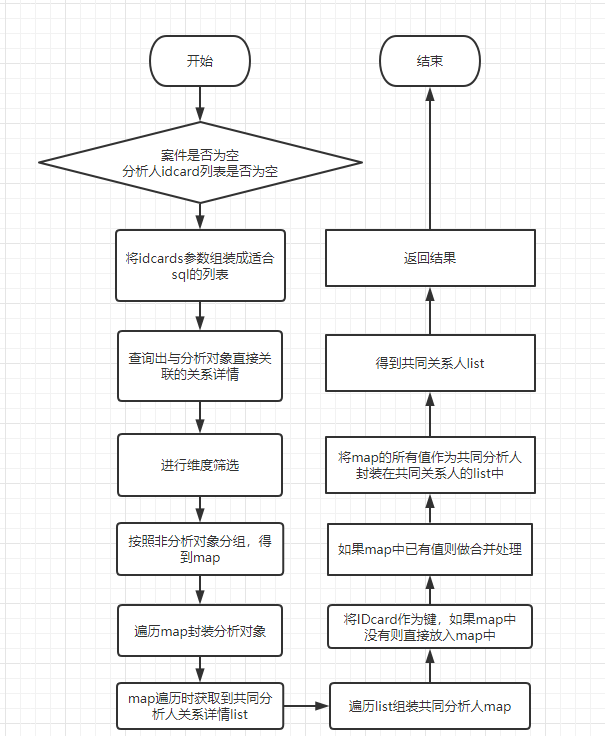
total integer($int32) 共同联系人数

}]

#### 类图



#### 流程图



#### 处理过程

1. 校验参数
2. 将参数idcards组装成适合sql的list
3. 查询出分析对象直接关联的关系详情
4. 进行维度筛选
5. 按照分析对象进行分组，得到map
6. 遍历map封装分析对象
7. 遍历分析对象关系详情list
8. 将list封装成map，以分析人作为主键，将相同的关系封装为一个共同关系实体
9. 最后将map的所有值作为共同关系人的共同分析对象list
10. 返回结果

#### 质量要求

使用HashMap保证时间复杂度较低以加快服务的响应速度，对参数进行校验以及对维度进行筛选以保证返回正确的结果。对非分析对象进行分组然后组装结果以降低空间复杂度和时间复杂度。

### 多时空伴随分析模型

#### 描述

分析案件中，单个分析对象与案件中的其他分析对象，在一段时间内的事件，在固定的时间、空间间隔范围内的伴随情况。

#### 输入

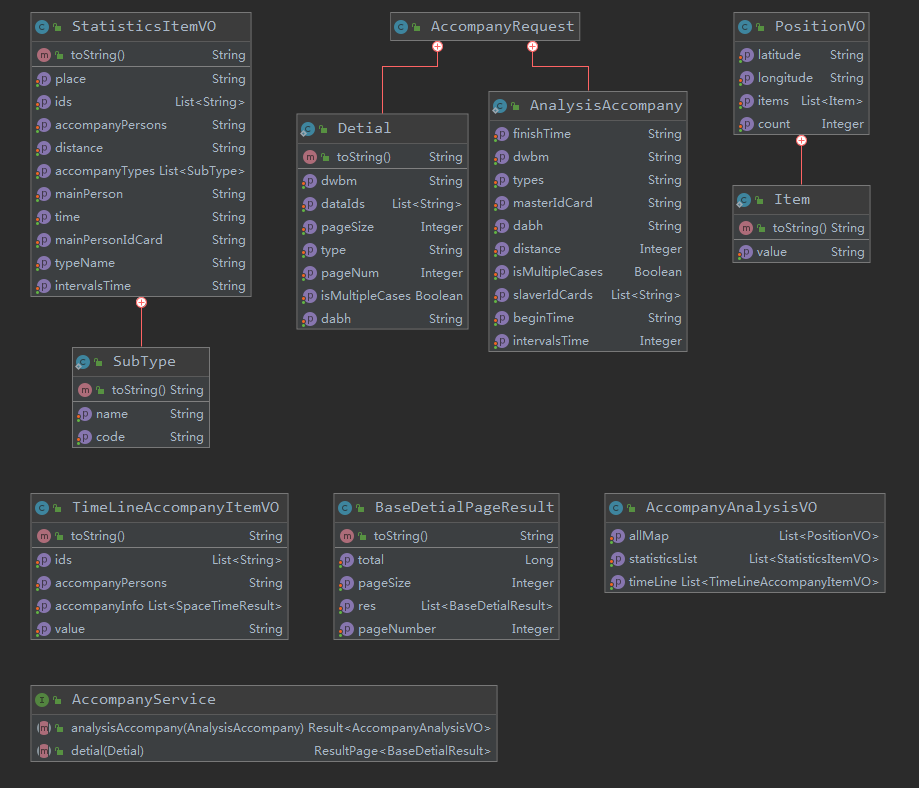
档案编号、分析对象、当前用户所属单位编码、事件起止时间、事件维度、

间隔时间、间隔距离、是否跨案件。

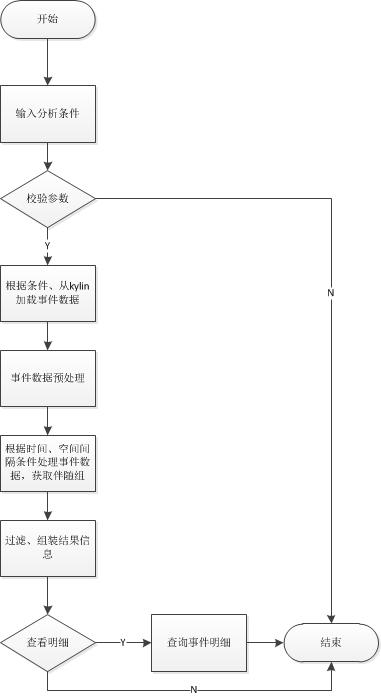
#### 输出

伴随事件位置信息、伴随事件组信息、事件详情。

#### 类图



#### 流程图



#### 处理过程

1. 用户选择分析对象、设定分析条件，
2. 校验参数，档案编号、分析对象、当前用户所属单位编码、事件起止时间为必传参数，其他参数预设默认值，
3. 通过用户所属单位编码和是否跨案件参数获取数据范围，
4. 根据数据范围从kylin加载事件数据（单次分析数据量不大于5万条），
5. 对加载的数据进行预处理，提取出关键字段（时间戳、经纬度）用于下一步处理，
6. 根据分析条件中的间隔时间和间隔距离，判定出伴随事件组，
7. 过滤掉出现次数少于两次的伴随组，
8. 组装出界面呈现的伴随结果数据，
9. 用户可以根据呈现的结果信息查看事件明细。

#### 质量要求

### 数据共振模型

#### 描述

分析案件中，单个分析对象与案件中其他分析对象之间的共振情况。

#### 输入

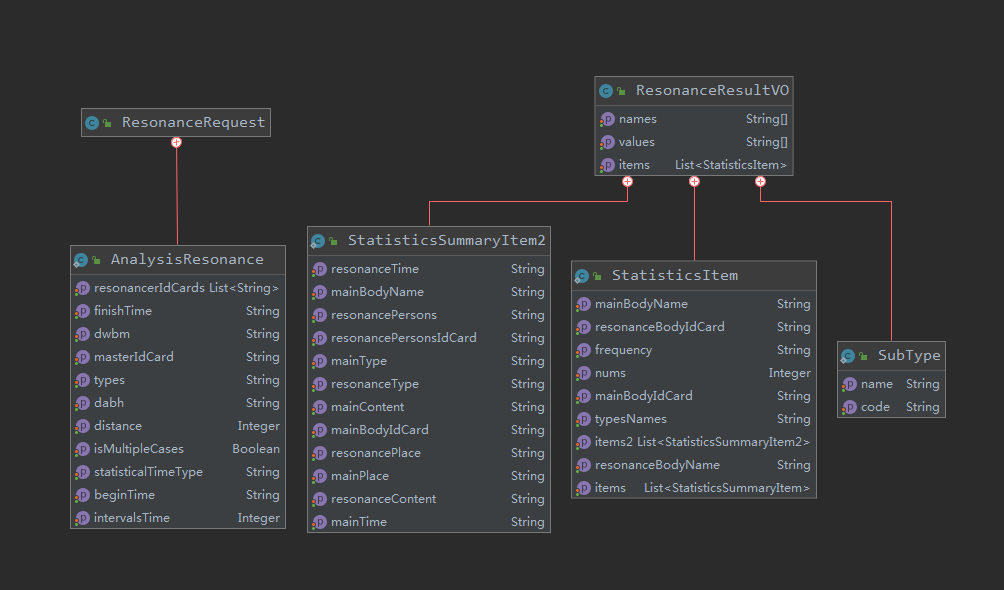
档案编号、分析对象、当前用户所属单位编码、事件起止时间、事件维度、

间隔时间、间隔距离、是否跨案件、统计时间维度。

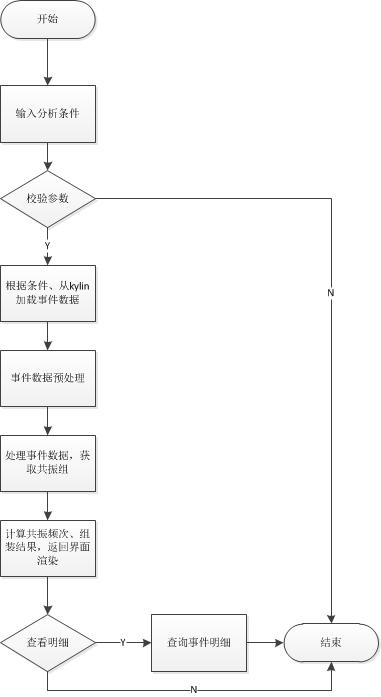
#### 输出

共振列表信息。

#### 类图



#### 流程图



#### 处理过程

1. 用户选择分析对象、设定分析条件，
2. 校验参数，
3. 通过用户所属单位编码和是否跨案件参数获取数据范围，
4. 根据数据范围从kylin加载事件数据（单次分析数据量不大于5万条），
5. 对加载的数据进行预处理，提取出关键字段（时间戳、经纬度）用于下一步处理，
6. 判定出共振组，
7. 计算出共振频率，组装界面呈现界面，
8. 返回数据给界面渲染。

#### 质量要求

分析数据必须有时间、经纬度。采用构建八叉树的方式来寻找事件间的相似事件，获取共振组。

### 关联资产模型

#### 描述

#### 输入

#### 输出

#### 类图

#### 流程图

#### 处理过程

#### 质量要求

### 碰面分析模型

#### 描述

分析案件中，各个分析对象之间碰面情况。

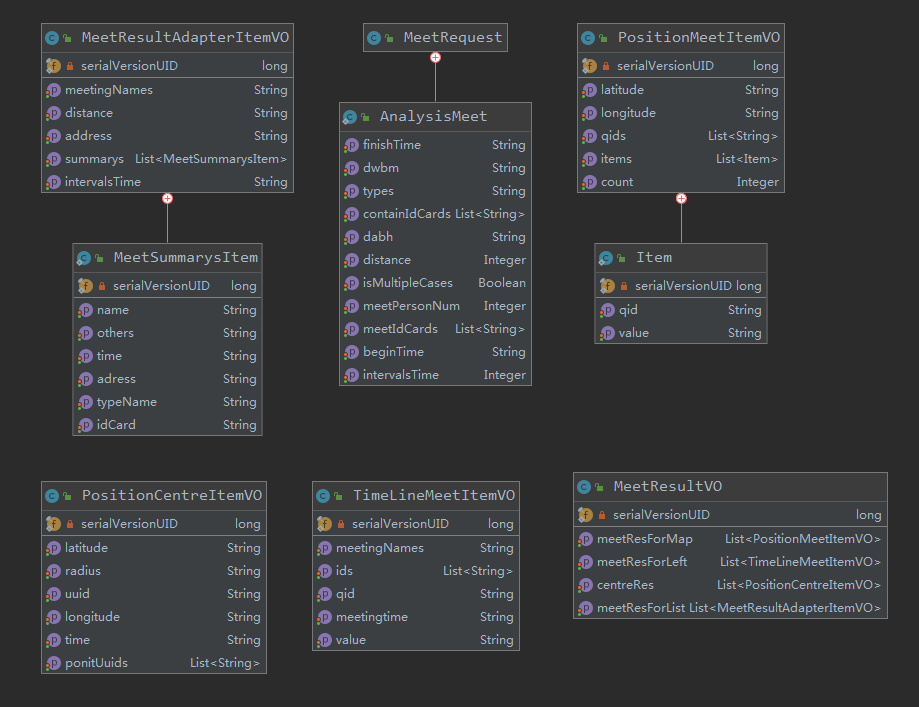
#### 输入

档案编号、分析对象集合、当前用户所属单位编码、时间起止时间、事件维度、间隔时间、间隔距离、是否跨案件、结果人数量、碰面结果集包含人员。

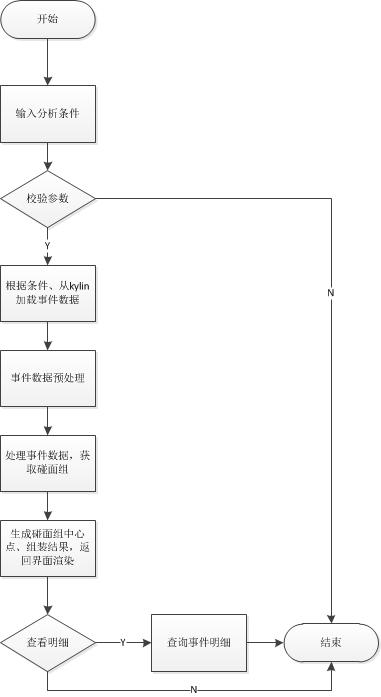
#### 输出

碰面事件位置信息，碰面事件组信息、事件详情。

#### 类图



#### 流程图



#### 处理过程

1. 用户选择分析对象、设定分析条件，
2. 校验参数，
3. 通过用户所属单位编码和是否跨案件参数获取数据范围，
4. 根据数据范围从kylin加载事件数据（单次分析数据量不大于5万条），
5. 对加载的数据进行预处理，提取出关键字段（时间戳、经纬度）用于下一步处理，
6. 判定出碰面组，获取结果集，
7. 在结果集里面过滤掉有重复事件的碰面组，
8. 根据分析条件中的结果人数量、包含人员信息参数过滤碰面组，
9. 为每个碰面组生成中心点（最小覆盖圆的圆心），
10. 返回数据给界面渲染。

#### 质量要求

分析数据必须有时间、经纬度。采用构建八叉树的方式来寻找事件间的相似事件，获取碰面组。

### 节假日腐败模型

#### 描述

在当前案件中对某涉案人员进行分析，查看其在节假日的交易情况。

#### 输入

{

节假日交易明细传参类

amount string 筛选交易金额

dabh\* string 档案编号

endTime string 结束时间 yyyyMMdd

holiday [选择的节假日列表]

idCard\* string 分析人员的身份证号

period string 节假日前后时间

startTime string 开始时间 yyyyMMdd

}

#### 输出

[{

节假日腐败前端展示类

date string 日期

transferOutput string 当日流出金额

transferItems [{

节假日交易明细前端展示类

accountTag string 账号标签，暂为空字段

cxkh string 户主账号

holiday string 节假日

je number 交易金额

jydfkh string 对方账号

jydfxm string 对方姓名

jydfzjhm string 身份证（对方）

jysj string 交易时间

jywdmc string 交易网点

mc string 户主姓名

summary string 摘要，暂为空字段

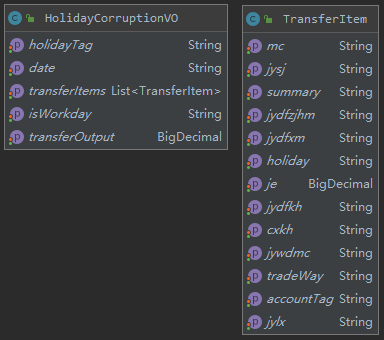
tradeWay string 交易方式，暂为空字段

}]

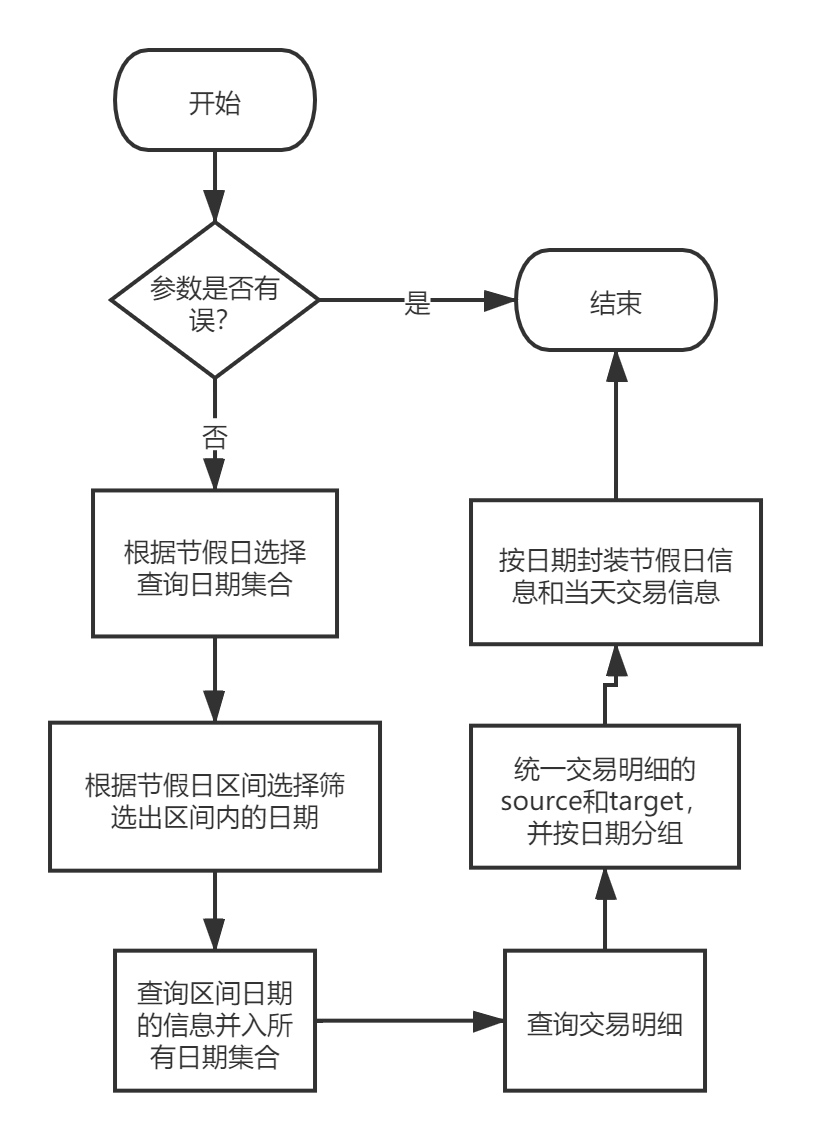
}]

#### 类图





#### 流程图



#### 处理过程

1. 判断参数传递是否完整规范

2. 根据节假日选择区间处理出应该筛选的日期集合

3. 统一交易源和目标位置

4. 以天为单位聚合统计

5. 封装返回

#### 质量要求

要求传递的档案编号字段不为空，选择的对象应当为人，日期区间最早应从2011年1月1日开始。

### 洗钱特征分析模型

#### 描述

通过多账单信息，按照一定的交易规则（流向、资金大小等）筛选 出异常资金交易情况，如某个周期内，一个账号流出到十个及以上账户； 或十个及以上的账户汇聚到一个账号；

#### 输入

|  |  |
| --- | --- |
| **dabh\*** | **String 档案编号** |
| **endTime** | **String 结束时间 yyyyMMdd** |
| **flow\*** | **integer($int32) 资金流向 0流入 1流出** |
| **idCards\*** | **String 身份证集合，多个身份证以逗号分隔** |
| **num** | **integer($int32) *example: 1* 流出重点对象人数** |
| **pageNum** | **integer($int32) *example: 1* 页码** |
| **pageSize** | **integer($int32) *example: 20* 每页条数** |
| **percentage** | **number($double) *example: 10* 分流百分比，只传百分号前面部分** |
| **period\*** | **integer($int32) *example: 7* 洗钱周期** |
| **range\*** | **integer($int32) *example: 50* 交易金额范围** |
| **sortName** | **String 排序字段** |
| **sortRule** | **String 排序规则(ASC,DESC)** |
| **startTime** | **String 开始时间 yyyyMMdd** |

#### 输出

rows [

当前页数据

[LaunderingVO{

description:

洗钱特征分析-展示类

accountType string

example: 建设银行

账号类型

amount number

example: 1000.01

交易金额

card string

example: 622783843837889

卡号

count integer($int32)

example: 1

流出人数

dealTime string

example: 20200511

交易时间

flowCardList [

流出卡号列表

FlowCardVO{

description:

洗钱特征分析流出卡号-展示类

accountType string

example: 建设银行

账号类型

amount number

example: 1234.56

流出金额

card string

example: 622783843837889

卡号

dealTime string

example: 2020-02-11 12:12:12

交易时间

idCard string

example: 511xxxxxxx

身份证号

location string

example: 安徽六安

归属地

percentage string

example: 11.22..%

分流百分比

username string

example: 李四

姓名

}]

idCard string

example: 110101199003077630

身份证号

location string

example: 安徽六安

归属地

percentage string

example: 12.11..%

分流百分比,如果是无限小数则以两个点表示：12.11…%

username string

example: 张三

姓名

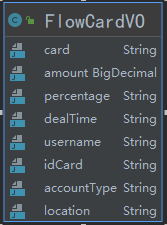
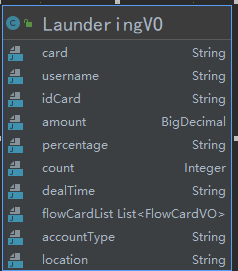
}]]

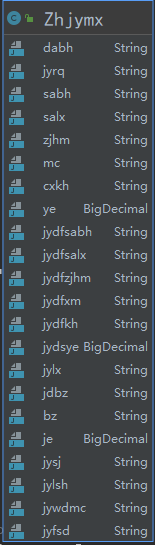
total integer($int64)

example: 123

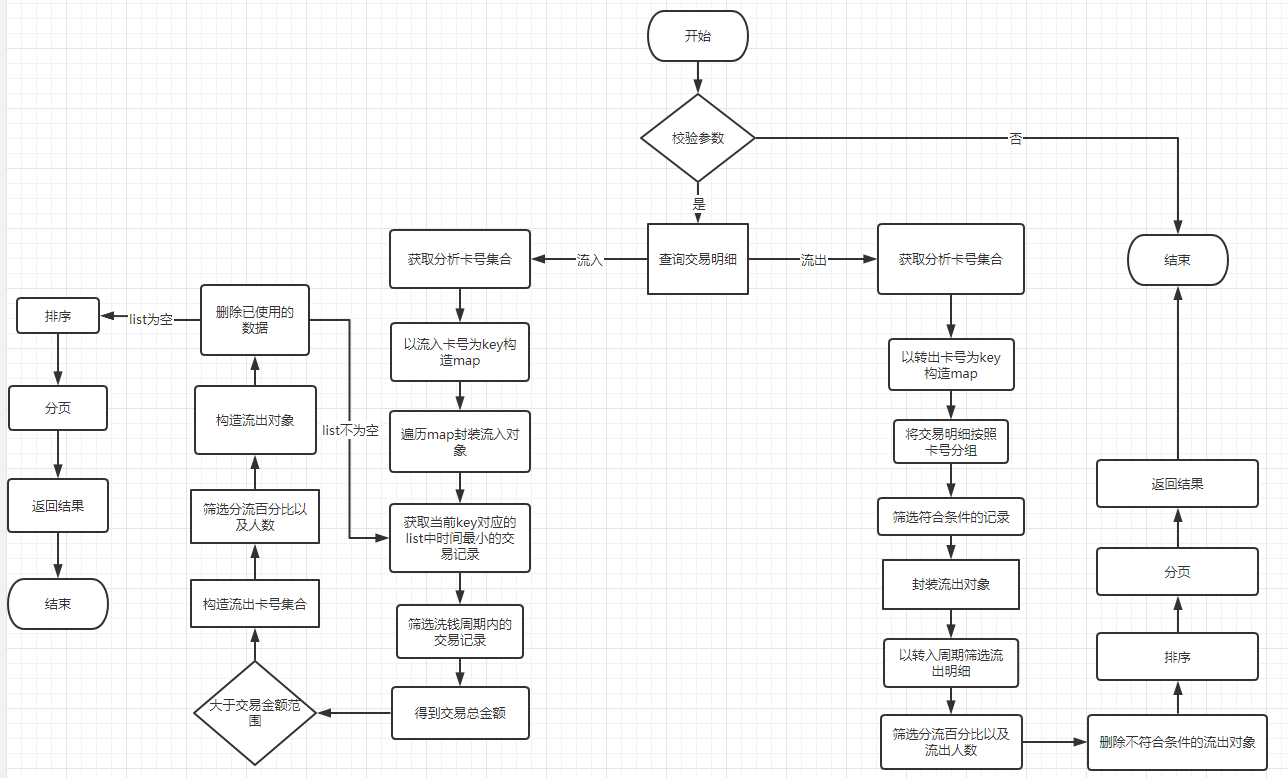
总行数

#### 类图





#### 流程图



#### 处理过程

1. 校验参数
2. 查询交易明细
3. 流入：
4. 获取分析卡号集合
5. 以流入卡号为key构造map，键为交易列表
6. 遍历map封装流入对象
7. 获取当前key对应的list中时间最小的交易记录
8. 筛选洗钱周期内的交易记录
9. 得到交易总金额
10. 大于交易金额范围
11. 构造流出卡号集合
12. 筛选分流百分比以及人数
13. 构造流出对象
14. 删除已使用的数据
15. 排序
16. 分页
17. 返回结果
18. 流出：
19. 获取分析卡号集合
20. 以转出卡号为key构造map
21. 将交易明细按照卡号分组
22. 筛选符合条件的记录
23. 封装流出对象
24. 以转入周期筛选流出明细
25. 筛选分流百分比以及流出人数
26. 删除不符合条件的流出对象
27. 排序
28. 分组
29. 返回结果
30. 结束

#### 质量要求

1. 校验参数保证搜索结果的正确性
2. 在sql中查询时带上时间范围缩小数据量
3. 将交易数据按照key分组以map来处理以降低时间复杂度，提升系统性能
4. 处理集合时边筛选边遍历减少遍历次数，提升性能

## 数据大屏

### 数据大屏

#### 描述

#### 输入

#### 输出

#### 类图

#### 流程图

#### 处理过程

#### 质量要求

## 专项分析

### 房产分析

#### 描述

#### 输入

#### 输出

#### 类图

#### 流程图

#### 处理过程

#### 质量要求

### 工商分析

#### 描述

在当前案件中对某涉案人员进行分析，得到涉及的所有企业与个体信息。

#### 输入

{

单一对象公共查询封装类

dabh\* string 档案编号

dwbm\* string 单位编码

identify\* string 分析对象标识（身份证、企业信用代码）

isMultipleCases\* boolean 是否多案件

}

#### 输出

{

被分析人工商信息展示

relatedOrganizationNum integer($int32) 关联机构数

beImportantNum integer($int32) 曾任高管机构数

beLegalKeeperNum integer($int32) 任法人机构数

canControlNum integer($int32) 实际控制权数

keepSharesNum integer($int32) 占股机构数

personInfo PersonInfo{

被分析人基本信息

age string 年龄

birthday string 生日

idCard string 身份证

joinTime string

name string 姓名

nation string 民族

nativePlace string

politicalLandscape string 政治面貌

position string 职务

sex string 性别

tagList [标签列表]

workUnit string 工作单位

}

beImportantList [{

曾任高管机构列表

address string 生产经营地址

businessScope string 经营范围

contributionAmount string 分析对象认缴出资金额（万元）

contributionRatio string 分析对象认缴出资比例

endDate string 经营期限至

idCard string 证件号码

institutionName string 机构名称

legalKeeper string

liveOn string 住所

nodeId string 统一社会信用代码

operationState string

phoneNum string 联系电话

position string 任职

registerCapital string 注册资本

setUpDate string 成立日期

startDate string 经营期限自

}]

beLegalKeeperList [{

任法人机构列表

address string 生产经营地址

businessScope string 经营范围

contributionAmount string 分析对象认缴出资金额（万元）

contributionRatio string 分析对象认缴出资比例

endDate string 经营期限至

idCard string 证件号码

institutionName string 机构名称

legalKeeper string

liveOn string 住所

nodeId string 统一社会信用代码

operationState string

phoneNum string 联系电话

position string 任职

registerCapital string 注册资本

setUpDate string 成立日期

startDate string 经营期限自

}]

canControlList [{

控制权展示类

contributionRatio string 分析对象认缴出资比例

institutionName string 目标机构名称（控股企业）

nodeId string 目标机构统一社会信用代码

investmentChain [{

控制权展示类

contributionRatio string 分析对象认缴出资比例

institutionName string 目标机构名称（控股企业）

nodeId string 目标机构统一社会信用代码

}]

}]

keepSharesList [{

占股机构列表

address string 生产经营地址

businessScope string 经营范围

contributionAmount string 分析对象认缴出资金额（万元）

contributionRatio string 分析对象认缴出资比例

endDate string 经营期限至

idCard string 证件号码

institutionName string 机构名称

legalKeeper string

liveOn string 住所

nodeId string 统一社会信用代码

operationState string

phoneNum string 联系电话

position string 任职

registerCapital string 注册资本

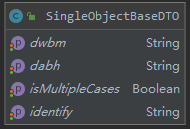
setUpDate string 成立日期

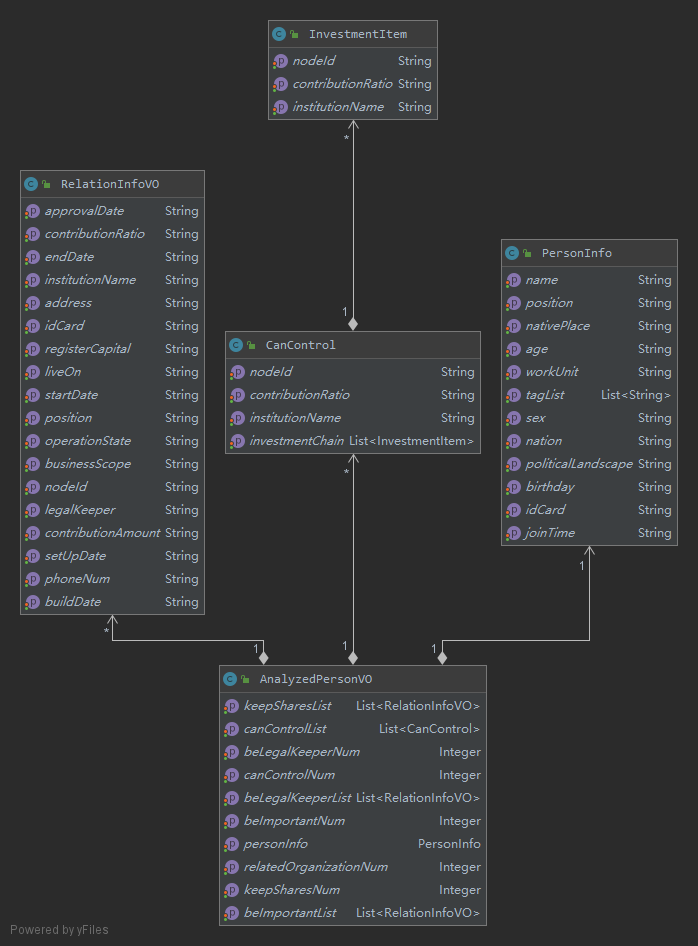
startDate string 经营期限自

}]

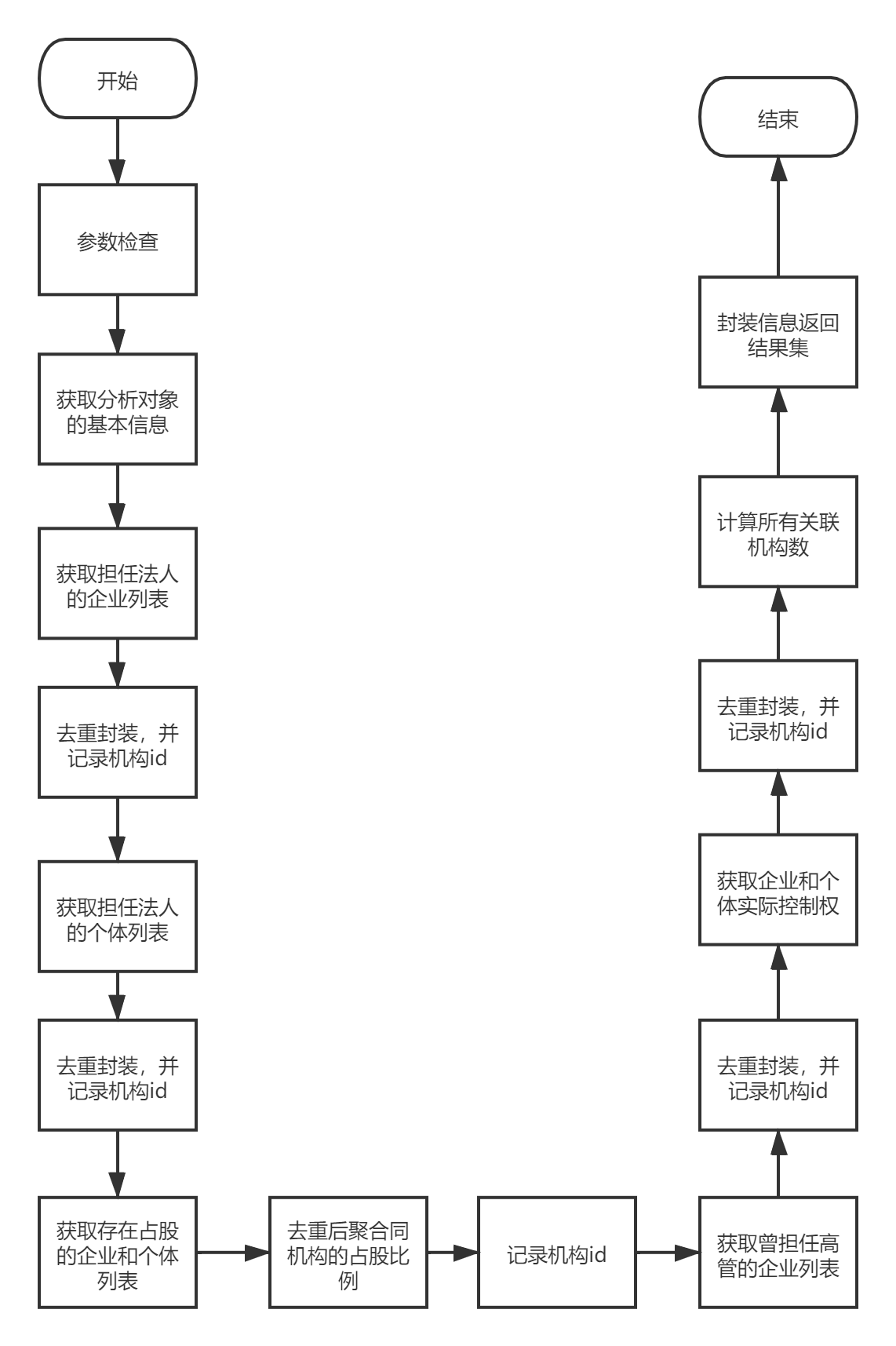
}

#### 类图





#### 流程图



#### 处理过程

1. 判断参数传递是否完整规范
2. 对于每一次获取的数据集都根据公司id进行去重，投资股份相关的则根据记录id去重，因为一家机构只能有一个法人，可以有若干股东。
3. 根据对象身份证号解析对象的年龄、性别，查询saxx表得到姓名信息进行基本信息封装
4. 根据对象身份证号获取在企业和个体中担任法人的机构数据，封装成统一机构信息
5. 根据对象身份证号获取在企业中担任主要人员的企业信息，封装成统一机构信息
6. 根据对象身份证号获取在企业和个体中查询到存在占股的机构信息：
   1. 先进行去重
   2. 按机构分组，将分析对象占股比例加和，超过100%的计为100%
   3. 封装成统一机构信息
7. 计算异常人数封装返回，每一次处理时记录机构的统一社会信用代码，最后根据这个set集合得到关联公司数。

#### 质量要求

针对寻找情人需求设计并实现本功能接口，满足去除户籍亲属关系的关联对象、可疑程度分类、按不同维度判断异常的业务点，统计出不同异常等级的人数，以及异常人员在每个维度上的次数。

要求传递的档案编号字段不为空，选择的对象必须为人，只有人才能拥有情人，所以要求需要传递idCard。

# 性能设计

## 性能瓶颈分析

本系统多个功能需要分析大量复杂数据，面临着数据量大，数据种类复杂，运算过程复杂等问题，传统架构设计面临着巨大挑战，影响系统功能。

## 性能设计方案

### 缓存

* 1. 主动缓存（适用于更新可能小的情况，如一些设置或者配置）

写场景：先将数据写入到数据库，写入成功后立即把数据同步到缓存。

或者写入到数据库后，把之前的缓存失效，下次请求的时候，加入到缓存中

* 1. 直写(异步的方式)

直接把Cache假装当成数据库，读写都针对Cache，然后Cache负责后面的数据同步问题,

以同步的方式 先写入数据库，写入成功后写入redis。

如果写入数据库一直没成功，采用日志记录的方式，单开个任务队列，读取日志，主动写入数据库并写入redis缓存

* 1. 其他想法：(异步的方式)

写入的时候，写入到redis,然后使用任务分发的方式，更新数据库，但是也需要和b一样，做数据库一致保证

### 预处理

对于需要聚合或分析的数据，进行预处理，在系统资源空闲时，对数据进行提前运算，存储到数据库中，保证在调用这些数据时，不在进行运算，缓解系统压力。