

大数据流计算和PG结合 实现资金实时监控

胡怡文

目录

CONTENTS

01

背景

02

DTC模型

03

技术方案

PART 01

背景

在金融领域，资金安全是永恒的话题，如何监控资金去向，防范资损风险是金融企业的核心能力之一。



任何一次的疏忽都有可能导致资损

01

狭义的资损：

由于金融企业的原因，导致直接资金损失，或产生赔付。

02

广义的资损：

任何由于系统故障、缺陷、人为操作、安全漏洞导致金融企业或金融企业的客户蒙受直接或间接损失的事件，都属于资损事件。



消息/调度等系统对同一笔请求重复发生了多次



上游业务系统对同一笔交易下发了多笔



调度系统由于积压，很短时间发送了很多调度请求



内部代码出现bug，我方对同一笔交易向下游发送了多次



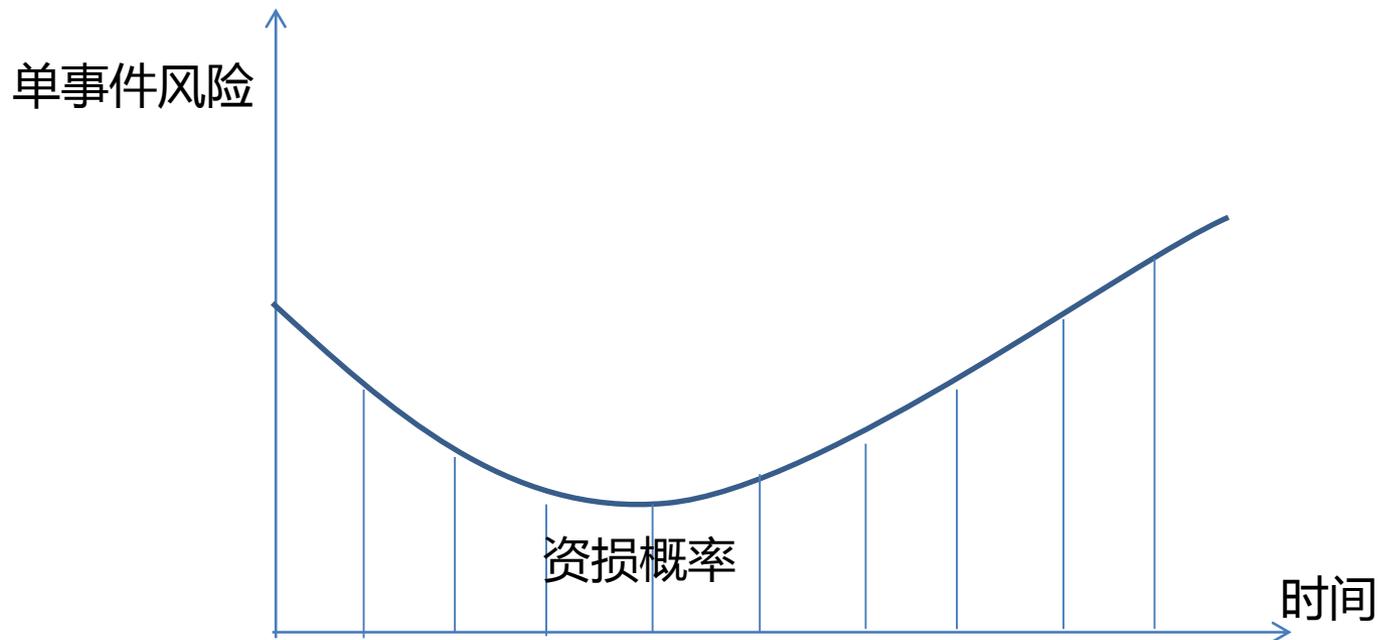
多台机器同时捞取数据库，设计不当捞出了重复的数据



HTTP Client默认设置重发三次，网络异常造成向下游重发



一：从数据的角度来看问题是怎么发生的？



资损是积分的结果，随着时间，风险一直在积聚，问题不知道哪一天就发生了

二：问题在哪里，你是怎么发现它的？



市人民医院血常规报告单 [质评合格 省内参考]

姓名： 病案号： 费别： 标本编号： 31
性别： 申请科室： 门诊抽血室 送检医师： 条码编号： Q300341757
年龄： 床号： 标本种类： 临床诊断：

序号	代码	项目名称	结果	单位	参考值	序号	代码	项目名称	结果	单位	参考值
1	WBC	白细胞	7.33	10 ⁹ /L	4-10	16	MONC	单核细胞	0.67	10 ⁹ /L	0-0.8
2	RBC	红细胞	4.76	10 ¹² /L	3.5-5.5	17	EON	嗜酸性粒细胞	0.11	10 ⁹ /L	0.05-0.5
3	HGB	血红蛋白	151	g/L	110-160	18	BASO	嗜碱性粒细胞	0.01	10 ⁹ /L	0-0.1
4	HCT	红细胞压积	44.1	%	36-50	19	RDW-CV	红细胞分布宽度-CV	11.9	10 ⁹ /L	10.9-15.4
5	MCV	红细胞平均体积	92.6	fL	82-100	20	RDW-SD	红细胞分布宽度-SD	39.3	%	37-54
6	MCH	平均血红蛋白量	31.7	pg	26-32	21	PDW	血小板分布宽度	10.1	fL	9-17
7	MCHC	平均血红蛋白浓度	342	g/L	320-360	22	MPV	平均血小板体积	9.1	fL	9-13
8	PLT	血小板	215	10 ⁹ /L	100-300	23	PCT	血小板压积	0.20	%	0.17-0.35
9	LYMPH	淋巴细胞比率	32.10	%	20-40	24	P-LCR	大型血小板比率	19.1	%	13-43
10	NEUTP	中性粒细胞比率	57.20	%	50-70	25	ESR	血沉	8		男:0-15
11	MONCP	单核细胞比率	9.10	%	3-8						
12	EOP	嗜酸性粒细胞比率	1.50	%	0.5-5						
13	BASCP	嗜碱性粒细胞比率	0.10	%	0-1						
14	LYMPHN	淋巴细胞数	2.35	10 ⁹ /L	0.8-4						
15	NEUT	中性粒细胞数	4.19	10 ⁹ /L							

核收时间：2009-03-21 08:45 报告时间：2009-03-21 09:15:32 检验者： 审核者：
备注： 此结果仅对本样本负责！

问题的发现是通过对数据的分析比对，当它违反某种规则时，就可以认定有事情发生了

三：我们有什么启发

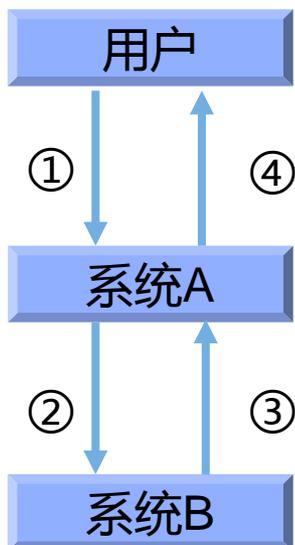
- 1、解铃还须系铃人，问题怎么来的，让它怎么走
- 2、发现问题的三要素
 - ☑ 实时数据采集
 - ☑ 掌握规则
 - ☑ 分析对比

PART 02

DTC模型

DTC(Dataflow-Transform-Check) , 主要包括以下三个环节 :



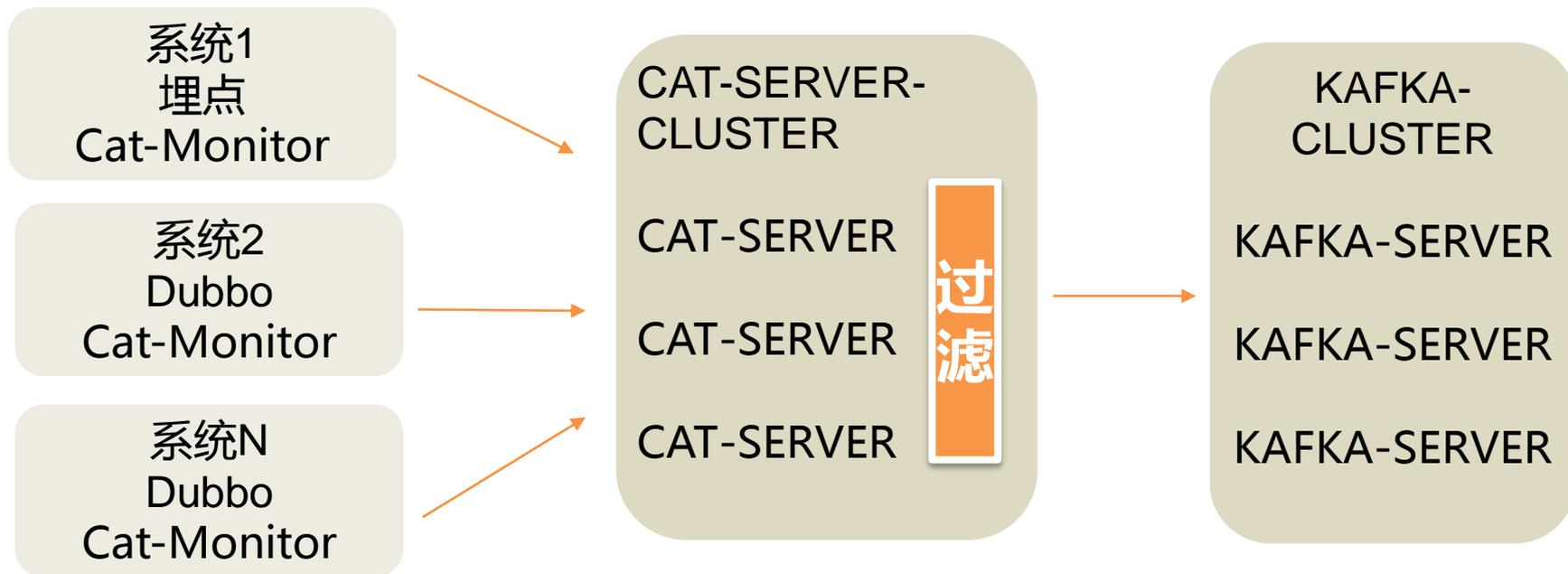


	交易类型	交易单号	账户	金额
①	发起交易	00001	act_001	100
②	转发交易	00001	act_001	100
③	完成交易	00001	act_001	100
④	返回状态	00001	act_001	100

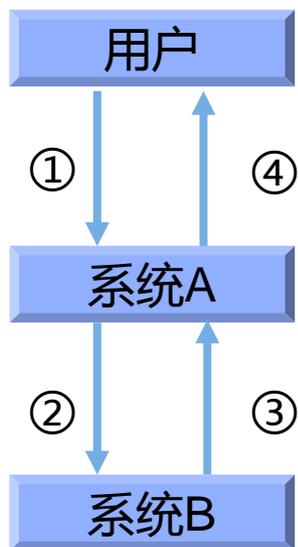


Dataflow

- 系统间调用日志的实时收集旁路系统
- 低负载，增加过滤环节
- 低侵入



在这个场景中，数据是CAT系统收集各系统的接口调用参数，并以日志的形式发送到Kafka集群。



比如产生这四条数据的接口如下表所示：

	接口名
①	com.pinganfu.mpmt.function1
②	com.pinganfu.acct.function2
③	com.pinganfu.acct.function2
④	com.pinganfu.mpmt.function1



Transform

1.清洗：对于特定字段的提取

2.数据标准化：

- 统一元数据
- 可配置的语义转换

```
{ // 数据1
  "transId": "00001", //交易单号
  "acct": "act_001", //账号
  "amt": 100 //金额
  "timestamp":1479385860}
{ // 数据2
  "txn_id": "00002", //交易单号
  "account": "act_002", //账号
  "amount": 100 //金额
  "timestamp":1479386023 }
```

trans_id	trans_acct	trans_amt
00001	act_001	100
00002	act_002	100

	source_field	target_field
1	transId	trans_id
1	acct	trans_acct
1	amt	trans_amt
2	txn_id	trans_id
2	account	trans_acct
2	amount	trans_amt





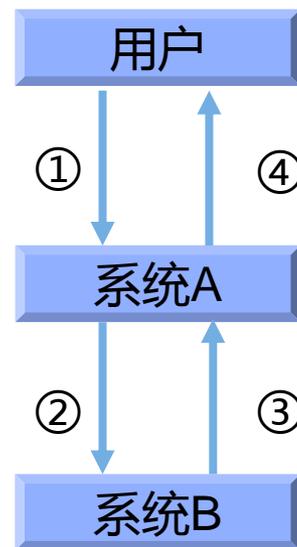
Check

- 基于业务规则的数据校验
- 可配置的规则库
- 灵活的算子

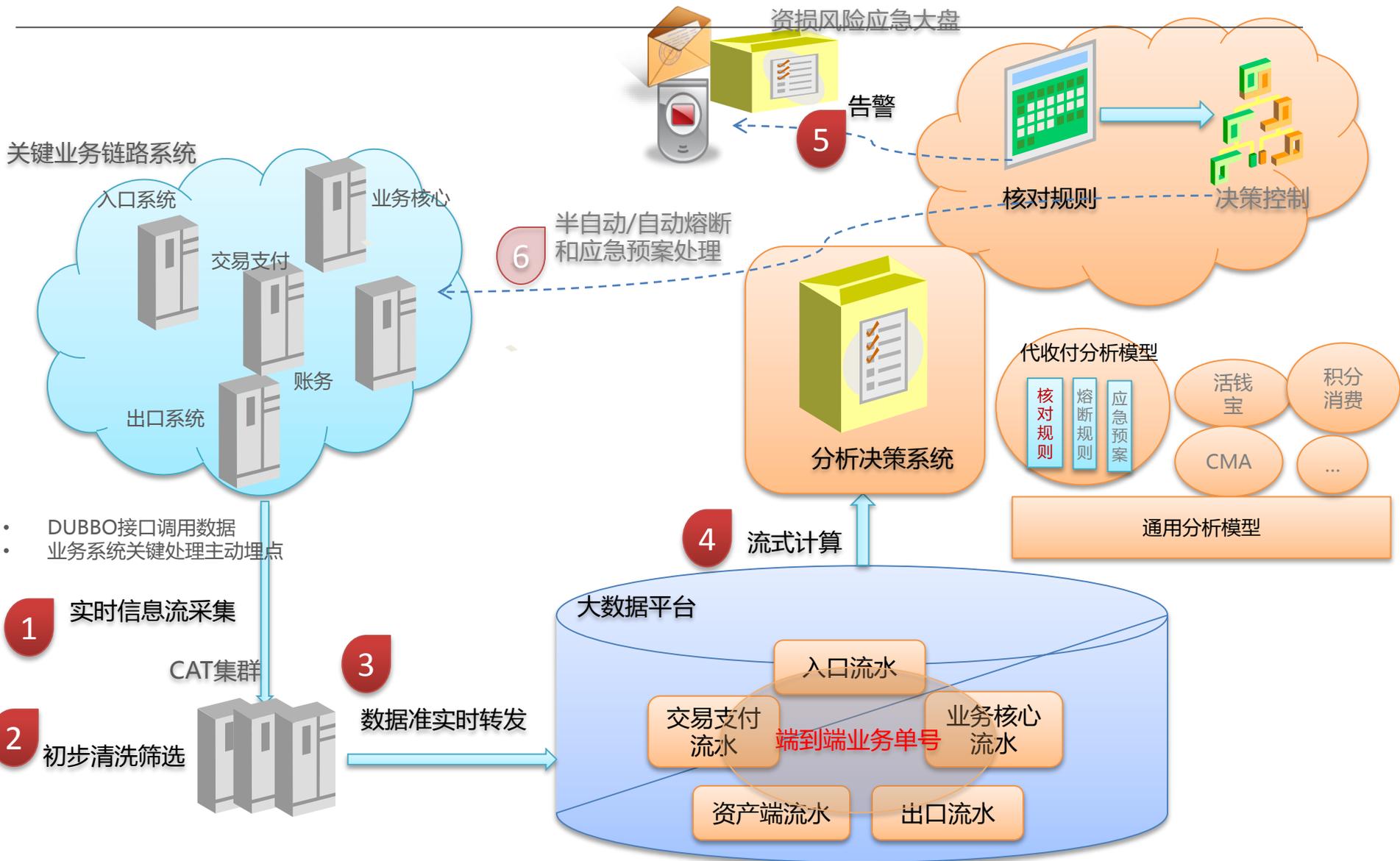


该场景核对规则的文字描述如下：

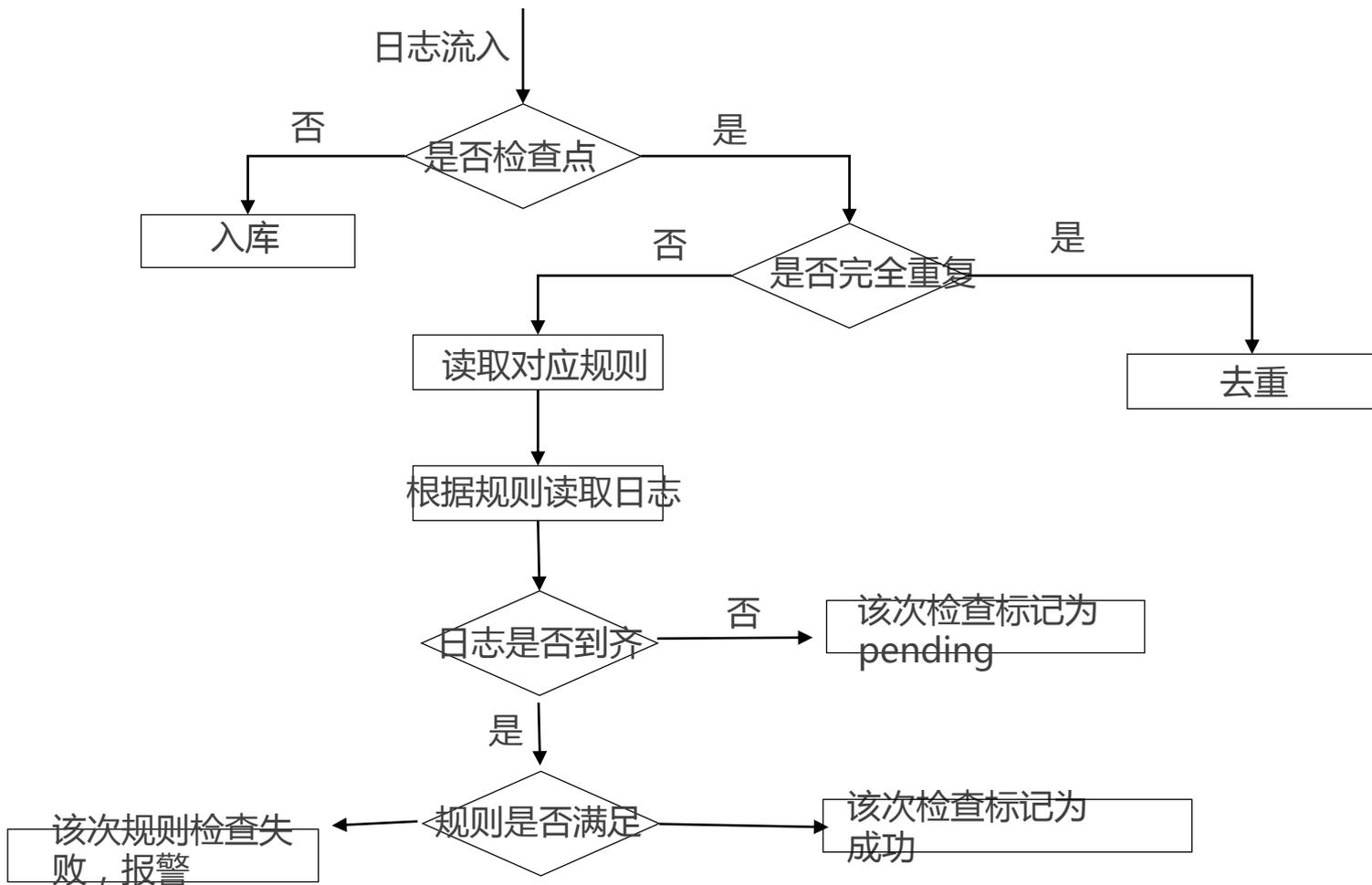
- 1、获取到数据①和 ②时，对数据进行清洗转换后直接存入数据库。
- 2、获取到返回数据③时，根据交易单号查询出数据①和 ②，比较数据③与数据①和②的交易账号，交易金额及状态是否相等。
- 3、获取到返回数据④时，根据交易单号查询出数据③，比较数据④和③的交易账号，交易金额和状态是否相等。

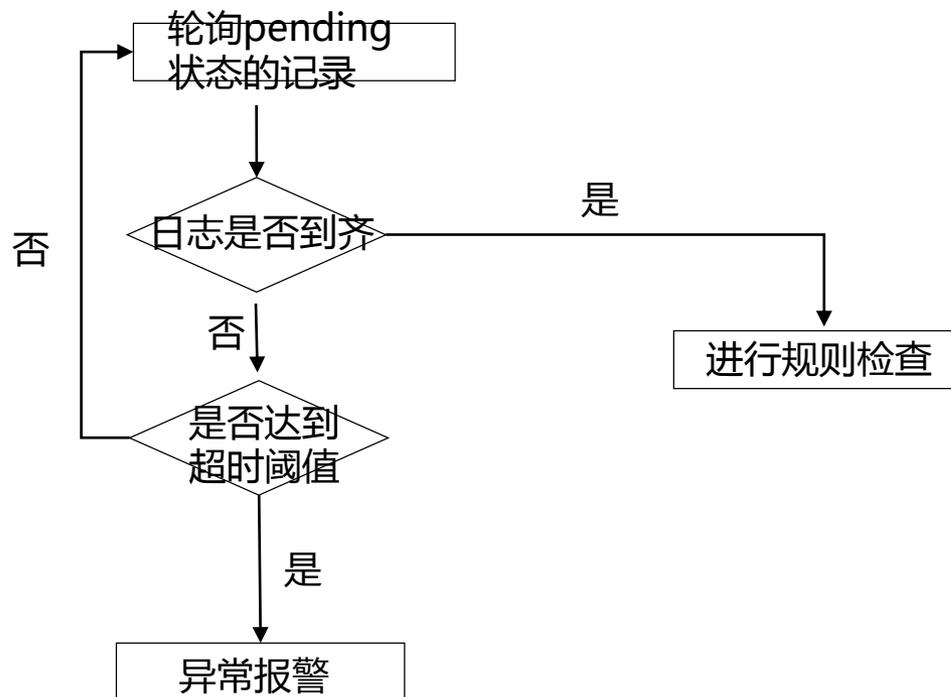


业务思考



规则引擎工作流程

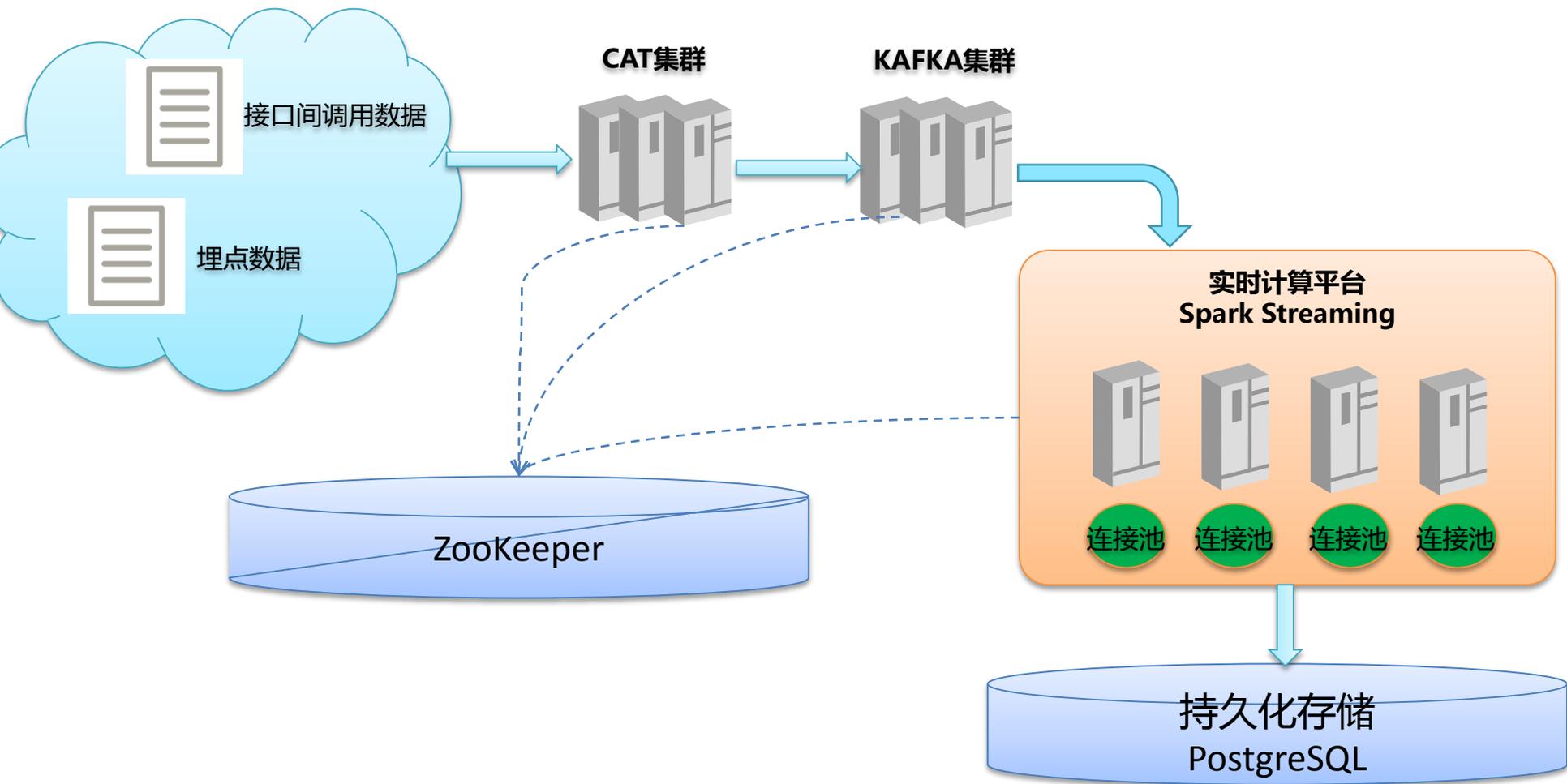


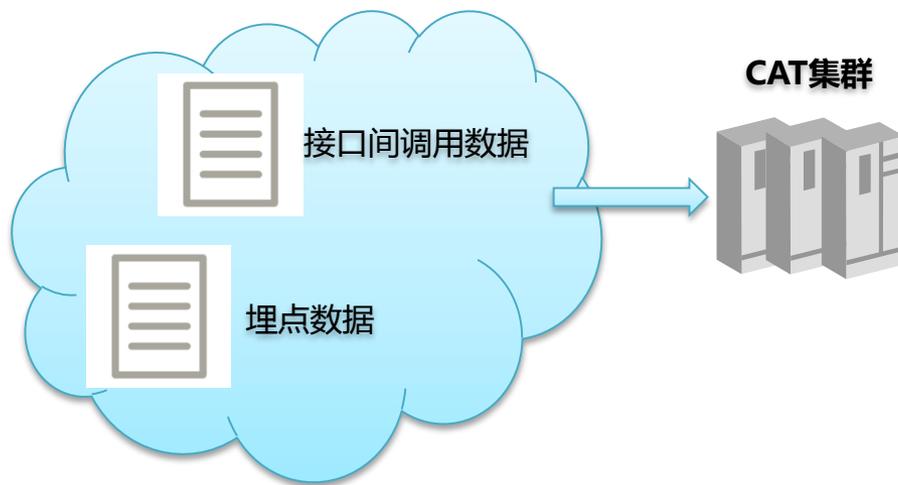


PART 03

技术方案

整体架构





CAT是一个实时的监控系统，通过收集应用系统间互相调用的接口数据以及各系统的埋点数据，进行实时的故障监控、业务统计等。本项目中，在收到上述数据的同时，转发给KAFKA

优势

应用侵入低
接入成本低
数据接入统计&报表
发送延迟低

性能

日均：5千万
TPS：3000
数据发送延迟：小于1s



为了缓解数据生产与消费速率的不一致，采用了KAFKA来缓存数据。同时，KAFKA自身的高可用、水平扩展等特点，保证了该架构能够应对流量高峰。

优势

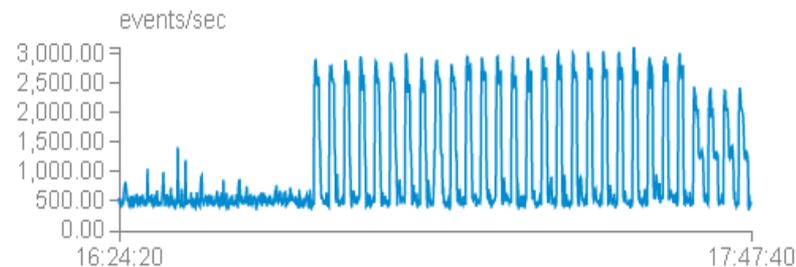
- 水平扩展
- 高可用
- 解耦生产者与消费者
- 顺序保证

1. 采用Spark Streaming对数据进行实时的清洗、转换、计算
2. 基于主集群部署，可以实现快速的资源调整

- 50 executors,400G内存
- 3000 TPS

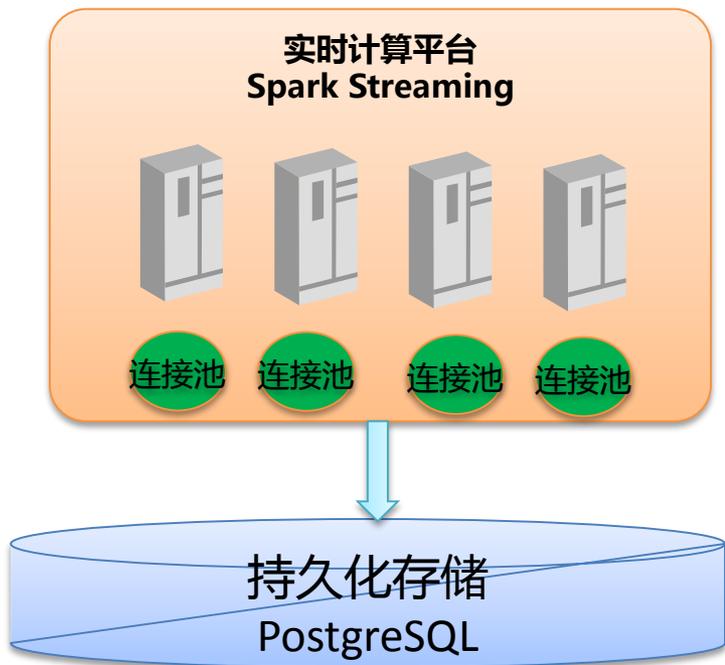
▼ Input Rate

Avg: 1058.55 events/sec



- 作业平均执行时间小于2秒

2016/12/20 19:08:00	0.9 s	2/2	116/116
2016/12/20 19:07:55	0.6 s	2/2	116/116
2016/12/20 19:07:50	0.7 s	2/2	116/116
2016/12/20 19:07:45	0.7 s	2/2	116/116
2016/12/20 19:07:40	1 s	2/2	116/116



优势

- 兼容Oracle语法
- 功能丰富
- 稳定性高
- 并发写入量大
- 复制延迟较低

性能

- TPS : 950
- QPS : 3200

THANKS