

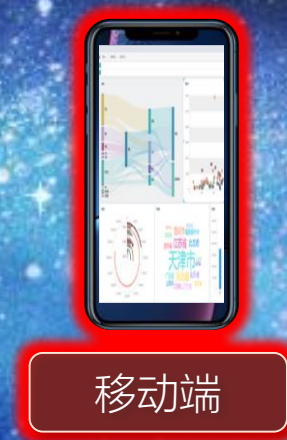
高性能在线计算方案

集算器SPL Base应用场景



加作者了解更多

在线计算应用场景



在线计算面临困境



前端应用-性能要求高!

几百个用户访问, 期望秒级响应
期望查询全量数据

数据库、数仓-性能不可控!

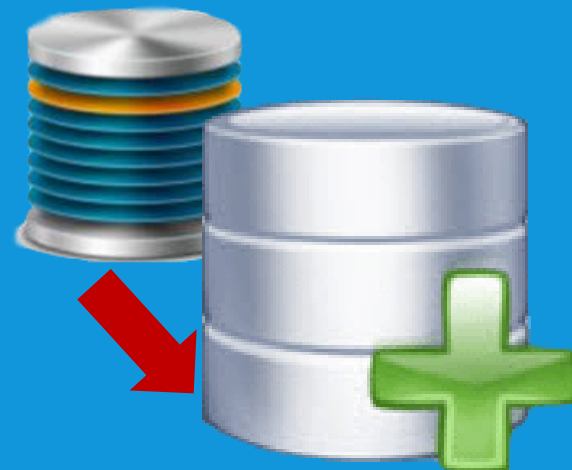
承担过多应用负载, 性能不稳定、受其他应用影响很大

现有方案问题：数据库、数仓扩容或更换



扩容成本高

数据库、数仓扩容成本高昂
数据仓库节点数有上限
继续增加节点不能有效提速



换掉不可行

换其他数据库、数仓，牵扯多部门
多个其他应用，成本太高
真的换了，也不能保证更快

现有方案问题：增加前置数据库



重复建设

前端期望查询全量数据，
前置数据库和数仓规模相同也要集群

速度不够快

行存数据库无法
实现千万数据量秒级响应

无路由能力

无法实现高频热点数据前置，
大量冷数据后置的路由功能

现有方案问题：增加内存数据库



价格过于昂贵

内存数据库，
购买价格几百万、上千万



服务费用超高

按年缴纳超高服务费，
需要原厂维护，一次几万元



容量有限

容量由单机内存决定
无法横向扩展

解决方案：用集算器实现高性能在线计算后台



支持大并发

集算器可并行集群
实现多用户大并发访问

列存速度快

内置列存数据文件
实现千万数据量秒级响应

全内存模式

支持全内存模式运行，
可集群、可并行、高可用

成功案例：海量账户大并发实时查询项目

大并发需求

银行账户明细查询，单台服务器要求支撑60并发。

数据规模

2015年9月到2018年9月明细数据超过3亿条，且要关联多个维表

集算器响应时间 **0.5** 秒

几行代码即完成
查询和关联计算

		B	C
		=A1.import@t0	加载代码表
		=A2.import@b0	
		=A3.create()	创建事实表游标
		=A4.fetch()	过滤事实表并取数
4	,SATXN_LL,FK_SA_ACN_KEY,SA_DDP_ACCT_NO_DEP_SA_CR_AMT,SA_TX_AMT,SA_OPUN_COD,SA_DSCR_P_COD;\${where}}		
5	=B4.switch(SA_OPUN_COD,B1:CM_OPUN_COD;SA_DSCR_P_COD,B2:CM_DSCR_P_COD)	=A5.new(CORPORATION, DAY_ID,SATXN_LL,FK_SA_ACN_KEY,SA_DDP_ACCT	事实表结果关联码表
6	return B5		返回结果



查询时间不超过**5秒!**

比肩专业列存数据仓库

每台服务器支持**50并发**

实际支持几百用户访问无压力

数据量超过**3千万条**

按照条件过滤查找结果

自助分析工具全兼容

JDBC配置改为集算器即可

成功案例：银行多用户大数据量自助分析项目



成功案例：银行回单查询

优化需求

回单报表查询一个分行、一个月的数据，响应时间18分钟，常造成内存溢出

数据规模

千万级

集算器提速 **108** 倍

场景	优化前	优化后	提升倍数
回单查询	18分钟	10秒	108倍

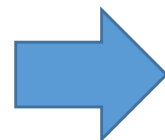
1000



分段并行 集群方案

集算器针对复杂SQL的优化

```
Select F1, F2, F3,  
  (select FF1 from TABLE1 WHERE...) AS F4,  
  (select min(FF1) from TABLE2 GROUP BY...) AS F5,  
From  
  (select FFF1 from TABLE3 WHERE...) T1  
Left join  
  (select FFFF1 from TABLE4 WHERE...) T2  
On T1.FFF2=T2.FFFF2  
WHERE  
T1.FFF2 in (select min(FFFF1) from TABLE5 GROUP  
BY...)
```



将子查询转成JOIN计算

剥开多层嵌套SQL转成单层计算

。 。 。



在线计算前端SQL
多层嵌套并包含子查询



OLAP引擎
只能优化简单SQL



集算器对这类SQL
做了深度优化

集算器提供创新的预汇总能力

扫码阅读



《多维分析预汇总的方案探讨》



部分预汇总，有效权衡
空间与时间的矛盾



对时间段统计给出
切实可行的预汇总手段

集算器内存计算



为什么SPL能够实现高性能内存计算？

离散数据集模型

SPL



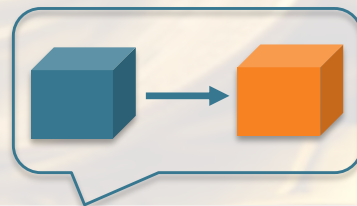
SPL高性能的原因

- ✓ 良好的**复用、引用**机制，减少数据复制；同样的内存，SPL能做**更多**的计算任务！
- ✓ 充分利用内存特性，外键可用指针实现**预关联**，JOIN可在常数时间内完成！
- ✓ 更多高性能优化算法：有序分组、归并关联、...

集算器内存计算：SQL与SPL比较



从合同表中筛选出1998年的销售记录

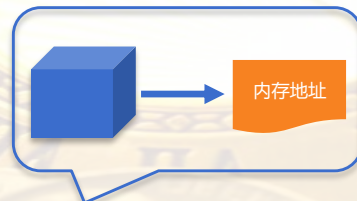


1 SELECT * FROM 合同表 WHERE YEAR(销售日期)=1998

2

SQL

查询过滤的结果总是新产生,这就会把数据再复制一遍,不仅消耗时间,而且占用更多内存!



A

1 =合同表.select(year(销售日期)==1998)

2

SPL

A.select()的计算结果,还是原成员构成的新集合,数据并没有复制,仅复制内存记录地址,速度更快且占用内存更小!

集算器内存计算：预关联加速JOIN

客户	
客户ID	主键
客户名称	
地址	
...	

订单	
订单ID	主键
客户ID	
订购日期	
...	



 数据库JOIN：
查询时临时计算关联

客户	
客户ID	主键
客户名称	
地址	
...	

指针

订单	
订单ID	主键
客户ID	
订购日期	
...	



 集算器JOIN

预先计算好关联
以各种方式存放在内存中
查询时无需计算关联

集算器混合计算：实现T+0实时计算

目前T+0在线计算常用方式与存在问题

① 历史和当期数据同库存储

大量的历史数据会导致高昂的数据库成本
(存储成本和性能成本)

② 历史和当期数据分库存储

需要数据库具有跨库运算能力，但实施复杂度较高，性能较低；当数据库类型不同时难以实现

- 集算器可以基于多个**异构数据库**完成报表T+0查询；
- 还可以将历史数据存放到IO性能更佳的**文件系统**中采用集群运算获得更高性能和更低成本

集算器可编程路由

数据路由

频繁访问的热数据与大量冷数据分路路由，路由规则可编程

SQL解析

强大的SQL解析函数，轻松拆分SQL各子句，用于执行路由规则

本地高性能计算

热数据全内存、温数据本地文件存储。集算器高性能计算，快速响应大并发访问请求

SQL转换

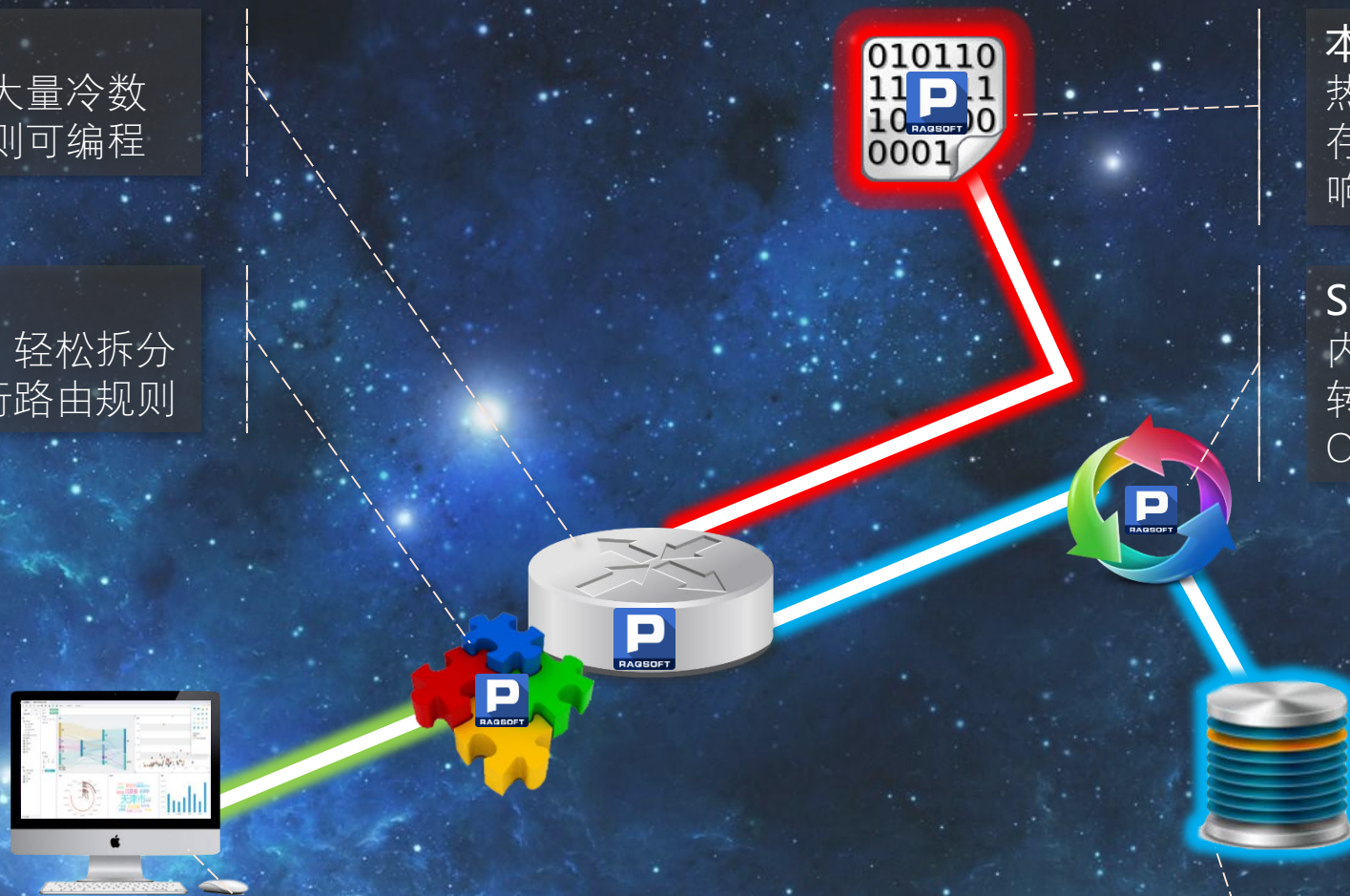
内置SQL转换函数，将标准SQL转换为各种数据库SQL，GP、TD、ORACLE等全兼容

前台展现

多维分析、OLAP、报表、查询、大屏、移动端、管理驾驶舱

后台数据库、数据仓库

大量冷数据后置，无须前置库重复存储全量数据，结构改变很小



集算器数据温度分层策略



热数据：全内存

高频访问的热数据全内存存储。支持备胎式内存集群、可横向扩展，单机内存容量要求底



温数据：本地文件

中频度访问的温数据采用本地二进制列存文件存储，容量大，性能高



冷数据：数据库、数仓

大量冷数据后置，前置库无须重复存储全量数据，结构改变很小

- 结束 -

The End



润乾官微
更多精彩



本文作者
加微信吧

THANK YOU