集算器

高性能计算专家

专用高性能数据库（HPDB）

润乾软件出品

集算器是什么?

专用高性能数据库

跑批计算库

在线查询库

多维分析库

内存高速库

集算器解决什么？

跑不完

半夜跑批跑不完，出错了来不及再来
月末年头更是担惊受怕...

查询慢

看个报表等10分钟，业务人员拍桌子...
人多了、时间跨度大了，就查不了了

反应钝

关联统计运算慢，界面拖拽迟钝；预汇总方案占用空间太大且功能盲区多

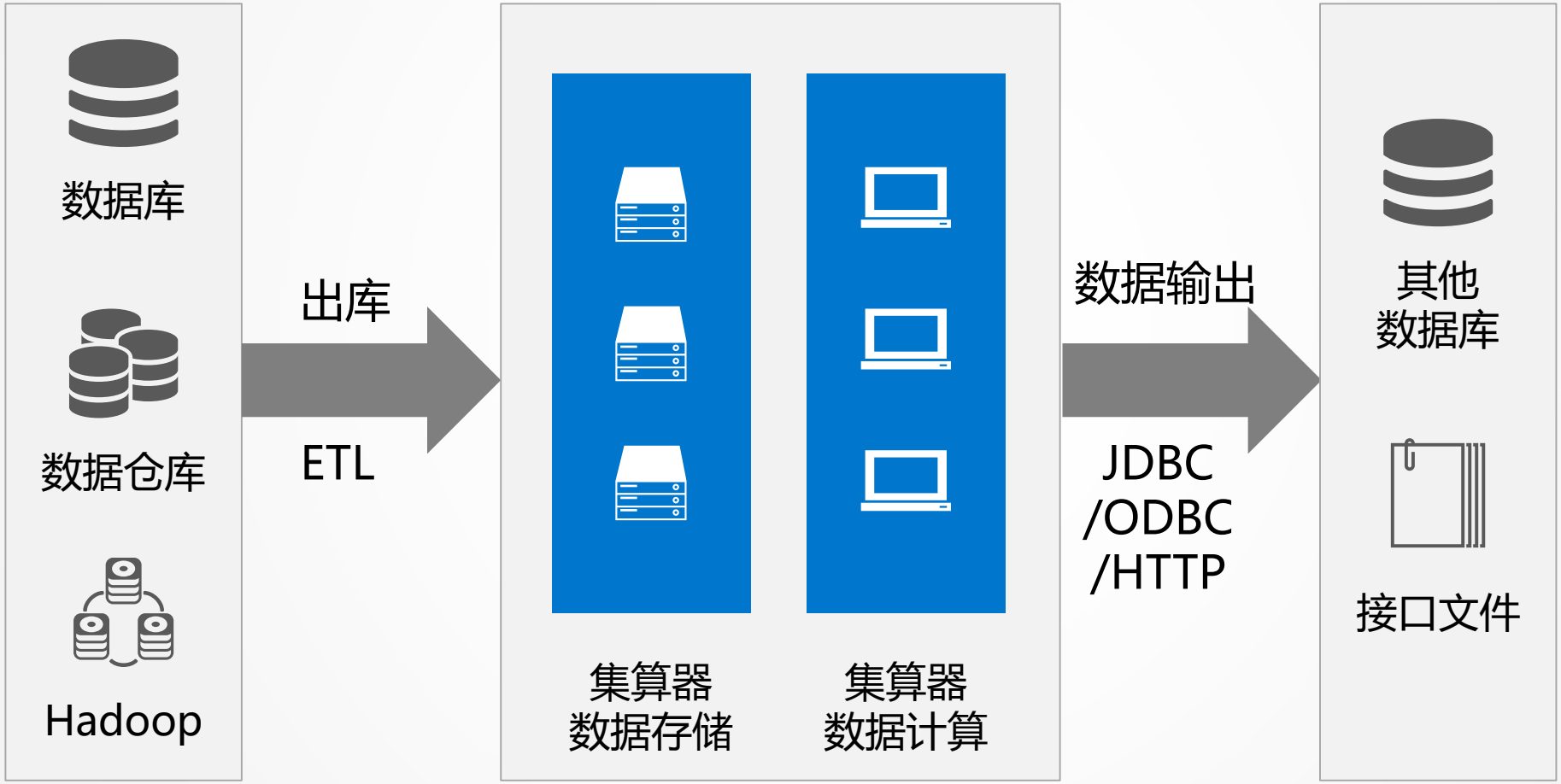
代价高

花了很多钱上了内存数据库，结果性能还是不理想...

无问单机与集群、无问国际大牌与国产新秀、无问MPP与HADOOP，同等硬件条件下集算器平均提速10倍起！

跑批计算库

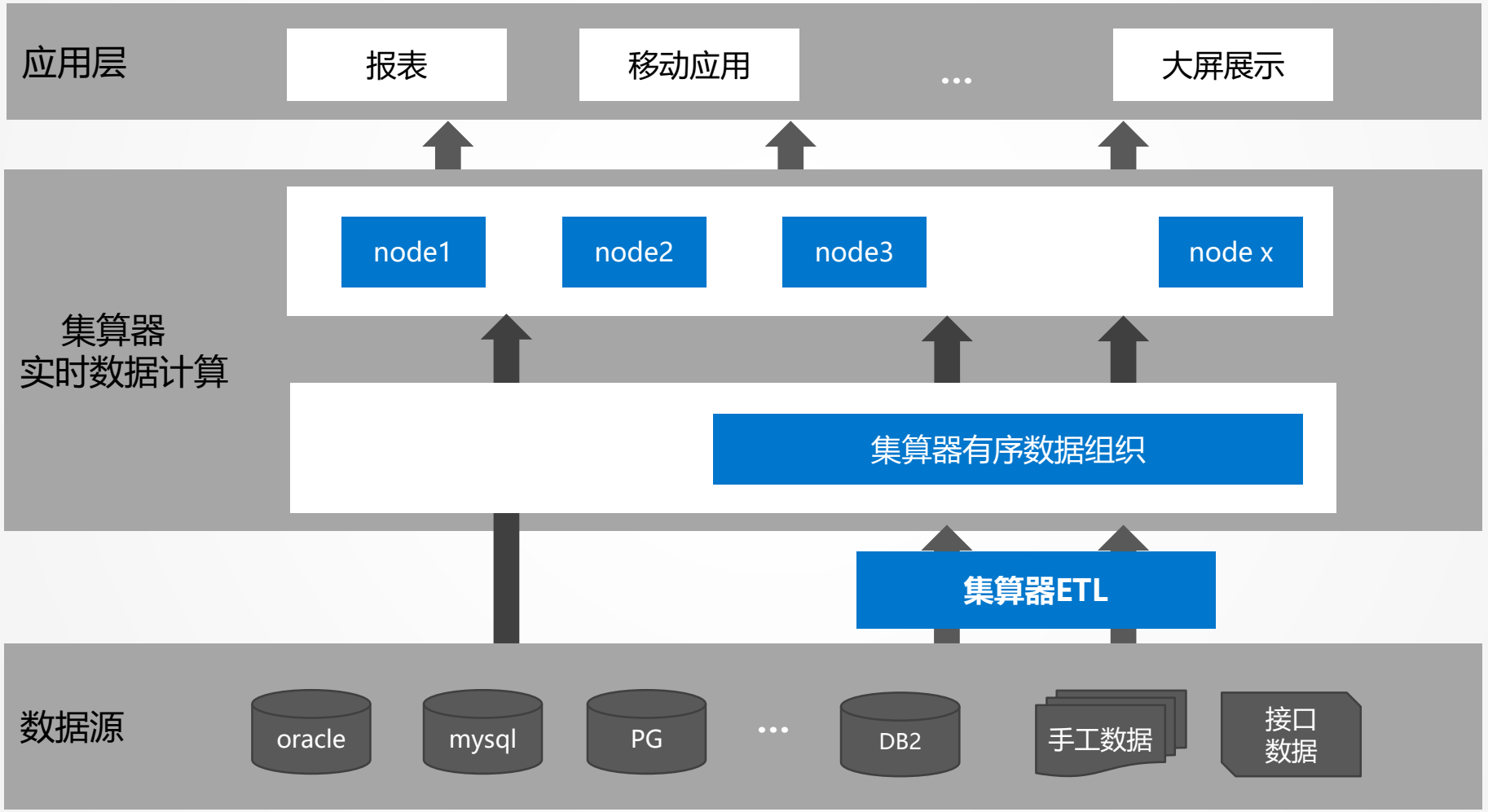
【场景特征】无并发，不必实时，数据量巨大，对时间窗口要求高



*详细内容：http://c.raqsoft.com.cn/article/1546954918511

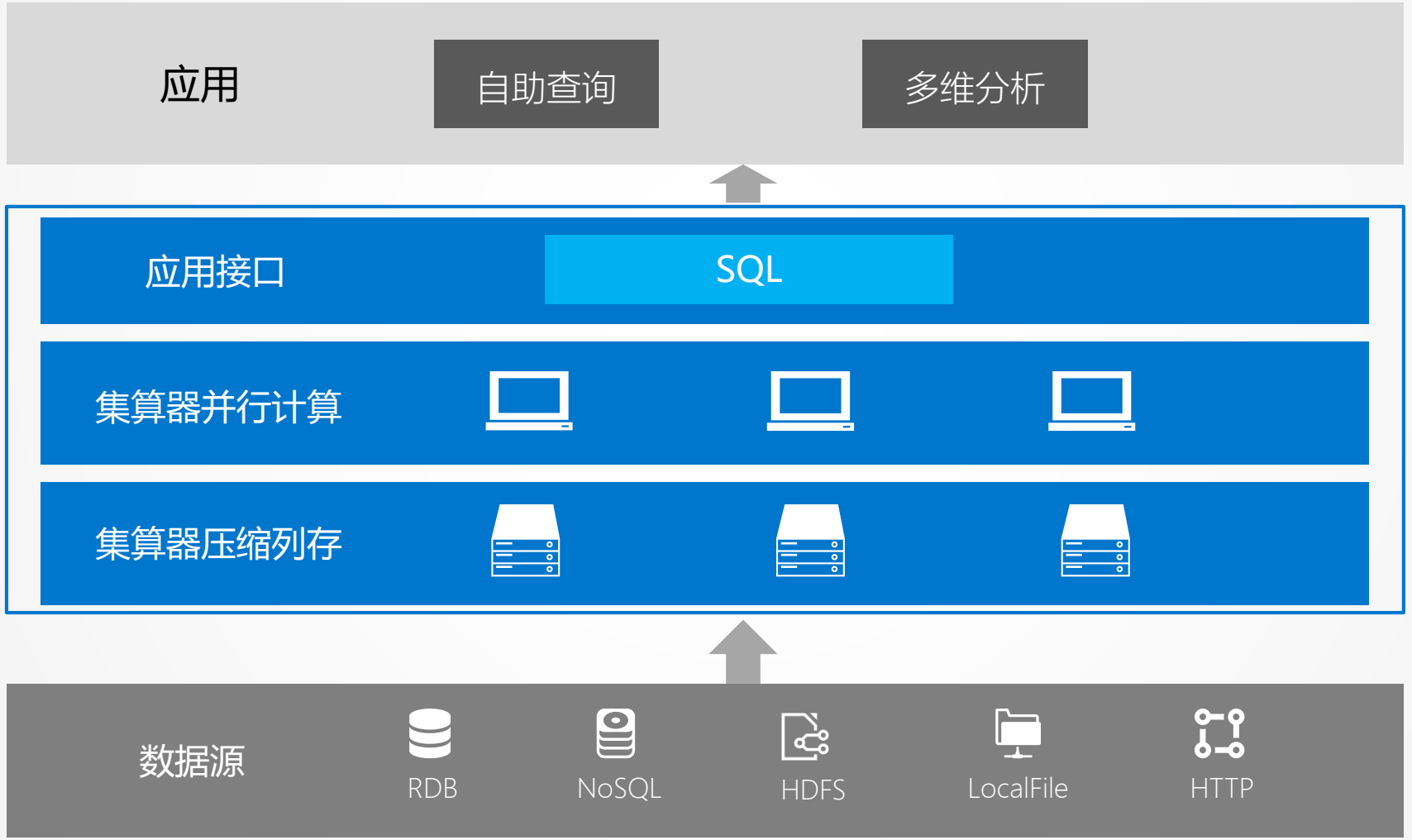
在线查询库

【场景特征】多并发，业务可能复杂，秒级响应，大数据需集群支持



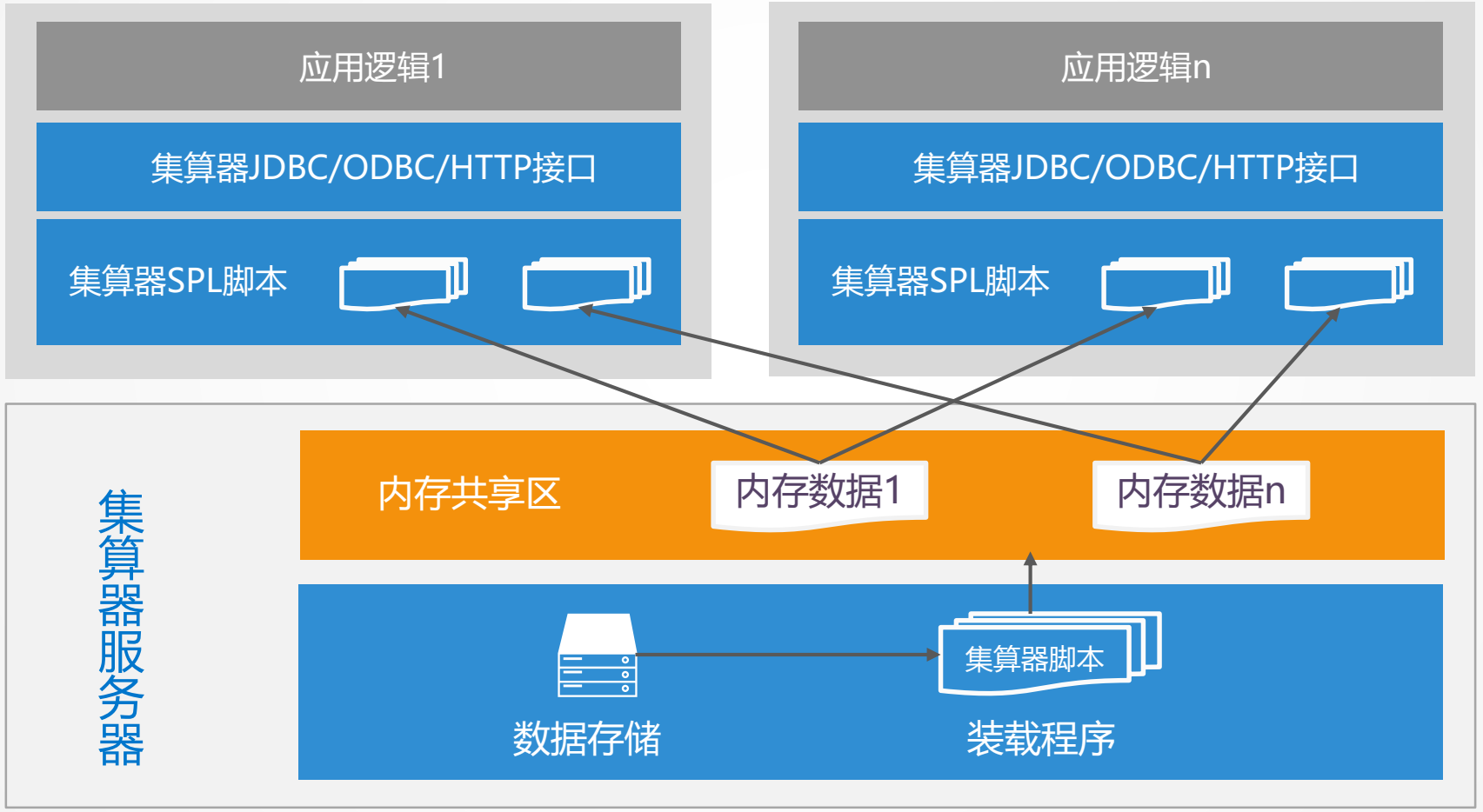
多维分析库

【场景特征】多并发，实时响应，计算相对规整，支持通用BI工具



内存高速库

【场景特征】对性能要求非常高，关联运算复杂



*详细内容：http://c.raqsoft.com.cn/article/1551756139734

性能优化案例1

保险行业历史保单关联业务跑批性能优化

规则复杂

什么才算是同一辆车？车架号相同、车辆vin码相同、车牌号和种类形同。是否贷款车、商业险和交强险等等

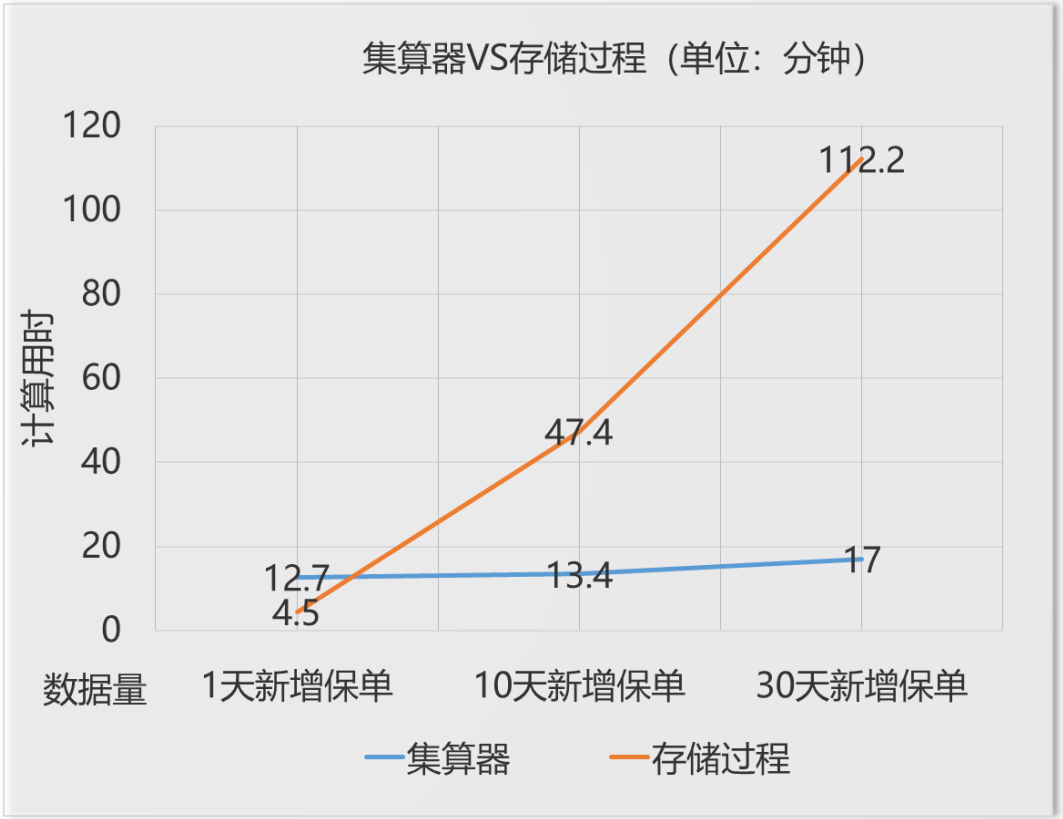
数据量大

一个省的数据几亿条

跑批时间长

30天新增保单2个小时，90天新增保单长时间跑不出结果

面临现状



优化效果

*案例详情：http://c.raqsoft.com.cn/article/1548165744762

性能优化案例2

海量账户大并发实时查询解决方案

数据量大，超过3亿条

2015年09月到2018年09月

并发高访问人数多

几十万、上百万人访问

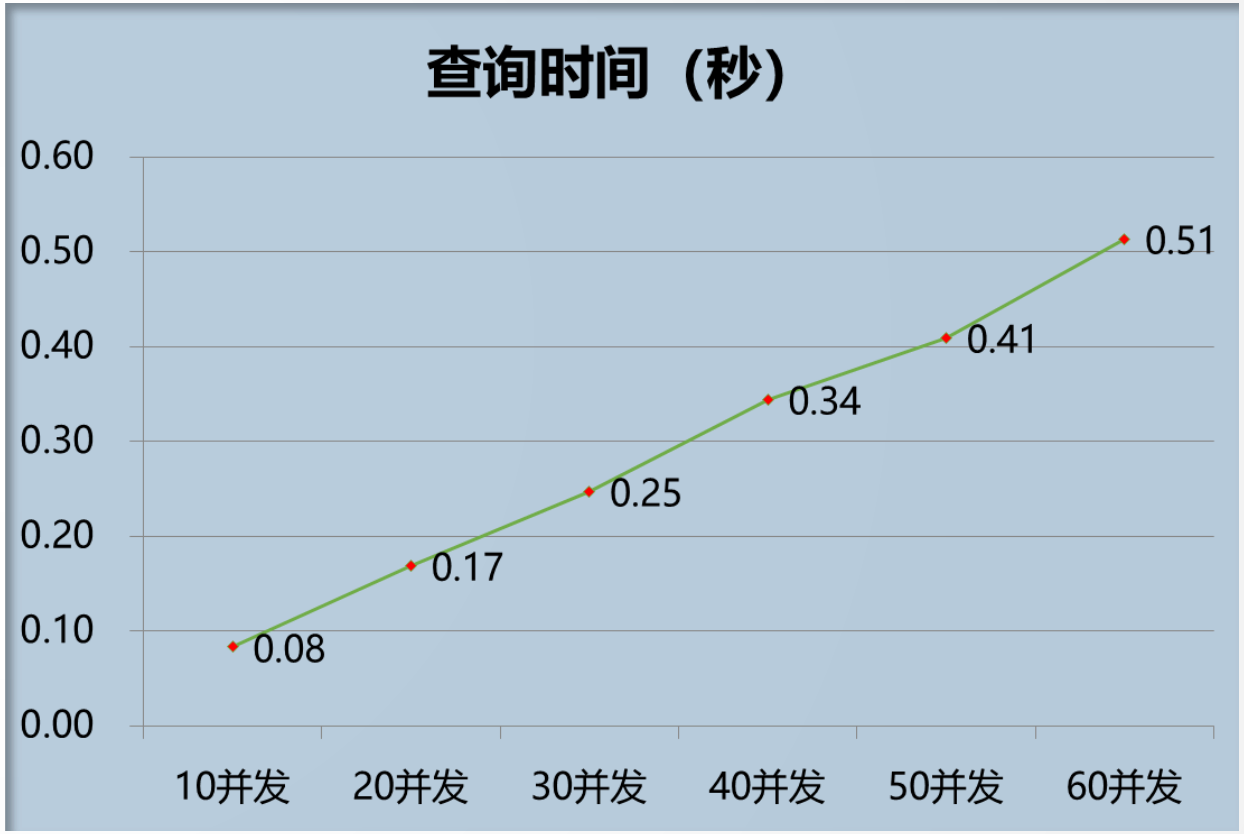
查询时间不能超过1秒

使用6台ES服务器基本达到响应要求

但无法再实现代码表关联

代码表改变时要花数小时重新生成数据

面临现状



集算器：仅用单服务器达到同样性能，同时实现代码关联

*案例详情：http://c.raqsoft.com.cn/article/1547693738179

性能优化案例3

银行业多用户大数据量自助分析提速方案

自助分析系统-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

数据仓库-性能不可控

承担过多应用负载，只能为自助分析系统开放5个连接，性能受其他应用影响很大

面临现状

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

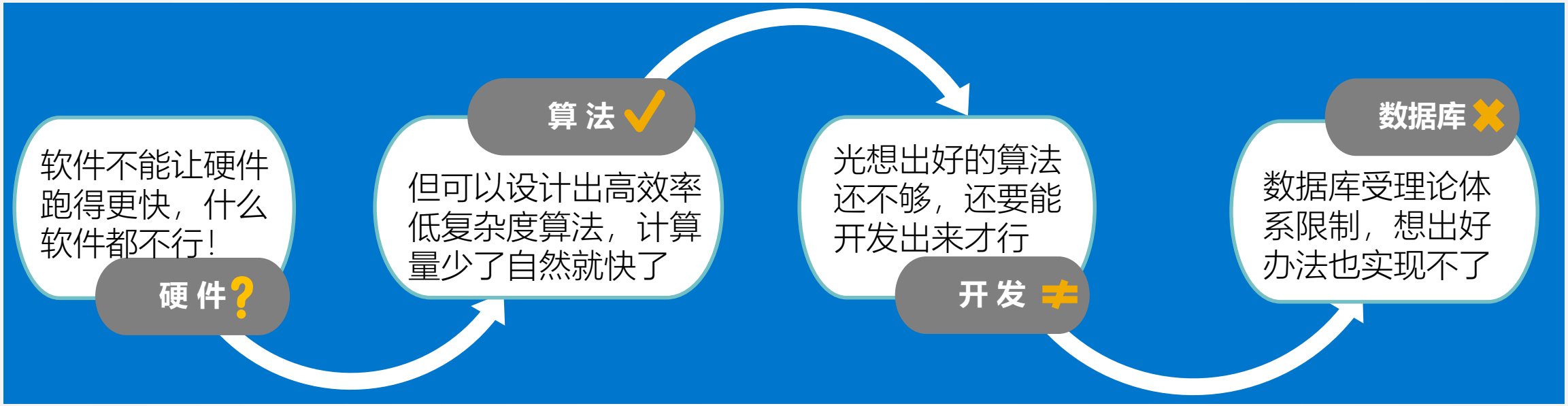
自助分析工具全兼容

JDBC配置改为集算器即可

集算器应用效果

*案例详情：http://c.raqsoft.com.cn/article/1548412219379

集算器高性能计算理念



那咋办呢？

往后看！

哦，原来是这样

对咯，说破了不神奇

那找程序员去做呗

没有这么容易滴

那不是只能干瞪眼吗？

嘿嘿，大多数情况就是这样滴

因此 **高性能计算** = **算法设计** + **算法实现** → **成为制约高性能计算的瓶颈**

SQL，NoSQL，NewSQL，Hadoop，都会限制算法实现

集算器高性能来自于创新计算体系

【类比】计算1+2+3+...+100=?

普通人这么算

1+2=3
3+3=6
6+4=10
10+5=15
15+6=21
21+7=28
...

高斯这么算

1+100=101
2+99=101
...

一共有50个101
50*101=5050

高斯很聪明，想到了高效的算法，但更关键的是：那时人们已经发明了**乘法！**

思考题：1亿行数据取前10名在SQL下会怎么做?

SQL理论上就是大排序后取前10个，效率很低，程序员都知道有不必大排序的办法实现这个运算，却无法用SQL表达，只能用指望数据库引擎自动优化，但复杂情况时数据库并不会优化!
数据库的SQL就象只有加法的算术体系，而集算器的SPL则发明了乘法!
集算器还有更多乘法（高性能计算和存储库），人人都能成为高斯（快速实现高性能算法）。

集算器提供的部分高性能计算机制

遍历技术	高效关联	高性能存储	分布式计算
延迟游标	外键指针化	有序压缩存储	抢先式负载均衡
聚合理解	外键序号化	自由列式存储	Fork-reduce
有序游标	有序归并	层次序号式定位	外存冗余式容错
遍历复用	主子同维表一体化	片状索引及缓存	内存备胎式容错
预过滤遍历	单边HASH连接	倍增自由分段并行	集群维表

各场景下应用的技术

跑批计算库

- 高性能存储
- 遍历复用
- 关联分类

在线查询库

- 高性能存储
- 有序利用
- 分布集群

多维分析库

- 高性能存储
- 高效关联
- 并行计算

内存高速库

- 高性能存储
- 高效关联
- 内存复用

集算器性能表现

*数据规模：100亿行；集群数量：4

测试结果（时间单位：秒）					
测试用例	Intel X86芯片			国产飞腾芯片	
	SPL读文件计算	SPL读数据库计算	数据库中SQL计算	SPL读文件计算	SPL读数据库计算
连接后并集	-	3.8	>1小时	-	-
连接后交集	-	3.9	>1小时	-	-
多对多连接遍历	**69**	103	>1小时	**93**	268
有序分组遍历	**100**	647	>1小时	**102**	2037
多步过程计算	**272**	848	>1小时	**377**	>1小时
大分组	**39**	155	2573	**56**	2493
大表关联分组	**111**	566	>1小时	**178**	2106
批量键值查询	**15**	>1小时	>1小时	**15**	>1小时

【注】SPL是润乾集算器采用的程序设计语言；SQL是关系数据库采用的程序设计语言

国产飞腾芯片上运行的润乾集算器可以超越Intel芯片上分布式数据库的性能

FAQ 常见问题

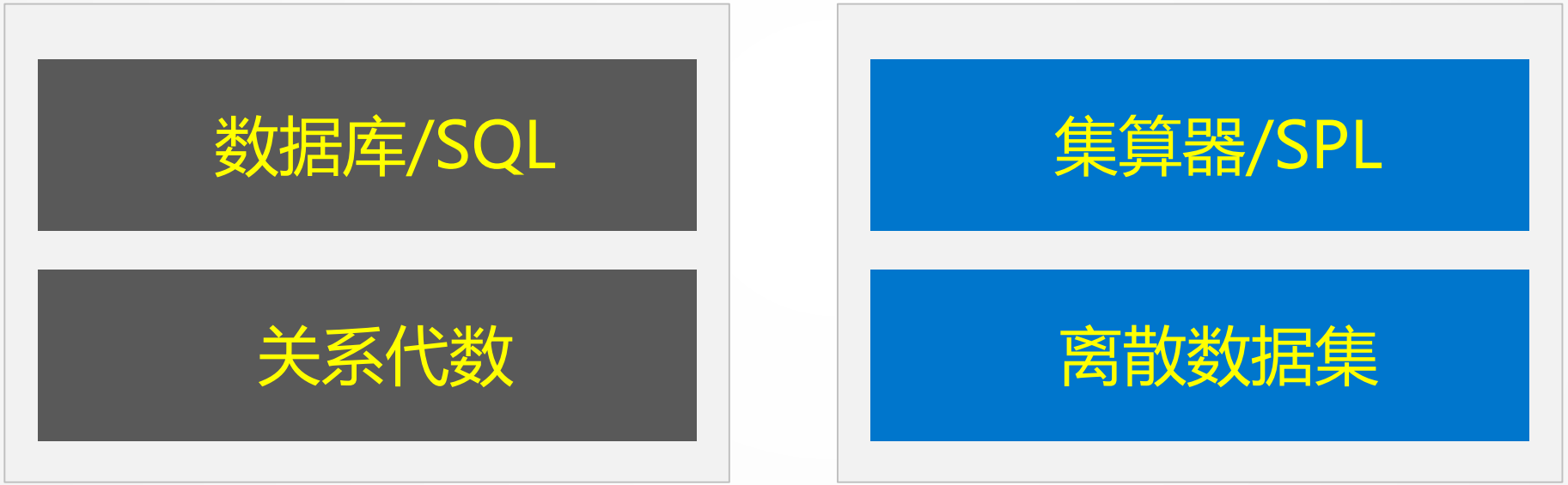
集算器要自行存储数据吗

必须！数据密集型计算的存储是性能保障，传统RDB和HADOOP的低效存储无法实现高性能

集算器针对内存、外存、集群都设计有专用高效数据组织方案，适应于多种运算场景

集算器基于开源或数据库技术?

集算器基于全新的计算模型，无开源技术可以引用，从理论到代码全部自主创新



基于创新理论的集算器不能再使用SQL实现高性能，SQL无法描述大部分低复杂度算法
仅对于运算形式规整的多维分析可提供高性能SQL接口，以适应各种前端BI工具

集算器的学习难度如何?

集算器专门用于性能优化，提供了专用的SPL语法

学习SPL不难，数小时即可掌握，数周就能熟练

难的是设计优化算法!

SPL学会如翻掌

优化算法想断肠

所以我们设计了

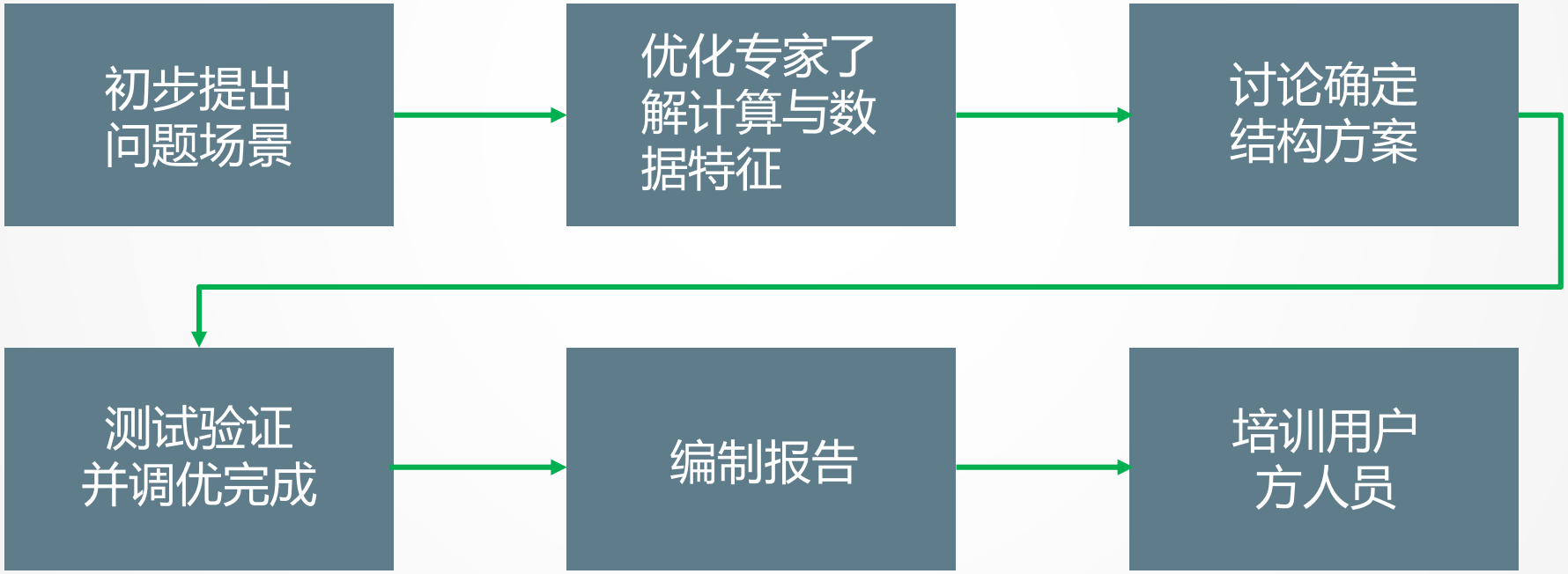
下面的优化流程

性能优化流程

最初的1-2个场景，由润乾高级工程师介入配合用户实现

大多数程序员习惯了SQL思维方式，不熟悉高性能算法，需要用一两个场景训练和理解

几十种性能优化套路经历过也就学会了，算法设计和实现并不是那么难



授人以鱼不如授人以渔！

创新技术
推动应用进步！


各设备故障分析
近7天设备的总故障率
近7天
近1天
近6小时
1.53%
0.95%
2.40%
近7天故障情况
全年故障综合分析
近7天故障次数前5的设备
服务器CPU使用率
今日
昨日
今天硬盘情况