

友乾营

# 报表性能分析

天下武功唯快不破 · 报表性能分析

# 报表运行过程

友乾营



数据准备



数据传输



报表计算



报表呈现

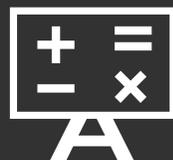
# 报表性能问题爆发程度



数据准备



数据传输



报表计算



报表呈现

# 怎么定位是哪个阶段的问题?

## 分析 报表 日志

INFO: 开始运算报表, 首先取数 → 数据准备&数据传输

INFO: 取数结束,开始运算 → 报表计算

INFO: 计算结束:

INFO: 开始生成HTML → 报表呈现

INFO: 生成HTML结束

# I 选择合适的日志级别

- 日志级别要修改：`raqsoftReportLog.properties`
- 日志输出级别可为`OFF,SEVERE,WARNING,INFO,DEBUG`，优先级依次降低，若为`OFF`则不输出任何
- 注意：配置文件中输入信息的说明！

开发调试阶段：`DEBUG/INFO`

系统上线后：`SEVERE`

# 1

**报表数据准备慢**

# 报表数据准备慢

INFO: 开始运算报表, 首先取数.....

[2019-03-30 09:38:12]

INFO: 下面开始打出sql

[2019-03-30 09:38:12]

INFO: ds1=SELECT 产品.产品ID,产品.产品名称,产品.类别ID,产品.单价,产品.单位数量,产品.库存量 FROM 产品 where 产品名称 like '%胡椒%'

[2019-03-30 09:58:22]

INFO: 取数结束,开始运算

[2019-03-30 09:59:52]

INFO: 计算结束:

数据准备花费: 20分10秒

**重点优化对象!**

# 报表数据准备慢

## 运算慢 (计算密集型)

- **关联多**  
SQL存在多个表JOIN
- **嵌套多**  
计算逻辑复杂，SQL需要多层嵌套

## 定位问题时

可将SQL放到数据库端执

行看时间

# 数据准备慢跟报表工具无关

**这个锅不能背!**

当然,如果是你SQL写的烂就说不过去了~

# 报表数据准备慢 - 解决思路

优化方向：提升算法效率

## 关联多优化手段：

- 建宽表（空间换时间）- 治标不治本
- 使用更高效的关联算法
- 使用更高效的存储格式（预关联）

## 嵌套多优化手段：

- 预计算（空间换时间）- 实用性很差
- 使用高性能算法（RDB很难实现）

可以从根本上解决关联性能问题

2

**数据传输慢**

# 报表数据传输慢

## 取数慢 (数据密集型)

- **数据量大**  
数据集不多，但数据集数据量很大
- **数据集多**  
单个数据集数据量不大，但数据集多

## 定位问题时：

- 首先关注数据库类型
- 其次确认数据规模（取数总量达十万级）
- 最后确认SQL在数据库端执行时间

# 不同数据库JDBC-IO性能不同

## 关系数据库

- MySQL < Oracle < MSSQL < DB2

## 非关系数据库

- IO性能高

注意

# 报表传输慢 - 解决思路

优化方向：提升数据传输效率

## 数据量大的优化手段：

- 数据外置，提升IO性能（要支持混算）
- 内存化，减少读盘
- 更换存储，使用SSD/磁盘阵列

## 数据集多的优化手段：

- 多线程并行取数（DB连接数足够）
- 沿用数据量大的优化手段

# ！数据传输慢跟报表工具无关

应该说，不直接相关！



数据库常用的应用程序接口是JDBC，其性能无法修改

但像HIVE还有更高效的CLI接口，报表工具如果支持则IO性能更高！

3

**报表计算慢**

# I 报表计算慢

## 格间关联多

多个数据集通过单元格表达式实现关联，如：

```
ds2.select(name,id==A  
3)
```

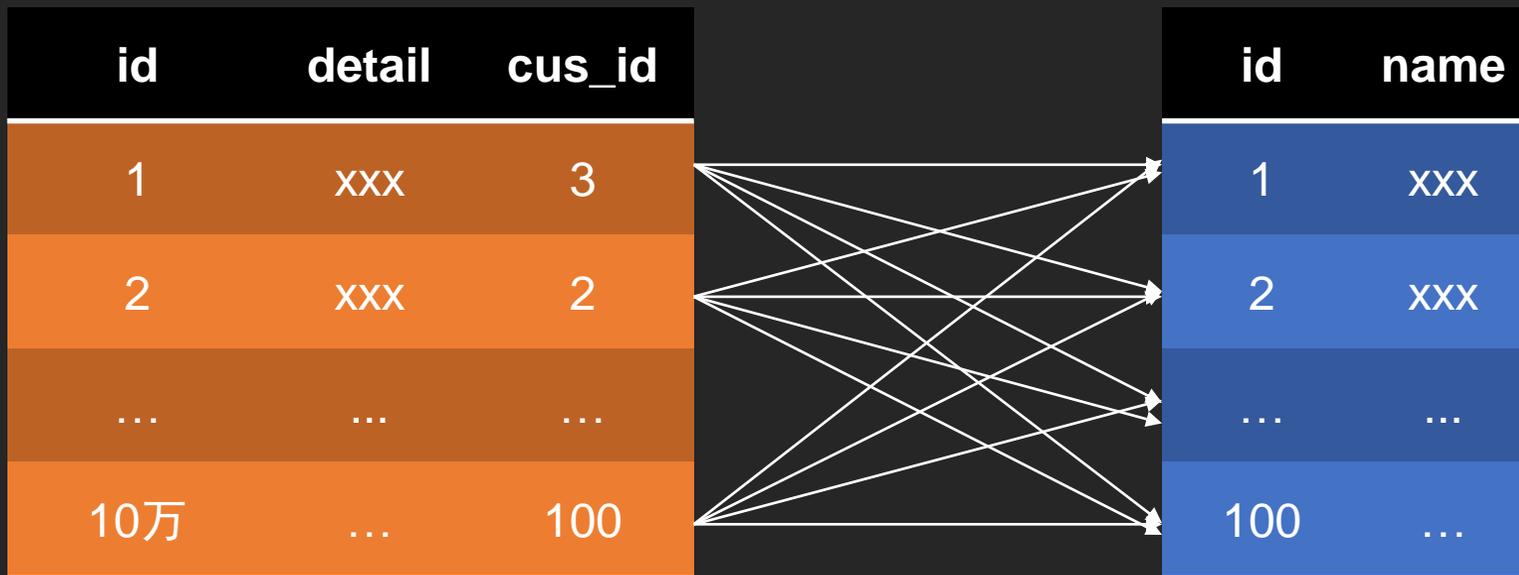
## 隐藏格多

报表中存在过渡格，但又不需要展现，设置成隐藏

## 其他

- 隐藏行列要设置在行列首格上
- 设置显示值时使用  
`ds.select(, {num_exp = 1})`
- .....

# 报表计算慢 – 格间关联多 – 解决思路



使用报表表达式进行多数据集关联，引擎会采用顺序遍历的方式完成关联，其性能远远低于数据库常用的HASH关联，相对集算器则性能更低

**在数据准备阶段处理好数据关联**

# I 报表计算慢 – 隐藏格多 – 解决思路

统计各地区前五的销售业绩，第六名以后全部归并为其他以润乾报表为例：

	A	B	C	D	E	F
1(头)						
2(头)		地区	排名	销售	订单数量	订单金额
3			=&D3	=ds1.Select(雇员ID,true,订单金额)	=ds1.订单数量	=ds1.订单金额
4		=ds1.Group(地区,false,... ds1.sum(订单金额),true)	--	其他	=sum(E3{C3>5})	=sum(F3{C3>5})
5			=B3+"地区合计"		=ds1.sum(订单数量)	=ds1.sum(订单金额)

按订单金额降序排列

大于5的行隐藏，  
表达式：if(value())>5,true,false)

销售不超过5人则隐藏行，  
表达式：if(count(D3{>5})<6,true,false)

"其他"销售人员订单数量和订单金额汇总

在数据准备阶段算好，将计算后结果返回报表

# 其他性能相关报表开发技巧

- 隐藏行列要设置在行列首格上
- 设置显示值时使用`ds.select(, , {num_exp=1})`
- 使用 `||` 和 `&&`, 而非 `or`和`and`
- 慎用`eval field`等函数
- 使用空白格减少占用
- 上线后开启缓存
- ...

以后会有独立的专题去讲

# 不同报表工具引擎性能差异很大

友乾营

得益于润乾报表的非线性报表模型

(行业标准)

润乾报表的性能表现是最好的

如果定位到报表计算端的问题，其实就可以考虑更换报表工具了

4

**报表呈现慢**

# I 报表呈现慢

报表呈现阶段：

报表计算结束后，引擎将计算后结果解析、进行页面渲染生成HTML，并返回给用户查看的过程

## 页面渲染慢

- 报表格式复杂
- 报表内容过大（如未分页）

## 用户查看慢

- 网络慢，HTML传输慢
- 客户端环境原因

# 报表格式会影响页面渲染速度

线路	车站	时间段	进出站	合计	一票通					一卡通		
					小计	车站工作票	出站票	单程票	福利票	小计	不记名成人卡	纪念卡2
总计			进站	30786	17669	4414	4391	4435	4429	13117	4337	4070
	合计		进站	30786	17669	4414	4391	4435	4429	13117	4337	4070
	白石桥南	小计	进站	4591	2635	639	599	597	800	1956	573	629
		09:00-10:00	进站	1516	863	248	204	197	214	653	242	170
		10:00-11:00	进站	1256	651	156	103	108	284	605	220	177
		11:00-12:00	进站	1819	1121	235	292	292	302	698	111	282
	车公庄	小计	进站	4322	2380	485	700	711	484	1942	723	576
		09:00-10:00	进站	1369	817	198	287	194	138	552	242	101
		10:00-11:00	进站	1427	704	157	128	288	131	723	267	160
		11:00-12:00	进站	1526	859	130	285	229	215	667	214	315
	车公庄西	小计	进站	4499	2721	617	660	685	759	1778	537	557
		09:00-10:00	进站	1294	745	166	166	175	238	549	103	188
		10:00-11:00	进站	1674	985	237	263	287	198	689	249	193
		11:00-12:00	进站	1531	991	214	231	223	323	540	185	176

隔行异色、不规则合并格等

# 不同报表工具的差异很大

润乾报表的模型做的很好  
页面渲染很快

如果定位到报表呈现端的问题，那可能也要考虑更换报表工具了

5

**如何估算报表的性能消耗**

# I 估算报表占用内存

简单估算方法:

按每个单元格0.5KB计算

报表占用内存 = 并发量\*单元格数\*0.5K

准确估算方法:

使用Tool.sizeof(报表对象)来查看报表大小



Tool.java

## 下期预告

# 《如何实现海量数据的清单和分组报表》

如何解决数据库翻页效率低、容易出现数据错误的问题？

如何实现海量数据分组报表？

如何基于非关系数据库实现？

下周三19:30

逐一探讨上述问题，并分享“异步双线程”解决思路！

掰开揉碎，讲给你听！

# 好多乾

## 润乾线上直销系统



玩转好多乾（省钱攻略）

<http://www.raqsoft.com.cn/wx/hdq-strategy-save.html>

玩转好多乾（赚钱攻略）

<http://www.raqsoft.com.cn/wx/hdq-strategy.html>