

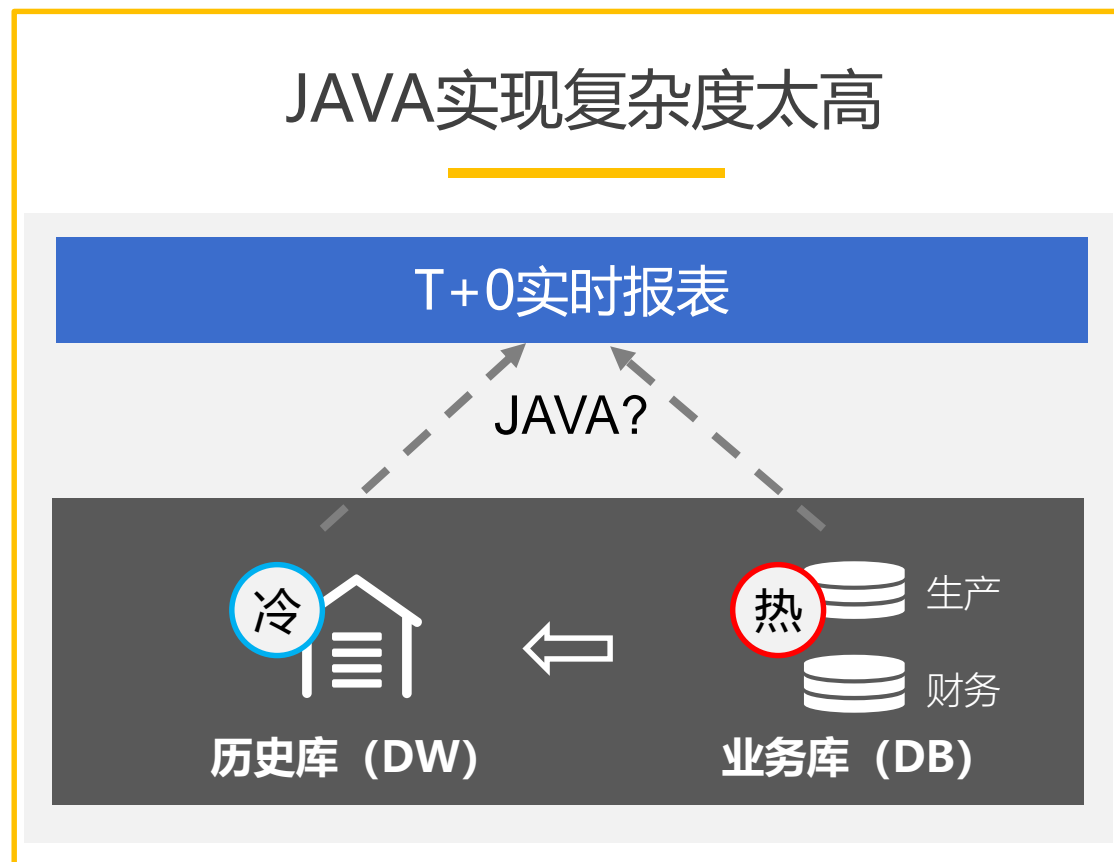
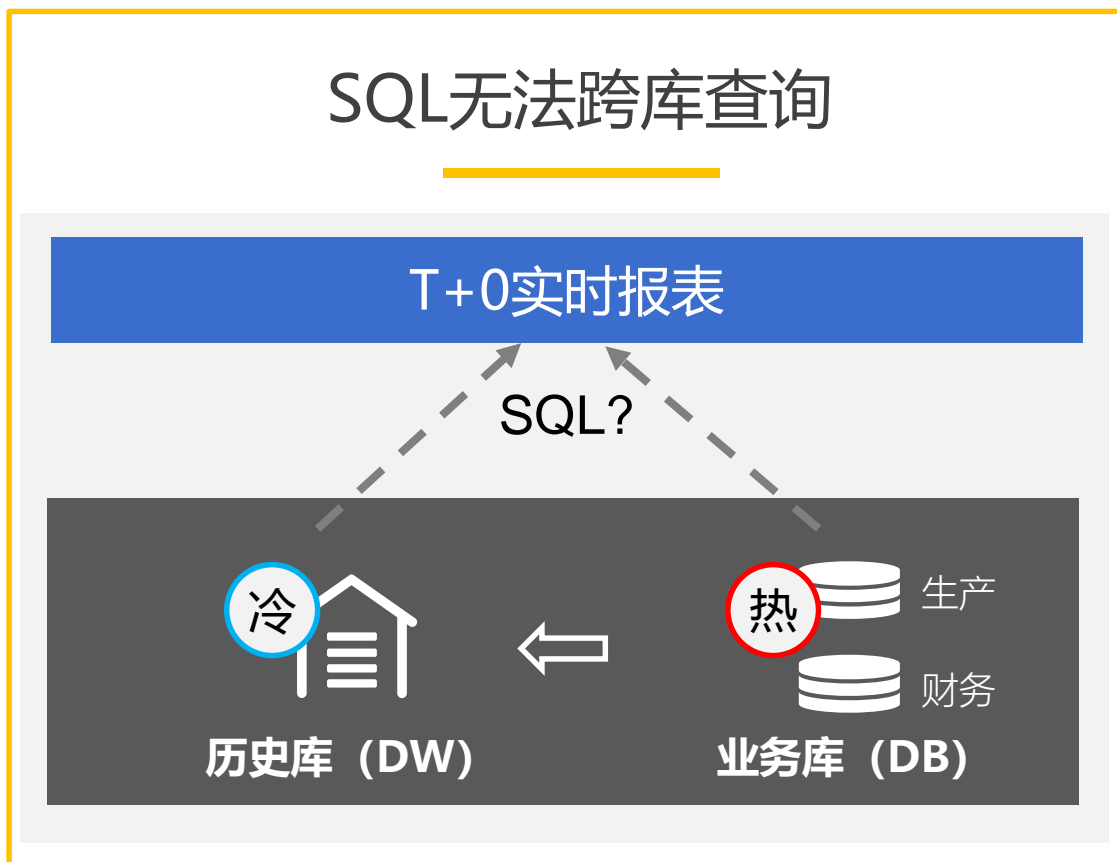
怎样开发T+0实时报表

润乾报表



为什么经常不能T+0?

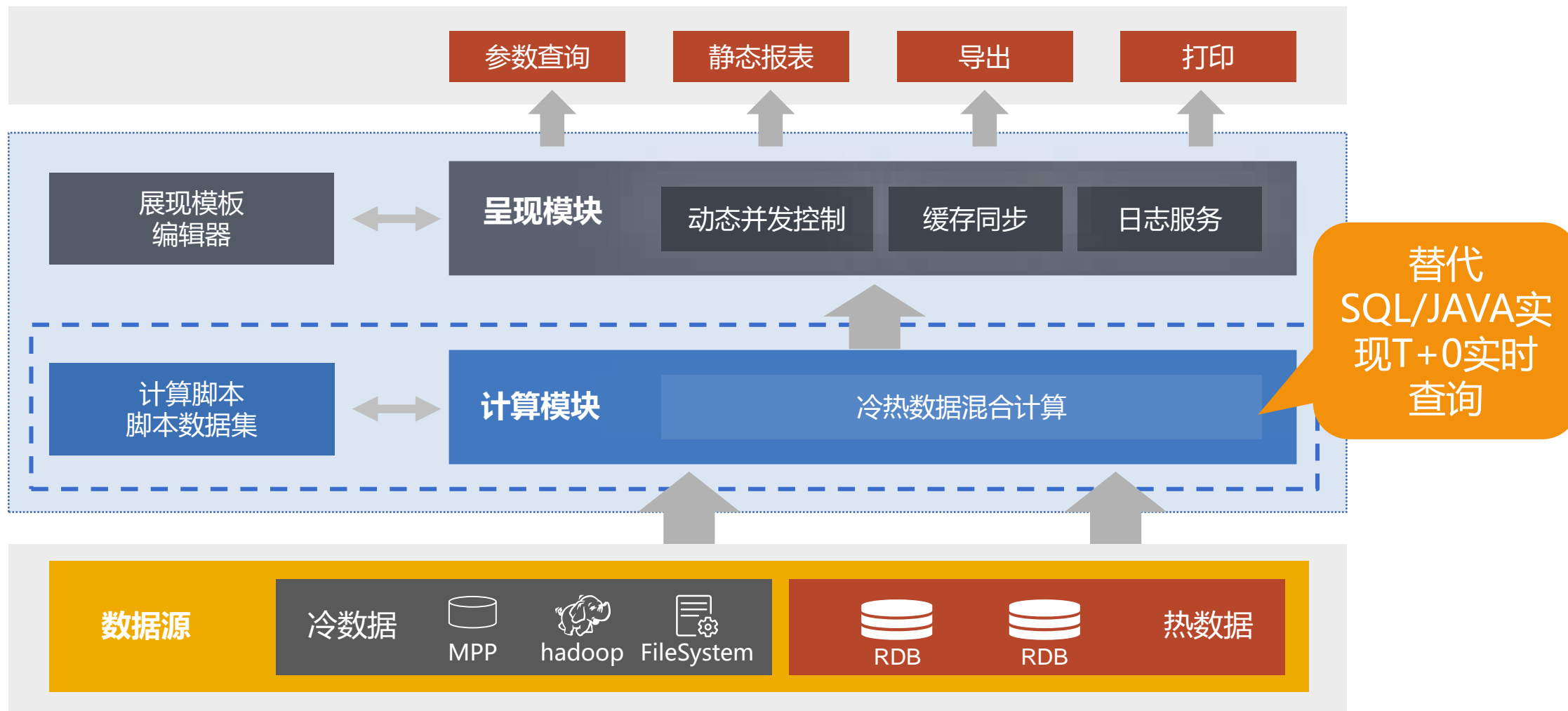
冷热数据分离后，报表通常就无法完成T+0实时查询，只能进行T+1或T+n查询
原因是SQL无法跨库（尤其是异构源）查询；使用JAVA硬编码复杂度又太高



润乾报表方案



润乾报表通过计算模块完成冷热数据实时查询，实现T+0实时报表



分类



历史库可能采用同构数据库、异构数据库或文件系统 存储历史数据，这里将分别给出实现方式

同构库

历史库与实时库
(业务库) 采用相
同类型数据库



异构库

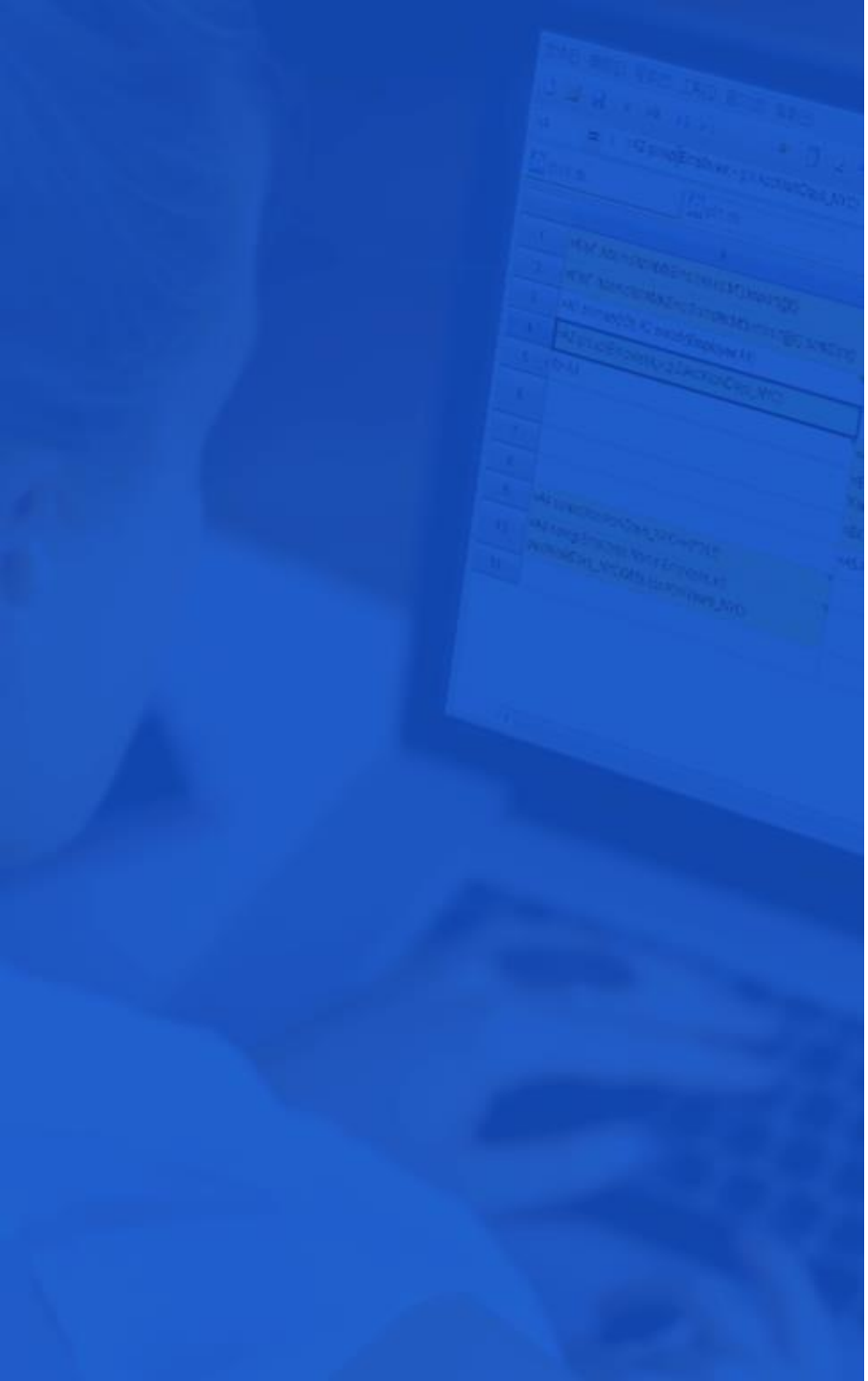
历史库与实时库
(业务库) 采用不
同类型数据库



文件源

实时库 (业务库)
采用不同类型数据
库; 历史冷数据存
储在文件中





同构库

举例



数据分布：超过3个月的订单数据会被归档到历史库中，历史库和业务库均采用MySQL

【报表需求】 查询今年每个月，各地区的销售情况

【用于比较】单库时SQL写法 (MySQL) :

```
select year(订购日期)*100+month(订购日期) 订购年月,货主地区,sum(订单金额) 订单金额,sum(运货费) 运货费 from orders where 订购日期>=date('2019-01-01') and 订购日期 <= date('2019-12-31') group by year(订购日期)*100+month(订购日期),货主地区
```

序号	订单ID	客户ID	雇员ID	订购日期	运货商	运货费	
1	10248	VINET	5	2018-07-04	3	32.38	余
2	10249	TOMSP	6	2018-07-05	1	11.61	谢
3	10250	HANAR	4	2018-07-08	2	65.83	谢
4	10251	VICTE	3	2018-07-08	1	41.34	隋
5	10252	SUPRD	4	2018-07-09	2	51.300000...	刘
6	10253	HANAR	3	2018-07-10	2	58.17	谢
7	10254	CHOPS	5	2018-07-11	2	22.98	林
8	10255	RICSU	9	2018-07-12	3	148.33	方
9	10256	WELLI	3	2018-07-15	2	13.97	何
10	10257	HILAA	4	2018-07-16	3	81.91	王
11	10258	ERNSH	1	2018-07-17	1	140.51	王
12	10259	CENTC	4	2018-07-18	3	3.25	林
13	10260	OTTIK	4	2018-07-19	1	55.09	徐

历史冷数据

序号	订单ID	客户ID	雇员ID	订购日期	运货商	运货费	
1	10688	VAFFE	4	2019-10-01	2	299.09	方
2	10689	BERGS	1	2019-10-01	2	13.42	李
3	10690	HANAR	1	2019-10-02	1	15.8	谢
4	10691	QUICK	2	2019-10-03	2	810.05000...	刘
5	10692	BSBEV	9	2019-10-03	2	61.02	蔡
6	10693	WHITC	3	2019-10-06	3	139.34	蔡
7	10694	QUICK	8	2019-10-06	3	398.36	刘
8	10695	WILMK	7	2019-10-07	1	16.72	唐
9	10696	WHITC	8	2019-10-08	3	102.55	蔡
10	10697	LINOD	3	2019-10-08	1	45.52	黄
11	10698	ERNSH	4	2019-10-09	1	272.47	王
12	10699	MORGK	3	2019-10-09	3	0.58	方
13	10700	SAVEA	3	2019-10-10	1	65.1	苏

当期热数据

➤ 润乾报表实现-同构库

1. 新建报表，设置参数

使用润乾报表新建报表，并设置查询日期起止参数dbegin和dend

2. 设置脚本数据集，实现冷热数据查询

数据集设置

名称	类型	数据源
ds1	脚本数据集	

脚本数据集

缓存文件名

	A
1	=[connect("curr_db"),connect("his_db")]
2	=A1.(~.query@x("select year(订购日期)*100+month(订购日期) 订购年月,货主地区,sum(订单金额) 订单金额,sum(运货费) 运货费 from orders where 订购日期>=? and 订购日期<=? group by year(订购日期)*100+month(订购日期),货主地区",dbegin,dend))
3	=A2.conj()
4	=A3.groups(订购年月,货主地区,sum(订单金额):订单金额,sum(运货费):运货费)
5	return A4

脚本说明



	A	B
1	<code>=[connect("curr_db"),connect("his_db")]</code>	/创建2个数据库连接，分别是当前实时库和历史库（数据源配置此处省略，请查看相关文档）
2	<code>=A1.(~.query@x("select year(订购日期)*100+month(订购日期) 订购年月,货主地区,sum(订单金额) 订单金额,sum(运货费) 运货费 from orders where 订购日期>=? and 订购日期 <= ? group by year(订购日期)*100+month(订购日期),货主地区",dbegin,dend))</code>	/根据A1中的2个连接和报表参数执行SQL分别查询冷热数据（按时间和地区分组汇总） 其中，dbegin和dend为报表参数
3	<code>=A2.conj()</code>	/合并查询结果
4	<code>=A3.groups(订购年月,货主地区;sum(订单金额):订单金额,sum(运货费):运货费)</code>	/针对合并结果再次按照月份和地区进行分组汇总
5	<code>return A4</code>	/为报表返回结果集



报表模板设计与呈现效果

根据已准备的数据集设计报表模板

D:\software\raqsoft\report\web\webapps\demo\WEB-INF\reportFiles\怎样开发T+0实时

	A	B	C	D
1	本年按月与地区订单情况汇总			
2	年月	地区	订单金额	运货费
3	=ds1.group(订购年月)	=ds1.select(货主地区)	=ds1.订单金额	=ds1.运货费

报表模板设计

发布报表，报表呈现效果如下：

本年按月与地区订单情况汇总

年月	地区	订单金额	运货费
201901	东北	2059425.13	33370.64
	华东	1638753.53	60509.10
	华北	5213404.28	239132.18
	华南	121825.94	8887.62
	西南	5158949.75	193132.70
201902	东北	2295295.78	88248.46
	华东	2460554.57	105008.48
	华北	2943345.38	120660.72
	华南	569395.20	13927.56
	西南	1055637.83	57346.26
201903	华东	3512775.78	177071.74
	华北	3867416.72	219180.18
	华东	52013.60	3830.58

报表展现效果



› 并行方式更高效

润乾报表还可以通过多线程并行执行SQL加快取数速度

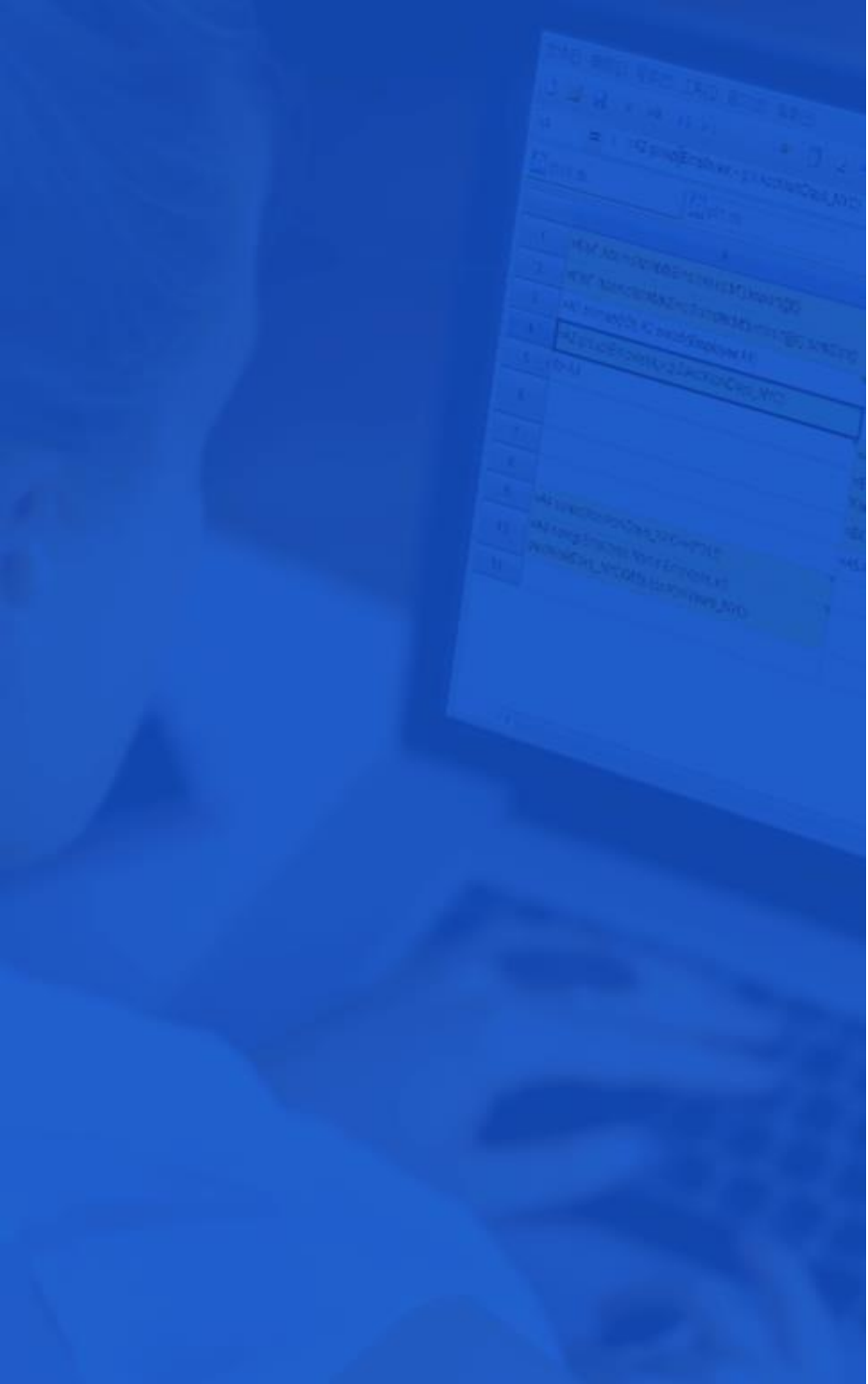
△ 设置并行数

使用并行方式执行SQL查询需要事先设置并行数，然后在脚本中通过fork启动多线程

更改raqsoftConfig.xml配置：

```
<parallelNum>6</parallelNum>
```

	A	B
1	=[connect("curr_db"),connect("his_db")]	
2	fork A1	=A2.query@x("select year(订购日期)*100+month(订购日期) 订购年月,货主地区,sum(订单金额) 订单金额,sum(运货费) 运货费 from orders where 订购日期>=? and 订购日期 <= ? group by year(订购日期)*100+month(订购日期),货主地区",dbegin,dend)
3	...	/其他部分与前例一致
4	...	
5	...	



异构库

举例



数据分布：超过3个月的订单数据会被归档到历史库MySQL中，业务库仍然使用Oracle

【报表需求】 查询今年每个月销售额最多的前三个地区

【用于比较】 单库时SQL写法 (Oracle) :

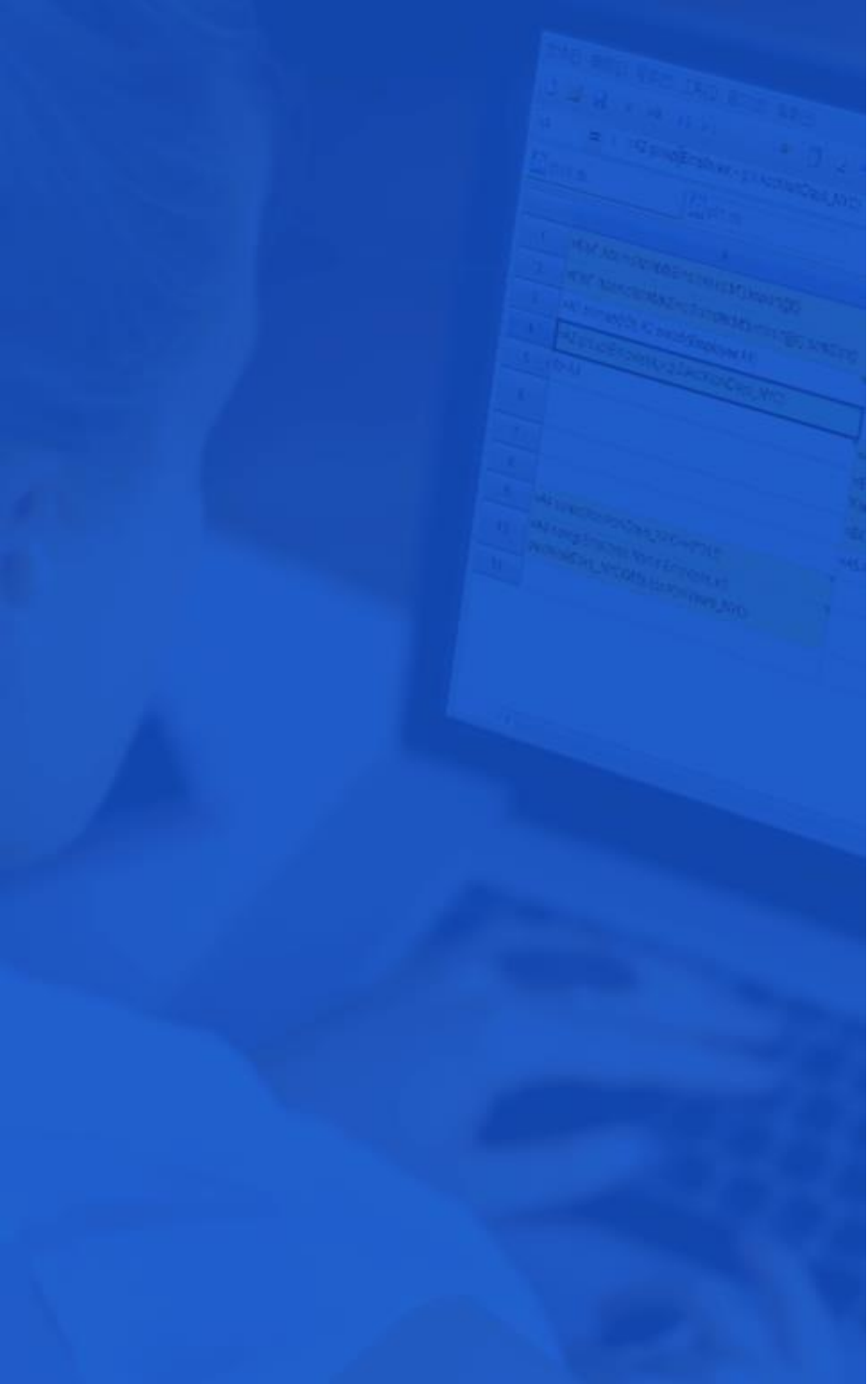
```
SELECT TB.*
FROM (SELECT o.*,
      ROW_NUMBER() OVER(PARTITION BY o.订购年月 ORDER BY o.订单金额 DESC) RN
FROM (select to_number(to_char(订购日期, 'YYYY')) * 100 +
      to_number(to_char(订购日期, 'MM')) 订购年月,
      货主地区,
      sum(订单金额) 订单金额,
      sum(运货费) 运货费
from orders
where 订购日期 >= to_date('2019-01-01', 'yyyy-MM-dd')
and 订购日期 <= to_date('2019-12-31', 'yyyy-MM-dd')
group by to_number(to_char(订购日期, 'YYYY')) * 100 +
      to_number(to_char(订购日期, 'MM')),
      货主地区) o) TB
WHERE TB.RN <= 3
```

➤ 润乾报表实现-异构库



这里仅展示脚本数据集部分，报表呈现部分与前例类似，不再赘述

	A	B	
1	=[[connect("curr_db"),"ORACLE],[connect("his_db"),"MYSQL"]]	/实时库是oracle, 历史库是mysql	
2	=SQL="select year(订购日期)*100+month(订购日期) 订购年月,货主地区,sum(订单金额) 订单金额,sum(运货费) 运货费 from orders where 订购日期>=? and 订购日期 <= ? group by year(订购日期)*100+month(订购日期),货主地区"	/SQL语句, 不同的数据库函数不同, 下面会根据数据库类型转换成对应的SQL	
3	fork A1	=SQL.sqltranslate(A3(2))	/根据数据库类型翻译对应数据库的SQL
4		=A3(1).query@x(B3,dbegin,dend)	
5	=A3.conj()		
6	=A5.groups(订购年月,货主地区;sum(订单金额):订单金额,sum(运货费):运货费)	=A6.sort(订购年月,订单金额:-1)	
7	=B6.group@o(订购年月).(~.to(1,3))	/取每个月份的前三名	
8	=A7.conj()	return A8	



文件源



➤ 举例

数据分布：超过3个月的订单数据会被归档到文件系统中，业务库仍然采用MySQL

【报表需求】 查询今年每个月销售额超过200万的地区

【用于比较】单库时SQL写法 (MySQL) :

```
select year(订购日期)*100+month(订购日期) 订购年月,货主地区,sum(订单金额) 订单金额,sum(运货费) 运货费
from orders
where 订购日期>=date('2019-01-01') and 订购日期 <= date('2019-12-31')
group by year(订购日期)*100+month(订购日期),货主地区
having 订单金额>=2000000
```

➤ 润乾报表实现-文件源



当历史数据被归档到文件系统中，如使用文本存储冷数据，热数据仍然在数据库中，脚本数据集如下：

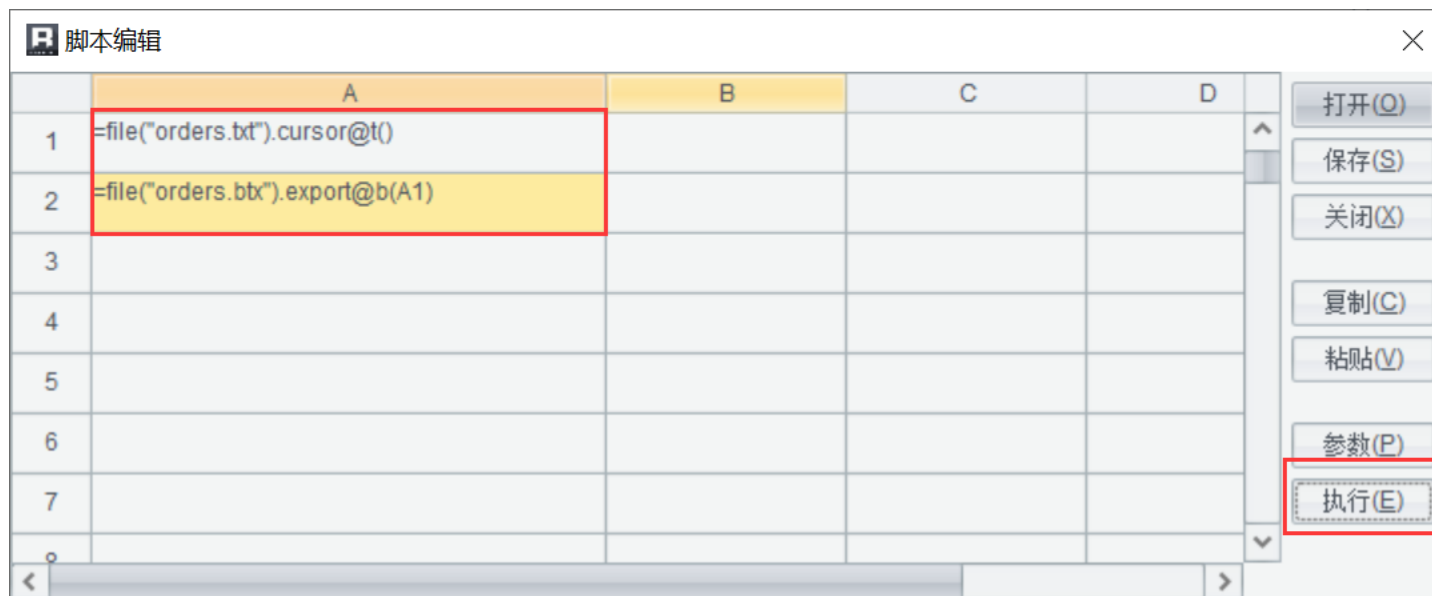
	A	B
1	=connect("curr_db")	/连接当前实时库
2	=A1.query@x("select year(订购日期)*100+month(订购日期) 订购年月,货主地区,sum(订单金额) 订单金额,sum(运货费) 运货费 from orders where 订购日期>=? and 订购日期 <=? group by year(订购日期)*100+month(订购日期),货主地区",dbegin,dend)	/根据参数汇总实时数据
3	=file("orders.txt").cursor@t(订购日期,货主地区,订单金额,运货费)	/建立历史冷数据文件游标
4	=A3.select(订购日期>=dbegin && 订购日期<=dend)	/根据参数过滤冷数据
5	=A4.groups(year(订购日期)*100+month(订购日期):订购年月,货主地区;sum(订单金额):订单金额,sum(运货费):运货费)	/汇总冷数据
6	=[A2,A5].conj()	/合并冷热汇总结果
7	=A6.groups(订购年月,货主地区;sum(订单金额):订单金额,sum(运货费):运货费)	/汇总合并后结果
8	=A7.select(订单金额>=2000000)	/筛选订单额大于200万的地区
9	return A8	

如何获得更高性能?

文本性能较差，润乾报表还提供了更高效的二进制数据格式：**简表**（也称集文件），使用简表存储历史冷数据可以获得更高的查询性能，润乾报表提供了从文本转化简表的功能

文本转简表

打开润乾报表IDE-工具-脚本编辑，编写转换脚本。点击执行，即可将文本转换为简表





› 简表的使用

简表的使用与文本基本一致（红色部分不同）：

	A	B
1	...	/省略部分与上述文本部分代码完全一致
2	...	
3	=file("orders.btx").cursor@b(订购日期,货主地区,订单金额,运货费)	/读取简表历史冷数据
4	...	
5	...	
6	...	
7	...	
8	...	
9	...	

处理使用脚本查询文件，润乾报表还提供了SQL查文件的方式，详情参考：[《有效减少数据库中间表的报表开发方法》](#)



➤ 润乾报表都能实施哪些计算

前面通过脚本实现了group by, where, having等计算, 还能做更多:

1. 分库的概念
2. where
3. order by
4. limit Y offset X
5. sum/count/avg/max/min
6. group by
7. group by having
8. distinct
9. count(distinct)
10. count(distinct) group by
11. 带join on的查询
 - 同步分库
 - 维表冗余
 - 维表内存化
12. 集合运算
13. 集合运算的简单情况
14. join on转为集合运算
15. 异构数据库分库

关于分库查询和T+0实时查询更多内容请参考：

分库

[《分库后的查询统计》](#)

T+0

[《高性能不停机 T+0 的实现》](#)

想要了解更多 请联系我们



技术内容请移步 乾学院
<http://c.raqsoft.com.cn>



优惠价购买请加入 好多乾
<http://sys.misdiy.com/hdq.html>

