

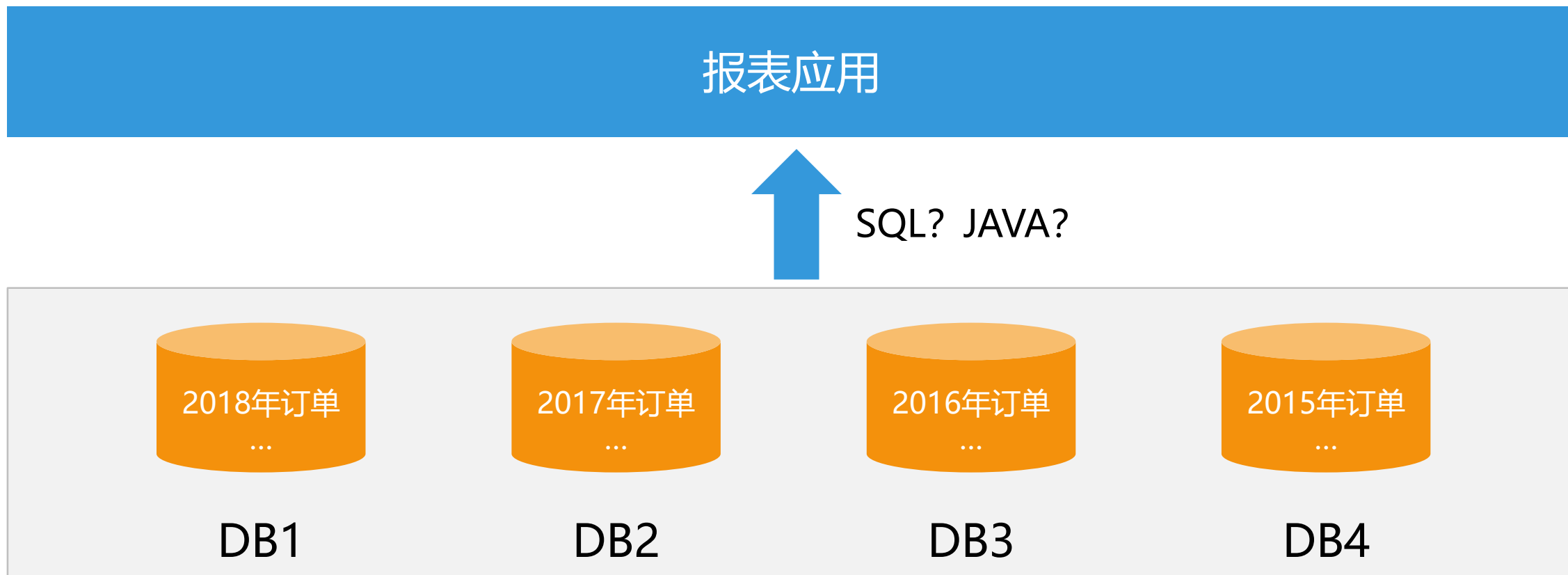
## 分库后的报表怎么做？



## ➤ 分库对报表开发的影响



数据分库以后报表就无法再使用SQL直接查询数据，往往要在应用端借助硬编码进行关联、排序等运算，报表开发复杂度增大





## 借助数据库中间件

通过数据库中间件使用SQL查询多库数据

缺点：架构复杂，对报表应用来说过于沉重；  
技术完善度不够；不支持异构库

报表应用

数据库中间件



RDB



RDB



RDB



RDB



RDB



RDB

## 应用端硬编码

使用JAVA等硬编码，处理多库数据运算

缺点：过于复杂，实施难度太大

报表应用

硬编码



RDB



RDB



RDB



RDB



RDB

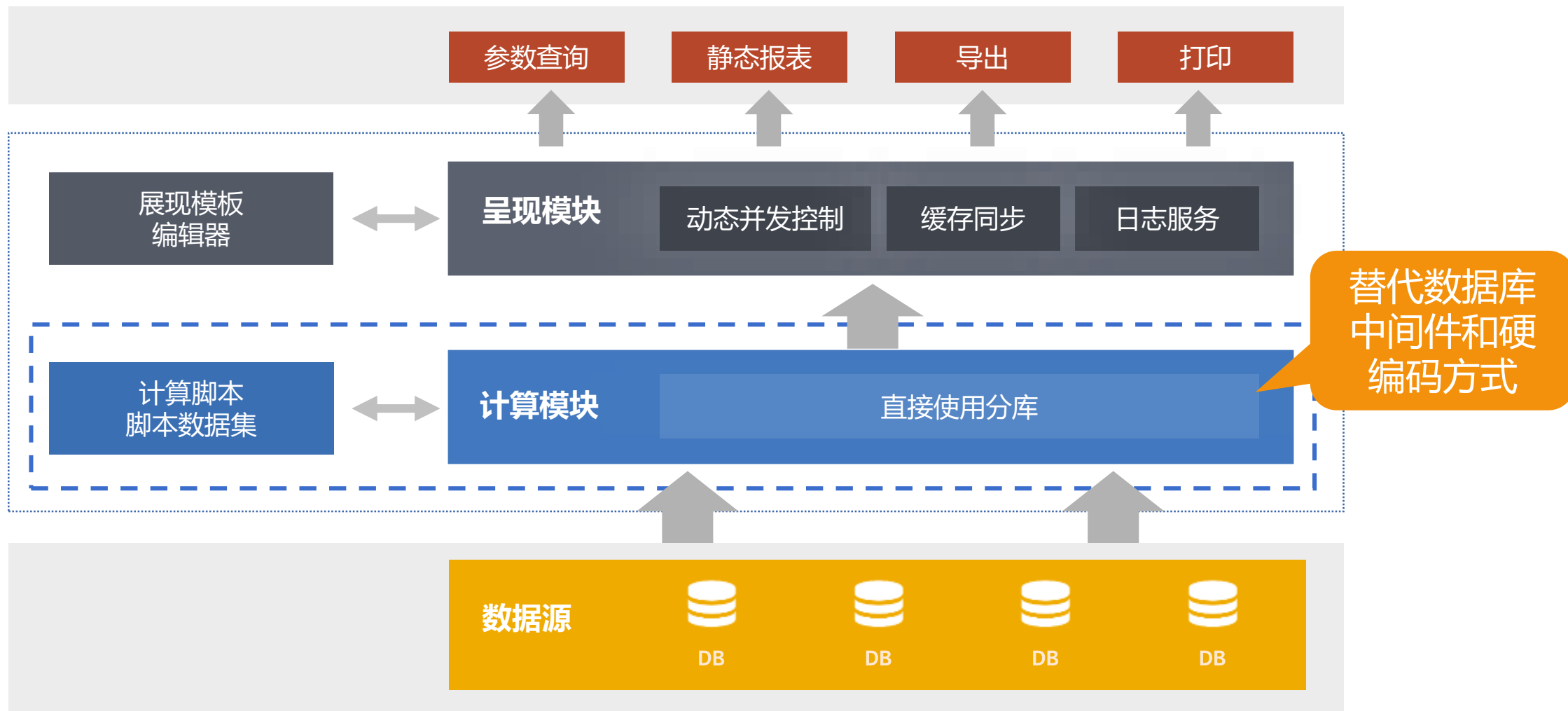


RDB

## ➤ 润乾报表方案



润乾报表通过计算模块**直接使用**分库，整体架构清晰，实现简单



## › 要解决哪些问题？



关键

1

如何简单快速地使用分库出报表

-趋近于使用单表做报表的简便程度

2

如何进行异构数据源间的关联运算

-当分库遇到异构数据源时如何关联计算

3

如何保证大数据量的查询性能

-当结果集比较大，内存容量不够时如何保证计算顺利进行

4

有无海量数据快速的呈现方式

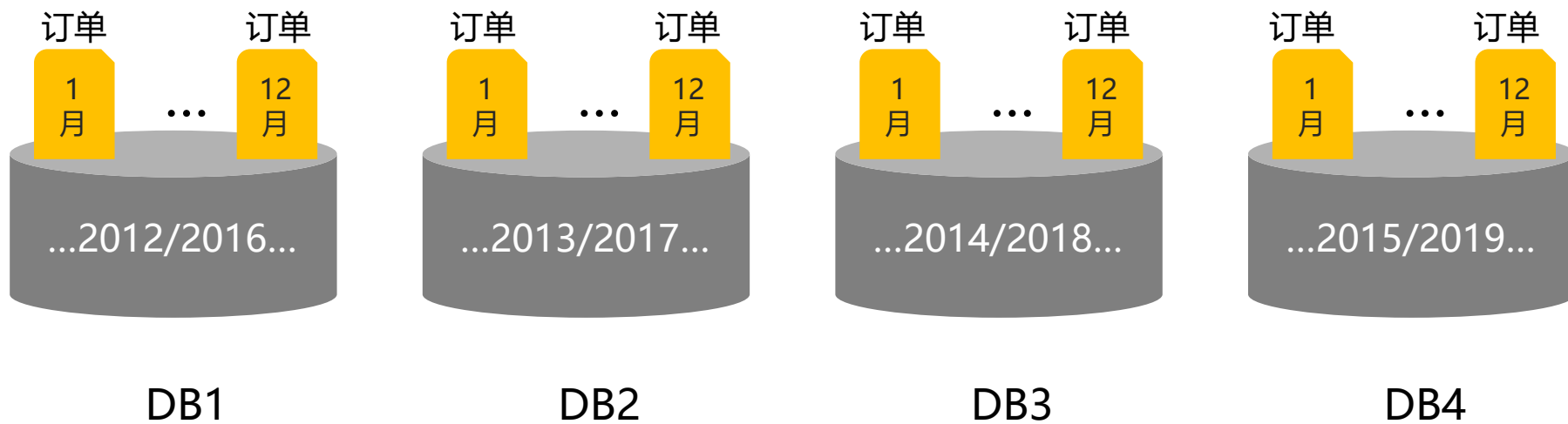
-查询的清单结果很多时，如何快速呈现

## 怎么做？举例



以横向拆分为例，订单交易数据按照年份%4分别存储在4个数据库中

【报表需求】查询指定时间范围的订单汇总数据



## ➤ 润乾报表实现



新建报表，设置查询日期起止参数

参数编辑

名称	描述	数据类型	值表达式	参数类型
dbegin	开始日期	日期	2012-01-01	普通参数
dend	终止日期	日期	2018-12-31	普通参数

确定(O)

取消(C)

设置数据集，这里使用脚本数据集查询分库数据

数据集设置

名称	类型	数据源
ds1	脚本数据集	

确定(O)

取消(C)

脚本数据集

缓存文件名

确定(O)

取消(C)

复制(C)

粘贴(P)

	A
1	=[connect("mydb1"),connect("mydb2"),connect("mydb3"),connect("mydb4")]
2	=A1.(~.cursor@x("select 客户ID,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 客户ID",dbegin,dend))
3	=A2.mcursor()
4	return A3.fetch()

## 脚本说明



	A	B
1	<code>=[connect("mydb1"),connect("mydb2"),connect("mydb3"),connect("mydb4")]</code>	/创建4个数据库连接，这里都是MySQL。（数据源配置此处省略，请查看相关文档）
2	<code>=A1.(~.query@x("select 客户ID,sum(订单金额) 订单金额,sum(运货费) 运货费 from orders where 订购日期&gt;=? and 订购日期 &lt;=? group by 客户ID",dbegin,dend))</code>	/根据A1中的4个连接和报表参数执行SQL查询数据（按客户分组汇总）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(客户ID;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客户进行分组汇总
	<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

发布报表，报表呈现效果如下：

	A	B	C
1(标)	按客户统计订单金额和运货费		
2(标)	="统计时段: "+string(dbegin)+"到"+string(dend)		
3(头)	客户	订单总额	运费总额
4	=ds1.select(客户ID)	=ds1.订单金额	=ds1.运货费

报表模板设计

按客户统计订单金额和运货费

统计时段: 2012-01-01到2018-12-31

客户	订单总额	运费总额
ALFKI	709803.83	872085.60
ANATR	2204233.41	116325.76
ANTON	4892448.55	165720.36
AROUT	9948984.37	410830.92
BERGS	14270158.23	1076267.76
BLAUS	3446221.42	370551.04
BLONP	11236684.65	1210103.88
BOUD	11236684.65	1210103.88

报表展现效果

## ➤ 还可以做更多...



此外，我们在《[分库后的查询统计](#)》中详细列举了分库后的各种运算，可以直接使用

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

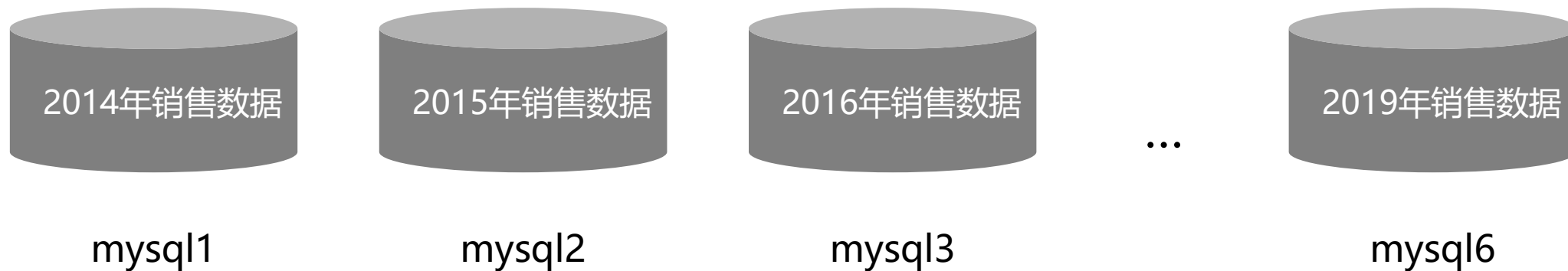
如何转换到润乾报表中使用呢？非常简单

## ➤ 润乾报表实现



以《分库后的查询统计》中“where”的例子来说明润乾报表的实现方法

【报表需求】查询单笔销售额大于500的所有订单记录






# ➤ 润乾报表实现

## 1.配置数据源

在报表设计器和WEB端分别配置6个MySQL的数据源连接，分别是mysql1到mysql6

## 2.设置参数

新建报表，设置报表参数：amount，用于过滤数据

 参数编辑

名称	描述	数据类型	值表达式	参数类型
amount	订单金额	数值	500	普通参数

确定(O)

取消(C)

## ➤ 润乾报表实现



### 3.设置数据集，准备数据

使用脚本数据集（内容与《分库后的查询统计》中一致）

数据集设置

名称	类型	数据源
ds1	脚本数据集	

确定(O)  
取消(C)

脚本数据集

缓存文件名

	A	B
1	=6.(connect("mysql"+string(~)))	
2	=SQL="select * from sales where amount >"+string(amount)	
3	fork A1	=A3.query@x(SQL)
4	=A3.conj()	
5	return A4	

确定(O)  
取消(C)  
复制(C)  
粘贴(P)



## ➤ 润乾报表实现

### 4.设置并行数

脚本A3中使用fork启动了多线程并行从6个数据库中取数，需要事先设置并行数  
更改raqsoftConfig.xml配置：

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

### 5.编辑报表表达式

根据已准备的数据集，设置报表表达式

	A	B	C	D	E
1	="单笔订单金额大于"+string(amount)+"的订单"				
2	订单	客户	销售	订单金额	订购日期
3	=ds1.select(orderid)	=ds1.client	=ds1.sellerid	=ds1.amount	=ds1.orderdate

### 6.发布报表，完成报表开发



## ➤ 润乾报表实现

### 4.设置并行数

脚本A3中使用fork启动了多线程并行从6个数据库中取数，需要事先设置并行数

### 4.设置报表表达式

根据已准备的数据集，设置报表表达式

	A	B	C	D	E
1	=“单笔订单金额大于”+string(amount)+“的订单”				
2	订单	客户	销售	订单金额	订购日期
3	=ds1.select(orderid)	=ds1.client	=ds1.sellerid	=ds1.amount	=ds1.orderdate

## ➤ 结果集较大时的呈现方式



当要呈现的结果集较大，如上例无法将数据全部加载到内存，润乾报表提供了异步呈现功能，可以分批读取数据，异步呈现

将上述脚本数据集改为：

	A	B
1	=6.(connect("mysql"+string(~)))	
2	=SQL="select * from sales where amount >"+string(amount)	
3	=A1.(~.cursor(SQL))	循环执行SQL，返回游标
4	=A3.conjx()	归并游标，此时为报表返回的仍然是游标

在润乾报表中，可以直接接收游标进行海量数据异步呈现，这样的报表称为大报表；大报表的使用请参考：[《海量清单与分组报表的实现》](#)