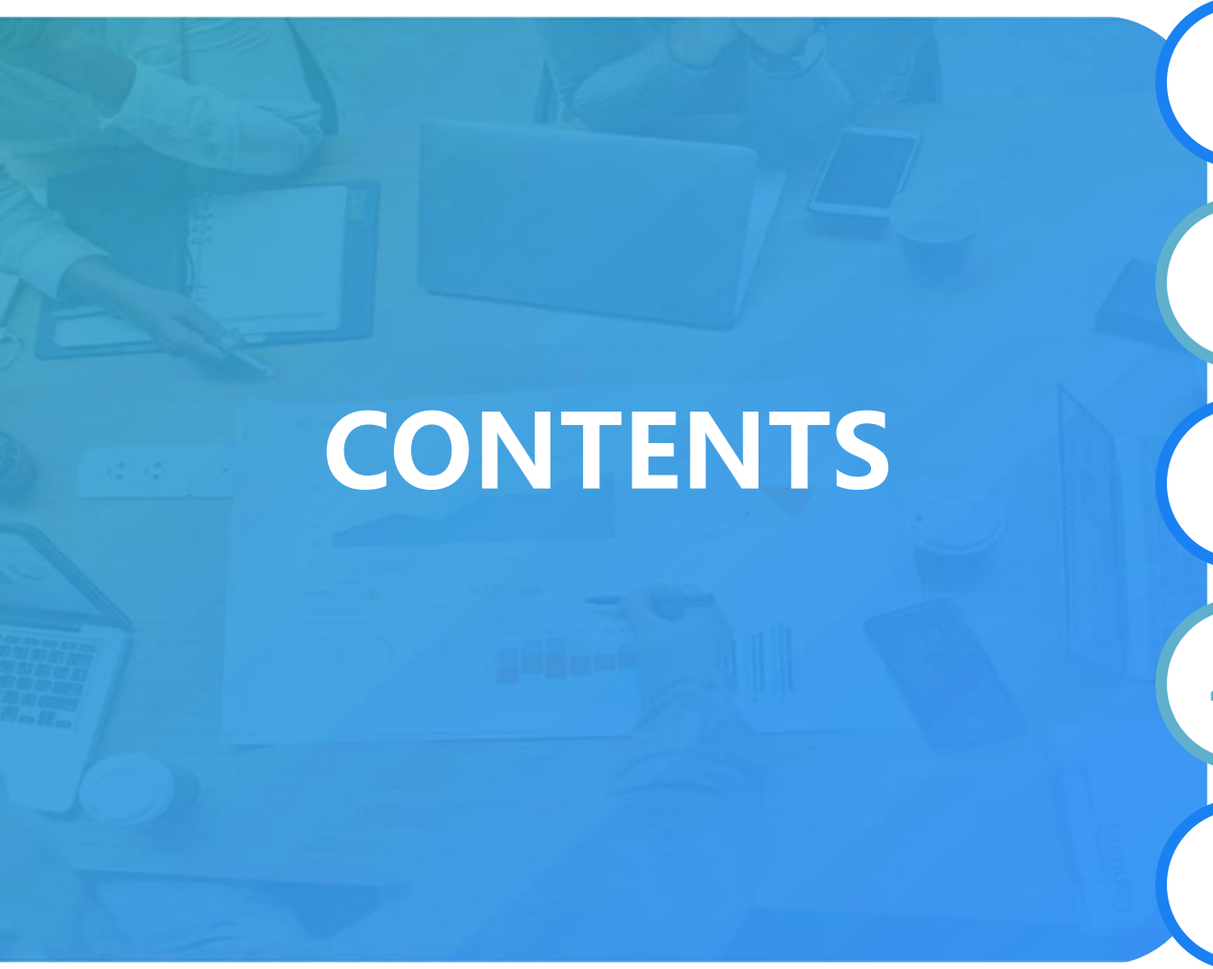
Cursor



CONTENTS

1 Delayed Calculation

2 Parallel Cursor

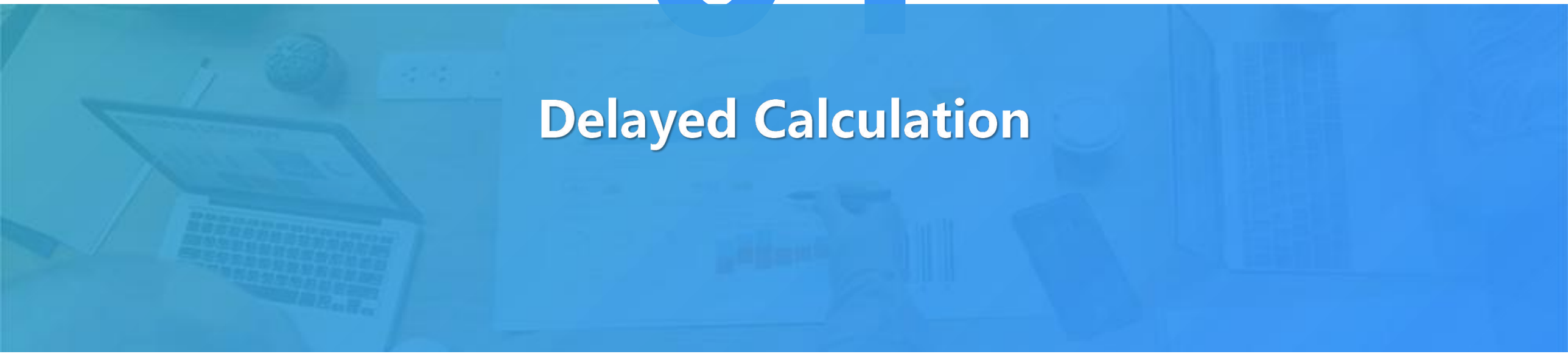
3 Channel

4 Ordered Cursors

5 Program Cursor

01

Delayed Calculation



Example: Select employees over 40 years old.

	A
1	=file("E:/txt/Employees.txt").import@t()
2	=A1.select(age(Birthday)>=40)

A1~A2 results:

Calculate immediately

Index	ID	Name	Gender	Post	Birthday	AccountNo	BasePay
1	1	Mike	Female	Sale	1968-12-0...	536936891...	5600.0
2	2	Jake	Male	Vice Presid...	1962-02-1...	964107677...	2500.0
3	3	Lucy	Female	Sale	1973-08-3...	665248245...	10800.0
4	4	Andy	Male	Sales Man...	1968-09-1...	650028860...	7500.0
5	5	Jim	Male	Sales Man...	1965-03-0...	441380247...	4700.0

Index	ID	Name	Gender	Post	Birthday	AccountNo	BasePay
1	1	Mike	Female	Sale	1968-12-0...	536936891...	5600.0
2	2	Jake	Male	Vice Presid...	1962-02-1...	964107677...	2500.0
3	3	Lucy	Female	Sale	1973-08-3...	665248245...	10800.0
4	4	Andy	Male	Sales Man...	1968-09-1...	650028860...	7500.0
5	5	Jim	Male	Sales Man...	1965-03-0...	441380247...	4700.0

	A	B
1	=file("E:/txt/Employees.txt").cursor@t()	/Generate cursor for file in external storage
2	=A1.select(age(Birthday)>=40)	/Attach Select calculation to cursor
3	=A2.fetch()	/Fetch data according to condition

A1~A3 results:

Calculate later

Value
com.raqsoft.dm.cursor.FileCursor@229db918

Value
com.raqsoft.dm.cursor.FileCursor@229db918

Index	ID	Name	Gender	Post	Birthday	AccountNo	BasePay
1	1	Mike	Female	Sale	1968-12-0...	536936891...	5600.0
2	2	Jake	Male	Vice Presid...	1962-02-1...	964107677...	2500.0
3	3	Lucy	Female	Sale	1973-08-3...	665248245...	10800.0
4	4	Andy	Male	Sales Man...	1968-09-1...	650028860...	7500.0
5	5	Jim	Male	Sales Man...	1965-03-0...	441380247...	4700.0

Example: Calculate the average wages of employees over 40 years old and under 40 years old in the company.

	A
1	=file("E:/txt/Employees.txt").import@t()
2	=A1.groups(age(Birthday)>=40:ifgt40;avg(BasePay))

A1~A2 results:

Calculate immediately

Index	ID	Name	Gender	Post	Birthday	AccountNo	BasePay
1	1	Mike	Female	Sale	1968-12-0...	536936891...	5600.0
2	2	Jake	Male	Vice Presid...	1962-02-1...	964107677...	2500.0
3	3	Lucy	Female	Sale	1973-08-3...	665248245...	10800.0
4	4	Andy	Male	Sales Man...	1968-09-1...	650028860...	7500.0
5	5	Jim	Male	Sales Man...	1965-03-0...	441380247...	4700.0

Index	ifgt40	avg(BasePay)
1	false	6940.0
2	true	6007.692307692308

	A
1	=file("E:/txt/Employees.txt").cursor@t()
2	=A1.groups(age(Birthday)>=40:ifgt40;avg(BasePay))

A1~A2 results:

Calculate immediately

Value
com.raqsoft.dm.cursor.FileCursor@7d632866

Index	ifgt40	avg(BasePay)
1	false	6940.0
2	true	6007.692307692308

Product information and sales information are stored in two tables. Please calculate total sales.

	A	B
1	=file("E:/txt/Products.txt").import@t().primary@i(ID)	/Import the Product table and establish an index
2	=file("E:/txt/Sales.txt").cursor@t()	/Create a cursor for traversal
3	=A2.select(quantity<=10)	/Filter the cursor and return a cursor too
4	=A3.switch(productid,A1:ID)	/Generate a joining pointer with a cursor returned
5	=A4.groups(;sum(quantity*productid.Price):total)	/Perform aggregation to calculate sum

Delayed Calculation

Calculate immediately

A1~A4 results:

Index	ID	Name	Category	Price
1	1	Apple juice	Low-end	18.0
2	2	Mile	Low-end	19.0
3	3	Tomato sa...	Low-end	10.0
4	4	Salt	Low-end	22.0
5	5	Sesame oil	Low-end	21.35

Value
com.raqsoft.dm.cursor.FileCursor@249b5d3b

Value
com.raqsoft.dm.cursor.FileCursor@249b5d3b

Index	total
1	142740.18000000008

Note: The cursor only traverses once, so it is not possible to repeat the fetching of data.

02

Parallel cursor

Find the sales records of January from the sales records of the products.

Single cursor

	A
1	=now()
2	=file("E:/txt/PRODUCT_SALE.txt").cursor@t()
3	=A2.select(month(DATE)==1)
4	=A3.fetch()
5	=interval@ms(A1,now())

Value
9022

Multiplex cursor

	A	B
1	=now()	
2	=file("E:/txt/PRODUCT_SALE.txt").cursor@mt()	
3	fork A2	=A3.select(month(DATE)==1)
4		=B3.fetch()
5	=A3.conj()	=interval@ms(A1,now())

Value
5950

Result:

Index	ID	PID	DATE	QUANTITY	SID
1	1211	10075052	2010-01-01	84	10225
2	2474	10098045	2010-01-01	106	10591
3	10576	10093980	2010-01-01	53	10720
4	12938	10069598	2010-01-01	30	10483
5	15091	10067138	2010-01-01	104	11000

Non-Multiplex cursor

	A	B
1	=now()	
2	=file("E:/txt/PRODUCT_SALE.txt").cursor@mt()	
3	fork A2	=A3.select(month(DATE)==1)
4	=A3.(~.fetch())	
5	=A4.conj()	=interval@ms(A1,now())

Value
8659

Note: If immediate calculation is not performed in fork, multithreading is only used to define cursors, which does not increase the speed. The calculation time is the same when the "fork" in A3 is replaced by "for".

Multi-cursor parallelism can be implemented in a simple way

	A
1	=now()
2	=file("E:/txt/PRODUCT_SALE.txt").cursor@mt()
3	=A2.select(month(DATE)==1)
4	=A3.fetch()
5	=interval@ms(A1,now())

値
5089

Multiplex cursor
Reading data and computing are parallel at the same time

	A
1	=now()
2	=file("E:/txt/PRODUCT_SALE.txt").cursor@t().mcursor()
3	=A2.select(month(DATE)==1)
4	=A3.fetch()
5	=interval@ms(A1,now())

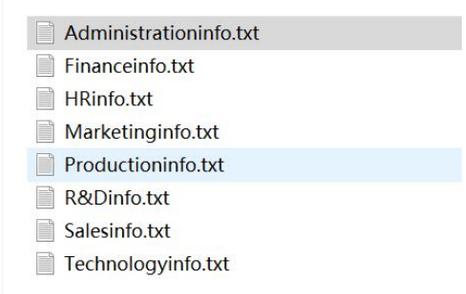
値
8183

Multiplex cursor
Read data single-threaded, compute multi-threaded

Index	ID	PID	DATE	QUANTITY	SID
1	1211	10075052	2010-01-01	84	10225
2	2474	10098045	2010-01-01	106	10591
3	10576	10093980	2010-01-01	53	10720
4	12938	10069598	2010-01-01	30	10483
5	15091	10067138	2010-01-01	104	11000

The employee information of each department is stored in one file separately.

Find New York State employees (multiple cursors consisting of multi-file cursors) from department staff information.



	A	B
1	=directory@p("E:/txt/employee_dept")	/Absolute path of employee information in various departments
2	=A1.(file(~).cursor@t()).mcursor()	/Generate a sequence of cursors and create multiple cursors
3	=A2.select(STATE=="New York").fetch()	/Fetch data using multiple cursors

A1~A3 results:

Index	Member
1	E:\txt\employee_dept\Administrationinfo.txt
2	E:\txt\employee_dept\Financeinfo.txt
3	E:\txt\employee_dept\HRinfo.txt
4	E:\txt\employee_dept\Marketinginfo.txt
5	E:\txt\employee_dept\Productioninfo.txt
6	E:\txt\employee_dept\R&Dinfo.txt
7	E:\txt\employee_dept\Salesinfo.txt
8	E:\txt\employee_dept\Technologyinfo.txt

Value
com.raqsoft.dm.cursor.MultipathCursors@72b83842

Index	EID	NAME	SURNAME	GENDER	STATE	BIRTHDAY	HIREDATE	DEPT	SALARY
1	2	Ashley	Wilson	F	New York	1980-07-19	2008-03-16	Finance	11000
2	220	Caleb	Smith	M	New York	1976-03-16	2008-01-01	Finance	7000
3	221	Sarah	Davis	F	New York	1982-09-04	2007-03-01	Finance	5000
4	229	Zachary	Taylor	M	New York	1984-09-12	2004-03-01	Finance	7000
5	180	Abigail	Smith	F	New York	1972-09-19	2007-05-01	HR	5000
6	25	Sarah	Davis	F	New York	1976-11-25	2006-11-25	Marketing	12000
7	46	Alexander	Johnson	M	New York	1978-08-20	2008-08-20	Marketing	10000
8	262	Mary	Jackson	F	New York	1987-06-11	2008-03-01	Marketing	6500
9	273	Emma	Williams	F	New York	1984-02-07	2007-11-01	Marketing	10000
10	446	Michael	Johnson	M	New York	1984-07-02	2005-08-01	Marketing	5000

03

Channel

The cursor can only perform one traversal, but sometimes we need to compute multiple data at the same time, so we need to press the data into the channel to reduce the number of traversals.

Example: In the lending business, 2200-01-01 is defined as overdue, with the number of overdue times/total times as the score, calculate the score of user and listing respectively.

	A	B
1	=file("E:/txt/user_repay_logs.csv").cursor@tc()	
2	cursor A1	=A2.groups(user_id;count(repay_date==date("2200-01-01"))/count(~):user_score)
3	cursor	=A3.groups(listing_id;count(repay_date==date("2200-01-01"))/count(~):listing_score)
…	…	
n	cursor	Can define other channels

A2~A3 results:

Index	user_id	user_score
1	1	0.0
2	2	0.0
3	3	0.05

Index	listing_id	listing_score
242110	242110	0.0
242111	242111	0.3333333333333...
242112	242112	0.0

Channel Returns Cursor

Calculating Median under big data conditions.

	A	B	
1	=file("E:/txt/CD5-0201FIC10202_2017.csv").cursor@tc()		/Create cursor
3	cursor A1	=A2.sortx(Value)	/Sort according to Value in Channel
4	cursor	=A3.total(count(~))	/Count the total number in another channel
5	=A2.skip((A3-1)\2)		/Skip the first half from the cursor after sorting
6	=A2.fetch@x(2-A3%2).avg(Value)		/Fetch the median

04

Ordered cursors

	A	B	C
1	=file("E:/txt/UseLogs.txt").cursor@t()		
2	for A1;UID	=A2.groups(UID,month(LOGIN):m_login;sum(SECOND):total_time)	
3		=to(12)\B2.(m_login)	
4		for B3	>B2.insert(B4,A2.UID,B4,0)
5		=@|B2.derive(if(m_login==1||total_time[-1]==0,null,total_time/total_time[-1]-1):raise_rate)	

Loop the UID ordered cursor, take out the same user each time, and calculate it. Set total_time to 0 if the user does not log in for a month. If it is the first month or the user didn't log in for the previous month, the growth rate is null.

B5 result:

Index	UID	m_login	total_time	raise_rate
1	10000001	1	1312198	(null)
2	10000001	2	984155	-0.24999504647926607
3	10000001	3	1241612	0.2616020850374179
4	10000001	4	1469796	0.18378044026636342
5	10000001	5	1551339	0.055479127715682974
6	10000001	6	1015243	-0.3455698593279741
7	10000001	7	1269041	0.2499874414302783
8	10000001	8	931809	-0.26573767120211245
9	10000001	9	1408993	0.5121049485463223
10	10000001	10	897755	-0.36283927599356425
11	10000001	11	1659301	0.8482782050782229
12	10000001	12	1488339	-0.10303254201618628
13	10000002	1	511107	(null)
14	10000002	2	700521	0.3705955895732205

	A	B
1	=file("E:/txt/Stock_Price.txt").cursor@t()	
2	=create(stockid,max_price_raise_rate)	
3	for A1;stockid	=A3.pmax(CL)
4		=A3.calc(B3,if(day(DT)==1,A3.CL,A3.CL/A3.CL[-1]-1))
5		=A2.insert(0,A3.stockid,B4)

Loop the ordered cursor of stockid, take out a set of data of the same stockid each time, find the position of the highest closing price, use this position to calculate the growth rate of the day, and fill the results into the table created in A2.

A3 result:

Index	stockid	max_price_raise_rate
1	1001	0.09691629955947145
2	1026	0.04958677685950419
3	1028	0.015592077538980176
4	1070	0.07007203667321549
5	1107	0.09358288770053469
6	1134	0.028019925280199365
7	1137	0.05385810460901075
8	1147	0.0014598540145984717
9	1206	14.01
10	1213	40.94

Organize the none-definite-rows mails

Mail message is as follows: Each mail begins with RECIPIENT, which is divided into three parts: recipient, sender and content. Please organize the mail into structured data.

RECIPIENT:730284595@xx.xx
SENDADDRESS:106383734@xx.xx
CONTENT: Harry Potter and the Sorcerer's Stone
CHAPTER ONE
THE BOY WHO LIVED
Mr. and Mrs. Dursley, of number four, Privet Drive, were proud to say that they were perfectly normal, thank you very much. They were the last people you'd expect to be involved in anything strange or mysterious, because they just didn't hold with such nonsense.
Mr. Dursley was the director of a firm called Grunnings, which made drills. He was a big, beefy man with hardly any neck, although he did have a very large mustache. Mrs. Dursley was thin and blonde and had nearly twice the usual amount of neck, which came in very useful as she spent so much of her time craning over garden fences, spying on the neighbors. The Dursleys had a small son called Dudley and in their opinion there was no finer boy anywhere.
The Dursleys had everything they wanted, but they also had a secret, and their greatest fear was that somebody would discover it. They didn't think they could bear it if anyone found out about the Potters Potter was Mrs Dursley's sister, but they hadn't met for several years; in fact, Mrs Dursley pretended she didn't have a sister, because her sister and her good-for-nothing husband were as unDursleyish as it was possible to be. The Dursleys shuddered to think what the neighbors would say if the Potters arrived in the street. The Dursleys knew that the Potters had a small son, too, but they had never even seen him. This boy was another good reason for keeping the Potters away; they didn't want Dudley mixing with a child like that
When Mr

RECIPIENT:717308235@xx.xx
SENDADDRESS:853474825@xx.xx
CONTENT:
When Mr Dursley woke up on the dull, gray Tuesday our story starts, there was nothing about the cloudy sky outside to suggest that strange and mysterious things would soon be happening all over the country Dursley hummed as he picked out his most boring tie for work, and Mrs Dursley gossiped away happily as she wrestled a screaming Dudley into his high chair
None of them noticed a large, tawny owl flutter past the window
At half past eight, Mr

RECIPIENT:206817811@xx.xx
SENDADDRESS:839801895@xx.xx
CONTENT:

Organize the none-definite-rows mails

Ordered cursors

Index	RECIPIENT	SENDADDRESS	CONTENT
1	730284595@xx.xx	106383734@xx.xx	Harry Potter an...
2	717308235@xx.xx	853474825@xx.xx	When MrDursl...
3	206817811@xx.xx	839801895@xx.xx	At half past eig...
4	983837696@xx.xx	1034188652@xx.xx	He got into his car ...
5	860688317@xx.xx	973356725@xx.xx	They were whisperi...
6	239022673@xx.xx	579899833@xx.xx	He didn't know why...
7	746677348@xx.xx	1034632680@xx.xx	He looked back at t...
8	264687557@xx.xx	351534290@xx.xx	It was a few secon...
9	352747334@xx.xx	348118929@xx.xx	Going to be any mo...
10	533754054@xx.xx	1057123562@xx.xx	"Owls... shooting st...

A6 result:

	A	B	
1	=file("E:/txt/email.txt").cursor()		
2	=create(RECIPIENT,SENDADDRESS,CONTENT)		
3	for A1;_1.split(":")(1)=="RECIPIENT"	=A3.(_1)(1).split(":")	
4		=A3.(_1)(3).split(":")	
		=B5.m(2:).concat(":")+A3.(_1).m(4:).concat()	
		=A2.insert(0,B3(2),B4(2),B6)	
		if A2.len()==500	=file("E:/txt/email_pre.txt").export@at(A2)
			>A2=create(RECIPIENT,SENDADDRESS,CONTENT)
5	=file("E:/txt/email_pre.txt").export@at(A2)		
6	=file(“E:/txt/email_pre.txt”).cursor@t().fetch@x(10)	/Read in to view the first 10 rows of results	

Each email begins with RECIPENT. Loop the cursor, When the first part of the message is "RECIPIENT “ before “: ” , the cursor starts reading until the next time it meets "RECIPENT". In this way, each loop represents an e-mail, which can be processed in the for loop. When the number of mails reaches 500, the data is exported and the table is emptied.

05

Program Cursor

The stepwise cursor introduced earlier is very convenient, but sometimes it is more complex to calculate at a certain step, and a single function can not complete the calculation. At this time, using the program cursor to return the intermediate results makes it easier to understand and maintain in the future.

Example: Organize the user information and screen out the users in New York State. The user information is as follows:

```
1	10308583	F	2007-02-07
2018-03-11	Michigan	10308583_***@mail.xxx	acappella
2	10902344	F	2011-02-17
2018-02-25	Ohio	10902344_***@mail.xxx	big-beat
3	10550284	F	2010-12-26
2018-10-23	Illinois	10550284_***@mail.xxx	acappella
4	10719361	M	2003-01-15
2018-07-01	Missouri	10719361_***@mail.xxx	new age
5	10329553	F	2015-09-21
2018-08-08	Texas	10329553_***@mail.xxx	ambient
6	10321518	F	2015-11-28
2018-07-07	Pennsylvania	10321518_***@mail.xxx	trip-hop
```

User information is divided into two lines. User information needs to be organized into structured data first.

Organize the user information first, then export it to a file, and then use cursor to read and find out the users of New York State.

	A	B
1	=file("E:/txt/user_info.txt").cursor()	
2	for A1,5000*2	=A2.group((#-1)\2)
3		=B2.(~.(~.array()).conj())
4		=B3.new(~(1):ID,~(2):UID,~(3):GENDER,~(4):RDATE,~(5):LDATE,~(6):STATE,~(7):EMAIL,~(8):STYLE)
5		=file("E:/txt/user_info_pre.txt").export@at(B4)
6	=file("E:/txt/user_info_pre.txt").cursor@t()	
7	=A6.select(STATE=="New York")	
8	=A7.fetch()	

Loop the cursor to organize the data into structured data, which is exported in batches and appended to the same file. Create cursor, attach the select operation, and get the final result.

At the end of the last loop, B2-B4 results:

B2

Index	Member
1	[[95001,10085776,M, ...],[2018-10-25,Florida,10085776_***@mail.xxx, ...]]
2	[[9500 10522585,F, ...],[2018-11-16,Texas,10522585_***@mail.xxx, ...]]
3	[[950 0073788,M, ...],[2018-07-31,Florida,10073788_***@mail.xxx, ...]]

Index	_1	_2	_3	_4
1	95001	10085776	M	2006-01-18
2	2018-10-25	Florida	10085776_***...	big-beat

B3

Index	Member
1	[95001,10085776,M, ...]
2	[95002,10522585,F, ...]
3	[95003,10073788,M, ...]

B4

Index	ID	UID	GENDER	RDATE	LDATE	STATE	EMAIL	STYLE
1	95001	10085776	M	2006-01-18	2018-10-25	Florida	10085776...	big-beat
2	95002	10522585	F	2007-03-03	2018-11-16	Texas	10522585...	brit-hop
3	95003	10073788	M	2005-07-03	2018-07-31	Florida	10073788...	jungle

Data Processing Program （prepare_user_data）

Program cursor

	A	B
1	=file("E:/txt/user_info.txt").cursor()	
2	for A1,5000*2	=A2.group((#-1)\2)
3		=B2.(~.(~.array()).conj())
4		=B3.new(~(1):ID,~(2):UID,~(3):GENDER,~(4):RDATE,~(5):LDATE,~(6):STATE,~(7):EMAIL,~(8):STYLE)
5		return B4

B4 result:

Index	ID	UID	GENDER	RDATE	LDATE	STATE	EMAIL	STYLE
1	95001	10085776	M	2006-01-18	2018-10-25	Florida	10085776...	big-beat
2	95002	10522585	F	2007-03-03	2018-11-16	Texas	10522585...	brit-hop
3	95003	10073788	M	2005-07-03	2018-07-31	Florida	10073788...	jungle

	A	B
1	=cursor("E:/esproc_test/cursor/prepare_user_data.dfx")	/Call data processing subroutine
2	=A1.select(STATE=="New York")	/Select users in New York state
3	=A2.fetch()	/Fetch the data

A3 result:

Index	ID	UID	GENDER	RDATE	LDATE	STATE	EMAIL	STYLE
1	1	10182061	M	2013-09-26	2018-03-11	New York	10182061...	big-beat
2	20	10325095	M	2010-03-27	2018-09-26	New York	10325095...	acappella
3	29	10533676	M	2011-02-01	2018-03-28	New York	10533676...	trip-hop
4	45	10023684	F	2005-05-12	2018-08-23	New York	10023684...	brit-hop
5	54	10052401	M	2011-08-25	2018-08-14	New York	10052401...	ambient

You can also use func to call in the same script

	A	B	C
1	=file("E:/txt/user_info.txt").cursor()		
2	func	for A2,5000*2	=B2.group((#-1)\2)
3			=C2.(~.(~.array()).conj())
4			=C3.new(~(1):ID,~(2):UID,~(3):GENDER,~(4):RDATE,~(5):LDATE,~(6):STATE,~(7):EMAIL,~(8):STYLE)
5			return C4
6	=cursor@c(A2,A1)		
7	=A7.select(STATE=="New York")		
8	=A8.fetch()		

A8 result:

Index	ID	UID	GENDER	RDATE	LDATE	STATE	EMAIL	STYLE
1	1	10182061	M	2013-09-26	2018-03-11	New York	10182061...	big-beat
2	20	10325095	M	2010-03-27	2018-09-26	New York	10325095...	acappella
3	29	10533676	M	2011-02-01	2018-03-28	New York	10533676...	trip-hop
4	45	10023684	F	2005-05-12	2018-08-23	New York	10023684...	brit-hop
5	54	10052401	M	2011-08-25	2018-08-14	New York	10052401...	ambient

Example: The order table has been sorted by time. It is necessary to remove duplication of data by date and product, and then count the number of records (groupx).

	A	B
1	=now()	
2	=file("E:/txt/PRODUCT_SALE.txt").cursor@t()	/Create cursor
3	=A2.groupx(date(DATE),PID)	/Remove duplication
4	=A3.skip()	/Count the number of records
5	=interval@ms(A1,now())	

A4, A5 results:

Value
9849397

Value
55647

Data is ordered by DATE, otherwise it can't be done like this.

	A	B	C
1	=now()		
2	=file("E:/txt/PRODUCT_SALE.txt").cursor@t()		
3	func	for A3;DATE	=B3.id(PID)
4			return C3
5	=cursor@c(A3,A2)		
6	=A5.skip()		
7	=interval@ms(A1,now())		

A3, A4 results:

Value
9849397

Value
12767

THANKS