



Scalable Performance Data Server®

Performance Session

SAS Institute Japan Ltd.
Katsumi Yamamoto
Mitsutoshi Hattori

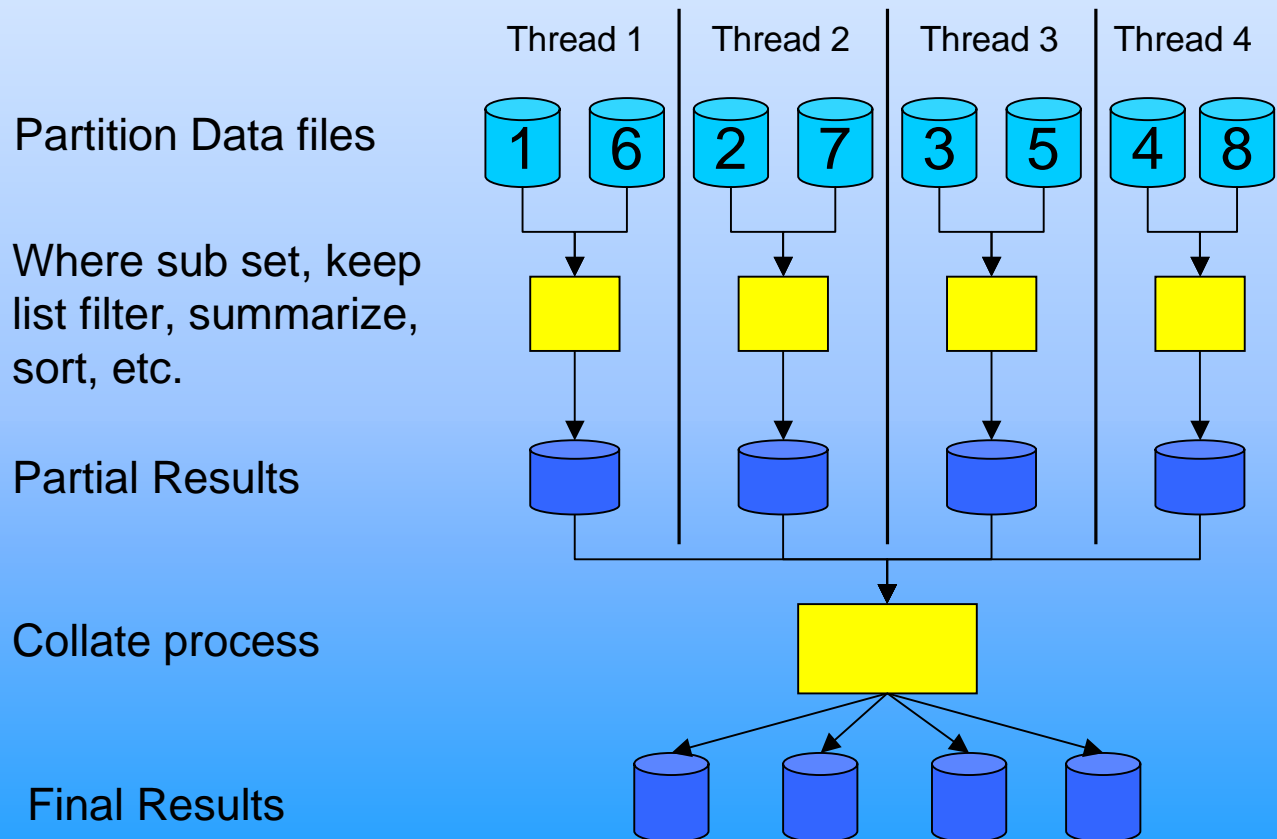


SPDServer 並列処理

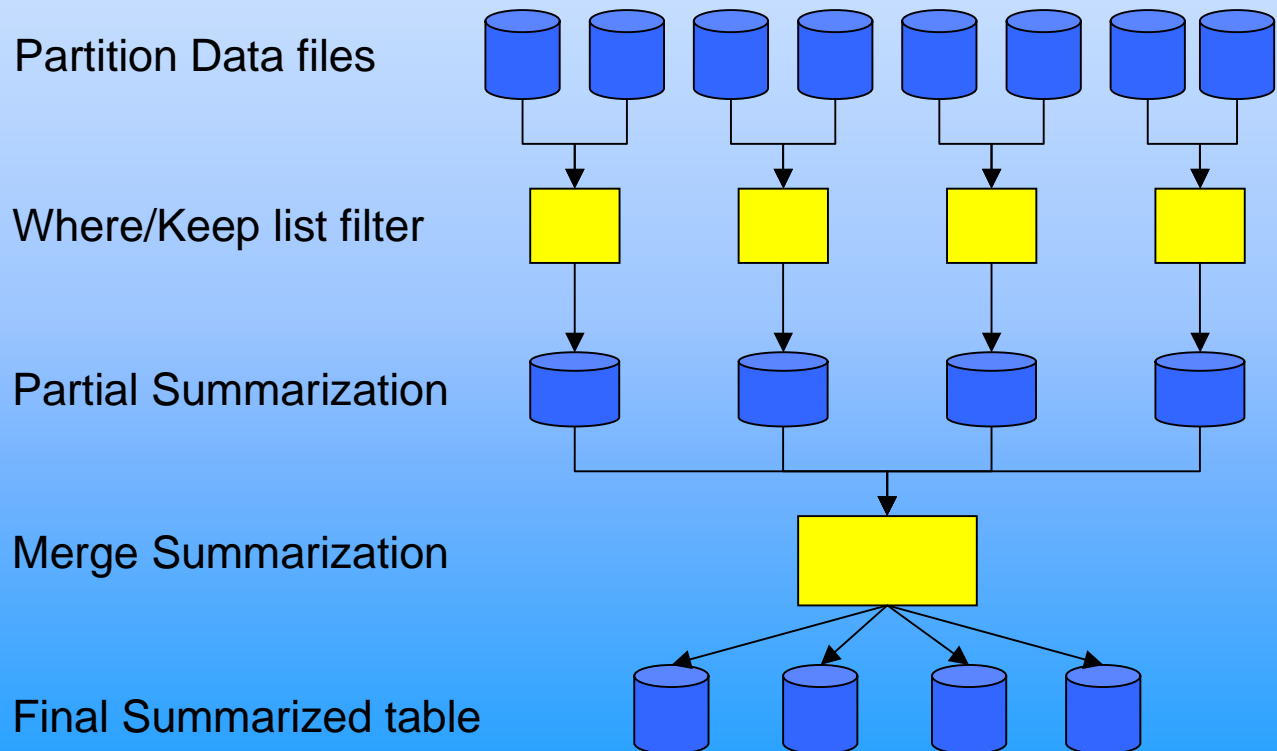
- **Full Table Scans**
- **Implicit and Explicit Parallel Sort**
- **Parallel Group By**
- **Index Evaluation**
- **Index Creation**

Parallel Processing

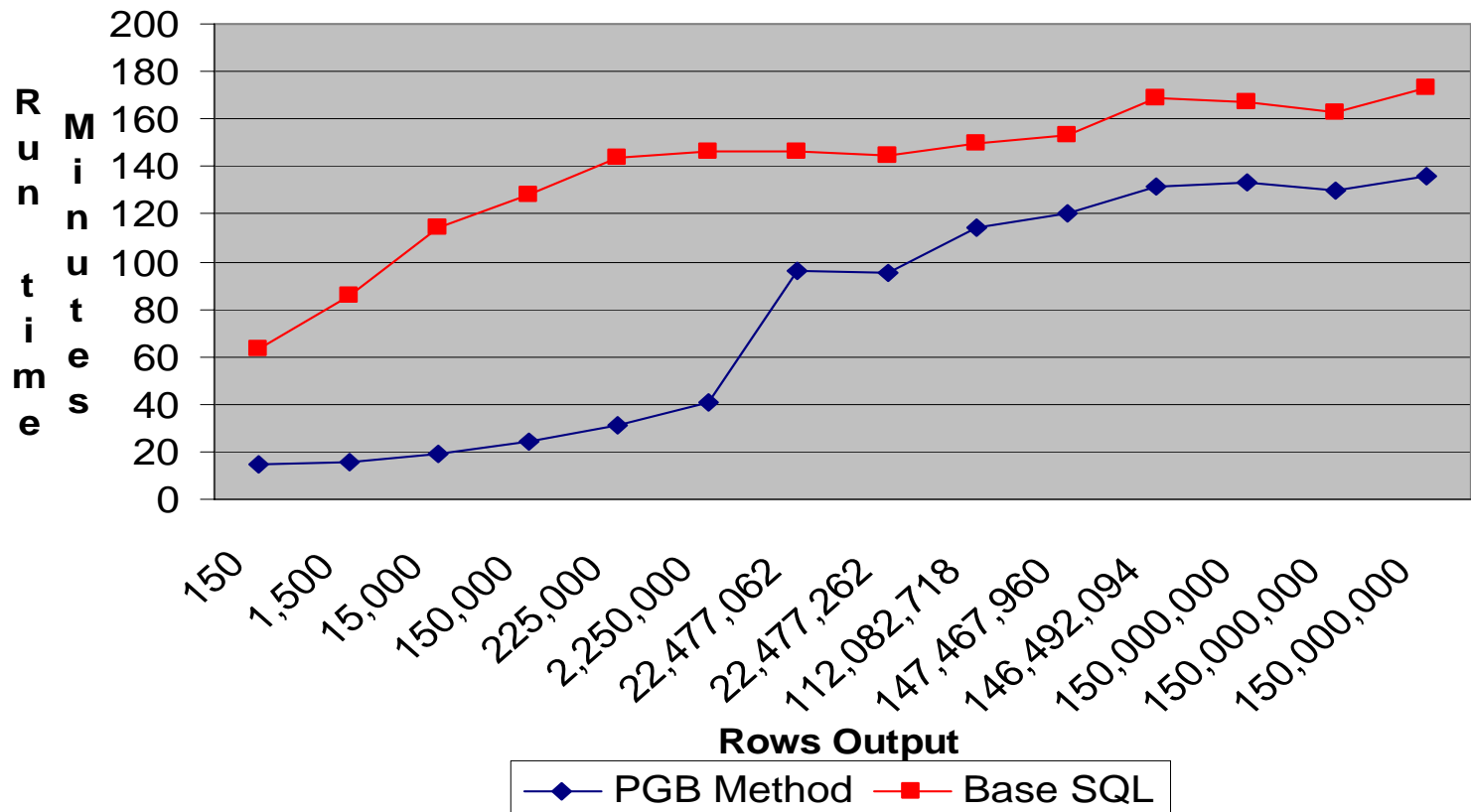
4 Threads



Parallel Group By

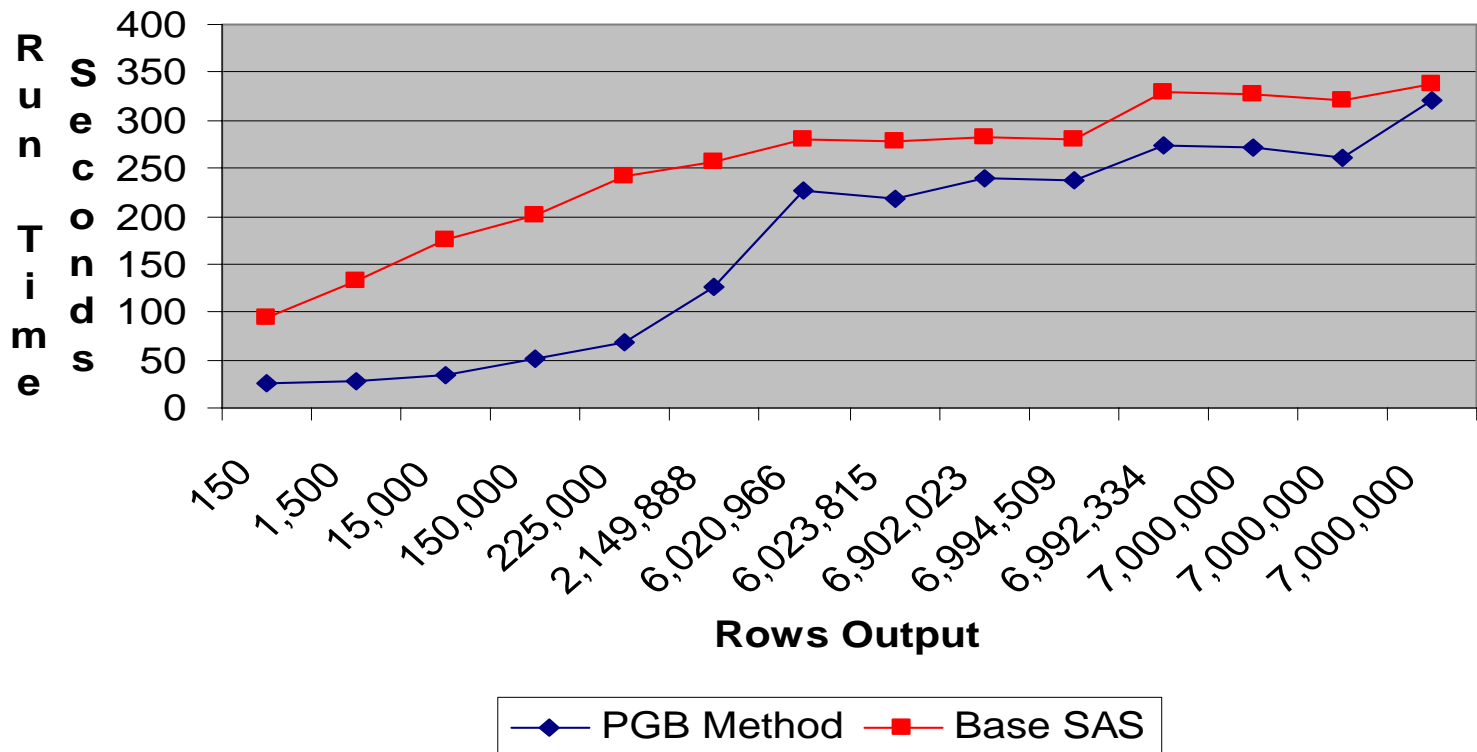


PGB SQL Method vs. Base SQL 150mill row, 12gig table



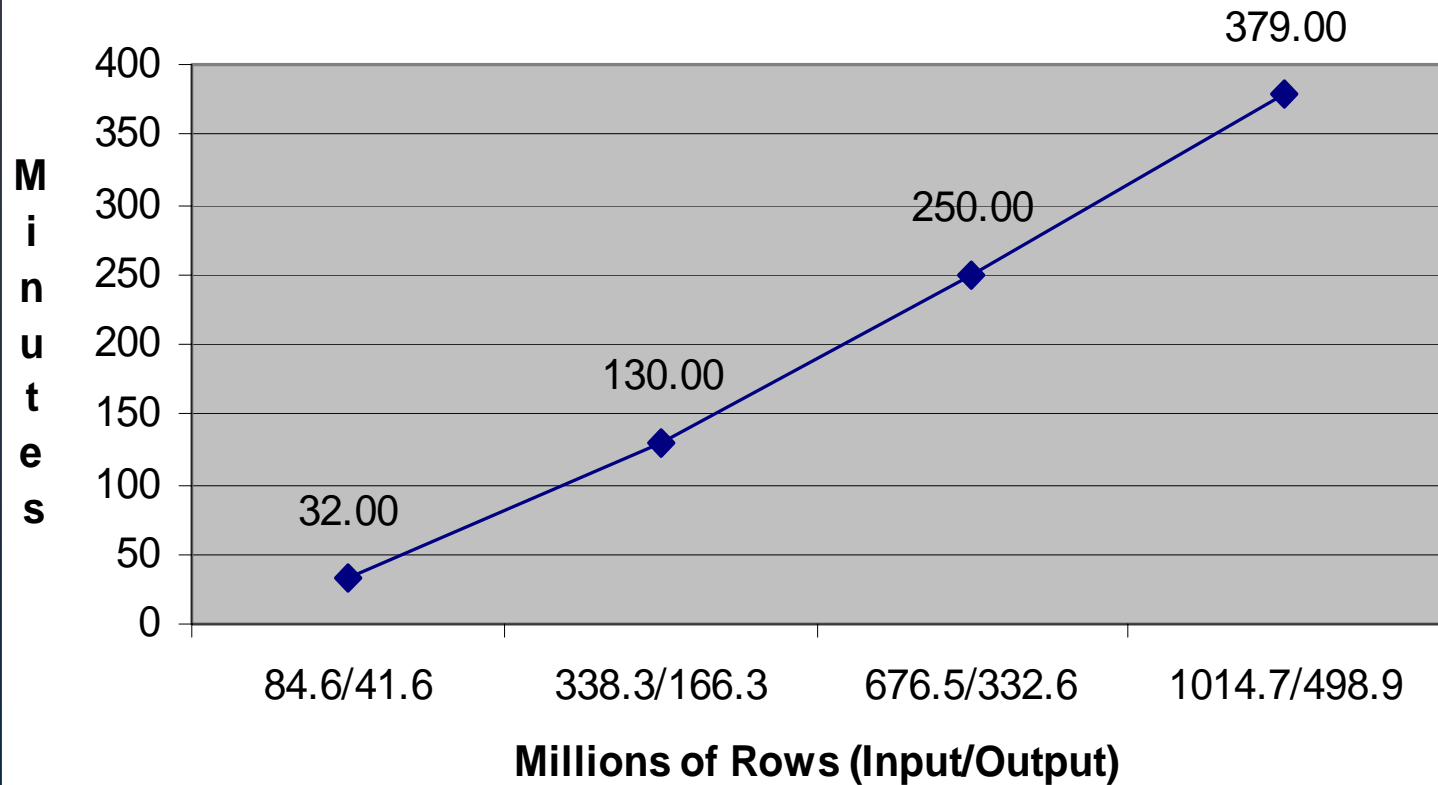
PGB . . . Parallel Group By

PGB Method vs. Base SQL 7mill row, 500 meg table

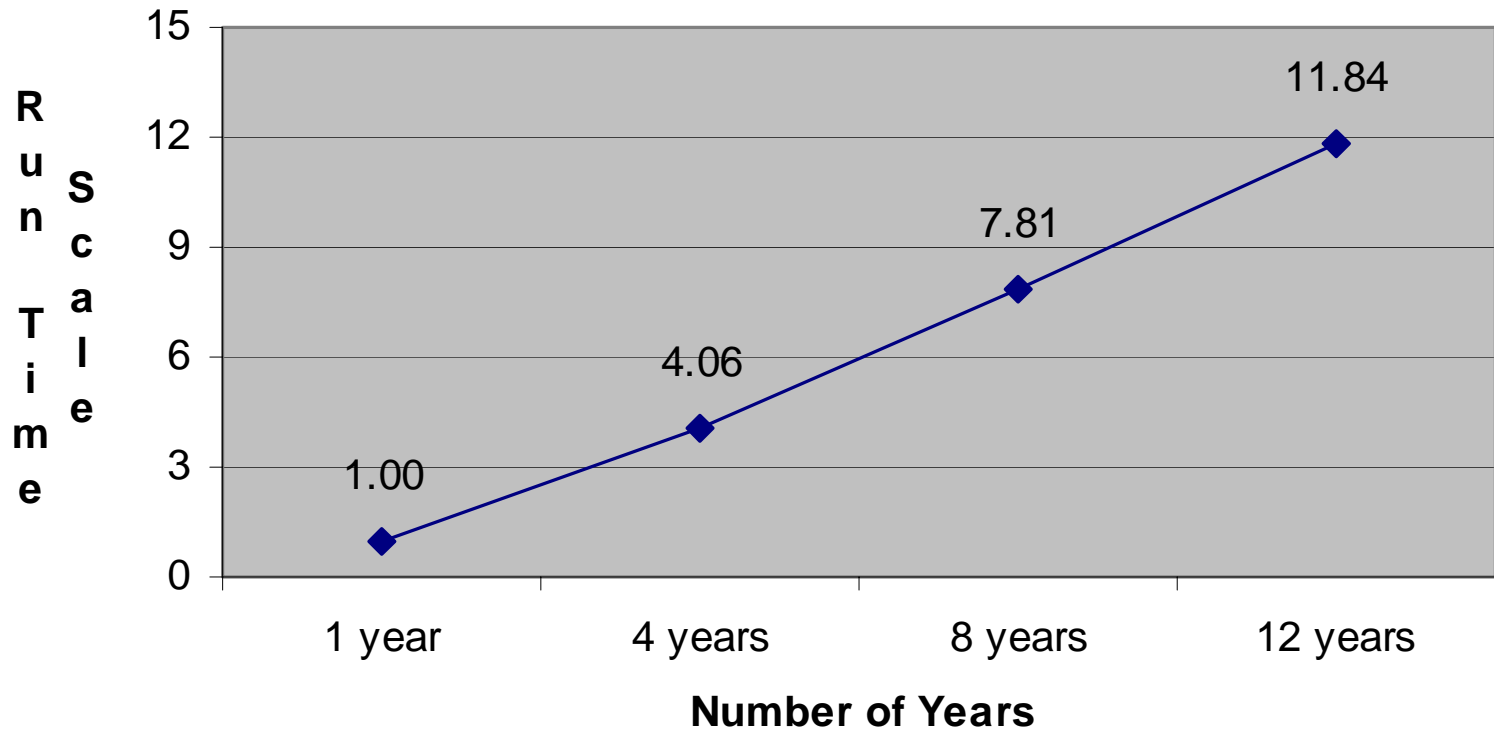


PGB . . . Parallel Group By

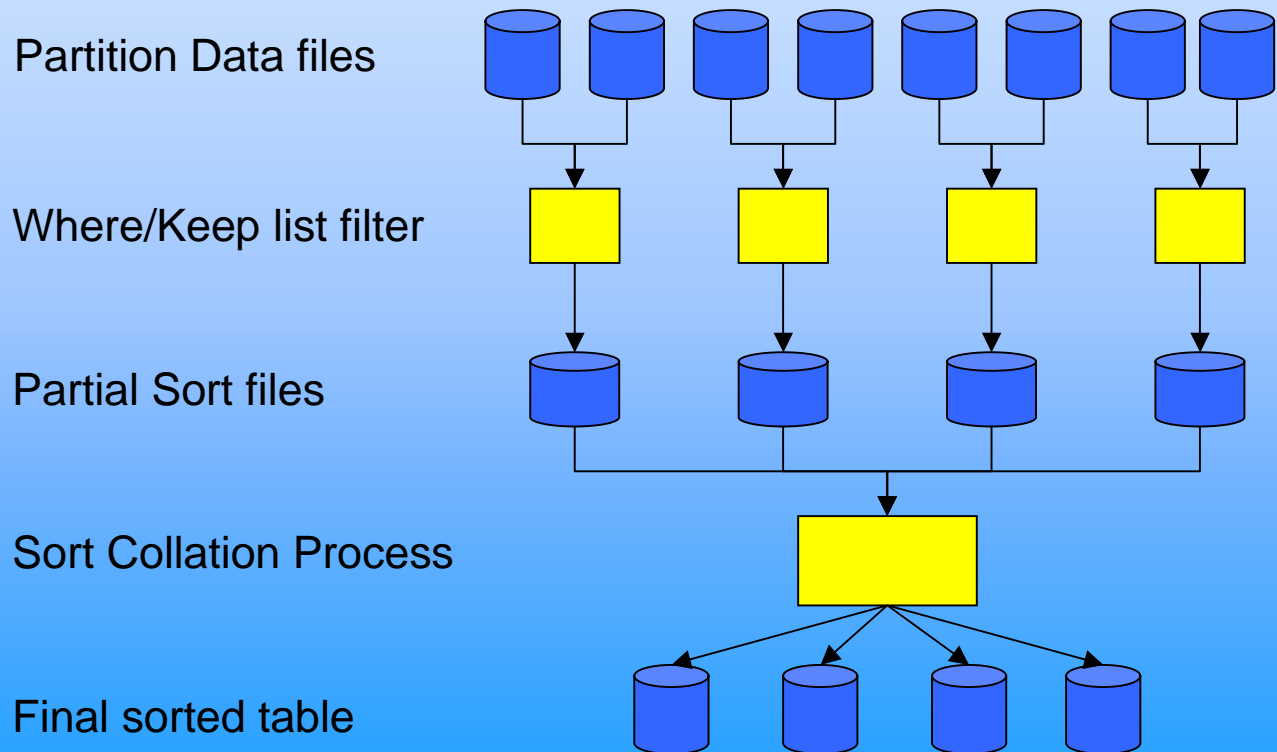
Parallel Group By Scalability



Parallel Group By Scalability



並列 Sort



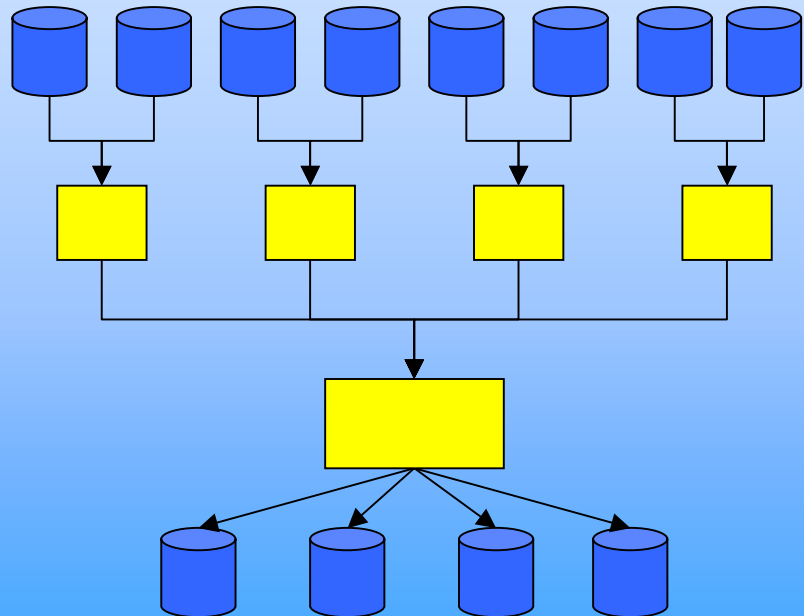
並列検索

Partition Data files

Where/Keep list filter

Write Process

* Final Results



* Final results may not be in input order

Interleaving tables

- Base SAS Example:

```
libname mylib '/data1/mylib' ;

proc sort data=mylib.my_data_table1
    out=work.temporary_a ;
    by column1 column2 ;
run ;

proc sort data=mylib.my_data_table2
    out=work.temporary_b ;
    by column1 column2 ;
run ;

Data mylib.new_table ;
    set work.temporary_a
        work.temporary_b ;
    by column1 column2 ;
    ...more lines of code...
    output ;
run ;
```

- SPD Server Example:

```
libname mylib sasspds 'domain' server=myserv.5130
    user='userid' prompt=yes ;
```

```
Data mylib.new_table ;
    set mylib.my_data_table1
        mylib.my_data_table2 ;
    by column1 column2 ;
    ...more lines of code...
    output ;
run ;
```

More SAS statements.

Merging tables

- Base SAS Example:

```
libname mylib '/data1/mylib' ;
```

```
proc sort data=mylib.my_data_table1  
    out=work.temporary_a ;  
    by column1 column2 ;  
run ;
```

```
proc sort data=mylib.my_data_table2  
    out=work.temporary_b ;  
    by column1 column2 ;  
run ;
```

```
Data mylib.new_table ;  
    merge work.temporary_a  
          work.temporary_b ;  
    by column1 column2 ;  
    ...more lines of code...  
    output ;  
run ;
```

- SPD Server Example:

- libname mylib sasspds
 'domain' server=myserv.5130
 user='userid' prompt=yes ;
- Data mylib.new_table ;
 - merge mylib.my_data_table1
 - mylib.my_data_table2 ;
 - by column1 column2 ;
 - ...more lines of code...
 - output ;
- run ;

並列Sortアルゴリズム

- SPD Server Code:

```
libname mylib sasspds 'domain'  
server=myserv.5130 user='userid' prompt=yes;
```

```
Data mylib.new_table ;
```

```
Set mylib.my_data_table
```

```
(keep=column_a column_b etc.
```

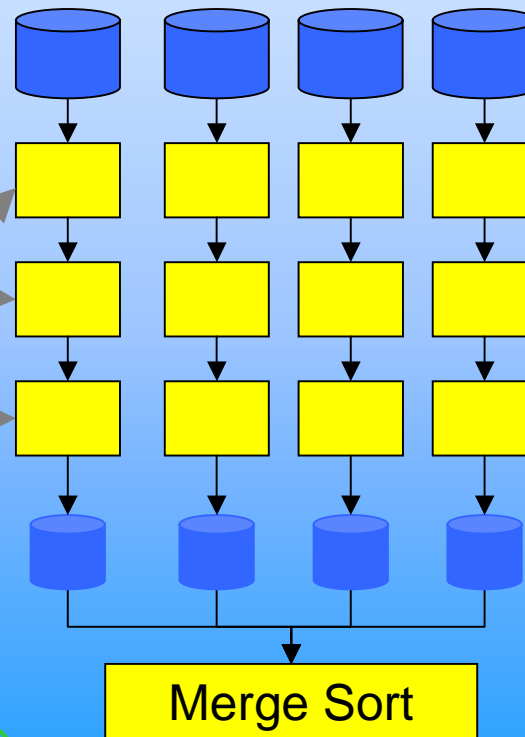
```
where=(column_a="constant')) ;
```

```
by column_a column_b;
```

```
...more lines of code...;
```

```
output ;
```

```
run ;
```



Data Partitions

Where Filter

Keep List Filter

Sorting

Partial Results

Merge Sort



Log from SPD Server:

```
1
2      libname spds301 sasspds 'spds301' host='zztop' serv='5150' user='anonymous' unixdomain=YES netcomp=NO ;
NOTE: Libref SPDS301 was successfully assigned as follows:
      Engine:      SASSPDS
      Physical Name: :29088/DATA04/spds301/
3
4      proc sort data=spds301.wloan_mrtgfct out=spds301.wloan_mrtgfct_sorted2 ;
5          by product_type_key rpt_date_key_nbr ;
6      run ;
```

NOTE: Sorting was performed by the data source.

NOTE: There were 26716310 observations read from the data set SPDS301.WLOAN_MRTGFCT.

NOTE: The data set SPDS301.WLOAN_MRTGFCT_SORTED2 has 26716310 observations and 49 variables.

NOTE: PROCEDURE SORT used:

real time	16:51.16
cpu time	8:56.77*

* CPU utilization not accurate because SPD Server is client server.

Log from Base SAS:

```
1
2      libname hud '/GENSTOR/hud' ;
NOTE: Libref HUD was successfully assigned as follows:
      Engine:      V8
      Physical Name: /GENSTOR/hud
3
4      proc sort data=hud.wloan_mrtgfct out= hud.wloan_mrtgfct_sorted ;
5          by product_type_key rpt_date_key_nbr ;
6      run ;
```

NOTE: There were 26716310 observations read from the data set HUD.WLOAN_MRTGFCT.

NOTE: The data set HUD.WLOAN_MRTGFCT_SORTED has 26716310 observations and 49 variables.

NOTE: PROCEDURE SORT used:

real time	31:09.13
cpu time	29:41.69



Hybrid Index

- Segmented for parallel index evaluation
- One index handles all needs
 - Joins
 - Where clause sub setting
 - Tables scans with by statement
- Incorporates both Bitmap and B-Tree Technology
 - Built in intelligence determines index type created

Hybrid Index

Data table

Row	column_a	column_b
1	A	X
2	A	Y
3	A	X
4	A	W
.... More Rows		
8189	C	W
8190	B	Z
8191	B	Y
8192	B	Z
8193	B	W
8194	B	X
8195	C	Z
8196	C	Y
.... More Rows....		
16381	D	X
16382	D	Y
16383	E	Z
16384	F	W
....More Rows....		

I
n
d
e
x

M
e
t
a

d
a
t
a

Index combines **bitmap** and **B-tree** technology

Rows 1 through 8192 (Index Segments 1)

Value	column_a	Value	column_b
A	1111...0000	W	0001...1000
B	0000...0111	X	1010...0000
C	8189	Y	0100...0010
		Z	0000...0101

Rows 8193 through 16384 (Index Segments 2)

Value	column_a	Value	column_b
B	1100...0000	W	1000...0001
C	0011...0000	X	0100...1000
D	0000...1100	Y	0011...0110
E	16383		
F	16384		

... More index segments...

Sub Setting Where

clause • Indexes

column_a

column_b

column_c

• Example 1:

Where column_a in ('A','B');

Index
Meta
data

Segment 1

Segment 1

Segment 1

Segment 2

Segment 2

Segment 2

Segment 3

Segment 3

Segment 3

Segment 4

Segment 4

Segment 4

Segment 5

Segment 5

Segment 5

Segment 6

Segment 6

Segment 6

Segment 7

Segment 7

Segment 7

Segment 8

Segment 8

Segment 8

.....

More Segments

.....

Sub Setting Where

clause • Indexes

column_a

column_b

column_c

• Example 2:

Where **column_a** in ('A','B','C')
and **column_b** in ('R','S','T')
and **column_c** in (1,2,5,7,8) ;

Index
Meta
data

Segment 1	Segment 1	Segment 1
Segment 2	Segment 2	Segment 2
Segment 3	Segment 3	Segment 3
Segment 4	Segment 4	Segment 4
Segment 5	Segment 5	Segment 5
Segment 6	Segment 6	Segment 6
Segment 7	Segment 7	Segment 7
Segment 8	Segment 8	Segment 8
.....	More Segments

Sub Setting Where

clause • Indexes

column_a

column_b

column_c

• Example 3:

Where column_a in ('H','J','L')
and column_b in ('R','S','T')
and column_c in (11,19,61) ;

Index
Meta
data

Segment 1	Segment 1	Segment 1
Segment 2	Segment 2	Segment 2
Segment 3	Segment 3	Segment 3
Segment 4	Segment 4	Segment 4
Segment 5	Segment 5	Segment 5
Segment 6	Segment 6	Segment 6
Segment 7	Segment 7	Segment 7
Segment 8	Segment 8	Segment 8
.....	More Segments

Sub Setting Where

clause • Indexes

column_a

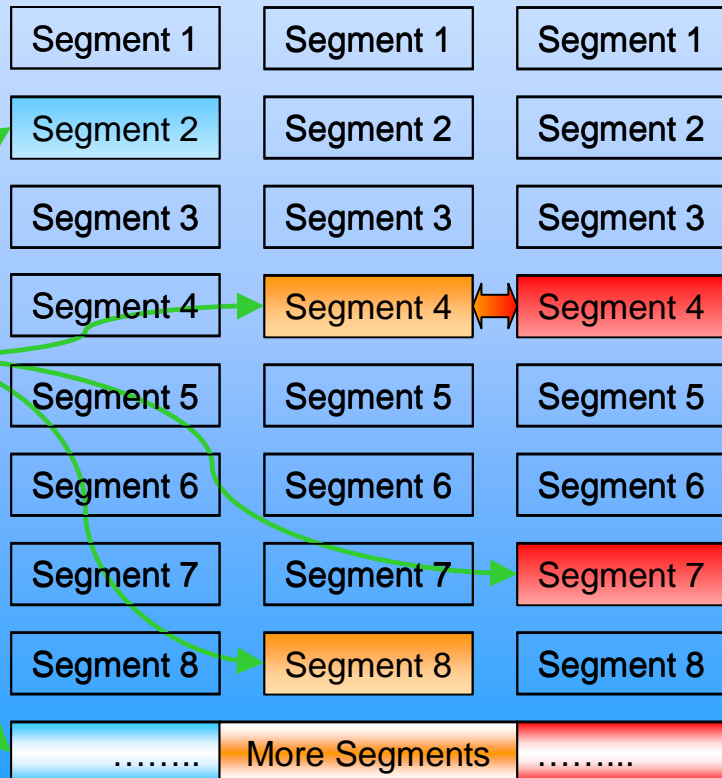
column_b

column_c

• Example 4:

Where column_a in ('D','G')
or column_b in ('N','U')
or column_c in (12,15);

Index
Meta
data

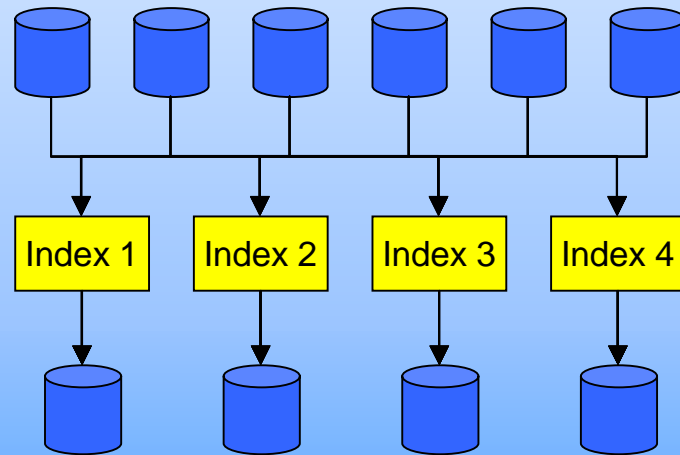


並列 Index 作成

Partition Data files

Index Creation Threads

4 Indexes Created





Log from Base SAS:

9 proc datasets library=basesas ;

The SAS System

11:13 Monday, July 9, 2001

-----Directory-----

Libref: HUD
Engine: V8
Physical Name: /GENSTOR/hud
File Name: /GENSTOR/hud
Inode Number: 3050
Access Permission: rwxrwxr-x
Owner Name: spdsmgr
File Size (bytes): 8192

#	Name	Memtype	File Size	Last Modified
1	PROD_SUM	DATA	16384	09JUL2001:00:07:17
2	WLOAN_MRTGFCT	DATA	10547503104	28JUN2001:16:52:10
3	WLOAN_MRTGFCT_SORTED	DATA	10547535872	09JUL2001:11:13:57

10 modify wloan_mrtgfct_sorted ;

11 index create agency_nm mortgage_age whole_loan_key_nbr

12 rpt_date_key_nbr product_type_key ;

NOTE: Simple index agency_nm has been defined.

NOTE: Simple index mortgage_age has been defined.

NOTE: Simple index whole_loan_key_nbr has been defined.

NOTE: Simple index rpt_date_key_nbr has been defined.

NOTE: Simple index product_type_key has been defined.

13 quit ;

NOTE: PROCEDURE DATASETS used:

real time 36:45.50

cpu time 35:42.48

Log from SPD Server SAS:

```
12 proc datasets library=spds301 nodetails ;
```

The SAS System

12:13 Monday, July 9, 2001

-----Directory-----

Libref: SPDS301

Engine: SASSPDS

Physical Name: :28933/DATA04/spds301/

Local Host Name: zztop

Local Host IP addr: 10.6.2.43

Server Hostname: N/A

Server IP addr: 10.6.2.43

Server Portno: 55655

Free Space (Kbytes): 375710649

Metapath: '/DATA04/spds301/'

Indexpath: '/IDX1/spds301/'

Datapath: '/DATA01/spds301/' '/DATA03/spds301/' '/DATA02/spds301/' '/DATA04/spds301/'

'/DATA05/spds301/' '/DATA06/spds301/' '/DATA07/spds301/' '/DATA08/spds301/'

'/DATA09/spds301/'

```
13 modify wloan mrtgfct sorted;
```

```
14 index create agency_nm mortgage_age whole_loan_key_nbr
```

```
15 rpt_date key_nbr product_type key;
```

NOTE: Simple index agency_nm has been defined.

NOTE: Simple index mortgage_age has been defined.

NOTE: Simple index whole_loan_key_nbr has been defined.

NOTE: Simple index rpt_date_key_nbr has been defined.

NOTE: Simple index `product_type_key` has been defined.

```
16      quit ;
```

NOTE: PROCEDURE DATASETS used:

```
real time      8:24.54
```

```
cpu time      0.03 seconds *
```

* CPU utilization not accurate because SPD Server is client server.