

Partitioned Tables in DB2 for z/OS V8



Peter G Backlund
Peter Backlund DB2-Konsult AB
BacklundDB2@telia.com

Disclaimer



- I work as an independent DB2 consultant
- Who is independent?
- The opinions and statements made in this presentation are entirely my responsibility

Contents



■ C - Tablespaces in General

■ D - Partitioned Tables

C.1 – Tablespaces in V8



ALTER TABLESPACE is unchanged – almost

LOCKPART YES is default

Automatic Space Management

Changes to the REORG Utility

Partitioning

C.2 - Automatic Space Management



PRIQTY -1 or omitted

- TSQTY, IXQTY (DSNZPARM)
- Default 1 CYL (10 CYL for LOB)

SECQTY 0

- No secondary extent

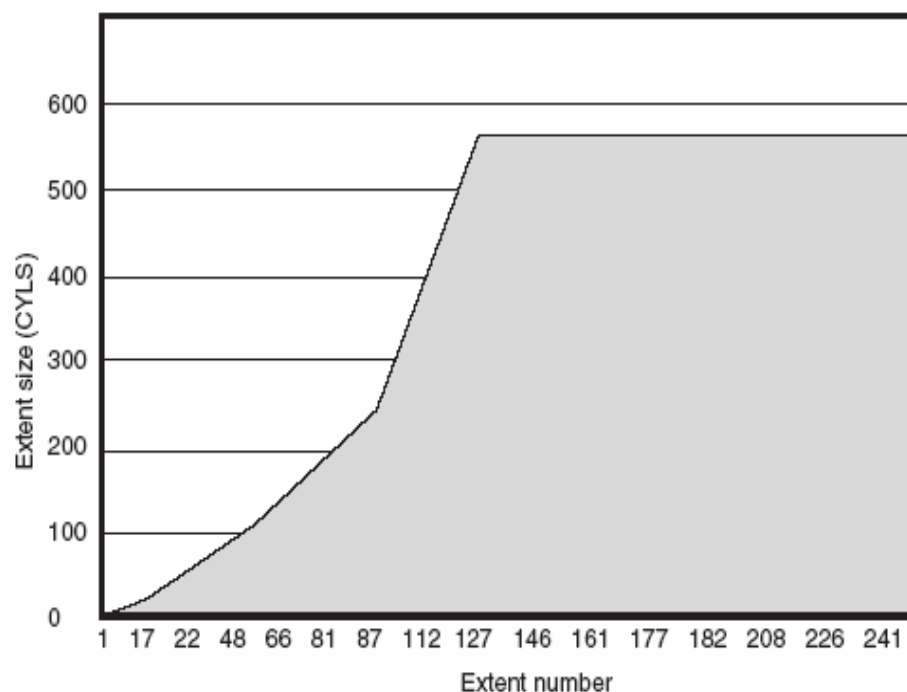
SECQTY -1 or omitted

- Intricate formulas, sliding scale

C.3 - Sliding Scale



Sliding scale for a 64-GB data set



C.4 - Tablespaces - Reorg



REORG *catalog tablespace* SHRLEVEL REFERENCE

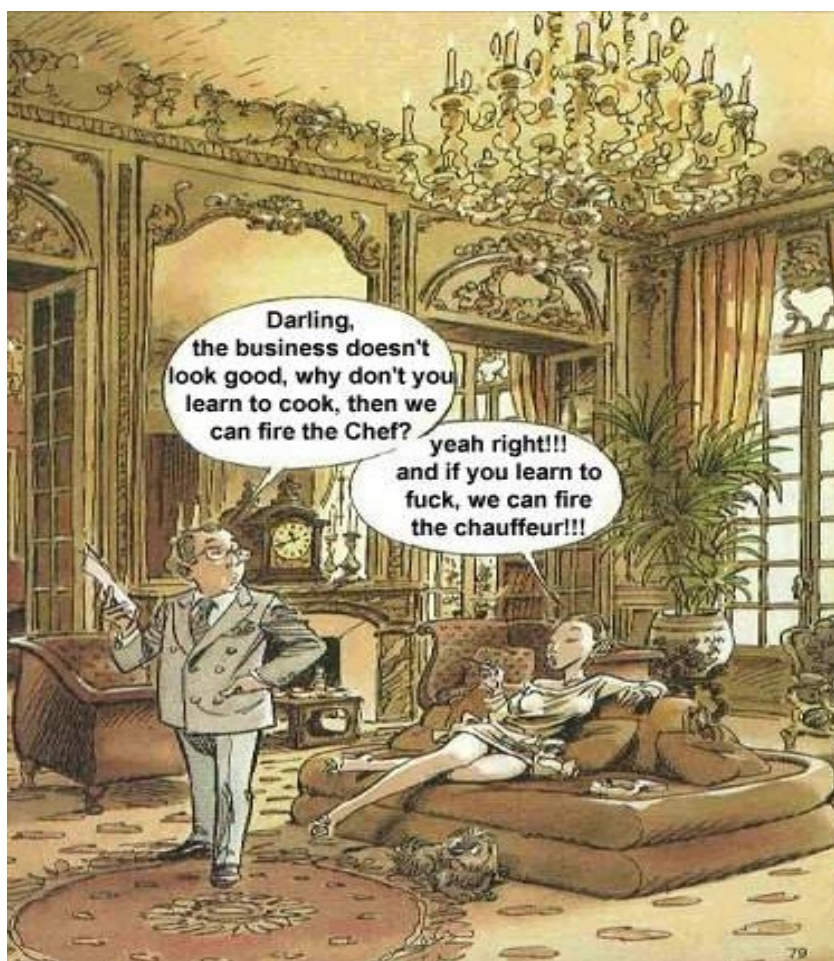
REORG ... DISCARD SHRLEVEL CHANGE

REORG creates new Clsize

If there is no Cluster Index, the "first" index is used

REORG ... SCOPE PENDING

REORG ... REBALANCE - later



D.1 - Partitioned Tables – V8



This is one of the areas with big changes in DB2 for z/OS V8

Instead of Index Partitioning, we should use Table Partitioning

- Separate Clustering from Partitioning
- Add Partition
- Rotate Partition

D.2 - Index Partitioning



Create Tablespace *tspart* Numparts 4

Create Table *tbpart(shopno,state)* In *tspart* - **Incomplete!**

Create Index *ixpart* On *tbpart(shopno,state)*

Cluster (Part 1 Values(199,'ZZ')
 , Part 2 Values(299)
 , Part 3 Values(399,'ZZ')
 , Part 4 Values(499,'ZZ'))

Partitioning Index

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

Table

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

Partitioning Index

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

Table

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

NPI

AZ
AZ
AZ
AZ
AZ
CA
CA
CA
CA
CA
NY
NY
NY
NY

Partitioning Index

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

Table

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

NPI - Unique

AZ	101
AZ	105
AZ	206
AZ	303
AZ	402
CA	202
CA	204
CA	206
CA	404
NY	103
NY	206
NY	301
NY	303

D.3 - Table Partitioning



Create Tablespace *tspart* Numparts 4

Create Table *tbpart(shopno,state)* In *tspart*

Partition By (*shopno*)

(Partition 1 Ending At (199)

,Partition 2 Ending At (299)

,Partition 3 Ending At (399)

,Partition 4 Ending At (499))

No Index is needed!

Last partition limit enforced!

Table

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

Table

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

Where State = 'NY'

**Where State = 'NY'
And Shopno = 303**

**Where State = 'NY'
And Shopno = ?**

D.4 - Table Partitioning - Indexes



An index on a partitioned table can be

- Partitioning
- Secondary

If the index key columns match or comprise a superset of the columns of the table's partitioning key columns (same ASC/DESC),

then the index is a partitioning index;

else a secondary index

Partitioning Index

101
103
105
202
204
206
206
206
206
301
303
303
402
404

Table

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

Partitioning Index

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

Table

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

Partitioning Index

101
103
105
202
204
206
206
206
301
303
303
402
404

Table

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

Secondary Index

AZ	101
AZ	105
AZ	206
AZ	303
AZ	402
CA	202
CA	204
CA	206
CA	404
NY	103
NY	206
NY	301
NY	303

D.5 - Table Partitioning - Indexes



An index on a partitioned table can be

- Partitioned
- Nonpartitioned

This is determined by the keyword PARTITIONED on Create Index

Partitioned Partitioning Index

101
103
105
202
204
206
206
206
301
303
303
402
404

Table

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

Data Partitioned Secondary Index

AZ	101
AZ	105
NY	103
AZ	206
CA	202
CA	204
CA	206
NY	206
AZ	303
NY	301
NY	303
AZ	402
CA	404

Table

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

DPSI

AZ
AZ
NY
AZ
CA
CA
CA
NY
AZ
NY
NY
AZ
CA

Table

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

DPSI

AZ
AZ
NY
AZ
CA
CA
CA
NY
AZ
NY
NY
AZ
CA

Where State = 'NY'

**Where State = 'NY'
And Shopno = 303**

**Where State = 'NY'
And Shopno = ?**

Partitioned Partitioning Index

101
103
105
202
204
206
206
206
301
303
303
402
404

Table

101	AZ
103	NY
105	AZ
202	CA
204	CA
206	AZ
206	CA
206	NY
301	NY
303	AZ
303	NY
402	AZ
404	CA

Data Partitioned Secondary Index

AZ	101
AZ	105
NY	103
AZ	206
CA	202
CA	204
CA	206
NY	206
AZ	303
NY	301
NY	303
AZ	402
CA	404

Non Partitioned Secondary Index

AZ	101
AZ	105
AZ	206
AZ	303
AZ	402
CA	202
CA	204
CA	206
CA	404
NY	103
NY	206
NY	301
NY	303

Non-Unique

D.6 - Table Partitioning - Clustering



Any index which is defined on a table partitioned table can be defined as the clustering index

We can also ALTER an index between NOT CLUSTER and CLUSTER

Partitioned Partitioning Index

101
103
105
202
204
206
206
206
301
303
303
402
404

Table

101	AZ
105	AZ
103	NY
206	AZ
202	CA
204	CA
206	CA
206	NY
303	AZ
301	NY
303	NY
402	AZ
404	CA

Clustering Data Partitioned Secondary Index

AZ	101
AZ	105
NY	103
AZ	206
CA	202
CA	204
CA	206
NY	206
AZ	303
NY	301
NY	303
AZ	402
CA	404

D.7 - Add Partition



ALTER TABLE *tbpart* ADD PARTITION ENDING AT (599)

No partition number specified

Inherits attributes from previous partition (including REORP)

Table is immediately available

Packages and Dynamic Cache invalidated

Last partition limit enforced

Partitioned Partitioning Index

101
103
105
202
204
206
206
206
301
303
303
402
404
501
502

Table

101	AZ
105	AZ
103	NY
206	AZ
202	CA
204	CA
206	CA
206	NY
303	AZ
301	NY
303	NY
402	AZ
404	CA
501	NY
502	NY

Clustering Data Partitioned Secondary Index

AZ	101
AZ	105
NY	103
AZ	206
CA	202
CA	204
CA	206
NY	206
AZ	303
NY	301
NY	303
AZ	402
CA	404
NY	501
NY	502

D.8 - Maximum Number of Partitions



Table 56. Maximum number of partitions when DSSIZE = 0

Type of table space	Number of existing partitions	Maximum partitions
non-large	1 to 16	16
non-large	17 to 32	32
non-large	33 to 64	64
large	N/A	4096

Table 55. Maximum number of partitions allowed

DSSIZE	Page size 4KB	Page size 8KB	Page size 16KB	Page size 32KB
1GB-4GB	4096	4096	4096	4096
8GB	2048	4096	4096	4096
16GB	1024	2048	4096	4096
32GB	512	1024	2048	4096
64GB	256	512	1024	2048

D.9 - Rotate Partition



ALTER TABLE *tbpart* ROTATE PARTITION FIRST TO LAST
ENDING AT (699) RESET

Table is immediately available – but...

RESET to remind that all rows in first partition are deleted – NPI!

Inherits REORP from last partition

Packages and Dynamic Cache invalidated

Last partition limit enforced

Partitioned Partitioning Index

601
601
602
202
204
206
206
206
301
303
303
402
404
501
502

Table

601	AZ
602	AZ
601	NY
206	AZ
202	CA
204	CA
206	CA
206	NY
303	AZ
301	NY
303	NY
402	AZ
404	CA
501	NY
502	NY

Clustering Data Partitioned Secondary Index

AZ	601
AZ	602
NY	601
AZ	206
CA	202
CA	204
CA	206
NY	206
AZ	303
NY	301
NY	303
AZ	402
CA	404
NY	501
NY	502

D.10 - Partition Rebalancing



For Index Partitioning we can change partition limits with ALTER Index

For Table Partitioning we can do the same with ALTER Table

The partition and its neighbour are placed in REORG

Remember REORG SCOPE PENDING!

D.11 - Partition Rebalancing



REORG TABLESPACE ts REBALANCE

Can be at the PART level,
- physical partition numbers are used

Careful if many duplicates

Not CHANGE or SCOPE PENDING

If Partitioning <> Clustering we have to do a second reorg – AREO*

D.12 - Converting to Table Partitioning



The easiest way,

- ALTER partix NOT CLUSTER
- ALTER partix CLUSTER

...

Drop Partitioning Index

Alter Table ... Add Partition

Alter Table ... Rotate First to Last Partition

Alter Table ... Alter Partition n

Create Index ... Partitioned

D.13 – Partitioned Tables – What is Missing?



ALTER to LARGE (or better DSSIZE)
ALTER DSSIZE

ALTER Index to
Partitioned / Not Partitioned

DROP Partition

REORG REBALANCE should be one-step