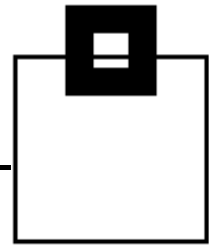
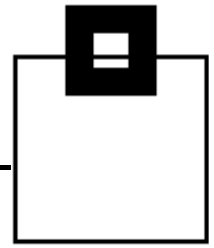


Are *you* a Runstats Master?

Ulf Heinrich
SEGUS, Inc
u.heinrich@segus.com

AGENDA

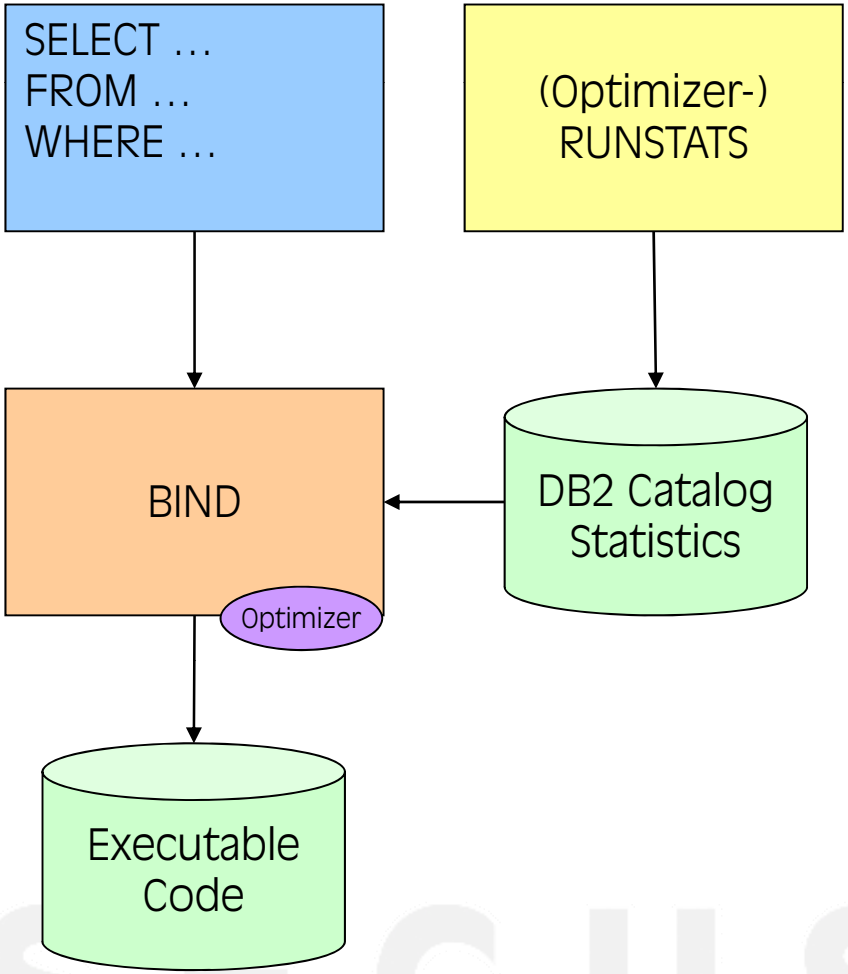


- DB2 Statistics
  - DB2 Catalog Statistics versus DB2 Realtime Statistics
  - DB2 Optimizer and access path relevance

- DB2 RUNSTATS basics
- IBM recommendations through the ages
- DB2 RUNSTATS advanced
  - SYSCOLDIST explained

- RUNSTATS real world QA
- Statistics HealthCheck catalog checker

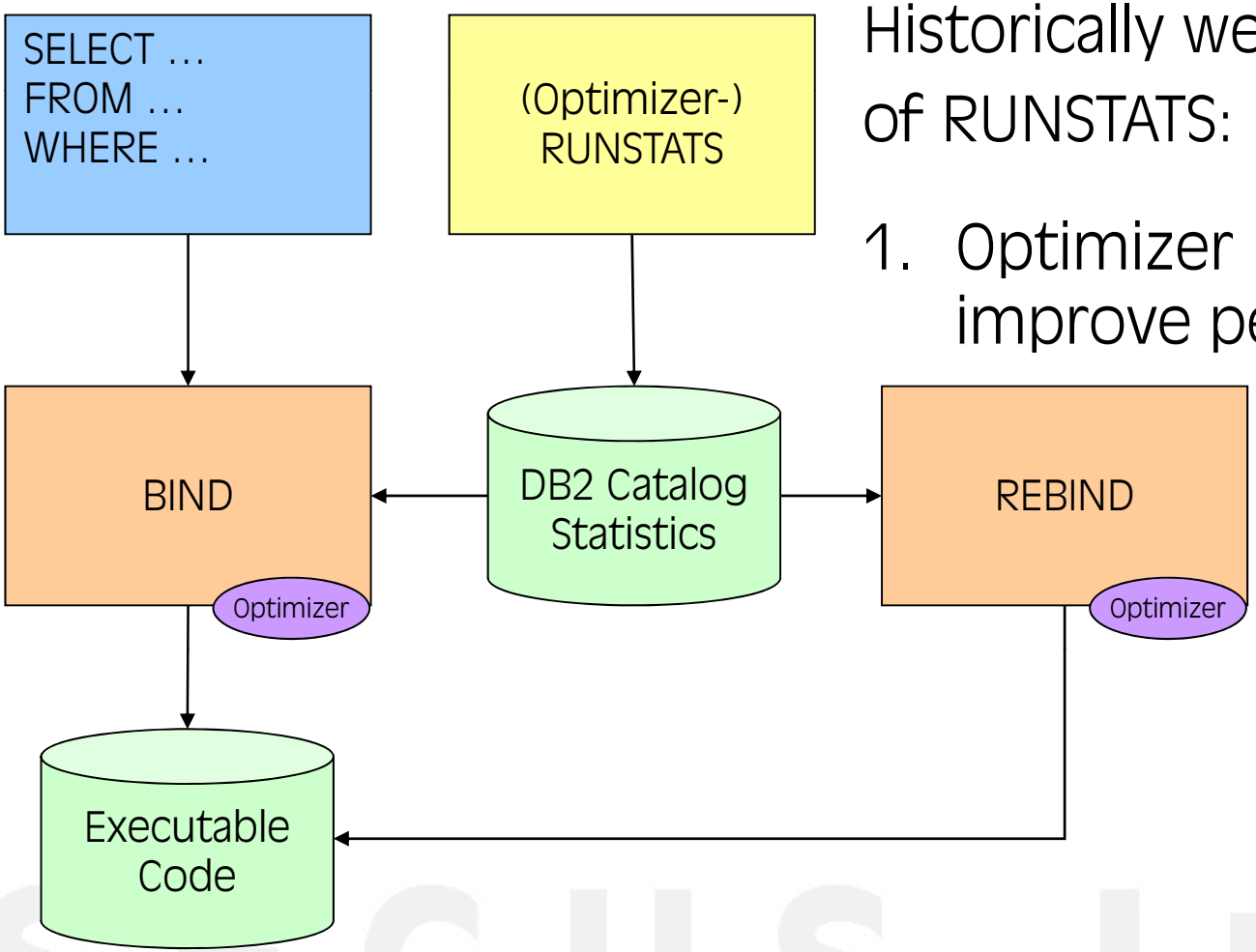
RTS vs. DB2 Catalog Statistics



Historically we have two types of RUNSTATS:

1. Optimizer RUNSTATS to improve performance

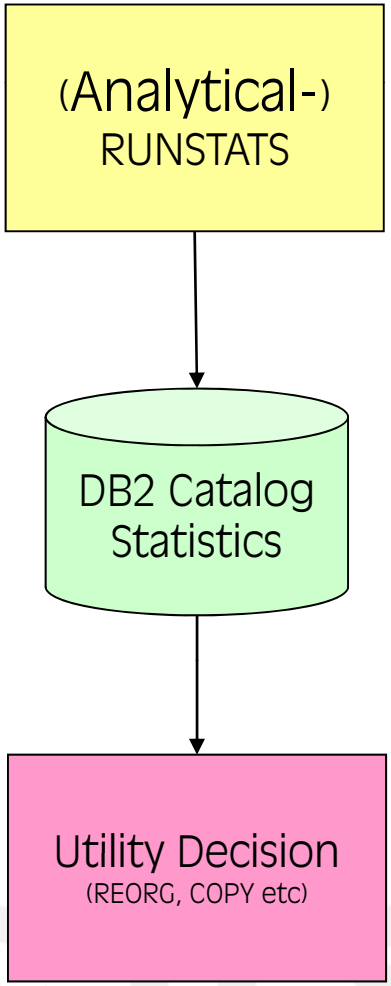
RTS vs. DB2 Catalog Statistics



Historically we have two types of RUNSTATS:

1. Optimizer RUNSTATS to improve performance

RTS vs. DB2 Catalog Statistics



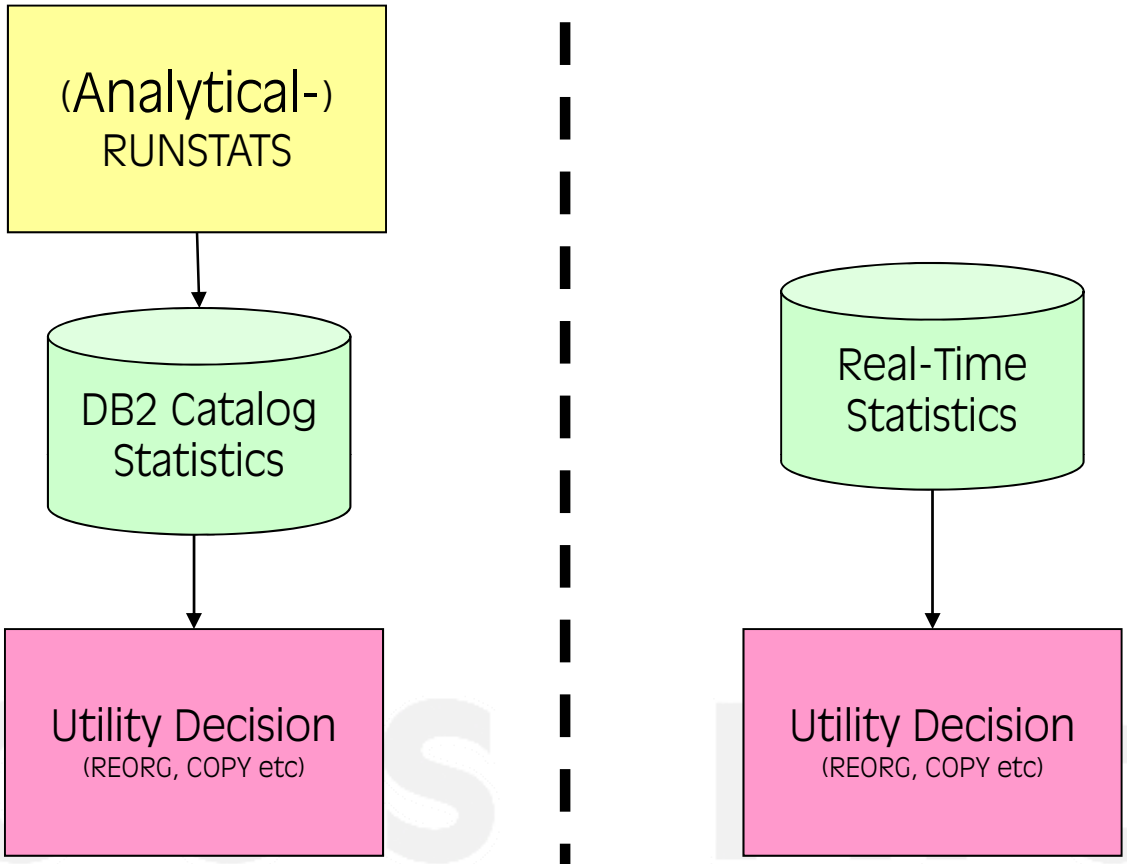
Historically we have two types of RUNSTATS:

1. Optimizer RUNSTATS to improve performance
2. Analytical RUNSTATS for administrative tasks (e.g.threshold based utilities)

RTS vs. DB2 Catalog Statistics

Since DB2 V7 we have two repositories

RUNSTATS → DB2 catalog | RTS → RTS objects



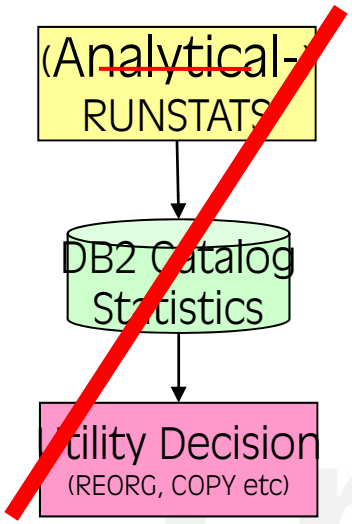
RTS vs. DB2 Catalog Statistics

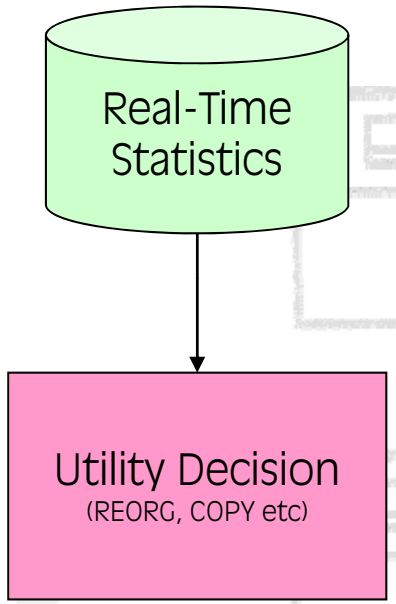
- Stop using catalog statistics for analytical RUNSTATS
- Use Real-Time Statistics for accurate statistics

→ Save CPU! Eliminate analytical RUNSTATS

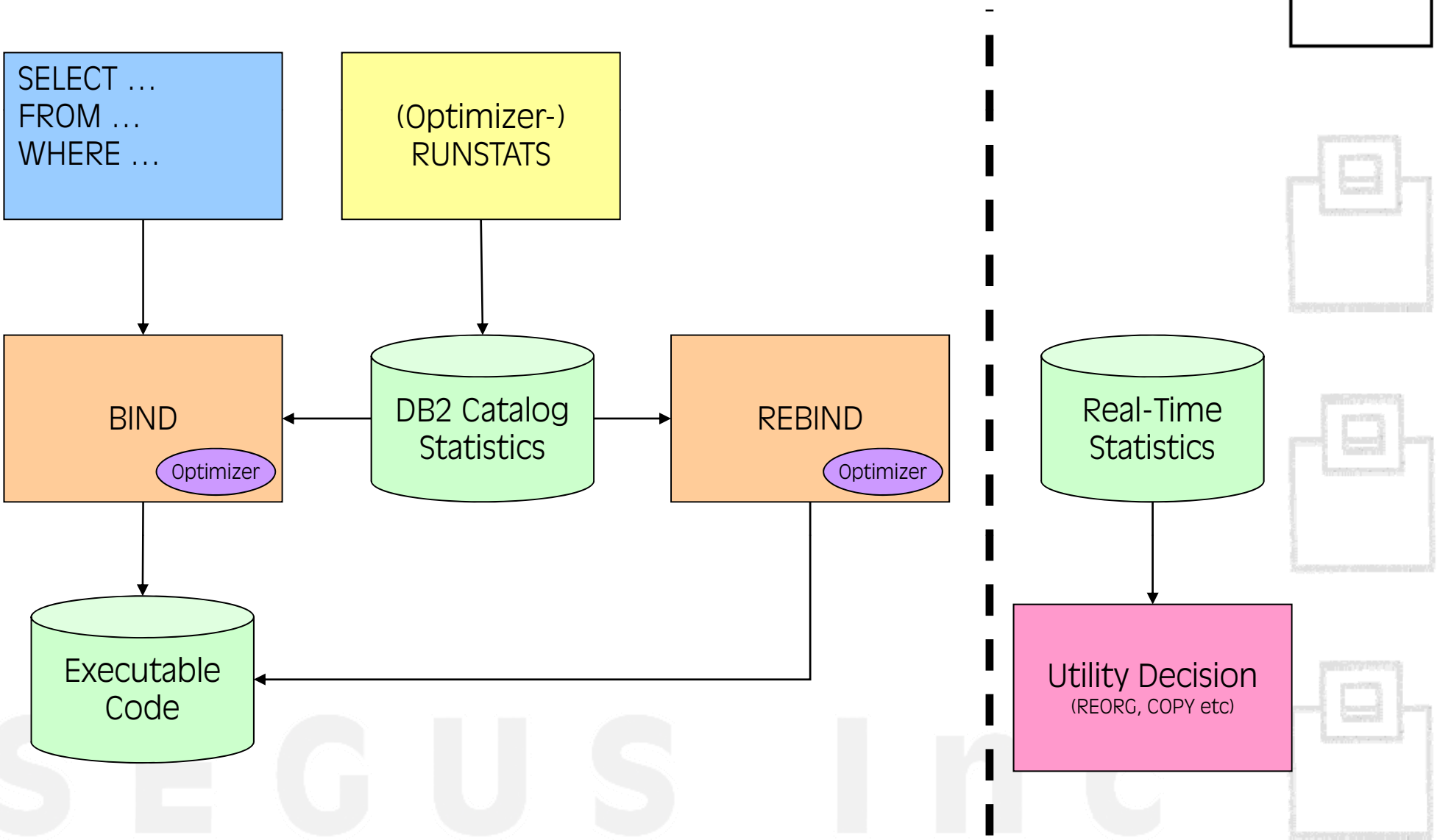
→ Eliminate the "lag" time

→ Secure your optimizer base

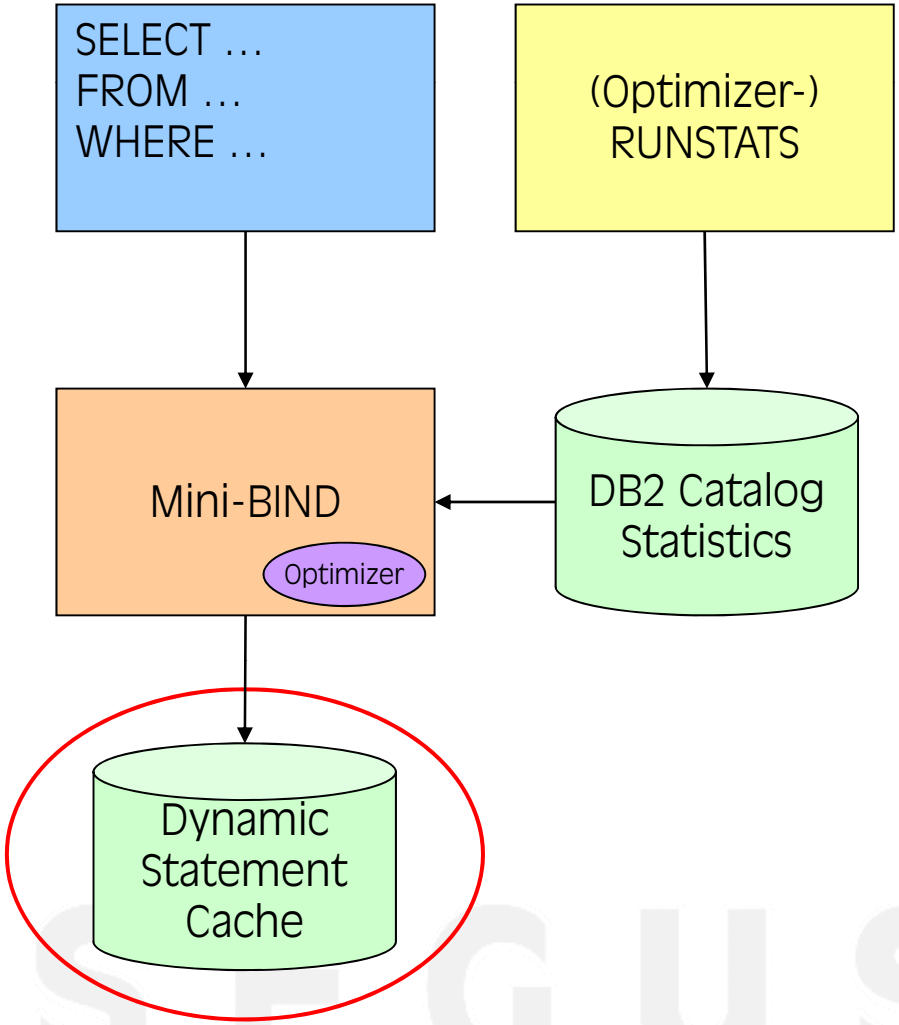




RTS vs. DB2 Catalog Statistics



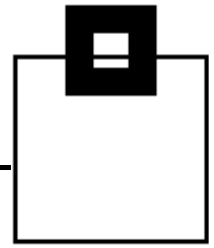
RTS vs. DB2 Catalog Statistics



Access Paths for dynamic SQL are determined on the fly and stored in the DSC.

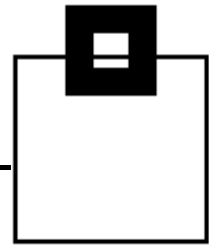
RUNSTATS invalidates and flushes the DSC for an object.

Basic RUNSTATS knowledge



- Different types of RUNSTATS
  - RUNSTATS tablespace
  - RUNSTATS index
  - REORG / LOAD with Inline RUNSTATS
- Different types of statistics
  - Pure access path statistics
    - Those used by BIND in its process of optimization to determine access path
  - Parallelism access path statistics
    - Those used by BIND in its process of optimization to determine the degree of parallelism
  - Space statistics
    - Those used by the DBA to monitor space usage; to assist in capacity planning; to determine frequency of reorg; etc.

Basic RUNSTATS knowledge



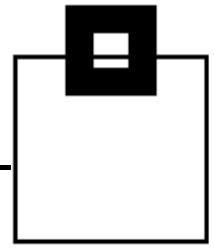
- The RUNSTATS Utility

```
RUNSTATS TABLESPACE <DB>.<TS>
 TABLE(<CR>.<TB>)
  COLGROUP (<CO_A>, <CO_B>)
    FREQVAL COUNT 10 MOST
    HISTOGRAM NUMQUANTILES 100
  SAMPLE 25

 INDEX(ALL)
  KEYCARD FREQVAL NUMCOLS 1 COUNT 10
          FREQVAL NUMCOLS 2 COUNT 10
          FREQVAL NUMCOLS 3 COUNT 10
          HISTOGRAM NUMCOLS 4 NUMQUANTILES 100

  REPORT NO
  UPDATE ALL
  HISTORY NONE
```

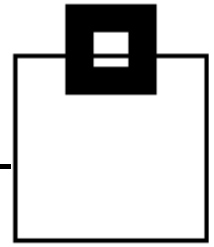
Catalog tables used for access path



- SYSIBM.SYSCOLDIST
- SYSIBM.SYSCOLSTATS *
- SYSIBM.SYSCOLUMNS
- SYSIBM.SYSINDEXES
- SYSIBM.SYSINDEXPART
- SYSIBM.SYSKEYTARGETS V9 only (same as SYSCOLUMNS)
- SYSIBM.SYSKEYTGTDIST V9 only (same as SYSCOLDIST)
- SYSIBM.SYSROUTINES
- SYSIBM.SYSTABLES
- SYSIBM.SYSTABLESPACE
- SYSIBM.SYSTABSTATS

•* degree of parallelism only and after APAR PK62804 also „sometimes“ used to bound filter factor estimates.

Columns used for access path



SYSCOLDIST / SYSKEYTGTDIST 9
- CARDF
- COLGROUPCOLNO / KEYGROUPKEYNO
- COLVALUE / KEYVALUE
- FREQUENCYF
- HIGHVALUE 9
- LOWVALUE 9
- NUMCOLUMNS / NUMKEYS
- QUANTILENO 9
- STATSTIME
- TYPE

SYSCOLUMNS / SYSKEYTARGETS 9
- COLCARDF / CARDF
- HIGH2KEY
- LOW2KEY
- STATS_FORMAT 9

SYSCOLSTATS
- COLCARD
- HIGHKEY
- LOWKEY

SYSINDEXES
- CLUSTERING*
- CLUSTERRATIOF
- DATAREPEATFACTORF 9
- FIRSTKEYCARDF
- FULLKEYCARDF
- NLEAF
- NLEVELS

SYSTABLES
- CARDF
- EDPROC*
- NPAGES
- NPAGESF
- PCTROWCOMP

SYSROUTINES
- CARDINALITY*
- INITIAL_INSTS*
- INITIAL_IOS*
- INSTS_PER_INVOC*
- IOS_PER_INVOC*

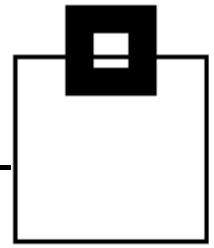
SYSINDEXPART
- LIMITKEY*

SYSTABLESPACE
- NACTIVE
- NACTIVEF

SYSTABSTATS
- CARDF
- NPAGES

* These columns are not updated by RUNSTATS

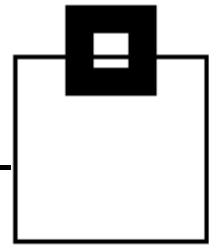
So what?



OK, we now know all the info that DB2 uses to choose its access path. What can we do with this info?

- We can change it to, hopefully, improve an SQL.
- We can delete some of it to, hopefully, improve an SQL.
- We can insert into it to, hopefully, improve an SQL.
- We can ignore it and simply trust that DB2
  „Knows what it is doing…"
- We can mess it all up.
- We can use clever add-ons to *really* mess it all up!
- We can use clever software to see what state all these statistics are currently in. Good, Bad, or Awful.

IBM Recommendations DB2 2.3



Correlations in the catalog (DB2 Administration Guide)

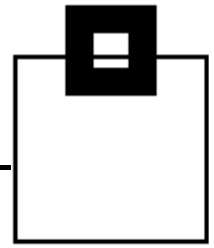
Relationships exist among certain columns of certain tables:

- Columns within SYSCOLUMNS
- Columns in the tables SYSCOLUMNS and SYSINDEXES
- Columns in the tables SYSCOLUMNS and SYSCOLDIST (well actually SYSFIELDS in those days!)

If you plan to update some values, keep in mind the following correlations:

- COLCARDF and FIRSTKEYCARDF/FULLKEYCARDF
- COLCARDF, LOW2KEY and HIGH2KEY. For non-default COLCARDF

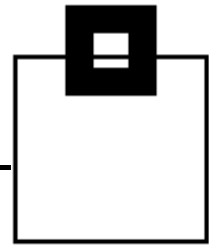
IBM Recommendations DB2 3.1



Correlations in the catalog (DB2 Administration Guide)

No change

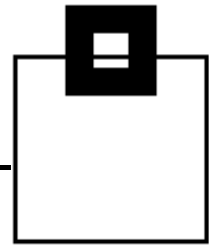
IBM Recommendations DB2 4.1



Correlations in the catalog (DB2 Administration Guide)

No change

IBM Recommendations DB2 5.1



Correlations in the catalog (DB2 Administration Guide)

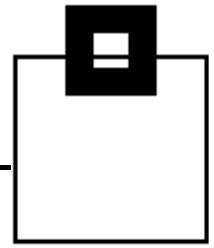
Relationships exist among certain columns of certain tables:

- Columns in the tables SYSCOLUMNS, SYSCOLDIST, and SYSINDEXES

If you plan to update some values, keep in mind the following correlation:

- CARDF in SYSCOLDIST. CARDF is related to COLCARDF and FIRSTKEYCARDF and FULLKEYCARDF. It must be the minimum:
- A value between FIRSTKEYCARDF and FULLKEYCARDF if the index contains the same set of columns
- A value between MAX(colcardf of each col) and the product of all the columns COLCARDFs in the group

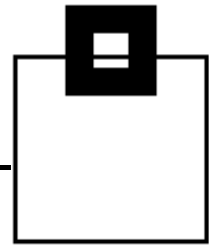
IBM Recommendations DB2 6.1



Correlations in the catalog (DB2 Administration Guide)

No change

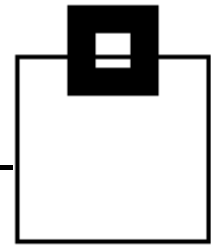
IBM Recommendations DB2 7.1



Correlations in the catalog (DB2 Administration Guide)

No change

IBM Recommendations DB2 8.1

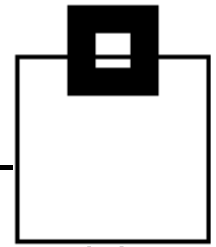


Correlations in the catalog (DB2 Administration Guide)

If you plan to update some values, keep in mind the following correlations:

- The COLCARDF, LOW2KEY, and HIGH2KEY gained a note:
  „If the COLCARDF is 1 or 2 DB2 uses LOW2KEY and HIGH2KEY as domain statistics to generate frequencies“
- CARDF in SYSTABLES. CARDF must be equal or larger than any other cardinalities, such as COLCARDF, FIRSTKEYCARDF, FULLKEYCARDF, and CARDF in SYSCOLDIST
- FREQUENCYF and COLCARDF or CARDF. The number of frequencies collected must be less than or equal to COLCARDF for the column or CARDF for the column group
- FREQUENCYF. The sum of frequencies collected for a column or column group must be less than or equal to 1

IBM Recommendations DB2 8.1



In the „Filter factors and catalog statistics“ chapter (DB2 Administration Guide)

Recommendation: If query performance is not satisfactory, consider the following actions:

- Collect cardinality statistics on all columns that are used as predicates in a WHERE clause.

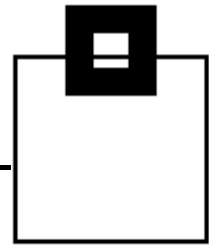
- Collect frequencies for all columns with a low cardinality that are used as COL op literal predicates.

- Collect frequencies for a column when the column can contain default data, the default data is skewed, and the column is used as a COL op literal predicate.

- Collect KEYCARD on all candidate indexes.

- Collect column group statistics on all join columns.

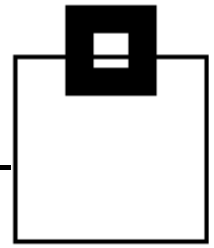
IBM Recommendations DB2 9.1



Correlations in the catalog (DB2 Performance Monitoring and Tuning guide – New book!)

No change.

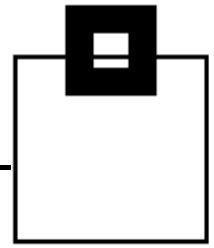
IBM Recommendations DB2 9.1



In the „Filter factors and catalog statistics“ chapter (DB2 Performance Monitoring and Tuning Guide)

There is a new section all about HISTOGRAM statistics.

SYSCOLDIST contents explained



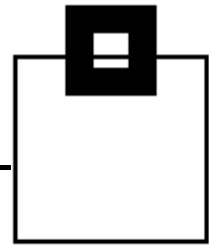
The SYSCOLDIST is used for two, three from V9, separate functions:

- Frequencies
- Cardinalities
- Histograms in V9 and above

The column TYPE can contain „C“ „F“ , if V8 and above „N“ for non-padded frequency values and, if V9 and above, „H“.

Regardless of the TYPE value the columns TBOWNER, TBNAME, NAME, COLGROUPCOLNO, NUMCOLUMNS and STATSTIME are used for the same purpose.

SYSCOLDIST contents explained



TBOWNER and TBNAME are self explanatory.

NAME is the always just the first column name.

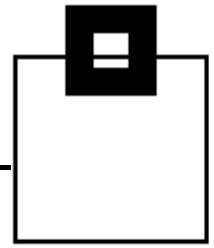
These three columns are also the non-unique index.

COLGROUPCOLNO for a single column object is an empty string, for a multi-column object it contains a string of two byte smallint fields which contain the column numbers from the original table.

NUMCOLUMNS is the number of columns in this group.

STATSTIME is the time when RUNSTATS inserted this entry. It is also used when there is a complete duplicate so that DB2 uses the last inserted value.

SYSCOLDIST contents explained

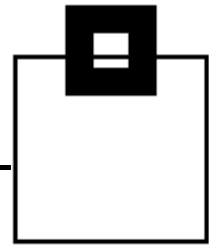


For frequency data the following columns are used:

COLVALUE contains the actual data from the 1 – n columns in the group simply concatenated together. This data might be readable or it might not. Caution must be used when inserting or changing this data as numeric and date, time forms must be the internal DB2 format and not the external format (E.g. The high bit must be flipped for numbers) and watch out for NULLable columns!

FREQUENCYF contains a floating point value between 0.0 and 1.0 which is the frequency that this value occurs for this group in the table.

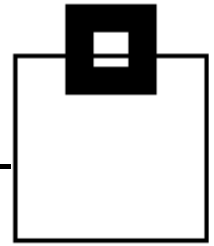
SYSCOLDIST contents explained



For cardinality data the following column is used:

CARDF contains a floating point value which is how many different values for this group occur in the data.

SYSCOLDIST contents explained



For histogram data the following columns are used:

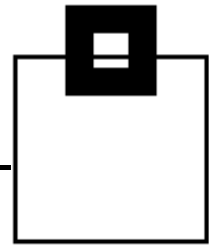
COLVALUE contains the actual data from the 1 – n columns in the quantile simply concatenated together. This data might be readable or it might not. Caution must be used when inserting or changing this data as numeric and date, time forms must be the internal DB2 format and not the external format (E.g. The high bit must be flipped for numbers) and watch out for NULLable columns!

FREQUENCYF contains a floating point value between 0.0 and 1.0 which is the frequency that this value occurs for this quantile.

HIGHVALUE and LOWVALUE contain the upper and lower boundaries of this quantileno

QUANTILENO is the ordinary number of this quantile (1 - 100)

SYSCOLDIST contents explained

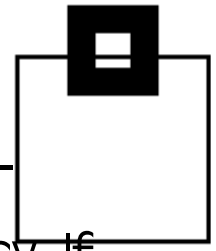


SYSCOLDIST in a nut shell:-

- Frequencies are good for - COL op literal
- Histograms are possibly good for - COL op literal
- Cardinalities are good for - everything!

And remember that a frequency without a cardinality is like a warm beer. Nice to look at, perhaps, but the optimizer will not touch it!

SYSCOLDIST contents explained



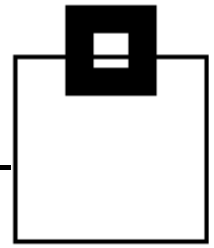
In this example you can see the importance of cardinality and frequency. If no frequency data existed in the SYSCOLDIST DB2 would assume that all values are equally distributed.

Here the default filter factor would be 1/5 (1/COLCARDF) or 20%

Category	Frequency
Infant	5%
Child	15%
Adolescent	25%
Adult	40%
Senior	15%

This would lead the optimizer to under-estimate by 50% for ADULT and to over-estimate by 400% for INFANT.

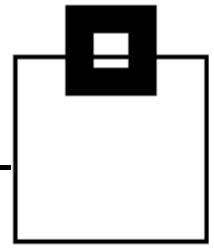
Runstats Q & A



We now know which data is used and where, so now comes a list of RUNSTATS questions:

- FREQVAL NUMCOLS 3 COUNT 10 – What does this do?
- FREQVAL NUMCOLS 3 COUNT 0 - What does this do?
- Use of COLGROUP in DB2 V8
- Use of HISTOGRAM in DB2 V9
- Use of SAMPLE
- Use of REOPT(ONCE)
- Does use of REORG INDEX with inline statistics cause problems?
- What happens to frequencies & cardinalities when not specified in the RUNSTATS utility run?

Runstats Q & A



FREQVAL NUMCOLS 3 COUNT 10

According to the DB2 documentation the NUMCOLS is the number of leading index columns to sample so you actually need to give:

RUNSTATS ROYTEST2.ROYTEST2 TABLE(ALL) INDEX(ALL KEYCARD

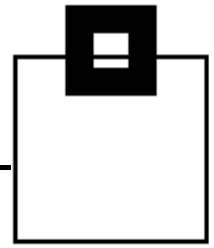
FREQVAL NUMCOLS 1 COUNT 10

FREQVAL NUMCOLS 2 COUNT 10

FREQVAL NUMCOLS 3 COUNT 10)

To get the results you expect!

Runstats Q & A

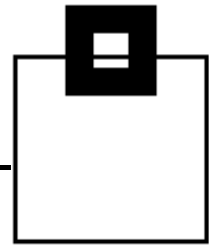


FREQVAL NUMCOLS 3 COUNT 0

This is more of a „hidden feature“ and it deletes all of the multi-column frequencies.

However it does NOT delete any entries created by COLGROUP processing in V8.

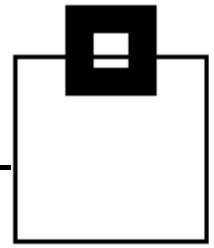
Runstats Q & A



Use of COLGROUP in DB2 V8

This is a very powerful addition to DB2 V8 as it completly removes the need for the old DSTATS program. DSTATS had serious performance problems and column type limitations which made its use complex, time consuming, and error prone.

Runstats Q & A

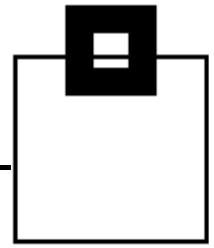


Use of HISTOGRAM in DB2 V9

This is a very powerful addition to DB2 but, as always, must not be used for EVERY table! The use of HISTOGRAM must be weighed up and evaluated on a case by case basis.

Note that if you start to see performance problems then just delete all TYPE H rows from SYSCOLDIST for the relevant TABLES(s)

Runstats Q & A



Use of SAMPLE

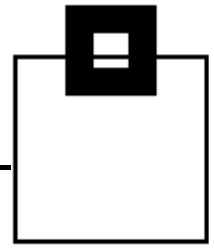
According to the DB2 Utilties documentation this is only for non-indexed columns, however from my tests it has an impact on indexed columns as well.

The problem with SAMPLE is that even using SAMPLE 100 DB2 still gets it wrong…

Col	1	5	10	20	21	22	23	24	25	99	100
TBNAME	62	121	162	178	178	180	180	178	180	180	180
CLNAME	11647	11647	5442	4854	11647	5057	5340	3293	6707	7210	7168

The actual values were 181 and 6871. So be very careful when using this!

Runstats Q & A

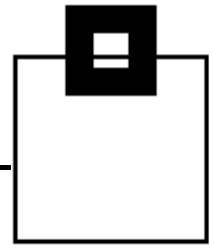


Use of REOPT(ONCE) in DB2

This is a very interesting addition to DB2 as it enables DB2 to do its dynamic SQL mini-bind only once. This can be very good for performance… or not…

If you are a SAP customer then the change that SAP did in ecc5 to use the REOPT(ONCE) could be a CPU killer! The problem is that the first run SQL might not actually reflect the normal SQL that is executed over the day. The way out of this problem is to actually delete all frequency records from the SYSCOLDIST for the relevant tables.

Runstats Q & A

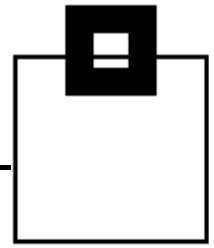


Does use of REORG INDEX with inline statistics cause problems?

YES!

The problem is that the inline stats will *only* update the index statistics and *not* any of the table ones. This leads, very quickly, to the statistics „drifting apart“. The solution here is either run a tablespace runstats after an index reorg, manually update the table statistics, or do not use inline statistics.

Runstats Q & A



What happens to your frequencies, cardinalities and histogram data when you do a RUNSTATS run without using FREQVAL, KEYCARD and /or HISTOGRAM?

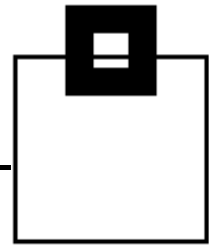
For frequencies the leading column information is replaced (and then the COLGROUPCOLNO is correctly set to an empty string) and all the other data stays in the catalog. This can be a good and a bad thing!

The cardinalities stay there! Again this can be a good and a bad thing!

The histogram data stays there!

This means that if you have ever run a RUNSTATS with either FREQVAL or KEYCARD and since then without these keywords you probably have „interesting“ data in the SYSCOLDIST.

Statistics HealthCheck



Now that you have learned everything that RUNSTATS does, the question is:

How do I know that the statistics I have are correct?

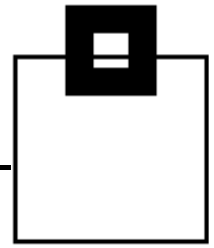
The answer is:

Run **Statistics HealthCheck**

Statistics HealthCheck

Request the freeware at www.SEGUS.com

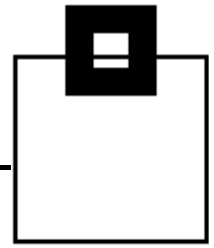
Statistics HealthCheck



Here are some of the critical violations:

```
Critical violations
Frequency < 0  . . . . . . . . . . . . . . . . . . . :          0
Frequency > 100  . . . . . . . . . . . . . . . . . . :          0
Frequency count > cardf  . . . . . . . . . . . . . . :          0
Frequency count > colcardf . . . . . . . . . . . . . :          0
Frequency sum < 0  . . . . . . . . . . . . . . . . . :          0
Frequency sum > 100  . . . . . . . . . . . . . . . . :          0
SYSCOLDIST cardf outside allowable range . . . . . :          1
SYSCOLUMNS - low2key high2key empty  . . . . . . . :          0
SYSCOLUMNS aggregate colcardf < SYSCOLDIST cardf :          1
SYSCOLUMNS 1st cardf <> SYSINDEXES firstkeycardf :          0
SYSCOLUMNS 1st cardf <> SYSINDEXES fullkeycardf  :          0
SYSINDEXES 1 col ix firstkeycardf <> fullkeycardf:          0
SYSTABLES cardf < SYSCOLDIST cardf . . . . . . . . :          0
SYSTABLES cardf < SYSCOLUMNS colcardf  . . . . . . :          0
SYSTABLES cardf < SYSINDEXES firstkeycardf . . . . :          0
SYSTABLES cardf < SYSINDEXES fullkeycardf  . . . . :          0
SYSTABLES no RUNSTATS  . . . . . . . . . . . . . . :         72
```

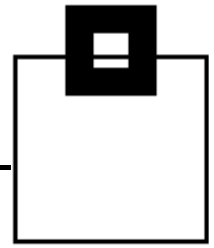
Statistics HealthCheck



Here is an example from the detail report:

```
VIOLATION  DA.S1N PA.T1N
I - CLUSTERING IX NOT CLUSTERED . : PA.X1NPK
S - MISSING INDEX KEYCARD . . . . : PA.X1N022
S - MISSING INDEX KEYCARD . . . . : PA.X1N042
S - MISSING LEADING COLUMN FREQ . : PA.X4010E ADRESS_STREET
S - MISSING LEADING COLUMN FREQ . : PA.X1N017 LBS
W - MISSING INDEX FREQVAL NUMCOLS : PA.X1N017 NUMCOLS 00002
S - MISSING LEADING COLUMN FREQ . : PA.X1N022 POSTCODE
W - MISSING INDEX FREQVAL NUMCOLS : PA.X1N022 NUMCOLS 00002
W - MISSING INDEX FREQVAL NUMCOLS : PA.X1N022 NUMCOLS 00003
S - MISSING LEADING COLUMN FREQ . : PA.X1N042 ADRESS_CITY
W - MISSING INDEX FREQVAL NUMCOLS : PA.X1N042 NUMCOLS 00002
W - MISSING INDEX FREQVAL NUMCOLS : PA.X1N042 NUMCOLS 00003
W - MISSING INDEX FREQVAL NUMCOLS : PA.X1N042 NUMCOLS 00004
W - MISSING INDEX FREQVAL NUMCOLS : PA.X1N042 NUMCOLS 00005
```

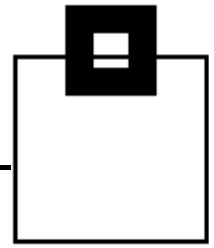
Garbage in the SYSCOLDIST



A colleague is on holiday and the telephone rings with end-users or management saying that the system performance is bad. You grab your RUNSTATS JCL, submit the job and all is well.

Or is it?

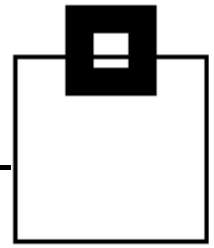
Garbage in the SYSCOLDIST



Here is the query that you have found:

```
SELECT A.ICTYPE , A.TIMESTAMP
FROM SYSIBM.SYSCOPY A
WHERE A.ICTYPE IN ('D','I','F','S','W','Y','R','X','Z')
  AND A.ICBACKUP NOT IN ('RP','RB')
  AND A.DBNAME  = ?
  AND A.TSNAME  = ?
  AND (A.DSNUM  = ? OR (? > 0 AND A.DSNUM = 0))
  AND A.TIMESTAMP = (SELECT MAX(B.TIMESTAMP)
                     FROM SYSIBM.SYSCOPY B
                     WHERE B.ICTYPE IN
                              ('D','I','F','S','W','Y','R','X','Z')
                       AND B.ICBACKUP NOT IN ('LB','RP','RB')
                       AND B.DBNAME  = ?
                       AND B.TSNAME  = ?
                       AND (B.DSNUM  = ? OR (? > 0 AND B.DSNUM = 0))
                    )
WITH UR
;
```

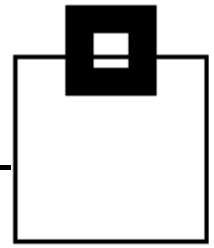
Garbage in the SYSCOLDIST



This is the actual RUNSTATS you should run…

```
RUNSTATS TABLESPACE DSNDB06.SYSCOPY
         TABLE(SYSIBM.SYSCOPY)
         COLGROUP(ICBACKUP) FREQVAL COUNT 10
         COLGROUP(ICTYPE) FREQVAL COUNT 1
         COLGROUP(DSNUM) FREQVAL COUNT 1
         SORTDEVT SYSDA
SHRLEVEL CHANGE REPORT YES
```

The End



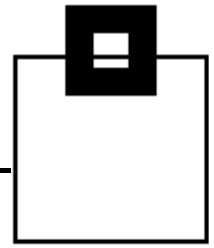
Many thanks for your attention and if you have any questions I will be pleased to answer them!

By the way, you are all

official

RUNSTATS Masters!

Conclusion

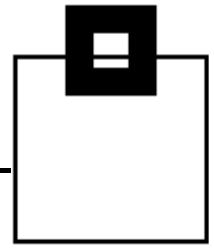


- RUNSTATS enhancements of DB2 V8 and 9 open up significant access path improvements
- Be careful with your DB2 Catalog statistics

  Garbage in - Garbage Out!
- Don't forget to REBIND if you want to exploit what you paid for

- Realtime Statistics provide current statistics without RUNSTATS
  - Good for administrative tasks like threshold based utilities
  - Statistics for volatile tables without compromising access paths
  - Provides individual data for clone and base table

Contact:



u.heinrich@segus.com
www.segus.com