IDUG

2025

Atlanta, GA | June 8-12

NA Db2 TECH CONFERENCE

RUNSTATS Master - reloaded

Roy Boxwell, SEGUS Inc.

Session Code: C03

Platform:

Db2 for z/OS

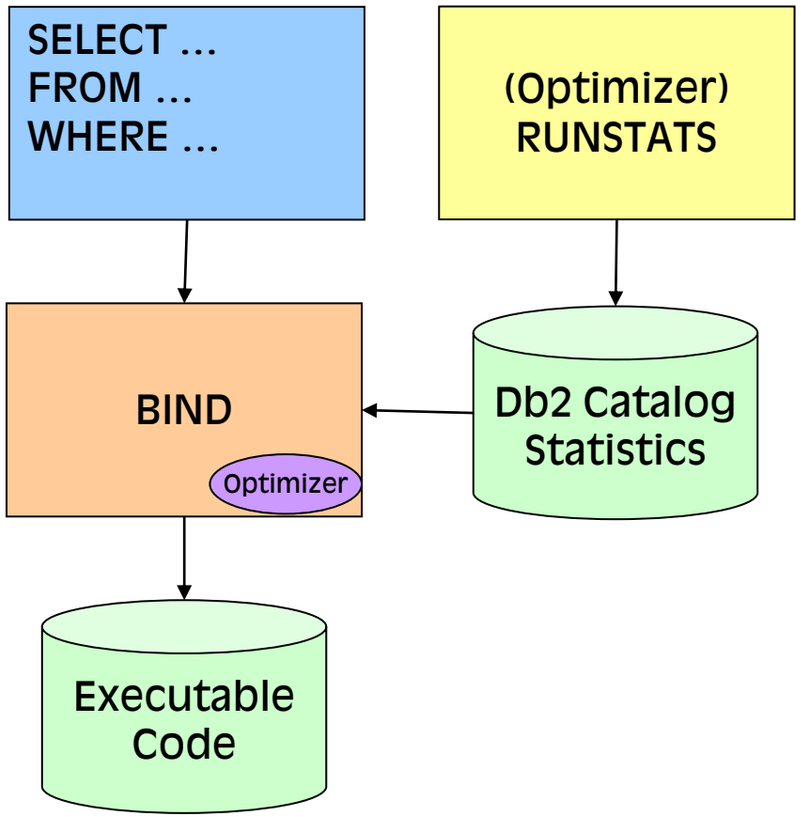
AGENDA

- RTS vs Db2 Catalog Statistics
- Basic RUNSTATS knowledge
- Db2 Optimizer decision sources
- IBM recommendations through the ages
- SYSCOLDIST contents explained
- RUNSTATS Q&A

AGENDA

- RTS vs Db2 Catalog Statistics
- Basic RUNSTATS knowledge
- Db2 Optimizer decision sources
- IBM recommendations through the ages
- SYSCOLDIST contents explained
- RUNSTATS Q&A

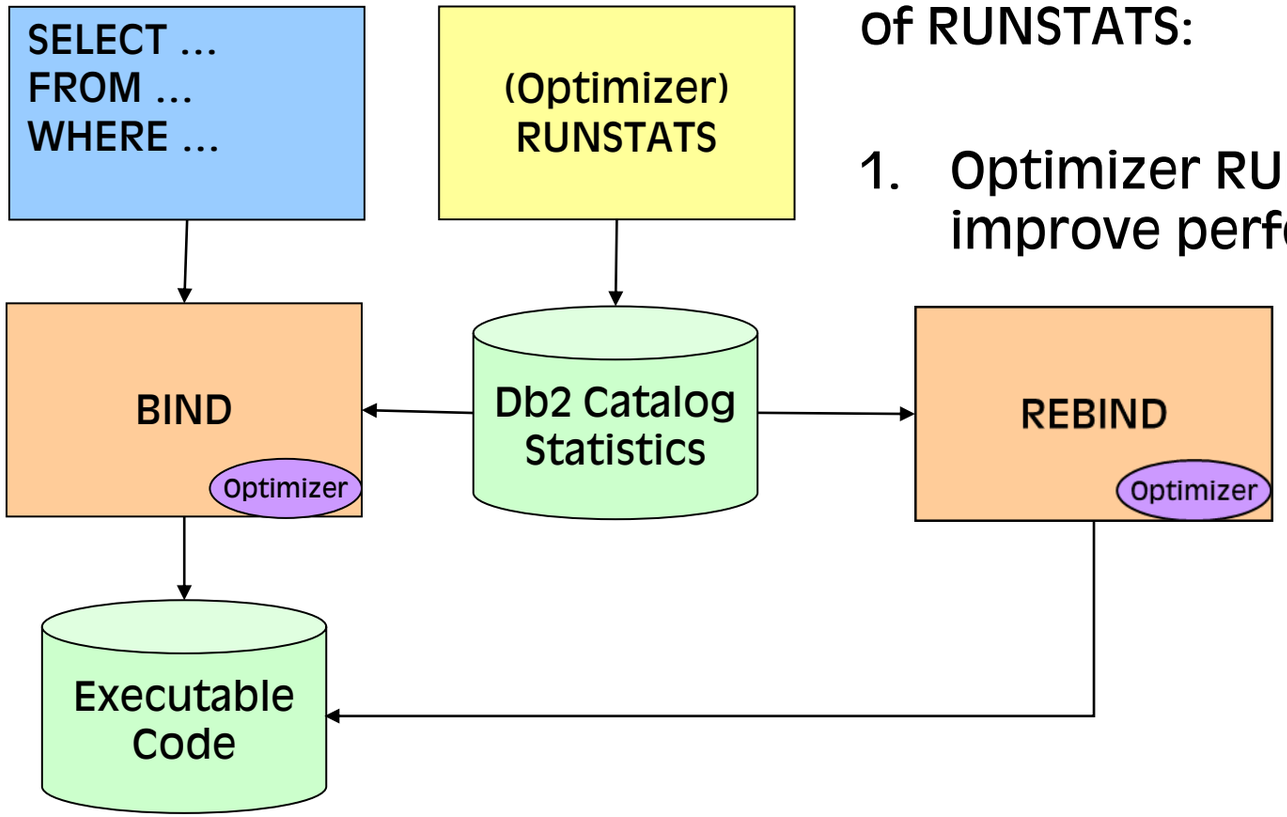
RTS vs Db2 Catalog Statistics (1:6)



Historically, we had two types of RUNSTATS:

1. Optimizer RUNSTATS to improve performance

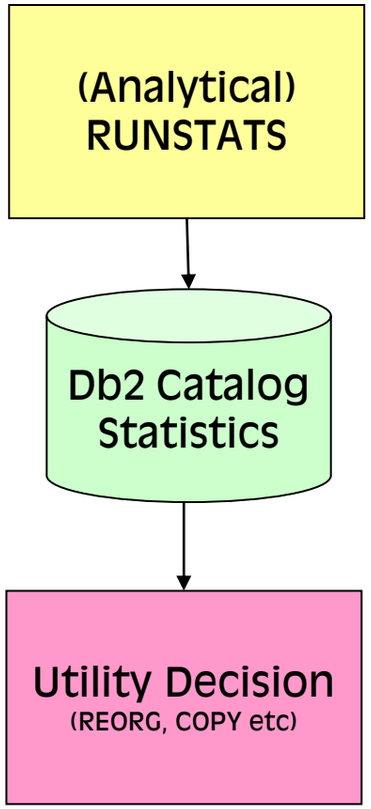
RTS vs Db2 Catalog Statistics (2:6)



Historically, we had two types of RUNSTATS:

1. Optimizer RUNSTATS to improve performance

RTS vs Db2 Catalog Statistics (3:6)

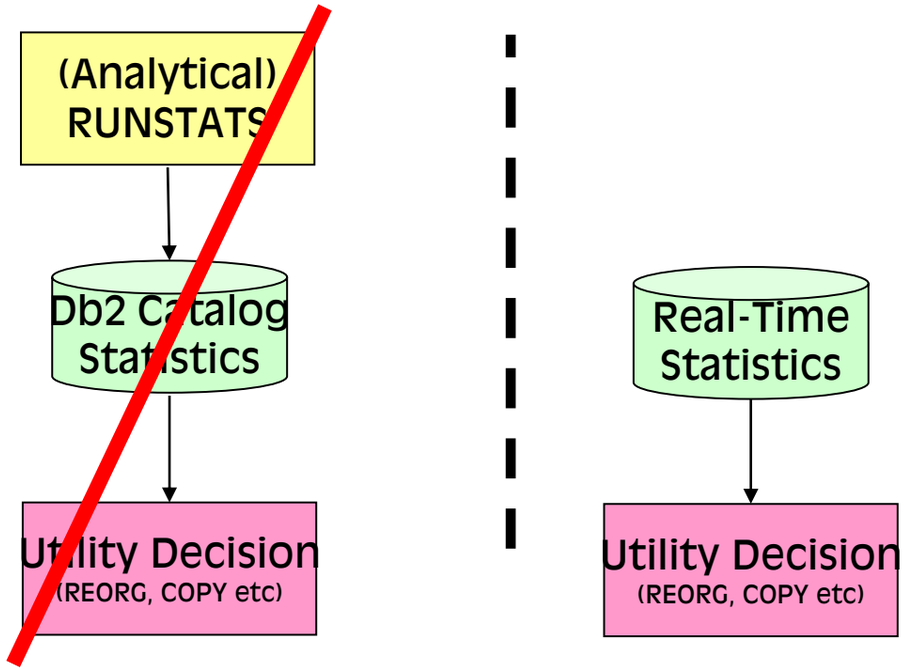


Historically, we had two types of RUNSTATS:

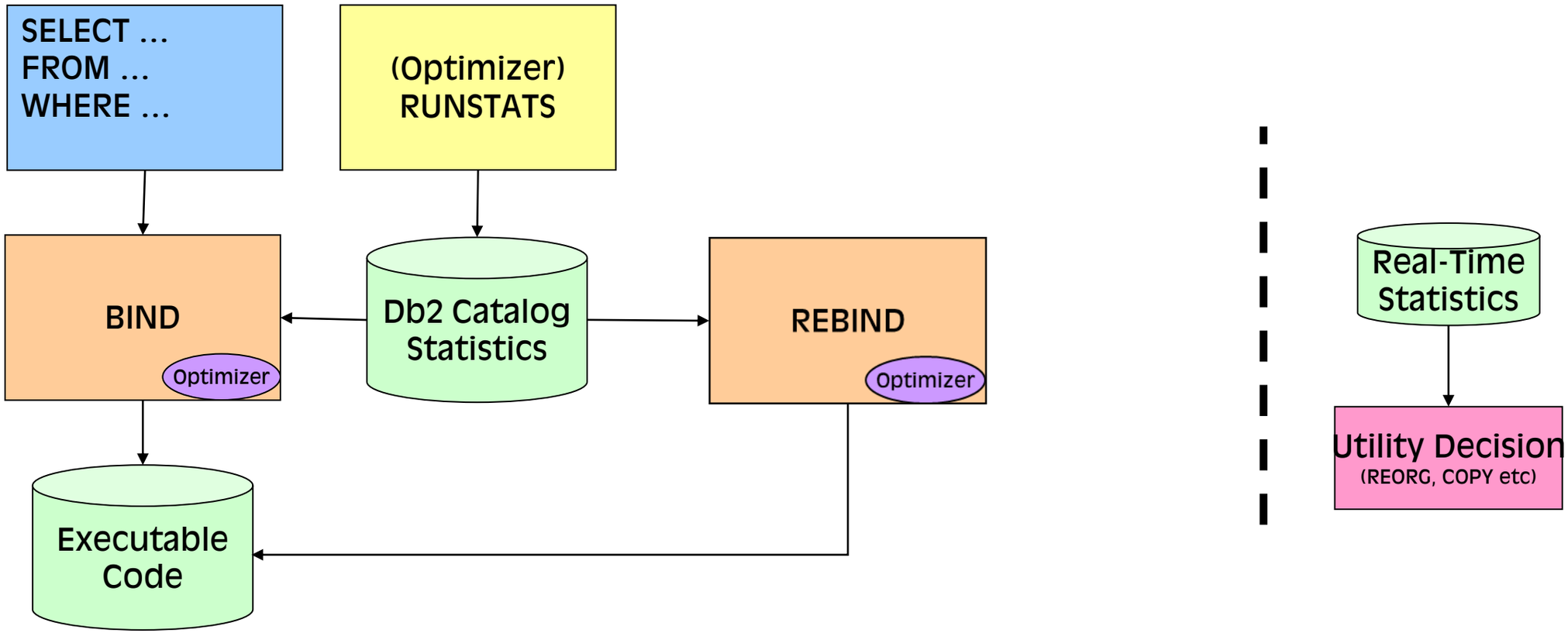
1. Optimizer RUNSTATS to improve performance
2. Analytical RUNSTATS for administrative tasks (e.g., threshold-based utils)

RTS vs Db2 Catalog Statistics (4:6)

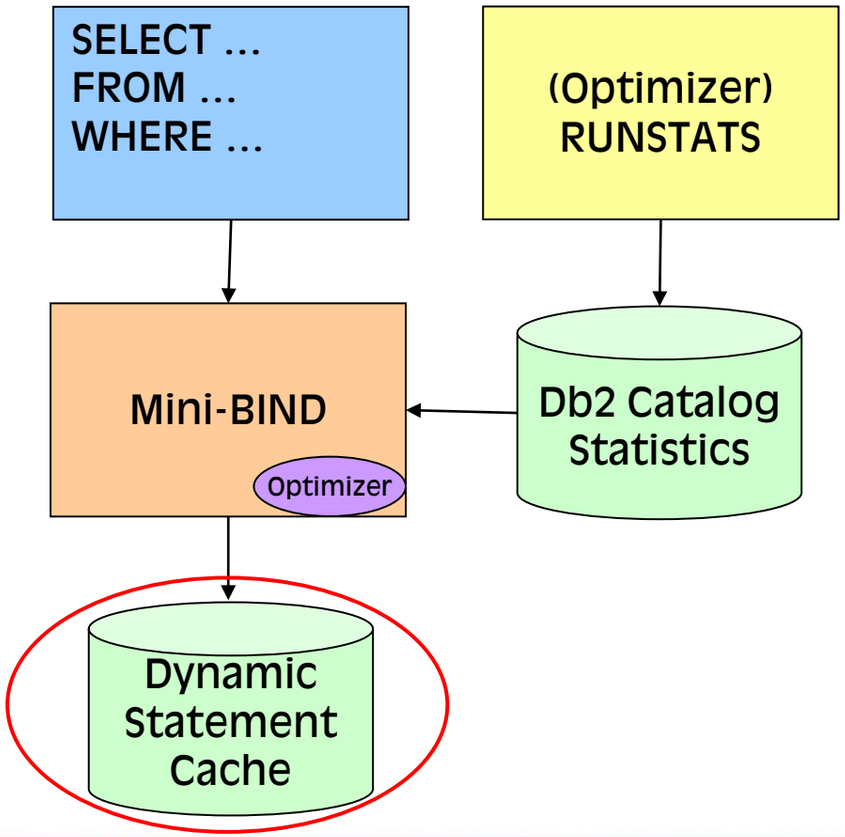
- Do not use Db2 Catalog statistics for analysis
- Use Real-Time Statistics for analysis
- Saves CPU & Eliminates the analytical RUNSTATS
- Secures your optimizer base
- RUNSTATS only updates RTS TOTALROWS / TOTALENTRIES when SHRLEVEL NONE is used!



RTS vs Db2 Catalog Statistics (5:6)



RTS vs Db2 Catalog Statistics (6:6)



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

LRU, RUNSTATS, ALTER, DROP, REVOKE, Db2 RESTART all invalidate and flush the DSC for any given object.

AGENDA

- RTS vs Db2 Catalog Statistics
- Basic RUNSTATS knowledge
- Db2 Optimizer decision sources
- IBM recommendations through the ages
- SYSCOLDIST contents explained
- RUNSTATS Q&A

Basic RUNSTATS knowledge (1:4)

- The RUNSTATS utility
  - Gathers summary information about the characteristics of data in table spaces, indexes and partitions.
  - Can invalidate the dynamic statement cache.
  - Optionally:
    - Reports the statistics
    - Updates the Db2 catalog
    - Updates the Db2 catalog history tables

Basic RUNSTATS knowledge (2:4)

- The RUNSTATS utility
  - Gathers summary information about the characteristics of data in table spaces, indexes and partitions.
  - Can invalidate the dynamic statement cache.
  - Optionally:
    - Reports the statistics
    - Updates the Db2 catalog
    - Updates the Db2 catalog history tables
      - Who uses these by the way?

Basic RUNSTATS knowledge (3:4)

- 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 (4:4)

```
RUNSTATS TABLESPACE <DB>.<TS> INVALIDATECACHE YES
 TABLE(<CR>.<TB>)
        COLGROUP (<CO_A>, <CO_B>)
          FREQVAL COUNT 10 MOST
          HISTOGRAM NUMQUANTILES 100
        TABLESAMPLE SYSTEM AUTO

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

        SHRLEVEL CHANGE
        UPDATE ALL
        HISTORY NONE
```

AGENDA

- RTS vs Db2 Catalog Statistics
- Basic RUNSTATS knowledge
- Db2 Optimizer decision sources
- IBM recommendations through the ages
- SYSCOLDIST contents explained
- RUNSTATS Q&A

Db2 Optimizer decision sources (1:10)

SYSIBM.SYSCOLDIST	
SYSIBM.SYSCOLSTATS	(degree of parallelism and „sometimes“ used to bound filter factor estimates)
SYSIBM.SYSCOLUMNS	
SYSIBM.SYSINDEXES	
SYSIBM.SYSINDEXPART	
SYSIBM.SYSINDEXSPACESTATS	
SYSIBM.SYSKEYTARGETS	(same as SYSCOLUMNS)
SYSIBM.SYSKEYTGTDIST	(same as SYSCOLDIST)
SYSIBM.SYSROUTINES	
SYSIBM.SYSTABLES	
SYSIBM.SYSTABLESPACE	
SYSIBM.SYSTABLESPACESTATS	
SYSIBM.SYSTABSTATS	

Db2 Optimizer decision sources (2:10)

SYSCOLDIST / SYSKEYTGTDIST
CARDF
COLGROUPCOLNO / KEYGROUPKEYNO
COLVALUE / KEYVALUE
FREQUENCYF
HIGHVALUE
LOWVALUE
NUMCOLUMNS / NUMKEYS
QUANTILENO

SYSCOLUMNS / SYSKEYTARGETS
COLCARDF / CARDF
HIGH2KEY
LOW2KEY
n/a / STATS_FORMAT

SYSCOLSTATS
COLCARD
HIGHKEY/HIGH2KEY
LOWKEY/LOW2KEY

SYSINDEXES
CLUSTERING*
CLUSTERRATIOF
DATAREPEATFACTORF
FIRSTKEYCARDF
FULLKEYCARDF
NLEAF
NLEVELS

SYSINDEXPART
LIMITKEY*

SYSINDEXSPACESTATS
NLEAF
NLEVELS

SYSTABLES
CARDF
EDPROC*
NPAGES
NPAGESF
PCTROWCOMP

SYSTABLESPACE
NACTIVEF

SYSTABLESPACESTATS
NPAGES
TOTALROWS* (SHRLEVEL NONE)

SYSTABSTATS
CARDF
NPAGES

SYSROUTINES
CARDINALITY*
INITIAL_INSTS*
INITIAL_IOS*
INSTS_PER_INVOC*
IOS_PER_INVOC*

* Columns are not updated by RUNSTATS
_ Columns are not updatable

Db2 Optimizer decision sources (3:10)

For EXPLAIN use:

SIMULATED_ _CPU_COUNT	Number of logical CPUs. Default: Off. Values 0 (Off) or 1 – 255.
SIMULATED _CPU_SPEED	CPU Speed (Microseconds of SRB execution time per service unit) Default: Off. Values 0 (Off) or 1 – 2,147,483,647 (If less than 27 then 27)

Db2 Optimizer decision sources (4:10)

For real use:

MAXRBLK	RID Block Size in Kilobytes. Default: 1,000,000. Values: 0 or 128 to 2,000,000.
MAXSORT_IN_MEMORY	Maximum memory of an ORDER BY and/or GROUP BY for a query in Kilobytes. Default: 2,000. Values: 1,000 up to SRTPOOL size.

Db2 Optimizer decision sources (5:10)

For real use:

NPGTHRSH — Favor Index access for tables or partitions that have fewer data pages than this value.
Default: 1. (SAP: 10)
Values: 0 to 2,147,483,647.

0: Db2 always chooses access path based on cost estimates.

1+: Db2 attempts to use a matching index with the most MATCHCOLS that contains fewer data pages than this value.

Note: For default stats (-1) Db2 uses -1 for the comparison and not 501! The values used are NPAGESF in SYSTABLES or NPAGES in SYSTABSTATS.

Db2 Optimizer decision sources (6:10)

SRTPOOL — Sort Pool size in Kilobytes.
Default 20,000.
Values: 240 to 128,000.

STARJOIN — STAR JOIN available?
Default: Disabled.
Values: Disabled, Enabled, 1 to 32768.

Disabled	No STARJOIN attempted
Enabled	Fact table has a minimum of 25 times the cardinality of the largest dimension table. The ZPARM in MACLIB states „no ratio checks“!
1	The single table with the highest cardinality is the fact table, but if there are two or more equal then star join is not enabled.
2 – 32768	The cardinality of the fact table must be x times the greatest dimension table cardinality.

Db2 Optimizer decision sources (7:10)

SJTABLES — Minimum number of tables in the query block before considering STARJOIN.
Default: 10.
Values: 0 to 32767.

Values:

0	Consider STARJOIN when 10 or more tables are in the query block
1 – 3	Always consider STARJOIN
4 – 225	Only consider STARJOIN if at least this many tables exist in the query block
226+	Never consider STARJOIN

MXDTCACH — Max Data caching in Megabytes for Sparse or Pair-wise access.
Default: 20.
Values: 0 to 512.

Db2 Optimizer decision sources (8:10)

Then we have the parallel access control ZPARMS:

CDSSRDEF — Default value for CURRENT DEGREE special register if not explicity set in the SQL statement.
Default: 1.
Values:

1	The query will not use parallelism
ANY	The query can use parallelism

Note: IBM recommend leaving this at 1 as it is not a good idea to switch on parallel access globally.

Db2 Optimizer decision sources (9:10)

PARAMDEG	MAX DEGREE field. To limit the maximum degree of parallelism for a parallel group. When you specify a non-zero value Db2 will limit the number of tasks. Default 0. Values 0 to 254.
PARAMDEG_DPSI	MAX DEGREE field when a DPSI is driving parallelism. When you specify a non-zero value Db2 will limit the number of tasks. Default 0. Values:

DISABLE	DPSI not used to drive parallelism
0	Use the PARAMDEG
1	Multiple child tasks but one at a time when a DPSI is used to drive
2 to 254	Maximum number of parallel tasks

Db2 Optimizer decision sources (10:10)

Finally, we have the 80 Bufferpool definitions which all get used as well as the VPPSEQT if you have parallel processing!

This all is read, sorted and processed by the optimizer in the blink of an eye and normally gives us a good access path!

I am always impressed.

AGENDA

- RTS vs Db2 Catalog Statistics
- Basic RUNSTATS knowledge
- Db2 Optimizer decision sources
- IBM recommendations through the ages
- SYSCOLDIST contents explained
- RUNSTATS Q&A

IBM Recommendations DB2 V2.3

Correlations in the catalog (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 V3

Correlations in the catalog (Administration Guide)

- No change

IBM Recommendations DB2 V4

Correlations in the catalog (Administration Guide)

- No change

IBM Recommendations DB2 V5

Correlations in the catalog (Administration Guide)

- Relationships exist among certain columns of certain tables:
  - Columns in the tables SYSCOLUMNS, SYSCOLDIST, and SYSINDEXES
  - Columns with table space statistics and columns for partition-level statistics.
- If you plan to update values, keep in mind the following correlation:
  - CARDF in SYSCOLDIST. CARDF is related to COLCARDF and FIRSTKEYCARDF and FULLKEYCARDF. It must be at 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 RUNSTATS changes DB2 V5

RUNSTATS (Utility Guide and reference)

- Db2 invalidates the Dynamic Statement Cache.
- For LARGE tablepaces the following are all set to -1:
  - COLCARD in SYSCOLUMNS
  - CARD in SYSTABLES
  - CARD in SYSINDEXPART
  - FAROFFPOS in SYSINDEXPART
  - NEAROFFPOS in SYSINDEXPART
  - FIRSTKEYCARD in SYSINDEXES
  - FULLKEYCARD in SYSINDEXES

IBM Recommendations DB2 V6

Correlations in the catalog (Administration Guide)

- No change

IBM RUNSTATS changes DB2 V6

RUNSTATS (Utility Guide and reference)

- Use the STATISTICS keyword with LOAD, REBUILD INDEX and REORG to eliminate the need to execute RUNSTATS.
- LOB Spaces can be RUNSTATEd but only for SPACE info.
- For LARGE tablepaces use of SAMPLE can reduce the number of rows processed.

IBM Recommendations DB2 V7

Correlations in the catalog (Administration Guide)

- No change

IBM RUNSTATS changes DB2 V7

RUNSTATS (Utility Guide and reference)

- New keyword HISTORY to populate the SYSxxxx_HIST tables.

- New keyword FORCEROLLUP:
  - If set to YES, forces aggregation from partition level to space level even if some parts are empty.
  - If set to NO, then no aggregation will be done.

  - Note: If some part are empty then message DSNU623I will be issued ***if*** STATISTICS ROLLUP ZPARM is set to NO.

IBM RUNSTATS changes DB2 V7

RUNSTATS (Utility Guide and reference)

- New keyword HISTORY to populate the SYSxxxx_HIST tables.
  Shhh! Do not mention SYSTABLESPACE_HIST
- New keyword FORCEROLLUP:
  - If set to YES, forces aggregation from partition level to space level even if some parts are empty.
  - If set to NO, then no aggregation will be done.

  - Note: If some part are empty then message DSNU623I will be issued ***if*** STATISTICS ROLLUP ZPARM is set to NO.

IBM Recommendations DB2 V8

Filter factors and catalog statistics (Chapter 33 in 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 V8

Correlations in the catalog (Administration Guide)

If you plan to update 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 RUNSTATS changes DB2 V8

RUNSTATS (Utility Guide and reference)

- New keywords COLGROUP and FREQVAL – Big change!
  - New keywords SORTDEVT & SORTNUM for DFSORT.
- HISTORY option decoupled from UPDATE option.
- New „just flush the DSC“ UPDATE NONE REPORT NO allowed.

IBM Recommendations DB2 V9

Improving filter factors (Chapter 33 in Performance Monitoring and Tuning Guide)

- New section all about HISTOGRAM statistics.

IBM Recommendations DB2 V9

Correlations in the catalog (Chapter 36 in Performance Monitoring and Tuning Guide)

- No change.

IBM RUNSTATS changes DB2 V9

RUNSTATS (Utility Guide and reference)

- New keywords HISTOGRAM and NUMQUANTILES.

IBM Recommendations DB2 10

Improving filter factors (Chapter 34 in Managing Performance)

- No change.

IBM Recommendations DB2 10

Correlations in the catalog (Chapter 37 in Managing Performance)

- No change.

IBM RUNSTATS changes DB2 10

RUNSTATS (Utility Guide and reference)

- Not allowed on the newly visible DSNDB01 objects.
- PROFILE support.
- New keyword TABLESAMPLE SYSTEM. Only for Universal Tablespaces (UTS) which is real page sampling not like SAMPLE which is only row based.
- KEYCARD was deprecated.

IBM Recommendations Db2 11

Improving filter factors (Chapter 38 in Managing Performance)

- SYSIBM.SYSSTATFEEDBACK for BIND, REBIND and PREPARE.
- <Schemaname>.DSN_STAT_FEEDBACK for EXPLAIN usage.

IBM Recommendations Db2 11

Correlations in the catalog (Chapter 41 in Managing Performance)

- No change.

IBM RUNSTATS changes Db2 11

RUNSTATS (Utility Guide and reference)

- New keyword STATCLGMEMSRT to improve performance by avoiding an external sort on a single column group. To use this you must see if you are getting message DSNU1388I in your RUNSTATS output and then use this keyword or, what I would do, is simply raise the value of ZPARM STATCLGSRT.

- New keyword RESET ACCESSPATH delete & reset all access path data to do with this object.
  - With new sub-keyword HISTORY ACCESSPATH to save data into SYSTABLES_HIST and SYSINDEXES_HIST before the RESET.

IBM Recommendations Db2 12

Improving filter factors (Chapter 56 in Managing Performance)

- No change.

IBM Recommendations Db2 12

Correlations in the catalog (Chapter 59 in Managing Performance)

- No change.

IBM RUNSTATS changes Db2 12 (1:2)

RUNSTATS (Utility Guide and reference)

- New keyword INVALIDATECACHE NO/YES, default is **NO** – In the past a RUNSTATS ***always*** flushed the DSC – Not any more!
- New keyword REGISTER YES/NO, default YES.
  - NO: It reduces the data sharing overhead but might read uncommitted data.
  - YES: All pages that are read by RUNSTATS are registered in the coupling facility.
    REGISTER YES is always used for indexes.

IBM RUNSTATS changes Db2 12 (2:2)

RUNSTATS (Utility Guide and reference)

- USE PROFILE – From Db2 12 FL507 this deletes all non-profile contained statistics (useful to remove stale stats) and outputs DSNU1390I if any found.
- Query to help find old, stale stats delivered:

```
SELECT TYPE, NUMCOLUMNS, TBOWNER,
       TBNAME, NAME, MIN(STATSTIME) AS MINSTAT,
       COUNT(*) AS OCCURANCES
FROM SYSIBM.SYSCOLDIST CD
WHERE STATSTIME  < CURRENT TIMESTAMP - 1 MONTH
  AND (TYPE IN ('C', 'H')
    OR NUMCOLUMNS > 1
    OR STATSTIME < CURRENT TIMESTAMP - 1 YEAR)
  AND NOT EXISTS (SELECT 1
                  FROM SYSIBM.SYSINDEXES IX
                  WHERE IX.TBCREATOR = CD.TBOWNER
                    AND IX.TBNAME    = CD.TBNAME
                    AND IX.STATSTIME > '0001-01-01-00.00.00.000000'
                    AND CD.STATSTIME BETWEEN IX.STATSTIME - 8 DAYS
                                         AND IX.STATSTIME + 8 DAYS)
  AND NOT EXISTS (SELECT 1
                  FROM SYSIBM.SYSTABLES TB
                  WHERE TB.CREATOR   = CD.TBOWNER
                    AND TB.NAME      = CD.TBNAME
                    AND TB.STATSTIME > '0001-01-01-00.00.00.000000'
                    AND CD.STATSTIME BETWEEN TB.STATSTIME - 8 DAYS
                                         AND TB.STATSTIME + 8 DAYS)
GROUP BY TYPE, NUMCOLUMNS, TBOWNER, TBNAME, NAME
ORDER BY TYPE, NUMCOLUMNS, TBOWNER, TBNAME, NAME
WITH UR;
```

IBM Recommendations Db2 13

Improving filter factors (Chapter 56 in Managing Performance)

- No change.

IBM Recommendations Db2 12

Correlations in the catalog (Chapter 59 in Managing Performance)

- No change.

IBM RUNSTATS changes Db2 13

RUNSTATS (Utility Guide and reference)

- No change.

AGENDA

- RTS vs Db2 Catalog Statistics
- Basic RUNSTATS knowledge
- Db2 Optimizer decision sources
- IBM recommendations through the ages
- SYSCOLDIST contents explained
- RUNSTATS Q&A

SYSCOLDIST contents explained (1:7)

- SYSCOLDIST is used for:
  - Cardinalities
  - Frequencies
  - Histograms
- Column TYPE can contain:
  - C – Cardinality
  - F – Frequency
  - H – Histogram
  - N – Non-padded frequency – Very rare indeed!
- Regardless of the TYPE value, columns TBOWNER, TBNAME, NAME, COLGROUPCOLNO, NUMCOLUMNS, and STATSTIME are used for the same purpose.

SYSCOLDIST contents explained (2:7)

- TBOWNER, TBNAME, and NAME (first column name only) columns are also the non-unique index.
- COLGROUPCOLNO for a single column object is an empty string, for a multi-column object contains a string of two byte SMALLINT, which contains 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, which is also used when there is a complete duplicate so that Db2 uses the last inserted value.

SYSCOLDIST contents explained (3:7)

For cardinality data:

- CARDF contains the number of distinct values in the column group.

SYSCOLDIST contents explained (4:7)

For frequency data:

- 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.
- 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 (5:7)

For histogram data:

- CARDF contains the number of distinct values in the column group within the QUANTILE.
- COLVALUE contains the actual data from the 1 – n columns in the group simply concatenated together.
- 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 within the QUANTILE.
- HIGHVALUE & LOWVALUE are the high and low bounds for the QUANTILE.
- QUANTILENO is the quantile sequence number.

SYSCOLDIST contents explained (6:7)

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!

Remember: A frequency without a cardinality is like a warm beer. It might be nice to look at, but the optimizer will not touch it!

SYSCOLDIST contents explained (7:7)

Importance of cardinality and frequency.

If no frequency data existed in the SYSCOLDIST Db2 assumes all values are equally distributed. 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.

AGENDA

- RTS vs Db2 Catalog Statistics
- Basic RUNSTATS knowledge
- Db2 Optimizer decision sources
- IBM recommendations through the ages
- SYSCOLDIST contents explained
- **RUNSTATS Q&A**

RUNSTATS Q&A (1:17)

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.
- Use of HISTOGRAM.
- Use of SAMPLE & TABLESAMPLE.
- What about PROFILE usage?
- 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 (2:17)

FREQVAL NUMCOLS 3 COUNT 10

NUMCOLS is the number of leading index columns to sample. Here you get ten frequencies for the first three columns only. You actually need to provide:

```
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 thirty frequencies you expected!

RUNSTATS Q&A (3:17)

FREQVAL NUMCOLS 3 COUNT 0

- Like a „hidden feature“
- Deletes all of the multi-column frequencies for 3 columns
- Does NOT delete any entries created by COLGROUP processing.

RUNSTATS Q&A (4:17)

Use of COLGROUP

```
TABLE(creator.name)
  COLGROUP(col1, col2, …)
    FREQVAL COUNT nn xxxx
```

- Very powerful addition to Db2
- Gives the optimizer much better data to base its costs on for the special multi-column access predicate case.
- Only 255 COLGROUPS allowed!
- Avoid using more than COUNT 100 and never over 1000!

RUNSTATS Q&A (5:17)

Use of HISTOGRAM

```
TABLE(creator.name)
  COLGROUP(col1, col2, …)
    FREQVAL COUNT nn xxxx
    HISTOGRAM NUMQUANTILES nn
```

- Should not be used for EVERY table!
- HISTOGRAM must be weighed up and evaluated on a case by case basis.

Note: If you start to see performance problems then just delete all TYPE=„H“ rows from SYSCOLDIST for the relevant TABLES(s)

RUNSTATS Q&A (6:17)

Use of SAMPLE

- ZPARM STATPGSAMP=YES or SYSTEM (Default) overrides use of SAMPLE and forces TABLESAMPLE SYSTEM AUTO for UTS – Much faster than SAMPLE.

- Problem with SAMPLE:
  Even using SAMPLE 100, Db2 still gets it wrong…

RUNSTATS Q&A (7:17)

Use of SAMPLE

Percent / Column	1	5	10	20	25	40	70	90	99	100
TBNAME	26	121	195	299	323	452	398	396	400	400
NAME	401	415	914	1084	1137	1467	1134	1151	1138	1136
OBID	39	138	209	354	403	754	719	791	737	736

4,887 rows with actual values of 416, 1,064 and 800.
Be very careful when using SAMPLE.

RUNSTATS Q&A (8:17)

Use of TABLESAMPLE SYSTEM xxx

Percent / Column	1	5	10	20	25	40	70	90	99	100/ AUTO
TBNAME	4887	93	210	298	329	422	379	403	401	400
NAME	4887	529	1946	1300	893	1163	1178	1113	1126	1136
OBID	1	145	314	342	387	736	729	789	739	736

4,887 rows with actual values of 416, 1,064 and 800.
Be very careful when using TABLESAMPLE SYSTEM xxx.

RUNSTATS Q&A (9:17)

Non-Use of SAMPLE

COLCARD & COLCARDF is still wrong even when you do no sampling at all…Running TABLESAMPLE SYSTEM NONE and without SAMPLE and with exactly the same data as before:

TBNAME is 400

NAME is 1,136

OBID is 736

4,887 rows with actual values of 416, 1,064 and 800.

Be very careful when using TABLESAMPLE SYSTEM xxx.

RUNSTATS Q&A (10:17)

Non-Use of SAMPLE

Your only hope, if you need 100% accurate statistics, is to count it all yourself...

```
SELECT COUNT(DISTINCT TBNAME) FROM ROYBOY.SYSTABLES ;
416
SELECT COUNT(DISTINCT NAME  ) FROM ROYBOY.SYSTABLES ;
1064
SELECT COUNT(DISTINCT OBID  ) FROM ROYBOY.SYSTABLES ;
800
```

RUNSTATS Q&A (11:17)

What about PROFILE usage?

You must set up your profiles first and then you can happily update them with data thus making the RUNSTATS syntax easier.

However, if you have enabled the auto-update of profiles, which is ***on*** by default, then you must make sure that your profiles are not growing out of control!

All profile data lives in the SYSIBM.SYSTABLES_PROFILES table.

RUNSTATS Q&A (12:17)

What about PROFILE usage?

The way you set up your system is to first populate your profile table by using the „dummy“ RUNSTATS syntax SET PROFILE and then either FROM EXISTING STATS ***or*** include normal COLGROUPs etc. in the syntax, **both** are not supported!

From then on, just adding USE PROFILE with/without the INCLUDE(NPI) syntax, and no other columns, colgroups, freqs etc., is all you need.

If no profile is found you will get an automatic

```
COLUMN ALL INDEX ALL
```

style runstats.

RUNSTATS Q&A (13:17)

What about PROFILE usage?

Naturally, you can also use the UPDATE PROFILE syntax to update the stored profiles. Again, this is a „dummy“ RUNSTATS with the new desired columns, COLGROUPs, freqs etc.

Since Db2 12 FL507 USE PROFILE automatically deletes any statistics that it finds in the catalog that are not in the profile. This is very good to remove „stale“ statistics.

Finally, if you are fed up with the Profile you can use the DELETE PROFILE syntax to clean up. Once again, this is a „dummy“ RUNSTATS and no colgroup defs etc. are allowed in the syntax.

RUNSTATS Q&A (14:17)

What about PROFILE usage?

Applying statistics recommendations to statistics profiles automatically came in Db2 12. To enable this feature you do **nothing** as it is, by default on. This was, in my opinion, a bad thing!

ZPARM STATFDBK_SCOPE set to ALL by default

ZPARM STATFDBK_PROFILE set to YES by default

SYSIBM.SYSTABLES column **STATS_FEEDBACK** set to Y by default

Out-of-the box it starts automatically creating (for TYPE='C' with NUMCOLS > 1 and TYPE='F' or 'H‘) and updating profiles…

RUNSTATS Q&A (15:17)

Does use of REORG INDEX with inline statistics cause problems?

- YES!
- 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“.
- Solution: Either run a table space RUNSTATS after an index REORG, manually update the table statistics, or do not use inline statistics.
  - Recommendation is ***no*** inline statistics

RUNSTATS Q&A (16:17)

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 all the other data stays in the catalog. This can be a good and a bad thing.
  - The cardinality data remains.
  - The histogram data remains.

RUNSTATS Q&A (17:17)

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

- 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.
- Here, the RESET ACCESS PATH might come in handy!
- Or, if using PROFILEs and in Db2 12 FL507 and above, the USE PROFILE will take care of it for you!

CONCLUSION (1:2)

- RUNSTATS enhancements in Db2 over the years have opened up significant access path improvements.
- Be careful with your Db2 Catalog statistics

  → Garbage in - Garbage Out!
- Don’t forget to REBIND in order to exploit the new Db2 version that you paid for.
- Real-time statistics provide current statistics, instantly without RUNSTATS

  → Also partly used by the optimizer!

CONCLUSION (1:2)

Many thanks for your attention and now….

CONCLUSION (1:2)

Many thanks for your attention and now….

you are all official

CONCLUSION (1:2)

Many thanks for your attention and now….

you are all official

RUNSTATS Masters!

IDUG

2025

Atlanta, GA | June 8-12

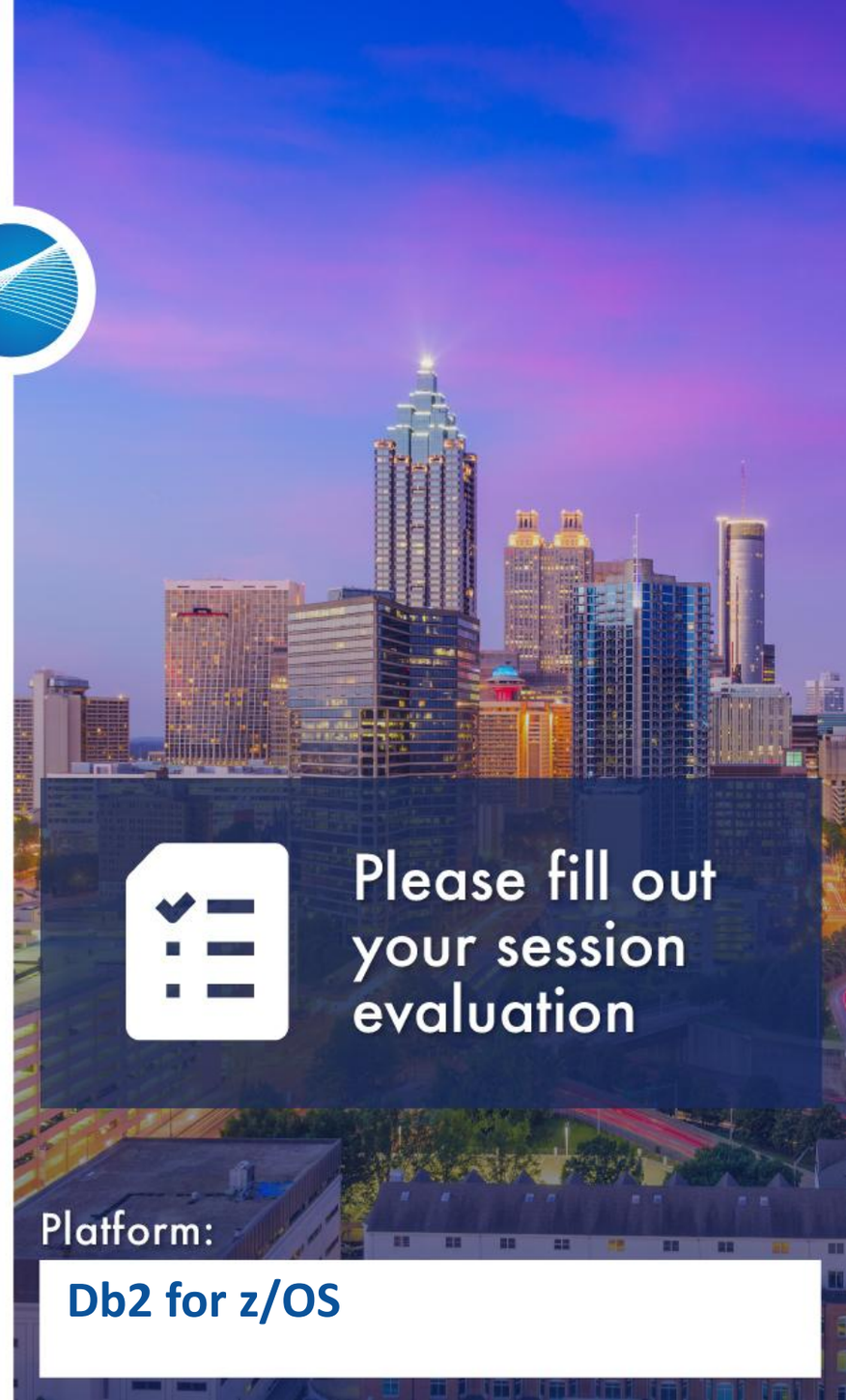
NA Db2 TECH CONFERENCE

RUNSTATS Master - reloaded

Roy Boxwell, SEGUS Inc.

Contact: r.boxwell@segus.com

Session Code: C03




IDUG
2025 NA Db2 Tech Conference