

Mastering Dynamic SQL for DB2 z/OS

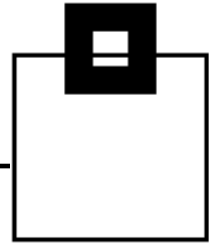
-


Can I control a dynamic world?

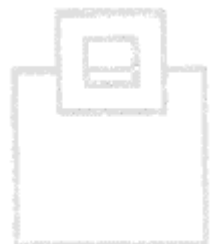


SEGUS Inc

Agenda

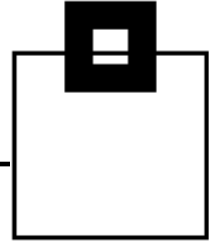


- Dynamic SQL at a glance:
 - Characteristics
 - DB2 setup and support
 - DB2 commands and features
- See the possibilities:
 - Dynamic SQL Management & Protection
using  **Bind ImpactExpert for DB2 z/OS**



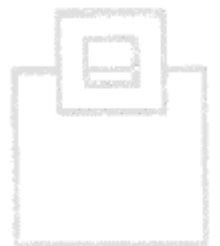
SEGUS Inc

Dynamic SQL at a glance



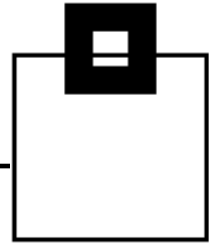
Characteristics:

- It's flexible
 - SQL statements can be built and executed on the fly
- It's dynamic
 - access paths are determined ad-hoc
- It's state of the art
 - widely supported in today's programming languages
- It's difficult to control
 - Statement and access path is only available at runtime
- It's expensive
 - Optimization and tuning is difficult

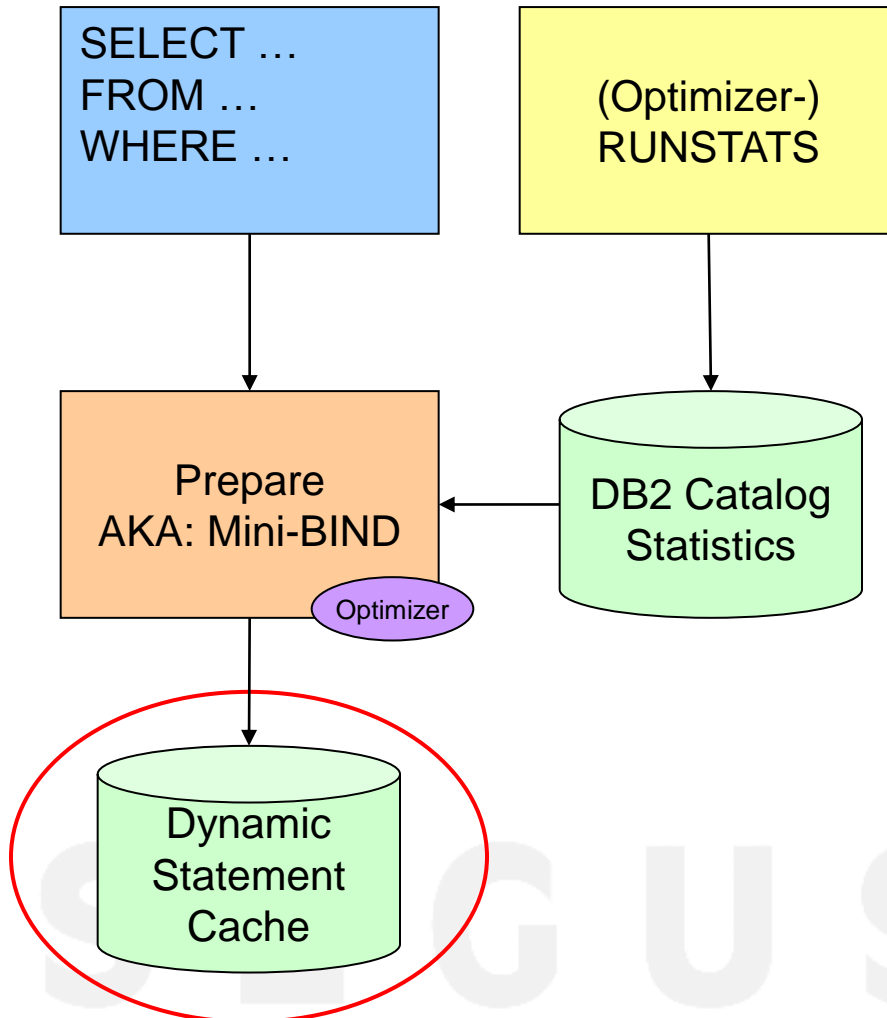


SEGUS Inc

Dynamic SQL at a glance



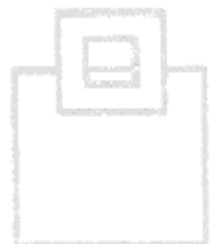
Characteristics:



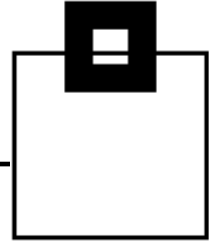
Access Paths for dynamic SQL are determined on the fly and can be stored in the DSC.



RUNSTATS, ALTERS, DB2 RESTART invalidates and flushes the DSC for an object.



Dynamic SQL at a glance



DB2 Setup and Support:

CACHEDYN->

NO

K NO

E

E

P

D

Y

N

A

M

I

C

NO

YES

- No skeletons cached in EDMP
- Only full prepares
- No prepared statements kept across commits (note1)
- No statement strings kept across commits

NONE

- No skeletons cached in EDMP
- Only full prepares
- No prepared statements kept across commits (note 1)
- Stmt strings kept across commits – implicit prepares

LOCAL

YES

- Skeletons cached in EDMP
- 1st prepare full; others short (note 2)
- No prepared statements kept across commits (note 1)
- No statement strings kept across commits

Global

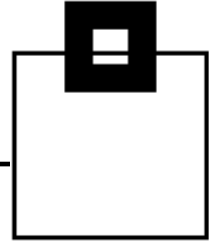
- Skeletons cached in EDMP
- 1st prepare full; others short (note 2)
- Prepared stmts across commits – avoids prepares (note 3)
- Stmt strings kept across commits – implicit prepares

FULL

AKA: Prepare Avoidance

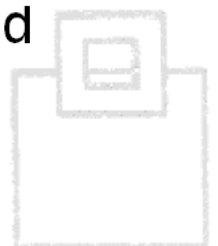
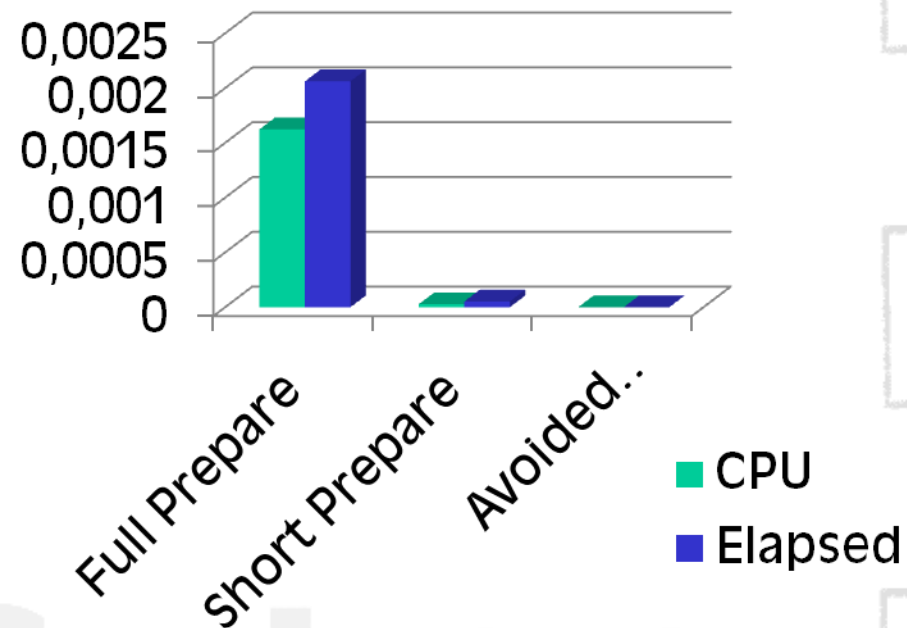
Note 1: unless a cursor WITH HOLD is open, Note 2: unless invalidated or flushed out due to LRU, Note 3: assuming MAXKEEPD > 0

Dynamic SQL at a glance



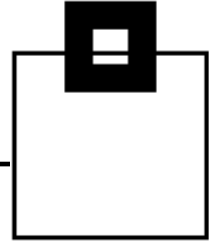
DB2 Setup and Support:

- Save money with the right setup
- Exploit the full flavor of caching
 - MAXKEEPD>0
 - CACHEDYN=YES
 - KEEPDYNAMIC(YES)



SEGUS Inc

Dynamic SQL at a glance



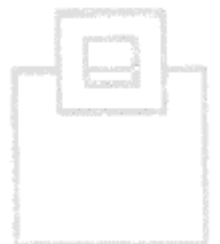
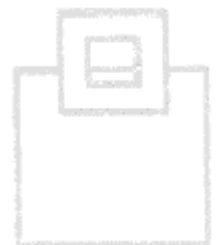
DB2 Commands and Features:

Different to static SQL, the dynamic world allows to “re-explain” an access path using for example SPUFI, DSNTDP2:

- EXPLAIN STMTCACHE ALL
- EXPLAIN STMTCACHE STMTID
- EXPLAIN STMTCACHE STMTTOKEN

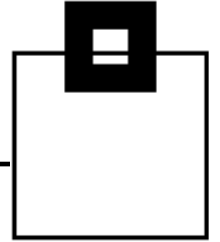


→ EXPLAIN STMTCACHE does not go through the EXPLAIN process, but tells you exactly about the current access!



SEGUS Inc

Dynamic SQL at a glance



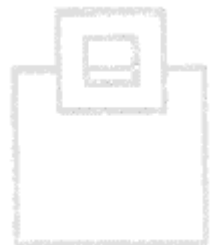
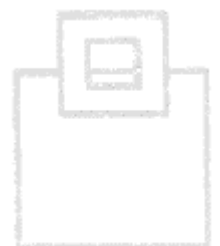
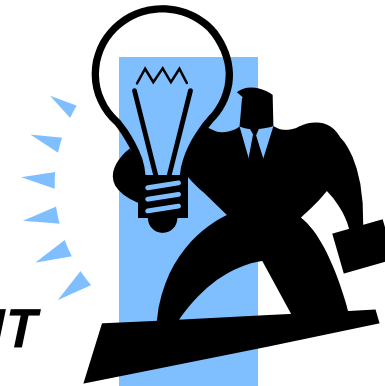
So far, so good ...

Knowing how to handle it, opens up great opportunities for

- Packaged applications like SAP
- Less-mainframe-skilled developers
- Interactive multi-platform solutions
- The mainframe competing with the distributed environment
- Cost efficient and well performing applications

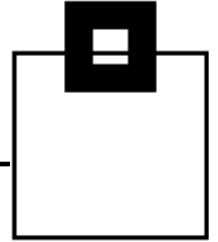
→ **The key is the Dynamic Statement Cache ...**

... USE IT



SEGUS Inc

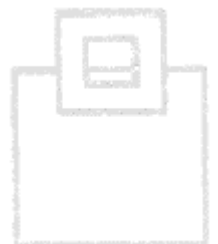
See the possibilities



How SEGUS manages dynamic SQL reliably:

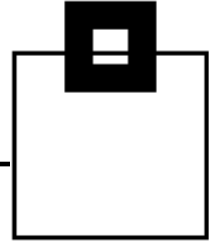


- Dynamic SQL management and protection:
- Protecting your production environment from unforeseen performance degradations requires quality assurance.
- SEGUS trend analysis system allows to pre-check the results from a
 - New application version
 - New statistics
 - REORG with Inline Statistics
 - RUNSTATS TABLESPACE/INDEX
 - LOAD with Inline Statistics
 - New DB2 version (or APARs affecting performance)

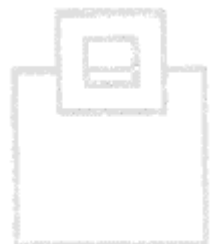
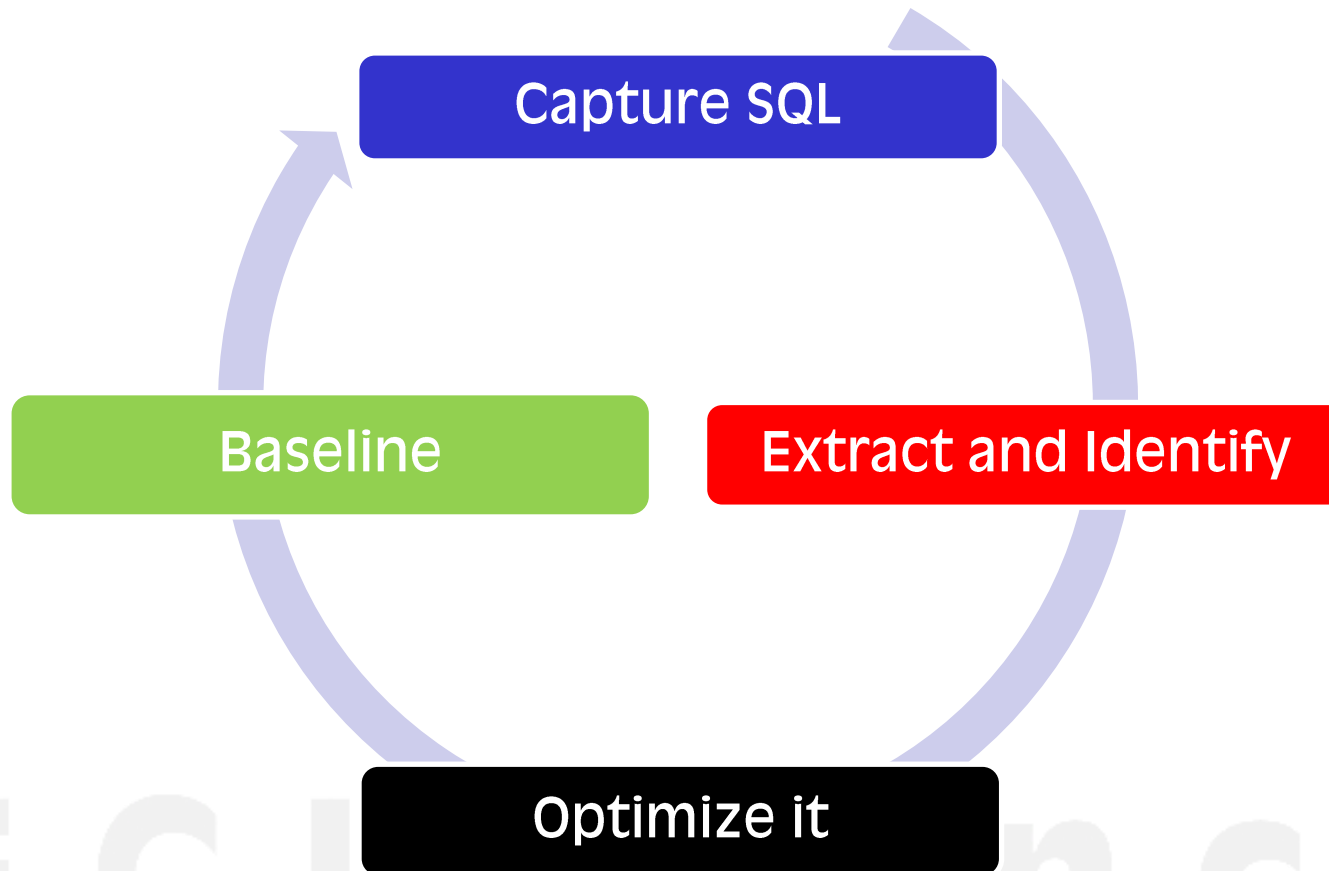


SEGUS Inc

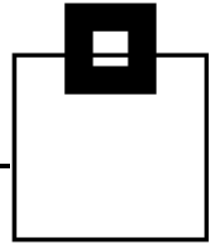
See the possibilities



How to manage dynamic SQL reliably:



See the possibilities



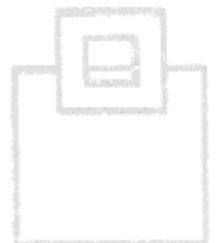
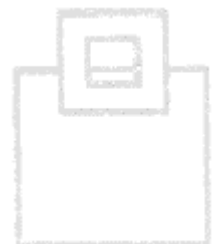
ImpactExpert for DB2 z/OS -- Dynamic Statement Cache (1/8) -- Stmt 1 from 117

Command ==> _____ Scroll ==> CSR
DB2: Q91A

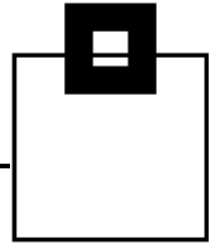
Primary cmd: END, F(ilter), Z(oom), L(ocate) getpages

Line cmd: Z(oom), A(nalyze), E(dit statement), S(tatement text), T(able),
X(EXecute)

	StmtID	Program	Lineno	UserID	Qualifier	Executes	Getpages	S
	-----	-----	-----	-----	-----	-----	-----	-
-	2162	IQADBACP	1086	NEWMANN	NEWMANN	14	245	V
-	2164	IQADBACP	1094	NEWMANN	NEWMANN	36	222	V
-	2152	IQADBACP	1086	NEWMANN	NEWMANN	3	61	V
-	2154	IQADBACP	1086	NEWMANN	NEWMANN	7	48	V
-	2247	IQADBACP	1042	NEWMANN	NEWMANN	1	48	V
-	2250	IQADBACP	1042	NEWMANN	NEWMANN	1	48	V
-	2192	IQADBACP	1082	NEWMANN	NEWMANN	10	47	V
-	2208	IQADBACP	1042	NEWMANN	NEWMANN	1	47	V
-	2138	IQADBACP	1082	NEWMANN	NEWMANN	12	39	V
-	2150	IQADBACP	1086	NEWMANN	NEWMANN	3	24	V
-	2155	IQADBACP	1086	NEWMANN	NEWMANN	7	24	V
-	2253	IQADBACP	1022	NEWMANN	NEWMANN	1	23	V
-	2255	IQADBACP	1090	NEWMANN	NEWMANN	3	21	V
-	2256	IQADBACP	1094	NEWMANN	NEWMANN	4	20	V
-	2142	IQADBACP	1086	NEWMANN	NEWMANN	8	18	V
-	2112	IQADBACP	1082	NEWMANN	NEWMANN	0	16	V



See the possibilities

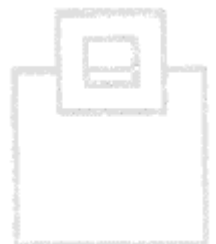
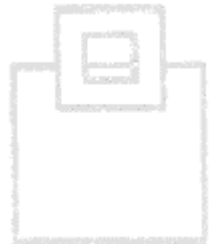


```
ImpactExpert for DB2 z/OS -- Dynamic Statement Cache (4/8) -- Stmt 1 from 117
Command ==> _____ Scroll ==> CSR
                                         DB2: Q91A
```

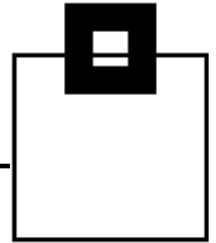
```
Primary cmd: END, F(ilter), Z(oom), L(ocate) getpages
```

```
Line   cmd: Z(oom), A(nalyze), E(dit statement), S(tatement text), T(able),
          X(EXecute)
```

StmtID	Synchr. Buffer Rd	Synchr. Buffer Wr	Rows examined	Rows processed	Index Scans	Tablespc. Scans
2162	0	0	74	37	52	15
2164	0	0	0	185	0	74
2152	4	0	38	19	30	4
2154	0	0	16	8	16	0
2247	0	0	101	2	2	1
2250	0	0	101	2	2	1
2192	0	0	844	2	4	11
2208	4	0	100	39	2	1
2138	0	0	13	13	13	0
2150	0	0	8	4	8	0
2155	0	0	8	0	8	0
2253	0	0	3	1	0	0
2255	0	0	0	7	3	3
2256	0	0	0	2	0	8
2142	0	0	0	9	0	9
2112	0	0	1	0	0	1



See the possibilities



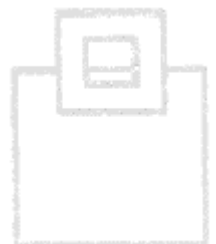
ImpactExpert for DB2 z/OS -- Dynamic Statement Cache (6/8) -- Stmt 1 from 117

Command ==> _____ Scroll ==> CSR
DB2: Q91A

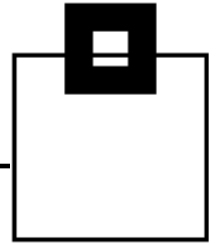
Primary cmd: END, F(ilter), Z(oom), L(ocate) getpages

Line cmd: Z(oom), A(nalyze), E(dit statement), S(tatement text), T(able),
X(EXecute)

StmtID	Total CPU	Average CPU	Total Elapse	Average Elapse
-----	HHHH:MM:SS.ttt	HHHH:MM:SS.ttt	HHHH:MM:SS.ttt	HHHH:MM:SS.ttt
2162	0.040	0.003	0.373	0.027
2164	0.047	0.001	0.128	0.004
2152	0.014	0.005	0.104	0.035
2154	0.007	0.001	0.007	0.001
2247	0.006	0.006	0.006	0.006
2250	0.006	0.006	0.006	0.006
2192	0.005	0.001	0.005	0.001
2208	0.013	0.013	0.089	0.089
2138	0.004	-	0.004	-
2150	0.002	0.001	0.002	0.001
2155	0.002	-	0.002	-
2253	-	-	-	-
2255	0.004	0.001	0.004	0.001
2256	0.004	0.001	0.004	0.001
2142	0.002	-	0.002	-
2112	-	-	-	-



See the possibilities



```
ImpactExpert for DB2 z/OS - Limit DSC Snapshot -----  
Command ==> _____ DB2: Q91A
```

```
Primary cmd: END
```

```
MEMBER      : _____ Blank(Connected DB2) / *(All members) / member name
```

```
NO LIMITATION : X
```

```
HIGHEST VALUES : _
```

```
EXCEED THRESHOLD: _ THRESHOLD: _____
```

```
For limitation to highest values or exceeding of specified threshold
```

```
EXECUTIONS      : _ ROWS PROCESSED      : _ SORTS                      : _
```

```
BUFFER READS    : _ ROWS EXAMINED       : _ PARALLEL GROUPS           : _
```

```
BUFFER WRITES   : _ INDEX SCANS         : _ RID EXCEED DB2 LIMITS   : _
```

```
GETPAGES        : _ TABLE SPACE SCANS : _ RID EXCEED STORAGE      : _
```

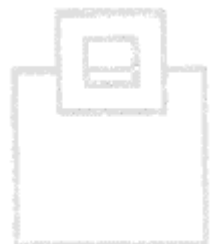
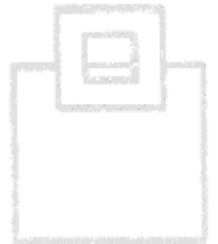
```
For limitation to highest values only
```

```
ELAPSE TIME     : _ CPU TIME             : _
```

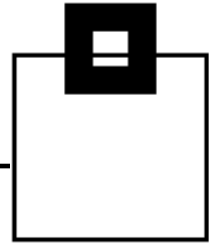
```
WAIT TIME FOR ...
```

```
SYNCHRONOUS I/O : _ SYNCR. EXECUTION : _ READS OTHER THREADS : _
```

```
LOCK AND LATCH  : _ GLOBAL LOCKS   : _ WRITES OTHER THREADS : _
```



See the possibilities



```
ImpactExpert for DB2 z/OS - Filter Dynamic Statement Cache -----
Command ==> _____ DB2: Q91A

Primary cmd: END

FIRST TABLE : _____
CREATOR       : _____

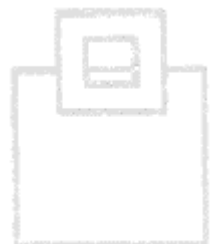
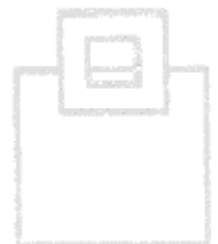
FIRST TABLE : _____
NAME          : _____

QUALIFIER     : _____

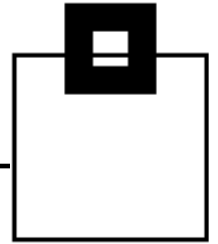
PRIMARY       : _____
AUTHID        : _____

SELECT X      CURRENT USERS   between _____ and _____ (Integer)
INSERT X      STMT COUNT      between _____ and _____ (Integer)
UPDATE X      AVG CPU TIME     between _____ and _____ (MM:SS.TTT)
DELETE X      AVG ELAPSE TIME between _____ and _____ (MM:SS.TTT)
              AVG GETPAGES     between _____ and _____ (Integer)

Total stmts   104
OUTPUT LIMIT: 10000 0 - 25000 Max number of statements to be displayed
```



See the possibilities



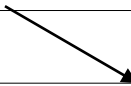
How to manage dynamic SQL reliably:

Step 1 – Find the candidates – you may need to aggregate!

Level 1: Ignore values, spacing, cursor names, select clauses

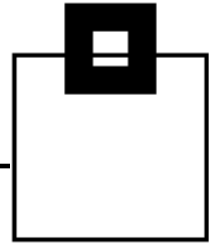
....

SQL-Text	Count	CPU-Time
SELECT ... WHERE COL = 'ABC'	1	1s
SELECT ... WHERE COL = 'BCD'	1	1s
SELECT ... WHERE COL = 'CDE'	1	1s
SELECT ... WHERE COL = 'DEF'	1	1s
SELECT ... WHERE COL = 'EFG'	1	1s
...		



SQL-Text	Count	CPU-Time
SELECT ... WHERE COL = 'ABC'	10,000	10,000s
...		

See the possibilities



How to manage dynamic SQL reliably:

Step 1 – Find the candidates – you may need to aggregate!

... Level 2: Level1 + operators in predicates

Level 3: Level2 + right hand side

Level 4: Aggregate on object level

Optionally (for all levels): table creator

```
SELECT COLX, COL2 FROM CRE1.TAB1 WHERE COL5 = 'ABC'
SELECT COL1, COL2 FROM CRE1.TAB1 WHERE COL5 = '123' => Level 1
SELECT COL1, COL2 FROM CRE1.TAB1 WHERE COL5 > '123' => Level 2
SELECT COL1, COL2 FROM CRE1.TAB1 WHERE COL5 > :HV1 => Level 3
SELECT COL1, COL2 FROM CRE1.TAB1 WHERE COL7 > :HV1 => Level 4
SELECT COL1, COL2 FROM CRE2.TAB1 WHERE COL5 > :HV1 => Level 3
                                                    + tbcreator
```

SELECT COL1, COL2 FROM CRE1.TAB1 WHERE COL = 'ABC'

See the possibilities

ImpactExpert for DB2 z/OS - Explain Data (1/5) ----- Entry 1 from 1
Command ==> _____ Scroll ==> CSR

DB2: Q91A

Primary cmd: END, T(Explain Text), V(iolations), R(unstats), P(redicates),
S(tatement Text), PR(int Reports), Z(oom), SAVExxx, SHOWxxx

Line cmd: Z(oom), I(ndexes of table), S(hort catalog), T(able), X(Index)

DSN = NEWMANN.ADB2.IN

Member = DELME

Stmt = 1

Milliseconds: 433 Service Units: 1696 Cost Category: B

QBNO	QBTYP	CREATOR	TABLE NAME	ACCS	MTCH	IX	METH	PRNT	TABL	PRE	MXO
PLNO	TABNO	XCREATOR	INDEX NAME	TYPE	COLS	ON	OD	QBLK	TYPE	FTCH	PSQ
1	SELECT	SYSIBM	SYSTABLES	R	0	N	0	0	T	S	0
1	1										

See the possibilities

ImpactExpert for DB2 z/OS -- Index Overview ----- Index 1 from 3

Command ==> _____ Scroll ==> CSR
DB2: Q91A

Primary cmd: END, CAN(cel), SE(tup), Z(oom), L(ocate) creator

Line cmd: C(olumns), D(atabase), K(PacKages), P(artitions), T(able), Z(oom)

	IX Creator	IX Name	Created by	Created timestamp
	TB Creator	TB Name	Database Indexspace	Statstime
-	SYSIBM	DSNDTX01	SYSIBM	0001-01-01-00.00.00.000000
	SYSIBM	SYSTABLES	DSNDB06 DSNDTX01	0001-01-01-00.00.00.000000
-	SYSIBM	DSNDTX02	SYSIBM	0001-01-01-00.00.00.000000
	SYSIBM	SYSTABLES	DSNDB06 DSNDTX02	0001-01-01-00.00.00.000000
-	SYSIBM	DSNDTX03	SYSIBM	2003-09-21-23.27.05.275288
	SYSIBM	SYSTABLES	DSNDB06 DSNDTX03	0001-01-01-00.00.00.000000

See the possibilities

```
Analyze for DB2 z/OS ----- Violations ----- LINE 00000001 COL 001 080
Command ==> _____ Scroll ==> CSR
                                         DB2: Q91A

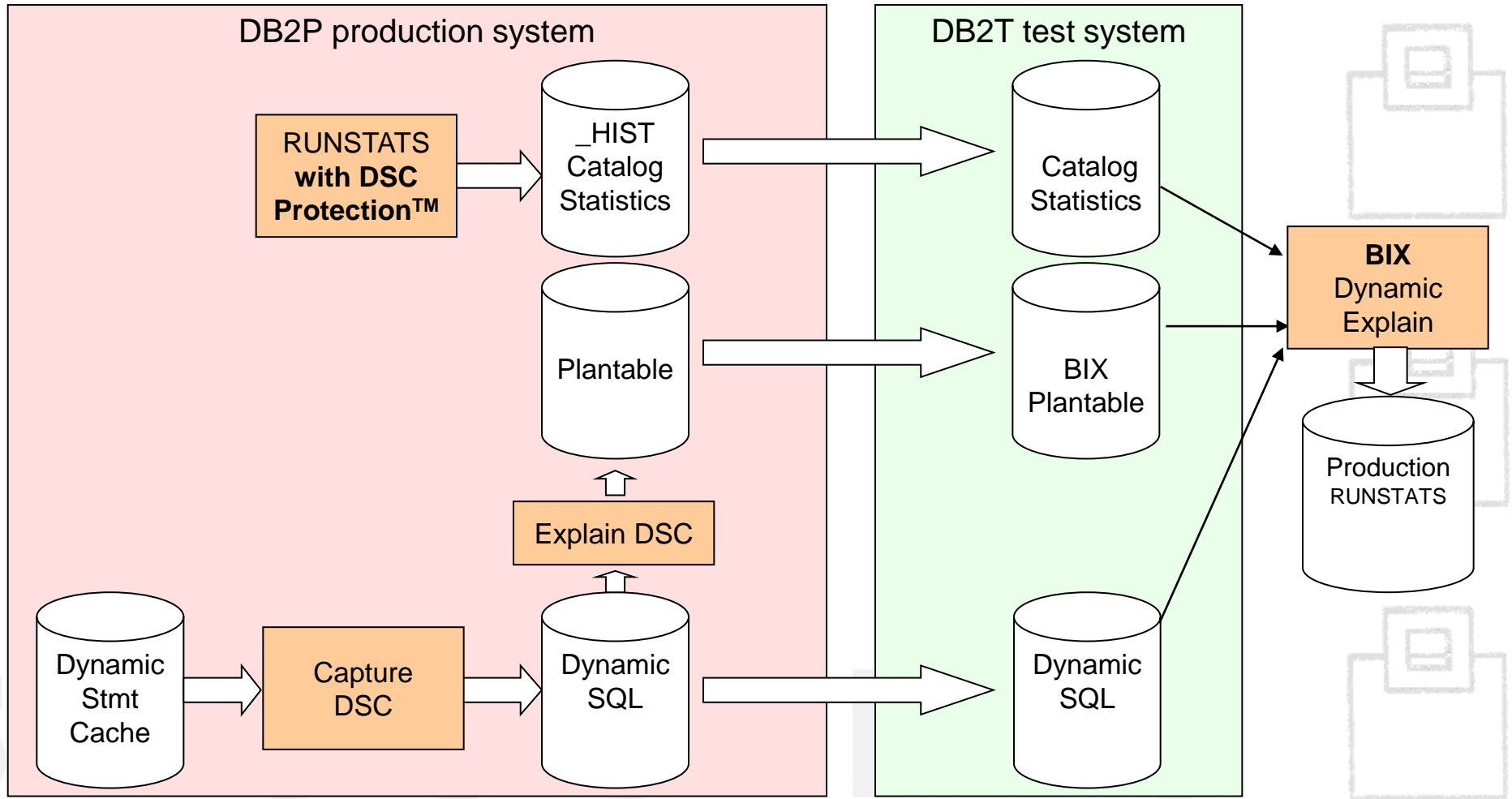
Primary cmd: END, E(xplain Data), T(Explain Text), R(unstats), P(predicates),
             S(statement Text), PR(int Reports), SAVExxx, SHOWxxx

DSN = NEWMANN.ADB2.IN                      MEMBER = DELME
STMT = 1

-----
----- RULE-NO.: 9072 (WARNING) -----
Predicate is stage 2 (neither stage 1 nor indexable). QBLOCKNO: 1, Access:
STAGE2, Predicate: 5 BETWEEN SYSIBM.SYSTABLES.DBID AND SYSIBM.SYSTABLES.OBID
Try to rewrite the predicate as stage 1 or indexable or try to add another (
stage 1 or indexable) predicate for this column(s) to the WHERE or ON clause.
----- RULE-NO.: 9201 (WARNING) -----
A predicate like: '(EXPR) BETWEEN COL1 AND COL2' should be rewritten like:
'(EXPR) >= COL1 AND (EXPR) <= COL2'.
Then the predicates are INDEXABLE.
----- RULE-NO.: 9065 (WARNING) -----
SELECT * can lead to unnecessary data transfer. QBLOCKNO(s) affected: 1.
Select only columns which are really used by your application.
----- RULE-NO.: 9070 (SEVERE-ERROR) -----
Runstats check found critical rule violations.
Please look into the runstats report.
----- RULE-NO.: 9099 (WARNING) -----
```

See the possibilities

How SEGUS manages dynamic SQL reliably: Dynamic SQL management and protection: Proactive QA



See the possibilities

```
ImpactExpert for DB2 z/OS ----- Job Overview (1/4) ----- Job 1 from 1
C +-----+ CSR
M | ----- Export ----- | Q91A
P | Command ==> _____ |
  | Primary cmd: END         |
L |                           |
  | _ Extract production DDL using DDL-Generator (optional) |
  |                                                           |
  | _ Extract production catalog statistics using ProductionSimulator | tatus
  |                                                           | -----
  | _ Extract history statistics using ProductionSimulator         | -----
  |                                                           |
  | _ Capture and explain dynamic statement cache on production   |
  |                                                           |
  | _ Export dynamic SQL from production system                   |
  |                                                           |
  +-----+-----+
```

See the possibilities

```
ImpactExpert for DB2 z/OS ----- Job Overview (1/4) ----- Job 1 from 1
C +-----+-----+-----+-----+-----+-----+-----+-----+CSR
M | ----- Import ----- |Q91A
P | Command ==> _____ |
  | Primary cmd: END       |
L | _____ |
  | _ Apply production DDL into test environment using DDL-Generator |
  | _____ |
  | _ Apply history statistics into test using ProductionSimulator |atus
  | _____ |
  | _ Import dynamic SQL into test environment                     |----
  | _____ |
  | _____ |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

SEGUS Inc

See the possibilities

```
ImpactExpert for DB2 z/OS ----- Job Overview (1/4) ----- Job 1 from 1
Command ==> _____ Scroll ==> CSR
Mode: DSC PROTECTION DB2: Q91A
Primary cmd: END, A(11), D(SC Extracts), E(Imported DSC Extracts), N(ew),
             I(mport), X(EXport), R(efresh), Z(oom), L(ocate) submit time
Line  cmd: S(tatements), A(uthIDs), D(elete), R(eset statistics),
             V(iew extract), Z(oom)
```

Job name	Step name	Submitter	Submit time	PLTB owner	Job status
BAIMDSC	CHECKDYN	HOPPE	2010-07-27-15.13.57	HOPPE	DONE

SEGUS Inc

See the possibilities

ImpactExpert for DB2 z/OS ----- Statements Summary ----- Status 1 from 6

Command ==> _____ Scroll ==> CSR

Mode: DSC PROTECTION

DB2: Q91A

Primary cmd: END, R(efresh), Z(oom)

Line cmd: S(elect), R(eset statistics)

AUTHID : *

Status	Statements
-----	-----
PROCESSED	286
- IMPROVED	1
- UNCHANGED	282
- WORSENER	3
NON-DETERMINABLE	0
ERROR	0
-----	-----

SEGUS Inc

See the possibilities

ImpactExpert for DB2 z/OS -- Dynamic SQL Overview (1/12) ----- Stmt 1 from 3

Command ==> _____ Scroll ==> CSR

Mode: DSC PROTECTION

DB2: Q91A

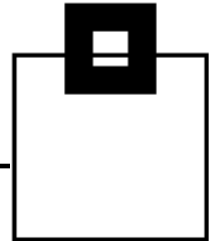
Primary cmd: END, SE(tup Analyze), Z(oom), L(ocate) stmtid old

Line cmd: S(elect), DO/DN(Dynamic analyze Old/New), R(eset statistics),
EO/EN(Edit and analyze Old/New), VO/VN(View Old/New), Z(oom)

			Bad access types		Bad access types	
			old		new	
StmtID	old	StmtID new	Imp	old		new
-----	-----	-----	---	-----	-----	-----
---	90	90	WRS	NMIX		LP, SORT
---	93	93	WRS	NMIX		LP, SORT
---	250	250	WRS	TS, LP		TS, NMIX, MSJN
-----	-----	-----	---	-----	-----	-----

SEGUS Inc

See the possibilities



```
ImpactExpert for DB2 z/OS ----- Access Paths ----- LINE 00000001 COL 001 080
Command ==> _____ Scroll ==> CSR
Mode: DSC PROTECTION DB2: Q91A
Primary cmd: END, C(atalog data), D(etails on/off), S(tatement text)
```

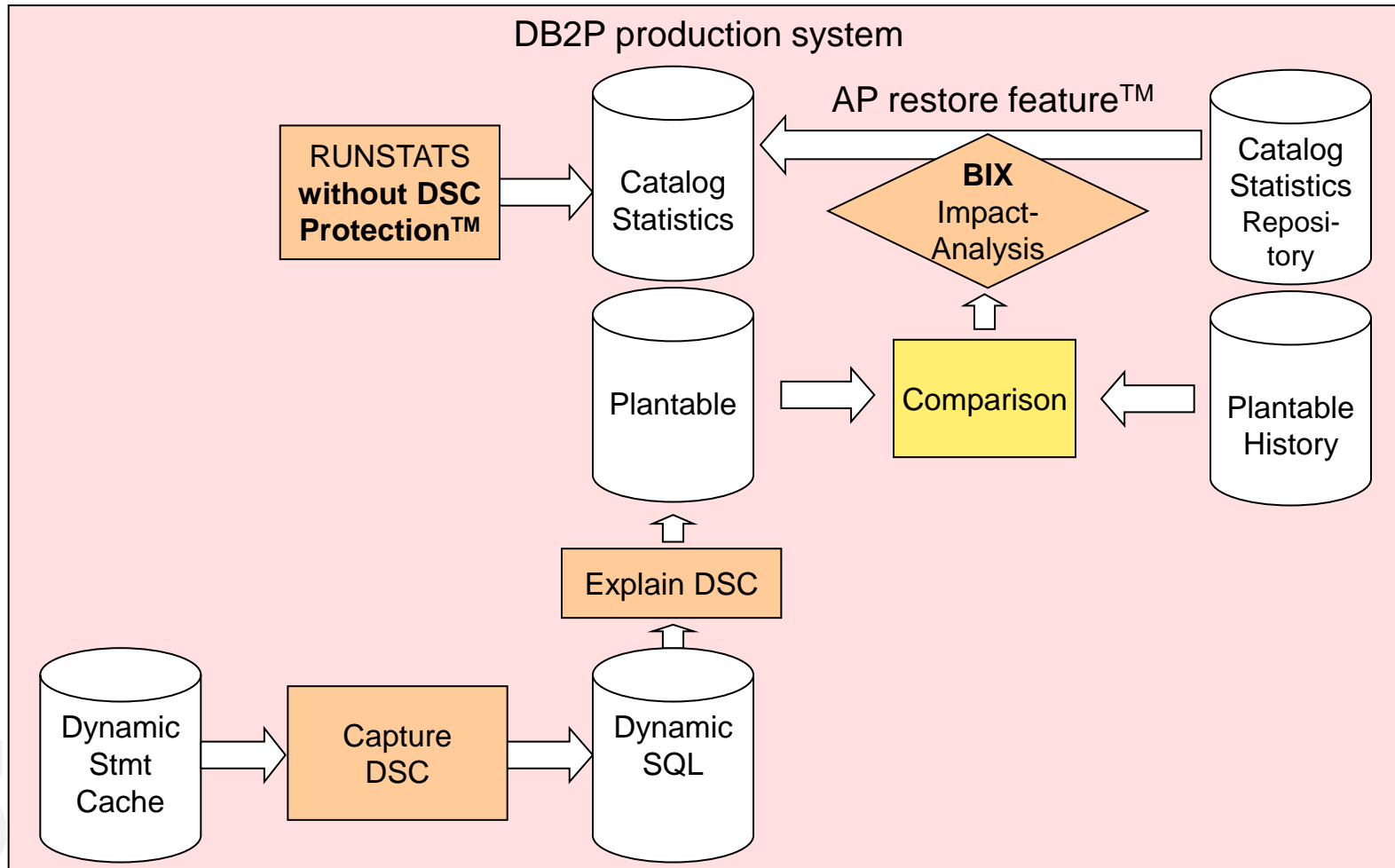
```
RunID old . TEST-TSPCE RunID new . TEST-TSPC
Created TS. 2010-07-26-17.51.57.257322 Created TS. 2010-07-26-17.51.57.257322
StmtID old. 93 StmtID new. 93
ExplainID . 1 ExplainID . 2
```

Access path OLD -----										Access path NEW -----									
TABLE	QB	PN	AC	MA	ME	IX	PR	TABLE	QB	PN	AC	MA	ME	IX	PR				
INDEX			TY	CO	TH	ON	FT	INDEX			TY	CO	TH	ON	FT				
IQATB003	1	1	I	1			N	IQATB003	1	1	I	1			N				
IQAXB0031								IQAXB0031											
IQATB005	1	2	I	0	1		N	IQATB005	1	2	I	1	1	N	L				
IQAXB0051								IQAXB0053											
IQATH001	1	3	I	2	1		N	IQATH001	1	3	I	2	1	N					
IQAXH0011								IQAXH0011											
									1	4		0	3	N					
Milliseconds:			1					Milliseconds:			3								
Serviceunits:			2					Serviceunits:			5								



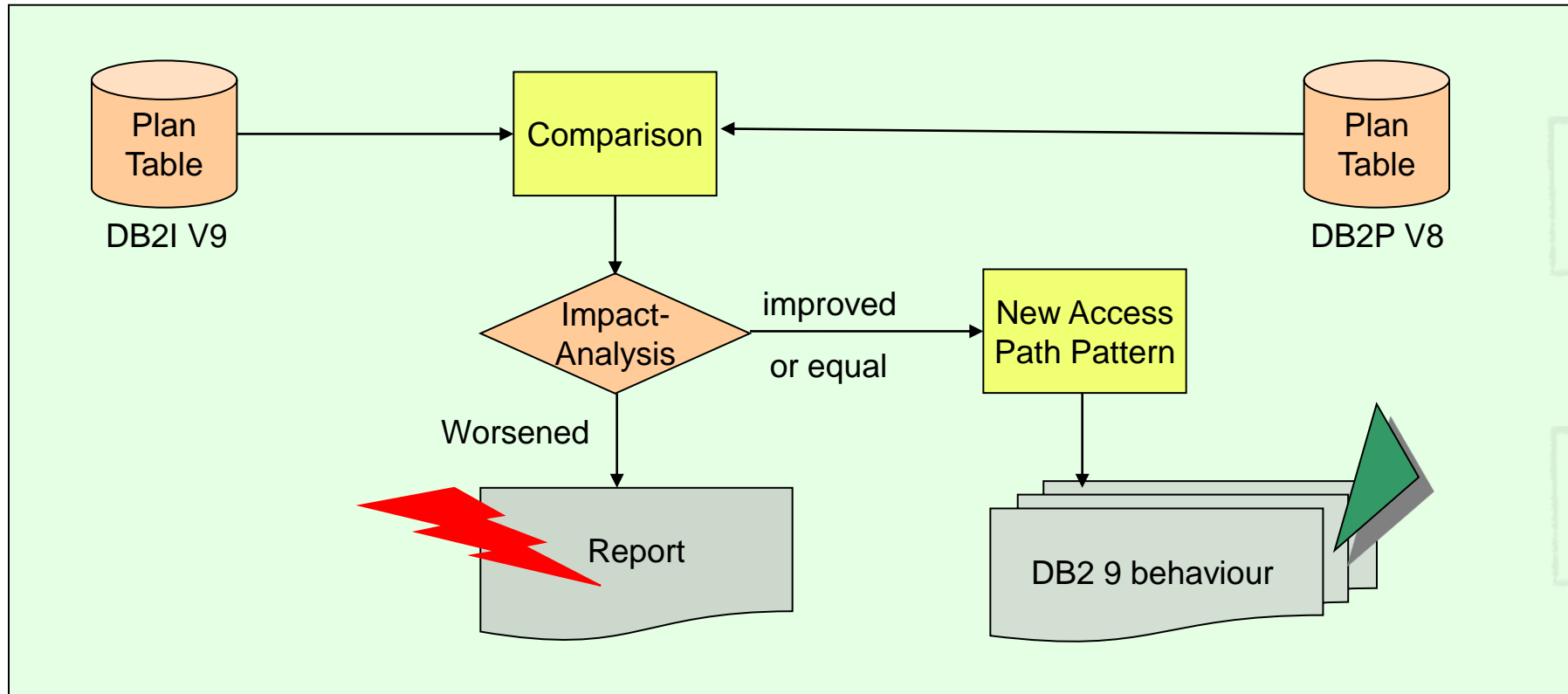
See the possibilities

How SEGUS manages dynamic SQL reliably: Dynamic SQL management and protection: Reactive QA



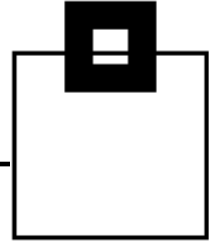
See the possibilities

■ BIX – Version Migration control (**EarlyPrecheck™**)



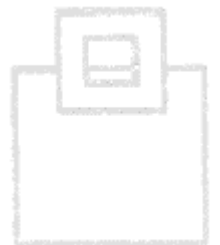
SEGUS Inc

See the possibilities

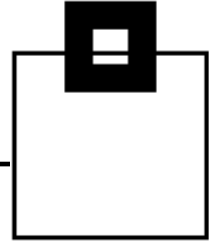


What can you expect from  **Bind ImpactExpert:**

- Flexibility in developing and running your applications
- Same level of insight & protection like in your static world
- Proactive and/or reactive procedures following your needs
- Cost efficiency in development and operations



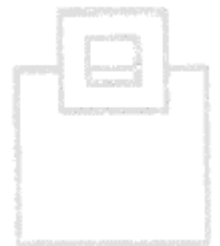
SEGUS Inc



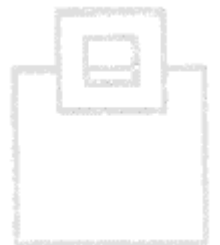
Questions?



Roy Boxwell: r.boxwell@seg.de



Ulf Heinrich : u.heinrich@segus.com



SEGUS Inc