



DB2 V9 Application Migration Based on V8 Experiences

Gerald S Hodge
HLS Technologies, Inc.

Agenda



- Some Background
- Orwellian Truth
- Optimization Service Center
- DSNZPARMs
- CCSID and recompiling
- V9 Access Path Experiences
- Appendixes

Background



- Member of DB2 Gold Team since 1996
- Member of PWD since 1994
- Company has provided rebranded DB2 tools to IBM since 1998
- Participated in over 87 V7 to V8 conversions
- Company has its own Z9 running Z/OS 1.9
- Company runs DB2 V7, V8 and V9 for development

Orwellian Truth



- We are told to completely rebind in V8
- We are told to completely rebind in V9
- This is what we were told to do for V7 to V8
- IBM is no longer stating this for V7 to V8
- An assessment of your business risks is required

Optimization Service Center

- New with V9 and replaces Visual Explain
- Retrofitted to V8
- OSC APIs are available to everyone
- A purchased product is required for full IBM OSC functionality
- What is it and how does it work?

OSC Requires Packages and Tables

Subsystem Properties

Connect to subsystem
Specify the authorization information for the subsystem and click Connect. When the subsystem is

Steps

1. Location
2. Connect
3. Manage Users

Complete the following fields and click Connect.

User ID:

Password:

SQL ID:

Remember login ID

The following fields describe the OSC status of this subsystem:

Connected:	✓	Connected
OSC packages bound:	✓	Bound
EXPLAIN enabled:	✓	BAKER1 is enabled
OSC Tables:	✓	Available

< Back Next > Finish Cancel

OSC DBA Requirement

Subsystem Properties

Manage Users

To grant access, type the authorization ID, select one or more functions, and click Add. To change access, select the authorization ID in the table, select only the functions that the user should access,

Authorization IDs that have SYSADM authority but have not been granted explicit privileges are not listed.

Authorizat...	OSC Basic	Query Annotation	Workload Control Center or Monitor	Workload Statistics Advisor	Stored Procedure for Workload Control Center
BAKER1	✓	✓		✓	
IBMUSER	✓	✓	✓	✓	✓
P390H	✓	✓	✓	✓	✓

Type an authorization ID to add, or select an authorization ID from the table. Select functions to enable for the authorization ID.

Authorization IDs:

Functions:

- Service Center
 - OSC Basic
 - Query Annotation
- Design Center
 - Workload Control Center or Monitor
 - Workload Statistics Advisor
 - Stored Procedure for Workload Control Center

Update Remove Clear

< Back Next > Finish Cancel

OSC DSNZPARM View Selection

▼ DB2 Subsystem Configurations

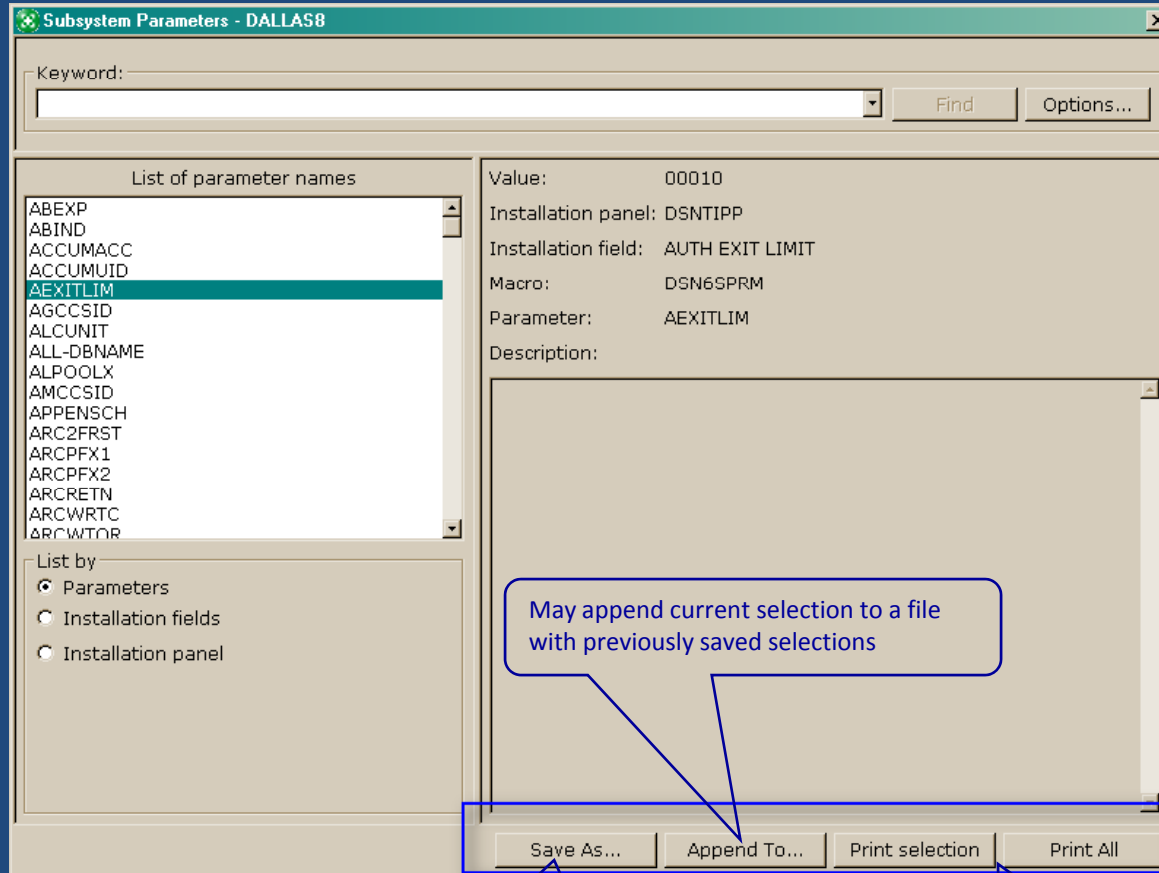
To add a subsystem, click Subsystem -> Add. To Connect to or enable a subsystem in the list, select the subsystem and click Connection -> Connect.

Subsystem ▾ Connection ▾ **Browse Subsystem Parameters** View Query Activity Manage Users

Subsystem	Status	Browse Subsystem Parameters		Port Number	User ID	SQLID	Comment
DALLAS7	<input type="checkbox"/> Disconnected	DALLAS7	66.228.132.34	446			DB2 Version 7 DBRM
DALLAS8	<input type="checkbox"/> Disconnected	DALLAS8	66.228.132.34	5023			DB2 Version 8 data...
DALLAS9	<input checked="" type="checkbox"/> OSC Enabled	DALLAS9	66.228.132.34	5025	BAKER1	BAKER1	DB2 Version 9 data...
HRDATA	<input type="checkbox"/> Disconnected	HRDATA	192.168.0.3	50100			
HRDEMO	<input type="checkbox"/> Disconnected	HRDEMO	192.168.0.3	50100			
OSCTUTORIAL	<input type="checkbox"/> Disconnected	DSNOSCDE...	OSCTUTORIAL	446			OSC Tutorial

- Click 'Browse Subsystem Parameters' to see DSNZPARMS
- May not be available for V9

OSC DSNZPARM View

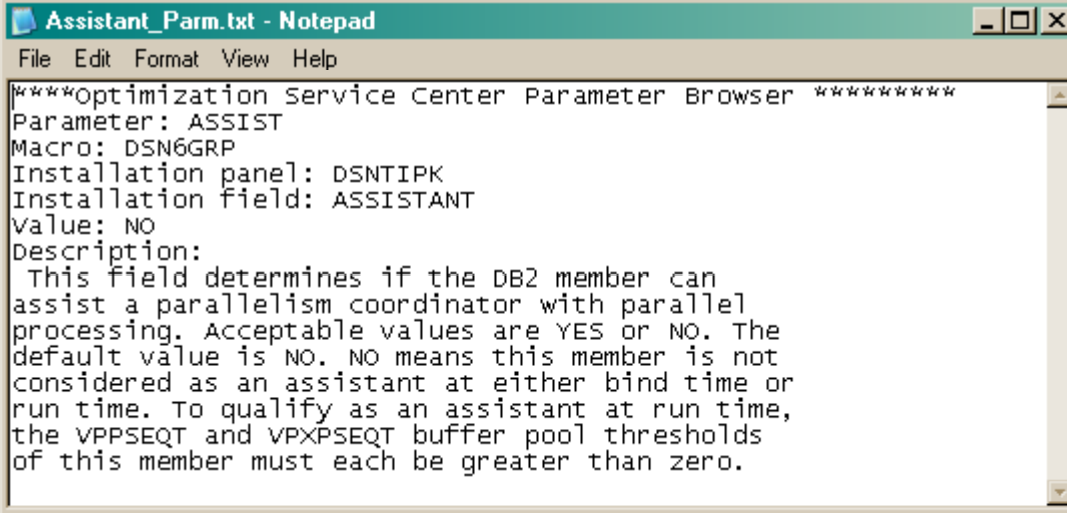


- Over 20 missing definitions
- No ability to compare parameters between versions

Only saves currently selected parameter settings and description

Print selected parameter or all listed parameters

OSC Saved Parameter Contents



```
Assistant_Parm.txt - Notepad
File Edit Format View Help
****Optimization Service Center Parameter Browser ****
Parameter: ASSIST
Macro: DSN6GRP
Installation panel: DSNTIPK
Installation field: ASSISTANT
Value: NO
Description:
  This field determines if the DB2 member can
  assist a parallelism coordinator with parallel
  processing. Acceptable values are YES or NO. The
  default value is NO. NO means this member is not
  considered as an assistant at either bind time or
  run time. To qualify as an assistant at run time,
  the VPPSEQT and VPXPSEQT buffer pool thresholds
  of this member must each be greater than zero.
```

- Note that you have the option to save the parameter settings where there is no description available

Describe View After Connection

The screenshot shows the 'Describe 3.0 from HLS Technologies' application window. At the top is a menu bar with 'File', 'Edit', 'Options', and 'Help'. Below the menu is a table with the following columns: Alias, Database, Node, IP Address, Port, and Comment. The table contains six rows of data. Below the table is a list of parameters, with 'AEXITLIM' selected. To the right of the list, the details for 'AEXITLIM' are displayed: Value: 00010, Install Panel: DSNTIPP, Install Field: AUTH EXIT LIMIT, Macro: DSN6SPRM, and Parameter: AEXITLIM. At the bottom left, there are radio buttons for 'Parameters', 'Install Fields', and 'Install Panels', with 'Parameters' selected. On the right, a 'Description:' field contains the text: 'Externalizes SPRMACAN as AEXITLIM. This is the number of ABENDS of the DB2 Access Control Authorization Exit before shutdown. Default is 10.' A blue callout box with a pointer to the 'AEXITLIM' parameter in the list contains the text: 'Compare with AEXITLIM description from earlier OSC slide'.

Alias	Database	Node	IP Address	Port	Comment
DALLAS7	DCS45CC1	NDE45038	66.228.132.34	446	DB2 Version 7 DBRM
DALLAS8	DALLAS8	NDE70DA	66.228.132.34	5023	DB2 Version 8 database
DALLAS9	DCSB8785	NDEB7BC1	66.228.132.34	5025	DB2 Version 9 database
HRDATA	HRDATA	NDE53988	192.168.0.3	50100	
HRDEMO	HRDEMO	NDE53988	192.168.0.3	50100	

Value: 00010
Install Panel: DSNTIPP
Install Field: AUTH EXIT LIMIT
Macro: DSN6SPRM
Parameter: AEXITLIM

Description:
Externalizes SPRMACAN as AEXITLIM. This is the number of ABENDS of the DB2 Access Control Authorization Exit before shutdown.
Default is 10.

Compare with AEXITLIM description from earlier OSC slide

Optimization Service Center

- We were not able to get V9 DSNZPARMs to display
- The issue was not with DSNWZP as that works in V9
- OSC replaces Visual Explain in V9 and Visual Explain is no longer supported beyond V8
- Target appears to be SYSDBAs

DSNZPARAMs



- In the Appendix is a comparison of one of our V8 Subsystems against a V9 Subsystem
- We offer a FREE JAVA program to examine and compare DSNZPARAMs
- We even provide the source code

Describe Compare Panel

- Compare contents of panel with contents of saved text file
- Save or print comparison table
- Export comparison as an MS Excel spreadsheet

Parameters	Panel Contents	File Contents
APPENSCH	UNICODE	EBCDIC
ARCPFX1	DSN910.DB9G.ARCLOG1	DSN810.ARCHLOG1
ARCPFX2	DSN910.DB9G.ARCLOG2	DSN810.ARCHLOG2
ARCRETN	00999	09999
CACHEDYN_FREELOCAL	00001	00000
CATALOG[DSN6SPRM]	DSN910	DSN810
CHKFREQ	0000050000	0000500000
DEFLTID	PUBLIC	IBMUSER
DSMAX	0000009960	0000010000
EDMDBDC	0000011700	0104857600
EDMPOOL	0000018142	0033554432
EDMSTMTC	0000056693	0104857600
ENSCHHEME	UNICODE	EBCDIC
GRPNAME	DSNCAT	
IDTHTOIN	00300	00120
IRLMPRC	DB9GIRLM	DB8GIRLM
IRLMSID	DJ9G	DJ8G

DSNZPARAMs and Optimization in V9

- MAX_OPT_CPU (Default 100)
- MAX_OPT_ELAP (No longer supported)
- MAX_OPT_STOR (Default is 40)
- OPTHINTS (Default is NO)
- OPTIXIO (Default is ON)
- OPTIXOPREF (Default is OFF)
- OPTPREF (no longer supported)

DSNZPARM and Optimization in V9

- OPTSUBQ1 (No Longer Supported)
- OPTXQB (Default is ON)
- DESCSTAT (Default is YES)
- SJTABLES (Default is 10)
- STARJOIN (Default is Disable)
- STATCLUS (Default is Enhanced)

CCSID and Recompiling

- CCSIDs are mandatory with V8
- Rebind in V7 to V8 and V8 to V9 should catch any DBRMs that are not rebindable going forward
- Rebind does not catch issues that can occur with CCSID changes that occur with a compile
- If you went from 0 to 500 this should not be an issue, otherwise...

Customer Example

- Recompiled in V7 and then V8
- “No Changes” to the source code
- CCSIDs did change
- SQL comparison found changes
- This is more likely with PL1, but this example is from a COBOL program

Code Comparison



- OLD DBRM RECORD:
- | E4E2405F 7E407DF0 7D40D6D9 C4C5D940 | US ~~4~~'0' ORDER |
- | C2E840D6 D9C4D9C5 C6406B40 D2C5E86D | BY ORDREF , KEY_ |
- | D9D8E2E3 6DD5D640 00000000 40404040 | RQST_NO |
- | 40404040 40404040 40404040 40404040 | |
- | 40404040 40404040 40404040 40404040 | |

- NEW DBRM RECORD:
- | E4E240B0 7E407DF0 7D40D6D9 C4C5D940 | US ~~4~~'0' ORDER |
- | C2E840D6 D9C4D9C5 C6406B40 D2C5E86D | BY ORDREF , KEY_ |
- | D9D8E2E3 6DD5D640 00000000 40404040 | RQST_NO |
- | 40404040 40404040 40404040 40404040 | |
- | 40404040 40404040 40404040 40404040 | |

Impact of Recompiling

- Rebinding is not your only exposure
- Recompiling for any reason presents an exposure
- How complete is your testing?
- What are your exposures?
- Be aware that sort sequencing may change

V9 Access Path Comparisons

Cost contrasts between DB2 Versions

- The IBM Optimizer People and Consultation Staff agree that I am completely wrong on this issue
- Because the methods for calculating COST are different from DB2 Version to DB2 Version one can not judge changes in the COST estimation to be relevant
- Unless the COST decreases

V9 Access Path Comparisons

Cost Contrasts between DB2 Versions

- If the COST value goes up by 50% or higher then Something has changed
 - 1.) The stats are different
 - 2.) The Buffer Pool setup is different
 - 3.) The Indexes are different
 - 4.) Access Path definitions have changed

V9 Access Path Comparisons Strawman and Development Systems

- Check Statistics
- Check Buffer Pool Setup
- Check Indexes
- Check SPRMCPU value in DSNZPARM
- Check order of tables in Joins
- Check point by point Plan Tables Columns

V9 Access Path Comparisons Production Systems

- Access Path most likely change
- Check order of tables in Joins
- Check point by point Plan Table Columns
- Most Access Path Changes are for the better
- Have a HINT Strategy
- Hinting back to an old Access Path will most likely give performance stability

V9 Access Path Comparisons

Hodge's Third Law

If anything of significance goes wrong, then it does not matter how many things went well.

V9 Access Path Comparisons

Corollary to Hodge's Third Law

- Your first objective in a DB2 migration is to stay employed
- Your second objective in a DB2 migration is to stay in business

V9 Access Path Comparisons

What we are seeing

- Our Testing
 - Seventeen test cases run on V5, V6, V7, V8, and V9
 - Thirteen are discussed here
 - Because you can recreate the tests, you can contrast your situation with our results
 - SPUI examples are given showing how to find DBRMs that have not been explained
 - A detailed discussion of how to setup the test cases and the test data
 - A large PDF document that details the 17 test cases
 - A ZIP file containing all the data needed to recreated the test cases in your environment
 - Easy to install
 - At www.hlstechnologies.com; You must have or create a Login

What is Reported

- The PLAN_TABLE has the Access Path
- The DSN_STATEMENT_TABLE has the estimated costs
- The DSN_FUNCTION_TABLE has the function information up to V8 where only UDFs are reported
- Remember this information is only for the Access Path selected not the others

Access Path Preference

- Direct Row access using ROWID
- One Fetch Index Scan using Min, MAX
- Unique Matching Index Scan using a predicate value
- Matching Index Scan Only
- Non-Matching Index Scan Only
- Matching Index Cluster Scan
- Matching Random Index Scan

Access Path Preference

- Multiple Matching Index Scan using AND and OR
- Non Matching Cluster Index Scan
- Segmented Table Space Scan
- Non Segmented Table Space Scan (in parallel or sequential)
- Non-matching Random Index Scan

Creating Access Path List

- The list is created by an algorithm
- The most likely to work are first
- Based on that list what might be better is tried
- If MAX_OPT_CPU is exceeded the list is stopped
- If MAX_OPT_ELAP is exceeded the list is stopped (No longer supported in V9)

What was done



- Created a test program that selected data from SYSIBM.SYSTABLES and SYSIBM.SYSCOLUMNNS
- 2 SQL statements
- Bind and EXPLAIN on DB2 V5, V6, V7, V8, and V9
- Made copies of SYSTABLES and SYSCOLUMNNS for test cases 6 to 13
- The same programs were run on V5, V6, V7, V8, and V9
- We changed SQL, Statistics, Indexes and some DSNZPARMs

First SQL

- SELECT NAME, TBNAME, COLNO, COLTYPE,
LENGTH FROM SYSIBM.SYSCOLUMNS WHERE
TBNAME = :SYSCOLUMNS.SYSCOL-
TBNAME

Second SQL

- `DECLARE SELECT-1 CURSOR FOR SELECT A.NAME, A.TBNAME, A.COLNO, A.COLTYPE, A.LENGTH FROM SYSCOLUMNS A INNER JOIN SYSTABLES B ON A.TBNAME = B.NAME AND A.TBCREATOR = B.CREATOR WHERE A.TBNAME = :SYSCOL-SYSCOLUMNS.SYSCOL-TBNAME AND A.TBCREATOR = :SYSCOL-SYSCOLUMNS.SYSCOL-TBCREATOR`

Case 1 SQL Change Where Clause

- old SQL
- DECLARE SELECT-1 CURSOR FOR
- SELECT NAME, TBNAME, COLNO, COLTYPE, LENGTH
- FROM SYSIBM.SYSCOLUMNNS
- WHERE TBNAME = :SYSCOL-SYSCOLUMNNS.SYSCOL-TBNAME
- new SQL
- DECLARE SELECT-1 CURSOR FOR
- SELECT NAME, TBNAME, COLNO, COLTYPE, LENGTH
- FROM SYSIBM.SYSCOLUMNNS
- WHERE TBNAME = :SYSCOL-SYSCOLUMNNS.SYSCOL-TBNAME
- AND TBCREATOR = :SYSCOL-SYSCOLUMNNS.SYSCOL-TBCREATOR

Summary for TEST01



	V5	V6	V7	V8	V9
Before MSU's	42	23	43	9	60
After MSU's	14	1	1	2	1
Access change	MC=2 List Prefetch	MC=2 No PF	MC=2 No PF	MC=2 No PF	MC=2 No PF

Case 1 Results

- SQL changes the index from match columns = 0 to match columns=2 – noticeable improvement in estimated MSU's
- All versions turn off sequential Prefetch
- New option for Prefetch in V9 – D means optimizer expects dynamic Prefetch

Case 2 SQL Change Select to Join

- old SQL
- DECLARE SELECT-1 CURSOR FOR
- SELECT NAME, TBNAME, COLNO, COLTYPE, LENGTH
- FROM SYSIBM.SYSCOLUMNS
- WHERE TBNAME = :SYSCOL-SYSCOLUMNS.SYSCOL-TBNAME
- new SQL
- DECLARE SELECT-1 CURSOR FOR
- SELECT A.NAME, A.TBNAME, A.COLNO, A.COLTYPE, A.LENGTH
- FROM SYSIBM.SYSCOLUMNS A
- INNER JOIN
- SYSIBM.SYSTABLES B
- ON A.TBNAME = B.NAME
- AND A.TBCREATOR = B.CREATOR
- WHERE A.TBNAME = :SYSCOL-SYSCOLUMNS.SYSCOL-TBNAME

Summary for TEST02



	V5	V6	V7	V8	V9
Before MSU's	41	3	4	2	60
After MSU's	14	23	43	9	9
Access change	Index Scan	NLJ No PF	NLJ No PF	NLJ No PF	NLJ No PF

Case 2 Results

- All versions stop using sequential Prefetch
- Cost estimation are very different between releases
- Because the new statement is a join, the SQL doesn't match so it shows deleted and added SQL which means the new access path is shown first then the deleted

Case 3 SQL Change Change Join Criteria

- old SQL
- DECLARE SELECT-1 CURSOR FOR
- SELECT A.NAME, A.TBNAME, A.COLNO, A.COLTYPE, A.LENGTH
- FROM SYSIBM.SYSCOLUMNS A
- INNER JOIN
- SYSIBM.SYSTABLES B
- ON A.TBNAME = B.NAME
- AND A.TBCREATOR = B.CREATOR
- WHERE A.TBNAME = :SYSCOL-SYSCOLUMNS.SYSCOL-TBNAME
- AND A.TBCREATOR = :SYSCOL-SYSCOLUMNS.SYSCOL-TBCREATOR
- new SQL
- DECLARE SELECT-1 CURSOR FOR
- SELECT A.NAME, A.TBNAME, A.COLNO, A.COLTYPE, A.LENGTH
- FROM SYSIBM.SYSCOLUMNS A
- INNER JOIN
- SYSIBM.SYSTABLES B
- ON A.TBNAME = B.NAME
- WHERE A.TBNAME = :SYSCOL-SYSCOLUMNS.SYSCOL-TBNAME
- AND A.TBCREATOR = :SYSCOL-SYSCOLUMNS.SYSCOL-TBCREATOR



Summary for TEST03

	V5	V6	V7	V8	V9
Before MSU's	2	3	4	2	2
After MSU's	36	23	43	9	5
Access change	NLJ	TS Scan Cartesian Join	Same	Same	Same

Case 3 Results

- V6 V7 V8 and V9 are the same index and then nested loop
- V6 and V7 estimates increase substantially, while V8 estimate increases somewhat
- The index is tbcreator, tbname so the match columns goes from 2 to 0 and will give a Cartesian product of all the columns with all the tables of that name

Case 4 Change Data List

- V5, V6 will not use VARCHAR columns in the index.
- V7 needs to have a DSNZPARM set (SPRMVCFK) to allow data use of VARCHAR from an Index
- Be careful if using VARCHAR access in V7 to update as result is padded with blanks and most programmers do not rewrite as VARCHAR



Summary for TEST04

	V5	V6	V7	V8	V9
Before MSU's	2	1	1	2	2
After MSU's	2	1	2	2	2
Access change	No Change	No change	Data access	Data access	Data access

Case 4 Results

- V6 has no change because it can not use VARCHAR
- V7 needed VCFK set to get index only on VARCHAR fields in index - V8 doesn't care
- The access changes from index only to data+index on V7 and V8
- VARCHAR index access is the default on V8, but V8 does not pad if index is defined as NOPAD
- **If programs re-write with the padded length you will enlarge the data**

Case 5 SQL Change Join to Subselect

- old SQL
- DECLARE SELECT-1 CURSOR FOR
- SELECT A.NAME, A.TBNAME, A.COLNO, A.COLTYPE, A.LENGTH
- FROM SYSIBM.SYSCOLUMNS A
- INNER JOIN
- SYSIBM.SYSTABLES B
- ON A.TBNAME = B.NAME
- AND A.TBCREATOR = B.CREATOR
- WHERE A.TBNAME = :SYSCOL-SYSCOLUMNS.SYSCOL-TBNAME
- AND A.TBCREATOR = :SYSCOL-SYSCOLUMNS.SYSCOL-TBCREATOR
- new SQL
- DECLARE SELECT-1 CURSOR FOR
- SELECT A.NAME, A.TBNAME, A.COLNO, A.COLTYPE, A.LENGTH
- FROM SYSIBM.SYSCOLUMNS A
- WHERE EXISTS
- (SELECT B.NAME
- FROM SYSIBM.SYSTABLES B
- WHERE A.TBNAME = B.NAME
- AND A.TBCREATOR = B.CREATOR)
- AND A.TBNAME = :SYSCOL-SYSCOLUMNS.SYSCOL-TBNAME
- AND A.TBCREATOR = :SYSCOL-SYSCOLUMNS.SYSCOL-TBCREATOR



Summary for TEST05

	V5	V6	V7	V8	V9
Before MSU's	2	2	NC	NC	2
After MSU's	3.6	3	NC	NC	2
Access change	NLJ to IX Mc = 2	NLJ to IX only	No Change	No Change	Chg join sequence

Case 5 Results

- V6 index – nested loop to index only
- V7 and V8 have no change
- Change join to Subselect - this won't change the access path
- Since the old join did not retrieve data, it can be replaced with a Subselect and have the same access path and return the same data
- V9 changed join sequence to reference SYSCOLUMNS first

Notes on Cases 6 through 13

- Expected path for the join would be an index read of SYSTABLES and a data read from SYSCOLUMNS
- The Optimizer must estimate from the statistics which table has the least cardinality and would be best to read first
- V8 & V9 appear to be more susceptible to statistics than V7, V6 and V5

Notes on Cases 6 through 13

- Old DBRM goes against SYSCOLUMNNS and SYSTABLES in the Catalog
- New DBRM goes against a copy of SYSCOLUMMS and SYSTABLES
- This is meant to simulate a movement from one subsystem to another

Statistic Changes

- All the examples use the same SQL with statistic changes
- DECLARE SELECT-1 CURSOR FOR
- SELECT A.NAME, A.TBNAME, A.COLNO, A.COLTYPE, A.LENGTH
- FROM SYSCOLUMNS A
- INNER JOIN
- SYSTABLES B
- ON A.TBNAME = B.NAME
- AND A.TBCREATOR = B.CREATOR
- WHERE A.TBNAME = :SYSCOL-SYSCOLUMNS.SYSCOL-TBNAME
- AND A.TBCREATOR = :SYSCOL-SYSCOLUMNS.SYSCOL-TBCREATOR
- and
- DECLARE SELECT-2 CURSOR FOR
- SELECT NAME, CREATOR, TYPE, DBNAME, TSNAME
- FROM SYSTABLES
- WHERE CREATOR = :SYSTAB-SYSTABLES.SYSTAB-CREATOR

Case 6 No RUNSTATS

- All statistics show -1.
- Results
- All access SYSCOLUMNNS first
- V5 index list Prefetch –NLJ – to index list Prefetch merge scan
- V6 index list Prefetch – merge scan - to list Prefetch
- V7 & V8 index only – nested loop - to list Prefetch
- V9 choose TS scan with index only to SYSTABLES
- DB2 chose list Prefetch over the index scan chosen with stats accessed SYSCOLUMNNS first
- V9 also changed the singleton select to TS scan

Summary for TEST06

	V6	V7	V8	V9
Before MSU's	2	1	2	2
After MSU's	34,280	34,192	34,299	1560
Access change	List PF merge	List PF NLJ	List PF NLJ	Syscolumns first with TS scan

Case 7 Tables are emptied

- RUNSTATS done on empty tables.
- Stats show 0 not -1
- RESULTS
- V5, V6, V7 and V8 are the same
- DB2 chooses index access even though stats show table with 0 rows – Tablespace Scan was expected
- V9 choose to reference SYSCOLUMNS first then index only to SYSTABLES
- V9 changed the singleton select to TS scan
- Single table select did not use list Prefetch

Summary for TEST07 - Join



	V6	V7	V8	V9
Before MSU's	1	1	1	1
After MSU's	1	1	1	2
Access change	No change	No change	No change	SYSCOLUMNS first



Summary for TEST07 - Single table cursor

	V6	V7	V8	V9
Before MSU's	4	5	4	4
After MSU's	1	1	1	1
Access change	No list Prefetch	No list Prefetch	No list Prefetch	TS scan

Case 8

- Cluster ratio for SYSCOLUMNS changed to 50%
- UPDATE SYSIBM.SYSINDEXES
- SET CLUSTERRATIO = 50
- ,CLUSTERRATIOF = +0.050000000000000000E+01
- ,FIRSTKEYCARD = 300
- ,FULLKEYCARD = 300
- ,FIRSTKEYCARDF = +0.300000000000000000E+03
- ,FULLKEYCARDF = +0.300000000000000000E+03
- WHERE CREATOR = 'PUBLIC8'
- AND NAME = 'DSNDCX01'

Case 8 Results

- V5 Index List Prefetch NLJ to Index Mc = 2 NLJ with list Prefetch
- V6 change both Join and single table cursor
- V7 change only join
- V8 change only single table cursor
- The estimates for V8 are much less
- V9 did not change any access paths

Summary for TEST08 - Join



	V6	V7	V8	v9
Before MSU's	2	2	2	2
After MSU's	2	4	2	2
Access change	Change table seq	Change table seq	No change	No change

Case 9

- Change host variable to not match the table for the join where clause
- A.TBCREATOR = :WS-PARM-VALUEC
- This is the first column in the both indexes
- RESULTS
- V5 IX List Prefetch NLJ to IX Mc = 0 NLJ
- V6 accidental change to single table cursor
- V6 match columns = 0 index
- V7 match columns = 0 index
- V8 match columns = 2 index
- V8 and V9 can still use the index even with data type mismatch

Summary for TEST09



	V6	V7	V8	V9
Before MSU's	2	1	5	2
After MSU's	7	4	5	2
Access change	Index MC=0	Index MC=0	No change	No change

Case 10

- Create new index and expect switch between Index 1 and Index 2
- CREATE INDEX PUBLIC10.DSNDCXX1
- ON SYSCOLUMNS (TBCREATOR ,TBNAME)
- RESULTS
- V5 No Change
- V6 change single table to TS scan and change join sequence
- V7 change the join sequence and add Prefetch
- V8 only change single table cursor to TS scan
- V8 Estimates are lower than V7
- V9 actually used the new index

Summary for TEST10 - Join



	V6	V7	V8	V9
Before MSU's	2	2		
After MSU's	6	5		
Access change	Chg join seq	Chg join seq	No change	No change

Summary for TEST10 - Single table cursor



	V6	V7	V8	V9
Before MSU's	4		5	2
After MSU's	4		6	5
Access change	TS scan	No change	TS scan	Higher est cost but used new IX

Case 11

- Dropped a column from the index
- Removed TBNAME from SYSCOLUMNNS index - change match columns from 2 to 1
- CREATE INDEX PUBLIC11.DSNDCX01
- ON SYSCOLUMNNS (TBCREATOR)
- RESULTS
- V5 IX Mc=2 NLJ to IX Mc = 1 NLJ
- V6 change single table to TS scan and change join sequence
- V7 change the join sequence and add Prefetch
- V8 only change single table cursor to TS scan
- V8 Estimates are lower than V7
- V9 changed the join sequence and index only to SYSTABLES

Summary for TEST11 - Join



	V6	V7	V8	V9
Before MSU's	2	1	2	2
After MSU's	10	18	13	9
Access change	NLJ List PF	SYSCOLUMNS first	SYSCOLUMNS first	SYSCOLUMNS first

Summary for TEST11 - Single table cursor



	V6	V7	V8	V9
Before MSU's	4		4	9
After MSU's	4		2	9
Access change	TS scan	No change	No list Prefetch	No change

Case 12

- Drop index on SYSCOLUMNNS
- Expect Tablespace Scan
- RESULTS
- V5 IX Mc=2 NLJ to TS NLJ
- V6 changed both single table cursor and join
- V7 only changed join
- V8 changed both single table cursor and join
- V9 only changed the join
- Join access does TS scan on SYSCOLUMNNS first then join to SYSTABLES

Summary for TEST12 - JOIN



	V6	V7	V8	V9
Before MSU's	2	1	2	2
After MSU's	22	56	27	79
Access change	Change join seq	TS scan SYSCOL	TS scan SYSCOL	TS scan SYSCOL

Summary for TEST12 - Single table cursor



	V6	V7	V8	V9
Before MSU's	4		5	4
After MSU's	4		6	4
Access change	TS scan	No change	TS scan	No change

Case 13

- Change Column Sequence in the index
- CREATE INDEX PUBLIC13.DSNDCX01
- ON SYSCOLUMNS (TBCREATOR ,NAME ,TBNAME)
- Access to SYSCOLUMNS now match columns = 1
- RESULTS
- V5 IX Mc = 2 List Prefetch NLJ to IX Mc =1 NLJ with List Prefetch
- V6 change both single table cursor and join
- V7 only change join
- V8 change both single table cursor and join
- V6 and V9 changed from match columns 2 to 1 because column name was moved in front of tbname in the index - DB2 still does non-matching scan using the index data but not part of match columns.
- V7 and V8 changed the join sequence to scan SYSCOLUMNS first then join to SYSTABLES
- V9 started with join to SYSCOLUMNS first so did not change

Summary for TEST13 - JOIN



	V6	V7	V8	V9
Before MSU's	2	2	2	3
After MSU's	4	3	2	3
Access change	SYSCOL match=1	Change join seq	Change join seq	No change

Summary for TEST13 - Single table cursor



	V6	V7	V8	V9
Before MSU's	4		5	2
After MSU's	4		6	3
Access change	TS scan	No change	TS scan	Match columns =12

Conclusions

- DB2 optimizer is becoming more dependent on statistics
- What statistics to collect becomes more important
- The default was `RUNSTATS TABLESPACE dbname.tsname TABLE INDEX`
- V6 forward will most likely produce more Access Path Changes than V7 to V8 or V8 to V9
- Some Access Paths will change between V7 and V8, and V8 to V9
- The V8 Optimizer comes up with lower estimates than V7 or V6

Reorg the tables – V8

- REORG TABLESPACE TEST16.TABLESP1 REUSE LOG NO SORTDATA SORTKEYS SHRLEVEL NONE Because the initial tables were built from unload and not sorted
- INDEX HAS 6,361 ENTRIES IN 2 LEVELS WITH CLUSTER RATIO 81
- REORG and rerun RUNSTATS
- RESULTS
- No changes

REORG the tables – V8 and change RUNSTATS request

- REORG TABLESPACE TEST16.TABLESP1 REUSE LOG NO SORTDATA
SORTKEYS SHRLEVEL NONE
- RUNSTATS TABLESPACE TEST17.TABLESP1 TABLE (ALL) INDEX
(ALL KEYCARD FREQVAL NUMCOLS 2 COUNT 10) REPORT YES
- INDEX HAS 6,361 ENTRIES IN 2 LEVELS WITH CLUSTER RATIO 100
- REORG and rerun RUNSTATS for SYSCOLUMNNS
- RESULTS
- Join always used SYSTABLES first and then SYSCOLUMNNS

RUNSTATS case Results

- The index DSNDCX01 has 3 columns
- Default RUNSTATS TABLESPACE only collects first key cardinality and full key cardinality
- The JOIN specified only 2 of the columns in the index
- Because DB2 didn't have cardinality for the 2 column combination, it tried to estimate the value from the cardinality of each column
- This value was sufficiently wrong that the wrong table was chosen as the first table of the join
- INDEX (ALL KEYCARD FREQVAL NUMCOLS 2 COUNT 10) collects cardinality for the 2 columns in combination
- FREQVAL is all that is required for host variable access

New in V8

- Index matching with 'WRONG' data types without using functions to change data type
- Will use the index for matching access and JOIN's
- Does have a CPU cost at run time
- No indication for PLAN_TABLE or access path

New in V8

- CREATE TABLE ____ VOLATILE
- Specifies that index access should be used on this table whenever possible for SQL operations. However, be aware that list Prefetch and certain other optimization techniques are disabled when VOLATILE is used.
- Like OPTIMIZE for 1 ROWS without SQL changes

New in V8

- Unpadded indexes
- HOWEVER – existing indexes will stay padded until ALTER INDEX PADDED
- When an index with at least one varying-length column is changed from PADDED to NOT PADDED, or vice versa, the index is placed in restricted REBUILD-pending status (RBDP). The index cannot be accessed until it is rebuilt from the table

New in V8

- READ INDEX backward
- V7 could SELECT MAX from ascending index
- V8 can use index forwards or backwards but must match all columns ascending and descending

You are Now a V9 Expert!

Please Come visit us at our IDUG Booth 418 to
receive your Expert Pin!





Gerald Hodge

HLS Technologies

ghodge@hlstechnologies.com

WWW.HLSTECHNOLOGIES.COM

Has copies of JCL and DBRM's to recreate these test cases
and the longer version of this presentation

For a WebEx discussion with more details contact

ewest@hlstechnologies.com