

E56

# IMS V8 DBRC Enhancements

Karen Ranson (ranson@us.ibm.com)



St. Louis, MO

Sept. 30 - Oct. 3, 2002



- **Handle large RECON records**
  - Eliminate planned and unplanned outages that are due to RECON record size growth
- **Provide automatic RECON "loss" notification**
  - All DBRC instances should automatically deallocate the "discarded" RECON without delay after a RECON reconfiguration
- **Support RECON command authorization**
  - Provide support for authorization control for DBRC commands
- **Eliminate specific DBRC/IMS abends**

# *DBRC Enhancements - Highlights*

---



- **16M RECON Record Size**

- RECON records up to 16M are supported

- **PRILOG Compression Enhancement**

- Attempted whenever a data set entry is added to PRILOG
- Overhead reduced to improve performance

- **Automatic RECON Loss Notification**

- A RECON reconfiguration is automatically propagated to other DBRC instances

# *DBRC Enhancements - Highlights...*

---



- **RECON Command Authorization Support**
  - Support is provided that allows users to control RECON access/update via:
    - DBRC batch commands (DSPURX00)
    - HALDB Partition Definition Utility
  - User exit implementation allows user flexibility
    - customize security criteria
    - maintain audit trail
  
- **Eliminate DBRC/IMS Abends**
  - Several DBRC/IMS abends are eliminated

# *16M RECON Record Size*

---



- **DBRC does its own RECON record spanning**
  - RECON records are written as multiple VSAM records
  - A RECON record "segment" fits into a single control interval (CI)
    - segment size =  $\text{MIN}(\text{record size, CI size})$  - some overhead
    - VSAM spanning is not used
  - Limit is 16M because of move-character-long (MVCL) instruction restriction

# *16M RECON Record Size - User interfaces*

---

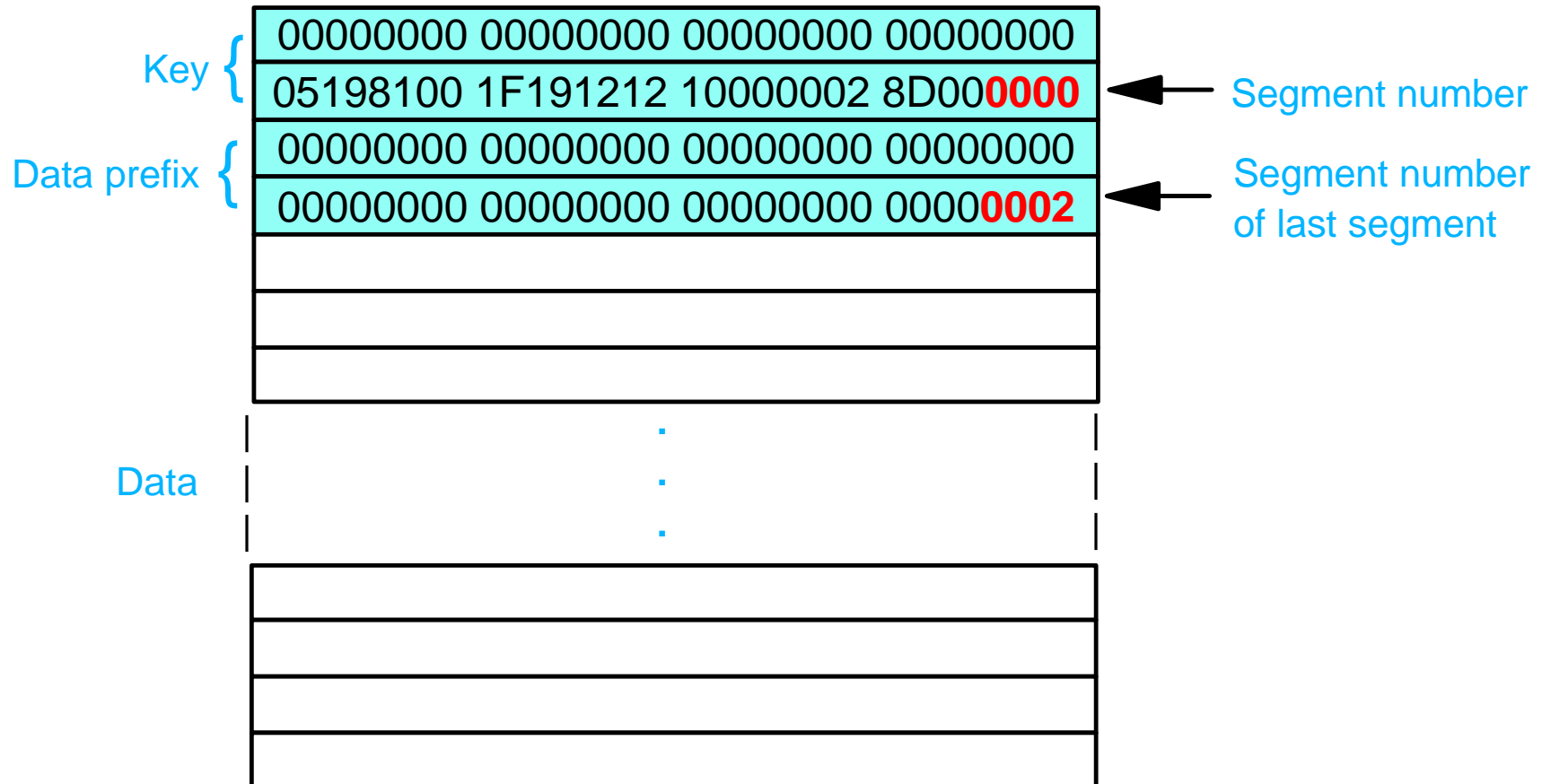


- **DBRC spanning of records is automatic**
- **Users may choose to adjust RECON data set attributes (record size, CI size, SPANNED/NONSPANNED)**
- **Use of SIZALERT may need to be adjusted**
- **Unsegmented RECON records are presented to the RECON I/O exit routine (DSPCEXT0)**

# 16M RECON Record Size - Segmenting example



## Logical RECON record



# 16M RECON Record Size - Segmenting example...



## Physical records (segments)

00000000 00000000 00000000 00000000
05198100 1F191212 10000002 8D00 <b>0000</b>
00000000 00000000 00000000 00000000
00000000 00000000 00000000 0000 <b>0002</b>

Segment 0

00000000 00000000 00000000 00000000
05198100 1F191212 10000002 8D00 <b>0001</b>

Segment 1

00000000 00000000 00000000 00000000
05198100 1F191212 10000002 8D00 <b>0002</b>

Segment 2

Data prefix only exists in first segment (segment 0)



# *PRILOG Compression Enhancement*

---



- **PRILOG compression is attempted whenever an OLDS archive job is run**
  - For RSR, Tracking log data set open
- **Oldest allocation information for each DBDS is kept in the LOGALL record**
  - Used to reduce the overhead of compression attempts
  - LIST output:
    - Earliest overall ALLOC on the log
    - DBDSs sorted in order of their earliest ALLOC
- **Indication given if nothing was compressed (DSP1150I)**
  - ALLOC
  - Checkpoint
  - Log retention period

# Automatic RECON Loss Notification (ARLN)

---



- **DBRC automatically deallocates the discarded RECON copy after a reconfiguration**
  - I/O error
  - CHANGE.RECON REPLACE
- **The Structured Call Interface (SCI) is required**
  - Used for intra-DBRC communication
  - There is a one-to-one correspondence between a RECON and an IMSplex
    - All DBRC instances using a given RECON join the same IMSplex
  - A DBRC address space can use SCI even if the IMS control region is not using SCI (i.e., no OM or RM)
  - If the control region and DBRC address spaces both use SCI, they must join the same IMSplex

# ARLN - User interfaces

---



- **Automatic RECON Loss Notification is optional**
- **The IMSplex name for the RECON is specified via**
  - the **DBRC SCI Registration exit** or
  - new EXEC statement keyword parameter, **IMSPLEX=**
- **The exit and/or the parm can be used**
  - if both, exit overrides the parm
- **ARLN is not available if:**
  - the exit indicates that SCI is not to be used
  - the exit returns an invalid IMSplex name
  - the SCI registration request fails
- **If SCI registration fails or could not be attempted because of an exit error, RECON access is denied**



## ● DBRC SCI Registration exit (DSPSCIX0)

- Function: Provide the IMSplex name to be used for a RECON
- Standard parameter list: R1-->parm pointer list
- Input:
  - DSN of one of the RECON data sets (RECON1, RECON2 or RECON3)
  - IMSPLEX parameter value
- Output:
  - IMSplex name (1 to 5 characters)
  - Return code

**RC00** - IMSplex name is used to register with SCI

**RC04** - No SCI registration - RECON access fails if the RECON contains an IMSplex name

**RC08** - No SCI registration - any IMSplex name found in RECON is ignored, RECON access is allowed

**RC12** - RECON access fails



## ● **DBRC SCI Registration exit (DSPSCIX0) ...**

- The sample exit supplied by IMS:
  - Lookup table matches RECON DSN with an IMSplex name
  - Returns the specified IMSPLEX parm (RC00)
  - RC04 if IMSPLEX parameter not specified
  
- If the SCI Registration exit is not found
  - IMS behaves as if the IBM supplied exit were used
  
- DSPSCIX0 must be found in an authorized data set, which can be a member of JOBLIB, STEPLIB, or LINKLIST.
  - If the library is concatenated, only the data set containing the exit needs to be authorized



- **First DBRC instance to join the IMSplex saves the IMSplex name in the RECON**
  - All subsequent V8 DBRC instances using the RECON must specify the same IMSplex name
- **CHANGE.RECON *IMSPLEX(imsplex\_name)*|*NOPLEX***
  - Used to change or reset the IMSplex name for the RECON
    - **Cannot be used to set the initial IMSplex name**
    - No other V8 DBRC instances can be active when the command is processed except those active before ARLN was activated (**DSP1124I**)
  - /RMCHANGE cannot specify these keywords
  - Any subsequent commands in the DBRC command utility (DSPURX00) job step will fail



- **DSP1123I DBRC REGISTERED WITH IMSPLEX ..... [USING  
EXIT]**
- **DSP1124I DBRC NOT PARTICIPATING IN IMSPLEX .....**
- **DSP1125I IMSPLEX NAME CONFLICT - DBRC IS USING .....  
BUT IMS IS USING .....**
- **DSP1135I SCI REGISTRATION FAILED, IMSPLEX NAME=.....,  
RC=....., RSN=.....**
- **DSP1136I RECON ACCESS DENIED, IMSPLEX NAME .....  
NOT VALID**
- **DSP1137I IMSPLEX MAY NOT BE CHANGED, DBRC ACTIVE  
FOR job1, job2, ...**

# *ARLN - messages...*

---



- **DSP1139I RECON ACCESS DENIED BY DSPSCIX0**
- **DSP1140I UNEXPECTED RETURN CODE FROM .....  
REQUEST, RC=....., RSN=.....**
- **DSP1141I RECON LOSS NOTIFICATION RECEIVED**
- **DSP1142I EXIT MODULE ..... FOUND IN UNAUTHORIZED  
LIBRARY: .....**
- **DSP1143I RECON ACCESS FORCED BY DSPSCIX0**
- **DSP1144I IMSPLEX PARAMETER NOT ALLOWED**
- **DSP1145I RECON LOSS NOTIFICATION NOT SENT**



# *ARLN - feature activation*

---



## **Problem 1**

- PLEX1 active, but not yet intended for use
- Job mistakenly submitted with IMSPLEX=PLEX1
  - PLEX1 recorded in RECON
- Existing jobs will issue DSP1124I
- New jobs will not run unless they also use PLEX1

## **Solution**

- DSPURX00 job for CHANGE.RECON NOPLEX with IMSPLEX=PLEX1 on its EXEC statement

# *ARLN - feature activation...*

---



## **Problem 2**

- PLEX1 active, but not yet intended for use
- DSPURX00 job mistakenly submitted with IMSPLEX=PLEX1, executing a CHANGE.RECON IMSPLEX(PLEX2) command
  - PLEX2 recorded in RECON
  - No PLEX2 for SCI registration
- Existing jobs will issue DSP1124I
- New jobs will not be able to run
  - Even CHANGE.RECON IMSPLEX(PLEX1) !

## **Solution**

- DSPSCIX0 which will yield a return code 8 (No SCI registration, ignore IMSplex name in the RECON)
- Use it with CHANGE.RECON job to fix or reset the PLEXname



## ● Recommendations

- Restrict access to **CHANGE.RECON**  
**IMSPLEX|NOPLEX**
- Use **DSPSCIX0** to control IMSPLEX parm usage
- Have a **DSPSCIX0** (RC=8) ready for use in emergencies

# *RECON Command Authorization support*

---



- **Allows the installation to control the use of DBRC commands**
- **Commands can be authorized at the "command verb" level, the "resource type" level or the "resource" level**
  - For example--
    - PersonX can issue CHANGE.DB for all databases
    - PersonY can issue CHANGE.DB only for databases AAA, BBB, and CCC
    - PersonZ can issue CHANGE.SUBSYS but not CHANGE.DB
    - PersonQ can issue all LIST commands, but cannot issue any other commands
- **Other RECON security issues are not addressed**
  - Any jobs using DBRC must have control-level access to the RECON

# *RECON Command Authorization support...*

---



- **The HALDB Partition Definition Utility is supported**
  - DBRC requests from the utility are converted to equivalent DBRC commands for the purpose of command authorization
    - Utility requests -- Query, Set, Change, Delete
    - DBRC commands -- LIST, INIT, CHANGE, DELETE
  
- **/RMxxxxxx commands are not supported**



## ● Resource Name Table (DSPRNTBL)

- Contains the list of all protected resources, i.e. DBRC commands
- This table cannot be modified
- Complete list in the *DBRC Guide and Reference*
- A profile, discrete or generic, must be defined (RDEF) covering each resource

# *RECON Command Authorization support...*

---



## ● Resource Name Table (DSPRNTBL)...

In general, protection is provided:

- for DBs - specified in DBD keyword, specific types (TYPEFP)
- for log types (OLDS, SLDS, etc.)
- for specific groups (GSG, DBDSGRP, etc.)
- for specific or groups of subsystems
- for commands that act on ALL records of a type
- for each keyword of the CHANGE.RECON command
  - INIT.RECON only protected at the verb.modifier level

## ● Resource name examples:

**CHANGE.RECON.CMDAUTH**

**INIT.DB.*dbname***

# RECON Cmd Auth - User interfaces

---



## ● User interfaces

### – CHANGE.RECON **CMDAUTH(SAF|EXIT|BOTH|NONE, safhlq)**

- Used to enable/disable command authorization for a RECON
- *safhlq* - specifies the high level qualifier of the resource profiles, required with SAF, EXIT, and BOTH
- **NONE** - turns command authorization off, *safhlq* cannot be specified
- Turning on command authorization uses the specified setting
- The user must be authorized with the current setting to disable command authorization
- Cannot specify CMDAUTH keyword with online command

### – INIT.RECON **CMDAUTH(SAF|EXIT|BOTH|NONE, safhlq)**

- **CMDAUTH(NONE)** is the default



# RECON Cmd Auth - User interfaces...

---



- **CMDAUTH(SAF, *safhlq*)**

- DBRC issues RACROUTE FUNC=AUTH to invoke RACF or an equivalent security product
- Checks if the user is authorized for READ access to resource profiles such as:
  - *safhlq*.CHANGE.PRILOG.OLDS
  - *safhlq*.DELETE.LOG.INACTIVE
  - *safhlq*.GENJCL.ARCHIVE.ssid
  - *safhlq*.INIT.DBDS.dbname
  - *safhlq*.NOTIFY.CA.grpname

where *safhlq* is the user-defined resource high level qualifier

- Uses the FACILITY resource class
- New error message (DSP1157I) if authorization denied

# RECON Cmd Auth - User interfaces...

---



- **CMDAUTH(EXIT, *safhlq*)**

- The DBRC Command Authorization Exit, DSPDCAX0, is called to perform command authorization
  - User exit - sample provided
- DSPDCAX0 must be found in an authorized data set, which can be a member of JOBLIB, STEPLIB, or LINKLIST
  - If the library is concatenated, only the data set containing the exit needs to be authorized
- New error message (DSP1154I) if DSPDCAX0 denies authorization

# RECON Cmd Auth - User interfaces...

---



- **CMDAUTH(BOTH, safhlq)**

- The security product is invoked first, then DSPDCAX0
- SAF return code and RACF return/reason codes passed to DSPDCAX0
  - from RACROUTE FUNC=AUTH call
- DSPDCAX0 return code overrides the security product
  - DBRC SAF error message (DSP1157I) suppressed



## ● **DSPDCABK - DSPDCAX0 parameter block**

- Resource name address
- Resource name length
- Address/length of high-level-qualifier
- Address/length of Command Verb
- Address/length of Command Qualifier
- Address/length of Command Modifier
- Userid
- Address of DSPDCAX0
- SAF return code
- RACF return code
- RACF reason code
- DSPDCAX0 return code
- Address of user area
- Size of user area (1024 bytes)

# *RECON Cmd Auth - 'gotchas'*

---



## Scenario 1

- **User permitted to use the LIST.LOG ALL command**
- **This does not automatically permit the user for:**
  - LIST.LOG ALLOLDS
  - LIST.LOG OLDS()
- **Must remember that these are all separate resources**
  - No logical relationship between resources!

## Recommendation

- **Define a LIST.LOG.\* resource**

# *RECON Cmd Auth - 'gotchas'...*

---



## Scenario 2

- **UserA permitted to the LIST.DB.\* resource**
  - UserA can use LIST.DB to display any DB or combination of DBs
- **LIST.DB.XYZ resource is then defined, UserB permitted**
  - ➔ **UserA is no longer allowed to issue LIST.DB DBD(XYZ)**
  - ➔ **LIST.DB ALL can still be used by UserA to display XYZ!**

## Recommendation

- **Create RACF groups with different security levels**
  - use these groups in the appropriate PErmits

# Eliminate DBRC/IMS Abends

---



## ● Deallocation request

- No abend during deallocation processing if:
  - the ALLOC record is not found
  - the ALLOC record already has a deallocation time
- Error messages are issued
  - DSP0153I (new) identifies the DBDS and ALLOC and DEALLOC times
  - DSP0300I indicates the specific error
- A dump is taken
- 'Prohibit further authorization' status is set for the DB/Area

## ● Authorization request

- No abend if the SUBSYS record becomes larger than the RECON physical record size
  - With **16M RECON Record Size**, the SUBSYS record is written as multiple RECON record segments

# *Eliminate DBRC/IMS Abends...*

---



- **No abend if recording an EEQE causes the DBDS record to exceed the RECON physical record size**
  - Abend eliminated by **16M RECON Record Size**
- **Maximum number of EEQEs per DB is 32,767**
  - **DB I/O Error request**
    - IMS abend U0602 still occurs if the maximum number of EEQEs is exceeded
    - DFS0612I message with a new return code of 32
  - **CHANGE.DBDS ADDEQE()**
    - New message DSP1146I



# LIST output - RECON record



## RECON

RECOVERY CONTROL DATA SET, IMS **V8R1**  
DMB#=7 INIT TOKEN=01225F2206572F  
NOFORCER LOG DSN CHECK=CHECK17 STARTNEW=NO  
TAPE UNIT=3400 DASD UNIT=3400 TRACEOFF SSID=IMSA  
LIST DLOG=NO CA/IC/LOG DATA SETS CATALOGED=NO  
**MINIMUM VERSION = 6.1**  
LOG RETENTION PERIOD=00.001 00:00:00.0  
**COMMAND AUTH=SAF HLQ=HLQ70**  
SIZALERT DSNUM=15 VOLNUM=16 PERCENT= 95  
LOGALERT DSNUM=3 VOLNUM=16

## TIME STAMP INFORMATION:

TIMEZIN = %SYS

OUTPUT FORMAT: DEFAULT = LOCORG NONE PUNC YY  
CURRENT = LOCORG NONE PUNC YY

**IMSPLEX = \*\* NONE \*\***

-DDNAME-	-STATUS-	-DATA SET NAME-
RECON1	COPY1	IMSTESTL.IMS.RECON1
RECON2	COPY2	IMSTESTL.IMS.RECON2
RECON3	SPARE	IMSTESTL.IMS.RECON3

# LIST output - PRILOG record

---



```

PRILOG                                RECORD SIZE=      344
START = 98.254 12:34:56.7             *  SSID=IMS11   VERSION=8.1
STOP  = 00.000 00:00:00.0             #DSN=1
GSGNAME=**NULL**
FIRST RECORD ID= 0000000000000001    PRILOG TOKEN= 0

DSN=RLDS.LOG1                          UNIT=3400
START = 98.254 12:34:56.7             FIRST DS LSN= 0000000000000001
STOP  = 00.000 00:00:00.0             LAST  DS LSN= 0000000000000000
FILE SEQ=0001      #VOLUMES=0002

VOLSER=VOL001 STOPTIME = 99.254 12:34:56.0
  CKPTCT=0      CHKPT ID = 00.000 00:00:00.0
  LOCK SEQUENCE#= 939000000000

VOLSER=VOL003 STOPTIME = 00.000 00:00:00.0
  CKPTCT=0      CHKPT ID = 00.000 00:00:00.0
  LOCK SEQUENCE#= 000000000000
```

# *LIST output - LOGALL record*

---



LOGALL

START = 98.254 12:34:56.7 \*

**EARLIEST ALLOC TIME = 98.254 12:34:56.8**

DBDS	ALLOC=2	-DBD-	-DDN-	-ALLOC-
		DBVHDJ05	CJVHDG1E	1
		DHVNTZ02	HIDAM	1



- **Version 8 Migration/Coexistence supports V6 and V7**
- **Time History Table is deleted**
  - was used for coexistence with V5 format timestamps
- **COEX|NOCOEX keywords removed from INIT.RECON and CHANGE.RECON commands**
  - accepted for compatibility



- Use the **CHANGE.RECON UPGRADE** command to migrate your RECONS to version 8
  - DSPURU00 is obsolete - only used for V5 migration
- Every RECON record grew by at least 32 bytes (**16M RECON Record Size**)
  - most grew even more (other enhancements, reserved space)
- **DBRC spanning of records is automatic**
  - Upgrade breaks records into segments as necessary

# *Coexistence - 16M RECON Record Size*

---



- **Unlimited RECON record size is supported in coexistence mode**
  - Down-level releases can read and write segmented RECON records once the RECON has been upgraded to V8
- **Use of SIZALERT may need to be adjusted**
- **Unsegmented RECON records are presented to the RECON I/O exit routine (DSPCEXT0)**
- **Vendor code may need to be modified to handle segmented records**



- **MINVERS(61|71|81)**

- New keyword for **INIT.RECON/CHANGE.RECON**
  - MINVERS(61) is the default for INIT.RECON and upgrade
  - Can only be changed using version 8
- Added to support APPC/OTMA SMQ Enablement
- Minimum IMS version allowed to sign on to DBRC

# *Coexistence - Miscellaneous*

---



- Down-level releases cannot participate in Automatic RECON Loss Notification
- RECON Command Authorization support is not available in down-level releases