E03

Exploring DBRC

Ron Blake

Silicon Valley Lab

email: rgblake@us.ibm.com

St. Louis, MO **Sept. 30 - Oct. 3, 2002**

RETURN TO INDEX

Agenda

- ★ DBRC - what and why
- ★ DBRC components
- ★ DBRC tasks
- ★ The RECON
- ★ DBRC commands
  - Database Registration
- ★ IMS calls to DBRC
- ★ DBRC diagnostics
- ★ DBRC's V8 enhancements

What is DBRC?

- DBRC is the IMS Data Base Recovery Control facility.
- It helps you control log and database recovery.
- It controls database access in a data sharing environment
- It allows you to generate JCL for various IMS Utilities
  - Data Base Recovery
  - Change Accumulation
  - Image Copy
  - Log Archive
  - Log Recovery

Why DBRC?

- Most IMS configurations require DBRC
- DBRC plays a key role in managing the log data needed to restart and recovery IMS online subsystems
- It controls the data sharing environment
- DBRC can simplify your recovery process

When to Use DBRC

- For IMS configurations that require DBRC include
  - Online: DB / DC, DCCTL, or DBCTL
  - Data sharing environments, including IMS Sysplex
  - Extended Recovery Facility (XRF)
  - Remote Site Recovery (RSR)
- For IMS batch jobs and utilities that access registered databases
- When you want DBRC to supervise, simplify your recovery process and protect your databases

DBRC Components

- The Recovery Control (RECON) datasets
- Command support through the Database Recovery Control utility (DSPURX00)
- Skeletal Job Control Language (JCL) for various IMS recovery utilities

DBRC Tasks

- Automatic tasks performed thru DBRC and IMS interaction
  - Log control for IMS
  - Recording recovery information in the RECON
  - Verifying that IMS database utilities have correct JCL input
  - Controlling the recovery of registered DBs
  - Controlling the IMS data sharing environment

- Command requested tasks
  - Recording recovery information in the RECON
  - JCL generation for IMS utilities
  - Listing information in the RECON

DBRC, the early days

- The IMS Backup and Recovery Facility became DBRC
  - Supported the main IMS Utilities
    - Data Base Recovery
    - Change Accumulation
    - Image Copy

      `DFSUICP0 still mentions the old name`

- DBRC first released in 1978
  - DBRC's unofficial birthday - 01 January 1978
    - Hardcoded in DSPUIN00 until IMS 6.1
    - Prior to IMS V6 DBRC would reject commands with times < 1978

The RECON Data Sets

- DBRC's primary resource
  - Keep status for the IMS System and DB activity
    - for Backup and Recovery
  - Control access and activity
    - IMS logs
    - IMS databases
    - Data sharing
- What they are physically
  - VSAM KSDS
    - 24 byte Key
    - 32 byte Key
- How you define them
  - AMS DEFINE CLUSTER
- Why you need 3

Creating the RECON

```
//*      DELETE AND DEFINE RECON DATA SET
//DEFRCN  EXEC PGM=IDCAMS
//DD1       DD VOL=SER=IAPP01,DISP=SHR,UNIT=3390
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD *
  DELETE (IMS61.RECON1 IMS61.RECON2 IMS61.RECON3 )
  SET MAXCC = 0
  DEFINE CLUSTER (NAME(IMS61.RECON1) FILE(DD1) -
         INDEXED KEYS(32 0) VOL(IAPP01) TRK(3 3) FSPC(20 20) -
         SHR(3 3) SPANNED NOREUSE UNORDERED )
  DEFINE CLUSTER (NAME(IMS61.RECON2) FILE(DD1) -
         INDEXED KEYS(32 0) VOL(IAPP01) TRK(3 3) FSPC(20 20) -
         SHR(3 3) SPANNED NOREUSE UNORDERED )
   DEFINE CLUSTER (NAME(IMS61.RECON2) FILE(DD1) -
         INDEXED KEYS(32 0) VOL(IAPP01) TRK(3 3) FSPC(20 20) -
         SHR(3 3) SPANNED NOREUSE UNORDERED )
/*
```

RECON Initialization

- After defining the RECON datasets initialize the RECONs

- Issue the DBRC INIT.RECON command
  - RECON must be empty
  - Builds the RECON Header and Header Extension records
    - Controls system options
    - Controls RECON management
      - **COPY 1**
      - **COPY 2**
      - **SPARE**

RECON Records

- RECON (Header, Header Extension, Time History Table)
- PRILOG, SECLOG, IPRI, ISEC
- PRIOLD, SECOLD, PRISLD, SECSLD, IPRIOL,ISECOL, IPRISL, ISECSL
- LOGALL
- GSG
- SSYS
- BACKOUT
- CAGRP
- CA
- DBDSGRP (DBGRP and RECOVGRP)
- DB
- DBDS
- ALLOC
- IMAGE
- REORG
- RECOV

DBRC Command Verbs

- BACKUP
  - Make a copy of the RECON Data Set
- CHANGE
  - Modify the contents an existing RECON record
- DELETE
  - Delete an existing RECON record
- GENJCL
  - Generate JCL for a utility or function
- INIT
  - Initialize (create) a RECON record
- LIST
  - List an existing RECON record
- NOTIFY
  - Create or change the contents an existing RECON record
- RESET
  - Change the status of the GSG record

DBRC Command Syntax

- Standard, command-language syntax
- Similar to TSO and Access Method Services
- Consists of
  - a verb
  - a modifier
  - one or more parameters
  - a period separates the verb and the modifier
    - BACKUP.RECON
      ```
      the default is RECON1
      ```
    - LIST.RECON STATUS
- DBRC Time Stamps
  - Standard Compressed format
    - yydddhhmmsst [offset]
    - LIST.LOG STARTIME(970231212121)
  - Standard Punctuated format
    - [yy]yy|ddd|hh|mm|ss|t
    - LIST.LOG STARTIME('97.023 12:12:12.1 PST')

The INIT Command

- Use the INIT command to create a RECON record.
  - ADS
  - CA
  - CAGRP
  - DB
  - DBDS
  - DBDSGRP
  - GSG
    - SG
  - IC
  - PART
  - RECON

INIT.RECON Command

- The first command to issue after defining the RECON
- The RECON must be empty
- Has many parameters, but defaults may suffice
- Example:
  - INIT.RECON SSID(IMS3)

RECON Record Contents

- The RECON record comprises 2 record types
  - The RECON Header record
  - The RECON Header Extension record
    - Internal processing information

- RECON Initialization Token
  - Used for DBRC Usage Indicator (DUI) processing
- Processing Options (SHARECTL, FORCER, etc.)
  - Time Stamp Format Information
- Status indicators include
  - Minimum Version
  - RECON DD Names
  - RECON Data Set Names
  - COPY1, COPY2, and SPARE

The LIST Command

- Use the LIST command to display existing RECON records.
  - BKOUT
  - CAGRP
  - DB
  - DBDS
  - DBDSGRP
  - GSG
  - HISTORY
  - LOG
    - for the PRILOG Family of records
    - for a category of records
  - SUBSYS
  - RECON

LIST.RECON Command

- LIST.RECON
  - Lists the contents of the entire RECON
  - Generally creates lots of output
  - takes time
  - DBRC holds the RECONs for the duration of the command

- LIST.RECON STATUS
  - the RECON record may contain all you need
  - generates less than a page of output
  - quick
  - the IMS MTO can issue via the /RM command
    - /RML DBRC='RECON STATUS'.

RECON Record Example

```
RECON
 RECOVERY CONTROL DATA SET, IMS/ESA V6R1        COEXISTENCE ENABLED
 DMB#=8                                INIT TOKEN=A4060F2231139F
 NOFORCER  LOG DSN CHECK=CHECK17     STARTNEW=NO
 TAPE UNIT=3480      DASD UNIT=SYSDA     TRACEOFF   SSID=**NULL**
 LIST DLOG=NO                   CA/IC/LOG DATA SETS CATALOGED=NO
 LOG RETENTION PERIOD=00.001 00:00:00.0

 TIME STAMP INFORMATION:

   TIMEZIN = %SYS

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

 -DDNAME-       -STATUS-        -DATA SET NAME-
  RECON1         COPY1           IMSTESTL.IMS.RECON1
  RECON2         COPY2           IMSTESTL.IMS.RECON2
  RECON3         UNAVAILABLE
THT
    -LOCAL START-    -OFFSET-
 0000.000 00:00:00.0 -08:00
```

RECON Record Example (V8)

```
RECON
 RECOVERY CONTROL DATA SET, IMS V8R1
 DMB#=11                              INIT TOKEN=02226F1953352F
 NOFORCER  LOG DSN CHECK=CHECK17     STARTNEW=NO
 TAPE UNIT=3400       DASD UNIT=SYSDA     TRACEOFF   SSID=**NULL**
 LIST DLOG=YES                 CA/IC/LOG DATA SETS CATALOGED=NO
 MINIMUM VERSION = 6.1
 LOG RETENTION PERIOD=00.000 00:15:00.0
 COMMAND AUTH=NONE  HLQ=**NULL**
 SIZALERT DSNUM=15       VOLNUM=16     PERCENT= 95
 LOGALERT DSNUM=3        VOLNUM=16

 TIME STAMP INFORMATION:

   TIMEZIN = %SYS              -LABEL- -OFFSET-
                                PDT     -07:00
                                PST     -08:00

   OUTPUT FORMAT:  DEFAULT = 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
```

The BACKUP Command

- Use the BACKUP command to create a backup copy of the RECON.
  - BACKUP.RECON
    - BOTH

      or
    - RECON1 or RECON2

- The command invokes IDCAMS REPRO
  - Note: backup to a sequential data set has a 32 KB restriction

The CHANGE Commands

- Use the CHANGE command to alter existing RECON records.
  - ADS
  - BKOUT
  - RECON
    - CATDS
    - TRACEON or TRACEOFF
    - LOGALERT, SIZALERT
    - LISTDL or NOLISTDL
  - DB
  - DBDS
  - DBDSGRP
  - CA
    - INVALID
  - CAGRP
  - IC or UIC
  - PRILOG (SECLOG)
    - ERROR
  - SUBSYS

INIT.DB Command

- Use the INIT.DB command to register each database whose recovery you want DBRC to control

- For a non High Availability Large Database (HALDB) use
  - INIT.DB

- For a High Availability Large Database (HALDB) use
  - INIT.DB and INIT.PART

    or

  - the HALDB Partition Definition utility

DB Record Contents

- DBD name
- Status flags and counters
- Data sharing level supported
- Data sharing control information
- Current sharing status (i.e. authorized subsystems)

DB Record Example

```
DB
 DBD=DBOHIDK5              IRLMID=*NULL       DMB#=3      TYPE=IMS
 SHARE LEVEL=3             GSGNAME=**NULL**    USID=0000000002
 AUTHORIZED USID=0000000002  RECEIVE USID=0000000002 HARD USID=0000000002
 RECEIVE NEEDED USID=0000000000
 DBRCVGRP=**NULL**
 FLAGS:                        COUNTERS:
  BACKOUT NEEDED       =ON         RECOVERY NEEDED COUNT   =0
  READ ONLY            =OFF        IMAGE COPY NEEDED COUNT =0
  PROHIBIT AUTHORIZATION=OFF       AUTHORIZED SUBSYSTEMS   =1
  RECOVERABLE          =YES        HELD AUTHORIZATION STATE=0
                            EEQE COUNT            =0
  TRACKING SUSPENDED   =NO         RECEIVE REQUIRED COUNT  =0
  OFR REQUIRED         =NO

 ASSOCIATED SUBSYSTEM INFORMATION:
                                        ENCODED  B/O NEEDED
   -SSID-       -ACCESS INTENT-     -STATE-      -COUNT-    -SS ROLE-
    SYS3                                 0            1          ACTIVE
```

INIT.DBDS Command

- Use the INIT.DBDS command to register a DBDS or DEDB area
  - DBRC examines the IMS DBDLIB data set
    - to verify the DBDS or DEDB area exists
    - to obtain the DBDS's data set identifier (DSID), its database organization (DBORG), and its dataset organization (DSORG)

- Use its parameters to specify
  - names of your skeletal JCL members
  - IC GENMAX value
  - IC REUSE/NOREUSE
  - Recovery Period (RECOVPD)

- The INIT.DBDS command fails if you issue it while the DB is in use

DBDS Record Contents

- DBD and DD names
- Data Set Name (DSN)
- Database type
- Status flags and counters
- Image Copy information
- Recovery information
- Change Accumulation information
- Skeletal JCL execution member names

DBDS Record Example

```
DBDS
 DSN=DBOHIDK5.CKXXI01O                                          TYPE=IMS
 DBD=DBOHIDK5  DDN=CKOHIG1O DSID=001 DBORG=HIDAM  DSORG=OSAM
 CAGRP=CAGRP2    GENMAX=2     IC AVAIL=0     IC USED=0     DSSN=00000001
 NOREUSE         RECOVPD=0       OTHER DDN=**NULL**
 DEFLTJCL=**NULL**  ICJCL=ICJCL     OICJCL=OICJCL    RECOVJCL=RECOVJCL
 RECVJCL=ICRCVJCL
 FLAGS:                             COUNTERS:
   IC NEEDED      =OFF
   IC RECOMMENDED =ON
   RECOV NEEDED   =OFF
   RECEIVE NEEDED =OFF               EEQE COUNT              =0
```

INIT.DBDSGRP Command

- Use an INIT.DBDSGRP command to define a group

- DB Group (DL/1databases or DEDB areas)
  - use the DBGRP parameter

- DBDS Group (DBDSs or DEDB areas)
  - use the MEMBERS parameter

- Recovery Group (DL/1databases or DEDB areas)
  - used with Online Recovery Services (ORS) recoveries
  - use the RECOVGRP parameter

DBDSGRP Record Example

```
DBDSGRP
GRPNAME=FJKGRP                  #MEMBERS=5       -DBD-     -DDN/AREA-
                                                 DIVNTZ02   DBHVSAM1
                                                 DIVNTZ02   DBHVSAM2
                                                 DHVNTZ02   HIDAM
                                                 DHVNTZ02   HIDAM2
                                                 DXVNTZ02   XDLBT04I
```

DBGRP Record Example

```
DBGRP
 GRPNAME=DBGRP1                  #MEMBERS=6      -DBD/AREA-
                                                  DIVNTZ02
                                                  DHVNTZ02
                                                  DXVNTZ02
                                                  DB21AR0
                                                  DB21AR1
                                                  DB21AR2
```

INIT.CAGRP Command

- INIT.CAGRP command to specify the DBDSs you want to belong to a specific Change Accumulation Group

- You must have done an INIT.DBDS for each of the DBDSs you identify on the INIT.CAGRP command

CAGRP Record Contents

- CA group name
- Status flags
- name of the CA skeletal JCL for GENJCL
- names of the DBDS members of the group

CAGRP Record Example

```
CAGRP
 GRPNAME=CAGRP1    GRPMAX=3     CA AVAIL=0       CA USED=0
 NOREUSE CAJCL=CAJCL             DEFLTJCL=**NULL**
                                 #MEMBERS=4    -DBD-        -DDN-
                                               DEDBJN21  DB21AR1
                                               DEDBJN21  DB21AR3
                                               DEDBJN21  DB21AR6
                                               DEDBJN21  DB21AR7
```

INIT.IC Command

- Use the INIT.IC command to create an IMAGE copy record in the RECON to define the image copy dateset available for use during a subsequent execution of a supported Image Copy Utility.
- Each INIT.IC command creates one IMAGE copy record
- You can specify a duplicate image copy dataset

IMAGE Record Example

```
IMAGE
 RUN     = 02.226 12:03:24.6          *  RECORD COUNT =33
 STOP    = 00.000 00:00:00.0             BATCH      USID=0000000004

IC1
 DSN=IMSVS.DBVHDJ05.CJVHDG1E.IC.IC120322        FILE SEQ=0001
 UNIT=SYSDA                    VOLS DEF=0001 VOLS USED=0001
                        VOLSER=222222



IMAGE
 RUN     = 02.226 12:04:27.3          *  RECORD COUNT =33
 STOP    = 02.226 12:04:27.6             CONCUR     USID=0000000005

IC1
 DSN=IMSVS.DBVHDJ05.CJVHDG1E.IC.IC120424        FILE SEQ=0001
 UNIT=SYSDA                    VOLS DEF=0001 VOLS USED=0001
                        VOLSER=222222
```

INIT.PART Command

- Use an INIT.PART command to register a HALDB Partition
  - DBRC creates the RECON HALDB partition structure
    - a PART record
    - the partition DB record
    - one or more DBDS records (determined by the DBD specification

      similar to a non-HALDB DBDS

      TYPE=PART
  - Specify
    - names of skeletal JCL members
    - IC REUSE or NOREUSE
    - IC GENMAX value

- Command fails if the HALDB is being used by the HALDB Partition Definition utility

HALDB Record Contents

- DBD name
- Status flags and counters
- Data sharing level supported
- Data sharing control information
- Current sharing status (i.e. authorized subsystems)

HALDB Record Example

```
DB
 DBD=DBHDOJ01                          DMB#=2       CHANGE#=5      TYPE=HALDB
 SHARE LEVEL=0                  GSGNAME=**NULL**
 PSNAME=**NULL**  DBORG=PHDAM     DSORG=OSAM     CURRENT PARTITION ID=00004
 FLAGS:                              COUNTERS:
   RECOVERABLE            =YES          PARTITIONS                =4
                                         DATA SET GROUP MEMBERS   =1
```

DB PART Record Contents

- Similar to a non-HALDB DB record, but with more partition related fields
- Partition High Key / String in character and hexidecimal forms
- Randomizer information
- Free Space information
- OSAM Block Size

DB PART Record Example

```
DB
 DBD=PDHDOJA   MASTER DB=DBHDOJ01                CHANGE#=2     TYPE=PART
 USID=0000000002  AUTHORIZED USID=0000000002  HARD USID=0000000002
 RECEIVE USID=0000000002  RECEIVE NEEDED USID=0000000000
 DBRCVGRP=**NULL**
 DSN PREFIX=IMSTESTS.DBHDOJ01                          PARTITION ID=00001
 PREVIOUS PARTITION=**NULL**    NEXT PARTITION=PDHDOJB
 OLRIMSID=**NULL**    ACTIVE DBDS=A-J      M-V EXIST=NO
 RANDOMIZER:
   NAME=DFSHDC20  ANCHOR=3    HIGH BLOCK#=3         BYTES=25
 FREE SPACE:
   FREE BLOCK FREQ FACTOR=0    FREE SPACE PERCENTAGE=0
 PARTITION HIGH KEY/STRING (CHAR):               (LENGTH=5   )
   J0200
 PARTITION HIGH KEY/STRING (HEX):
   D1F0F2F0F040404040404040404040404040404040404040404040404040

 OSAM BLOCK SIZE:
   A = 4096

 FLAGS:                              COUNTERS:
   BACKOUT NEEDED        =OFF          RECOVERY NEEDED COUNT   =0
   READ ONLY             =OFF          IMAGE COPY NEEDED COUNT =0
   PROHIBIT AUTHORIZATION=OFF          AUTHORIZED SUBSYSTEMS   =0
                                       HELD AUTHORIZATION STATE=0
                                       EEQE COUNT              =0
   TRACKING SUSPENDED    =NO           RECEIVE REQUIRED COUNT  =0
   OFR REQUIRED          =NO
   PARTITION INIT NEEDED =NO
   ONLINE REORG ACTIVE   =NO
   PARTITION DISABLED    =NO
```

INIT.SG Command

- INIT.SG command to define a service group as a member of a Global Service Group (GSG)
- Used for Remote Site Recovery (RSR) functions

INIT.GSG Command

- Use the INIT.GSG command to define (create) a Global Service Group (GSG) in a RECON
- Used for Remote Site Recovery (RSR) functions
- Command fails is RSRFEAT=NO is specified in IMSCTRL macro
- Must define the GSG to every RECON used by an IMS subsystem in the Global Service Group

GSG Record Contents

- GSG name
- Service group name and role
- time stamps
  - Start time of current log
  - Highest active site time
- Status flags and counters

GSG Record Example

```
GSG
 GSGNAME=IMSGSG1      #SGS=2      -SGNAME-      -ROLE-
                                  STLSITE1     ACTIVE       LOCAL
                                  STLSITE2     TRACKING
 CURRENT PRILOG TOKEN       = 6           TAKEOVER TOKEN = 0
 MINIMUM PRILOG TOKEN       = 1           DSN SEQ NUMBER = 0
 START TIME OF CURRENT LOG = 02.226 11:49:19.2
 HIGHEST ACTIVE SITE TIME  = 00.000 00:00:00.0
 TRACKING SUBSYSTEM ID     = **NULL**
 TAKEOVER IN PROGRESS
```

Skeletal JCL

- DBRC provides Skeletal JCL execution Partitioned Data Set (PDS) members and uses them as a model for performing keyword substitution and JCL generation.
- Skeletal JCL supplied
  - ARCHJCL
  - CAJCL
  - LOGCLJCL
  - ICJCL
  - OICJCL
  - ICRCVJCL
  - RECOVJCL
  - JOBJCL
- DSPUPJCL also provided
  - User JCL for rebuilding the Index and/or ILE data sets for a Partition of a HALDB.
- User Options:
  - modify supplied skeletal member slightly
  - create own skeletal member

GENJCL Command

- Use the GENJCL command to generate the JCL and utility control statements to execute these IMS utilities
  - GENJCL.ARCHIVE
    - Log Archive
  - GENJCL.CA
    - CA utility
  - GENJCL.CLOSE
    - Log Recovery utility
  - GENJCL.IC or OIC
    - Image Copy or Online Image Copy utility
  - GENJCL.RECEIVE
    - Database Recovery utility
  - GENJCL.RECOV
    - Database Recovery utility
- Use the GENJCL.USER command to generate JCL or any kind of user determined output

IMS calls to DBRC

- IMS and DBRC need to communicate in a number of environments
  - Batch
  - Online
  - Utilities
- Communication is done via macros
  - DFSBRLSC, the IMS-DBRC Interface macro
  - DFSBRLSB, the common parameter block macro
- Typical Functions
  - Initialization
  - Termination
  - Signon
  - Signoff
  - Authorization
  - DB Open
  - Log Processing (Open, Close,EOV)
  - Archive
  - Backout

DFSBRLSC Functions and DBRC Processing

DBRC Processing depends on the DFSBRLSC FUNCTION

- DFSBRLSC ID=DBA
  - FUNC= SIGNON or SIGNOFF
  - Subsystem signon/off processing (DSPSSIGN)
- DFSBRLSC ID=DBA
  - FUNC=ALLOC or DEALLOC
  - DBDS Allocation/Deallocation processing (DSPALD00)
- DFSBRLSC ID=DLG
  - DASD Log processing (DSPOLD00)
  - FUNC=
    - OPEN
    - Switch
    - Close
- DFSBRLSC ID=ARC
  - FUNC=INIT, EOJ, ARCOMPL
  - Archive processing (DSPARC00)

DFSBRLSC Functions and DBRC Processing ...

- DFSBRLSC ID=ICU
  - Image copy utility processing
  - FUNC= INIT, EOD, EOJ
- DFSBRLSC ID=CAU
  - Change accumulation utility processing
  - FUNC=INIT, EOJ
- DFSBRLSC ID=RVU
  - Database recovery utility processing
  - FUNC= INIT, EOJ
- DFSBRLSC ID=DBA
  - Allocation processing
  - FUNC=ALLOC

DFSBRLSC Functions and DBRC Processing ...

- DFSBRLSC ID=
  - DB Unload utility processing
  - FUNC=
- DFSBRLSC ID=
  - DB Reload utility processing
  - FUNC=
- DFSBRLSC ID=RVU
  - Batch Backout utility processing
  - FUNC= INIT, EOJ
- DFSBRLSC ID=
  - Log Recovery utility processing
  - FUNC=

DBRC Exit Processing (Initial)

The IMS-DBRC Interface Macro, DFSBRLSC, provides a number of functions that drive specific DBRC processing. In general, DBRC Exit Processing follow this flow on the "Initial" exit:

- ALLOCATE the RECONs
- RESERVE the RECONs
  - Done in DD Name sequence
  - QNAME is DSPURI01; RNAME is RECON DSN
- OPEN the RECONs
- VERIFY if needed
- READ RECON Header and Header Extension for each RECON
  - Determine Copy 1, 2, and Spare
  - Reconfigure if needed and possible
  - DEALLOCATE unused RECON datasets
- Cleanup the RECON if needed
- Perform requested DFSBRLSC Function
- Release the active RECONs

DBRC Exit Processing (Subsequent)

In general, DBRC Exit Processing follow this flow on subsequent DFSBRLSC exits.

- RESERVE the RECONs
  - Done in DD Name sequence
  - QNAME is DSPURI01; RNAME is RECON DSN
- Invalidate buffers
- READ RECON Header and Header Extension for each RECON
  - Determine Copy 1, 2, and Spare
  - Reconfigure if needed and possible
  - DEALLOCATE unused RECON datasets
- VERIFY if needed
- Cleanup the RECON if needed
- Perform requested DFSBRLSC Function
- If Function=TERM then CLOSE RECONs
- Release the active RECONs

Image Copy Processing

- Generate the Image Copy JCL
  - Do it yourself
  - Let DBRC do it with GENJCL.IC command
- Execute generated IC JCL
- The IMS IC utility
  - Calls DBRC to
    - signon
    - validate the IC JCL
  - Does the image copy
  - Calls DBRC upon completion to
    - create the IC record and put it in the RECON
    - signoff

IC Record Contents

- Image Copy dataset name
- type
  - BATCH
  - ONLINE
  - CONCUR
  - SMSCIC
  - SMSNOCIC
- time stamps
  - Run
  - Stop
- Record count

IMAGE Record Example

```
IMAGE
 RUN      = 02.226 12:03:24.6              *  RECORD COUNT =33
 STOP     = 00.000 00:00:00.0                 BATCH      USID=0000000004

IC1
 DSN=IMSVS.DBVHDJ05.CJVHDG1E.IC.IC120322          FILE SEQ=0001
 UNIT=SYSDA                          VOLS DEF=0001 VOLS USED=0001
                                     VOLSER=222222



IMAGE
 RUN      = 02.226 12:04:27.3              *  RECORD COUNT =33
 STOP     = 02.226 12:04:27.6                 CONCUR     USID=0000000005

IC1
 DSN=IMSVS.DBVHDJ05.CJVHDG1E.IC.IC120424          FILE SEQ=0001
 UNIT=SYSDA                          VOLS DEF=0001 VOLS USED=0001
                                     VOLSER=222222
```

Change Accumulation Processing

- Generate the Change Accumulation JCL
  - Do it yourself
  - Let DBRC do it with GENJCL.CA command
- Execute generated CA JCL
- The IMS CA utility
  - Calls DBRC to validate the CA JCL
  - Processes the change accumulation
  - Calls DBRC upon completion to create the CA record and put it in the RECON

CA Record Contents

- CA Dataset name
- CA Group name
- Time stamps
  - Stop
  - Run
  - Purge
- Status flags
- Details about DBDs and DDNs processed

CA Record Example

```
CA
 DSN=IMSVS.CAGRP1.CA.CA170401                        FILE SEQ=1
 CAGRP=CAGRP1        STOP    = 02.226 09:36:08.4                *
                     UNIT=SYSDA      VOLS DEF=1    VOLS USED=1
                                     VOLSER=222222
 RUN      = 02.226 09:04:28.0
  DBD=DEDBJN21 DDN=DB21AR1  PURGETIME = 02.226 09:35:52.3
    CHANGES ACCUMULATED=YES COMPLETE CA=YES INDOUBT EEQES=NO
    LSN     = 000000000000      DSSN   = 0000000001
    LRID    = 0000000000000412  USID   = 0000000002
  DBD=DEDBJN21 DDN=DB21AR3  PURGETIME = 02.226 09:35:53.2
    CHANGES ACCUMULATED=YES COMPLETE CA=YES INDOUBT EEQES=NO
    LSN     = 000000000000      DSSN   = 0000000001
    LRID    = 000000000000042D  USID   = 0000000002
  DBD=DEDBJN21 DDN=DB21AR6  PURGETIME = 02.226 09:35:54.4
    CHANGES ACCUMULATED=YES COMPLETE CA=YES INDOUBT EEQES=NO
    LSN     = 000000000000      DSSN   = 0000000001
    LRID    = 0000000000000447  USID   = 0000000002
  DBD=DEDBJN21 DDN=DB21AR7  PURGETIME = 02.226 09:35:55.0
    CHANGES ACCUMULATED=YES COMPLETE CA=YES INDOUBT EEQES=NO
    LSN     = 000000000000      DSSN   = 0000000001
    LRID    = 000000000000045C  USID   = 0000000002
```

IMS Batch Log Processing

- At Log Open, IMS calls DBRC with log information to record in the RECON
  - DBRC uses its Notify processing (DSPURT00)
    - Create PRILOG record
    - Create the LOGALL base record
    - Update the subsystem record
- At Log End of Volume (EOV), IMS calls DBRC with log information to write in the RECON
  - DBRC updates the PRILOG with volume information
- At Log Close, IMS calls DBRC with log information to record in the RECON
  - DBRC updates the PRILOG
- For dual logging, IMS makes similar calls to DBRC to write information about the secondary log in the RECON - the SECLOG record.

IMS DASD Log Processing

- IMS DASD Log open
  - Create PRIOLD record
  - Create SECOLD record
  - Create PRILOG record
  - Create SECLOG record
    - DSPURT00
  - Create the LOGALL base record
  - Update the SSYS record
  - NOTIFY
    - DSPURT00
- IMS Online Log Switch
- IMS Online Log Close

IMS DASD Log Processing ...

- IMS Online Log Switch
  - Update PRIOLD record
  - Update SECOLD record

- IMS DASD Log close
  - Update PRIOLD record
  - Update SECOLD record
  - Update PRILOG record
  - Update SECLOG record

PRILOG Record Contents

- Start time
- Stop time
- Record size
- Subsystem name (i.e. SSID)
- Log Version
- Global Service Group Name (GSGNAME)
- PRILOG Token
- Number of data sets (i.e. #DSN)
- First Record ID
- PRILOG family of records

Each data set entry contains the following

- Dataset name
- DSN start time stamp
- DSN stop time stamp
- First Dataset Log Sequence Number
- Last Dataset Log Sequence Number
- Unit, File, Volume, and Checkpoint information

Archive Processing (OLDS)

- GENJCL.ARCHIVE

- Archive Utility calls DBRC to
  - Add SLDS entry to PRISLD/SECSLD records
  - Add RLDS entry to PRISLD/SECSLD records
    - Add SLDS entry to PRISLD/SECSLD records when archive did not create a separate RLDS
  - Set status of archived PRIOLD/SECOLD to ARC COMPLT
  - If all OLDS for a subsystem execution are archieved, set subsystem stop time in PRILOG/SECLOG/PRISLD/SECSLD records.

Archive Processing (SLDS or RLDS)

- Validate that SLDS or RLDS being archived exists in RECON
  - Match
    - DSN
    - file sequence number
    - volume serial numbers
  - Look for duplicates
  - Can take awhile

- At archive completion update the data set entry (DSE) with new
  - DSN
  - file sequence number
  - unit type
  - volume serial numbers

Signon Processing

- SIGNON
  - Normal
  - Abnormal
    - Signon Recovery Start
    - Signon Recovery End
- SIGNOFF
  - Normal
  - Abnormal
- DSPSSIGN

Normal Signon

- Build SSYS record for the subsystem
  - Fails if SSYS record already exists in the RECON

- Driven by
  - DLI/DBB batch initialization
    - Subsystem name is the job name
  - /NRE online processing
    - Subsystem name is the job name

Abnormal Signon

- Signon Recovery Start
  - Fails if SSYS record already exists in the RECON
  - Sets bit in SSYS record for the subsystem
- Signon Recovery End
  - Releases data base authorizations
  - When complete, environment looks like Signon Normal
- Driven by
  - Batch Backout utility
    - Subsystem name
  - /ERE online processing
    - Subsystem name is the IMSID

SSYS Record Contents

- Subsystem name
- IRLM name
- Subsystem type
  - Batch
  - Online
  - Tracker
- Log open time stamp
- Status flags
- Authorized database names and status
  - the size of this part of the SSYS record varies
  - the number of databases or areas can grow large

SSYS Record Example (V8)

```
SSYS
 SSID=SYS3       LOG START=02.226 11:48:07.3
 SSTYPE=ONLINE   ABNORMAL TERM=OFF   RECOVERY STARTED=NO    BACKUP=NO
 TRACKED=NO      TRACKER TERM=OFF    SHARING COVERED DBS=NO
 IRLMID=**NULL**   IRLM STATUS=NORMAL          GSGNAME=IMSGSG1

 AUTHORIZED DATA BASES/AREAS=1       VERSION=8.1    XRF CAPABLE=NO
                                                     ENCODED
   -DBD-          -AREA-    -LEVEL-   -ACCESS INTENT-   -STATE-
   DEDBDD01       DD01AR0      1          UPDATE             6
```

PRISLD Record Contents

- Start time
- Stop time
- Record size
- Subsystem name (i.e. SSID)
- Log Version
- Global Service Group Name (GSGNAME)
- PRILOG Token
- Number of data sets (i.e. #DSN)
- First Record ID
- PRILOG family of records: PRILOG,SECLOG,PRISLD,SECSLD ...

Each data set entry contains the following

- Dataset name
- DSN start time stamp
- DSN stop time stamp
- First Dataset Log Sequence Number
- Last Dataset Log Sequence Number
- Unit, File, Volume, and Checkpoint information

PRILOG Record Example

```
PRILOG
 START = 04.060 22:33:57.6              *    SSID=SYS3
 STOP  = 04.060 22:35:18.8                   #DSN=1
 GSGNAME=**NULL**
 FIRST RECORD ID= 0000000000000000       PRILOG TOKEN= 0
 EARLIEST CHECKPOINT = 00.000 00:00:00.0

 DSN=IMSVS.RLDSP.SYS3.D04060.T2233576.V00                UNIT=SYSDA
 START = 04.060 22:33:57.6               FIRST DS LSN= 0000000000000000
 STOP  = 04.060 22:35:18.8               LAST  DS LSN= 0000000000000000
 FILE SEQ=0001    #VOLUMES=0001

   VOLSER=000000 STOPTIME = 04.060 22:35:18.8
     CKPTCT=1    CHKPT ID = 04.060 22:34:04.2
```

LOGALL Record Contents

- Log Start time stamp
  - this time identifies the PRILOG
  - it is identical to PRILOG start time
- Earliest allocation time stamp
- Number of Database datasets allocated
- DBD, DDN, and Alloc counts for each DBDS
  - This area of the LOGALL record varies in size depending on the number of DBDS allocated.
  - Allocation count relates to the number of ALLOC records that exist in the RECON for a given DBDS for the log.

LOGALL Record Example

```
LOGALL
 START    = 04.060 22:33:57.6                    *
 DBDS ALLOC=4                          -DBD-     -DDN-     -ALLOC-
                                       DIVNTZ02 DBHVSAM1   1
                                       DBVHDJ05 CJVHDG1E   1
                                       DBOHIDK5 CKOHIG1O   1
                                       DXVHIDK5 CKVHIIXK   1
```

PRISLD Record Example

```
PRISLD
 START = 04.060 22:33:57.6              *    SSID=SYS3
 STOP  = 04.060 22:35:18.8                   #DSN=1
 GSGNAME=**NULL**
 FIRST RECORD ID= 0000000000000000       PRILOG TOKEN= 0

 DSN=IMSVS.SLDSP.SYS3.D04060.T2233576.V00                UNIT=SYSDA
 START = 04.060 22:33:57.6               FIRST DS LSN= 0000000000000000
 STOP  = 04.060 22:35:18.8               LAST  DS LSN= 0000000000000000
 FILE SEQ=0001    #VOLUMES=0001

   VOLSER=000000 STOPTIME = 04.060 22:35:18.8
     CKPTCT=1    CHKPT ID = 04.060 22:34:04.2
```

PRIOLD Record Contents

- OLDS DD name
- Number of OLDS DD entries
- OLDS Data Set Name (DSN)
- PRILOG time
- Subsystem name
- Earliest Checkpoint time stamp
- Status
  - Archive status (e.g. ARC COMPLT, ARC STARTED,)
  - ARCHIVE JOB NAME
  - Available or Not
  - FEOV
- START and STOP time stamps
- FIRST and LAST DS LSN

PRIOLD Record Example

```
PRIOLD
 SSID=SYS3             # DD ENTRIES=2
 EARLIEST CHECKPOINT = 04.060 22:39:17.2

 DDNAME=DFSOLP00   DSN=IMSTESTL.IMS01.OLDSP0
 START = 04.060 22:37:54.9               FIRST DS LSN= 0000000000000000
 STOP  = 04.060 22:39:13.6               LAST  DS LSN= 0000000000000000
 STATUS=ARC COMPLT                         FEOV=NO    AVAIL
 PRILOG TIME=04.060 22:37:54.9           ARCHIVE JOB NAME=JT223914

 DDNAME=DFSOLP01   DSN=IMSTESTL.IMS01.OLDSP1
 START = 04.060 22:39:13.7               FIRST DS LSN= 0000000000000000
 STOP  = 04.060 22:39:42.0               LAST  DS LSN= 0000000000000000
 STATUS=ARC COMPLT                         FEOV=NO    AVAIL
 PRILOG TIME=04.060 22:39:13.7           ARCHIVE JOB NAME=JT223942
```

DBDS Update Processing

- Performed when first update to a DBDS occurs following allocation

- Call DBRC to
  - Check that DB is registered to DBRC
    - Match DBD, DDN, and DSN to RECON record
  - Create and put ALLOC record in RECON
  - Update LOGALL record
  - Return DSSN

ALLOC Record Contents

- ALLOC time stamp
- ALLOC LRID
- DEALLOC time stamp
- DEALLOC LRID
- Update set id (i.e. USID)
- Dataset sequence number (i.e. DSSN)
- PRILOG start time stamp

ALLOC Record Example

```
ALLOC
 ALLOC   =04.060 22:34:21.2          *  ALLOC LRID =0000000000000000
 DSSN=0000000001 USID=0000000000 START = 04.060 22:33:57.6

ALLOC
 ALLOC   =04.060 22:36:35.4          *  ALLOC LRID =0000000000000000
 DSSN=0000000002 USID=0000000000 START = 04.060 22:36:34.9

ALLOC
 ALLOC   =04.060 22:38:16.8          *  ALLOC LRID =0000000000000000
 DSSN=0000000003 USID=0000000000 START = 04.060 22:37:54.9
```

DBDS Record Example (V6)

```
DBDS
DSN=DBOHIDK5.CKXXI01O                                TYPE=IMS
DBD=DBOHIDK5  DDN=CKOHIG1O DSID=001 DBORG=HIDAM  DSORG=OSAM
CAGRP=**NULL**  GENMAX=2     IC AVAIL=0     IC USED=0     DSSN=00000003
NOREUSE         RECOVPD=0
DEFLTJCL=**NULL**  ICJCL=ICJCL     OICJCL=OICJCL
 ECOVJCL=RECOVJCL
RECVJCL=ICRCVJCL
FLAGS:                       COUNTERS:
 IC NEEDED      =OFF
 RECOV NEEDED   =OFF
 RECEIVE NEEDED =OFF              EEQE COUNT           =0
```

DB Record Example (V6)

```
DB
 DBD=DBOHIDK5              IRLMID=*NULL       DMB#=2       TYPE=IMS
 SHARE LEVEL=3             GSGNAME=**NULL**   USID=0000000000
 AUTHORIZED USID=0000000000  RECEIVE USID=0000000000 HARD USID=0000000000
 RECEIVE NEEDED USID=0000000000
 FLAGS:                        COUNTERS:
  BACKOUT NEEDED       =ON         RECOVERY NEEDED COUNT   =0
  READ ONLY            =OFF        IMAGE COPY NEEDED COUNT =0
  PROHIBIT AUTHORIZATION=OFF       AUTHORIZED SUBSYSTEMS   =1
  RECOVERABLE          =YES        HELD AUTHORIZATION STATE=0
                            EEQE COUNT            =0
  TRACKING SUSPENDED   =NO         RECEIVE REQUIRED COUNT  =0
  OFR REQUIRED         =NO

 ASSOCIATED SUBSYSTEM INFORMATION:
                                      ENCODED  B/O NEEDED
   -SSID-   -ACCESS INTENT-      -STATE-      -COUNT-        -SS ROLE-
    SYS3                             0            1            ACTIVE
```

Database I/O Error Processing

- Performed for read and write errors

- Call DBRC to:
  - Update DB and DBDS records with Extended Error Queue Element (EEQE) information

  - If write error
    - set "Recovery Needed" flag in DBDS record
    - set "Recovery Needed" counter in DB record

Database Reorganization

- Create and execute the DB Reorganizaton JCL

- The IMS Reorganization utility
  - Calls DBRC upon completion to create the REORG record and put it in the RECON

REORG Record Contents

- The REORG record is small and contains:
  - Time stamp when a reorganization occurred for the DBDS
    - RUN=
  - The Update Set IDentifier
    - USID=
- Actions that cause DBRC to create and write the record to the RECON:
  - NOTIFY.REORG
  - An IMS Utility

Database Recovery Processing

- Generate the Database Recovery JCL
  - Do it yourself
  - Let DBRC do it with GENJCL.RECOV command
- Execute generated RECOV JCL
- The IMS Database Recovery utility
  - Calls DBRC to validate the JCL
  - Processes the database recovery
  - Calls DBRC upon completion to create the RECOV record and put it in the RECON

RECOV Record Contents

- The RECOV record is small and contains:
  - The Database Recovery utility execution time for the DBDS
    - RUN=timestamp
  - The time to which the DBDS was recovered
    - RECOV TO= timestamp
  - The Update Set IDentifier, USID
    - RUN USID=
    - RECOV TO USID=
- Actions that cause DBRC to create and write the record to the RECON:
  - NOTIFY.RECOV
  - An IMS Utility

Authorization Processing

- Performed at initialization time for batch and first schedule for online
- Authorization for unregistered data bases granted unless FORCER specified
- Database registered if DBD, DDN, and DSN match RECON
- General processing flow:
  - Check status in DB and DBDS records
  - Check current authorizations for compatibility with this one
  - If DBs requested can be granted, update SSYS and DB records

SSYS Record Example (V7)

```
SSYS
 SSID=SYS3       LOG START=02.226 09:02:39.7
 SSTYPE=ONLINE   ABNORMAL TERM=OFF   RECOVERY STARTED=NO    BACKUP=NO
 TRACKED=NO      TRACKER TERM=OFF    SHARING COVERED DBS=NO
 IRLMID=**NULL**   IRLM STATUS=NORMAL          GSGNAME=IMSGSG1

 AUTHORIZED DATA BASES/AREAS=1       VERSION=7.1
                                                        ENCODED
   -DBD-          -AREA-    -LEVEL-   -ACCESS INTENT-   -STATE-
   DEDBDD01       DD01AR0      1          UPDATE            6
```

DBDS Record Example (V7)

```
DBDS
 DBD=DEDBDD01  AREA=DD01AR0                                        TYPE=FP
 SHARE LEVEL=1             DSID=001 DBORG=DEDB   DSORG=VSAM
 GSGNAME=IMSGSG1              USID=0000000003
 AUTHORIZED USID=0000000003  RECEIVE USID=0000000003 HARD USID=0000000003
 RECEIVE NEEDED USID=0000000000
 CAGRP=**NULL**  GENMAX=2     IC AVAIL=0     IC USED=1     DSSN=00000002
 NOREUSE         RECOVPD=0      NOVSO  PREOPEN    NOPRELOAD
 CFSTR1=**NULL**          CFSTR2=**NULL**          NOLKASID
 DEFLTJCL=**NULL**  ICJCL=ICJCL    RECVJCL=ICRCVJCL  RECOVJCL=RECOVJCL
 DBRCVGRP=**NULL**
 FLAGS:                              COUNTERS:
   PROHIBIT AUTHORIZATION=OFF          AUTHORIZED SUBSYSTEMS   =1
                                       HELD AUTHORIZATION STATE=6
   IC NEEDED              =OFF         ADS AVAIL #             =1
   RECOV NEEDED           =OFF         REGISTERED ADS #        =1
   DATABASE LEVEL TRACK   =YES         EEQE COUNT              =0
   RECEIVE NEEDED         =OFF
   OFR REQUIRED           =NO
   TRACKING SUSPENDED     =NO
   HSSP CIC IN PROGRESS   =NO

ADS LIST:
                                                                  CREATE
  -ADS DDN--ADS DSN-                                          -STAT- -RUNNING-
  DD01AR0  DD01AR0                                            AVAIL     NO

ASSOCIATED SUBSYSTEM INFORMATION:
                            ENCODED
   -SSID-    -ACCESS INTENT-  -STATE-  -SS ROLE-
  SYS3          UPDATE           6      ACTIVE
```

Backout Error Processing

- Invoked for
  - Dynamic backout failures
  - /ERE backout failures
  - /ERE NOBMP

- Call DBRC to
  - Increment "Backout Needed" counter in DB record

Batch Backout Processing

- Generate the IMS Batch Backout utility JCL
- Execute Batch Backout utility JCL
- The IMS Batch Backout utility
  - Calls DBRC to validate the JCL
  - Performs any backouts needed
  - Calls DBRC upon completion to update the RECON

BACKOUT Record Contents

- Subsystem name (i.e. SSID=)
- Number of units of recovery (i.e. #UORS=)
- Recovery token
- time stamp
- PSB name
- Type (INFLT, ... ,)
- Information regarding the Associated Databases
  - DBD
  - Backed Out (Yes or No)
  - Dynamic Backout failure (Yes or No)

BACKOUT Record Example

```
BACKOUT
 SSID=SYS3       #UORS=2

   RECOVERY TOKEN=E2E8E2F340404040000000000100000000
   TIME=04.060 22:38:01.8                  PSB=PLVAPZ12
               INFLT   BMP   COLDEND
     ASSOCIATED DATA BASES=1

                  BACKED   DYN BKOUT
     -DBD-        -OUT -   -FAILURE-
     DIVNTZ02      NO         NO

   RECOVERY TOKEN=E2E8E2F340404040000000000200000000
   TIME=04.060 22:38:09.2                  PSB=PSBEJK05
               INFLT   BMP   COLDEND
     ASSOCIATED DATA BASES=3

                  BACKED   DYN BKOUT
     -DBD-        -OUT -   -FAILURE-
     DBVHDJ05      NO         NO
     DBOHIDK5      NO         NO
     DXVHIDK5      NO         NO
```

DBRC Diagnostics

Problem Determination and Problem Source Indentification

- Traces
  - DBRC Internal
  - External
- Tools
  - DSPURX00
  - AMASPZAP
  - AMS
    - IDCAMS
- DBRC Dumps
  - Important Control Blocks
    - DSPRSTG
    - DSPGDB
- Messages
  - DSP0300I

Diagnostics: Traces

Problem Determination and Problem Source Indentification

- DBRC Internal
  - Wrap around type
  - Indentifier: TRACETBL
  - Fixed size
    - 4800 entries
    - ZAP DSPTRACE to increase
- DBRC External
  - Started by
    - Start GTF for USR-type records
    - CHANGE.RECON command with TRACEON

Diagnostics: Tools

- DSPURX00
  - Use LIST commands to display RECON records of interest
    - LIST.RECON STATUS
    - LIST.HISTORY
- AMASPZAP
  - Display code
    - CSECT maintenance level in RESLIB
    - Determine offset of an instruction
  - Modify an instruction
    - Force an ABEND at a certain point in the code
- AMS
  - IDCAMS
    - Print RECON records of interest

Diagnostics: AMASPZAP JCL

Problem Determination and Problem Source Indentification

- DUMPT
- //DUMPT JOB BLAKE,MSGLEVEL=1
- //***
- //* JOB NAME: DUMPT   *
- //* JOB ENVIRONMENT: THIS JOB RUNS ON EC MACHINE 'IMSQA'   *
- //* JOB DESCRIPTION: THIS JOB USES AMASPZAP TO DUMP ALL OF A LOAD   *
- //* MODULE OR JUST A CSECT OF IT FROM A PARTICULAR LIBRARY AS   *
- //* SPECIFIED IN THE SYSLIB JCL STATEMENT.   *
- //* MEMBER REFERS TO THE LOAD MODULE   *
- //* CSECT REFERS TO A CSECT WITHIN THE LOAD MODULE   *
- //* ALL WILL DUMP ALL CSECTS WITHIN THE LOAD MODULE   *
- //***
- /*ROUTE PRINT SJFEVMD/BLAKE
- // EXEC PGM=AMASPZAP
- //SYSPRINT DD SYSOUT=A
- //SYSLIB DD DSN=IMS.RESLIB,DISP=SHR
- //SYSIN DD *
- DUMPT MEMBER (CSECT | ALL)
- /*

Diagnostics: Dumps

- DBRC Dumps
  - The DBRC internal trace table
    - Eyecatcher: TRACETBL
    - Wrap around type
    - Provides DBRC process flow details
    - Contains DFSBRLSC FUNC=, ID= codes

- Important DBRC Control Blocks
  - Router Storage: DSPRSTG
    - Contains table of DBRC parts and EPAs
    - Contains pointer to DSPGDB (+x'18')
    - Contains pointer to DBRC trace table (+x'20')

  - Global Data Block: DSPGDB
    - Pointer to DSPRSTG (+x'38')

Diagnostics: TRACETBL

Words 5-8 of the DBRC Trace Table ...

```
...  0554B480 0554FF80 0554C5C0 00000000      *TRACETBL.....-.........E{....*
...  0001BF30 0001E010 BAD9C6A7 08BEDD03      *DSPSTGET.....&........\..RFx....*
... 0001E010 0001E960 BAD9C6A7 08BF8183       *DSPSTGETe&.fe.....\...Z-.RFx..ac*
... 00000000 00000000  BAD9C6A7 08BFD483       *DSPURI00...8.............RFx..Mc*
... D7C8E8E2 C9C3C1D3 40D6D7C5 D5404040       *MODIRCLE    O...PHYSICAL OPEN   *
... 40404040 40404040  00000000  00000000      *                        ........*
...  0001E960 0001F020 BAD9C6A7 08C22703       *DSPSTGETe.....i&..Z-..0..RFx.B..*
...  0001E960 0001F020 BAD9C6A7 08C32F03       *DSPSTFREe......8..Z-..0..RFx.C..*
...  0001E960 0001F020 BAD9C6A7 08C6DB03       *DSPSTGETe.........Z-..0..RFx.F..*
```

Notes:
Words 5, 6, and 7 of the TRACETBL entry are:
- First trace entry
- Last trace enty
- Next trace entry

Word 3 (not shown) of the DSPURI00 entry contains the address of DSPGDB

Diagnostics: DSPRSTG

```
00000D68 05500F48 C0E00000 00000001   *       DSPRSTG .....&..{\......*
00000000 00000000 00000000 00000000    *.......8.......-.............*
00000000 00000000 00000000 00000000    *.............................*
00000000 00000000 C4E2D7E4 D9E2E2E3    *.....................DSPURSST*
00005DE0 00005E50 00000000 00000000    *...<.........)...)\..;&........*
00000000 00000000 00000000 00000000    *.............................*
00000000 00000000 00000000 00000000    *.............................*
C4E2D7C1 C4E2F0F0 80000001 05538440    *.............DSPADS00......d *
C4E2D7C1 D3C4F0F0 80000003 05538C88    *DSPADTIM.......}DSPALD00.......h*
50E050E0 58F050E4 05EF0000  00005130    *DSPAMS00.......\&\&\.0&U........*
C4E2D7C3 C1C2D5F0 80000007 0001A1A8    *&\&\.0&U....... DSPCABN0.......y*
C4E2D7C3 C8D2E6C4 80000009 05549F78    *DSPCEXT0.......hDSPCHKWD........*
C4E2D7C4 C5D8C540 8000000B 0554A4D0    *DSPDEQ00........DSPDEQE ......u}*
C4E2D7C4 D3E3F0F0 8000000D 0553B860    *&\&\.0&U........DSPDLT00.......-*
50E050E0 58F050E4 05EF0000 000051D0    *DSPDTM  ......f.&\&\.0&U.......}*
C4E2D7C8 C9C3C2C7 80000011 0553A640    *DSPHIC00......tQDSPHICBG......w *
C4E2D7C9 C3D7F0F0 80000013 0553E510    *DSPHICED........DSPICP00......V.*
C4E2D7D1 C2E2C1D3 00000015 055DE508    *DSPJBMAI.....)..DSPJBSAL.....)V.*
C4E2D7D1 C2E2C4C2 00000017 055E09F0    *DSPJBSCA.....)6QDSPJBSDB.....;.0*
C4E2D7D1 C2E2C9C3 00000019 055E1AA0    *DSPJBSEL.....)..DSPJBSIC.....;..*
C4E2D7D1 C2E2D9D3 0000001B 055E5388    *DSPJBSOL.....; -DSPJBSRL.....;.h*
```

RECON Record Types: Table 1

Common Name	Type	DSECT	List Name	Comment
RECON Header	X'01'	DSPRCNRC	RECON	
RECON Header Extension	X'01'	DSPRCR1	n/a	Not listed.
RECON Audit Record	X'02'	DSPMUPHD	n/a	Not listed; rarely seen.
PRILOG	X'05'	DSPLOGRC	PRILOG	
Interim PRILOG	X'06'	DSPLOGRC	IPRI	
LOGALL	X'07'	DSPLGARC	LOGALL	
SECLOG	X'09'	DSPLOGRC	SECLOG	
Interim SECLOG	X'0A'	DSPLOGRC	ISEC	
Change Accum Group	X'0F'	DSPCAGRC	CAGRP	
Change Accum Execution	X'11'	DSPCHGRC	CA	

RECON Record Types: Table 2

Common Name	Type	DSECT	List Name	Comment
Available CA	X'51'	DSPCHGRC	CA	
DBDS Group	X'16'	DSPDGRC	DBDSGRP	
Data Base Header	X'18'	DSPDBHRC	DB	
DB Partition	X'19'	DSPPTNRC	DB	
Data Base Data Set	X'20'	DSPDSHRC	DBDS	
Area Recov	X'20'	DSPDSHRC	DBDS	
Area Auth	X'21'	DSPDBHRC	DBDS	
ALLOC	X'28'	DSPALLRC	ALLOC	
Image Copy	X'2D'	DSPIMGRC	IMAGE	
Reorg	X'32'	DSPRRGRC	REORG	

RECON Record Types: Table 3

Common Name	Type	DSECT	List Name	Comment
Recovery	X'37'	DSPRCVRC	RECOV	
Available IC	X'6D'	DSPIMGRC	IMAGE	
Backout	X'35'	DSPBKORC	BACKOUT	
Global Service Group	X'3A'	DSPGSGRC	GSG	
Tracking Subsystem	X'3E'	DSPSSRC	SSYS	
Subsystem	X'3F'	DSPSSRC	SSYS	
Primary SLDS	X'43'	DSPSLDRC	PRISLD	
Tracking Primary SLDS	X'44'	DSPSLDRC	PRITSLD	
Interim PRISLD	X'45'	DSPSLDRC	IPRISL	Seldom seen
Interim Tracking Primary SLDS	X'46'	DSPSLDRC	IPRITSLD	Seldom seen

RECON Record Types: Table 4

Common Name	Type	DSECT	List Name	Comment
Secondary SLDS	X'47'	DSPSLDRC	SECSLD	
Tracking Secondary SLDS	X'48'	DSPSLDRC	SECTSLD	Seldom seen
Interim Secondary SLDS	X'49'	DSPSLDRC	ISECSL	Seldom seen
Interim Tracking Secondary SLDS	X'50'	DSPSLDRC	ISECTSLD	Seldom seen
Primary OLDS	X'53'	DSPOLDRC	PRIOLD	
Interim Primary OLDS	X'55'	DSPOLDRC	IPRIOL	Seldom seen
Secondary OLDS	X'57'	DSPOLDRC	SECOLD	
Interim Primary OLDS	X'59'	DSPOLDRC	ISECOL	Seldom seen

DBRC V8 Topics

- ★ DBRC Enhancements
- ★ 16 MB RECON records
- ★ PRILOG Compression
- ★ Automatic RECON Loss Notification
- ★ DBRC Command Authorization
- ★ Minimum Version Specification
- ★ GENJCL.IC Enhancements
- ★ DEDB Enhancements

Further Information

- IMS website: http://www-3.ibm.com/software/data/ims/
  - Publication Library
    - DBRC Guide and Reference

      IMS V8:

      IMS V7: SC26-9428-01

      IMS V6: SC26-8733-05
  - Presentations
    - DBRC RECON Security (White Paper)
    - Using GENJCL.USER to Allocate IMS HALDB Data Sets
  - Redbooks
    - Database Recovery Control (DBRC) Examples and Usage Hints (SG24-3333-01)