# E40

# DBRC,IMS Tools and Database Recovery

Rick Long
Silican Valley Lab


Anaheim, California October 23 - 27, 2000



RETURN TO INDEX

# Introduction

- Rick Long
  - Silicon Valley Lab (remotely)
  - IMS Data Propagator
  - Classic Connect
- Formerly
  - ITSO IMS Specialist
  - IMS Systems Programmer
  - IMS DBA
  - Application Programmer
- ricklong@us.ibm.com

# Objectives

- DBRC and Index Builder
  - Using DBRC and Index Builder to rebuild secondary index databases
- DBRC and IMS Recovery Saver
  - Using DBRC and IMS Recovery Saver to perform timestamp recoveries

# DBRC and Index Builder

- Backing up only primary data and primary index databases and using Index Builder to rebuild the secondary index databases during recovery processing
  - Saves time to run the backups
    - Backups run frequently
    - Recoveries run infrequently
  - Saves space
    - DASD (if backups are to DASD)
    - Fewer cartridges or labels on cartridges
  - Reduces IMS Logging
    - Secondary indexes registered as non recoverable databases
    - Database log records not written for these databases during update processing

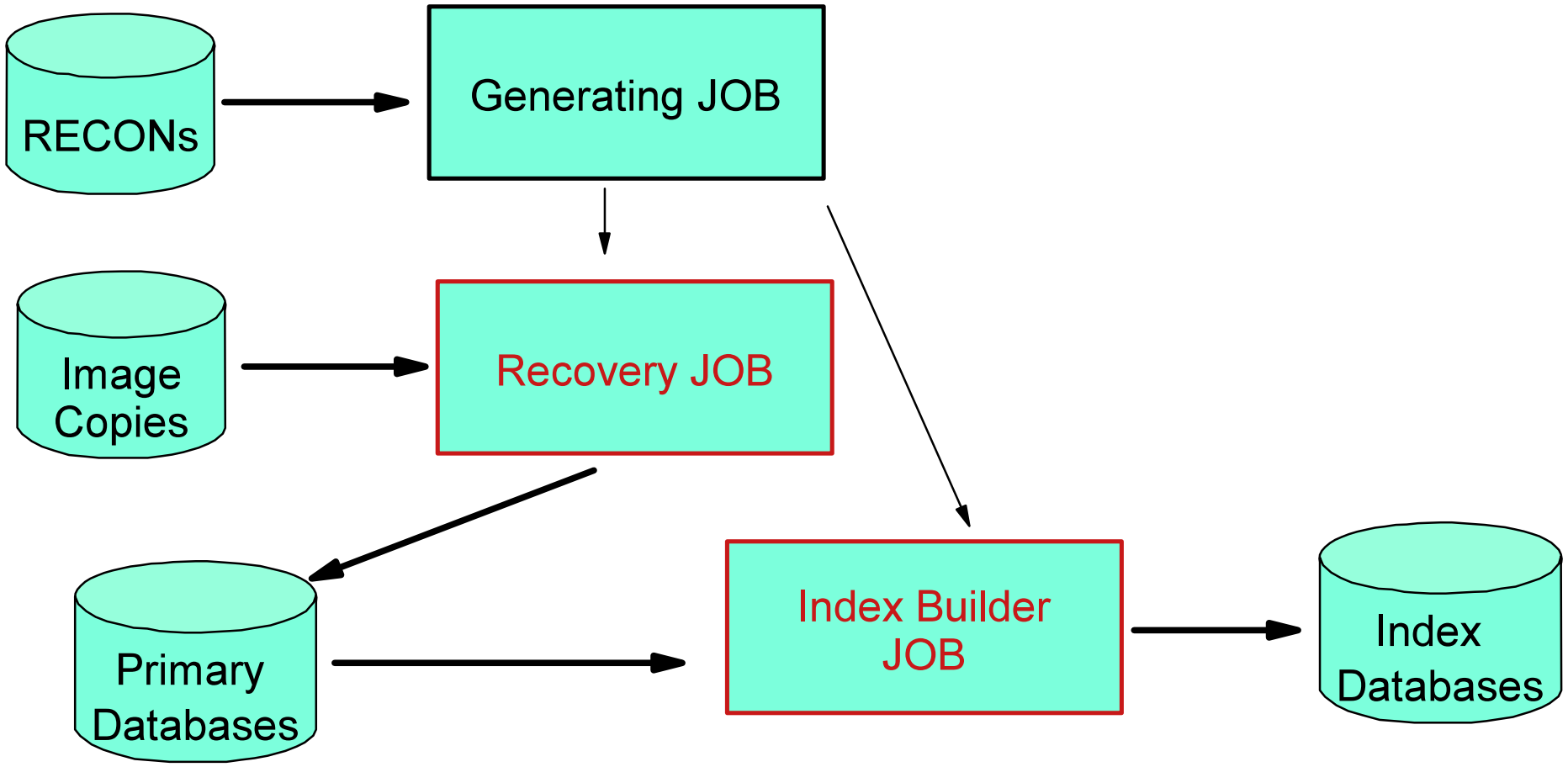
# Which Databases

- Databases which are candidates for this technique
  - Large secondary index databases
  - Databases with multiple secondary indexes
- Databases which are not candidates for this technique
  - Large databases with small index databases
    - Secondary Index database which use SPARSE indexing
  - Small databases where the DASD savings is very small

# Rebuilding Secondary Index Databases

- Rebuild secondary index database during recovery processing
  - Can be generated by DBRC in the same recovery generating JOB
  - Can create one JOB or many
  - Most JOB schedulers can track multiple JOBS

# Job Flow



# Steps

1. **Register Databases**
   - ► RECON Status set to FORCER
     - Data dababases and primary indexes as normal
     - Secondary index databases as non recoverable
   - ► RECON Status set to NORFORCER
     - – Registration not required
     - – Recommend to register as if FORCER is set (Production only)

2. **Register DBDSGRPs**
   - ► Create a DBDSGRP for all non secondary index databases
   - ► Create a second DBDSGRP with one entry for each database which is a source database for a secondary index

# Steps (continued)

3. **Create Skeletal JCL in JCLPDS**
   - One member for recovery steps
   - One member for Index Builder steps

4. **Build GENJCL JOB**
   - GENJCL.RECOV for recovery of primary DBDSGRP
   - GENJCL.USER for Index Builder DBDSGRP

# Database Registration

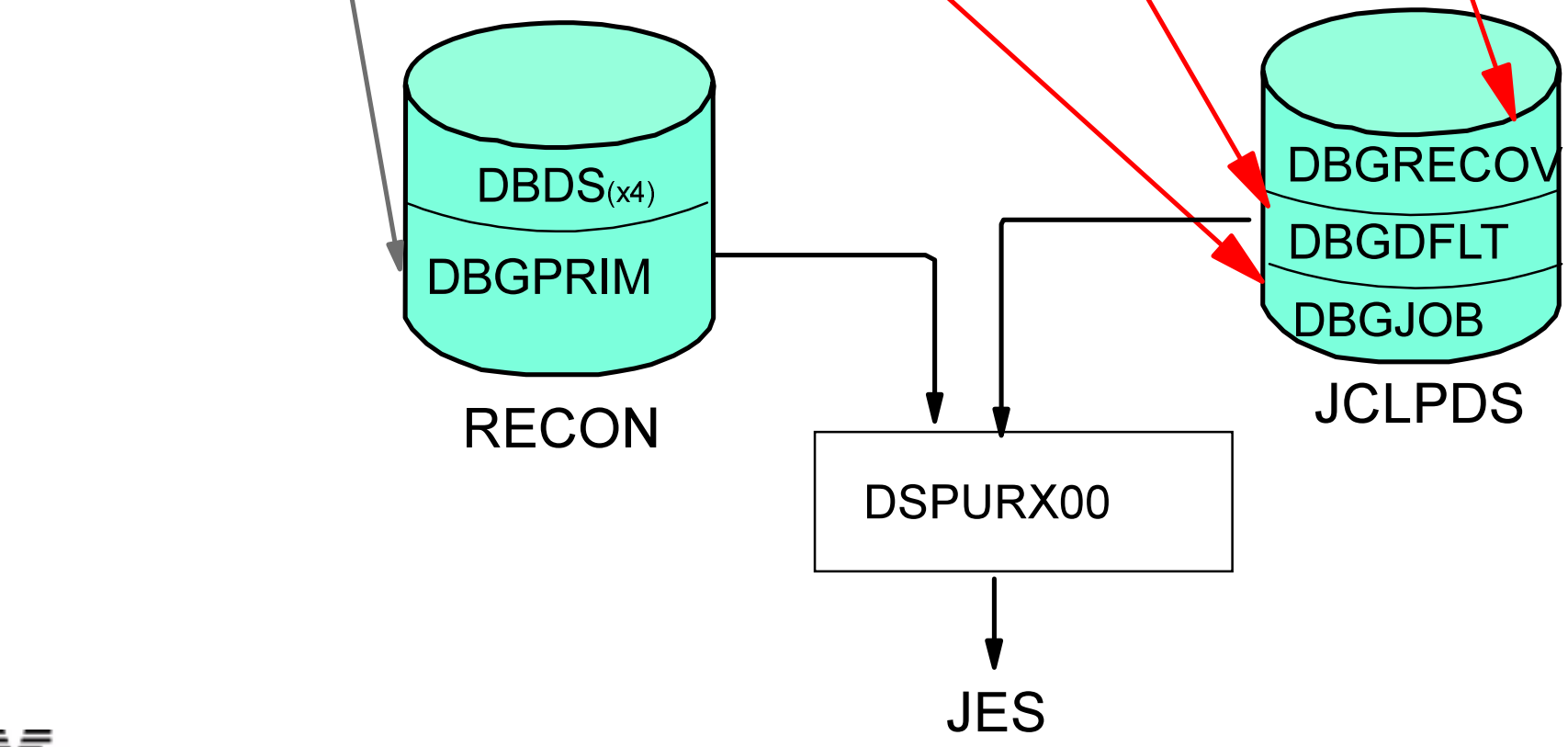
```
INIT.DB   DBD(DBGAMAP) -
          SHARELVL(1)
 INIT.DBDS DBD(DBGAMAP) -
          DDN(DBGAMAP1) -
          DSN(IMS.SJIMSC.DBGAMAP1) -
          GENMAX(10) -
          DEFLTJCL(DBGDFLT) -
           RECOVJCL(DBGRECOV)
 INIT.DBDS DBD(DBGAMAP) -
          DDN(DBGAMAP2) -
          DSN(IMS.SJIMSC.DBGAMAP2) -
          GENMAX(10) -
          DEFLTJCL(DBGDFLT)  -
          RECOVJCL(DBGRECOV)

INIT.DB   DBD(DBGAMAY) -
          NONRECOV -
          SHARELVL(1)
 INIT.DBDS DBD(DBGAMAY) -
          DDN(DBGAMAY) -
          DSN(IMS.SJIMSC.DBGAMAY)  -
          GENMAX(10)  -
          DEFLTJCL(DBGDFLT)  -
          RECOVJCL(DBGRECOV)
```

```
INIT.DB   DBD(DBGAMBP)  -
          SHARELVL(1)
 INIT.DBDS DBD(DBGAMBP)  -
          DDN(DBGAMBP)  -
          DSN(IMS.SJIMSC.DBGAMBP)  -
          GENMAX(10)  -
          DEFLTJCL(DBGDFLT)  -
          RECOVJCL(DBGRECOV)

 INIT.DB   DBD(DBGAMBX) -
          SHARELVL(1)
 INIT.DBDS DBD(DBGAMBX)  -
          DDN(DBGAMBX) -
          DSN(IMS.SJIMSC.DBGAMBX) -
          GENMAX(10)  -
          DEFLTJCL(DBGDFLT)  -

INIT.DB   DBD(DBGAMBY) -
          NONRECOV  -
          SHARELVL(1)
 INIT.DBDS DBD(DBGAMBY) -
          DDN(DBGAMBY) -
          DSN(IMS.SJIMSC.DBGAMBY) -
          GENMAX(10)  -
          DEFLTJCL(DBGDFLT) -
          RECOVJCL(DBGRECOV)
```

```
INIT.DB   DBD(DBGAMBY2) -
          NONRECOV -
          SHARELVL(1)
 INIT.DBDS DBD(DBGAMBY2)  -
          DDN(DBGAMBY2)  -
          DSN(IMS.SJIMSC.DBGAMBY2)  -
          GENMAX(10)  -
          DEFLTJCL(DBGDFLT)  -
          RECOVJCL(DBGRECOV)

INIT.DBDSGRP GRPNAME(DBGGRP1 )
MEMBERS((DBGAMAP ,DBGAMAP1),  -
              (DBGAMAP ,DBGAMAP2),  -
               (DBGAMAY ,DBGAMAY) ,  -
               (DBGAMBP ,DBGAMBP) ,  -
               (DBGAMBX ,DBGAMBX) ,  -
               (DBGAMBY ,DBGAMBY) ,  -
               (DBGAMBY2,DBGAMBY2))

INIT.DBDSGRP GRPNAME(DBGPRIM )
MEMBERS((DBGAMAP ,DBGAMAP1),  -
              (DBGAMAP ,DBGAMAP2),  -
              (DBGAMBP ,DBGAMBP) ,  -
              (DBGAMBX ,DBGAMBX))

INIT.DBDSGRP GRPNAME(DBGINDX )
MEMBERS((DBGAMAY ,DBGAMAY),  -
              (DBGAMBY ,DBGAMBY))
```

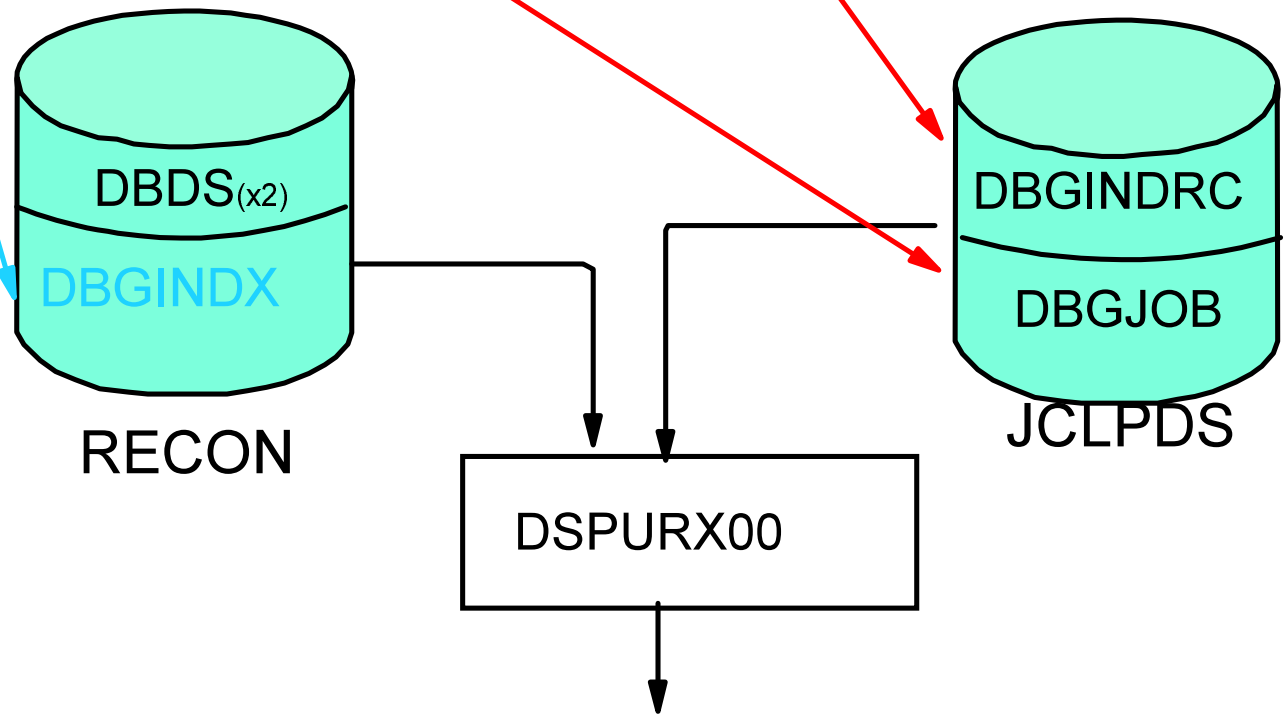
# Generating the Recovery

```
//DBRC    EXEC PGM=DSPURX00
//SYSIN DD *
   GENJCL.RECOV JOB(DBGJOB)  MEMBER(DBGRECOV) -
      GROUP(DBGPRIM) LIST DEFAULTS(DBGDFLT) ONEJOB
  GENJCL.USER  JOB(DBGJOB)  MEMBER(DBGINDRC)  -
     GROUP(DBGINDX)  LIST DEFAULTS(DBGDFLT) ONEJOB
```



# Generating the Index Builder job

```
//DBRC    EXEC PGM=DSPURX00
//SYSIN DD *
   GENJCL.RECOV GROUP(DBGPRIM) MEMBER(DBGRECOV) -
    JOB(DBGJOB)   LIST DEFAULTS(DBGDFLT) ONEJOB
  GENJCL.USER  JOB(DBGJOB) MEMBER(DBGINDRC)  -
      GROUP(DBGINDX) ONEJOB LIST
```



# Recovery Saver

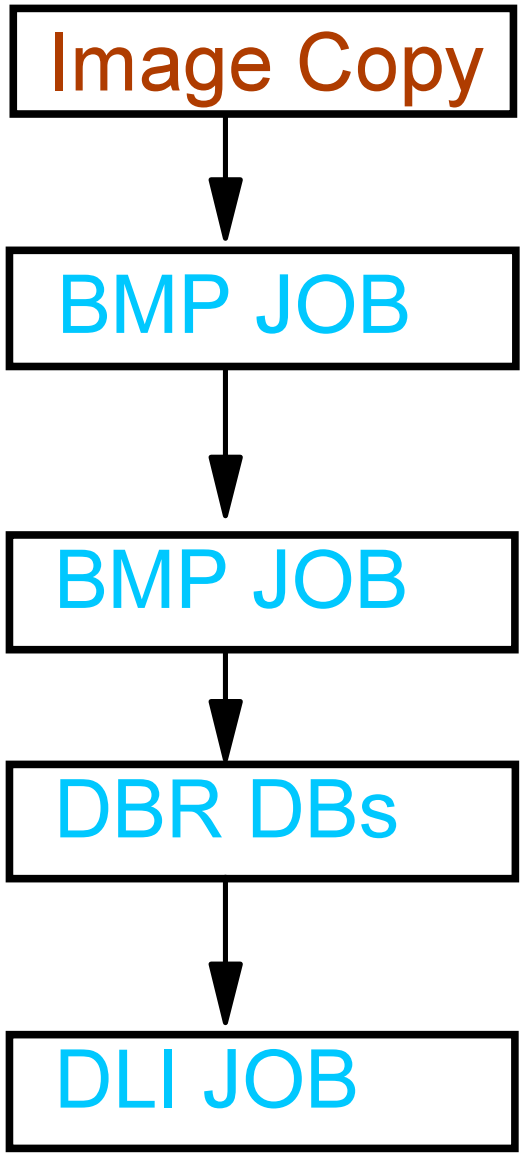
## IMS/ESA Recovery Saver

- Product number: 5655-A68
- Supports IMS/ESA Version 5 and version 6 RECONs
- Mixture of IMS/ESA Version 75 and version 6 logs

# Time Stamp Recovery

- Time Stamp Recovery
  - Recovery of a database to an earlier state
    - At 4: 00 pm, recover the database to the state it was in at 2: 00 pm
  - Typically used to recover from application defects
  - Should be used with due application considerations, especially in online environments
  - Requires prior existence of a recovery point
    - Database update activity quiesced from all systems
    - Data outage
  - Creation of recovery point requires coordinated actions on all data sharing systems

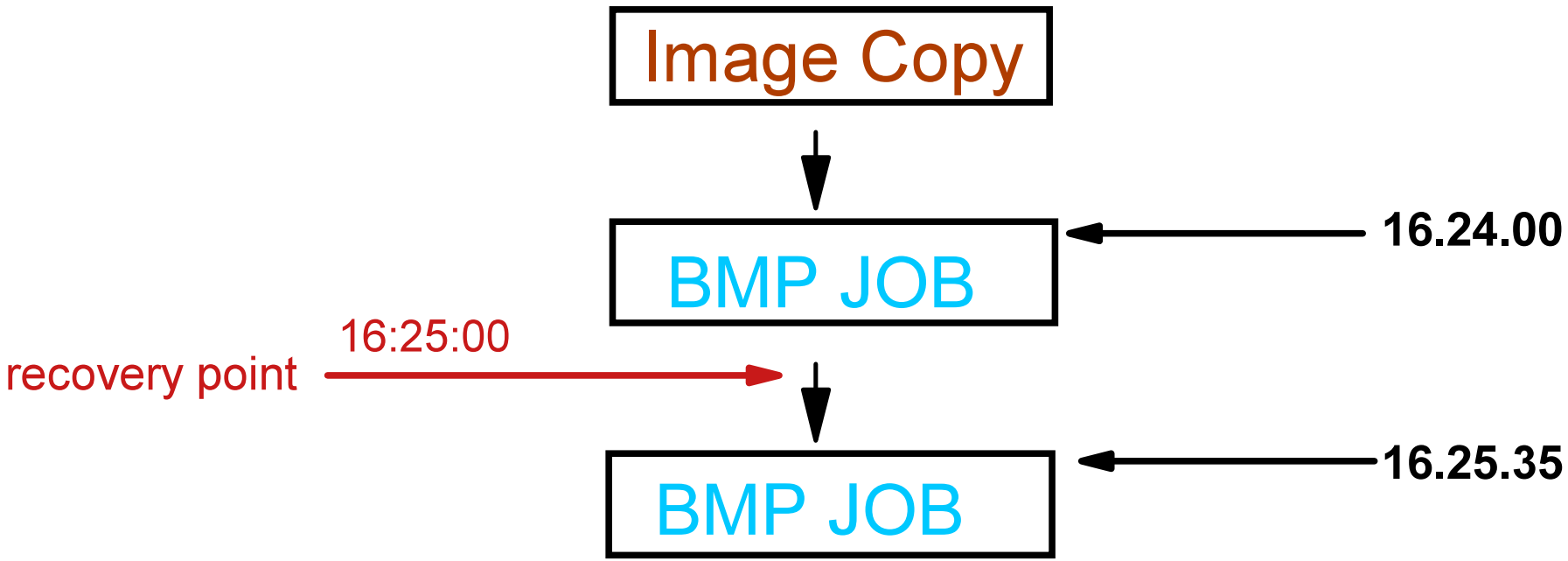
# Time Stamp Recovery Points



# What Recovery Saver Provides

- IMS Recovery Saver changes reality by conditioning a set of IMS logs and a copy of the RECON to allow IMS databases to be recovered to any time stamp
  - Trims IMS log data streams to a common time stamp
    - DB consistency with IMS/ ESA Version 5 logs
    - DB/ DC consistency with IMS/ ESA Version 6 logs
  - Updates the RECON to indicate that all IMS activity ceased at the common time stamp
- Allows time stamp recovery to any time stamp
- Assists in coordinated IMS - DB2 recovery

# Sample Situation of IMS RS Time Stamp Recovery



# Recovery without IMS RS

- The DBRC response to a timestamp recovery without using IMS RS

```
DSP0822I  RECOVERY REQUESTED TO TIMESTAMP 99.158 16:25:00.0
DSP0822I  DBD=DBGAMBP   DDN= DBGAMBP
DSP0823I  RECOVERY REQUEST INCONSISTENT WITH RECON ALLOC  RECORD  OF ALLOCTIME
DSP0823I  99.168 16:24:06.6
DSP0809I  SPECIFIED TIMESTAMP 99.158 16:25:00.0 IS INVALID
DSP0209I  PROCESSING TERMINATED WITH CONDITION CODE = 12
DSP0217I  THE FOLLOWING SYSIN RECORDS HAVE BEEN SKIPPED:
GENJCL.USER  GROUP(DBGPRIM) MEMBER(DBGCHNGE) JOB(CHNGJOB) -
    LIST DEFAULTS(DBGDFLT) ONEJOB
GENJCL.USER  GROUP(DBGINDX) MEMBER(DBGINDRC) JOB(DBGJOB) -
    LIST DEFAULTS(DBGDFLT) ONEJOB
```

# Steps

1. **Create a copy of the RECON**
   - BACKUP.RECON
   - IMS RS Backup Utility
     - Used if RECON LRECL > 32760
     - Backup to sequential dataset
     - Restore to backup RECON data set

2. **Modify backup RECON**
   - CHANGE.RECON STARTNEW CATDS
     - Need only if NONEW is set
     - Allows DBRC commands with only one available RECON
     - Allows recovery with only one available RECON
     - CATDS if not already set
   - LIST.RECON STATUS

# Steps (continued)

3. Run Recovery Saver to time stamp required
   - Use backup RECONs
   - PRIIN - Primary logs are to be used as input
   - CATIN - Log data sets to be allocated using the MVS catalog
   - NOMARKLOGS - Do not mark the primary logs as *in error*
   - NOMARKIC - Do not mark image copy data sets as *in error*

```
//IDMAIN  EXEC PGM=IDPMAIN
//RECON   DD DSN=IMS.SJIMSC.RECON.BACKUP
//SYSIN    DD *
  PRIIN
  CATIN
  NOMARKLOGS
  NOMARKIC
  CUTOFF 1999.156 16:25:00.00 VERIFY
  CATOUT
  ABEND
/*
```

# Steps (continued)

3. Run Recovery Saver to time stamp required
   - CUTOFF - Time stamp to close the logs at recovery time
     - Use VERIFY command on first run to check results
     - Remove VERIFY and rerun
   - CATOUT - Catalog the output data sets created
   - ABEND - Issue U1000 abend on error condition

```
//IDMAIN  EXEC PGM=IDPMAIN
//RECON   DD DSN=IMS.SJIMSC.RECON.BACKUP
//SYSIN    DD *
  PRIIN
  CATIN
  NOMARKLOGS
  NOMARKIC
  CUTOFF 1999.156 16:25:00.00 VERIFY
  CATOUT
  ABEND
/*
```

# Output of IMS RS

```
16:27:50 IDP0003I SCANNING RECON FOR OPEN BATCH LOGS, ARCHIVING GAPS, AND BATCH BACKOUT
LOGS
16:27:50 IDP0004I BUILDING LOG STREAM CONTROL BLOCKS

16:27:50 IDP0005I CUTOFF TIME ADJUSTED TO 1999.168 20:25:00.000000 GMT BASED ON RECON CONTENT

16:27:50 IDP0006I THE FOLLOWING LOG DATA SETS WILL BE INITIALLY SELECTED BASED ON RECON
CONTENT:
        SSID = IMSY      SS START = 1999.167 16:56:05.700838 GMT
        DS START = 1999.167 16:56:05.700838 GMT  DS END = 1999.168 20:26:14.657986 GMT
        UNIT TYPE = 3390      FILE SEQUENCE NUMBER = 0001  DATA SET TYPE = PRISLDS
        DSN = IMS.SJIMSC.SLDSP.IMSY.D99167.T1256057.V01
        VOLSERS = TOTTSI
16:27:50 IDP0007I ALLOCATING WORK FILES
16:27:50 IDP0008I WORK FILE WITH DD NAME = WORK001 ALLOCATED FOR SUBSYSTEM IMSY
              DSN = IMS.SJIMSY.IDPWORK.IMSY
16:27:50 IDP0009I PROCESSING INPUT LOGS FOR SUBSYSTEM IMSY
16:27:50 IDP0050I IMSY     ALLOCATED DSN IMS.SJIMSC.SLDSP.IMSY.D99167.T1256057.V01
16:27:51 IDP0051I IMSY     DEALLOCATED DSN IMS.SJIMSC.SLDSP.IMSY.D99167.T1256057.V01
16:27:51 IDP0010I CUTOFF TIME ADJUSTED TO 1999.168 20:25:00.000000 GMT BASED ON LOG CONTENT
```

# Output of IMS RS (continued)

```
16:27:51 IDP0011I THE FOLLOWING LOG DATA SETS WILL BE PROCESSED BASED ON LOG CONTENT:
       SSID = IMSY     SS START = 1999.167 16:56:05.700838 GMT
       DS START = 1999.167 16:56:05.700838 GMT  DS END = 1999.168 20:26:14.657986 GMT
       UNIT TYPE = 3390     FILE SEQUENCE NUMBER = 0001  DATA SET TYPE = PRISLDS
       DSN = IMS.SJIMSC.SLDSP.IMSY.D99167.T1256057.V01
       VOLSERS = TOTTSI
16:27:51 IDP0012I CREATING OUTPUT LOG FOR SUBSYSTEM IMSY
16:27:51 IDP0050I IMSY    ALLOCATED DSN = IMS.SJIMSY.IMSY.D1999167.T1656057
16:27:51 IDP0051I IMSY    DEALLOCATED DSN = IMS.SJIMSY.IMSY.D1999167.T1656057
16:27:51 IDP0013I THE FOLLOWING OUTPUT LOG DATA SETS HAVE BEEN CREATED:

       SSID = IMSY     SS START = 1999.167 16:56:05.700838 GMT
       DS START = 1999.167 16:56:05.700838 GMT  DS END = 1999.168 20:25:00.000000 GMT
       ALLOCATED UNIT TYPE = SYSDA    DBRC UNIT TYPE = 3390
       DSN = IMS.SJIMSY.IMSY.D1999167.T1656057
       VOLSERS = TOTTSJ
16:27:51 IDP0014I THE FOLLOWING PSBS REQUIRE BACKOUTS FOR IMSY
             NO BACKOUTS REQUIRED
16:27:51 IDP0065I WORK FILE WITH DD NAME = WORK001 DEALLOCATED
16:27:51 IDP0015I RECON CLEANUP OPERATIONS IN PROGRESS
16:27:51 IDP0016I PROGRAM COMPLETED NORMALLY
```

# Steps (continued)

4. Run recovery for databases

- GENJCL.RECOV (no time stamp)
- Must use the backup RECONs

```
//DBRC    EXEC PGM=DSPURX00
//RECON1   DD  DSN=IMS.SJIMSC.RECON1.BACKUP,DISP=SHR
//RECON2   DD  DSN=IMS.SJIMSC.RECON2.BACKUP,DISP=SHR
//SYSIN    DD  *
GENJCL.RECOV GROUP(DBGGRP2) MEMBER(DBGRECOV) -
  JOB(DBGJOB) LIST DEFAULTS(DBGDFLT) ONEJOB -
  USERKEYS(%IRSAVE,'YES')
GENJCL.USER  GROUP(DBGGRP2) MEMBER(DBGCHNGE) -
  JOB(CHNGJOB) LIST DEFAULTS(DBGDFLT) ONEJOB
/*
```

```
%DELETE (%IRSAVE NE 'YES')
//RECON1    DD DSN=IMS.SJIMSC.RECON1.BACKUP,DISP=SHR
//RECON2    DD DSN=IMS.SJIMSC.RECON2.BACKUP,DISP=SHR
%ENDDEL
```

# Steps (continued)

4. Run recovery for databases

- GENJCL.USER to update production RECON with recovery information
- Generated JOB needs to use production RECONs

```
//DBRC    EXEC PGM=DSPURX00
//RECON1   DD  DSN=IMS.SJIMSC.RECON1.BACKUP,DISP=SHR
//RECON2   DD  DSN=IMS.SJIMSC.RECON2.BACKUP,DISP=SHR
//SYSIN    DD  *
GENJCL.RECOV GROUP(DBGGRP2) MEMBER(DBGRECOV) -
  JOB(DBGJOB) LIST DEFAULTS(DBGDFLT) ONEJOB -
  USERKEYS(%IRSAVE,'YES')
GENJCL.USER  GROUP(DBGGRP2) MEMBER(DBGCHNGE) -
  JOB(CHNGJOB) LIST DEFAULTS(DBGDFLT) ONEJOB
/*
```

```
//RLOP%STPNO JOB (999,POK),'RLONG ITSO SJ'
//DBRC    EXEC PGM=DSPURX00
//STEPLIB  DD  DSN=IMS.SJIMSC.RESLIB,DISP=SHR
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  *
```

```
CHANGE.DBDS DBD(%DBNAME) DDN(%DDNAME) RECOV
NOTIFY.RECOV DBD(%DBNAME) DDN(%DDNAME) CURRENT
CHANGE.DBDS DBD(%DBNAME) DDN(%DDNAME) ICON
```

# Steps (continued)

5. Run backouts as required
   - Check the output of the IMS RS job to find the names of the PSBs which require backout
   - Use COLDSTART on the control card to perform backout on deferred backouts

6. Run an Image Copy
   - To avoid having problems with future recoveries it is recommend that you take an image copy to reset the recovery point.


# APPENDIX

# Rebuild Secondary Index Example


**The next 7 pages show the jobs used to rebuild the secondary index databases after recovering the primary databases. I have tried to show all the relevant jobs, skeletal JCL and generated output.**

**The jobs make use of the MVS JCL INCLUDE function to provide the DD cards for the VSAM cluster definitions.**

# GENJCL for Recovery

```
//RLONGRCV JOB (@TS1,FA-C),'RICK LONG - DBG',CLASS=A,
//  MSGCLASS=U,NOTIFY=&SYSUID
//DBRC    EXEC PGM=DSPURX00
//STEPLIB    DD  DSN=IMS.SJIMSC.RESLIB,DISP=SHR
//IMS        DD  DSN=IMS.SJIMSC.DBDLIB,DISP=SHR
//JCLPDS     DD  DSN=IMS.SJIMSC.JCLLIB,DISP=SHR
//*JCLOUT    DD  SYSOUT=*
//JCLOUT     DD  SYSOUT=(*,INTRDR)
//SYSPRINT DD  SYSOUT=*
//SYSIN      DD  *
 GENJCL.RECOV GROUP(DBGPRIM) MEMBER(DBGRECOV)
     JOB(DBGJOB)   LIST DEFAULTS(DBGDFLT) ONEJOB
  GENJCL.USER  GROUP(DBGINDX) MEMBER(DBGINDRC)
      JOB(DBGJOB)  LIST DEFAULTS(DBGDFLT) ONEJOB
//
```

```
%RESLIB='IMS.SJIMSC.RESLIB'
%DBDLIB= 'IMS.SJIMSC.DBDLIB'
%PARMLIB = 'IMS.SJIMSC.UTIL'
%JCLDFLT = 'DBGDFLT'
```

```
//DELDEF  EXEC PGM=IDCAMS,REGION=3072K
 //SYSPRINT  DD SYSOUT=*
 //SYSIN       DD DSN=IMS.SJIMSC.UTIL(%DBDDN),DISP=SHR
 //RCV%STPNO EXEC PGM=DFSRRC00,
 //        PARM='UDR,DFSURDB0,%DBNAME,,,,,,,,,,Y'
 //STEPLIB     DD DISP=SHR,DSN=%RESLIB
 //DFSRESLB  DD DISP=SHR,DSN=%RESLIB
 //SYSPRINT   DD SYSOUT=*
 //IMS           DD DISP=SHR,DSN=%DBDLIB
 //%DBDDN     DD DSN=%DBDSN,
 //                DISP=OLD,AMP='BUFND=30'
 //DFSUDUMP DD DSN=%ICDSN,
 //                DISP=OLD,DCB=BUFNO=10
 %DELETE  (%CADSN EQ '' )
 //DFSUCUM   DD DSN=%CADSN,
 //                DISP=OLD,DCB=BUFNO=10
 %ENDDEL
 %DELETE  (%CADSN NE '' )
 //DFSUCUM     DD DUMMY
 %ENDDEL
 //DFSVSAMP  DSN=IMS.SJIMSC.PROCLIB(DFSVSMDB),DISP=SHR
 //SYSIN         DD  *
 %RCSYSIN
 %SELECT  RLDS((%DBNAME,%DBDDN),FROM(%DSLLGTM))
 //DFSULOG    DD DSN=%LOGDSN,
 //                DCB=RECFM=VB,DISP=OLD
 %ENDSEL
 %DELETE  (%LOGSEL EQ 'YES')
 //DFSULOG    DD DUMMY
 %ENDDEL
```

# GENJCL for Index Builder

```
//RLONGRCV JOB (@TS1,FA-C),'RICK LONG - DBG',CLASS=A,
//  MSGCLASS=U,NOTIFY=&SYSUID
//DBRC    EXEC PGM=DSPURX00
//STEPLIB    DD  DSN=IMS.SJIMSC.RESLIB,DISP=SHR
//IMS        DD  DSN=IMS.SJIMSC.DBDLIB,DISP=SHR
//JCLPDS     DD  DSN=IMS.SJIMSC.JCLLIB,DISP=SHR
//*JCLOUT    DD  SYSOUT=*
//JCLOUT     DD  SYSOUT=(*,INTRDR)
//SYSPRINT DD  SYSOUT=*
//SYSIN      DD  *
 GENJCL.RECOV GROUP(DBGPRIM) MEMBER(DBGRECOV)
    JOB(DBGJOB)   LIST DEFAULTS(DBGDFLT) ONEJOB
  GENJCL.USER  GROUP(DBGINDX) MEMBER(DBGINDRC)
     JOB(DBGJOB)  LIST DEFAULTS(DBGDFLT) ONEJOB
//
```

```
%RESLIB='IMS.SJIMSC.RESLIB'
%DBDLIB= 'IMS.SJIMSC.DBDLIB'
%PARMLIB = 'IMS.SJIMSC.UTIL'
%JCLDFLT = 'DBGDFLT'
```

```
//SYSIN  DD DSN=IMS.SJIMSC.UTIL(DBGAMAY),DISP=SHR
```

IMS.SJIMSC.RUNI(X1)

```
//SYSIN DD DSN=IMS.SJIMSC.UTIL(DBGAMBY),DISP=SHR
//         DD DSN=IMS.SJIMSC.UTIL(DBGAMBY2) ,DISP=SHR
```

IMS.SJIMSC.RUN(X2)

```
//DELDEF  EXEC PGM=IDCAMS,REGION=3072K
//SYSPRINT  DD SYSOUT=*
// INCLUDE MEMBER=I%GRPINDX
//*
//I%STPNO    EXEC PGM=IIUSTART
//STEPLIB     DD DSN=DBT.IIU.SIIULMOD,DISP=SHR
//              DD DSN=%RESLIB,DISP=SHR
//IMS           DD DSN=%DBDLIB,DISP=SHR
//              DD DSN=DBT.IIU.SIIULMOD,DISP=SHR
//IIUIN         DD DSN=%PARMLIB(X%GRPINDX),
//              DISP=SHR
// DFSURIDX   DD DUMMY
//IIUSOUT      DD SYSOUT=*
//IIUPRINT     DD SYSOUT=*
//IIUSNAP      DD SYSOUT=*
//IIUDEBUG     DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//DFSSTAT      DD SYSOUT=*
```

```
 PROC    BLD_SECONDARY,DBGAMAP,SELECTED
INDEX    DBGAMAY
INPUT    DLI,BUF=(16,64)
```

IMS.SJIMSC.UTIL(X1)

```
 PROC    BLD_SECONDARY,DBGAMBP,SELECTED
INDEX    DBGAMBY
INDEX    DBGAMBY2
INPUT    DLI,BUF=(16,24)
```

IMS.SJIMSC.UTIL(X2)

# Rebuild Secondary Indexes - Student Notes


- 1. The JCL INCLUDE members contain the IDCAMS program to the AMS delete and define statements for the secondary index databases. Member X1contains the secondary index DBGAMAY sourced by DBGAMAP. Member X2 contains secondary indexes DBGAMBY & DBGAMAY2 sourced by DBGAMBP. The JCL INCLUDE library is pointed to by the JCLLIB statement

- 2. The X1 and X2 members in the UTIL library contain the Index Builder Control cards for the Index Builder utilitiy. Only one control card per source database is required. Two INDEX statements are used to request that both indexes for DBGAMBP be created.

- 3. The Default member points to the correct libraries for this IMS system. Each IMS system needs to have its own JCLPDS.

# Generated Recovery JOB(s)

```
//RLOP1 JOB (999,POK),'DBRC IMSC',
//       CLASS=A,MSGCLASS=V,MSGLEVEL=(1,1),
//       TIME=(1440),REGION=6M
//MYLIB JCLLIB ORDER=(IMS.SJIMSC.JCLLIB,IMS.SJIMSC.RUN)
//DELDEF     EXEC PGM=IDCAMS,REGION=3072K
//SYSPRINT   DD SYSOUT=*
//SYSIN      DD DSN=IMS.SJIMSC.UTIL(DBGAMAP1),DISP=SHR
//RCV2       EXEC  PGM=DFSRRC00,
//                 PARM='UDR,DFSURDB0,DBGAMAP,,,,,,,,,,Y'
//STEPLIB    DD  DISP=SHR,DSN=IMS.SJIMSC.RESLIB
//DFSRESLB   DD  DISP=SHR,DSN=IMS.SJIMSC.RESLIB
//SYSPRINT   DD  SYSOUT=*
//IMS        DD  DISP=SHR,DSN=IMS.SJIMSC.DBDLIB
//DBGAMAP1 DD DSN=IMS.SJIMSC.DBGAMAP1,
//                 DISP=OLD,AMP='BUFND=30'
//DFSUDUMP DD DSN=IMS.SJIMSC.DBGAMAP1.BKUP.G0012V00,
//                 DISP=OLD,DCB=BUFNO=10
//DFSUCUM    DD DUMMY
//DFSVSAMP   DD DISP=SHR,
//                 DSN=IMS.SJIMSC.PROCLIB(DFSVSMDB)
//SYSIN      DD *
S  DBGAMAP  DBGAMAP1
//DFSULOG    DD DSN=IMS.SJIMSC.DBGB01.G0201V00,
//                 DISP=OLD,DCB=RECFM=VB,DCB=BUFNO=10
//               DD DSN=IMS.SJIMSC.SLDSP.IMSY.D99158.T1351528.V00,
//                 DISP=OLD,DCB=RECFM=VB,DCB=BUFNO=10
//               DD DSN=IMS.SJIMSC.SLDSP.IMSY.D99158.T1404053.V00,
//                 DISP=OLD,DCB=RECFM=VB,DCB=BUFNO=10
//DELDEF     EXEC PGM=IDCAMS,REGION=3072K
//SYSPRINT   DD SYSOUT=*
//SYSIN      DD DSN=IMS.SJIMSC.UTIL(DBGAMAP2),DISP=SHR
//RCV3       EXEC PGM=DFSRRC00,
//                 PARM='UDR,DFSURDB0,DBGAMAP,,,,,,,,,,Y'
//STEPLIB    DD DISP=SHR,DSN=IMS.SJIMSC.RESLIB
//DFSRESLB   DD DISP=SHR,DSN=IMS.SJIMSC.RESLIB
//SYSPRINT   DD SYSOUT=*
//IMS        DD DISP=SHR,DSN=IMS.SJIMSC.DBDLIB
//DBGAMAP2   DD DSN=IMS.SJIMSC.DBGAMAP2,
//           DISP=OLD,AMP='BUFND=30'
//DFSUDUMP   DD DSN=IMS.SJIMSC.DBGAMAP2.BKUP.G0012V00,
//           DISP=OLD,DCB=BUFNO=10
//DFSUCUM    DD DUMMY
//DFSVSAMP   DD DSN=IMS.SJIMSC.PROCLIB(DFSVSMDB),
//           DISP=SHR
//SYSIN      DD *
S  DBGAMAP  DBGAMAP2
//DFSULOG    DD DUMMY
//DELDEF  EXEC PGM=IDCAMS,REGION=3072K
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD DSN=IMS.SJIMSC.UTIL(DBGAMBP),DISP=SHR
//RCV4 EXEC PGM=DFSRRC00,
//         PARM='UDR,DFSURDB0,DBGAMBP,,,,,,,,,,Y'
//STEPLIB      DD DISP=SHR,DSN=IMS.SJIMSC.RESLIB
//DFSRESLB   DD DISP=SHR,DSN=IMS.SJIMSC.RESLIB
//SYSPRINT    DD SYSOUT=*
/IMS            DD DISP=SHR,DSN=IMS.SJIMSC.DBDLIB
//DBGAMBP    DD DSN=IMS.SJIMSC.DBGAMBP,
//               DISP=OLD,AMP='BUFND=30'
//DFSUDUMPDD
//               DSN=IMS.SJIMSC.DBGAMBP.BKUP.G0012V00,
//               DISP=OLD,DCB=BUFNO=10
//DFSUCUM    DD DUMMY
//DFSVSAMP DD DSN=IMS.SJIMSC.PROCLIB(DFSVSMDB)
//SYSIN         DD *
S  DBGAMBP DBGAMBP
//DFSULOG    DD DSN=IMS.SJIMSC.DBGB01.G0201V00,
//               DISP=OLD,DCB=RECFM=VB,DCB=BUFNO=10
//  DD DSN=IMS.SJIMSC.SLDSP.IMSY.D99158.T1351528.V00,
//               DISP=OLD,DCB=RECFM=VB,DCB=BUFNO=10
//  DD DSN=IMS.SJIMSC.SLDSP.IMSY.D99158.T1404053.V00,
//              DISP=OLD,DCB=RECFM=VB,DCB=BUFNO=10
```

# Generated Recovery JOB

(continued)

```
//DELDEF    EXEC PGM=IDCAMS,REGION=3072K
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD DSN=IMS.SJIMSC.UTIL(DBGAMBX),DISP=SHR
//DELDEF    EXEC PGM=IDCAMS,REGION=3072K
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD DSN=IMS.SJIMSC.UTIL(DBGAMBX),DISP=SHR
//RCV5      EXEC PGM=DFSRRC00,
//          PARM='UDR,DFSURDB0,DBGAMBX,,,,,,,,,,Y'
//STEPLIB   DD DISP=SHR,DSN=IMS.SJIMSC.RESLIB
//DFSRESLB  DD DISP=SHR,DSN=IMS.SJIMSC.RESLIB
//SYSPRINT  DD SYSOUT=*
//IMS       DD DISP=SHR,DSN=IMS.SJIMSC.DBDLIB
//DBGAMBX   DD DSN=IMS.SJIMSC.DBGAMBX,
//          DISP=OLD,AMP='BUFND=30'
//DFSUDUMP DD DSN=IMS.SJIMSC.DBGAMBX.BKUP.G0012V00,
//          DISP=OLD,DCB=BUFNO=10
//DFSUCUM   DD DUMMY
//DFSVSAMP DD DSN=IMS.SJIMSC.PROCLIB(DFSVSMDB),
//          DISP=SHR
//SYSIN     DD *
S  DBGAMBX  DBGAMBX
//DFSULOG   DD  DSN=IMS.SJIMSC.DBGB01.G0201V00,
//              DISP=OLD,DCB=RECFM=VB,DCB=BUFNO=10
//          DD DSN=IMS.SJIMSC.SLDSP.IMSY.D99158.T1351528.V00,
//              DISP=OLD,DCB=RECFM=VB,DCB=BUFNO=10
//          DD DSN=IMS.SJIMSC.SLDSP.IMSY.D99158.T1404053.V00,
//              DISP=OLD,DCB=RECFM=VB,DCB=BUFNO=10
//RLOP1 JOB (999,POK),'DBRC IMSC',
//      CLASS=A,MSGCLASS=V,MSGLEVEL=(1,1),
//      TIME=(1440),REGION=6M
//MYLIB JCLLIB ORDER=(IMS.SJIMSC.JCLLIB,IMS.SJIMSC.RUN)
//*
```

```
//DELDEF  EXEC PGM=IDCAMS,REGION=3072K
//SYSPRINT  DD SYSOUT=*
// INCLUDE MEMBER=I1
//I2    EXEC PGM=IIUSTART
//STEPLIB  DD DSN=DBT.IIU.SIIULMOD,DISP=SHR
//        DD DSN=IMS.SJIMSC.RESLIB,DISP=SHR
//IMS     DD DSN=IMS.SJIMSC.DBDLIB,DISP=SHR
//        DD DSN=DBT.IIU.SIIULMOD,DISP=SHR
//IIUIN   DD DSN=IMS.SJIMSC.UTIL(X1),DISP=SHR
//DFSURIDX DD DUMMY
//IIUSOUT  DD SYSOUT=*
//IIUPRINT DD SYSOUT=*
//IIUSNAP  DD SYSOUT=*
//IIUDEBUG DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//DFSSTAT  DD SYSOUT=*
//DELDEF  EXEC PGM=IDCAMS,REGION=3072K
//SYSPRINT  DD SYSOUT=*
// INCLUDE MEMBER=I2
//I3    EXEC PGM=IIUSTART
//STEPLIB  DD DSN=DBT.IIU.SIIULMOD,DISP=SHR
//        DD DSN=IMS.SJIMSC.RESLIB,DISP=SHR
//IMS     DD DSN=IMS.SJIMSC.DBDLIB,DISP=SHR
//        DD DSN=DBT.IIU.SIIULMOD,DISP=SHR
//IIUIN   DD DSN=IMS.SJIMSC.UTIL(X2),DISP=SHR
//DFSURIDX DD DUMMY
//IIUSOUT  DD SYSOUT=*
//IIUPRINT DD SYSOUT=*
//IIUSNAP  DD SYSOUT=*
//IIUDEBUG DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//DFSSTAT  DD SYSOUT=*
```

# IMS Recovery Saver Example


**The next 5 pages show the jobs used to run IMS Recovery Saver. I have tried to show all the relevant jobs, skeletal JCL and generated output.**

**I have modified the members here slightly to fit the relevant information on the page. I have tried to ensure I have not changed anything which would have affected the generation of the jobs.**

**This example is intended to show what kinds of things can be done.**

# IMS Recovery Saver Example

- Create backup RECON for use by IMS RS
  - Allocate new RECON VSAM cluster
  - Backup RECON
  - CHANGE.RECON
    - STARTNEW to allow use with only one RECON
    - CATDS (if not already set)

```
//RLONBRCN JOB (@TS3,FA33),'RICKLONG ITSOSJ ',MSGCLASS=U,
// CLASS=A,NOTIFY=&SYSUID
//ALLOCATE EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN    DD  DSN=IMS.SJIMSC.UTIL(RECNBKUP),DISP=SHR
//*
//DBRC    EXEC PGM=DSPURX00
//STEPLIB  DD  DSN=IMS.SJIMSC.RESLIB,DISP=SHR
//RECON1   DD  DSN=IMS.SJIMSC.RECON1,DISP=SHR
//BACKUP1  DD  DSN=IMS.SJIMSC.RECON1.BACKUP,DISP=SHR
//RECON2   DD  DSN=IMS.SJIMSC.RECON2,DISP=SHR
//BACKUP2  DD  DSN=IMS.SJIMSC.RECON2.BACKUP,DISP=SHR
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  *
 BACKUP.RECON BOTH
//*
//STATUS  EXEC PGM=DSPURX00
//STEPLIB  DD  DSN=IMS.SJIMSC.RESLIB,DISP=SHR
//RECON1   DD  DSN=IMS.SJIMSC.RECON1.BACKUP,DISP=SHR
//RECON2   DD  DSN=IMS.SJIMSC.RECON2.BACKUP,DISP=SHR
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  *
 LIST.RECON STATUS
 CHANGE.RECON STARTNEW CATDS
 LIST.RECON STATUS
//
```

# IMS Recovery Saver Example

- Run IMS RS with time stamp to condition IMS logs
  - DEFAULTS DD sets output data sets attributes
  - SYSIN DD defines time stamp, output log data

```
//RLONGIDP JOB (@TS1,FA33),'R.LONG',MSGCLASS=U,
//  CLASS=A,NOTIFY=&SYSUID
//IDMAIN  EXEC PGM=IDPMAIN,REGION=0M
//STEPLIB   DD DSN=IMS.RECVSAVE.LOADLIB,DISP=SHR
//SYSPRINT  DD SYSOUT=*
//RECON     DD DSN=IMS.SJIMSC.RECON1.BACKUP,DISP=OLD
//DEFAULTS  DD *
OUTUNIT SYSDA
OUTSPACE 0050
OUTHLQ IMS.SJIMSY
WRKHLQ IMS.SJIMSY
WRKSPACE 0100
DEFREL 6.1
//SYSIN     DD *
 PRIIN
 CATIN
 NODELSSYS
 NOMARKLOGS
 NOMARKIC
 CUTOFF 1999.168 16:25:00.00
 CATOUT
 ABEND
//*
```

# IMS Recovery Saver Example

- This is the job to create the recovery and notify the production RECON that the recovery has been run
  - The first GENJCL creates the recovery job
  - The second GENJCL creates a DBRC job to NOTIFY the production RECONs of the recovery
  - The JCLPDS members are on the next page

```
//DBRC    EXEC PGM=DSPURX00
//STEPLIB  DD  DSN=IMS.SJIMSC.RESLIB,DISP=SHR
//RECON1   DD DSN=IMS.SJIMSC.RECON1.BACKUP,
//                DISP=SHR
//RECON2   DD DSN=IMS.SJIMSC.RECON2.BACKUP,
//                DISP=SHR
//JCLPDS   DD  DSN=IMS.SJIMSC.JCLLIB,DISP=SHR
//JCLOUT   DD  DSN=IMS.SJIMSC.RSRUN(GENDRECV),
//                DISP=SHR
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  *
 GENJCL.RECOV GROUP(DBGGRP2) MEMBER(DBGRECOV) -
 JOB(DBGJOB)  LIST ONEJOB  USERKEYS(%IRSAVE,'YES')
 GENJCL.USER  GROUP(DBGGRP2) MEMBER(DBGCHNGE)
   JOB(CHNGJOB) LIST DEFAULTS(DBGDFLT) ONEJOB
/*
```

# JCLPDS Members:

DBGRECOV

```
//DELDEF  EXEC PGM=IDCAMS,REGION=3072K
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD DSN=IMS.SJIMSC.UTIL(%DDNAME),DISP=SHR
//RCV%STPNO EXEC PGM=DFSRRC00,
//         PARM='UDR,DFSURDB0,%DBNAME,,,,,,,,,,Y'
%DELETE (%IRSAVE NE 'YES')
//RECON1    DD DSN=IMS.SJIMSC.RECON1.BACKUP,DISP=SHR
//RECON2    DD DSN=IMS.SJIMSC.RECON2.BACKUP,DISP=SHR
%ENDDEL
//STEPLIB   DD DISP=SHR,DSN=IMS.SJIMSC.RESLIB
//IMS         DD DSN=IMS.SJIMSC.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//%DBDDN    DD DSN=%DBDSN,DISP=OLD,AMP='BUFND=30'
//DFSUDUMP  DD DSN=%ICDSN,DISP=OLD,DCB=BUFNO=10
%DELETE  (%CADSN EQ '')
 //DFSUCUM   DD DSN=%CADSN,DISP=OLD,DCB=BUFNO=10
%ENDDEL
%DELETE  (%CADSN NE '')
//DFSUCUM   DD DUMMY
%ENDDEL
//DFSVSAMP  DD DSN=IMS.SJIMSC.UTIL(VSAMP),DISP=SHR
//SYSIN     DD *
%RCSYSIN
%SELECT  RLDS((%DBNAME,%DBDDN),FROM(%DSLLGTM))
//DFSULOG   DD DSN=%LOGDSN,DISP=OLD
%ENDSEL
%DELETE  (%LOGSEL EQ 'YES')
//DFSULOG   DD DUMMY
%ENDDEL
```

```
CHANGE.DBDS DBD(%DBNAME) DDN(%DDNAME) -
 RECOV
 NOTIFY.RECOV DBD(%DBNAME) DDN(%DDNAME) -
 CURRENT
```

DBGCHNGE

```
//RLOP%STPNO JOB (999,POK),'RLONG ITSO SJ',
//         CLASS=A,MSGCLASS=U,REGION=6M
//DBRC    EXEC PGM=DSPURX00
//STEPLIB   DD  DSN=IMS.SJIMSC.RESLIB,DISP=SHR
 //SYSPRINT DD  SYSOUT=*
//SYSIN    DD  *
```

CHNGJOB

```
//RLOP%STPNO JOB (999,POK),'DBRC IMSC',
//         CLASS=A,MSGCLASS=V,MSGLEVEL=(1,1),
//         TIME=(1440),REGION=6M
//*
```

DBGJOB

# Generated Recovery JOB(s)

RETURN TO INDEX

- Student Notes

```
//RLOP1 JOB (999,POK),'DBRC IMSC',
//  C LASS=A,MSGCLASS=V,MSGLEVEL=(1,1),
//        TIME=(1440),REGION=6M
//DELDEF  EXEC PGM=IDCAMS,REGION=3072K
//SYSPRINT  DD SYSOUT=*
//SYSIN         DD DSN=IMS.SJIMSC.UTIL(DBGAMBP),DISP=SHR
//RCV2 EXEC PGM=DFSRRC00,
//         PARM='UDR,DFSURDB0,DBGAMBP,,,,,,,,,,Y'
//RECON1     DD DSN=IMS.SJIMSC.RECON1.BACKUP,DISP=SHR
//RECON2     DD DSN=IMS.SJIMSC.RECON2.BACKUP,DISP=SHR
//STEPLIB      DD DISP=SHR,DSN=IMS.SJIMSC.RESLIB
//IMS            DD DSN=IMS.SJIMSC.DBDLIB,DISP=SHR
//SYSPRINT   DD SYSOUT=*      //SYSPRINT    DD SYSOUT=*
//DBGAMBP   DD DSN=IMS.SJIMSC.DBGAMBP,
//                DISP=OLD,AMP='BUFND=30'
//DFSUDUMP DD DSN=IMS.SJIMSC.DBGAMBP.BKUP.G0014V00,
//           DISP=OLD,DCB=BUFNO=10
//DFSUCUM    DD DUMMY
//DFSVSAMP  DD,DSN=IMS.SJIMSC.UTIL(VSAMP),DISP=SHR
//SYSIN     DD *
S  DBGAMBP  DBGAMBP
/*
//DFSULOG   DD DSN=IMS.SJIMSY.IMSY.D1999167.T1656057,
//           DCB=(RECFM=VB,BUFNO=10),DISP=SHR
//DELDEF  EXEC PGM=IDCAMS,REGION=3072K
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD DSN=IMS.SJIMSC.UTIL(DBGAMBX),DISP=SHR
//RCV3 EXEC PGM=DFSRRC00,
//         PARM='UDR,DFSURDB0,DBGAMBX,,,,,,,,,,Y'
//STEPLIB      DD DISP=SHR,DSN=IMS.SJIMSC.RESLIB
//RECON1       DD DSN=IMS.SJIMSC.RECON1.BACKUP,DISP=SHR
//RECON2       DD DSN=IMS.SJIMSC.RECON2.BACKUP,DISP=SHR
//SYSPRINT   DD SYSOUT=*
//IMS            DD DSN=IMS.SJIMSC.DBDLIB,DISP=SHR
//DBGAMBX   DD DSN=IMS.SJIMSC.DBGAMBX,
//           DISP=OLD,AMP='BUFND=30'
//DFSUDUMP  DD DSN=IMS.SJIMSC.DBGAMBX.BKUP.G0014V00,
//           DISP=OLD,DCB=BUFNO=10
//DFSUCUM   DD DUMMY
//DFSVSAMP  DD DSN=IMS.SJIMSC.UTIL(VSAMP0),DISP=SHR
//SYSIN     DD *
S  DBGAMBX  DBGAMBX
/*
//DFSULOG   DD DSN=IMS.SJIMSY.IMSY.D1999167.T1656057,
//           DCB=(RECFM=VB,BUFNO=10),DISP=SHR
//RLOP1 JOB (999,POK),'RLONG ITSO SJ',
//        CLASS=A,MSGCLASS=U,REGION=6M
//DBRC    EXEC PGM=DSPURX00
//STEPLIB  DD  DSN=IMS.SJIMSC.RESLIB,DISP=SHR
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  *
  CHANGE.DBDS DBD(DBGAMBP) DDN(DBGAMBP) RECOV
  NOTIFY.RECOV DBD(DBGAMBP) DDN(DBGAMBP) CURRENT
  CHANGE.DBDS DBD(DBGAMBX) DDN(DBGAMBX) RECOV
  NOTIFY.RECOV DBD(DBGAMBX) DDN(DBGAMBX) CURRENT
```