

Unit 11e – Prepare the IVP Application and IMS System for Execution

IMS CATALOG / IMS Managed ACB is missing here ;
Extra set of foils available in class KC220

IV3E001T: Build System

Help

Execution (LST Mode) - DBT

Row 59 to 75 of 215

Command ==>

Scroll ==> CSR

Action Codes: Brm Doc Edm eNt eXe spR

JOB/Task	Step	Title.....
----------	------	------------

IV3E101J	E1	JOB - Allocate Data Sets
----------	----	--------------------------

IV3E201J	E2	JOB - DBDGENs
----------	----	---------------

IV3E202J	E2	JOB - PSBGENs
----------	----	---------------

IV3E203J	E2	JOB - ACBGEN
----------	----	--------------

IV3E204J	E2	JOB - MFS Language Utility
----------	----	----------------------------

IV3E206J	E2	JOB - Assembly/Bind Applications
----------	----	----------------------------------

IV3E207J	E2	JOB - Assembly/Bind DFSIDEF0 (Defaults)
----------	----	---

IV3E301J	E3	JOB - Create Dynamic Allocation Members
----------	----	---

IV3E302J	E3	JOB - Add Control Statements to IMS.PROCLIB
----------	----	---

IV3E303J	E3	JOB - Add CSL Members to IMS.PROCLIB
----------	----	--------------------------------------

IV3E304J	E3	JOB - Add CQS Members to IMS.PROCLIB
----------	----	--------------------------------------

IV3E305J	E3	JOB - Define EXEC PARM Defaults
----------	----	---------------------------------

IV3E306T	E3	TASK - Syntax Checker Sample
----------	----	------------------------------

----- STEPS NOT SHOWN -----

IV3E308J	E3	JOB - Define DRA Start-up Table
----------	----	---------------------------------

----- STEPS NOT SHOWN -----

IV3E313J	E3	JOB - Copy Jobs to IMS JOBS
----------	----	-----------------------------

IV3E314J	E3	JOB - Copy DBRC Skeletons to IMS.PROCLIB
----------	----	--

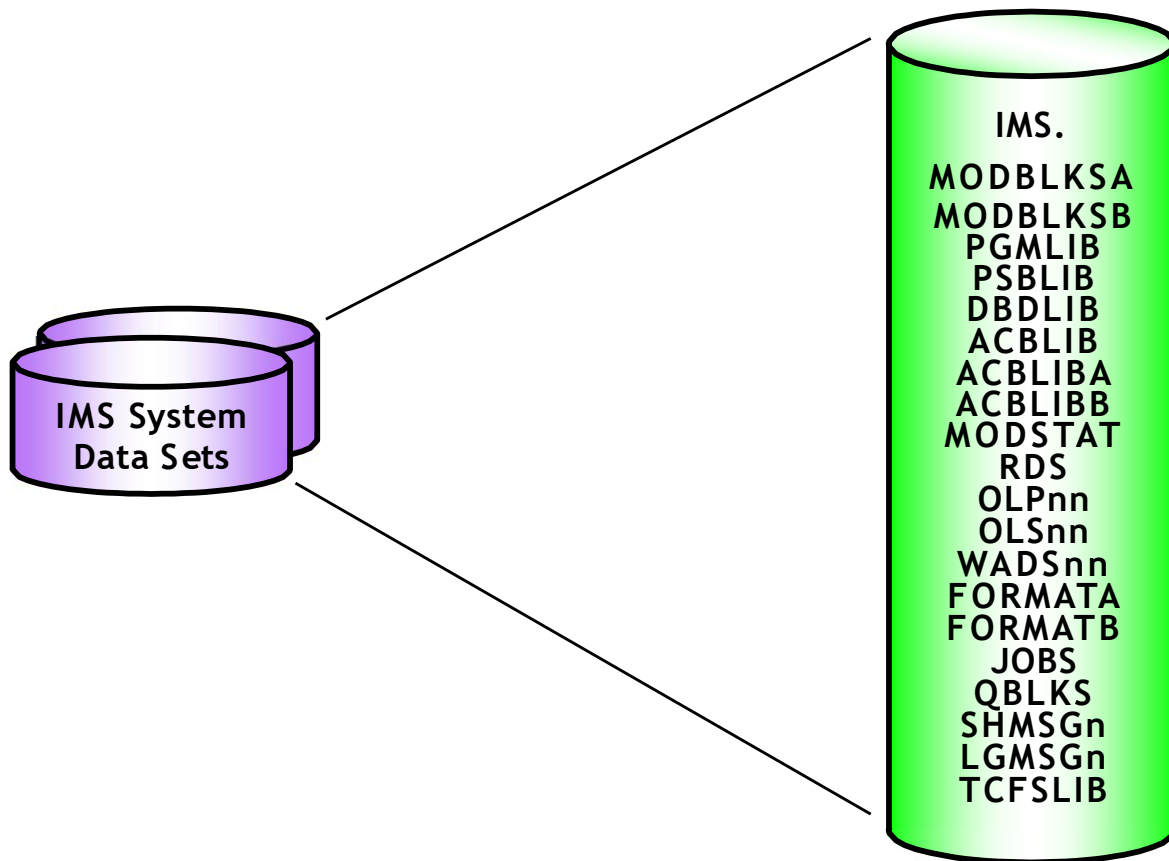
IV3E001T: Build System Page 4

Notes:

In this unit, we will first be focusing on some of the *E* steps. The E steps complete the process of preparing an IMS system to be started; these E steps also prepare applications for execution by performing activities such as Assemblies, links, as well as DBD, PSB and ACB Gens.

Some steps are added, deleted or renumbered when designing IVP for new IMS versions.

IV3E101J: Allocate Data Sets overview



Notes:

These are all system data sets. Some are obviously used by IMS (for example, the OLDS and WADS log data sets); others like PSBLIB are not directly accessed by IMS but ARE common data sets for ALL applications.

IV3E101J: Allocate Data Sets

```
//SCRATCH EXEC PGM=IDCAMS,DYNAMNBR=300
```

```
//SYSPRINT DD SYSOUT=*
```

```
//SYSIN DD *
```

```
DELETE IMS.PGMLIB
```

```
ALLOCATE -
```

```
DSNAME ('IMS.PGMLIB')
```

```
FILE (PGMLIB)
```

```
RECFM(U)
```

```
BLKSIZE (6233)
```

```
DSORG (PO)
```

```
NEW CATALOG
```

```
SPACE (6,2) CYL
```

```
DIR (46)
```

```
VOL (IMS123)
```

```
UNIT (3390) SMS ????'
```

```
DELETE IMS.PSBLIB
```

```
ALLOCATE
```

```
DSNAME ('IMS.PSBLIB')
```

```
FILE (PSBLIB)
```

```
RECFM(U)
```

```
BLKSIZE (6233)
```

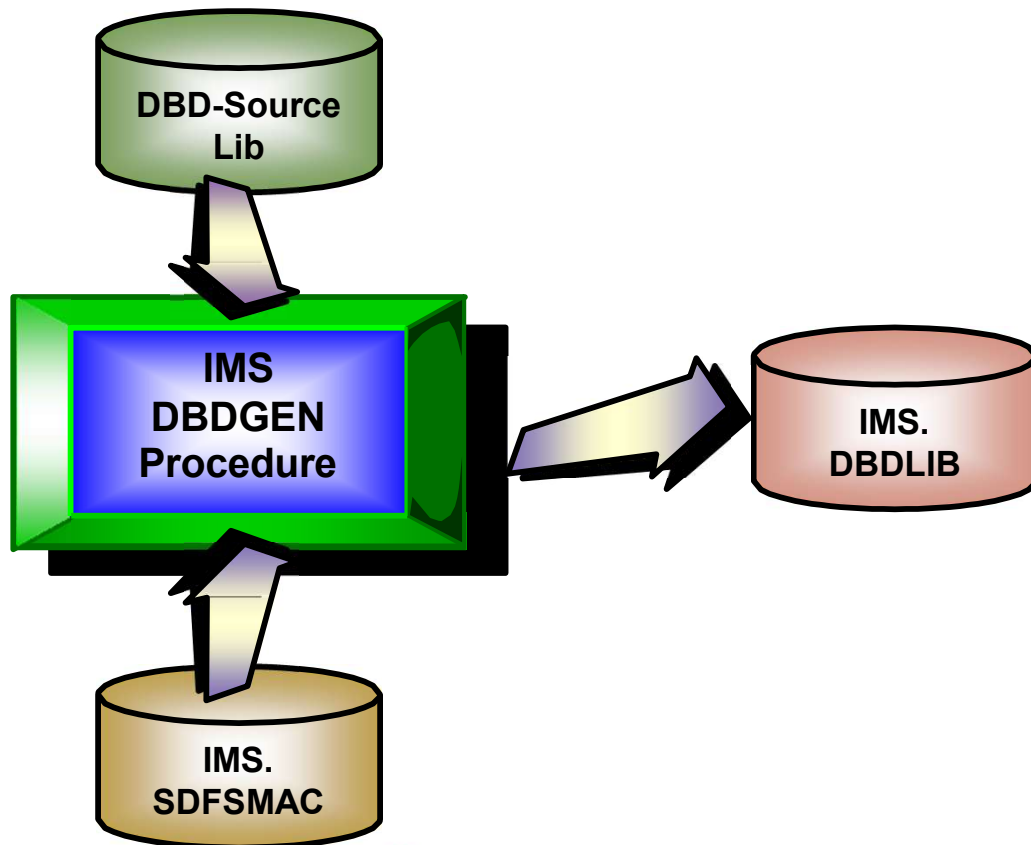
```
DSORG (PO)
```

```
NEW CATALOG
```



KISSLICH CONSULTING

IV3E201J: DBDGEN overview



IV3E201J: DBDGEN JCL

```
//DI21PART EXEC PROC=DBDGEN,MBR=DI21PART,SOUT='*'
XX      PROC MBR=TEMPNAME,SOUT='*',RGN=256K,SYS2=
XXC     EXEC PGM=ASMA90,REGION=&RGN,PARM='OBJECT,NODECK'
IEFC653I SUBSTITUTION JCL - PGM=ASMA90,REGION=256K,PARM='OBJECT,NODECK'
XXSYSLIB DD DSN=IMS.&SYS2.MACLIB,DISP=SHR
IEFC653I SUBSTITUTION JCL - DSN=IMS.SDFSMA,DISP=SHR
XXSYSLIN DD UNIT=SYSDA,DISP=(,PASS),SPACE=(80,(100,100),RLSE),
XX      DCB=(BLKSIZE=80,RECFM=F,LRECL=80)
XXSYSPRINT DD SYSOUT=&SOUT,DCB=BLKSIZE=1089,SPACE=(121,(300,300),RLSE,,ROUND)
IEFC653I SUBSTITUTION JCL - SYSOUT=*,DCB=BLKSIZE=1089,SPACE=(121,(300,300),RLSE,,RO
XXSYSUT1 DD UNIT=SYSDA,DISP=(,DELETE),SPACE=(CYL,(10,5))
//C.SYSIN DD DISP=SHR,
//      DSN=IMS.SDFSISRC(DI21PART)
XXL     EXEC PGM=IEWL,PARM='XREF,LIST',
XX      COND=(0,LT,C),REGION=120K
XXSYSLIN DD DSN=*.C.SYSLIN,DISP=(OLD,DELETE)
XXSYSPRINT DD SYSOUT=&SOUT,DCB=BLKSIZE=1089,SPACE=(121,(90,90),RLSE)
IEFC653I SUBSTITUTION JCL - SYSOUT=*,DCB=BLKSIZE=1089,SPACE=(121,(90,90),RLSE)
XXSYSLMOD DD DISP=SHR,DSN=IMS.&SYS2.DBDLIB(&MBR)
IEFC653I SUBSTITUTION JCL - DISP=SHR,DSN=IMS.DBDLIB(DI21PART)
XXSYSUT1 DD UNIT=(SYSDA,SEP=(SYSLMOD,SYSLIN)),
XX      SPACE=(1024,(100,10),RLSE),DISP=(,DELETE)
//*
```



KISSLICH CONSULTING

IV3E201J: DBDGEN input

1 *

```
11      DBD      NAME=DI21PART,ACCESS=(HISAM,VSAM)
12      DATASET  DD1=DI21PART,DEVICE=3390,OVFLW=DI21PARO,      X
          SIZE=(2048,2048),RECORD=(678,678)
13      SEGM     NAME=PARTROOT,PARENT=0,BYTES=50,FREQ=250
14      FIELD    NAME=(PARTKEY,SEQ),TYPE=C,BYTES=17,START=1
15      SEGM     NAME=STANINFO,PARENT=PARTROOT,BYTES=85,FREQ=1
16      FIELD    NAME=(STANKEY,SEQ),TYPE=C,BYTES=2,START=1
17      SEGM     NAME=STOKSTAT,PARENT=PARTROOT,BYTES=160,FREQ=2
18      FIELD    NAME=(STOCKEY,SEQ),TYPE=C,BYTES=16,START=1
19      SEGM     NAME=CYCCOUNT,PARENT=STOKSTAT,BYTES=25,FREQ=1
20      FIELD    NAME=(CYCLKEY,SEQ),TYPE=C,BYTES=2,START=1
21      SEGM     NAME=BACKORDR,PARENT=STOKSTAT,BYTES=75,FREQ=0
22      FIELD    NAME=(BACKKEY,SEQ),TYPE=C,BYTES=10,START=1
23      DBDGEN
```

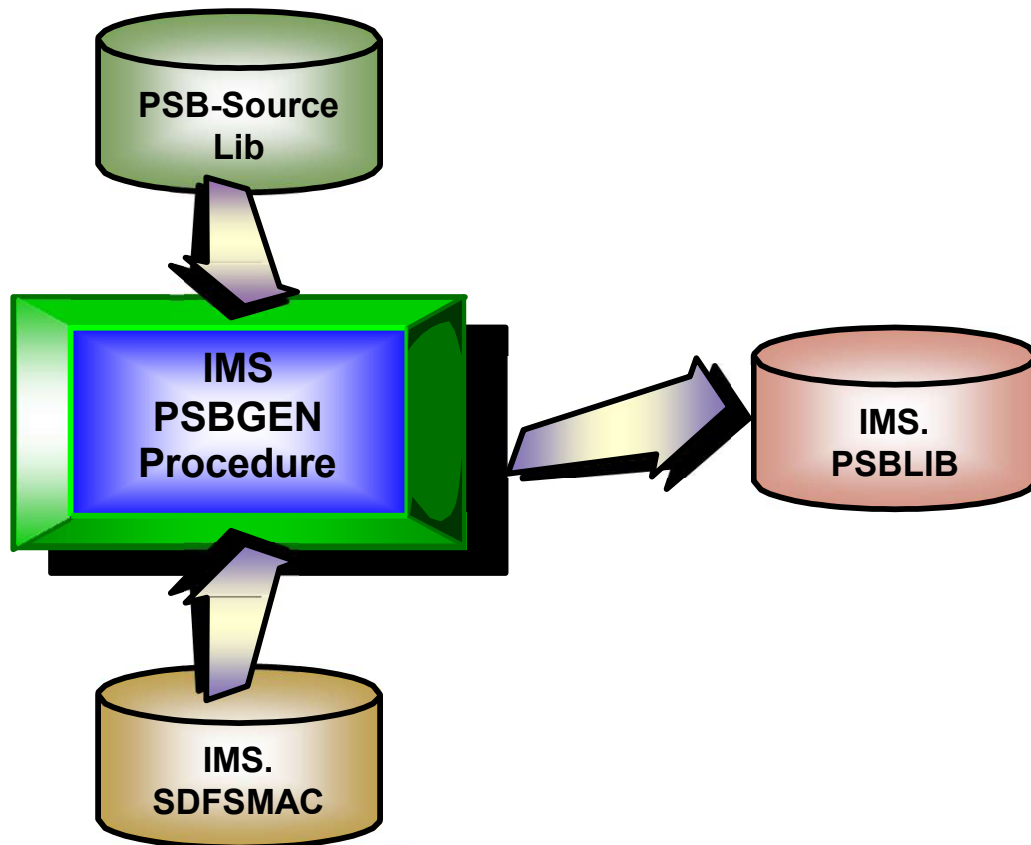
IV3E201J: DBDGEN Job output

```
67+*,* * * * *
68+*,*
69+*,* RECOMMENDED VSAM DEFINE CLUSTER PARAMETERS
70+*,*
71+*,* * * * *
73+*,* * * * *
74+*,*
75+*,* *NOTE1
76+*,* DEFINE CLUSTER (NAME(DI21PART) -
77+*,* INDEXED KEYS (17,6) -
78+*,* RECORDSIZE (678,678)) -
79+*,* DATA (CONTROLINTERVALSIZE (2048))
80+*,*
81+*,* *NOTE1 - SHOULD SPECIFY DSNAME FOR DD DI21PART
82+*,*
83+*,* * * * *
85+*,* * * * *
86+*,*
87+*,* *NOTE2
88+*,* DEFINE CLUSTER (NAME(DI21PARO) NONINDEXED -
89+*,* RECORDSIZE (678,678) -
90+*,* CONTROLINTERVALSIZE (2048))
91+*,*
92+*,* *NOTE2 - SHOULD SPECIFY DSNAME FOR DD DI21PARO
93+*,*
94+*,* * * * *
```



KISSLICH CONSULTING

IV3E202J: PSBGEN overview



IV3E202J: PSBGEN JCL

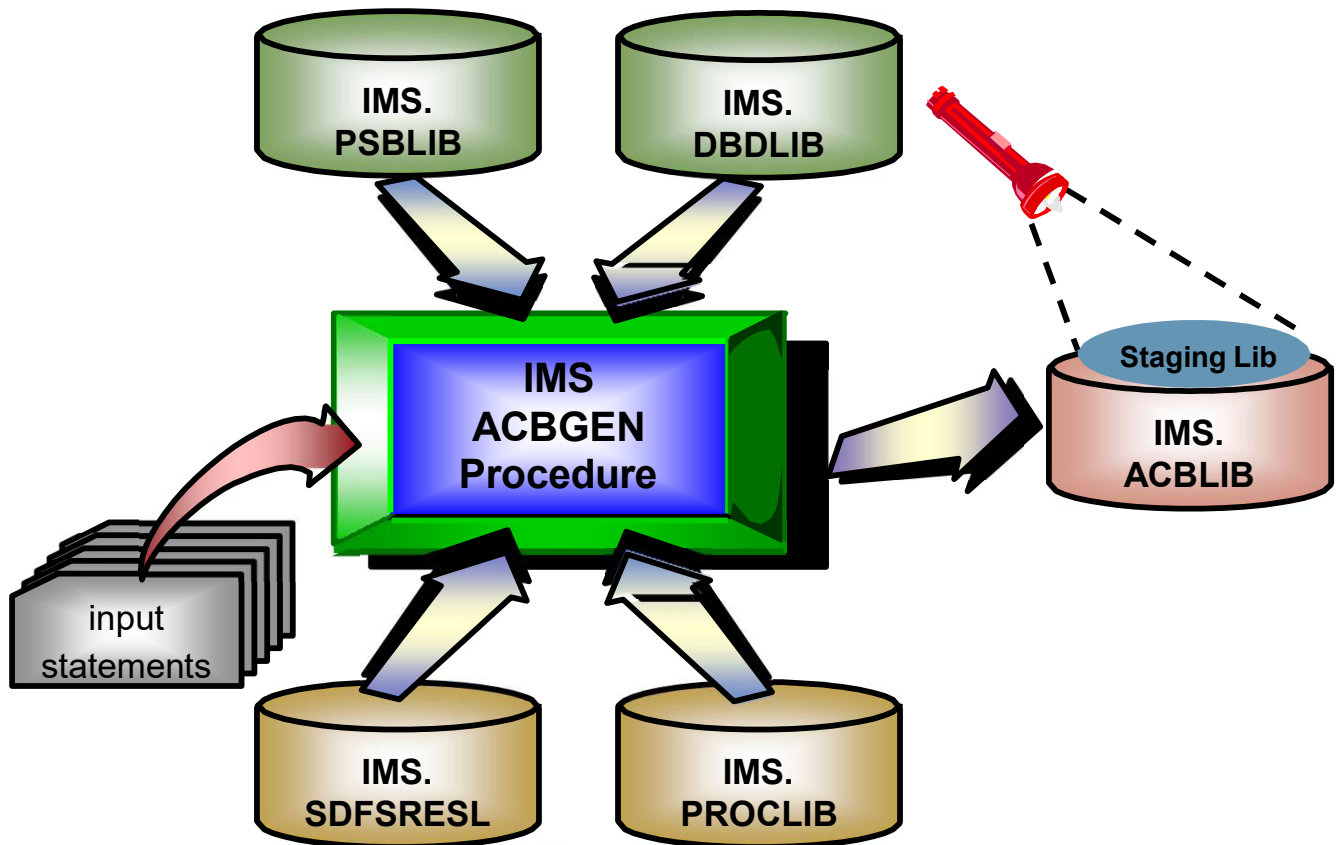
```
//DFSSAM11 EXEC PROC=PSBGEN,MBR=DFSSAM01,SOUT='*'
XX      PROC MBR=TEMPNAME,SOUT='*',RGN=256K,SYS2=
XXC     EXEC PGM=ASMA90,REGION=&RGN,PARM='OBJECT,NODECK'
IEFC653I SUBSTITUTION JCL - PGM=ASMA90,REGION=256K,PARM='OBJECT,NODECK'
XXSYSLIB DD DSN=IMS.&SYS2.MACLIB,DISP=SHR
IEFC653I SUBSTITUTION JCL - DSN=IMS.SDFSIMAC,DISP=SHR
XXSYSLIN DD UNIT=SYSDA,DISP=(,PASS),SPACE=(80,(100,100),RLSE),
XX      DCB=(BLKSIZE=80,RECFM=F,LRECL=80)
XXSYSPRINT DD SYSOUT=&SOUT,DCB=BLKSIZE=1089,SPACE=(121,(300,300),RLSE,,ROUND)
IEFC653I SUBSTITUTION JCL - SYSOUT=*,DCB=BLKSIZE=1089,SPACE=(121,(300,300),RLSE,,RO
XXSYSUT1 DD UNIT=SYSDA,DISP=(,DELETE),SPACE=(CYL,(10,5))
//C.SYSIN DD DISP=SHR,
//      DSN=IMS.SDFSISRC(DFSSAM11)
XXL     EXEC PGM=IEWL,PARM='XREF,LIST',
XX      COND=(0,LT,C),REGION=120K
XXSYSLIN DD DSN=*.C.SYSLIN,DISP=(OLD,DELETE)
XXSYSPRINT DD SYSOUT=&SOUT,DCB=BLKSIZE=1089,SPACE=(121,(90,90),RLSE)
IEFC653I SUBSTITUTION JCL - SYSOUT=*,DCB=BLKSIZE=1089,SPACE=(121,(90,90),RLSE)
XXSYSLMOD DD DISP=SHR,DSN=IMS.&SYS2.PSBLIB(&MBR)
IEFC653I SUBSTITUTION JCL - DISP=SHR,DSN=IMS.PSBLIB(DFSSAM01)
XXSYSUT1 DD UNIT=(SYSDA,SEP=(SYSLMOD,SYSLIN)),
XX      SPACE=(1024,(100,10),RLSE),DISP=(,DELETE)
//*
```



IV3E202J: PSBGEN input

```
10 *****
11 *
12 *      CHANGES:
13 *      APAR...  ID  PREREQ.  DATE....  DESCRIPTION.....
14 *      KNQ0115  01          11/17/91  ADD PCB LABEL FOR AIB INTERFACE
15 *
16 DBPCB01  PCB      TYPE=DB, DBDNAME=DI21PART, PROCOPT=L, KEYLEN=43
17          SENSEG  NAME=PARTROOT, PARENT=0
18          SENSEG  NAME=STANINFO, PARENT=PARTROOT
19          SENSEG  NAME=STOKSTAT, PARENT=PARTROOT
20          SENSEG  NAME=CYCCOUNT, PARENT=STOKSTAT
21          SENSEG  NAME=BACKORDR, PARENT=STOKSTAT
22          PSBGEN  LANG=COBOL, PSBNAME=DFSSAM01
*      IMS PROGRAM SPECIFICATION BLOCK GENERATION FOR DFSSAM01
```

IV3E203J: ACBGEN overview



IV3E203J: ACBGEN JCL

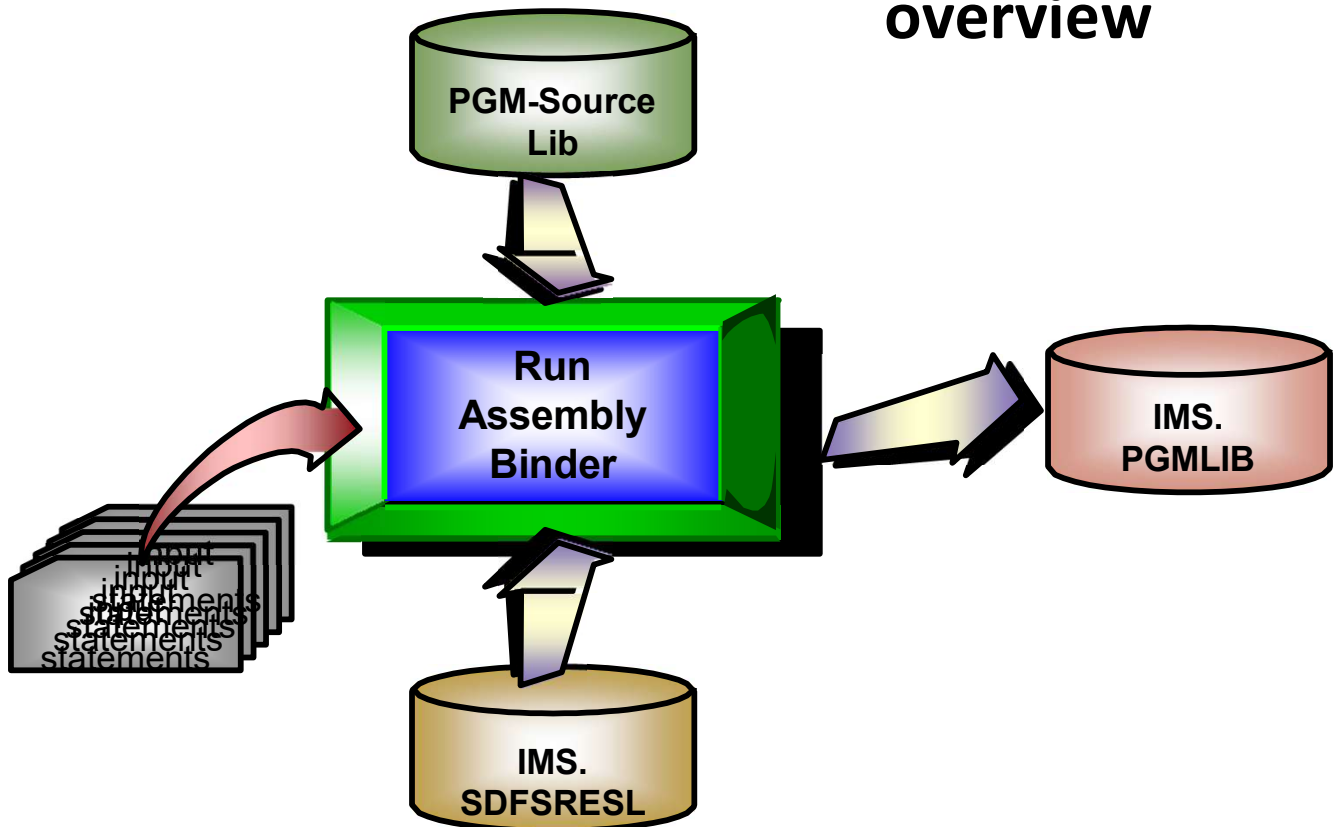
```
2 //ACBGEN EXEC PROC=ACBGEN,SOUT='*',COMP='POSTCOMP'
3 XX PROC SOUT=A,COMP=,RGN=256K,SYS2=
4 XXG EXEC PGM=DFSRR00,PARM='UPB,&COMP',
   IEF653I SUBSTITUTION JCL - PGM=DFSRR00,PARM='UPB,POSTCOMP',REG=&RGN
   IEF653I SUBSTITUTION JCL - REGION=256K
5 XXSYSRINT DD SYSOUT=&SOUT
   IEF653I SUBSTITUTION JCL - SYSOUT=*
6 XXSTEPLIB DD DSN=IMS.&SYS2.SDFSRESL,DISP=SHR IEF653I
   SUBSTITUTION JCL - DSN=IMS.SDFSRESL,DISP=SHR
7 XXDFSRESLB DD DSN=IMS.&SYS2.SDFSRESL,DISP=SHR IEF653I
   SUBSTITUTION JCL - DSN=IMS.SDFSRESL,DISP=SHR
8 XXIMS DD DSN=IMS.&SYS2.PSBLIB,DISP=SHR
   IEF653I SUBSTITUTION JCL - DSN=IMS.PSBLIB,DISP=SHR
9 XX DD DSN=IMS.&SYS2.DBDLIB,DISP=SHR
   IEF653I SUBSTITUTION JCL - DSN=IMS.DBDLIB,DISP=SHR
10 XXIMSACB DD DSN=IMS.&SYS2.ACBLIB,DISP=OLD
   IEF653I SUBSTITUTION JCL - DSN=IMS.ACBLIB,DISP=OLD
11 XXSYSUT3 DD UNIT=SYSDA,SPACE=(80,(100,100))
12 XXSYSUT4 DD UNIT=SYSDA,SPACE=(256,(100,100)),DCB=KEYLEN=8
13 XXCOMPCTL DD DISP=SHR,
   XX DSN=IMS.&SYS2.PROCLIB(DFSACBCP)
   IEF653I SUBSTITUTION JCL - DSN=IMS.PROCLIB(DFSACBCP)
14 //G.SYSIN DD *,DCB=BLKSIZE=80
```

BUILD PSB=ALL



KIESSLICH CONSULTING

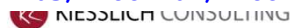
IV3E206J: Assembly/Bind Applications overview



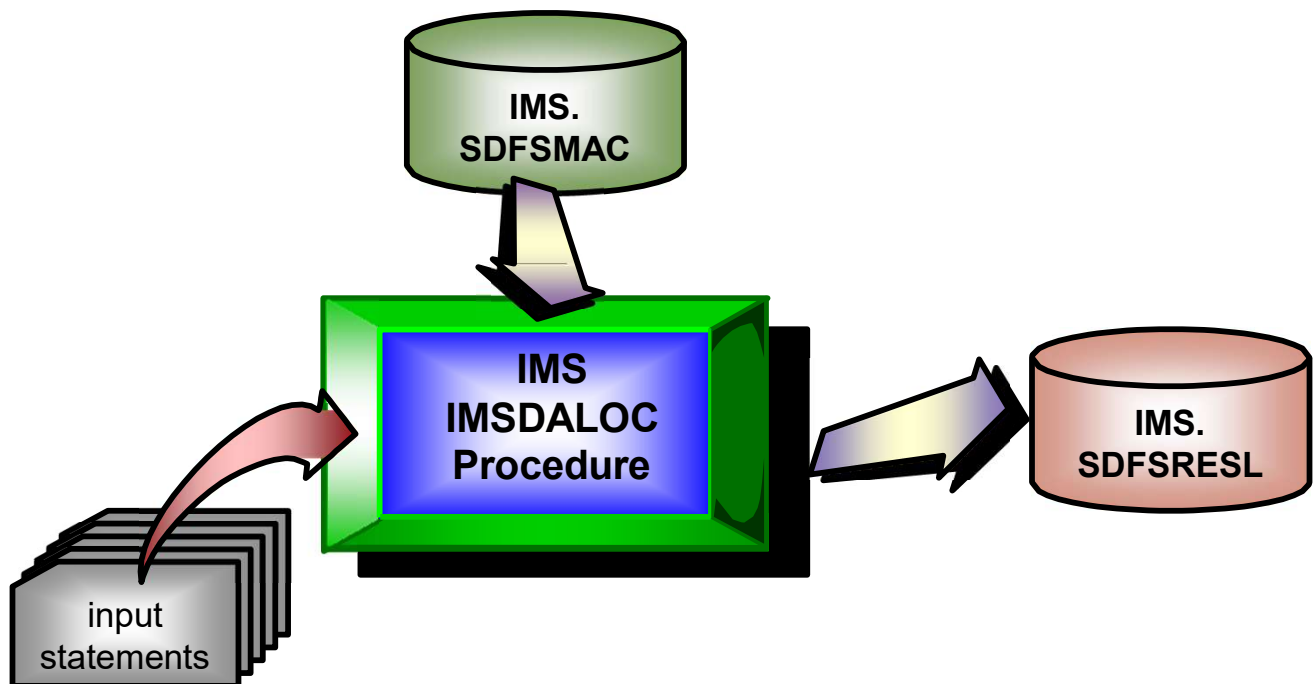
IV3E206J: Assembly/Bind Applications JCL

```
//C          EXEC PGM=ASMA90,PARM='OBJECT,NODECK'
//SYSPRINT DD SYSOUT=*
//SYSLIB DD DISP=SHR,DSN=IMS.ADFSMAAC
//          DD DISP=SHR,DSN=SYS1.MACLIB
//SYSLIN DD UNIT=3390,DISP=(,PASS,DELETE),SPACE=(CYL,(1,1)),
//          DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//SYSUT1 DD UNIT=3390,DISP=(,DELETE,DELETE),SPACE=(CYL,(1,1))
//SYSUT2 DD UNIT=3390,DISP=(,DELETE,DELETE),SPACE=(CYL,(1,1))
//SYSUT3 DD UNIT=(3390,SEP=(SYSLIB,SYSUT1,SYSUT2)),
//          DISP=(,DELETE,DELETE),SPACE=(CYL,(1,1))
//SYSIN DD DISP=SHR,DSN=IMS.SDFSISRC(&SRC)
//L          EXEC PGM=IEWL,
//          PARM='XREF,LIST',COND=(0,LT,C)
//SYSPRINT DD SYSOUT=*
//SYSLIN DD DISP=(OLD,DELETE,DELETE),
//          DSN=*.C.SYSLIN,VOL=REF=*.C.SYSLIN
//          DD DDNAME=SYSIN
//RESLIB DD DISP=SHR,DSN=IMS.SDFSRESL
//SYSLMOD DD DISP=SHR,DSN=IMS.PGMLIB
//SYSUT1 DD UNIT=(3390,SEP=(SYSLMOD,SYSLIN)),
//          DISP=(,DELETE,DELETE),SPACE=(CYL,(1,1))
//          PEND

//*****
***
//* LINK-EDIT THE IMS SAMPLE APPLICATION REXX PROGRAMS -
//* DFSSAM01/DFSSAM02/DFSSAM03/DFSSAM04/DFSSAM05/DFSSAM06/DFSSAM07
```



IV3E301J: Create Dynamic Allocation Members overview



IV3E301J: Create Dynamic Allocation

Members input statements (1 of 2)

```
//STEP01 EXEC PROC=IMSDALOC, SOUT='*'  
//ASSEM.SYSIN DD *
```

```
4 DFSMDA TYPE=INITIAL
```

```
10 DFSMDA TYPE=DATABASE, DBNAME=DI21PART
```

```
11 DFSMDA TYPE=DATASET, DDNAME=DI21PART,  
          DSNAME=IMS.DI21PART,  
          DISP=SHR
```

```
11 DFSMDA TYPE=DATASET, DDNAME=DI21PARO,  
          DSNAME=IMS.DI21PARO,  
          DISP=SHR
```

```
46 DFSMDA TYPE=DFSDCMON, DDNAME=IMSMON,  
          DSNAME=IMS.MONITOR.DATASET, DISP=SHR
```

```
54 DFSMDA TYPE=RECON, DDNAME=RECON1,  
          DSNAME=IMS.RECON1
```

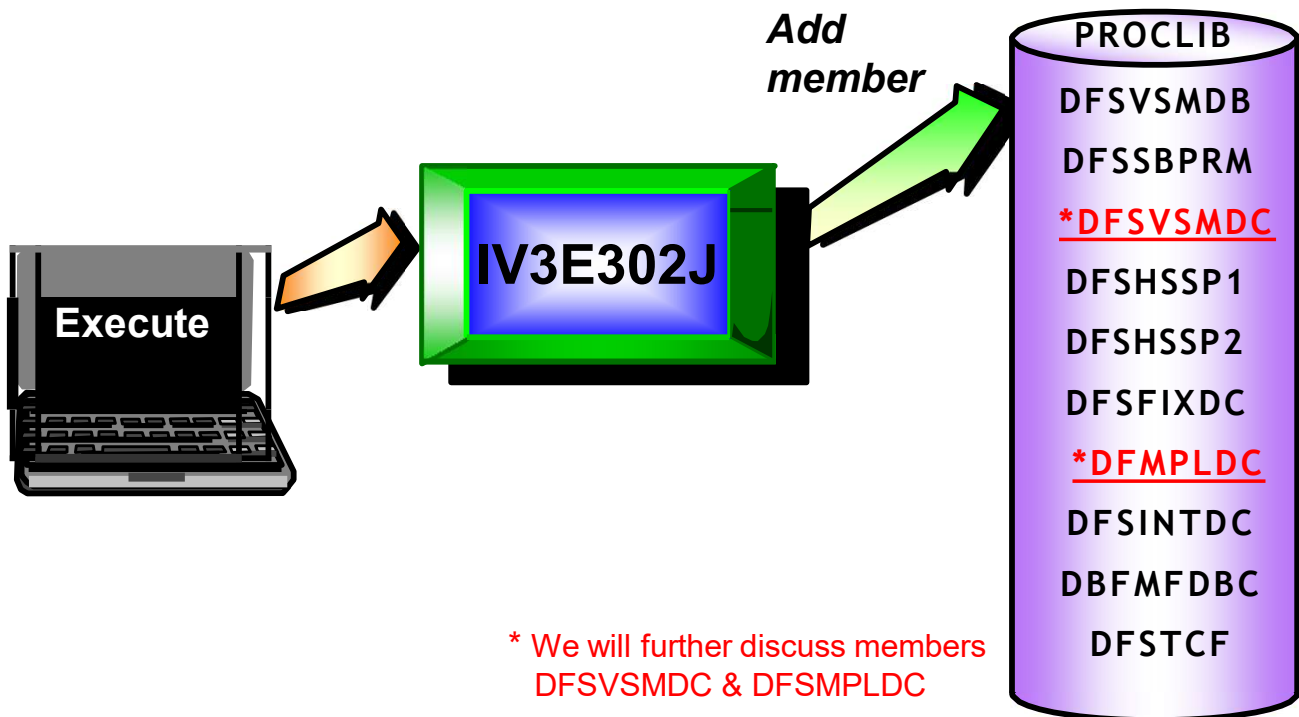


KIESSLICH CONSULTING

IV3E301J: Create Dynamic Allocation Members input statements (2 of 2)

```
58 * OLDS
59 *
60 DFSMDA TYPE=OLDS,DDNAME=DFSOLP00, X
        DSNAME=IMS.OLP00
        .
        .
        .
67 * WADS
68 *
69 DFSMDA TYPE=WADS,DDNAME=DFSWADS0, X
        DSNAME=IMS.WADS0
        .
        .
        .
71 *
72 * IMSLOGR (SLDS)
73 *
74 DFSMDA TYPE=SLDS,UNIT=TAPE,DDNAME=IMSLOGR SLDS 'UNIT' TYPE
75 *
76 * END
77 *
78 DFSMDA TYPE=FINAL <== MUST BE LAST DYNALLOC INPUT STATEMENT
```

IV3E302J: Add Control Statements to IMS.PROCLIB overview



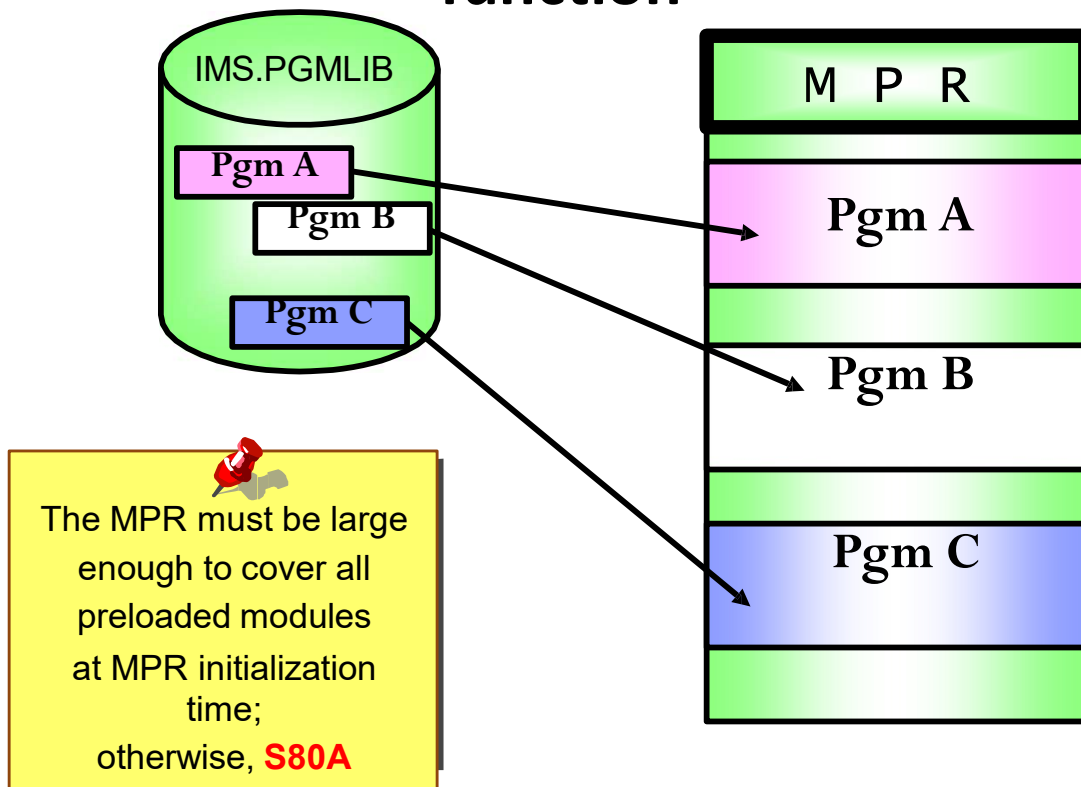
IV3E302J: Add DFSVSMDC Control Statements to IMS.PROCLIB

```
//TCFUPDT  PROC MBR=TEMPNAME
//T        EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT2   DD DISP=SHR,
//          DSN=IMS.TCFSLIB(&MBR)
//SYSIN    DD DUMMY
//          PEND
//*****
//*
//DFSVSMDC EXEC PROC=PROCUPDT, MBR=DFSVSMDC
//P.SYSUT1 DD *
VSRBF=4096,5
VSRBF=2048,5
VSRBF=1024,5
VSRBF=512,5
IOBF=(2048,5,Y,Y)
SBONLINE,MAXSB=10
OPTIONS,BGWRT=YES,INSERT=SKP,DUMP=YES,DUMPIO=YES
OPTIONS,VSAMFIX=(BFR,IOB),VSAMPLS=LOCL
OPTIONS,DL/I=OUT,LOCK=OUT,DISP=OUT,SCHD=OUT,DLOG=OUT,LATC=ON,SUBS=ON
OPTIONS,STRG=ON
OLDSDEF OLDS=(00,01,02,03,04,05,99),BUFNO=005,MODE=DUAL
WADSDEF WADS=(0,1,8,9)
```



KIESSLICH CONSULTING

DFSMPLxx: The IMS DB/DC Preload function



LLA / VLF

- LLA/VLF is similar to the Obsolete *Virtual Fetch Option* and provides the following functions:
 - Maintains in-storage copies of production library directories
 - Stages selected modules (determined by LLA/VLF) to a VLF virtual storage data space
 - Can improve performance by keeping frequently used modules in virtual storage
- LLA/VLF has advanced storage and program management capability:
 - LLA/VLF uses its own LRU algorithm to determine which modules to keep in the data space
 - In this way, LLA/VLF usually makes better use of the virtual storage available
 - LLA/VLF *refresh* is by means of a command
- Setting up LLA/VLF for IMS DB/DC is done by z/OS Systems Programmers not IMS Systems Programmers:
 - You need to work with them to identify which IMS libraries would benefit from this facility



KIESSLICH CONSULTING

KC110 unit 11e page 22

Library LookAside/Virtual Lookaside Facility

IV3E305J: Define EXEC PARM defaults (DFSPBxxx member)

- Alternative to IMS procedure startup parameters
 - No parameter string length limitation
 - Individual parameters can still be overridden in IMS startup if required
- Multiple versions can be defined
 - The *PB* Suffix can be specified on IMS procedure

```
//IMSDBDC      EXEC IMSDBDC,PARM1='RGSUF=xxx'
```

- IMS parameter override priority:
 - Command parameter (PARM1/PARM2='.....') on z/OS Start command
 - Parameter value in IMS procedure
 - DFSPBxxx parameter member value
 - IMS Gen System Definition values

IV3E305J: Define EXEC PARM DFSPBxxx

member (1 of 4)

```
//DFSPBIMS EXEC PROC=PROCUPDT, MBR=DFSPBIMS
//P.SYSUT1 DD *
  RES=Y,
  FRE=00030,
  QBUF=0005,
  PST=5,
  SAV=005,
  EXVR=Y,
  SRCH=0,
  FBP=00048,
  PSB=0048,
  DMB=048,
  CIOP=,
  WKAP=048,
  PSBW=024,
  SPAP=,
  DBWP=024,
  SUF=I,
  FIX=DC,
  PRLD=DC,
  VSPEC=DC,
  SOD=,
  VAUT=1,
  BSIZ=02048,
  OTHR=005,
  DBFX=00010,
  DBBF=00050,
  MSDB=C,
```


IV3E305J: Define EXEC PARM DFSPBxxx

Member (2 of 4)

```
.  
.  
FMTO=D,  
AUTO=N,  
TRN=N,  
SGN=N,  
RCF=N,  
IMSID=IVP1,  
ISIS=2,  
LGNR=10,  
NLXB=,  
LSO=S,  
SSM=,  
WADS=D,  
ARC=01,  
UHASH=,  
QTU=075,  
QTL=050,  
DBRCNM=IVP61RC1,  
DLINM=IVP61DL1,  
CSAPSB=12,  
DLIPSB=40,  
HSBID=,  
HSBMBR=,  
PRDR=IVP61RD1,  
RECA=5,  
EPCB=0012,  
.  
.
```

IV3E305J: Define EXEC PARM DFSPBxxx

Member (3 of 4)

```
CRC=,  
HIOP=,  
FPWP=,  
EMHB=,  
SPM=,  
EMHL=256,  
ASOT=60,  
ALOT=09,  
DLQT=60,  
ETO=Y,  
RECASZ=4095,  
CHTS=1000,  
LHTS=256,  
NHTS=256,  
UHTS=256,  
DSCT=I,  
FESTIM=,  
APPC=,  
LUMP=,  
LUMC=,  
TRACK=,  
RSRMBR=,  
APPLID1=,  
APPLID2=,  
APPLID3=,  
.  
.
```

Are we sure these
parms are valid?

IV3E305J: Define EXEC PARM DFSPBxxx

Member (4 of 4)

USERVAR=,
IRLM=Y,
IRLMNM=IRLM,
PIMAX=,
PIINCR=,
AOIP=,
AOIS=,
GRNAME=,
OTMA=,
MAXPST=

IMS Syntax Checker

- Syntax Checker is a IMS ISPF application added in IMS V8
 - Its use has been optional since IMS
- Its primary functions are to:
 - Define, verify, and validate PROCLIB member DFSPBxxx parameters and value specifications prior to attempting an IMS start
 - For example, Syntax Checker ...
 - Reads and displays a PROCLIB member's parameters and values
 - Verifies that parameters and values are valid
 - Allows user to add/delete parameters and/or modify values
 - Allows user to save to same or different Proclib member
 - Identify new and obsolete parameters
 - Useful when migrating to new version of IMS
 - Provide detailed online help text at the parameter level
- Documented in IMS manual: *IMS System Definition Guide*

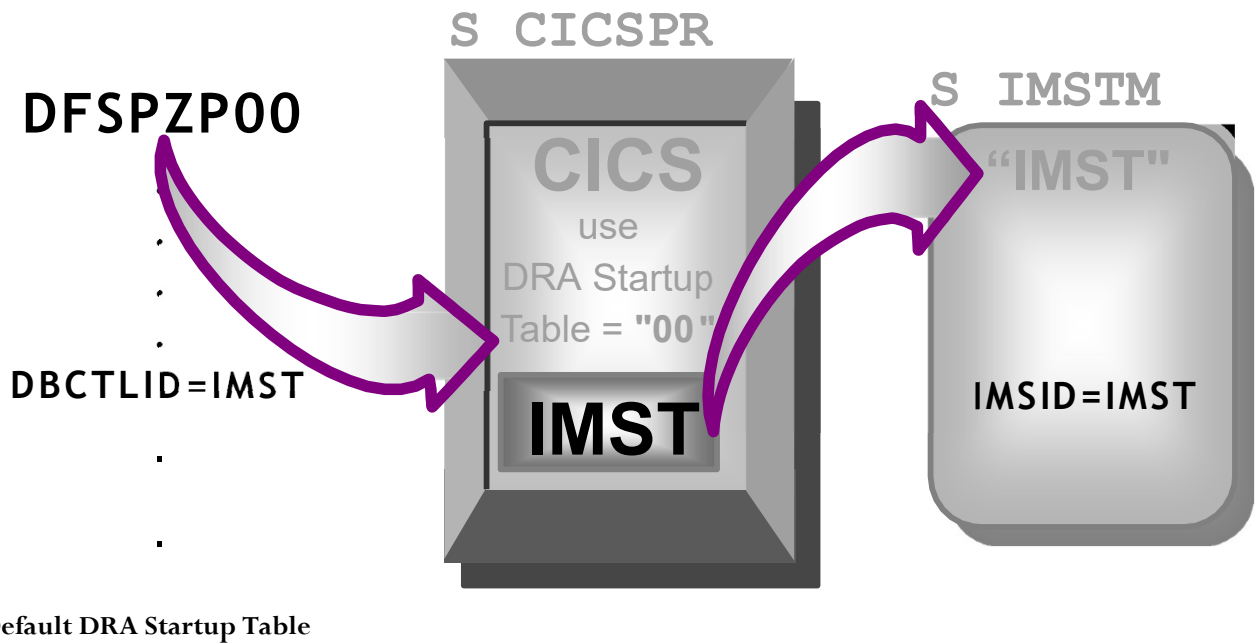
Not valid value

Not in release

Keyword help

DEMO !!!

IV3E308J: Define Database Resource Adapter Start-up Table



IV3E308J: Define DRA Start-up Table JCL (1 of 5)

```
//ASMDRA    PROC MBR=TEMPNAME
//*
//ASM       EXEC PGM=ASMA90,PARM='OBJECT,NODECK'
//SYSPRINT DD SYSOUT=*
//SYSLIB    DD DISP=SHR,DSN=IMS.ADFSMAC
//          DD DISP=SHR,DSN=SYS1.MACLIB
//SYSLIN    DD UNIT=3390,DISP=(,PASS,DELETE),SPACE=(CYL,(1,1)),
//          DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//SYSUT1    DD UNIT=3390,DISP=(,DELETE,DELETE),SPACE=(CYL,(1,1))
//SYSUT2    DD UNIT=3390,DISP=(,DELETE,DELETE),SPACE=(CYL,(1,1))
//SYSUT3    DD UNIT=(3390,SEP=(SYSLIB,SYSUT1,SYSUT2)),
//          DISP=(,DELETE,DELETE),SPACE=(CYL,(1,1))
//*
//LKED1     EXEC PGM=IEWL,COND=(0,LT,ASM),
//          PARM='NCAL,LET,LIST,XREF'
//SYSPRINT DD SYSOUT=*
//SYSLIN    DD DISP=(OLD,PASS,DELETE),
//          DSN=*.ASM.SYSLIN,VOL=REF=*.ASM.SYSLIN
//SYSLMOD    DD DISP=SHR,
//          DSN=IMS.SDFSRESL(&MBR)
//SYSUT1    DD UNIT=(3390,SEP=(SYSLMOD,SYSLIN)),
//          DISP=(,DELETE,DELETE),SPACE=(CYL,(1,1))
```

IV3E308J: Define DRA Start-up Table JCL (2 of 5)

```
//*****
//*      DRA WILL BE REPLACED BY SYSGEN !!!!
//*      ----> LINK/EDIT INTO IMS.ADFSLOAD
//*****
//LKED2    EXEC PGM=IEWL,COND=(0,LT,ASM) ,
//          PARM='NCAL,LET,LIST,XREF'
//SYSPRINT DD SYSOUT=*
//SYSLIN   DD DISP=(OLD,PASS,DELETE) , DSN=*.ASM.SYSLIN,VOL=REF=*.ASM.SYSLIN
//SYSLMOD  DD DISP=SHR,
//          DSN=IMS.ADFSLOAD (&MBR)
//SYSUT1   DD UNIT=(3390,SEP=(SYSLMOD,SYSLIN)) ,
//          DISP=(,DELETE,DELETE) , SPACE=(CYL,(1,1))
//*****
//*      LINK/EDIT INTO CICS.SIT.LOADLIB1
//*****
//LKED3    EXEC PGM=IEWL,COND=(0,LT,ASM) ,
//          PARM='NCAL,LET,LIST,XREF'
//SYSPRINT DD SYSOUT=*
//SYSLIN   DD DISP=(OLD,DELETE,DELETE) ,
//          DSN=*.ASM.SYSLIN,VOL=REF=*.ASM.SYSLIN
//SYSLMOD  DD DISP=SHR,
//          DSN=CICS.SIT.LOADLIB1 (&MBR)
//SYSUT1   DD UNIT=(3390,SEP=(SYSLMOD,SYSLIN)) ,
//          DISP=(,DELETE,DELETE) , SPACE=(CYL,(1,1))
```

IV3E308J: Define DRA Start-up Table JCL (3 of 5)

```
//*****
//*
//*   COPY DRA STARTUP ROUTER PGM  DFSPRC0
//*   INTO CICS.SIT.LOADLIB1
//*
//*****

//DFSPRC0 EXEC PGM=IEBCOPY
//SYSPRINT DD  SYSOUT=*
//SYSUT1 DD   DSN=IMS.SDFSRESL,DISP=SHR
//SYSUT2 DD   DSN=CICS.SIT.LOADLIB1,DISP=SHR
//SYSUT3 DD UNIT=VIO,DISP=(NEW,PASS)
//SYSUT4 DD UNIT=VIO,DISP=(NEW,PASS)
//SYSIN  DD DISP=SHR,DSN=IMS.INSTALIB(IEBCOPY)
//
//      PEND
//*****
//*
//* THE FOLLOWING STEP ASSEMBLES AND LINK-EDITS INTO IMS RESLIB
//* THE DEFAULT PARAMETERS USED BY THE IMS CONTROL REGION - IVP3
//*
//DFSPZPIV EXEC PROC=ASMDRA,MBR=DFSPZP00
//ASM.SYSIN DD *
```


IV3E308J: Define DRA Start-up Table JCL (4 of 5)

```
*
*      FUNCLV=  DEFAULT (1). ADAPTER FUNCTIONAL LEVEL.      *
*      DDNAME=  1 TO 8 CHARACTER DD NAME TO BE USED WITH    *
*                DYNAMIC ALLOCATION OF THE DBCTL RESLIB.      *
*                DEFAULT (CCTLDD).                            *
*      DSNAME=  1 TO 44 CHARACTER DATASET NAME OF THE        *
*                DBCTL RESLIB.                                *
*                DEFAULT (IMSESA.RESLIB).                    *
*      DBCTLID=XXXX-NAME OF THE DBCTL REGION                 *
*                DEFAULT = SYS1                              *
*      USERID=XXXXXXXX-NAME OF THE USER REGION              *
*      MINTHRD=XXX-MINIMUM NUMBER OF THREADS TO BE          *
*                AVAILABLE (MAXIMUM NUMBER IS 255)           *
*                DEFAULT = 1                                 *
*      MAXTHRD=XXX-MAXIMUM NUMBER OF THREADS TO BE          *
*                AVAILABLE (MAXIMUM NUMBER IS 255)           *
*                DEFAULT = 1                                 *
*      TIMER=XX-IDENTIFY TIMER VALUE IN SECONDS (DEFAULT 60) *
*      FPBUF=XXX-NUMBER OF FAST PATH BUFFERS TO BE ALLOCATED *
*                AND FIXED PER THREAD (DEFAULT 00)          *
*      FPBOF=XXX-NUMBER OF FAST PATH OVERFLOW BUFFERS TO BE *
*                ALLOCATED PER THREAD (DEFAULT 00)          *
*      SOD=X-OUTPUT CLASS TO BE USED FOR SNAP DUMP OF        *
*                ABNORMAL THREAD TERMINATIONS (DEFAULT A)    *
*      TIMEOUT=XXX-DRATERM TIMEOUT VALUE IN SECONDS (DEFLT 60) *
*      CNBA=XXXX -  TOTAL FP NBA BUFFERS FOR CCTL            *
*      AGN=XXXXXXXX-1 to 8 CHARACTER APPLICATION GROUP NAME  *
```

IV3E308J: Define DRA Start-up Table JCL (5 of 5)

```
EJECT
DFSPRP DSECT=NO, X
      FUNCLV=1, X CCTL FUNCTION LEVEL X
      DDNAME=CCTLDD, XXXXXXXX DDN FOR CCTL RESLIB DYNALOC X
      DSNAME=IMS.SDFSRESL, X
      DBCTLID=IMST, XXXX NAME OF DBCTL REGION IMSIDX
      USERID=CICSPR, XXXXXXXX NAME OF USER REGION CICSAPPLX
      MINTHRD=001, XXX MINIMUM THREADS X
      MAXTHRD=030, XXX MAXIMUM THREADS X
      TIMER=60, XX IDENTIFY TIMER VALUE - SECS X
      FPBUF=001, XXX FP FIXED BFRS PER THREAD X
      FPBOF=001, XXX FP OVFLW BFRS PER THREAD X
      SOD=V, XXX SNAP DUMP CLASS X
      TIMEOUT=060, XXX DRATERM TIMEOUT IN SECONDS X
      CNBA=001 XXX TOTAL FP NBA BFRS FOR CCTL X
*      AGN=IVP XXXXXXXX APPLICATION GROUP NAME X
      END
//BACKUP1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=IMS.INSTALIB,DISP=SHR
//SYSUT2 DD DSN=IMS.JCL.CNTL,DISP=SHR
//SYSUT3 DD UNIT=VIO,DISP=(NEW,PASS)
//SYSUT4 DD UNIT=VIO,DISP=(NEW,PASS)
//SYSIN DD *
C INDD=SYSUT1,OUTDD=SYSUT2
S M=( (IV2E304J,DRADC,R) )
```

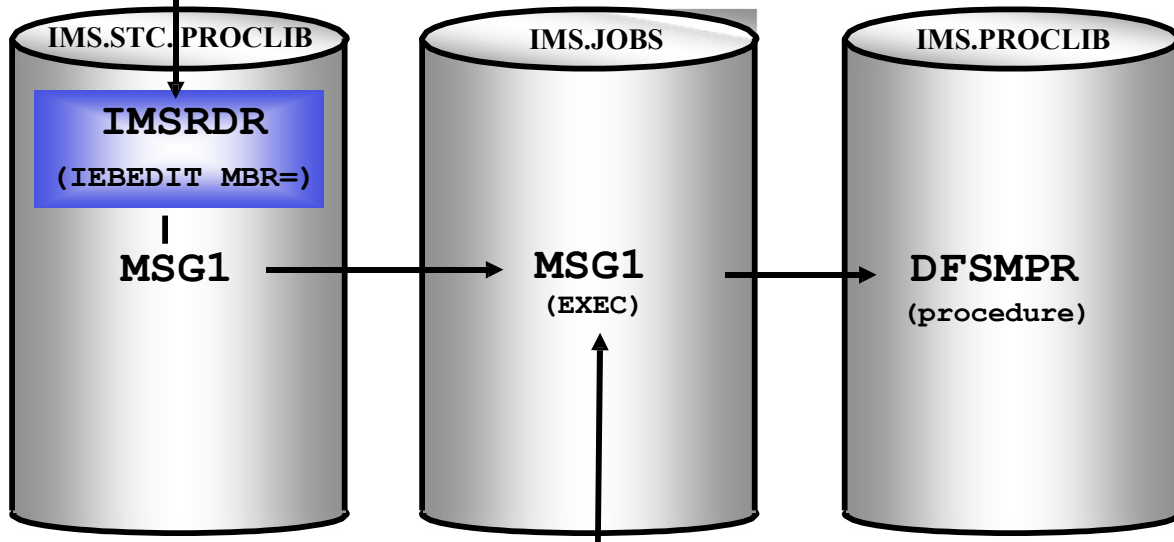


KISSLICH CONSULTING

IMS MPR Procedure (1 of 3)

/STA REG MSG1 (start of MPR "MSG1")

IMSRDR defined in DFSPBxxx member
Results in "S IMSRDR,MBR=MSG1"
command

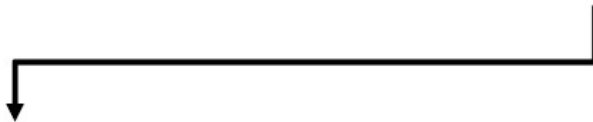


IV3E313J: copy

IMS MPR Procedure (2 of 3)

MEMBER IMSRDR IN IMS.STC.PROCLIB:

```
//          PROC   MBR=, CLASS=O
//IEFPROC EXEC PGM=IEBEDIT
//SYSPRINT DD DUMMY
//SYSUT1   DD DDNAME=IEFRDER
//SYSUT2   DD SYSOUT=(&CLASS,INTRDR),DCB=BLKSIZE=80
//SYSIN    DD DUMMY
//IEFRDER  DD DSN=IMS.JOBS(&MBR),DISP=SHR
```



MEMBER MSG1 IN IMS.JOBS:

```
//MSG1 JOB ,IMS,MSGLEVEL=1,PRTY=11,CLASS=A,MSGCLASS=A
//PROCTM JCLLIB ORDER=(IMS.PROCLIB)
//MSG1 EXEC DFSMPR,SOUT='*',RGN=2048K,SSM=,IMSID=IMSA
//* MPP-REG FOR SPROG SAMLE TRANSACTIONS
```

IMS MPR Procedure (3 of 3)

MEMBER **DFSMPR** IN IMS.PROCLIB:

```
//          PROC  SOUT=V,RGN=4M,SY2=,CL1=001,CL2=000,CL3=000,CL4=000,
//          OPT=N,OVLA=0,SPIE=0,VALCK=0,TLIM=00,PCB=000,PRLD=,
//          STIMER=,SOD=,DBLDL=,NBA=,OBA=,IMSID=,AGN=,VSFX=,
//          VFREE=,SSM=CTRL,PREINIT=,ALTID=,PWFI=N,
//          APARAM=,LOCKMAX=
// *
// REGION EXEC  PGM=DFSRR00,REGION=&RGN,TIME=1440,DPRTY=(12,0),
//          PARM=(MSG,&CL1&CL2&CL3&CL4,&OPT&OVLA&SPIE&VALCK&TLIM&PCB,
//          &PRLD,&STIMER,&SOD,&DBLDL,&NBA,&OBA,&IMSID,&AGN,&VSFX,
//          &VFREE,&SSM,&PREINIT,&ALTID,&PWFI,'&APARM',&LOCKMAX)
// *
/STEPLIB DD DSN=DB2P.RUNLIB.LOAD,DISP=SHR
/ DD DSN=IMS.&SYS2.PGMLIB,DISP=SHR
/ DD DSN=IMS.&SYS2.SDFSRESL,DISP=SHR
/ DD DSN=SYS1.COB2LIB,DISP=SHR
/PROCLIB DD DSN=IMS.&SYS2.PROCLIB,DISP=SHR
/DFSESL DD DSN=IMS.&SYS2.SDFSRESL,DISP=SHR
/ DD DSN=DB2P.DSNLOAD,DISP=SHR
/SYSUDUMP DD SYSOUT=&SOUT,
/ DCB=(LRECL=121,BLKSIZE=3129,RECFM=VBA),
/ SPACE=(125,(2500,100),RLSE,,ROUND)
```



KISSLICH CONSULTING

Start IMS MPP / Region review

Members to enable the Start of particular Message Regions

MEMBER **IMSRDR** IN IMS.STC.PROCLIB:

```
//          PROC   MBR=, CLASS=O
//IEFPROC EXEC PGM=IEBEDIT
//SYSPRINT DD   DUMMY
//SYSUT1   DD   DDNAME=IEFRDER
//SYSUT2   DD   SYSOUT=(&CLASS, INTRDR), DCB=BLKSIZE=80
//SYSIN    DD   DUMMY
//IEFRDER  DD   DSN=IMS.JOBS(&MBR), DISP=SHR
```

MEMBER **MSG1TM** IN IMS.JOBS:

```
//MSG1TM JOB   , IMS, MSGLEVEL=1, PRTY=11, CLASS=A, MSGCLASS=A
//PROCTM JCLLIB ORDER=(IMS.PROCLIB)
//MSG1TM EXEC  DFSMPR, SOUT='*', RGN=2048K, SSM=, IMSID=IMSA
//*      MPP-REG FOR   SPROG   SAMLE   TRANSACTIONS
```

MEMBER **DFSMPR** IN IMS.PROCLIB.



KIESSLICH CONSULTING

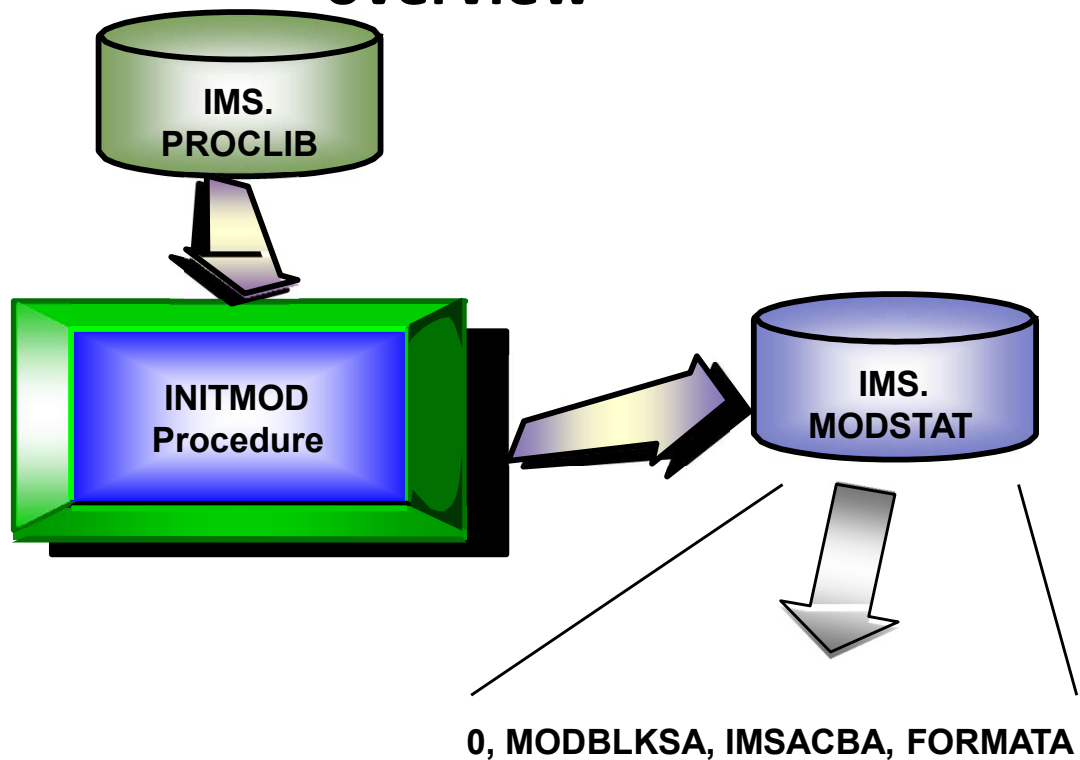
IV3E314J: Copy DBRC skeletons to IMS.PROCLIB

```
//ALLOC EXEC PGM=IEFBR14
//SYSPRINT DD SYSOUT=*
//* SKELETAL LIB INSTEAD OF PROCLIB
//DD1 DD DSN=IMS.JCLLIB,DISP=(,CATLG,DELETE),SPACE=(CYL,(1,1,200)),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),UNIT=3390,VOL=SER=IMS002
//*****
//COPYSKEL PROC MBR=TEMPNAME
//CS EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSIN DD DUMMY
//SYSUT2 DD DISP=SHR,DSN=IMS.JCLLIB(&MBR)
// PEND
//*****
//*
//ARCHJCL EXEC PROC=COPYSKEL,MBR=ARCHJCL
.
//CAJCL EXEC PROC=COPYSKEL,MBR=CAJCL
.
//ICJCL EXEC PROC=COPYSKEL,MBR=ICJCL
.
//LOGCLJCL EXEC PROC=COPYSKEL,MBR=LOGCLJCL
.
//OICJCL EXEC PROC=COPYSKEL,MBR=OICJCL
.
//RECOVJCL EXEC PROC=COPYSKEL,MBR=RECOVJCL
```



KISSLICH CONSULTING

IV3E317J: Initialize MODSTAT data set overview

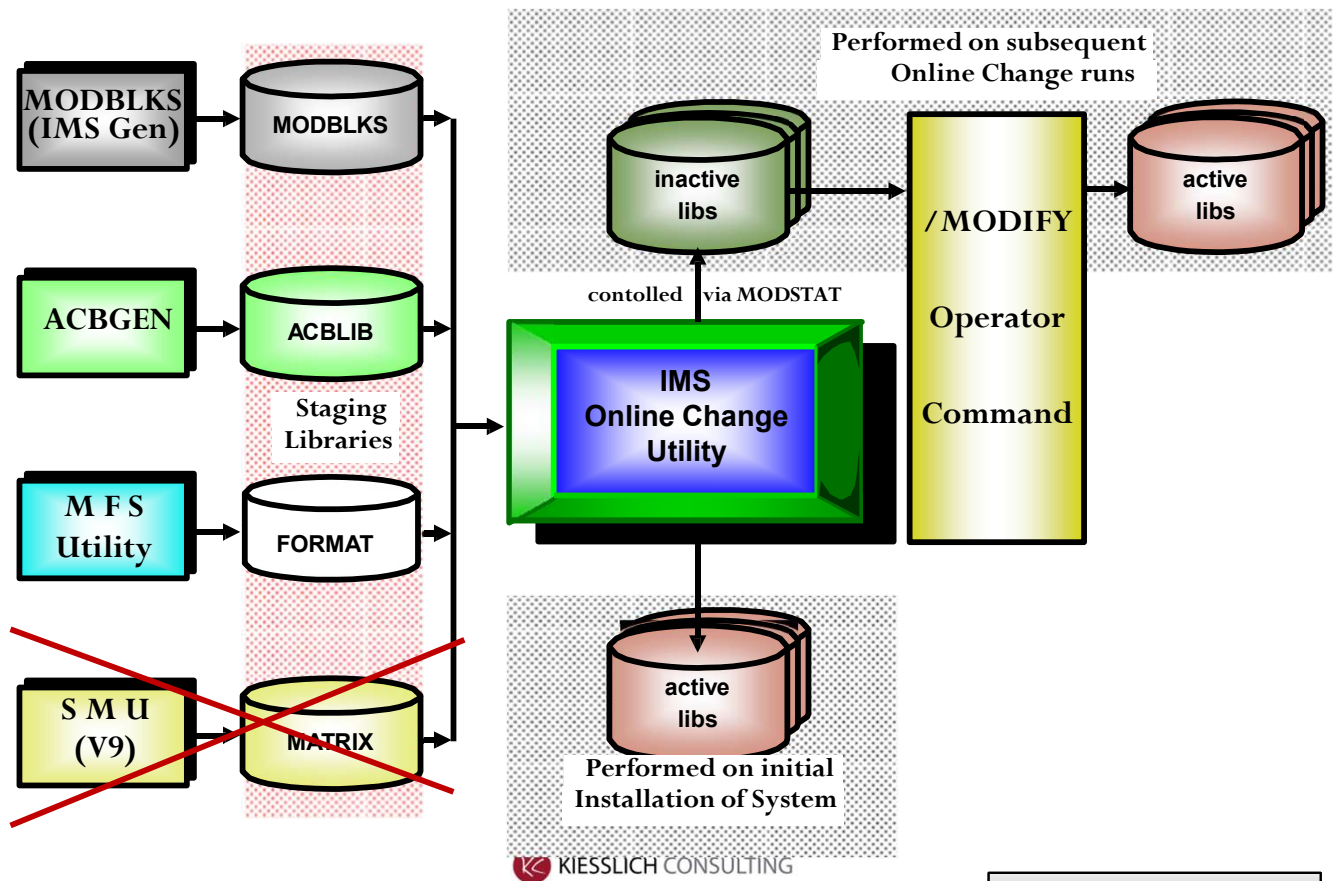


In an IMSPLEX you will probably migrate to the shared OLCSTAT instead of MODSTAT,
(except you build an OMPLEX with single / local / different IMS systems, each one keeps going with its local MODSTAT)

IV3E317J: Initialize MODSTAT JCL

```
.
.
.
2 //MODSTAT EXEC PROC=INITMOD,SOUT='*'
***
*****
***
***
***
***
***
***
***
***
***
***
***
***
***
***
***
***
***
3 XX PROC SYS=,SYS2=,SF=,SOUT=A
4 XXINIT1 EXEC PGM=IEBGENER
5 XXSYSPRINT DD SYSOUT=&SOUT
IEF653I SUBSTITUTION JCL - SYSOUT=*
6 XXSYSUT2 DD DSN=IMS.&SYS.MODSTAT&SF,DISP=OLD
IEF653I SUBSTITUTION JCL - DSN=IMS.MODSTAT,DISP=OLD
7 XXSYSIN DD DUMMY
8 XXSYSUT1 DD DISP=SHR,
XX DSN=IMS.&SYS2.PROCLIB(DFSMREC)
IEF653I SUBSTITUTION JCL - DSN=IMS . PROCLIB (DFSMREC)
.
.
```

IV3E318J: Copy Staging Libraries overview



IV3E318J: Copy Staging Libraries JCL (1 of 2)

```
***  
***          COPY MODBLKS TO MODBLKSA  
***  
2 //MODBLKS  EXEC PROC=OLCUTL,SOUT='*',TYPE=MODBLKS,IN=S,OUT=A  
***  
*** COPY MATRIX TO MATRIXA  
***  
3 XX          PROC  TYPE=,IN=,OUT=,SOUT=A,SYS=,SYS2=  
4 XXS         EXEC  PGM=DFSUOCU0,PARM=(&TYPE,&IN,&OUT)  
          IEF653I SUBSTITUTION JCL - PGM=DFSUOCU0,PARM=(MODBLKS,S,A)  
5 XXSTEPLIB   DD DSN=IMS.&SYS2.SDFSRESL,DISP=SHR IEF653I  
          SUBSTITUTION JCL - DSN=IMS.ADFSRESL,DISP=SHR  
6 XXMODBLKS   DD DSN=IMS.&SYS2.MODBLKS,DISP=SHR IEF653I  
          SUBSTITUTION JCL - DSN=IMS.MODBLKS,DISP=SHR  
7 XXMODBLKSA DD DSN=IMS.&SYS2.MODBLKSA,DISP=SHR IEF653I  
          SUBSTITUTION JCL - DSN=IMS.MODBLKSA,DISP=SHR  
8 XXMODBLKSB DD DSN=IMS.&SYS2.MODBLKSB,DISP=SHR IEF653I  
          SUBSTITUTION JCL - DSN=IMS.MODBLKSB,DISP=SHR  
9 XXIMSACB    DD DSN=IMS.&SYS2.ACBLIB,DISP=SHR  
          IEF653I SUBSTITUTION JCL - DSN=IMS.ACBLIB,DISP=SHR  
10 XXIMSACBA  DD DSN=IMS.&SYS2.ACBLIBA,DISP=SHR IEF653I  
          SUBSTITUTION JCL - DSN=IMS.ACBLIBA,DISP=SHR  
11 XXIMSACBB  DD DSN=IMS.&SYS2.ACBLIBB,DISP=SHR IEF653I  
          SUBSTITUTION JCL - DSN=IMS.ACBLIBB,DISP=SHR
```

IV3E318J: Copy Staging Libraries JCL (2 of 2)

```
***  
***          COPY ACBLIB TO ACBLIBA  
***  
2 //ACBLIB EXEC PROC=OLCUTL,SOUT='*',TYPE=ACB,IN=S,OUT=A  
***  
*** COPY MATRIX TO MATRIXA  
***  
3 XX          PROC TYPE=,IN=,OUT=,SOUT=A,SYS=,SYS2=  
4 XXS          EXEC PGM=DFSUOCU0,PARM=(&TYPE,&IN,&OUT)  
          IEF653I SUBSTITUTION JCL - PGM=DFSUOCU0,PARM=(MODBLKS,S,A)  
5 XXSTEPLIB DD DSN=IMS.&SYS2.SDFSRESL,DISP=SHR IEF653I  
          SUBSTITUTION JCL - DSN=IMS.SDFSRESL,DISP=SHR  
6 XXMODBLKS DD DSN=IMS.&SYS2.MODBLKS,DISP=SHR IEF653I  
          SUBSTITUTION JCL - DSN=IMS.MODBLKS,DISP=SHR  
7 XXMODBLKSA DD DSN=IMS.&SYS2.MODBLKSA,DISP=SHR IEF653I  
          SUBSTITUTION JCL - DSN=IMS.MODBLKSA,DISP=SHR  
8 XXMODBLKSB DD DSN=IMS.&SYS2.MODBLKSB,DISP=SHR IEF653I  
          SUBSTITUTION JCL - DSN=IMS.MODBLKSB,DISP=SHR  
9 XXIMSACB DD DSN=IMS.&SYS2.ACBLIB,DISP=SHR  
          IEF653I SUBSTITUTION JCL - DSN=IMS.ACBLIB,DISP=SHR  
10 XXIMSACBA DD DSN=IMS.&SYS2.ACBLIBA,DISP=SHR IEF653I  
          SUBSTITUTION JCL - DSN=IMS.ACBLIBA,DISP=SHR  
11 XXIMSACBB DD DSN=IMS.&SYS2.ACBLIBB,DISP=SHR IEF653I  
          SUBSTITUTION JCL - DSN=IMS.ACBLIBB,DISP=SHR
```

[illegible]

-
-
-

IMS DB/DC (DBT)

Execution

Help

Execution (LST Mode) - DBT

Row 104 to 120 of 215

Command ==>

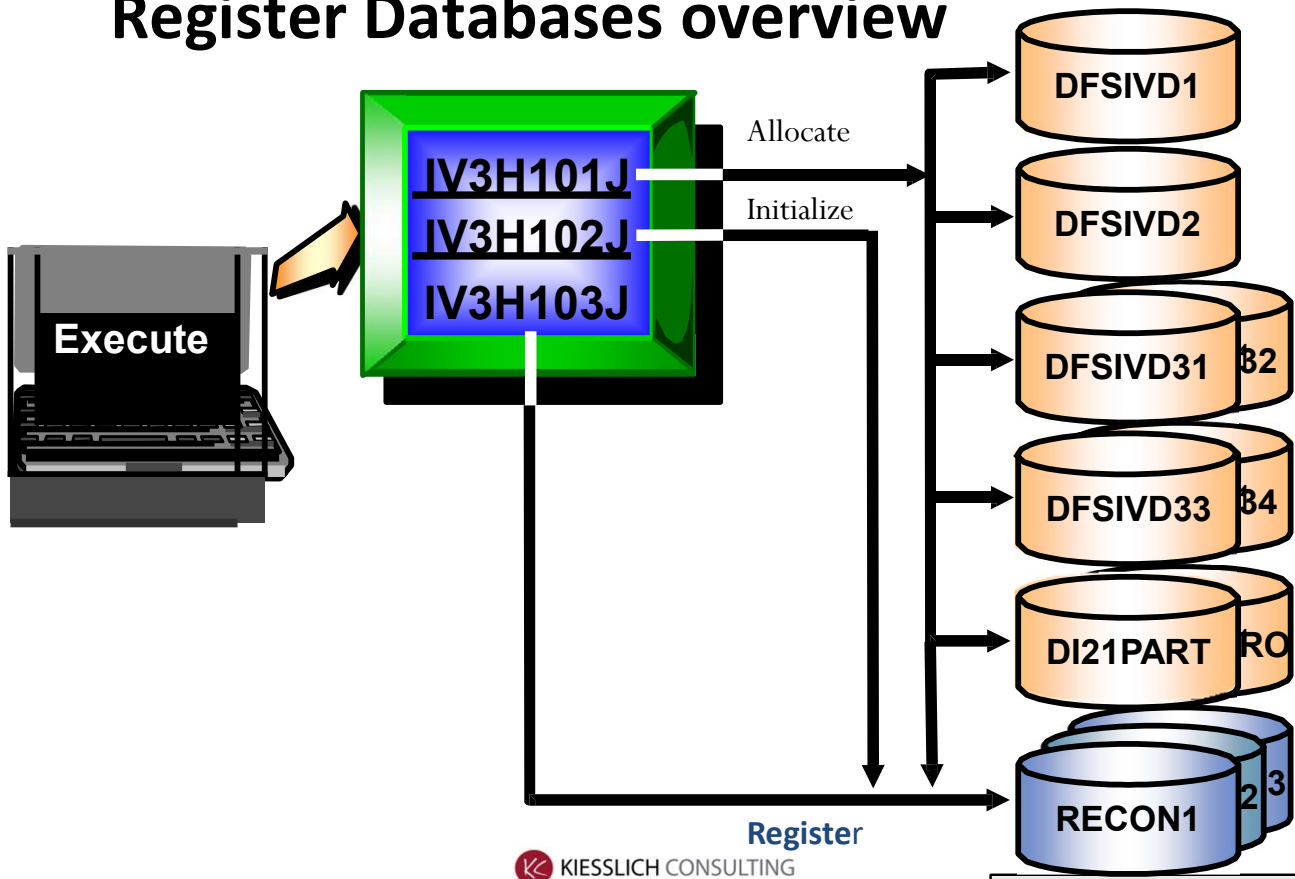
Scroll ==> CSR

Action Codes: Brm Doc Edm eNt eXe spR

JOB/Task	Step	Title.....
IV3H001T	H0	<u>NOTE - Step Introduction - IVP Execution - "DBT"</u>
IV3H101J	H1	JOB - Allocate Data Sets
IV3H102J	H1	JOB - Initialize RECON
IV3H103J	H1	JOB - Register Data Bases
IV3H104J	H1	JOB - Data Base Initial Load
IV3H105J	H1	JOB - Batch Image Copy
IV3H106T	H1	MVS - Clear MVS DUMPXX Data Sets
IV3H201J	H2	JOB - Start IRLM
IV3H203J	H2	JOB - Start DB/DC Region IVP1
IV3H204T	H2	IVP1 - Cold Start IMS
IV3H205T	H2	IVP1 - Review MTO Operator Commands
IV3H206T	H2	USER - Review User Operator Commands
IV3H207J	H2	JOB - FP BMP - DEDB Load
IV3H208J	H2	JOB - FF BMP - Online Image Copy
IV3H209J	H2	JOB - Concurrent Image Copy
IV3H210J	H2	JOB - FF BMP - HIDAM Update
IV3H211J	H2	JOB - FF BMP - HDAM Update

.
. .
. .

Allocate data sets / Initialize RECONs / Register Databases overview



IV3H101J: Allocate Data Sets JCL

```

      .
      .
      .
//*  SCRATCH AND REALLOCATE VSAM DATA BASE DATA SETS
//*          DATA BASE      DI21PART
//*
//SCRATCH  EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
//*
//ALLOCATE EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
//*
//*  SCRATCH AND REALLOCATE THE DBRC RECON DATA SETS
//*
//SCRATCH  EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
//*
//ALLOCATE EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
      .
      .
      .

```


IV3H101J: Allocate Database Data Sets

input (1 of 2)

```
DEFINE CLUSTER(
    NAME (IMS.DI21PART)
    INDEXED
    KEYS (17 06)
    FREESPACE (10 10)
    RECORDSIZE (582 582)
    SHAREOPTIONS (3 3)
    IMBED REPLICATE
    SPEED
    UNIQUE
    VOLUMES (SYSDA)
    CYLINDERS (3)
)
DATA (
    NAME (IMS.DI21PART.DATA)
    CONTROLINTERVALSIZE (4096)
)
INDEX (
    NAME (IMS.DI21PART.INDX)
    CONTROLINTERVALSIZE (4096)
)
```

IV3H101J: Allocate Database Data Sets

input (2 of 2)

```
DEFINE CLUSTER(
    NAME (IMS.DI21PARO)
    NONINDEXED
    FREESPACE (10 10)
    RECORDSIZE (582 582)
    SHAREOPTIONS (3 3)
    UNIQUE
    VOLUMES (SYSDA)
    CYLINDERS (03)
    CONTROLINTERVALSIZE (4096)
)
DATA (
    NAME (IMS.DI21PARO.DATA)
)
```

IV3H101J: Allocate RECON1 Data Sets input

```
•
•
• DEFINE CLUSTER(
• NAME(IMS.RECON1) FREESPACE(20 20) INDEXED
• KEYS(32 0)
• RECORDSIZE(4086 32600)
• SHAREOPTIONS(3 3) RECOVERY
• NOERASE NOREUSE UNORDERED UN
• VOLUMES(DBDC01)
• CYL(3 1)
• ) DATA(
• NAME(IMS.RECON1.DATA)
• ) INDEX(
• NAME(IMS.RECON1.INDX)
• )
•
•
•
```



SMS
is not
recommended
for RECONS

IV3H102J: Initialize RECON input to DBRC Utility

```

      .
      .
//D      EXEC PGM=DSPURX00
//STEPLIB DD DISP=SHR,DSN=IMS.SDFSRESL
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//IMS     DD DISP=SHR,DSN=IMS.DBDLIB
//JCLPDS  DD DISP=SHR,DSN=IMS.JCLLIB
//JCLOUT  DD SYSOUT=(A,INTRDR)
//D.SYSIN DD *
      INIT.RECON
              CATDS
              LISTDL
              NOFORCER
              CHECK44
              NONEW
              LOGRET('000 12.00')
              DASDUNIT(3390)
              TAPEUNIT(3400)
              LISTDL
      LIST.RECON
```

IV3H102J: Initialize RECON output (1 of 2)

```

      IMS VERSION 15 RELEASE 1 DATA BASE RECOVERY CONTROL                PAGE 0001
INIT.RECON          -
CATDS              -
NOFORCER           -
CHECK44            -
NONEW             -
LOGRET('000 12.00') -
DASDUNIT(3390)     -
TAPEUNIT(3400)     -
LISTDL
DSP0203I  COMMAND COMPLETED WITH CONDITION CODE 00
DSP0220I  COMMAND COMPLETION TIME 23.198 14:43:26.8
      IMS VERSION 15 RELEASE 1 DATA BASE RECOVERY CONTROL                PAGE 0002
LIST.RECON
23.198 14:43:21.697527          LISTING OF RECON                PAGE 0003
-----
RECON
RECOVERY CONTROL DATA SET, IMS V15R1
DMB#=0              INIT TOKEN=10128F1243244F
NOFORCER LOG DSN CHECK=CHECK44  STARTNEW=NO
TAPE UNIT=TAPE      DASD UNIT=3390  TRACEOFF  SSID=**NULL**
LIST DLOG=YES       CA/IC/LOG DATA SETS CATALOGED=YES
MINIMUM VERSION = 14.1  CROSS DBRC SERVICE LEVEL ID= 00001
REORG NUMBER VERIFICATION=NO
LOG RETENTION PERIOD=00.000 12:00:00.0
COMMAND AUTH=NONE  HLQ=**NULL**
ACCESS=SERIAL      LIST=STATIC
SIZALERT DSNUM=15   VOLNUM=16      PERCENT= 95
LOGALERT DSNUM=3    VOLNUM=16
```

IV3H102J: Initialize RECON output (2 of 2)

----- **OUTPUT CONTINUES** -----

TIME STAMP INFORMATION:

TIMEZIN = %SYS

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

IMSPLEX = ** NONE ** GROUP ID = ** NONE **

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

0DSP0127I NO LOG DATA SETS DEFINED IN RECON
0DSP0260I NO ONLINE LOG RECORDS FOUND IN RECON
0DSP0260I NO INTERIM RLDS/SLDS RECORDS FOUND IN RECON
0DSP0260I NO INT-ONLINE LOG RECORDS FOUND IN RECON
0DSP1004I NO GSG RECORDS FOUND IN RECON
0DSP0172I NO SUBSYSTEM RECORDS FOUND IN RECON
0DSP0177I NO BACKOUT RECORDS DEFINED IN RECON
0DSP0128I NO CHANGE ACCUMULATION GROUPS DEFINED IN RECON
0DSP0093I NO DBDSGRP RECORD FOUND IN RECON
0DSP0170I NO DATA BASE RECORDS FOUND IN RECON
0DSP0180I NUMBER OF RECORDS LISTED IS 1
1
IMS VERSION 15 RELEASE 1 DATA BASE RECOVERY CONTROL
0DSP0203I COMMAND COMPLETED WITH CONDITION CODE 00
0DSP0220I COMMAND COMPLETION TIME 23.198 14:43:28.000000

PAGE 0004



KIESSLICH CONSULTING

IV3H103J: Register Databases

```

      .
      .
//D      EXEC PGM=DSPURX00
//STEPLIB DD DISP=SHR,DSN=IMS.SDFSRESL
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//IMS      DD DISP=SHR,DSN=IMS.DBDLIB
//JCLPDS    DD DISP=SHR,DSN=IMS.JCLLIB
//JCLOUT    DD SYSOUT=(A,INTRDR)
//D.SYSIN    DD *

INIT.DB      DBD(DI21PART) SHARELVL(2) TYPEIMS

INIT.DBDS     DBD(DI21PART) DDN(DI21PART)          -
               DSN(IMS.DI21PART)                  -
               RECOVPD(5) GENMAX(5)

INIT.DBDS     DBD(DI21PART) DDN(DI21PARO)          -
               DSN(IMS.DI21PARO)                  -
               RECOVPD(5) GENMAX(5)
      .
      .

```



KIESSLICH CONSULTING

Upgrade DBRC RECON to new Version

```
//RECONUP JOB ,SYSPROG,MSGCLASS=V,CLASS=A,  
// MSGLEVEL=(1,1),NOTIFY=&SYSUID,REGION=4096K  
//*****  
//*  
//UPGRADE EXEC PGM=DSPURX00 New IMS Version  
//STEPLIB DD DISP=SHR,DSN=IMS.SDFSRESL (the „upgrade – to“ level) RESLIB library  
//*  
//SYSPRINT DD SYSOUT=*  
//RECON1 DD DISP=SHR,DSN=IMS.RECON1 existing RECON copy 1  
//RECON2 DD DISP=SHR,DSN=IMS.RECON2 existing RECON copy 2  
//SYSIN DD *  
CHANGE.RECON UPGRADE  
/*
```


IV3H104J: Database Initial Load JCL

```
.  
.  
//DI21PART EXEC PROC=DLIBATCH,SOUT='*',  
//      MBR=DFSSAM01,PSB=DFSSAM01,  
//      DBRC=N,IRLM=N  
//G.IEFRDER DD DUMMY,UNIT=3390  
//G.IEFRDER2 DD DUMMY,UNIT=3390  
//DI21PART DD DISP=OLD,DSN=IMS.DI21PART  
//DI21PARO DD DISP=OLD,DSN=IMS.DI21PARO  
//DFSVSAMP DD *  
//*  
//INPUT DD DISP=SHR,  
//      DSN=IMS.DFSISRC(MFDFSYSN)  
//*  
//DFSSTAT DD SYSOUT=*  
//*  
//SYSOUT DD SYSOUT=*
```

IV3H104J: Database Initial Load DFSSTAT (Statistics) output (1 of 2)

DFSSTAT Output:

```

.
.
.
*** PST ACCOUNTING STATISTICS ***
DB GU CALLS          0
DB GN CALLS          0
DB GNP CALLS         0
DB GHU CALLS         0
DB GHN CALLS         0
DB GHNP CALLS        0
DB ISRT CALLS        289
DB DLET CALLS         0
DB REPL CALLS         0
DB CALLS (TOTAL)     289
MSG GU CALLS         0
MSG GN CALLS         0
MSG ISRT CALLS        0
MSG PURGE CALLS      0
.
.
```

IV3H104J: Database Initial Load DFSSTAT (Statistics) Output (2 of 2)

*** VSAM BUFFER POOL STATISTICS ***

SUBPOOL BUFFER SIZE	4,096
TOTAL BUFFERS IN SUBPOOL	5
RETRIEVE BY RBA CALLS	12
RETRIEVE BY KEY CALLS	0
LOGICAL RECORDS INSERTED INTO ESDS	24
LOGICAL RECORDS INSERTED INTO KSDS	0
LOGICAL RECORDS ALTERED IN THIS SUBPOOL	10
TIMES BACKGROUND WRITE FUNCTION INVOKED	0
SYNCHRONIZATION CALLS RECEIVED	2
VSAM GET CALLS ISSUED	25
VSAM SCHBFR CALLS ISSUED	0
CONTROL INTERVAL REQUESTED ALREADY IN POOL	23
*CONTROL INTERVAL READ FROM EXTERNAL STORAGE	3
*VSAM WRITES INITIATED BY IMS	4
*VSAM WRITES TO MAKE SPACE IN THE POOL	0
PERM WRT ERROR BUFFS NOW IN THE SUBPOOL	0
LARGEST NUMB OF PERM ERR BUFFS EVEN IN THE SUBPOOL	0
*TOTAL I/O OPERATIONS	7