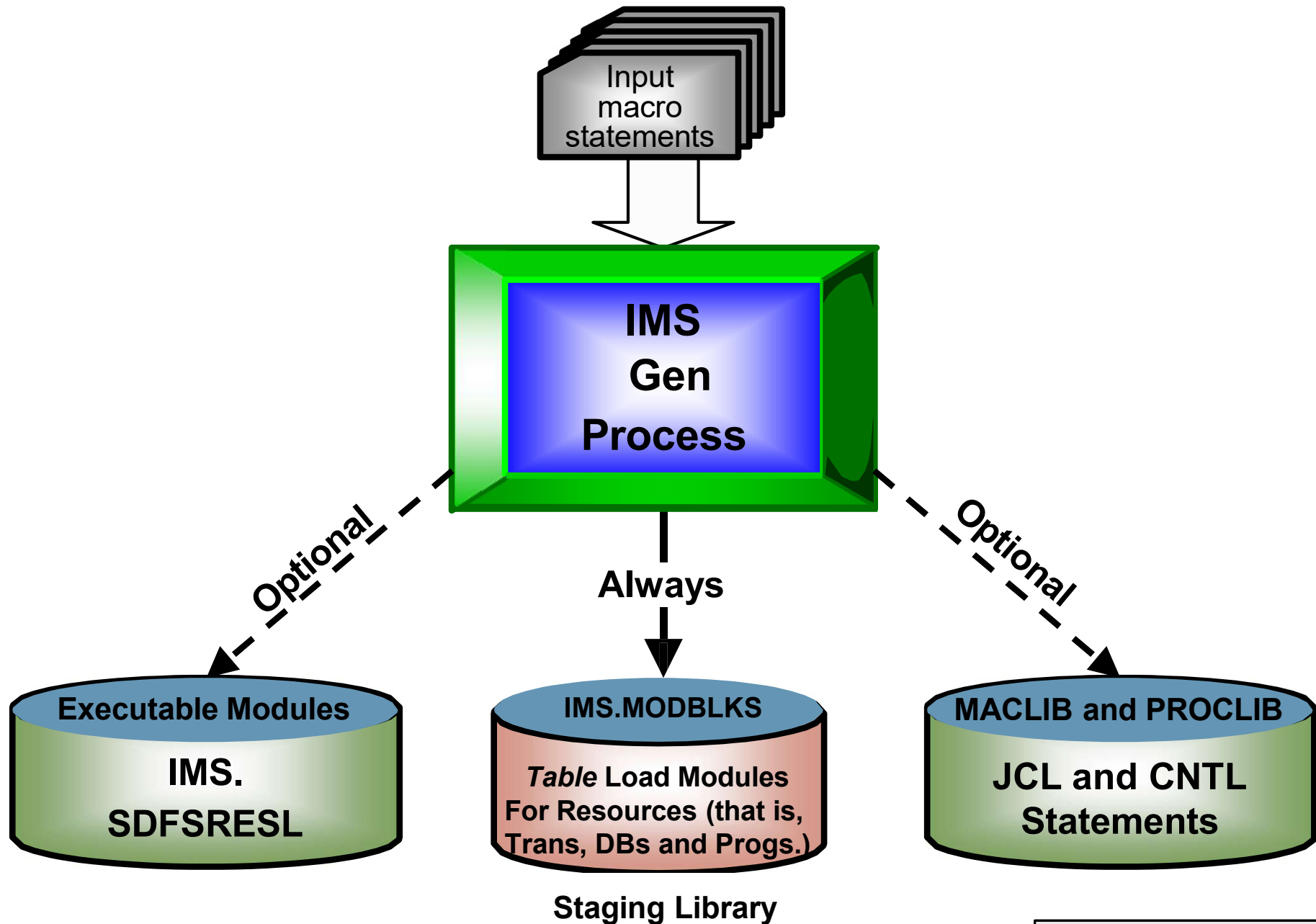
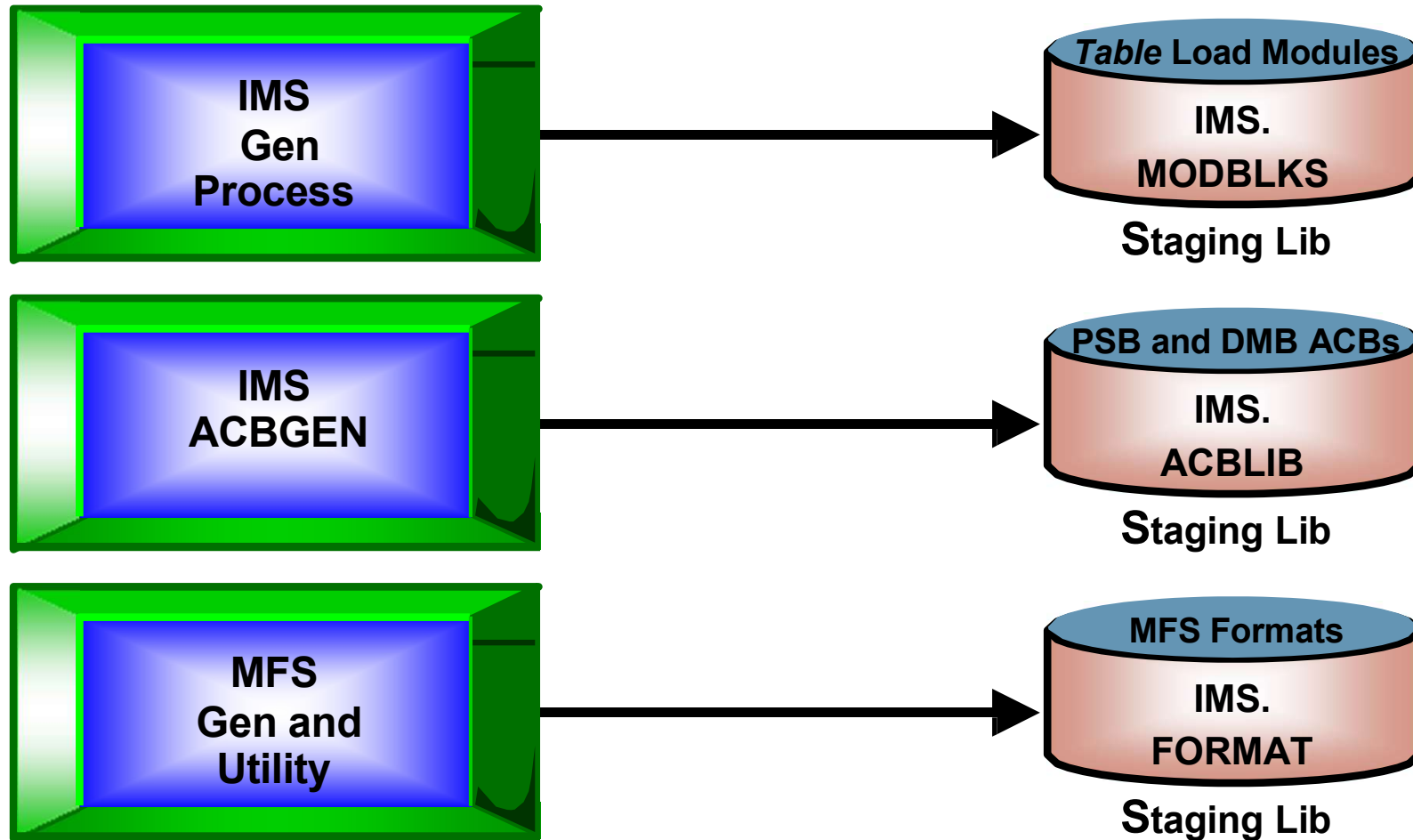


Unit 7 – IMSGEN + Online Change & Dynamic Resource Definition (DRD)

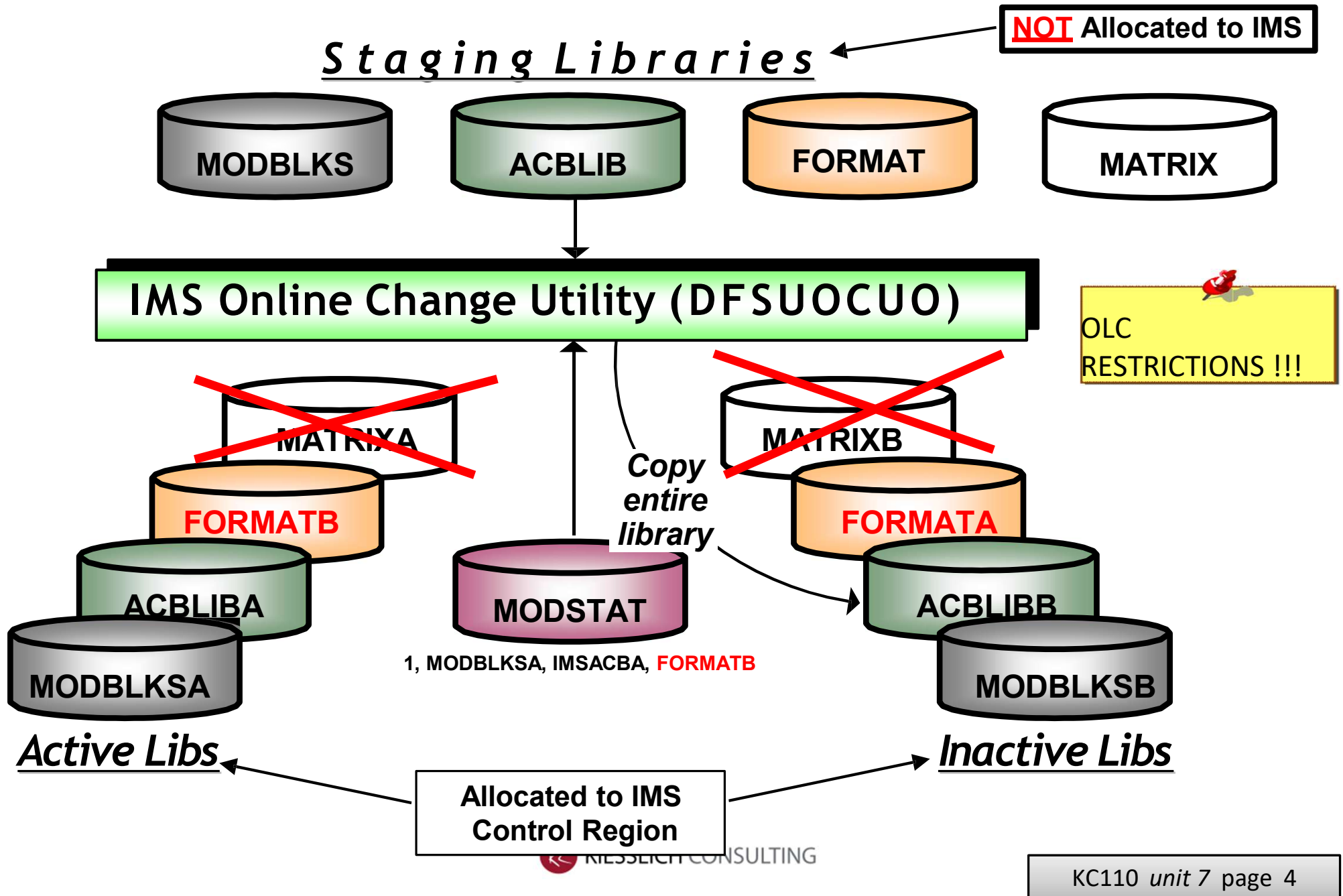
IMSGEN summary



All IMS staging libraries



IMS Online Change Utility



IMS Online Change Utility



OLC RESTRICTIONS:

- No Online Change for Terminals
 - ETO (or other OEM tool) instead
- No Additional Code can be added:
 - For example, OLC will not add FP to current system

RESIDENT DMBs/PSBs that are added or changed by OLC are NON-RESIDENT until NEXT IMS restart

Online Change Utility JCL

```
2 //OLCUTL EXEC OLCUTL, TYPE=ACB, IN=S, OUT=U, SOUT='*'
3 XX PROC TYPE=, IN=, OUT=, SOUT=A, SYS=, SYS2=, HLI=IMS
4 XXS EXEC PGM=DFSUOCU0, PARM=( &TYPE, &IN, &OUT)
5 XXSTEPLIB DD DSN=&HLI..&SYS2.SDFSRESL, DISP=SHR
IEF653I SUBSTITUTION JCL - DSN=IMS.ACBLIB, DISP=SHR
IEF653I SUBSTITUTION JCL - DSN=IMS.ACBLIBA, DISP=SHR
IEF653I SUBSTITUTION JCL - DSN=IMS.ACBLIBB, DISP=SHR
IEF653I SUBSTITUTION JCL - DSN=IMS.MODSTAT, DISP=SHR
```

```
IEF653I SUBSTITUTION JCL - DSN=IMS.MODBLKS, DISP=SHR
IEF653I SUBSTITUTION JCL - DSN=IMS.MODBLKSA, DISP=SHR
IEF653I SUBSTITUTION JCL - DSN=IMS.MODBLKSB, DISP=SHR
```

```
IEF236I ALLOC. FOR AZISI66 S OLCUTL
IEF237I 22A ALLOCATED TO STEPLIB
```

```
VPW//PRT004 DD SYSOUT=*, DCB=(RECFM=VBA)
```

IEBCOPY MESSAGES AND CONTROL STATEMENTS

COPY OUTDD=IMSACBB, INDD=IMSACB

```
1 IEB167I FOLLOWING MEMBER(S) COPIED FROM INPUT DATA SET REFERENCED BY IMSACB
- IEB154I ADFAAUDT HAS BEEN SUCCESSFULLY COPIED
IEB154I ADFABCTL HAS BEEN SUCCESSFULLY COPIED
IEB154I ADFABCTP HAS BEEN SUCCESSFULLY COPIED
IEB154I ADFABDDR HAS BEEN SUCCESSFULLY COPIED
```



Online Change command ACBLIB

example

EXAMPLE 1: ACB CHANGE

Step 1: Perform ACBGEN to the staging ACBLIB.

Step 2: Execute the Online Change Utility (copy Staging into Inactive ACBLIB).

Step 3: Issue the command to prepare the system for an ACBLIB change:

```
/MODIFY PREPARE ACBLIB
```

To list resources that still have work pending:

```
/DIS MODIFY ALL/DBS/PDS/RCS/TRS
```

To discontinue Online Change:

```
/MODIFY ABORT
```

Step 4: Switch Active and Inactive ACB Libraries; for example, activate ACBLIBB (if ACBLIBA was initially the Active Library) as the Active library and deactivate ACBLIBA:

```
/MODIFY COMMIT
```

Online Change command APPLCTN

example

EXAMPLE 2: APPLCTN (program) CHANGE

Step 1: Perform MODBLKS-GEN (or other IMSGEN type) into staging MODBLKS.

Step 2A: Execute the Online Change Utility (copy Staging into Inactive MODBLKS-Library).
Online Change Utility (copy Staging into Inactive MODBLKS-Library).

Step 2B -I: (Only if still using IMS V9 with SMU Security) Execute SMU IMS Security is used) into staging MATRIX library.

Step 2B -II: (Only if still using IMS V9 with SMU Security) Execute Online Change Utility (copy Staging into Inactive MATRIX-library).

Step 3: (All users) Issue the command to prepare the system for a MODBLKS (and optionally with V9 MATRIX) change:

/MODIFY PREPARE MODBLKS

Step 4: Switch Active and Inactive MODBLKS (and optionally with V9 MATRIX) libraries; for example, activate MODBLKSB (if MODBLKSA was initially the Active library) as the Active library and deactivate MODBLKSA:

/MODIFY COMMIT

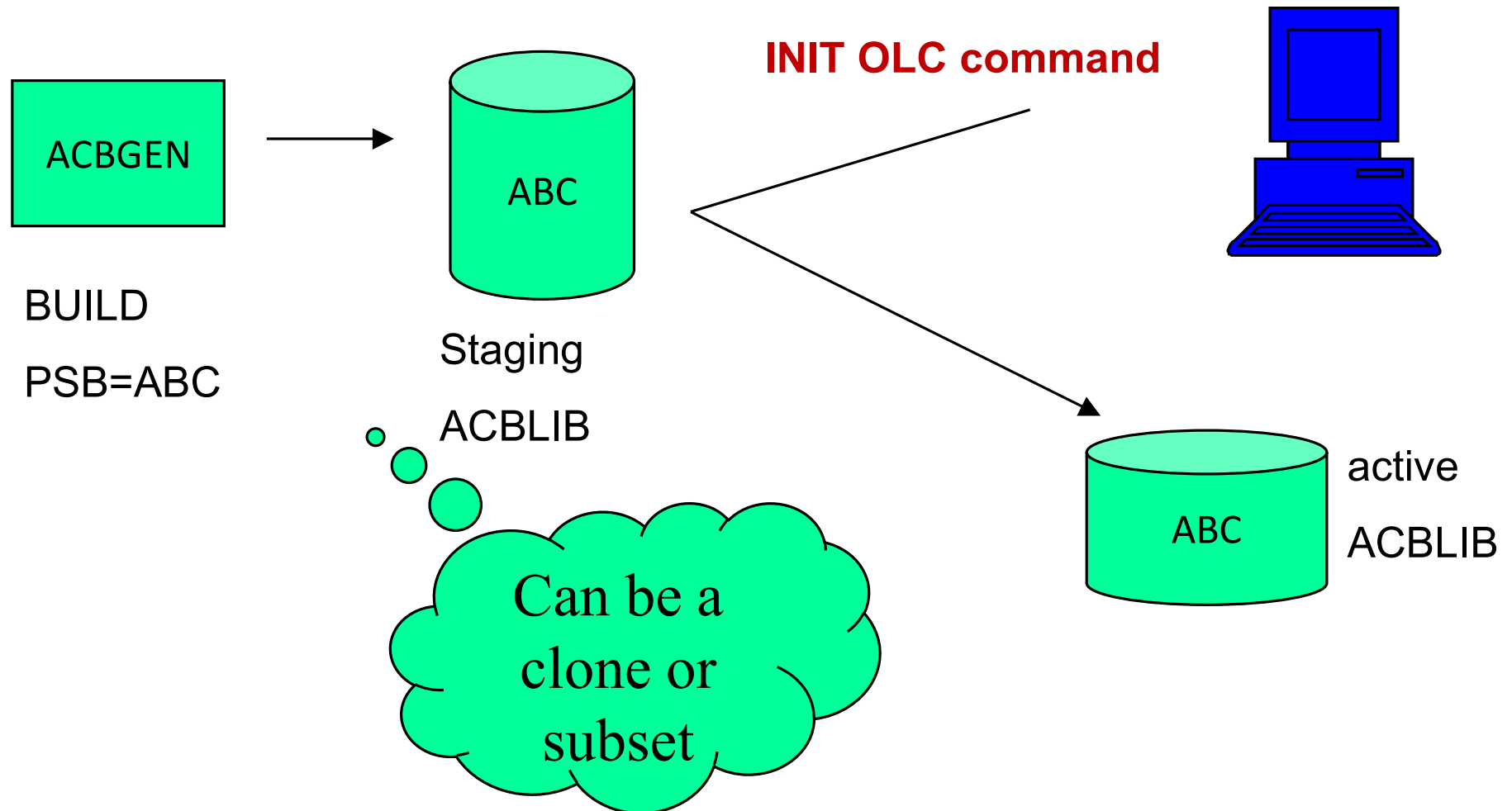
ACBLIB Member Online Change (MOLC)

- New capability since IMS 10 to add or change one or more members of the ACBLIB without the need to perform an online change on the entire library
 - **Does not support deletion of ACBLIB members**
- Only the resources that are affected by the member online change are quiesced, allowing for more concurrent activity during the online change process than the current full library switch online change
- Coexists with existing full library switch online change capability
- Goal is to improve usability and availability of online change over previous IMS versions

ACBLIB MOLC Requirements ...

- Uses existing IMS libraries (PSBLIB, DBDLIB, ACBLIB) and existing IMS control block generation processes (PSBGEN, DBDGEN, ACBGEN)
- Uses **IMS type-2** commands only
 - INIT OLC PHASE(PREPARE) TYPE(ACBMBR) ...
- Uses staging ACBLIB as the source ACBLIB
 - Full library switch OLC uses inactive ACBLIB (copied from staging ACBLIB)
- Must use OLCSTAT data set (not MODSTAT)
 - IMSplex must be using global online change
 - GOLC needs OLC=GLOBAL in DFSCGxxx or DFSDFxxx
 - Single IMS system cannot use MODSTAT
- CSL with RM required for multiple IMS systems
 - Resource structure recommended but not required
- CSL with SCI and OM required for single IMS system
 - Specify RMENV=N in DFSCGxxx or DFSDFxxx
- Coexists with existing online change ACBLIB full library switch capability
- No coexistence with previous IMS versions
 - All members in an IMSplex need to be at least at IMS 10

Overview of ACBLIB MOLC Processing



Steps to perform an ACBLIB MOLC (1)

Step 1: Build updated member(s)

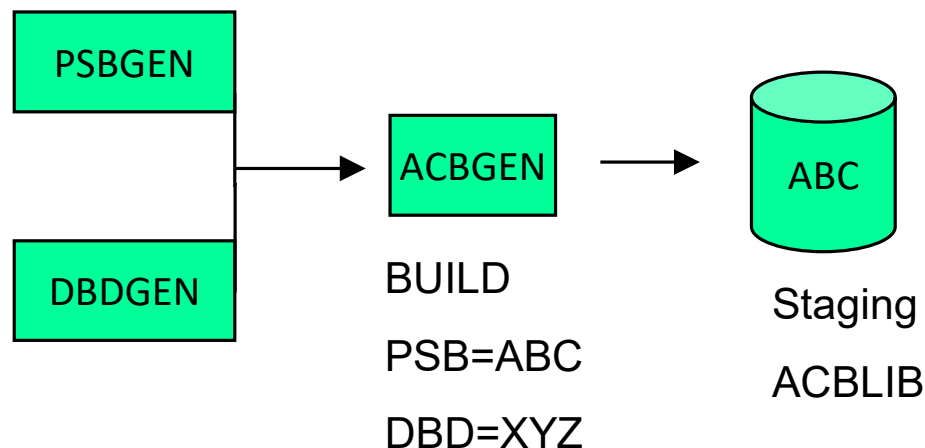
- Update PSB source and/or DBD source
- Perform PSBGEN/DBDGEN for affected members
- Perform ACBGEN into the staging ACBLIB library for affected members
 - Must use new BLDPSB=YES option on BUILD DBD= statement to get all PSBs rebuilt that are affected by a DBD change
 - BUILD DBD=(dbdname,..), BLDPSB=YES|NO
BLDPSB=YES is the default

BUILD DBD=(CUSTOMER,ORDER), BLDPSB=YES

Staging ACBLIB can be a clone or a subset of the active ACBLIB

ACBGEN enhancement will include IMS version in the utility output

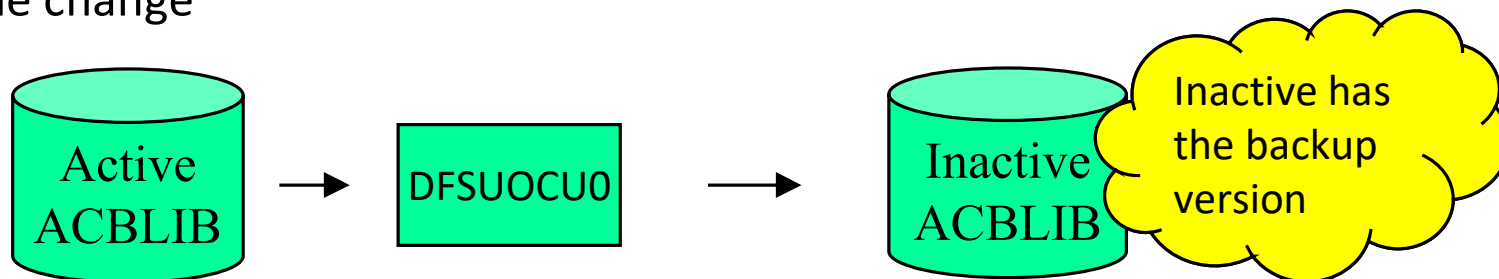
Allows user to be able to check the IMS library level used by ACBGEN



Steps to perform an ACBLIB MOLC (2)

Step 2: Create a Backup Copy (optional)

- Copy active library to the inactive library using the OLC copy utility (DFSUOCU0)
 - New TYPE=ACTVACB to create backup copy of the active ACBLIB
 - Optional to use DFSUOCU0, then user copy procedures can do copy
- Backup copy recommended (though optional) because ACBLIB member online change updates member(s) directly into the active ACBLIB dataset
- If a problem happens with ACBLIB member online change, then a full library switch online change could be used for recovery using this backup copy in the inactive library
- In an IMSplex where IMS subsystems share ACBLIBs (active/inactive), DFSUOCU0 needs to be executed on one IMS in the IMSplex
- In an IMSplex where IMS subsystems do not share ACBLIBs (active/inactive), DFSUOCU0 needs to be executed on every IMS in the IMSplex
- DFSUOCU0 TYPE=ACTVACB can also be used to synchronize the active ACBLIB to the inactive ACBLIB before doing a full library switch online change to ensure that members added to the active via MOLC will be included in the full library switch online change



Sample JCL for DFSUOCU0 for ACBLIB

Member Online Change

```
//DFSUOCU0 JOB          MSGLEVEL=1,MSGCLASS=A,CLASS=K
//STEP1    EXEC        OLCUTL, TYPE=ACTVACB,OUT=G,SOUT=*,
//          OLCLOCL='DUMMY,',OLCGLBL=,SYS=
//
```

The ACTIVE ACBLIB is copied to the INACTIVE ACBLIB (OUT=G)

```
//DFSUOCU0 JOB          MSGLEVEL=1,MSGCLASS=A,CLASS=K
//STEP1    EXEC        OLCUTL,TYPE=ACTVACB,OUT=O,SOUT=*,
//          OLCLOCL='DUMMY,',OLCGLBL=,SYS=
//
```

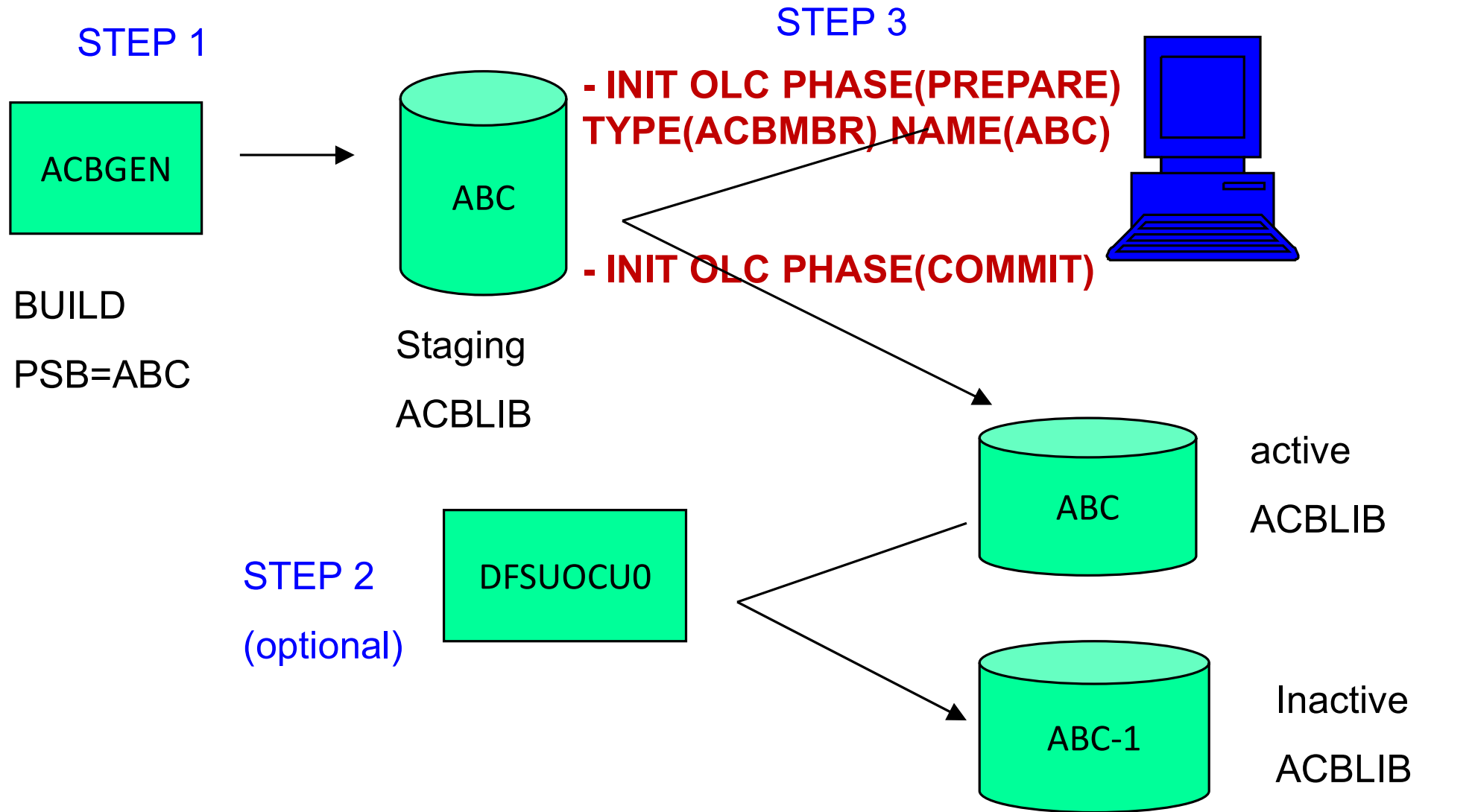
The ACTIVE ACBLIB is copied to the ACBLIB specified on the IMSACBO DD card in the OLCUTL procedure (OUT=O).

Steps to perform an ACBLIB MOLC (3)

Step 3: Bring members online via OLC INIT command

- First time setup (discussed later)
 - Allocation of staging library
 - Specify usage of ACBLIB – shared or dedicated
- Use INITIATE OLC PHASE(PREPARE) TYPE(ACBMBR) NAME(xxx,yyy) command to specify ACBs that ACBLIB member online change will process
 - Each member specified will be read from the staging ACBLIB, validation will be performed, and a prepare list will be created
- Use INITIATE OLC PHASE(COMMIT) to activate the changes
 - Updated members will be written to the active ACBLIB after verification that all members can be written successfully to the active ACBLIB
 - Only those resources affected by the change are quiesced
 - Updated members are copied directly into the active ACBLIB
 - New members go into the first active ACBLIB concatenation
 - Changed members go into the correct active ACBLIB concatenation
 - DOPT PSBs are not supported
- Process is similar to existing logic to determine what has changed

Total Process for ACBLIB MOLC



ACB MOLC Prereqs :

Specify Setup Parameters

- Ensure that the staging ACBLIB can be allocated by your IMS control region since it is now used by ACBLIB member online change
 - Create DFSDA member for ACBLIB staging library

DFSMDA TYPE=INITIAL

DFSMDA TYPE=IMSACB,DSNAME=STAGING.LIBRARY DFSMDA
TYPE=FINAL

- Or, add an IMSACB DD statement to your IMS procedure
 - If used, takes precedence over DFSMDA member

//IMSACB DD DSN=STAGING.LIBRARY, DISP=SHR

ACB MOLC Prereqs :

Specify Setup Parameters ...

- Specify whether ACBLIB is shared or dedicated in the IMSplex
 - ACBSHR=Y/N in DFSCGxxx PROCLIB member or DFSDFxxx PROCLIB member
 - Y – indicates that all of the IMSs in the OLCSTAT are using the same active and inactive ACBLIB
 - the default
 - N – indicates that each IMS in the OLCSTAT is using its own dedicated active and inactive ACBLIB
 - ACBSHR does not apply to the staging ACBLIB because it is not updated by ACBLIB member online change
 - The staging ACBLIB can be shared or dedicated
 - Multiple copies must be identical
 - All IMSs in the IMSplex sharing the OLCSTAT data set must specify the same value of ACBSHR=

ACB MOLC IMSPLEX Implications

- The ACBLIB member online change process will be coordinated among all IMSs sharing the OLCSTAT data set
 - OM chooses an IMS system to be the command master IMS
 - The command master IMS will coordinate the member OLC process with other IMSs sharing the OLCSTAT data set using RM
 - ACBSHR=Y/N in DFSCGxxx PROCLIB member or DFSDFxxx PROCLIB member specifies whether or not the active and inactive ACBLIB is shared among the IMSs in the OLCSTAT data set
 - All IMSs in the IMSplex sharing the OLCSTAT data set must specify the same value of ACBSHR=
 - Member OLC will update all the active ACBLIBs, whether shared or non-shared

ACB MOLC Prepare Phase

- Build updated members
 - INIT OLC PHASE(PREPARE) TYPE(ACBMBR) NAME(acb1,acb2,..)
 - Determine what changed
 - A DBD change will affect all the PSBs that refer to that DBD
 - Need to use BLDPSB=YES for new/changed DBDs to rebuild all related PSBs
 - Will only process members in the list and associated resources
 - All members in the list must be processed successfully or PREPARE fails
 - Create an add/change list for OLC process
 - Quiesce only affected resources
 - If PREPARE phase fails, the TERMINATE OLC command needs to be issued

ACB MOLC Commit Phase 1 + 2

- Update active ACBLIB
 - INIT OLC PHASE(COMMIT)
 - Updated member(s) are written to the active ACBLIB with encrypted name(s)
 - Member name is translated to hex characters and written as new member to the active ACBLIB
 - Existing members are not updated
 - Ensures all new/updated members fit in the active ACBLIB
 - Prevents x37 abends from creating inconsistent members in the active ACBLIB
 - If Commit Phase 1 fails, the TERMINATE OLC command needs to be issued
- Commits all members into the active ACBLIB
 - Delete old member(s)
 - Rename encrypted member(s) to the actual name(s)
 - Refresh TTRs for changed members
 - IMS now processes with new and/or changed members

Type2 Commands Used with ACBLIB MOLC

- INITIATE OLC PHASE(PREPARE) TYPE(ACBMBR)
NAME(mbrname)
- QUERY OLC SHOW(RSCLIST)
- QUERY MEMBER TYPE(IMS)
- INITIATE OLC PHASE(COMMIT)
- TERMINATE OLC

INIT OLC PHASE(PREPARE) Command

All IMSs in OLCSTAT must be V10 and up, otherwise command will fail

TYPE(ACBMBR) parameter

Specifies that a member online change is to be performed for ACBLIB members included in the NAME parameter

Mutually exclusive with any other TYPE parameter, including TYPE(ALL)

```
INITIATE OLC PHASE (PREPARE) TYPE(ACBMBR)  
NAME (acbmbr1,acbmbr2,..) OPTION(FRCNRML
```

INIT OLC PHASE(PREPARE) Command ...

- OPTION(FRCNRML) parameter
 - Only valid OPTION when TYPE(ACBMBR) specified
 - Allows a member online change to be processed if any IMS in OLCSTAT is shutdown normally
 - IMS that is down removed from OLCSTAT data set
 - When this IMS restarts and it has missed a member level online change, message DFS3433W ACBLIB MEMBER OLD ID MISMATCH MOLCID=yyyydddhhmmss is issued indicating a member level online change modify ID mismatch (MOLCID)
 - No OPTION(FRCABND support – if an IMS is down due to an abend the INIT OLC PHASE(PREPARE) TYPE(ACBMBR) command will fail; the user must do a full library switch OLC

INIT OLC PHASE(PREPARE) Command

Example

Response for: INIT OLC PHASE(PREPARE) TYPE (ACBMBR)
NAME (OLCDB105,OLCDX111)

MbrName	Member	CC	ACBSHR	DBDName	PSBName	Add	Chng
---------	--------	----	--------	---------	---------	-----	------

IMS2	IMS1	0	N				
IMS2	IMS1	0		OLCDB105		Y	
IMS2	IMS1	0		OLCDX111			Y
IMS2	IMS1	0		OLCDB111			Y
IMS2	IMS1	0		OLCDI111			Y
IMS2	IMS1	0			OLCPB105	Y	
IMS2	IMS1	0			OLCPB111		Y
IMS2	IMS2	0	N				
IMS2	IMS2	0		OLCDB105		Y	
IMS2	IMS2	0		OLCDX111			Y
IMS2	IMS2	0		OLCDB111			Y
IMS2	IMS2	0		OLCDI111			Y
IMS2	IMS2	0			OLCPB105	Y	
IMS2	IMS2	0			OLCPB111		Y

QUERY OLC Command

SHOW(RSCLIST) parameter support added

- Valid only when a TYPE(ACBMBR) online change is in progress after an INIT OLC PHASE(PREPARE) has been completed
- Returns the ACBLIB members that will be added/copied to or changed in the active ACBLIB
- Mutually exclusive with SHOW(ALL)

SHOW(ALL) includes

SHOW(ACTVLIB,DSN,LASTOLC,MBRLIST,MODID)

Need not specify LIBRARY(OLCSTAT) as is the case with global OLC

QUERY OLC SHOW(RSCLIST)

QUERY OLC SHOW(RSCLIST) Command

Example

Response for: QRY OLC SHOW(RSCLIST)

MbrName	CC	DBDName	PSBName	Add	Chng
IMS1	0	OLCDB105		Y	
IMS1	0	OLCDX111			Y
IMS1	0	OLCDB111			Y
IMS1	0	OLCDI111			Y
IMS1	0		OLCPB105	Y	
IMS1	0		OLCPB111		Y
IMS2	0	OLCDB105		Y	
IMS2	0	OLCDX111			Y
IMS2	0	OLCDB111			Y
IMS2	0	OLCDI111			Y
IMS2	0		OLCPB105	Y	
IMS2	0		OLCPB111		Y

QUERY MEMBER TYPE(IMS) Command

Can be issued after an INIT OLC PHASE(PREPARE) command has been successfully completed

New OLCMACB status added to command response to indicate that a member online change is in progress

All other global online change status will be returned in command response as well

```
QUERY MEMBER TYPE(IMS) SHOW(ALL)
```

QUERY MEMBER TYPE(IMS) Command Example

Response for: QRY MEMBER TYPE(IMS) SHOW(ALL)

MbrName	CC	Type	Status	LclAttr	LclStat	ModId
IMS1	0	IMS		GBLOLC	OLCPREPC, SECCMD, SECMSG	1
IMS2	0	IMS	OLCMACB, OLCPREPC			
IMS2	0	IMS		GBLOLC	OLCPREPC, SECCMD, SECMSG	1

INIT OLC PHASE(COMMIT) Command

Enhanced to support TYPE(ACBMBR) online change

Commits all members that were specified on the INIT OLC PHASE(PREPARE) command into the active ACBLIB

INIT OLC PHASE(COMMIT)

Response for: INIT OLC PHASE(COMMIT)

MbrName	Member	CC
---------	--------	----

IMS2	IMS1	0
------	------	---

IMS2	IMS2	0
------	------	---

TERMINATE OLC Command

- Used to abort a member online change that is in progress
- Can be issued after an INIT OLC PHASE(PREPARE) TYPE(ACBMBR) error or after an INIT OLC PHASE(COMMIT) error to terminate a member online change
- New versions of updated members will be deleted from the active ACBLIB, as long as they have not yet been committed with an INIT OLC PHASE(COMMIT) command
- Cannot be issued after the OLCSTAT data set has been updated after COMMIT complete
- Will write X'7010' log record indicating that the member online change has been terminated

TERMINATE OLC

TERMINATE OLC Command Example

Response for: TERMINATE OLC

MbrName	Member	CC
---------	--------	----

--

IMS2	IMS1	0
------	------	---

IMS2	IMS2	0
------	------	---

ACBLIB MOLC Migration

- Must be using OLCSTAT, not MODSTAT, for online change
 - Requires CSL to be set up
- OLCSTAT data set must be reformatted in IMS 10
 - OLCSTAT data set header increased from 80 bytes to 128 bytes
 - Coexistence APARS for IMS V8 (PK23401) and IMS V9 (PK23402) will allow IMS 10 to tolerate a V8/V9 OLCSTAT data set
 - Called Global Online Change Coexistence SPE
 - Global Online Change utility (DFSUOLC0) must be run
- IMSs in the IMSplex must be at IMS 10 to perform an ACBLIB member online change
- Supports XRF, FDBR, and DBCTL Warm Standby (no impact)
- Does not support RSR
- Does not support MSDBs

ACBLIB MOLC Summary

- Provides improved availability during online change process by limiting quiesced resources to only those being changed by specified ACBs
- ACBLIB member online change should be more efficient than full library switch global online change
- Complements DRD MODBLKS capability
 - Add new database definition or application definition via DRD
 - Activate associated DBDs/PSBs/ACBs via ACBLIB member online change
 - In any sequence

Dynamic allocation of ACBLIB data sets (1)

- IMS outage required prior to IMS 11 if you need to resize the ACBLIB data sets since they are allocated via JCL
- In IMS 11, DFSMDA can optionally be used instead of JCL allocation
 - Inactive ACBLIB dynamically allocated only when needed
 - Active ACBLIB will be allocated all the time
- Now the inactive ACBLIB can be resized when necessary
 - Perform an online change to switch the newly resized inactive ACBLIB to the active ACBLIB providing more space now
 - Resize the new inactive ACBLIB (previously too small active ACBLIB)
- Benefits
 - Manageability of ACBLIB is now improved
 - Users can increase the size of ACBLIB data sets without an outage
- At control region initialization, IMS checks for the presence of the IMSACBA / IMSACBB DD statements
 - If they exist, no DFSMDA members are used (same as pre-IMS 11)
 - If they do not exist, DFSMDA members are used

Dynamic allocation of ACBLIB data sets (2)

Need to create a DFSMDA member for each of the IMSACBA and IMSACBB data set concatenations

```
DFSMDA TYPE=INITIAL
DFSMDA TYPE=IMSACBA
DFSMDA
TYPE=DATASET,DSNAME=IMS.IMSPLEX.V11.ACBLIBA.DOPT
DFSMDA
TYPE=DATASET,DSNAME=IMS.IMSPLEX.V11.ACBLIBA
DFSMDA TYPE=IMSACBB
DFSMDA
TYPE=DATASET,DSNAME=IMS.IMSPLEX.V11.ACBLIBA.DOPT
DFSMDA
TYPE=DATASET,DSNAME=IMS.IMSPLEX.V11.ACBLIBB
DFSMDA TYPE=FINAL
END
```



Dynamic allocation statements for IMSACBA & IMSACBB data sets can be combined in the same job. They cannot be combined with other statements to dynamically allocate any other IMS data set.

Dynamic allocation of ACBLIB data sets (3)

- Only the active ACBLIB datasets are allocated at control region initialization !
- The inactive ACBLIB data sets are dynamically allocated during an online change process !
- After an online change, the inactive ACBLIB data sets are deallocated
- The same DFSMDA member is used to allocate data sets in both CTL and DLISAS !
 - **No inconsistencies between CTL and DLISAS can exist**
- Dynamic allocation of ACBLIB data sets is supported in all online configurations (IMS/TM, DBCTL, DCCTL, SAS and non-SAS, XRF, FDBR)
 - Not supported in batch
- Can be used for correcting errors with an unusable inactive ACBLIB
 - Due to error in copying staging ACBLIB to inactive ACBLIB in preparation for an online change
- Additional data sets can be added or changes made to the current data sets in the inactive ACBLIB concatenation

Dynamic allocation of ACBLIB data sets (4)

/DISPLAY MODIFY ALL command output now indicates new status for IMSACBA and IMSACBB

(A) Active, (I) Inactive, (U) Unallocated, () DFSMDA not used

LIBRARY	IMSACBA	(A)	IMSTESTG.DELTA1
		(A)	IMSTESTG.IMS10AC.ACBLIB1
		(A)	IMSTESTG.IMS10A.ACBLIB1
LIBRARY	FORMATA	(A)	IMSTESTG.MFS.FORMAT1
		(A)	IMSTESTG.MFS.FORMAT2
		(A)	IMSTESTG.FMT1
LIBRARY	MODBLKSA	(A)	IMSTESTG.MODBLKS1
LIBRARY	IMSACBB	(U)	IMSTESTG.DELTA2
		(U)	IMSTESTG.IMS10AC.ACBLIB2
		(U)	IMSTESTG.IMS10A.ACBLIB2
OR	LIBRARY	IMSACBB	() NO DFSMDA MEMBER
	LIBRARY	FORMATB	(I) IMSTESTG.MFS.FORMAT3
		(I)	IMSTESTG.MFS.FORMAT4
		(I)	IMSTESTG.FMT1
	LIBRARY	MODBLKSB	(I) IMSTESTG.MODBLKS2
DISPLAY MODIFY COMPLETE *08230/110121*			

SYSC

Dynamic Resource Definition (DRD)

Modifying resource definitions without DRD

- To add, change, or delete these resources in a running IMS system requires:

- MODBLKS Sysgen
- ACBGEN (if DATABASE or APPLCTN change)

< and >

- Online change process

- At some point during MODBLKS OLC process, all activity is quiesced
- One resource can prevent entire process from completing

< or >

- IMS restart

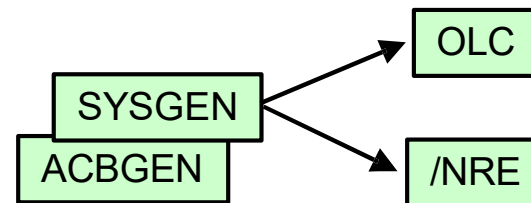
- Might not have available window to recycle IMS

- Process is more complex with multiple IMSs running in an IMSPLEX:

- Coordinated online change

< or >

- Multiple coordinated system restarts



What are Online Change limitations?

- System definition process required:
 - Coding IMS Gen macros
 - Running the complete IMS Gen process
- Adding or changing DBDs and PSBs also requires:
 - DBDGEN, PSBGEN, ACBGEN, and ACBLIB OLC
- Online change process requires three MODBLKS data sets:
 - Staging, Active, and Inactive versions of MODBLKS
 - Online change could also be performed for ACBLIB, FMTLIB, and in V9 MATRIX
 - ACBLIB and FMTLIB changes are not reflected in an IMS Gen or DRD
- Online change process can be difficult:
 - Prepare process can take a long time
 - Resources being changed must be *quiesced*
 - Stop queuing and drain queues
 - Commit process:
 - Stops *all* scheduling, at least momentarily
 - Might be difficult to resolve work-in-progress (CICS, ODBA, BMPs)

DRD is the solution

Dynamic Resource Definition

- Allow individual resource definitions to be:
 - Created
 - Deleted
 - Updated
 - without the need for MODBLKS OLC or IMS RESTART
 - Resources are:
 - Transactions
 - Routing codes
 - Programs
 - Databases
- Does not require resources NOT being changed to be quiesced

Objective and requirements of DRD

- **OBJECTIVE:** To improve the availability of the IMS Online Environment by allowing the user to dynamically define and enable certain resource definitions without the requirement for an IMS Sysgen plus an IMS restart or an Online Change and the unavailability associated with either
- Resources include:

RESOURCE	SYSGEN MACRO	CONTROL BLOCK
Database	DATABASE (DB/TM, DBCTL)	DDIR
Application Program (PSB)	APPLCTN (DB/TM, DBCTL, DCCTL)	PDIR
Transaction	TRANSACT (DB/TM, DCCTL)	SMB
Routing Code	RTCODE (DB/TM, DCCTL)	RCTE

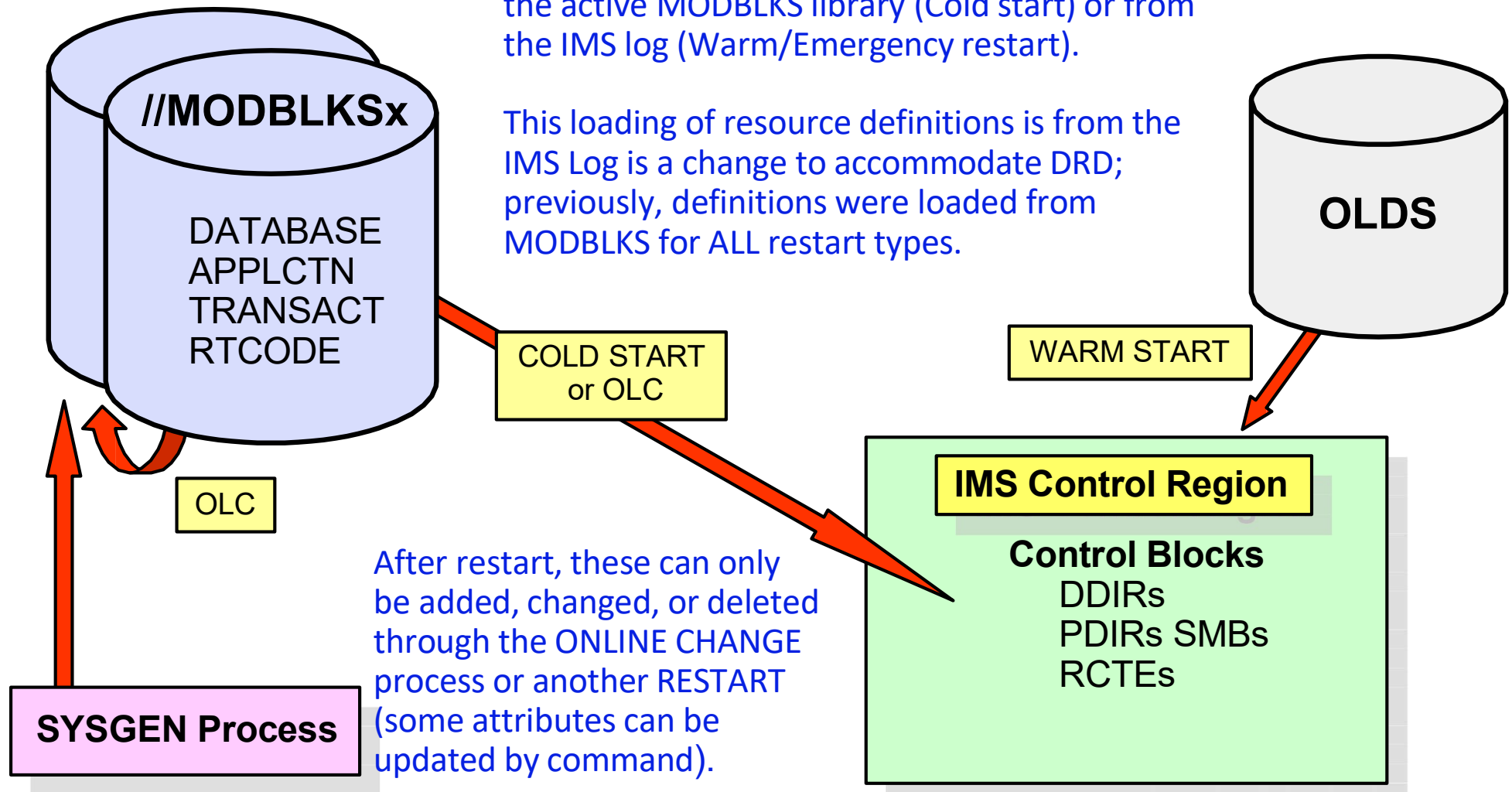
Major components of DRD

- New and enhanced Type-2 commands to dynamically CREATE, DELETE, and UPDATE IMS resource definitions traditionally found in MODBLKS library
 - Databases, programs, transactions, routing codes
- A set of *resource definition data sets* to contain both statically defined (Sysgen) and dynamically created (DRD) definitions
 - Resource definitions
 - Model descriptors
- New function to automatically IMPORT and EXPORT resource definitions from/to a resource definition data set
- An enhanced Type-2 command to QUERY the attributes and status of defined IMS resources and descriptors

Modifying MODBLKS resources without DRD

During IMS **V10** restart processing, resource definitions are loaded from the active MODBLKS library (Cold start) or from the IMS log (Warm/Emergency restart).

This loading of resource definitions from the IMS Log is a change to accommodate DRD; previously, definitions were loaded from MODBLKS for ALL restart types.

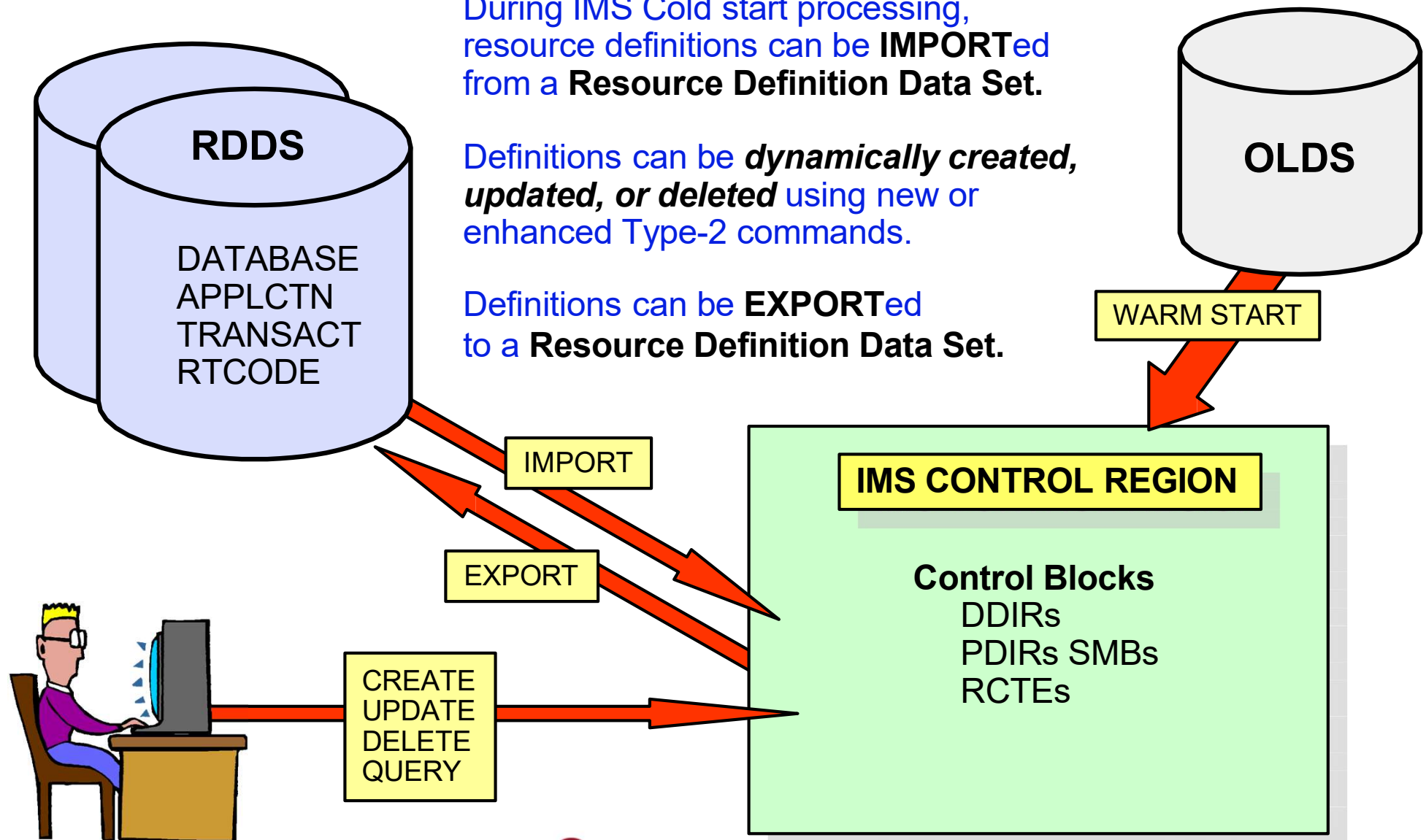


Modifying MODBLKS resources with DRD

During IMS Cold start processing, resource definitions can be **IMPORTED** from a **Resource Definition Data Set**.

Definitions can be *dynamically created, updated, or deleted* using new or enhanced Type-2 commands.

Definitions can be **EXPORTED** to a **Resource Definition Data Set**.



PROCLIB members (1 of 2)

- DFSPBxxx:
 - CSLG=xxx
 - Suffix to DFSCGxxx – CSL PROCLIB member
 - DFSDF=xxx
 - Suffix to DFSDFxxx – IMS Definition PROCLIB member – new in V10
- DFSCGxxx:
 - CSL PROCLIB member
 - DRD requires a CSL environment with SCI and OM; RM not required
 - MODBLKS=DYN
 - This enables DRD
 - Alternate, default value (no DRD) is MODBLKS=OLC (Online Change)
 - This PROCLIB member can be replaced with CSL section of DFSDFxxx
 - But if both are coded, this one (DFSCGxxx) overrides conflicting values in DFSDFxxx

PROCLIB members (2 of 2)

- DFSDFxxx:
 - New *IMS Definition* PROCLIB member
 - DFSDF=xxx in DFSPBxxx
 - Divided into Sections:
 - <SECTION=COMMON_SERVICE_LAYER>
 - Must include MODBLKS=DYN for DRD
 - Might define CSL parameters (alternative to DFSCGxxx)
 - > If both exist, DFSCGxxx parameters override DFSDFxxx parameters
 - <SECTION=DYNAMIC_RESOURCES>
 - Define DRD parameters
 - Other sections for other parameters:
 - <SECTION=DIAGNOSTIC_STATISTICS>
 - <SECTION=SHARED_QUEUES>

DRD Section of DFSDFxxx

- DRD is enabled by parameter in DFSCGxxx or CSL section of DFSDFxxx
 - MODBLKS=DYN
- Parameters in DFSDFxxx determine the automatic import and export functions of DRD

```
<SECTION=COMMON SERVICES>  
MODBLKS=DYN
```

```
<SECTION=DYNAMIC RESOURCES>  
RDDSDSN=(IMS10.RDDSDSN1,IMS10.RDDSDSN2,...)  
AUTOEXPORT=AUTO | RDDS | NO  
AUTOIMPORT=AUTO | RDDS | MODBLKS | NO  
IMPORTERR=ABORT | CONTINUE  
RDDSERR=ABORT | NOIMPORT
```

DRD resources

- Resources have names, attributes, and status
 - Name
 - 1-8 characters – with all the usual restrictions
 - Attribute
 - A property of the resource specified during resource definition or update
 - Status
 - The availability or usability of the resource
- During Cold start:
 - When DRD not enabled, resource control blocks loaded from MODBLKS
 - When DRD enabled, control blocks loaded from RDDS, MODBLKS, or not at all
- During Warm or Emergency restart:
 - Control blocks always loaded from restart log
- During execution
 - Resource control blocks can be dynamically created, updated, or deleted

Resource attributes

- A property of a resource defined during Sysgen process or using DRD CREATE or UPDATE process:
 - Example from Sysgen
DATABASE RESIDENT,ACCESS=UP,DBD=ACCTMSTR
 - Example from DRD
**CREATE DB NAME(ACCTMSTR)
SET(RESIDENT(Y),ACCTYPE(UPD))**
 - RESIDENT and UPDATE access are attributes of the ACCTMSTR database
- Default attributes:
 - Value used when not explicitly specified in Sysgen or DRD process
 - For example, DATABASE default for ACCESS
 - EX (exclusive) during Sysgen
 - Found in **default descriptor** for resource type being defined
- Default descriptors:
 - Identify default attributes for DRD *Create* process
 - Can be user-defined (CREATE DBDESC) or IMS-supplied (DFSDSDB1)

DRD descriptors

- A model (template) for defining (creating) a resource or another descriptor
 - Establishes defaults for attributes not SET in CREATE command
- IMS-defined descriptors
 - Provided with the IMS product
- User-defined descriptors
 - Defined by the user
- Current system default descriptor
 - Each resource type will have one current default descriptor
 - IMS-defined or user-defined

Resource Definition Data Sets

- Set of BSAM data sets containing definitions of resources and descriptors
 - Used in a *round-robin* fashion
- Each IMS has its own set of RDDSs
 - Cannot be shared across IMSPLEX
- Target for *exporting* definitions
 - Externalize resource and descriptor definitions
- Source for *importing* definitions
 - Load resource and descriptor definitions

IMS *direction* is to have a *Repository* to hold definitions which can be shared across all IMSs in an IMSPLEX..

Recoverability

- Resource and descriptor definitions ...
 - Exist for the life of IMS or until deleted
 - Are recovered across Warm and Emergency restart:
 - Definitions are logged when created, updated, or deleted, and at system checkpoint time
 - Definitions are restored from logs during restart processing
 - Are lost across Cold start unless:
 - Previously EXPORTed to RDDS < and then >
 - IMPORTed during Cold start
< or >
 - Sysgen-ed into MODBLKS library and IMPORTed during Cold start

Commands used in DRD

- Type-2 commands entered through OM interface to
 - CREATE, UPDATE, DELETE, and QUERY resources and descriptors

Command	Short Form	Purpose
CREATE	CRE	Create resource or descriptor definition
DELETE	DEL	Delete resource or descriptor definition
UPDATE	UPD	Update attributes of resource or descriptor definition Update status of resource
QUERY	QRY	Query attributes of resource or descriptor definition Query status of resource
**IMPORT	IMP	Import resource and descriptor definitions from RDDS
**EXPORT	EXP	Export resource and descriptor definitions from RDDS

****** Neither the *IMPORT* nor the *EXPORT* command are supported before V11 .
V12 is providing improvements here .

UPDATE command

```
UPDATE      rsc-type | desc-type  
            NAME (names)  
            SET (attr1 (val1) , attr2 (val2) , ... )  
            DEFAULT (Y)    << descriptors only
```

- NAME:
 - Can specify multiple names
 - Can use wild cards (* and/or %)
 - Update will apply to all names
- SET
 - All attributes of all resources or descriptors can be SET
- DEFAULT(Y) – For descriptors only:
 - “Y” makes descriptor the current system default descriptor
 - Cannot specify DEFAULT(N)
 - Must specify “Y” on another CRE or UPD

RDDS Extraction Utility

- RDDS Extraction Utility:
 - Creates Stage 1 macros or CREATE commands from definitions in RDDS:
 - Can be used to copy definitions to another system
 - Can be used to fall back to non-DRD use
 - Can be used to create documentation of system definitions
 - Stage 1 macros:
 - DATABASE, APPLCTN, TRANSACT, and RTCODE macros are written
 - Does not write descriptor information
 - Does not write other Stage 1 macros (IMSCTRL, IMSCTF, and so on)
 - CREATE commands
 - CREATE commands for DB, DBDESC, PGM, PGMDESC, TRAN, TRANDESC, RTC, and RTCDESC are written

RDDS Extraction Utility JCL

- Sample JCL

```
//MYJOB JOB CLASS=J,MSGCLASS=A,MSGLEVEL=(1,1)
//JOBLIB DD DSN=IMS10.SDFSRESL,DISP=SHR
//S1 EXEC PGM=DFSURDD0,MEMLIMIT=4G
//DFSRRDDS DD DSN=IMS10.RDDS01,DISP=SHR
//SYSOUT DD DSN=MY.RDDS.OUTPUT,DISP=(,CATLG,DELETE),
// UNIT=SYSDA,VOL=SER=ABC001,
// SPACE=(CYL,(1,1),RLSE),
// DCB=(LRECL=80,RECFM=FB,BLKSIZE=800)
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
OUTPUT=MAC
/*
//
```

- **SYSIN control statement:**

- OUTPUT=MAC creates Stage 1 macros
- OUTPUT=CMD creates CREATE commands

IMS REPOSITORY

- IMS Repository commands
 - IMS and RM IMSPLEX commands issued from SPOC
 - Batch interface commands
 - Repository Server commands issued through z/OS modify interface
- Comparison of DRD use with RDDS versus repository
- Using DRD with the IMS Repository in an online environment
- Managing the IMS Repository in an offline batch environment
- Migration to repository
- Security considerations
- DRD user interface enhancements
- IVP enhancements for repository
- Summary

IMS Repository Function

- A 'repository' is a generalized data storage facility that can be used to store various types of information
- The IMS Repository function is a centralized method for storing and retrieving resource definitions in an IMSPLEX
 - Enables multiple IMS systems in a multiple-IMS IMSPLEX to manage, store, share, and retrieve resource definitions
 - Enables a single IMS system in a single-IMS IMSPLEX to manage, store, share, and retrieve resource definitions
- Focus is on improving the systems management and resource management aspects of handling IMS resource definitions
 - Across multiple IMSs or for a single standalone IMS
 - For test systems, for production systems

IMS Repository Function Usage

- The various components of the IMS Repository function provide a centralized storage and retrieval solution for resource definitions
- In IMS 12, the resource and descriptor definitions for Dynamic Resource Definition (DRD) can be stored in an IMS Repository
 - Contains resource definitions for programs/transactions/databases/FP routing codes & descriptors
 - Called the IMSRSC, the IMS resource definition repository
 - Provides an alternative to using RDDSs (resource definition data sets) for DRD
 - But both the Repository and RDDS can coexist
 - Can replace one or more sets of RDDSs in an IMSPLEX with a single repository
 - Eliminates the need to manually coordinate and manage separate RDDSs per IMS across a multiple-IMS IMSPLEX
 - Provides an alternative to using MODBLKs with SYSGEN and online change
 - Considered a strategic alternative to the RDDS
- IMS 12 can retrieve the stored resource definitions from the IMSRSC Repository to dynamically generate runtime resources for DRD

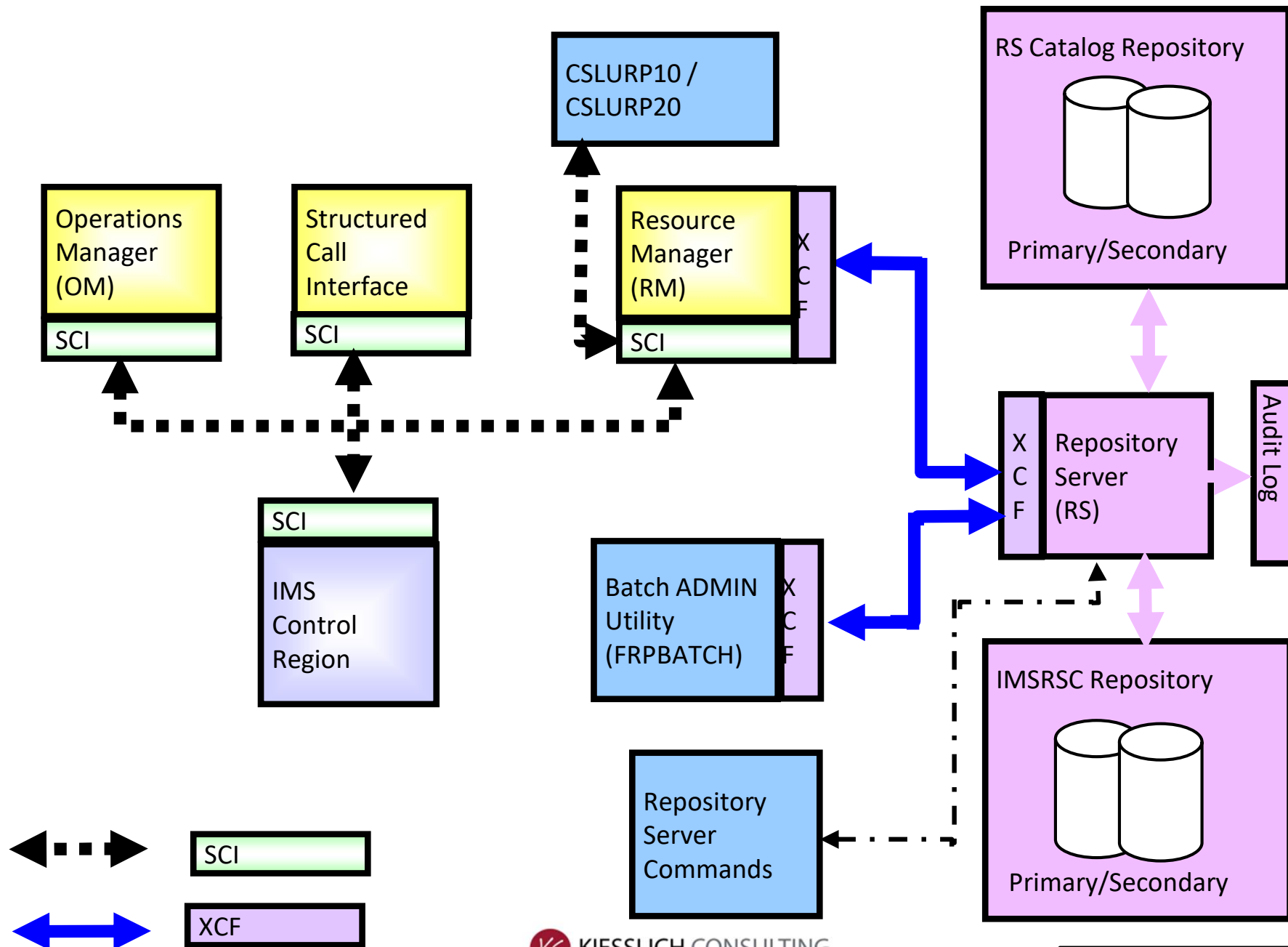
Resource Definitions

- DRD Resources in IMS's memory, i.e. those being used by the online IMS system, are referred to as -
 - **Runtime Resources** or **Runtime Resource Definitions**
- DRD Resources saved in a Repository or RDDs are referred to as -
 - **Stored Resources** or **Stored Resource Definitions**
- EXPORT command copies Runtime Resources to Stored Resources
 - Can EXPORT to RDDs or Repository
 - Executed on one IMS, it can specify that resource definitions exported to the Repository are to apply to multiple named IMS systems
 - But doesn't change any runtime resource definitions
- IMPORT command copies Stored Resources to Runtime Resources
 - Executed on one IMS, it can specify that stored resource definitions from the Repository to become runtime resource definitions in multiple IMS systems

IMS Repository Function Components

- A Repository Server (RS)
 - A new BPE-based address space managed by the Resource Manager (RM) CSL address space
- Repositories (data sets)
 - Catalog Repository
 - A catalog of the resource repositories
 - Used by the Repository Server
 - IMSRSC Repository(s)
 - Contains DRD stored resource definitions
- A Common Service Layer (CSL) IMSPLEX configuration consisting of
 - Operations Manager (OM)
 - Resource Manager (RM)
 - Structured Call Interface (SCI)
 - SPOC for entering type-2 commands
 - Optional Resource Structure with CQS address space
- Batch utilities
 - Batch ADMIN utility to manage the Catalog Repository
 - RDDS to / from Repository utilities

IMS Repository Function Architecture



IMS Repository Function Benefits

- Consolidation of resource definitions in a single place, the Repository
- DRD definitions are the initial implementation of the IMS Repository function (to optionally replace RDDSs)
- Full support for populating, managing, storing, sharing, and retrieving a consistent set of DRD stored resource definitions for multiple-IMS IMSPLEXes and single-IMS IMSPLEXes
 - Repository can be implemented without an outage
- Manual coordination of multiple RDDSs in a multiple-IMS IMSPLEX eliminated, replaced by basic functioning of the IMS Repository
- Improvements in IMSPLEX systems and resource management with the Repository
 - Via commands (3 types)
 - Via batch utilities (3 functions)
- A strategic direction for IMS architecture

DEMO ...

Help

PLEXA

IMS Manage Resources

Command ==> _____

Select an action and press Enter.

* Action _

1. Create new resources

2. Delete resources

3. Query resources

4. Update resources

5. Export resources

6. Import resources

7. Manage RDDS

F1=Help F12=Cancel

IMS Repository Function Components ...

Batch Utilities

Batch ADMIN utility (FRPBATCH)

Commands for managing content of the Catalog Repository , Functions such as

- ✓ ADD a new IMSRSC Repository,
- ✓ LIST the characteristics of an IMSRSC Repository,
- ✓ START or STOP an IMSRSC Repository

RDDS to / from repository utilities (Batch RM utilities)

- ✓ RDDS to Repository Utility (CSLURP10)
 - For migration
- ✓ Repository to RDDS Utility (CSLURP20)
 - For fallback