

The Tech World's Newest Catalog – the IMS Catalog

SHARE Session 17767

Nancy G. Stein - ngstein@us.ibm.com

IBM Washington Systems Center / IMS Team



August 10, 2015

#SHAREorg



SHARE is an independent volunteer-run information technology association that provides **education, professional networking and industry influence.**



Acknowledgements and Disclaimers

Availability. References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates.

The workshops, sessions and materials have been prepared by IBM or the session speakers and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While efforts were made to verify the completeness and accuracy of the information contained in this presentation, it is provided AS-IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other materials. Nothing contained in this presentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.

© **Copyright IBM Corporation 2015. All rights reserved.**

— **U.S. Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.**

IBM, the IBM logo, ibm.com, z/OS, IMS, ODM, DB2, CICS, WMQ, WAS, WOLA, RAA, RAD, RDz, RTC, are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or TM), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at

•“Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml

•Other company, product, or service names may be trademarks or service marks of others.

IMS Catalog Topics

- IMS Catalog Background
- What is IMS Metadata ?
- IMS Catalog Database
- IMS Catalog Enablement and Implementation
- Catalog Populate Utility
- New ACB Gen and Catalog Populate Utility
- Utilities for the IMS Catalog
- Application Programming with the Catalog
- IMS Catalog Configuration in an IMSplex
- Catalog Implementation Considerations
- Benefits and Summary

IMS Catalog Background

- Optional Feature Introduced in IMS 12, strongly recommended
 - IMS 12 GA'd in OCT 2011
 - IMS HALDB (High Availability Large Database)
- A trusted, online source for IMS Database and Application *metadata* information, fully managed by IMS
- Enables scalable and flexible IMS Open Database solutions
 - Applications no longer need to maintain local IMS database Java metadata
- Opens up metadata discovery and exchange via IMS Open Database
 - View and edit PSBs and DBDs
 - Cross reference DBDs to PSBs, and vice versa
- Applications and Tools can access the Catalog via standard interfaces
 - JDBC/SQL
 - IMS 13 Native COBOL SQL
 - Traditional IMS DL/I
- Major component of the IMS simplification and integration strategy
- Foundation for existing and future IMS features

Complete your session evaluations online at www.SHARE.org/Orlando-Eval

IMS Catalog Background

- IMS features dependent upon the Catalog being enabled:
 - IMS 13 Database Versioning
 - IMS 13 IMS Native SQL support for COBOL 5.1
 - IMS 13 .NET access to IMS data
 - IMS 14 Dynamic Database Definition
 - IMS 14 IMS Managed ACBs
- Some products which can utilize the IMS Catalog:
 - IMS Explorer for Development
 - IBM Data Studio
 - COGNOS 10.2
 - InfoSphere Data Architect, DataStage, Metadata Asset Manager
 - IBM WAS Liberty Profile, IBM MobileFirst, and IMS Mobile feature Pack
 - Rational Asset Analyzer

What Is Metadata?

"The database catalog of a database instance consists of metadata in which definitions of database objects such as base tables, views (virtual tables), synonyms, value ranges, indexes, users, and user groups are stored." (Wikipedia)

Metadata is "data about data". (Merriam-Webster)

"Metadata was traditionally in the card catalogs of libraries. As information has become increasingly digital, metadata are also used to describe digital data using metadata standards specific to a particular discipline. By describing the contents and context of data files, the usefulness of the original data/files is greatly increased." (Wikipedia)

"The main purpose of metadata is to facilitate in the discovery of relevant information, more often classified as resource discovery. Metadata also helps organize electronic resources, provide digital identification, and helps support archiving and preservation of the resource. Metadata assists in resource discovery by 'allowing resources to be found by relevant criteria, identifying resources, bringing similar resources together, distinguishing dissimilar resources, and giving location information.' (NISO Press) "
(Wikipedia)

IMS now has database catalog that meets all of these definitions. This session will show you how IMS does this.

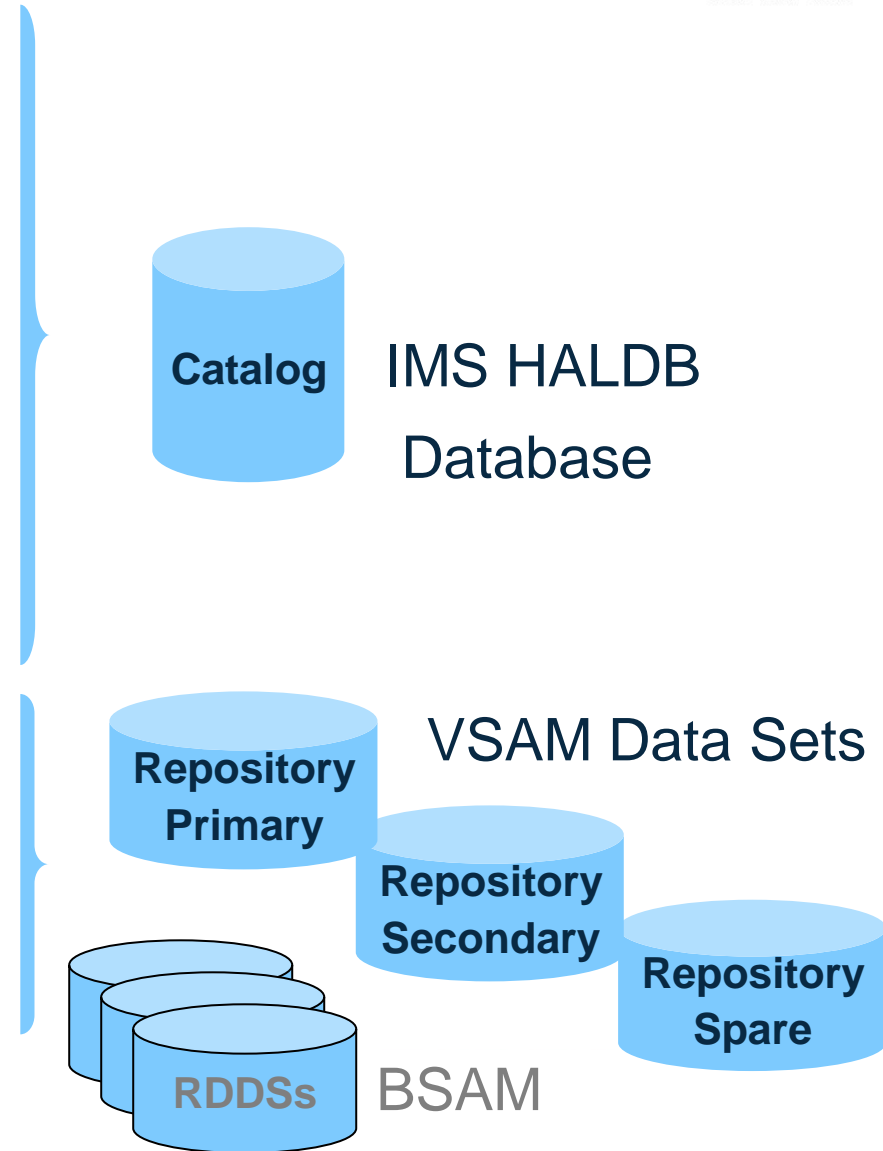
What Is Metadata?

- B**
 - **Business metadata**
 - Business rules, definitions, terminology, glossaries, algorithms and lineage using business language
 - Audience: Business users
- T**
 - **Technical metadata**
 - Defines source and target systems
 - Table and field structures and attributes
 - Derivations and dependencies
 - Audience: Specific tool users, AD, BI, ETL, profiling, modeling
- O**
 - **Operational metadata**
 - Information about application runtimes
 - Frequency, record counts, component by component analysis and other statistics
 - Audience: Operations, management and business users

Literally, “*data about data*” that *describes* your company’s *information from multiple perspectives*.

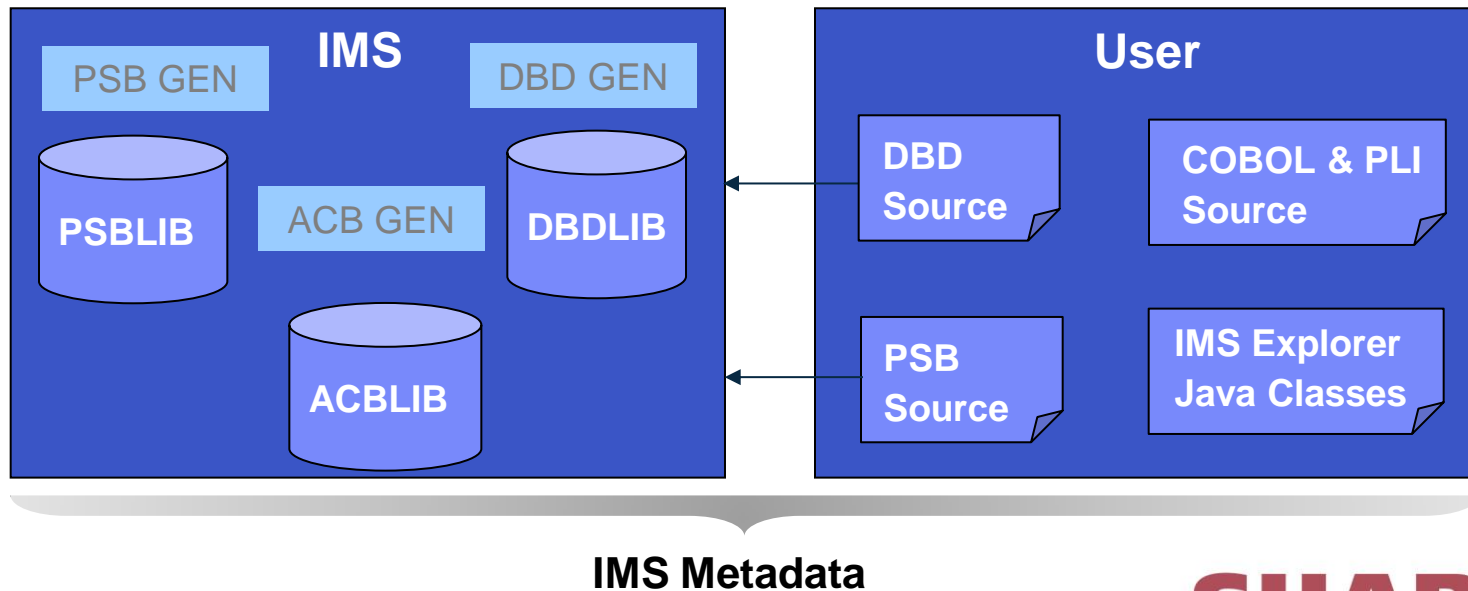
Types of IMS Technical Metadata and Storage Method

- **DB**
 - PSB/DBD resources
 - Database structure definitions
 - Physical database definitions
 - Segment definitions
 - Field definitions
 - Application
 - Data types
 - Application defined fields
 - Encodings
 - Redefines
 - User defined types
 - Structures
- **TM**
 - MODBLKS resources
 - Program definitions
 - Transaction definitions
 - MFS FORMAT resources
 - Application
 - Input/output message definitions

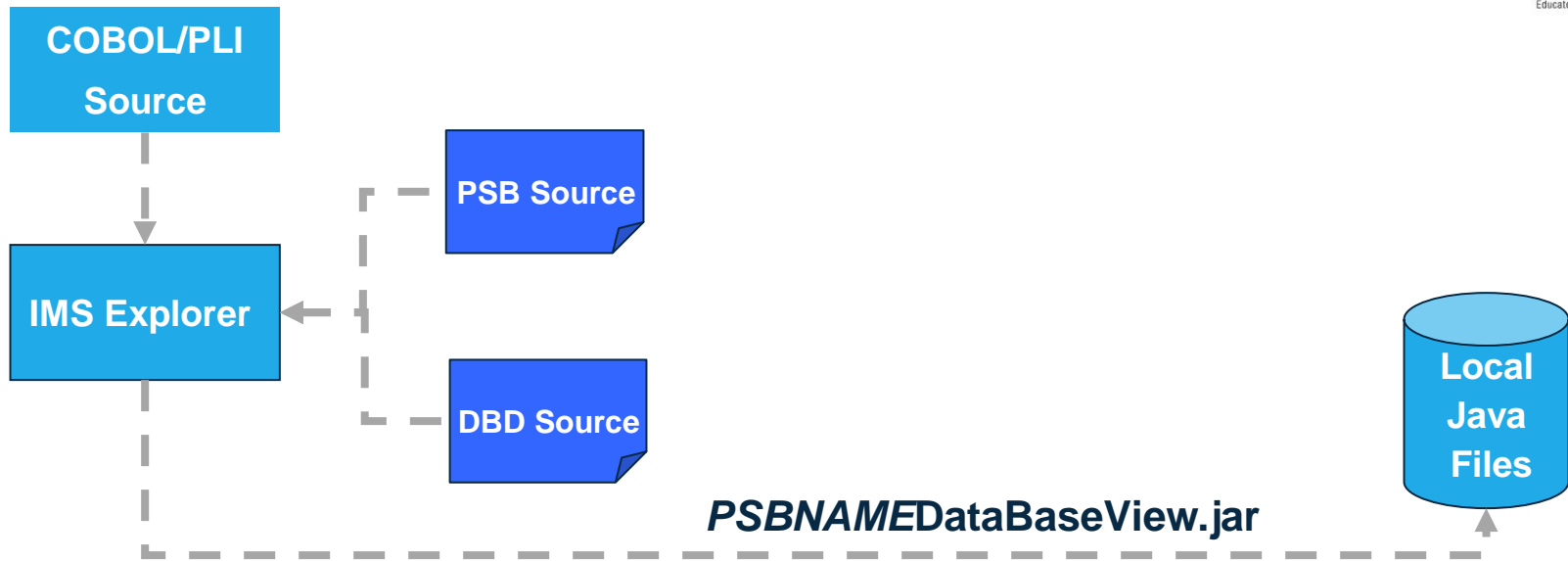


IMS Metadata Pre-Catalog

- Databases partially defined in the IMS DBD
 - Only key/seq and searchable fields needed by applications
 - Remaining segment data is not defined and not known to IMS
- Remaining database segment definition is defined within Applications
 - COBOL COPYBOOKS and PL/I INCLUDEs map the segment data
 - Applications can have different mappings for one segment



IMS Metadata Pre-Catalog



- IMS Explorer parses DBD, PSB and COBOL or PL/I source code
 - Generates ***PSBNAMEDataBaseView.java*** and **.class** files
 - Exported as .jar file to file system where application is deployed
- Key points
 - Metadata can be deployed to multiple servers and file locations
 - **Metadata is local, offline and not guaranteed to be current**

With the IMS Catalog

- Database and program resources defined to an IMS system and relevant application information can be stored as metadata in an IMS catalog
 - Databases, fields, segments, data types, mappings and more ...
- The IMS catalog can be updated when you create, alter or delete IMS database or program resources, or application information
- Updates to the IMS catalog are done only via integrated IMS utilities
 - Catalog Populate Utility, DFS3PU00 (initial catalog load and member update)
 - New ACBGEN and Catalog Populate Utility (DFS3UACB)
- Catalog metadata can be used to:
 - Enhance understanding of the data
 - Improve consistency of the data
 - Improve impact analysis of data
 - Improve development productivity
 - Improve data governance
- Key component of the overall IMS strategy:
 - Simplification
 - Integration

The IMS Catalog Database



- Contains metadata related to an IMS system's databases and programs
 - DBD and PSB and Application information
- IMS PHIDAM / OSAM HALDB database
 - Defined with 4 Data Set Groups
- One Secondary Index
 - Can be used to cross reference a DBD to PSBs and vice versa
- Unique features
 - DBRC use is optional for the IMS Catalog HALDB database
 - “Unregistered Catalog Database”
 - **Only** HALDB that isn't required to be defined to DBRC
 - IMS can manage allocation/creation of the Catalog DB data sets
 - Done via parameters in the “CATALOG” section of DFSDFXxx PROCLIB member

The IMS Catalog Database

- IMS provides the DBD and PSB source code for the Catalog database
- IMS provides the object code for the Catalog DBDs and PSBs
- PHIDAM DBD name is DFSCD000
- PSINDEX DBD name is DFSCX000
 - Used to cross reference DBDs to PSBs
- PSBs provided to load, read and update the Catalog database
 - DFSCPL00 used for initial load process
 - Only used by the Catalog Populate Utility (DFS3PU00)
 - DFSCP000 used for read access from COBOL/HLASM
 - DFSCP002 used for read access from PL/I
 - DFSCP003 used for read access from PASCAL
 - DFSCP001 used for update access
 - Used by ACBGEN (DFS3UACB), Catalog Populate Utility (DFS3PU00), and Purge Utility (DFS3PU10)
- Default catalog PCB is DFSCAT00
- ACBGEN required for all catalog PSBs before enabling, populating, and using the IMS Catalog

The IMS Catalog Database

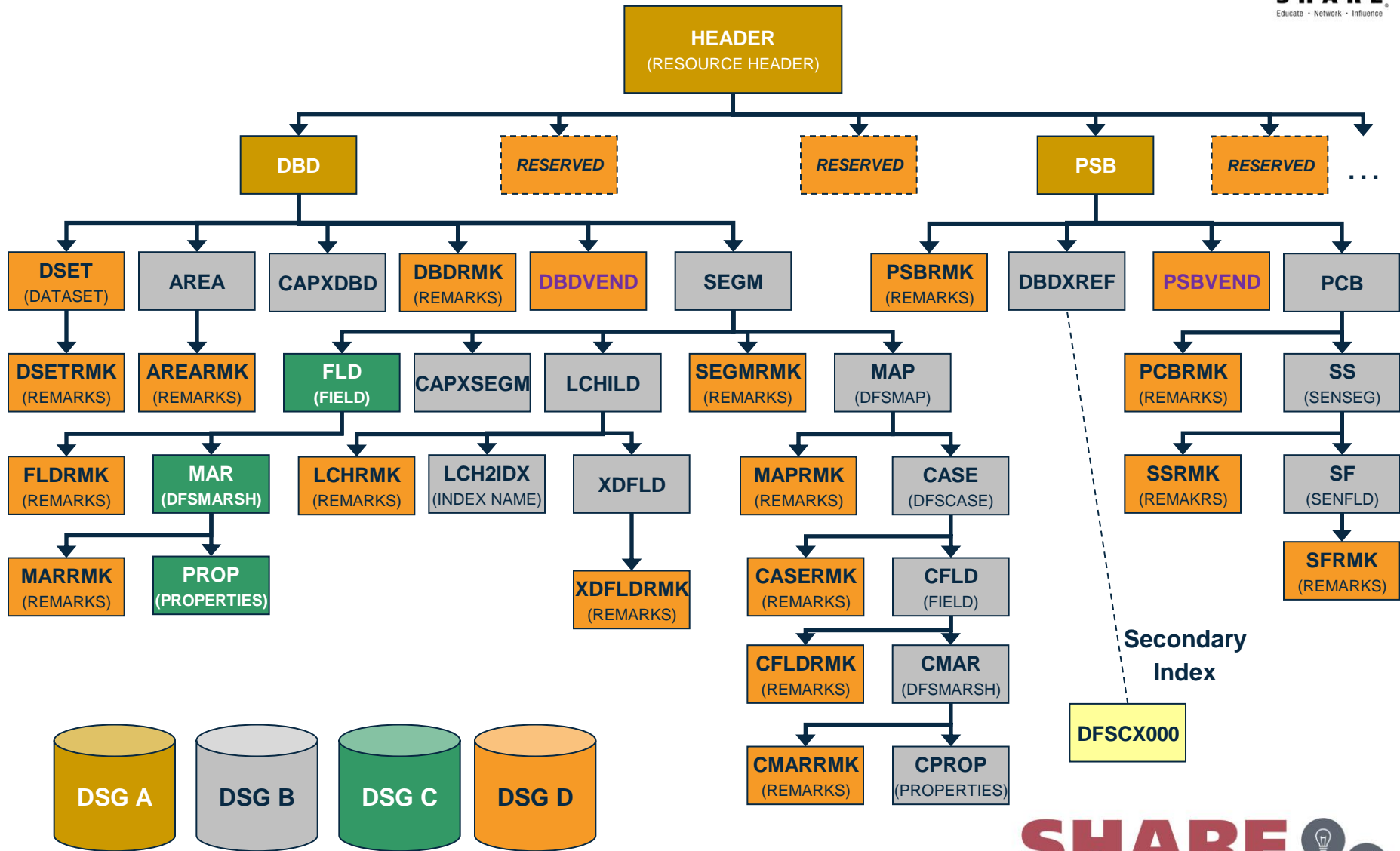
- IMS automatically adds an IMS Catalog PSB to each user PSB at run time once the Catalog is enabled

The Catalog PSB uses 12K in the DLIPSB pool and 500b in the CSAPSB pool for each user using the Catalog PSB. You may need to increase the size of your pools.

- Root segment of the Catalog database is a generic resource header
 - Indicates the type of resource → DBD or PSB
 - A dependent of the Root and its Children are a complete DBD or PSB
 - Allows multiple instances of a resource to be supported
 - Resources are differentiated by their ACBGEN timestamp
 - DBDGEN timestamp is used for Logical DBDs and GSAM
- An “instance” of the IMS catalog metadata is generated each time an ACBGEN is run.
- Catalog database segments typically correspond to macro statements in the DBD and PSB source
 - One segment at the first child level under both the DBD and PSB segments are available for ISV or customer use
 - DBDVEND
 - PSBVEND

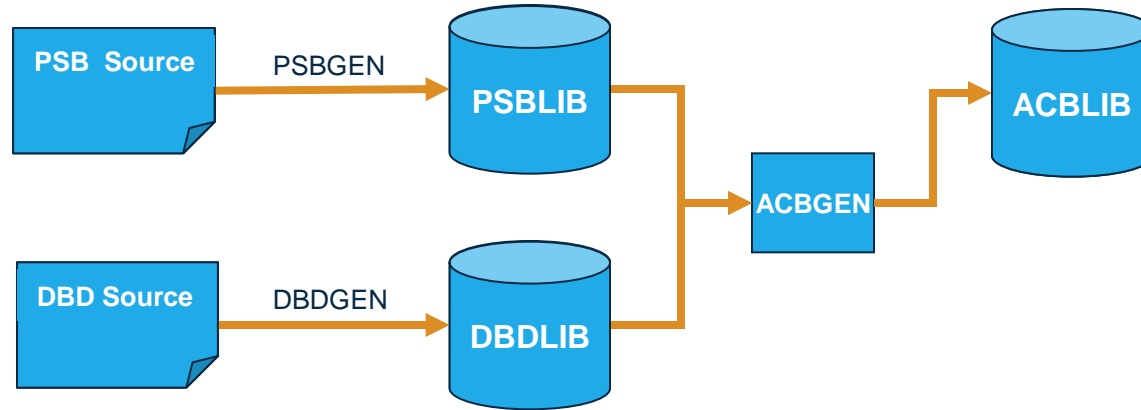
Let's look at the IMS Catalog database physical structure...

IMS Catalog Database Physical Structure



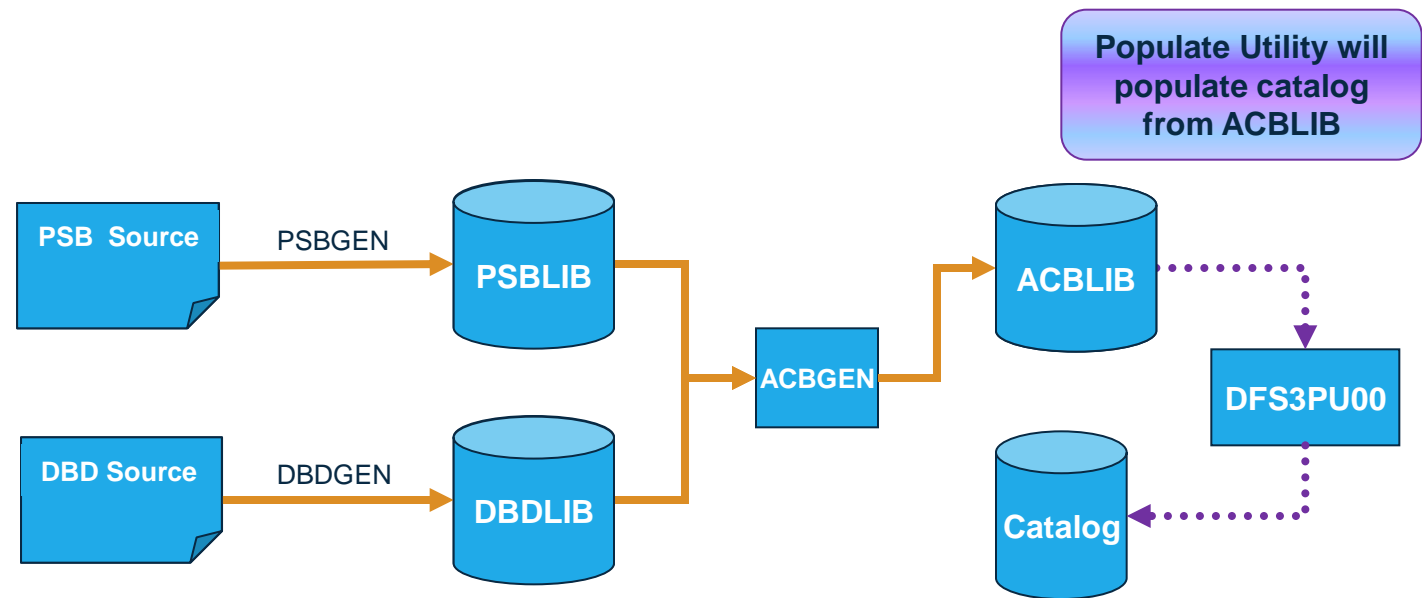
Complete your session evaluations online at www.SHARE.org/Orlando-Eval

Today's ACBGEN



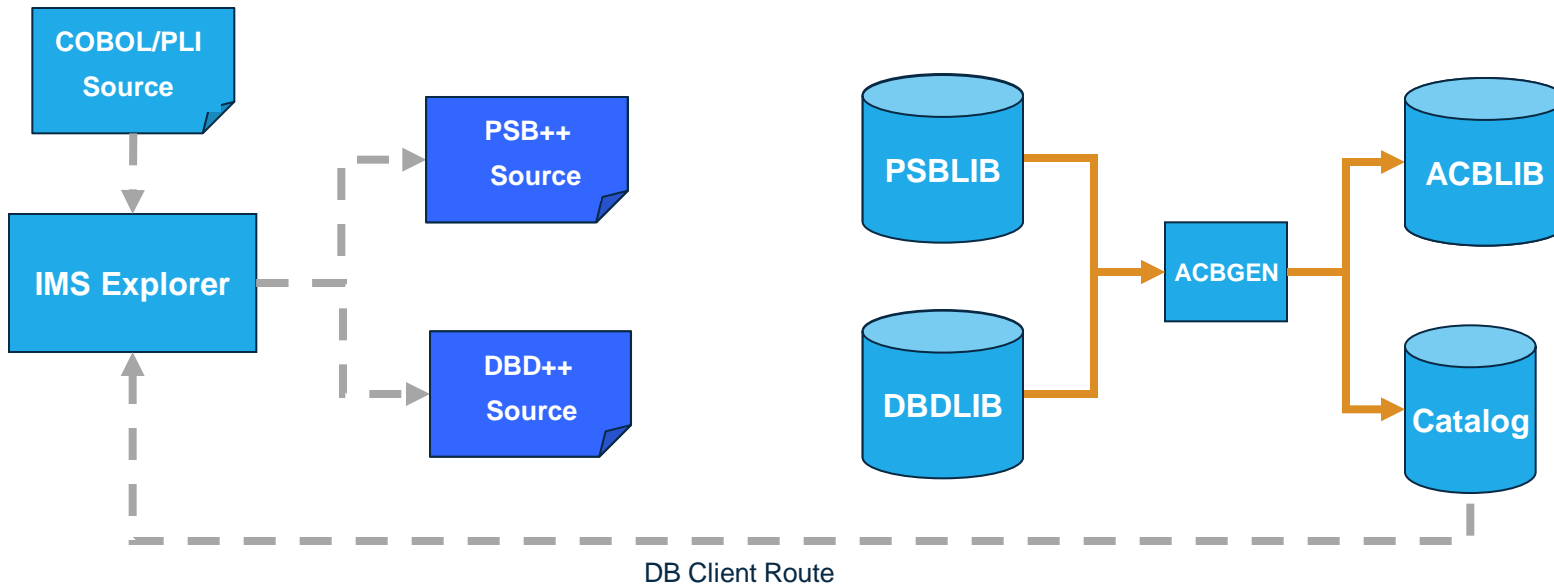
- ACBGEN creates an ACBLIB member with standard ACB info

IMS Catalog Lifecycle – Initial Load



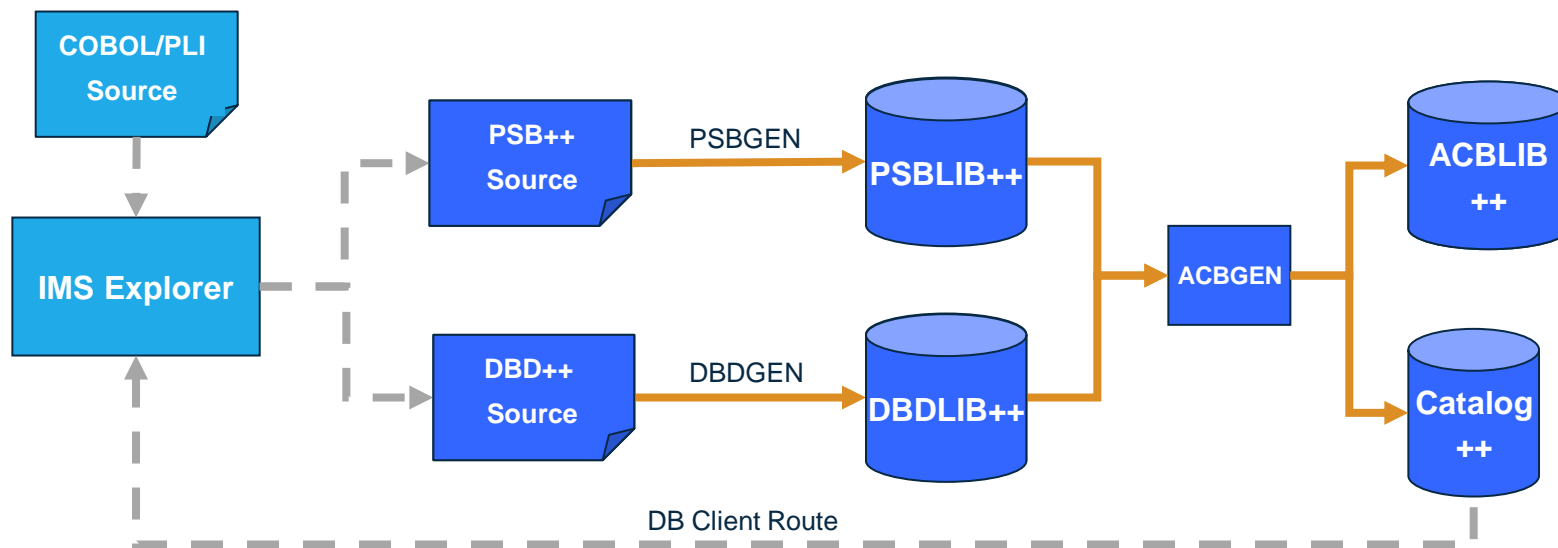
- Catalog Populate Utility populates the IMS catalog initially (done once)
 - Builds and loads the catalog metadata using the ACBLIB members

IMS Catalog Lifecycle – Extended DBD & PSB



- Using IMS Explorer for Development
 - View PSB & DBD information from the Catalog
 - Edit PSB & DBD source code
 - Incorporate application COPYBOOKs / INCLUDEs into DBD source code
 - Verify that PCB names are identified (macro label or PCBNAME=)

IMS Catalog Lifecycle – Extended ACB & Catalog



- PSBGEN and DBDGEN must be executed
- ACBGEN will populate ACBLIB and the Catalog
 - Populates ACBLIB with standard ACB info and extended info
 - Populates the Catalog with extended metadata
- Key points
 - Only way to update the Catalog is via the Populate Utility or new ACBGEN
 - Extended info is acquired via the IMS Explorer
 - Extended info stored in ACBLIB members for recoverability

IMS Catalog for Metadata

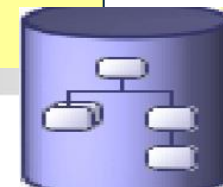
Contains IMS program & database resources, and relevant application metadata

Database Definitions

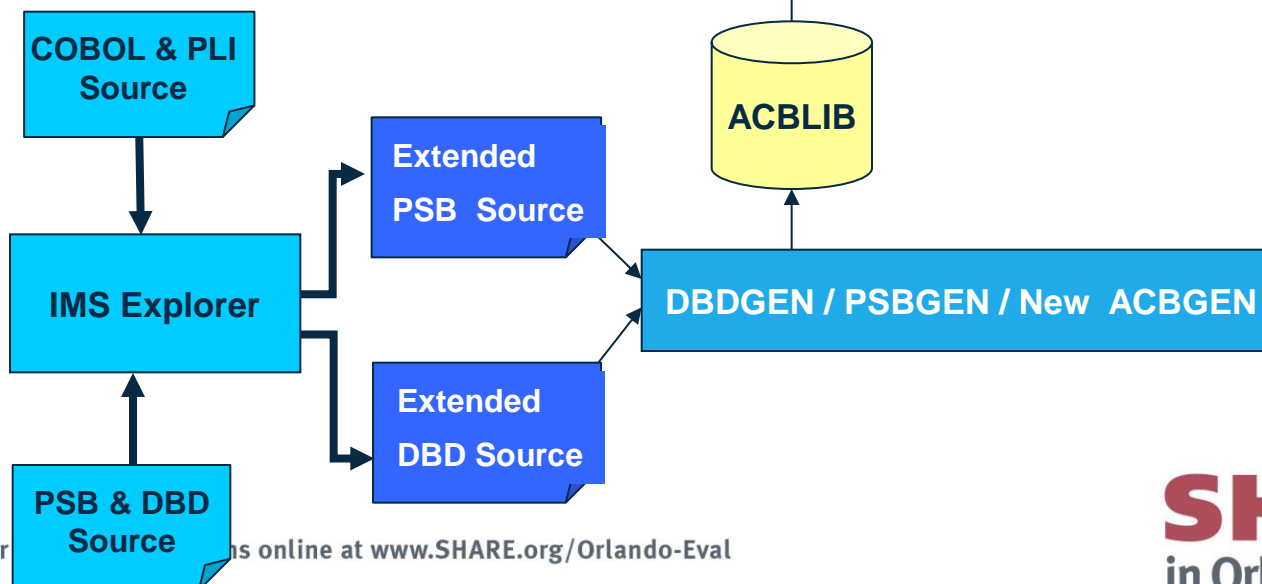
- Segments
- Seq Fields
- Search Fields

Application Program Specifications

- Arrays and Structures
- Field data types and data type conversion
- Field redefines
- Alternative Field maps for a segment

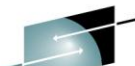


IMS Catalog Metadata



Complete your metadata online at www.SHARE.org/Orlando-Eval

Extended DBD Definition



- **DBD**
 - ENCODING=
 - Specifies default encoding of all character data in the database defined by this DBD
 - Value can be overridden in individual segments or fields
- **SEGM**
 - EXTERNALNAME=
 - Optional alias for NAME= parameter used by Java application programs to refer to segment
- **FIELD**
 - CASENAME=
 - Name of the map case when alternative mappings are defined for fields in a segment
 - DATATYPE=
 - Specifies the external data type of the field
 - EXTERNALNAME=
 - Optional alias for the NAME= parameter used by Java application programs to refer to field
- **DFSMARSH**
 - Specifies the DATATYPE converter routine to transform binary data to external data format
- **DFSMAP**
 - Defines alternative field mappings in a segment
- **DFSCASE**
 - Defines a map case for a segment type that uses DFSMAP alternative field mapping
- **REMARKS=**
 - Optional user comments: 1- to 256-character field added to DBDGEN statement

Extended PSB PCB Definition for AIBTDLI

- **PCB**
 - **EXTERNALNAME=**
 - Optional alias for the PCB label or the PCBNAME= parameter
 - Java application programs use the external name to refer to the PCB
 - **REMARKS=**
 - Optional user comments: 1- to 256-character field added to PSBGEN statements

IMS Catalog Enablement

- Copy the IMS supplied DBDs and PSBs for the IMS Catalog to your DBDLIB and PSBLIB data sets
- Run the existing ACB Maintenance utility to generate the IMS Catalog ACBs
- Activate the ACB library that contains the new IMS Catalog ACBs
- Code the new <SECTION=CATALOG> section in DFSDFxxx PROCLIB member
- Define the Catalog HALDB master and partitions to DBRC
- If you choose, manually allocate the Catalog HALDB data sets, or you can allow IMS to create the Catalog HALDB datasets automatically
- Load the IMS catalog by running the DFS3PU00, Catalog Populate utility
 - Recommended: execute this utility first with the DFSCP000 read PSB to get Catalog HALDB sizing statistics
- Image copy the Catalog database, it will be required !
- Restart IMS

IMS Catalog Enablement

- Add catalog DBDs and PSBs to your DBDLIB, PSBLIB & ACBLIB
 - Copy DBD and PSB object code from SDFSRESL to your DBDLIB and PSBLIB

```
//CPYCMEM EXEC PGM=IEBCOPY
//SDFSRESL DD DSN=SDFSRESL,DISP=SHR
//DBDLIB DD DSN=MYIMS.DBDLIB,DISP=OLD
//PSBLIB DD DSN=MYIMS.PSBLIB,DISP=OLD
//SYSIN DD *
COPY OUTDD=DBDLIB,INDD=((SDFSRESL,R)),LIST=YES
SELECT MEMBER=(DFSCD000,DFSCX000)
COPY OUTDD=PSBLIB,INDD=((SDFSRESL,R)),LIST=YES
SELECT MEMBER=(DFSCPL00,DFSCP000,DFSCP001,DFSCP002,DFSCP003)
```

- ACB GEN the catalog PSB resources into your ACBLIB

```
//CATABCB EXEC PGM=DFSRRRC00,PARM='UPB'
//STEPLIB DD DSN=SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=SDFSRESL,DISP=SHR
//IMS DD DSN=MYIMS.DBDLIB,DISP=SHR
// DD DSN=MYIMS.DBDLIB,DISP=SHR
//IMSACB DD DSN=IMS.ACBLIB,DISP=OLD
//SYSIN DD *
BUILD PSB=(DFSCPL00)
BUILD PSB=(DFSCP001)
BUILD PSB=(DFSCP000)
```

N.B. The MODBLKS resources for the Catalog databases and programs do not need to be defined !

IMS Catalog Enablement

- Modify DFSDFxxx PROCLIB Member
 - New CATALOG section(s) for catalog related parameters
 - Single section format <SECTION=CATALOG>
 - Multiple section format <SECTION=CATALOG*imsid*>
 - Multiple IMS systems sharing one DFSDFxxx PROCLIB member
 - **imsid** suffix must be a four character IMS ID
 - CATALOG section parameters
 - CATALOG=N | Y
 - Catalog is disabled or enabled
 - If enabled, IMS automatically creates catalog DDIR & PDIRs at IMS startup
 - ALIAS=DFSC | xxxx (no default value, use DFSC for no catalog alias)
 - Specifies any 1-4 alphanumeric value used as a Catalog database name prefix
 - Enables use of non-shared, Catalog databases within an IMSplex
 - Use in a data sharing environment where each IMS has its own Catalog database and all are registered in a single set of RECONS
 - At runtime, the alias Catalog database names are dynamically replaced with internal database names **DFSCD000** and **DFSCX000**
 - For standalone IMS system – use “**DFSC**” which is the standard Catalog database name prefix → **DFSCD000** and **DFSCX000**

IMS Catalog Enablement

- DFSDFxxx PROCLIB Member
 - CATALOG section parameters (continued)
 - Information used by Catalog Populate Utility to automatically allocate the Catalog database data sets
 - DATACLAS
 - Optional data class for SMS managed data sets
 - MGMTCLAS
 - Optional management class for SMS managed data sets
 - STORCLAS
 - Required storage class for SMS managed data sets
 - IXVOLSER
 - Volume serial number for primary and secondary catalog indices
 - Required for non-SMS managed data sets
 - SMSVOLCT
 - Number of volumes (1-20) created by the Catalog Populate utility for SMS-managed
 - SPACEALLOC
 - Free space % (0 to 9999) added to the IMS-computed size of the primary & secondary data set allocations

IMS Catalog Enablement

- DFSDFxxx PROCLIB Member
 - CATALOG section parameters (continued)
 - RETENTION=(INSTANCES=2 | nnnnn, DAYS=0 | nnnnn)
 - Specify either parameter or both parameters, max nnnnn value of 65535
 - Specifies retention criteria for the IMS catalog DBD and PSB metadata
 - INSTANCES=2 | nnnnn
 - By default IMS keeps two instances of the DBD and PSB metadata
 - Specifies the number of DBD and PSB instances retained in the IMS catalog before instances can be deleted by utility DFS3PU10
 - When DBD versioning is used, specifies the number of instances retained in the IMS catalog for each DBD version
 - DAYS=0 | nnn
 - By default IMS ignores the age of the DBD and PSB metadata instances
 - Specifies the number of days DBD and PSB instances are retained in the IMS catalog before instances can be deleted by utility DFS3PU10
 - GURCACHE=1 | 999
 - Specifies gigabytes of 64bit storage allocated for GURIN64 pool to cache XML documents generated as GUR call responses

IMS Catalog Enablement

- Definition of the HALDB structure
 - Partitioning of the catalog is the users responsibility
 - Minimum of 1 partition is required
 - Last partition must be able to contain the highest-key PSB record
 - Catalog HALDB uses the high-key selection method
 - No use of Partition Selection Exit allowed
 - Catalog Database Definition
 - For systems that use DBRC
 - Catalog database can be defined to the RECONs with the DBRC utility and commands
 - For systems that do not use DBRC
 - Catalog database must be defined to the Catalog Partition Definition data set using the Catalog Partition Definition Data Set utility, DFS3UCD0
 - If an ALIAS is used in the CATALOG sections of the DFSDfxxx member, each alias Catalog database must be defined

IMS Catalog Enablement

- Using DBRC DSPURX00 utility and commands to define the Catalog database to the RECONS

```
//DEFCAT EXEC PGM=DSPURX00
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//SYSPRINT DD SYSOUT=*
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//SYSIN DD *
INIT.DB DBD(DFSCD000) TYPHALDB SHARELVL(3)
INIT.PART DBD(DFSCD000) PART(DFSCD01) -
  DSNPREFIX(dsnprefix.DFSCD000) -
  BLOCKSIZE(4096) -
  KEYSTRNG(X'FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF')
INIT.DB DBD(DFSCX000) TYPHALDB SHARELVL(3)
INIT.PART DBD(DFSCX000) PART(DFSCX01) -
  DSNPREFIX(dsnprefix.DFSCX000) -
  KEYSTRNG(X'FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF')
/*
```

- If using a catalog alias prefix, replace **DFSC** in the database and partition names for the catalog and the catalog secondary index with the four character ALIAS name prefix
- You might need to define multiple alias name databases to the RECONS

IMS Catalog Enablement

- Using the Catalog Partition Definition Data Set utility, DFS3UCD0, to define the Catalog database (for systems that do not use DBRC)

```
//S1 EXEC PGM=DFS3UCD0,REGION=0M
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSHDBSC DD DSN=...,DISP=
//SYSPRINT DD SYSOUT=*
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//SYSIN DD *
HALDB=(NAME=DFSCD000)
PART=(NAME=DFSCD000,PART=partitionname,
      DSNPREFIX=dsnprefix,
      KEYSTRNG=keystring)
HALDB=(NAME=DFSCX000)
PART=(NAME=DFSCX000,PART=partitionname,
      DSNPREFIX=dsnprefix,
      KEYSTHEX=FFFFFFFFFFFFFFFF)

/*
```

Unregistered
Catalog database

- Catalog Partition Definition data set (//DFSHDBSC DD) is populated with the information specified in the HALDB and PART control cards
 - RECON-like information for catalog database partition definition and structure
- The name **DFSCD000** in the HALDB and PART statements contains the default catalog prefix **DFSC**. If your catalog uses an alias name prefix, substitute it in the JCL

IMS Catalog Enablement

- After Catalog database is defined in Catalog Partition Definition Data Set
 - Identify unregistered Catalog database names
 - UNREGCATLG parameter in the DATABASE section of the DFSDFxxx member

```
/* **** */
/* Database Section */
/* **** */
<SECTION=DATABASE>
UNREGCATLG=(DFSCD000,DFSCX000) /* Unregistered IMS catalog DB */
/* **** */
/* **** */
/* **** */
```

- If using an alias name prefix, replace **DFSC** in the UNREGCATLG database names with the four character alias name prefix
- Limitations of using an unregistered Catalog database
 - NO IMS Data Sharing support
 - NO OLR support
 - NO partition definition change support
 - User must rebuild catalog partitions
 - Manual recovery required for unregistered Catalog databases

IMS Catalog Enablement

- After Catalog database is defined in Catalog Partition Definition Data Set
 - Create a new DFSMDA dynamic allocation member for the Catalog Partition Definition data set

```
//DYNALOC JOB
//*
//STEP EXEC IMSDALOC
//SYSIN DD *
    DFSMDA TYPE=INITIAL
    DFSMDA TYPE=CATDBDEF, DSNAME=dsn
    DFSMDA TYPE=FINAL
    END
/*
```

- **dsn** is the name of the Catalog Partition Definition data set
 - Catalog Partition Definition data set was allocated in the DFS3UCD0 utility on the //DFSHDBSC DD
- Dynamically allocate Catalog Partition Definition data set in any IMS job step

Catalog Populate Utility Statistics Report

CATALOG DFSCD000

PARTITION DFSCD01

NUMBER OF SEGMENTS INSERTED INTO THE CATALOG

SC	SEGMENT	INSERTED SEGMENTS	DSG	PARENT	AVERAGE SEGS/PARENT
1	HEADER	4228	A		
2	DBD	2530	A	HEADER	0.6
3	CAPXDBD	7	D	DBD	0.0
5	DSET	2599	D	DBD	1.0
7	AREA	139	D	DBD	0.1
9	SEGM	16337	B	DBD	6.5
10	CAPXSEGM	1	D	SEGM	0.0
12	FLD	16426	C	SEGM	1.0
14	MAR	16426	C	FLD	1.0
17	LCHILD	2687	B	SEGM	0.2
20	XDFLD	134	B	LCHILD	0.0
33	PSB	1840	A	HEADER	0.4
35	PCB	9190	B	PSB	5.0
37	SS	75274	B	PCB	8.2
39	SF	1105	B	SS	0.0
41	DBDXREF	8886	D	PSB	4.8

SEGMENT	WITHIN EXISTING HEADER	DUPLICATES NOT INSERTED
DBD	71	0
PSB	72	0

ESTIMATED SPACE REQUIREMENT TO HOLD INSERTED SEGMENTS		
DSG	BLKSIZE	BLOCKS
A	4096	596
B	4096	9343
C	4096	8214
D	4096	236

DSG	RECORDS
L	8886
X	4230

SECONDARY INDEX	RECORDS
DFSCX000	8886

Consider using 8K
Blocksize for the
Catalog DB

N.B. Run the Catalog Populate Utility, DFS3PU00, with the READ PSB, DFSCP000, to get Catalog sizing statistics before you actually run it with the LOAD PSB, DFSCPL00

IMS Catalog Populate Utility

- Populate the IMS Catalog
 - Load the IMS Catalog using IMS Catalog Populate utility, DFS3PU00
 - Each ACB member is decoded, converted to catalog format, loaded into the catalog
 - Reads ACBLIB, DBDLIB and PSBLIB datasets as input
 - Data sets can be concatenated but only first occurrence of an ACB member is used
 - DBDLIB needed for Logical databases and GSAM databases
 - PSBLIB needed to determine which GSAM databases go into the catalog
 - Catalog database DBD and PSB segments will have a version and contain a timestamp that matches the ACB member timestamp
 - Used to associate an ACB member with a catalog member
 - Timestamp exceptions
 - DBDGEN timestamp for Logical and GSAM DBs
 - PSBGEN timestamp for GSAM only PSBs

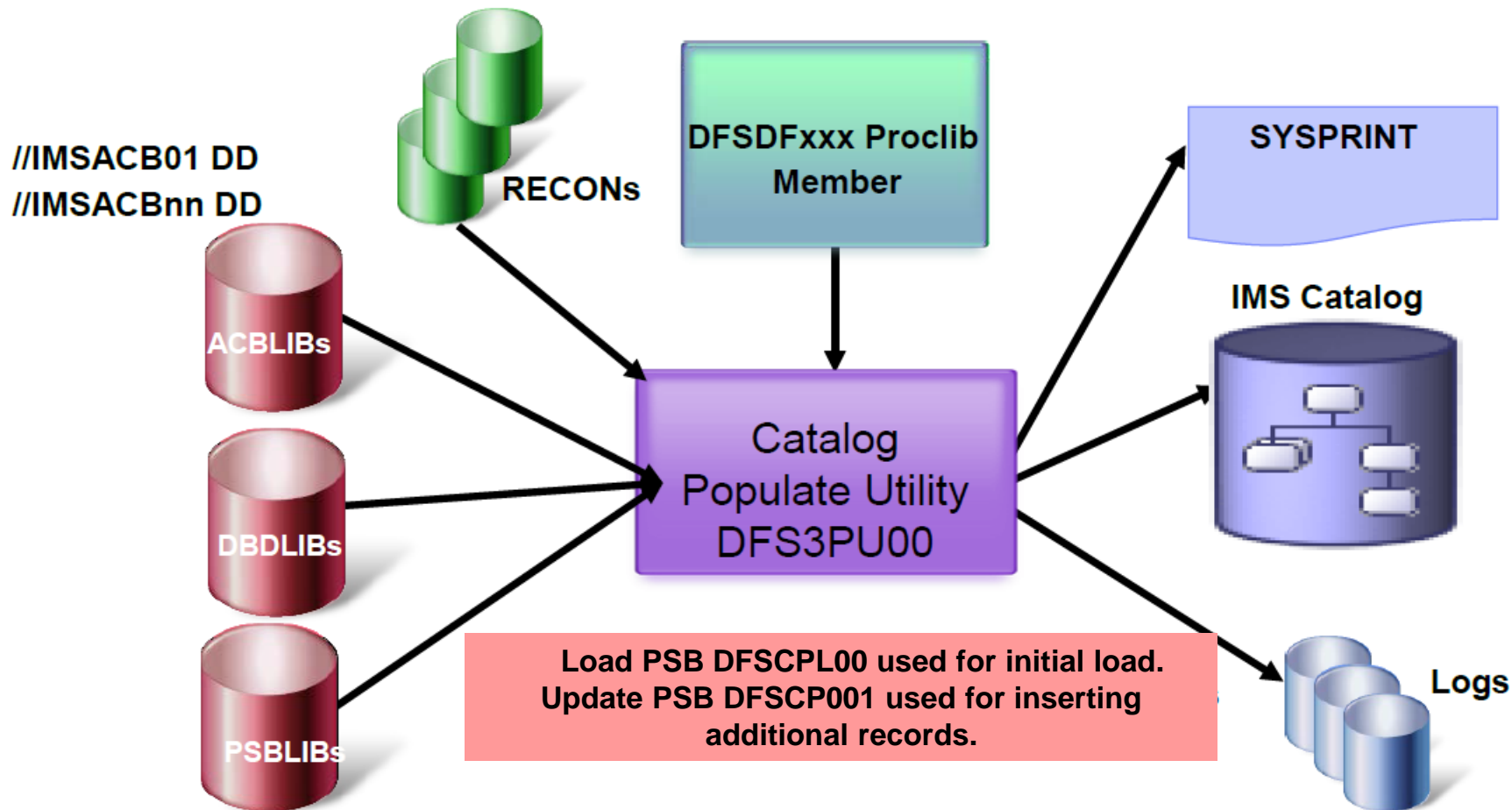
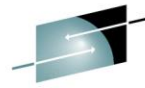
IMS Catalog Populate Utility

- Populate the IMS Catalog (continued)
 - If any of the database data sets do not exist,
 - DFS3PU00 creates them automatically - the DFSCD000 database data sets are:
 - Four data set group data sets for the IMS catalog segments
 - The primary index data set
 - The indirect list data set (ILDS)
 - DFSCX000 secondary index data set
 - DFS3PU00 calculates the size of the database data sets based on both the size of the ACB library and the expansion percentages that you can specify in the DFSDfxxx PROCLIB member
 - The DFSDfxxx member is also where you specify the volume serial number for the VSAM key-sequenced data sets or the SMS storage class, data class, and management class for all data sets

IMS Catalog Populate Utility

- Populate the IMS Catalog (continued)
 - The DFS3PU00, Catalog Populate utility, can run as a IMS Batch or BMP job
 - Requires IMS logs for backout / recovery
 - Requires IRLM if catalog is shared and catalog active in an IMS subsystem
 - Business as usual for data sharing
 - Requires DBRC if the Catalog HALDB is defined in the RECON
 - If using Catalog Partition Definition Data Set
 - User's responsible to ensure online catalog access has ceased
 - Business as usual for non-registered full function DB

IMS Catalog Populate Utility

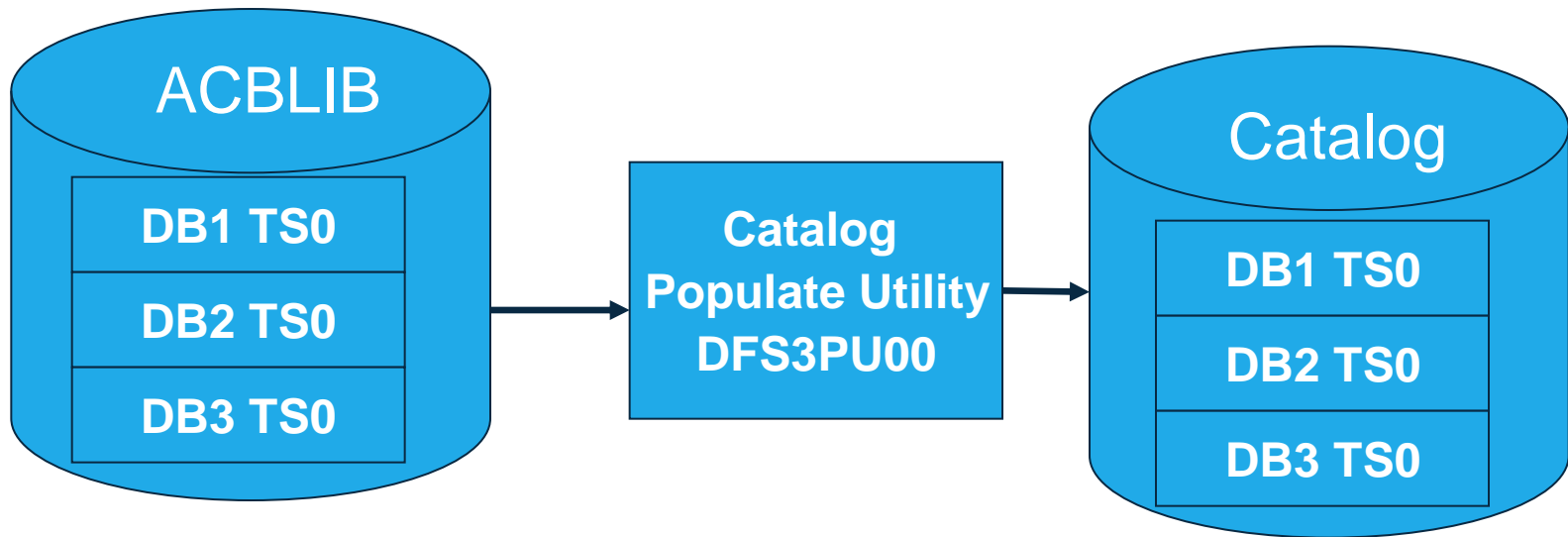


Take an Image Copy of the Catalog database after the initial populate!

(Required if using DBRC)

You must restart the IMS system !!!

Post Catalog Populate Utility

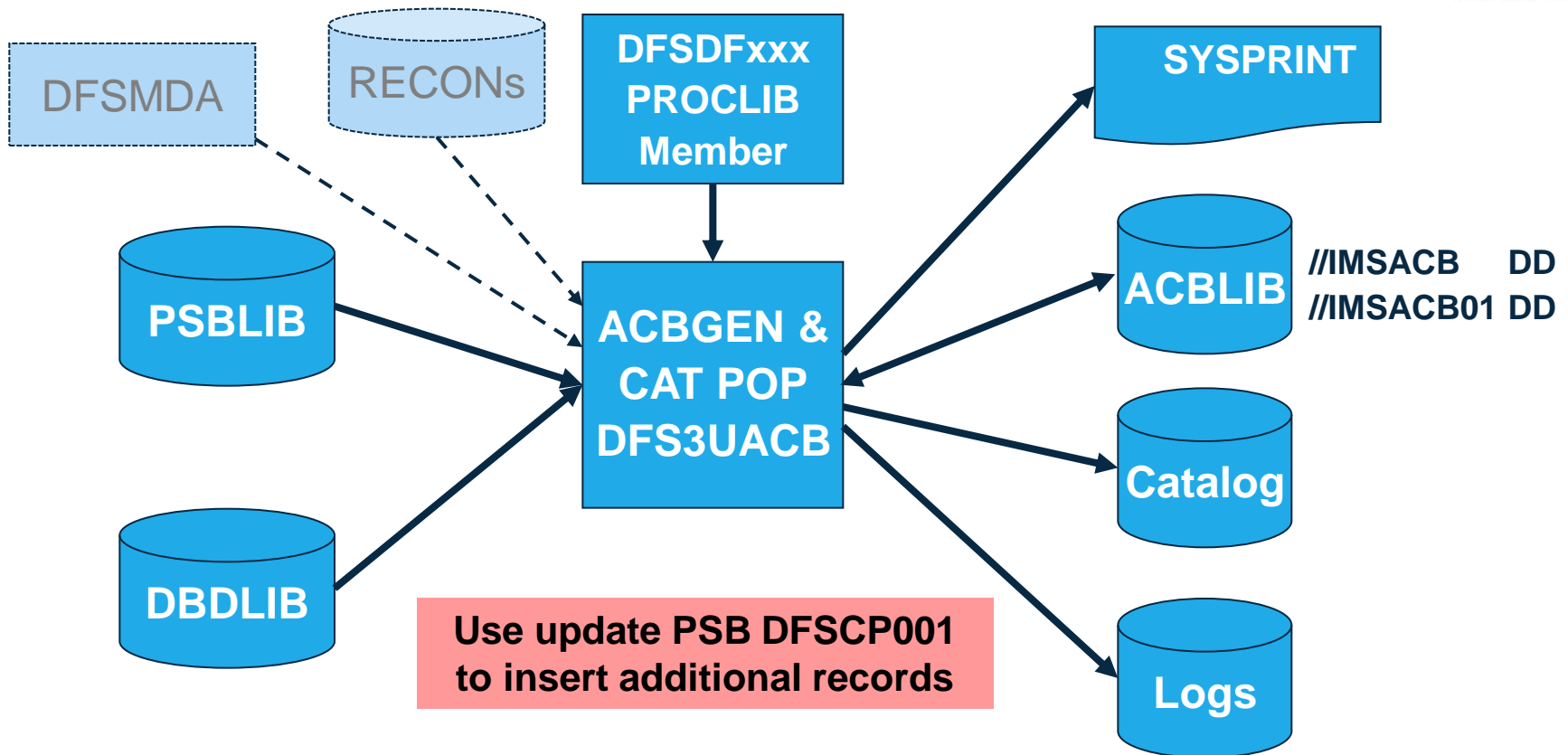


- Start with 3 members in an ACBLIB
- Run the Catalog Populate Utility, **DFS3PU00**
- DBD & PSB catalog records created will have timestamp TS0
- This is the ACB member timestamp

Adding IMS Catalog Metadata

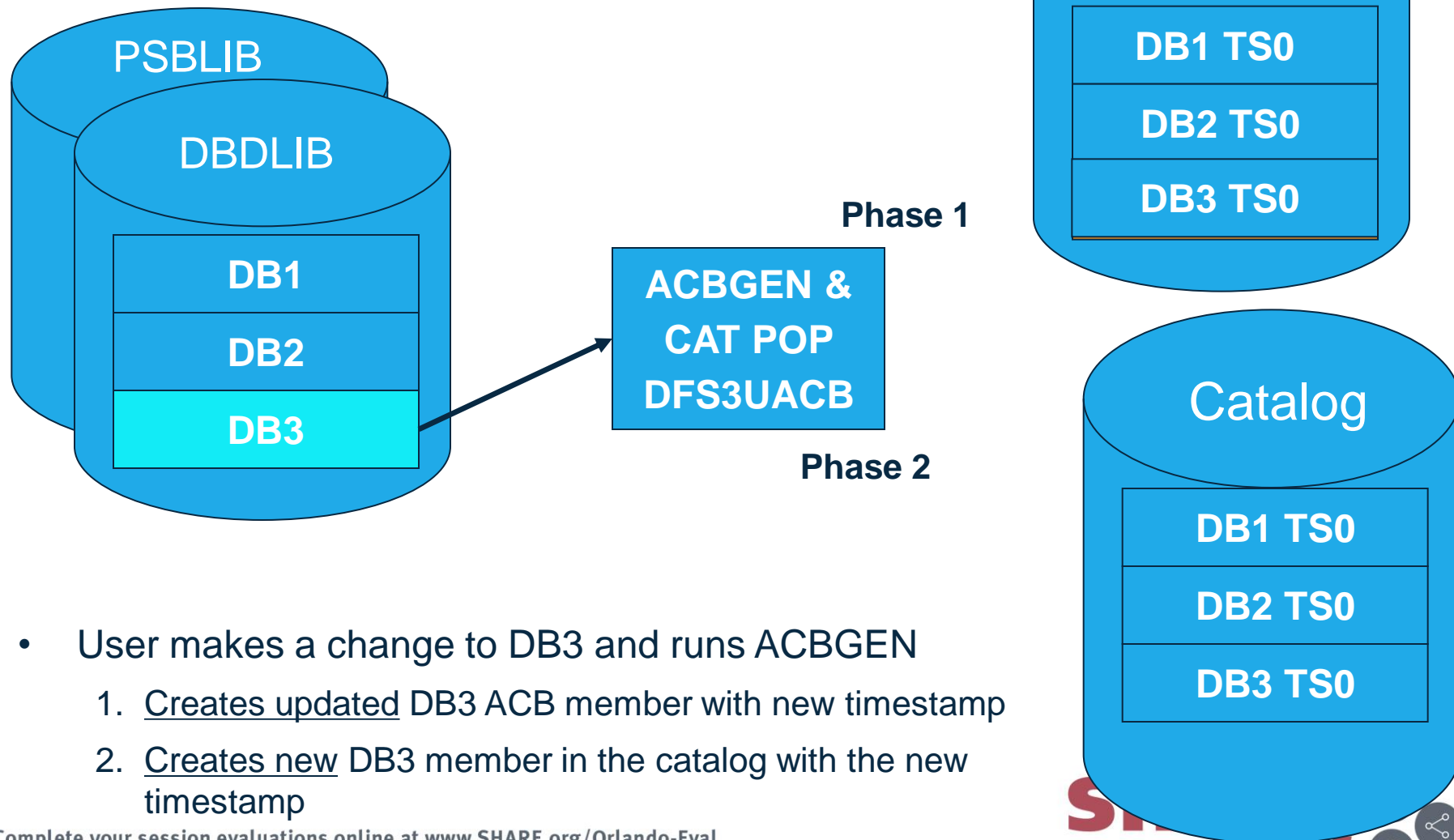
- New ACB Generation and Catalog Populate utility, **DFS3UACB**
 - Replaces existing ACBGEN Utility, DFSUACB0, if IMS catalog enabled
 - Generate ACBLIB member and create catalog metadata in a single job step
 - Phase 1 - ACBGEN
 - DBDLIB and PSBLIB members used as input
 - Validation is unchanged
 - ACB member is written to ACBLIB with new ACBGEN timestamp
 - Phase 2 – IMS catalog update
 - Generated ACB is decoded, converted to catalog segments, inserted into catalog
 - DBD and PSB records containing metadata created and inserted
 - Corresponding ACB member timestamp saved as timestamp in catalog DBD and PSB segments - ensures validity and consistency of ACBLIB and catalog
- New ACBGEN (DFS3UACB) and new Catalog Populate (DFS3PU00) utilities are the only updaters of the IMS Catalog
 - IMS online and IMS batch regions will never update catalog data
 - IMS online and IMS batch regions will only retrieve data from the catalog

New IMS ACBGEN



- New integrated ACBGEN process includes updates to the IMS catalog
- DFSDfxxx PROCLIB member has the catalog information
- DFSMDA member used to dynamically allocate the catalog datasets if not registered to DBRC

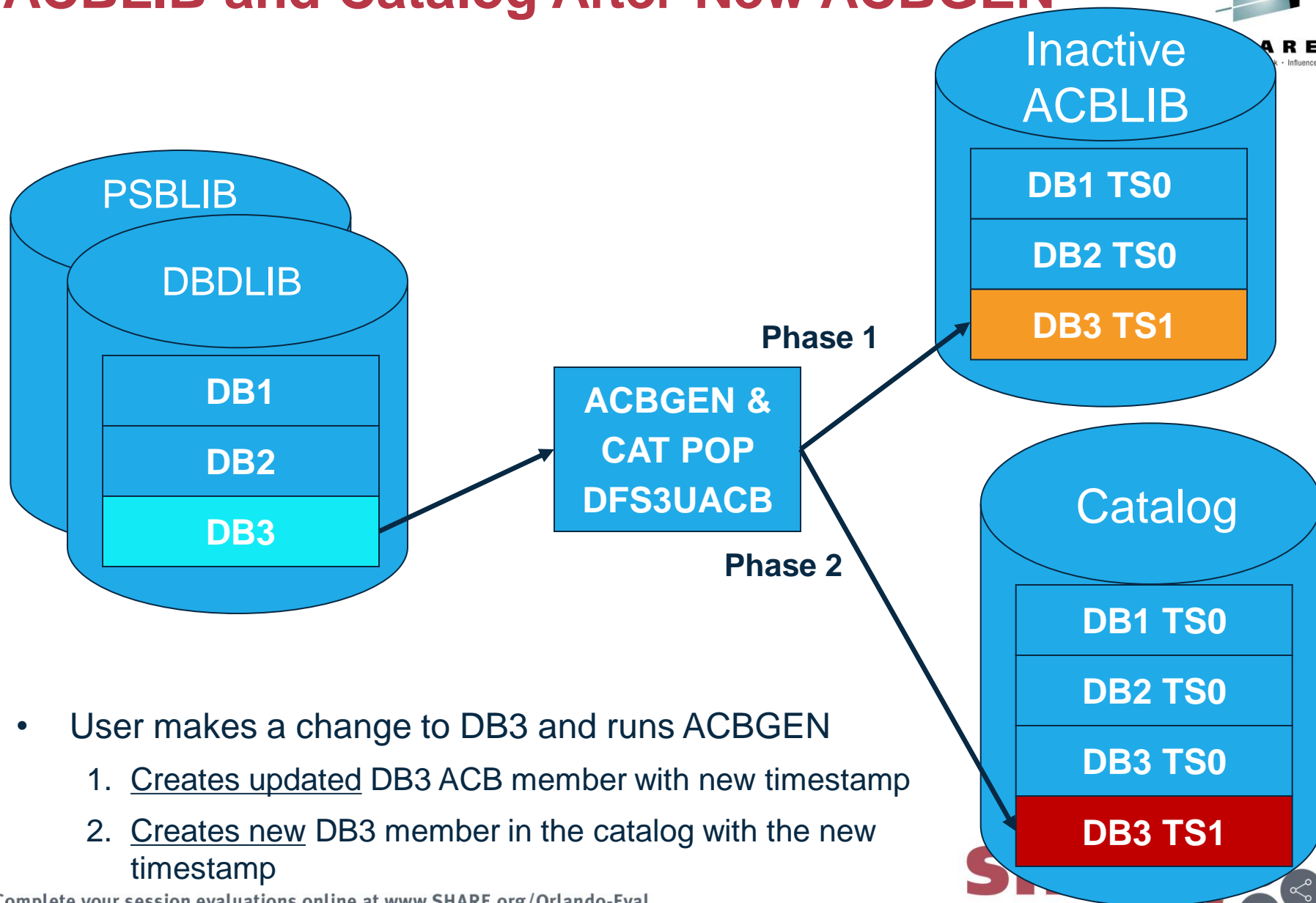
ACBLIB and Catalog After New ACBGEN



- User makes a change to DB3 and runs ACBGEN
 1. Creates updated DB3 ACB member with new timestamp
 2. Creates new DB3 member in the catalog with the new timestamp

Complete your session evaluations online at www.SHARE.org/Orlando-Eval

ACBLIB and Catalog After New ACBGEN

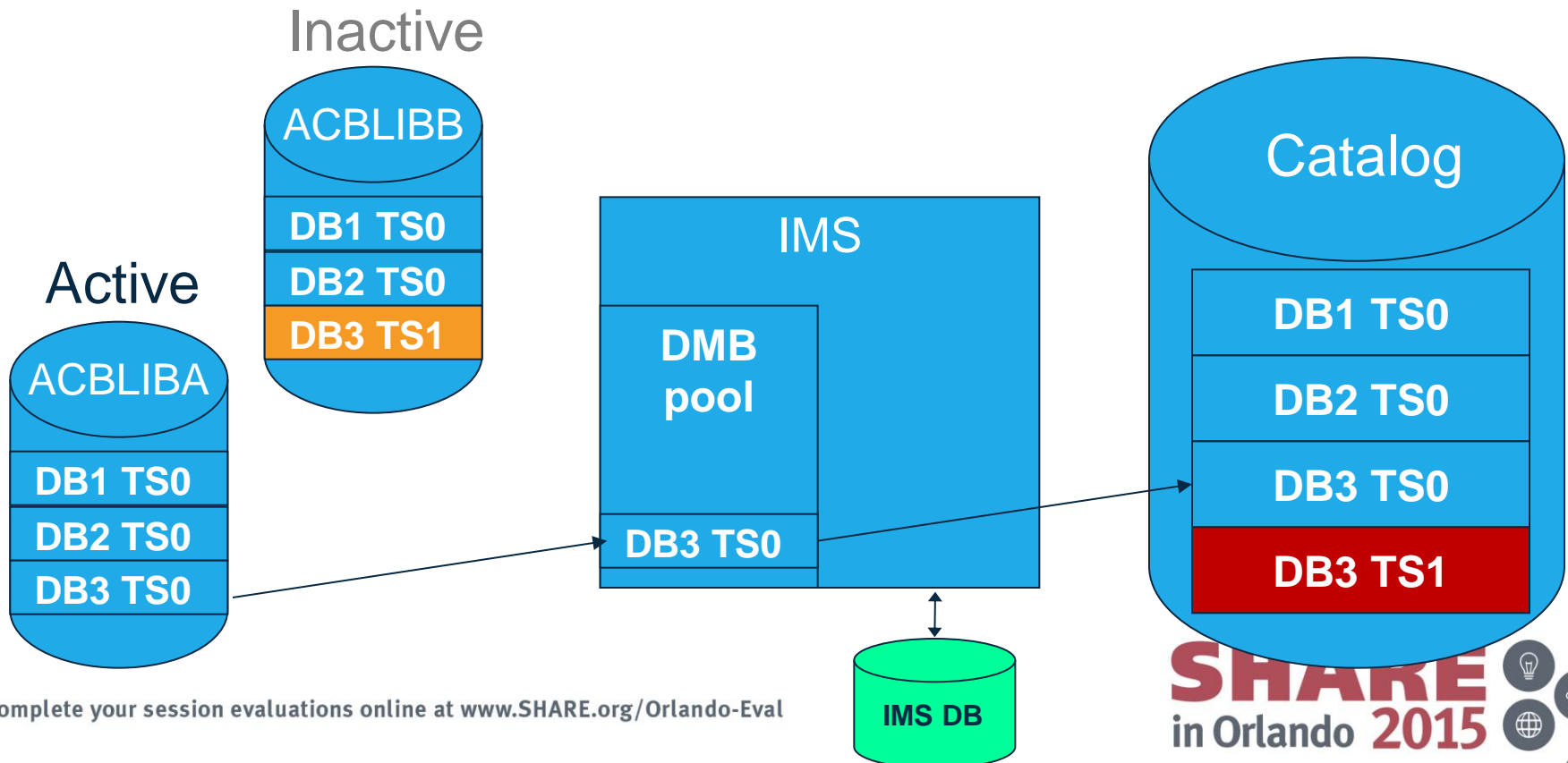


- User makes a change to DB3 and runs ACBGEN
 1. Creates updated DB3 ACB member with new timestamp
 2. Creates new DB3 member in the catalog with the new timestamp

Complete your session evaluations online at www.SHARE.org/Orlando-Eval

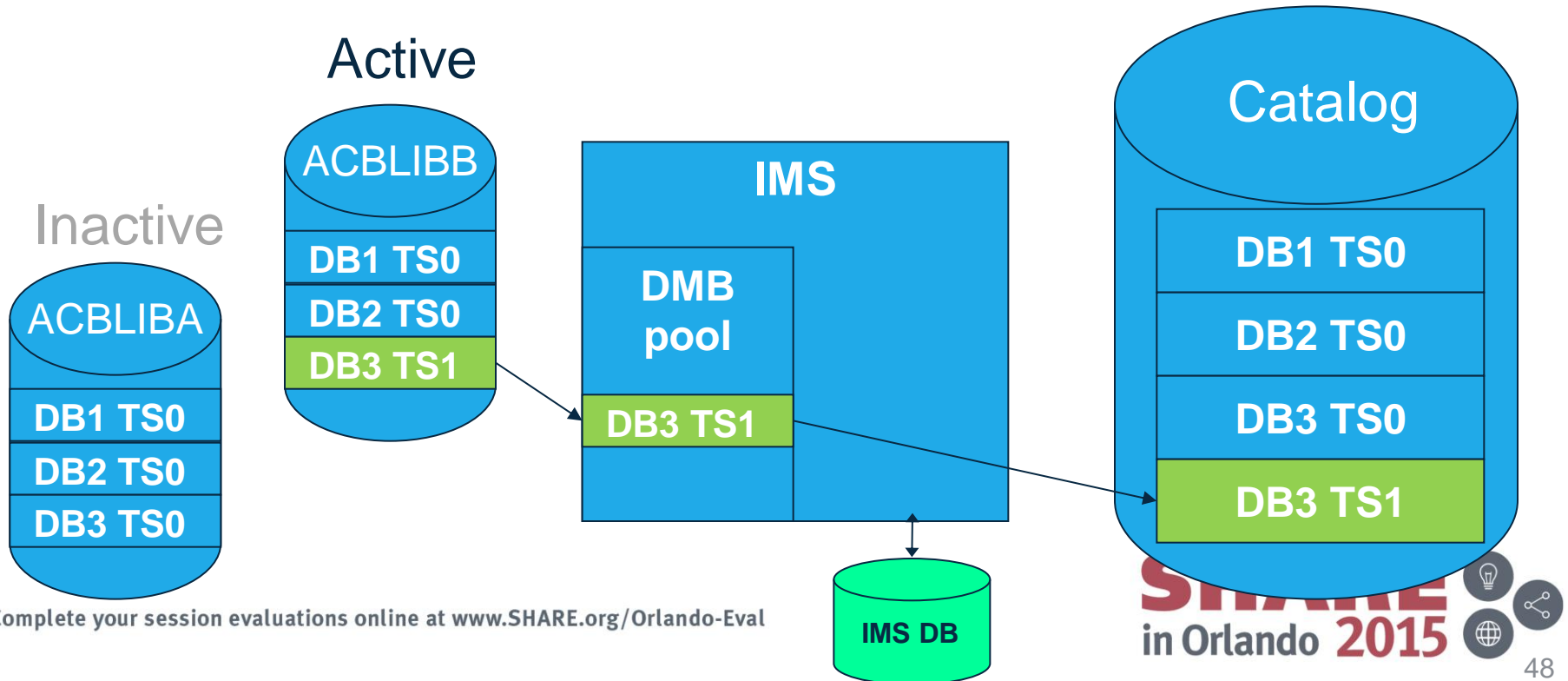
Catalog Timestamps

- Application request is made to read DB3
 1. IMS determines active DB3 member has timestamp TS0
 2. Internal DL/I call issued to retrieve member DB3 from IMS catalog
 3. IMS retrieves metadata from catalog member DB3 with timestamp TS0



Catalog Timestamps

- Initiate OLC to switch from ACBLIBA to ACBLIBB
 - Activates DB3 ACB with timestamp TS1
- Application request is made to read DB3
 1. Internal DL/I call issued to retrieve member DB3 from IMS Catalog
 2. IMS determines active DB3 member has timestamp TS1
 3. IMS retrieves Catalog member DB3 with timestamp TS1



Utilities for the IMS Catalog

- **IMS Catalog Populate utility (DFS3PU00)**
 - Loads or inserts records into the IMS catalog
 - Estimates the size of the IMS catalog data sets
- **ACB Generation and Catalog Populate utility (DFS3UACB)**
 - Generates the ACBLIB members and loads the IMS catalog metadata in one step
- **IMS Catalog Record Purge utility (DFS3PU10)**
 - Removes unnecessary versions of IMS catalog records from the IMS catalog database based on criteria that you specify
- **IMS Catalog Alias Names utility (DFS3ALI0)**
 - Defines a complete list of catalog aliases to IMS
- **IMS Catalog Copy utility (DFS3CCE0, DFS3CCI0)**
 - Copies IMS catalog database definitions and metadata between IMS systems
 - Migrate an IMS catalog database from a test to production environment
 - Copy a production IMS catalog database to another installation for system cloning or DR purposes (etc...)
- **IMS Catalog Partition Definition Data Set utility (DFS3UCD0)**
 - When DBRC is not used to manage the catalog database:
 - Creates the IMS catalog partition definition data set and populates it with information about the IMS catalog HALDB partitions

IMS Catalog Cleanup with Purge Utility

- Run the Catalog Purge Utility DFS3PU10 to remove Catalog entries

```
//BATCH EXEC PGM=DFSRRRC00,  
// PARM=(DLI,DFS3PU10,DFSCP001,,,,,,,,,,,,,IMB1,,Y,Y,IRLB,,,,,,,,,,,,,DFSDF=0B1')  
//STEPLIB DD DSN=IMSCFG.IMB1.SDFSRESL,DISP=SHR  
// DD DSN=IMSCFG.IMSB.USERLIB,DISP=SHR  
//DFSRESLB DD DSN=IMSCFG.IMB1.SDFSRESL,DISP=SHR  
//PROCLIB DD DSN=IMSCFG.IMSB.PROCLIB,DISP=SHR  
//IMS DD DSN=IMSCFG.IMSB.PSBLIB,DISP=SHR  
// DD DSN=IMSCFG.IMSB.DBDLIB,DISP=SHR  
//SYSUT1 DD DSN=IMSCFG.IMSB.CATPURG,DISP=OLD  
//IEFRDER DD DSN=&&IMSLOG1,DISP=(NEW,DELETE),  
// DCB=(RECFM=VB,LRECL=4092,BLKSIZE=4096,BUFNO=5),  
// SPACE=(CYL,(200,75)),UNIT=SYSDA  
//IEFRDER2 DD DSN=&&IMSLOG2,DISP=(NEW,DELETE),  
// SPACE=(CYL,(200,75)),UNIT=SYSDA  
//DFSVSAMP DD DSN=IMSCFG.IMSB.PROCLIB(DFSVSMB),DISP=SHR  
//SYSPRINT DD SYSOUT=*  
//SYSIN DD *  
MODE BOTH  
//*
```

- The Catalog Purge Utility **will not** delete DBDs and PSBs from the Catalog automatically

Existing Database Utility Support

- HALDB Index/ILDS Rebuild utility (DFSPREC0)
- HD Reorganization Unload utility (DFSURGU0)
- HD Reorganization Reload utility (DFSURGL0)
- Batch Backout utility (DFSBBO00)
- Database Recovery utility (DFSURDB0)
- Database Image Copy utility (DFSUDMP0 / DFSUDMT0)
 - Only batch image copies for an unregistered Catalog

Application Programming with the IMS Catalog

SHARE
Educate • Network • Influence

- IMS catalog PSBs resident for application programs
 - DFSCP000 - High-level assembler and COBOL applications
 - DFSCP002 - PL/I applications
 - DFSCP003 - PASCAL applications
- PCBs included to support different catalog processing models:
 - DFSCAT00 - Primary PCB to access all data in the catalog database
 - DFSCATSX – Used access the catalog database via the catalog 2nd index
 - DFSCATX0 – Used to access the catalog 2nd index as a database
- All catalog processing is performed with PROCOPT=G
- New “GUR” DL/I call to access Catalog DB DBD and PSB records
 - Application programs can use the GUR call

New “GUR” DL/I Call

- Get Unique Record
 - Retrieves an XML formatted Catalog database “record”
 - Restricted to use with the IMS Catalog database
- Functions like a GU followed by a series of GNP calls
 - Less overhead than issuing GU & GNP to retrieve all the metadata for a catalog member
- Returns the entire database DBD or PSB record in one call
 - Using an AIB token, the call can be continued if the I/O area is too small for the entire catalog database record
- Data returned in XML format
 - The XML schemas are included in the IMS.ADFSSMPL data set:
 - DFS3XDBD.xsd (for DBD records)
 - DFS3XPSB.xsd (for PSB records)
 - Support for DFSDDLTO and IMS REXX

IMS Batch Processing with the Catalog

- Enabling the IMS catalog for IMS batch processing
 - Specify DFSDFxxx member on job EXEC parm
 - Requires JCL change to implement
- Exit Routine (DFS3CDX0)
 - Optional Catalog Definition user exit routine
 - Alternative to specifying DFSDFxxx member through job JCL
 - Available if users cannot or choose not to modify job JCL



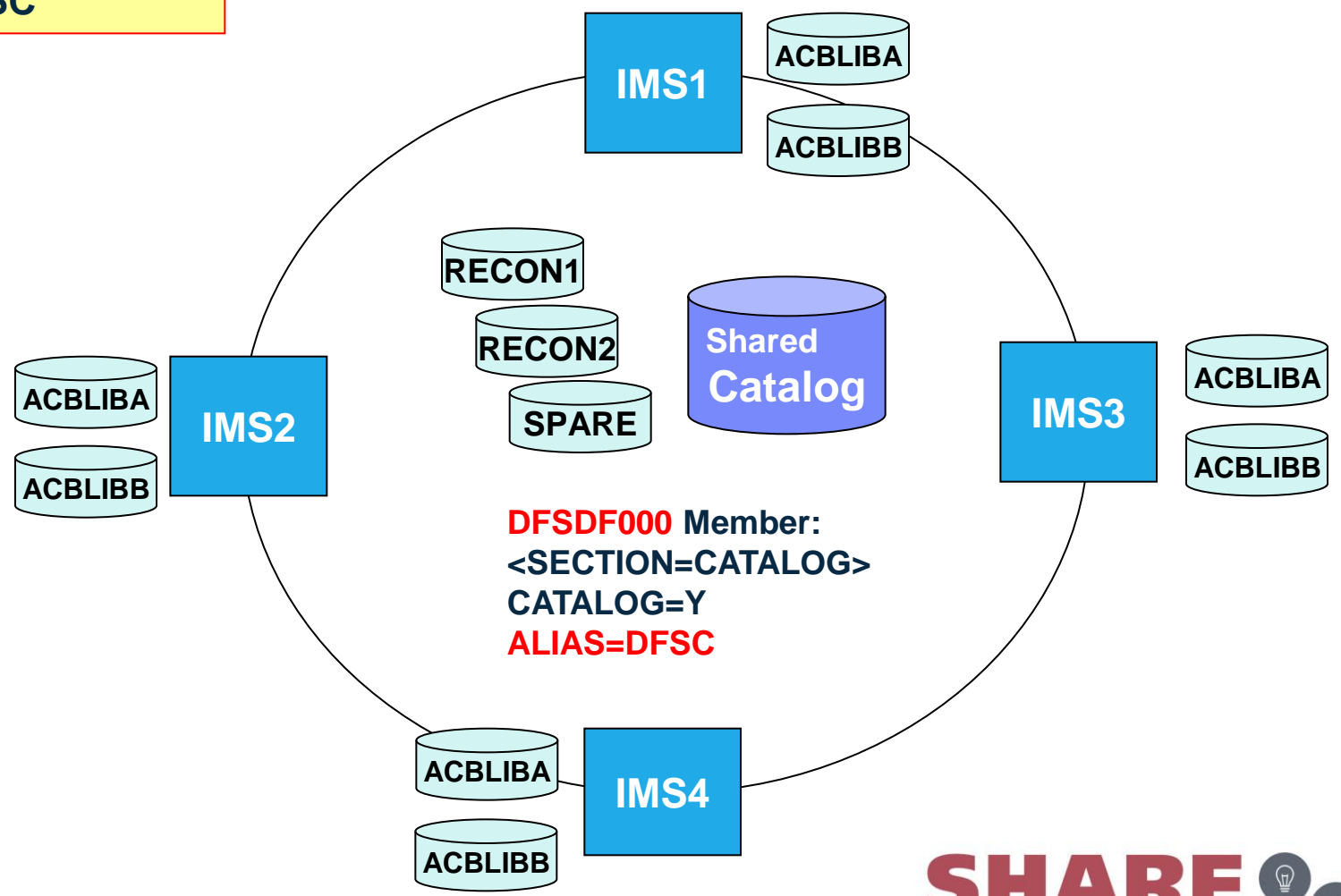
Chapter 9. IMS catalog definition and tailoring

The IMS catalog is an optional system database that stores metadata about your databases and applications. As a system database, IMS performs many of the tasks required to define the catalog for you. If you need greater control, you can perform some of these tasks yourself.

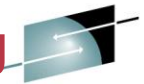
The following topics provide the information you need to define and enable the IMS catalog.

Multiple IMSES, Cloned ACBLIBs, Shared Catalog

Shared CATALOG DB
Shared DFSDFxxx
ALIAS=DFSC



Multiple IMSes, Shared ACBLIBs, each IMS has it's own Catalog



SHARE
Educate • Network • Influence

Non-Shared CATALOG DBs
Shared DFSDFxxx
Separate <SECTION=CATALOGimsid>
ALIAS=xxxx

DFSDF000 Member:

<SECTION=CATALOGIMS1>

CATALOG=Y

ALIAS=CAT1

<SECTION=CATALOGIMS2>

CATALOG=Y

ALIAS=CAT2

<SECTION=CATALOGIMS3>

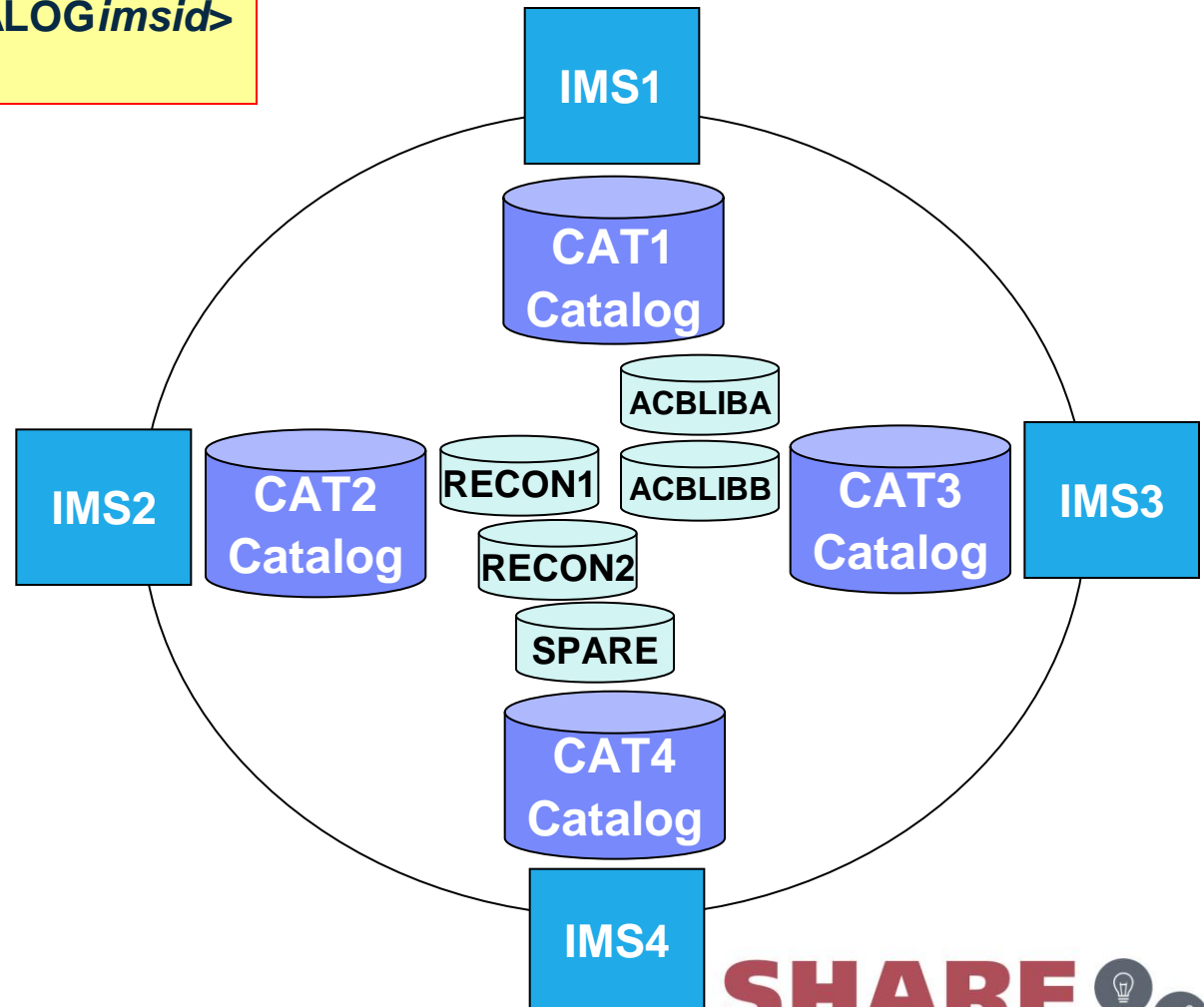
CATALOG=Y

ALIAS=CAT3

<SECTION=CATALOGIMS4>

CATALOG=Y

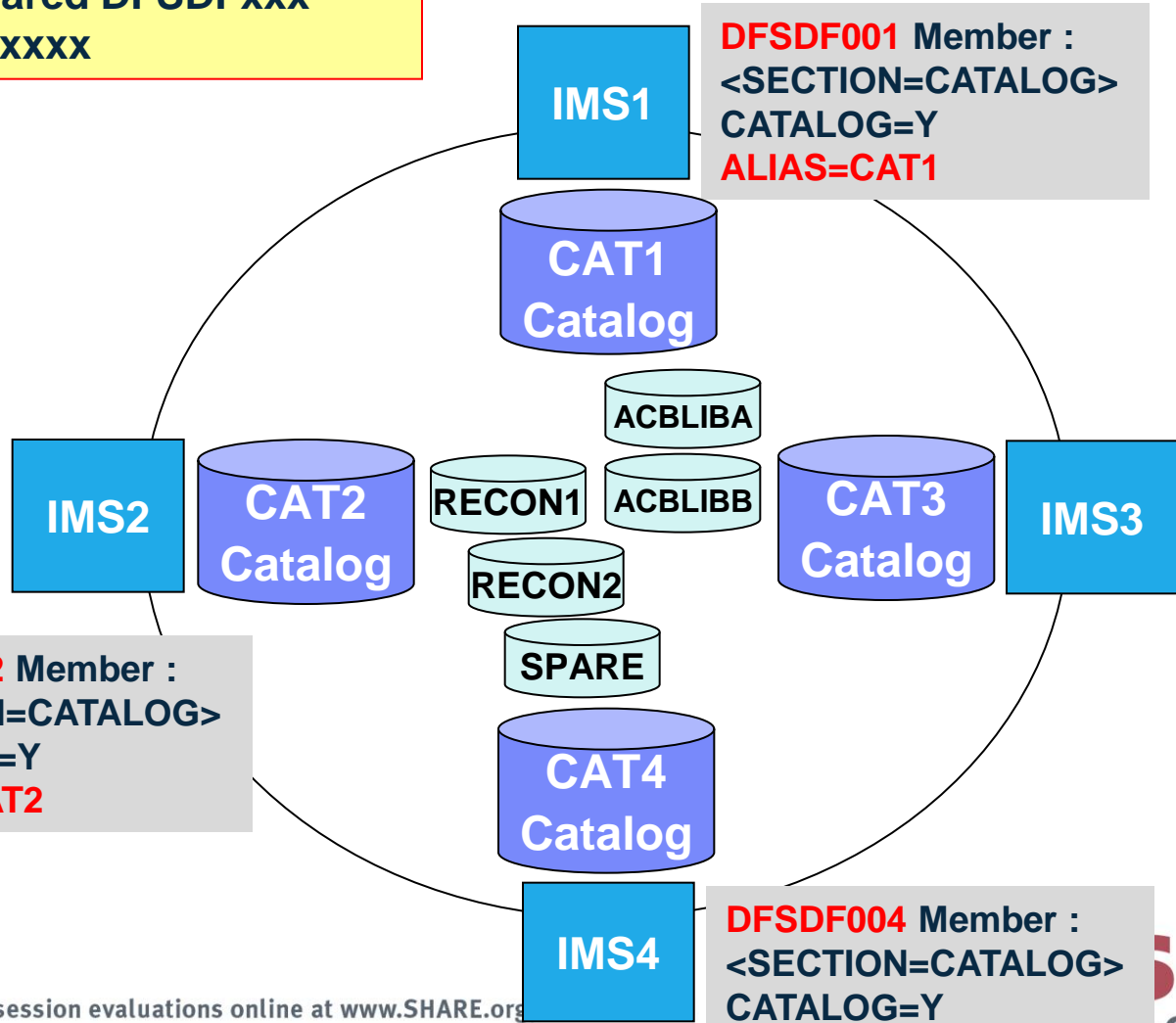
ALIAS=CAT4



Complete your session evaluations online at www.SHARE.org/Orlando-Eval

Multiple IMSes, Shared ACBLIBs, each IMS has it's own Catalog

Non-Shared CATALOG DBs
Non-Shared DFSDFxxx
ALIAS=xxxx



DFSDF001 Member :
<SECTION=CATALOG>
CATALOG=Y
ALIAS=CAT1

DFSDF002 Member :
<SECTION=CATALOG>
CATALOG=Y
ALIAS=CAT2

DFSDF003 Member :
<SECTION=CATALOG>
CATALOG=Y
ALIAS=CAT3

DFSDF004 Member :
<SECTION=CATALOG>
CATALOG=Y
ALIAS=CAT4

Some IMS Catalog Considerations

- Make sure every one is on-board
 - Work as a team, Sysprogs, DBAs and Developers, to establish an implementation plan
 - Communicate the plan and revisions
- Think about, discuss and develop
 - Cleaning up the ACBLIB before you start
 - Procedure for maintaining DBD, PSB and COPYBOOK source
 - Process for rolling into production new DBDs and PSBs
 - Management and cleanup of Catalog DBD and PSB instances
 - Access to objects - tooling makes administration easy
 - Procedure for auditing changes
 - Changes made without visibility or created with wrong ACBGEN
 - Decide who will run the ACBGENs
 - Who builds and maintains the IMS catalog HALDB database
 - Partitioning / Repartitioning
 - OSAM Blocksize
 - Sizing the Catalog depends on number and size of DBDs and PSBs
 - DB Performance Tuning
 - Care and Feeding (ie. Maintenance)



Some IMS Catalog Considerations

- The IMS Catalog is a HALDB Database
 - Catalog database management is required
 - Review/adjust database buffer pool definitions
 - Perform routine management and maintenance on the Catalog database
 - Image Copy, Pointer Checker, Reorg, etc...
 - Catalog database will need to be REORG'd
 - If Catalog database is defined to the RECONs
 - » HALDB OLR non-disruptive reorganization is supported
 - If Catalog database is not defined to the RECONs
 - » HALDB OLR can't be supported and a reorg utility must be used
 - Support with existing backup and recovery procedures
 - Image copy, recovery, backout utilities etc...
 - If Catalog database is not defined to the RECONs recovery is limited
 - Same as non-registered, full function database recovery procedure

IMS Catalog Benefits

- Trusted, online source for IMS metadata
- Leveraged by the Universal Database Drivers
- No need for “local” Java metadata classes
- Direct access to IMS metadata in the catalog
 - IMS Explorer for Development
 - “GUR” DL/I call
 - Java Clients
 - Other Tooling

IMS Catalog Summary

- IMS Catalog
 - Contains information about IMS database resources, program resources, and relevant application metadata
 - Must use new ACBGEN
 - Populate and maintain the Catalog metadata
 - The IMS Catalog: The Next Step in Simplification and Enterprise Integration
<http://www-01.ibm.com/software/sw-events/teleconference/D108305C25358T22.html>
 - IMS 12: The IMS Catalog – Redpaper: REDP-4812-00 (Sept 2012)
<http://www.redbooks.ibm.com/redpieces/abstracts/REDP4812.html?Open>

- IMS Enterprise Suite Explorer for Development
 - Generate or update DBD/PSB source used to populate the catalog
 - Enables IMS metadata discovery from the IMS Catalog
 - IMS Enterprise Suite Explorer: The new face of IMS
<http://www-01.ibm.com/software/sw-events/teleconference/T929253D59548D24.html>

THANK
YOU

Complete your session evaluations online at www.SHARE.org/Orlando-Eval