IMS 14 Database Enhancements

SHARE Session 17755

Dennis Eichelberger - deichel@us.ibm.com
Nancy Stein - ngstein@us.ibm.com

IBM Washington System Center / IMS Team

August 13, 2015

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.

Database Enhancements

- Database Enhancements
- DBRC Enhancements
- IMS Management of ACBs

OSAM HALDB 8G Support with No OLR

- **Challenge Addressed**
  - When moving to OSAM HALDB, the database size must be reduced from 8G to 4G

Target Market:

All OSAM HALDB users who want to have the 8 GB OSAM data set capacity for their HALDB partitions

- **Solution**
  - Allow 8GB OSAM support for HALDBs
    - Online reorganization support would still limit OSAM database to 4G
      - OSAM HALDB that is 8GB capable cannot use the HALDB online reorganization process
    - Option at the database level
      - Some HALDBs can be defined 8GB, others can be 4GB & use OLR
  - DBRC commands enhanced to set 8G option for OSAM HALDB master databases

- Provides increased scalability for OSAM HALDB databases

Automatic SDEP Buffer Management

Target Market:

All IMS Fast Path SDEP users

- **Challenge Addressed**
  - Shared SDEP CIs not being deleted due to one low activity IMS system
  - Undeleted SDEP may cause DEDB Area to fill and cause a database outage

- **Solution**
  - IMS 14 allows SDEP CIs to be automatically cleaned up on a regular basis
    - User Defined Frequency
    - User Defined Interval
  - Reduce the physical delete lag from logical delete

- Automates existing manual processes associated with SDEP processing
- Reduces potential DEDB Area outages associated with SDEP usage

FP DEDB Alter Enhancements

Target Market:

IMS Fast Path users requiring 24x7 availability of DEDB Areas

- **Challenge Addressed**
  - Need to change the DEDB Area definition SIZE, UOW, and/or ROOT parameters to increase the size of an Area when SDEPs are defined
  - Need to add an Area to a DEDB without bringing the DB offline
  - Need to support XRF and FDBR environments when ACBSHR=N

- **Solution**
  - Enhance the existing IMS 13 DEDB Alter capabilities
    - Allow alter when SDEPs are defined
    - Allow the addition of one or more Areas to the end of a DEDB
    - Segment Edit/Compression exit can be added
      - Assumes the user has an exit that can handle the mix of compressed and non-compressed segments
      - IMS does not provide an exit that supports a mix of compressed and non-compressed segments
    - Allow alter when using XRF/FDBR with ACBSHR=N

  - Improves data availability by allowing selected DEDB changes without taking the database offline

FDBR Resolve In-Doubt Notification Exit – DFSFIDN0

- **Challenge Addressed**
  - It can be difficult to code the DFSFIDN0 exit to properly resolve in-doubt work when DB2 is involved in the unit of work

Target Market:

All FDBR customers that use the ESAF interface (e.g., DB2) and wish to identify in-doubt UORs after an abend

- **Solution**
  - New sample DFSFIDN0 user exist that issues a new message for each in-doubt ESS UOR
    - DFS3722I identifies in-doubt work by: Subsystem Name, Subsystem Type, Resolve-in-doubt Action, and Origin Application Schedule Number (OASN)
    - Automation can monitor the messages and act upon the in-doubt UOR
  - Improves usability when using both FDBR and DB2 z/OS
  - Can reduce the time DB2 holds locks during an IMS abend

Database Enhancements

- **FP 64-Bit Buffers for High Speed Utilities**
  - UOWs which have large numbers or segments in IOVF cannot be processed due to a lack of available 31-bit ECSA
  - New keywords on utility SYSIN DD controls storage allocation location
    - FPUBUF64 or FPUBUF31
- **Open Database Manager (ODBM) Accounting**
  - Provide optional SMF records that can be used for application charge back
  - Enabled through new parameter in CSLDIxxx
    - LOGOPT=(ACCOUNTING)
  - SMF type 29, subtype 01 records
- **OSAM DEB information moved to 31 bit private storage**
  - Increase the number of OSAM datasets open at a time
  - Provide 24-bit virtual storage relief

Database Enhancements

- **DBRC Enhancements**
  - New REPAIR.RECON command can be used to verify and fix inconsistencies in the RECON data set related to DMB numbers
  - DBRC migration and coexistence with IMS 12 and IMS 13

IMS Dynamic Database

Target Market:

All IMS DB customers

- **Challenge Addressed**
  - Database schemas are static and changes require system definition and generation
- **Solution**
  - Allow optional IMS management of ACBs
    - IMS Catalog is the trusted source of database and program definitions
    - ACB, DBD & PSB libraries no longer used or needed by IMS
    - IMS system can use either Catalog or ACBLIB for all definitions
  - Allow optional use of Data Definition Language (DDL) to affect database and schema changes
  - Provide an audit trail to capture information about IMS Catalog updates
- **Business Value**
  - Removes the external need for PSBGEN, DBDGEN, and ACBGEN
  - Support the use of industry standard DDL for database and schema change
    - Use DDL-authoring tools prevalent in the market

IMS Management of ACBs

Target Market:

All IMS DB customers

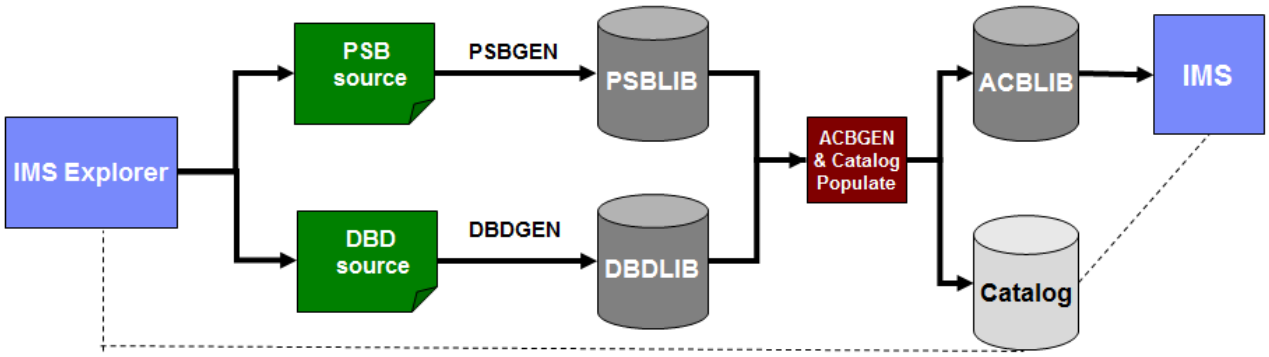
- **Challenge Addressed**
  - Database schemas are static and changes require system definition and generation

- **Solution**
  - Allow optional IMS management of ACBs
    - IMS Catalog is the trusted source of database and program definitions
    - ACB, DBD & PSB libraries no longer used or needed by IMS
    - IMS system can use either Catalog or ACBLIB for all definitions

- **Business Value**
  - Removes the external need for PSB, DBD, and ACB GEN processes
  - Dynamic implementation of IMS database and program resource definitions
    - No system outage required to create or modify IMS databases
    - Improved IMS database availability

IMS Catalog Review

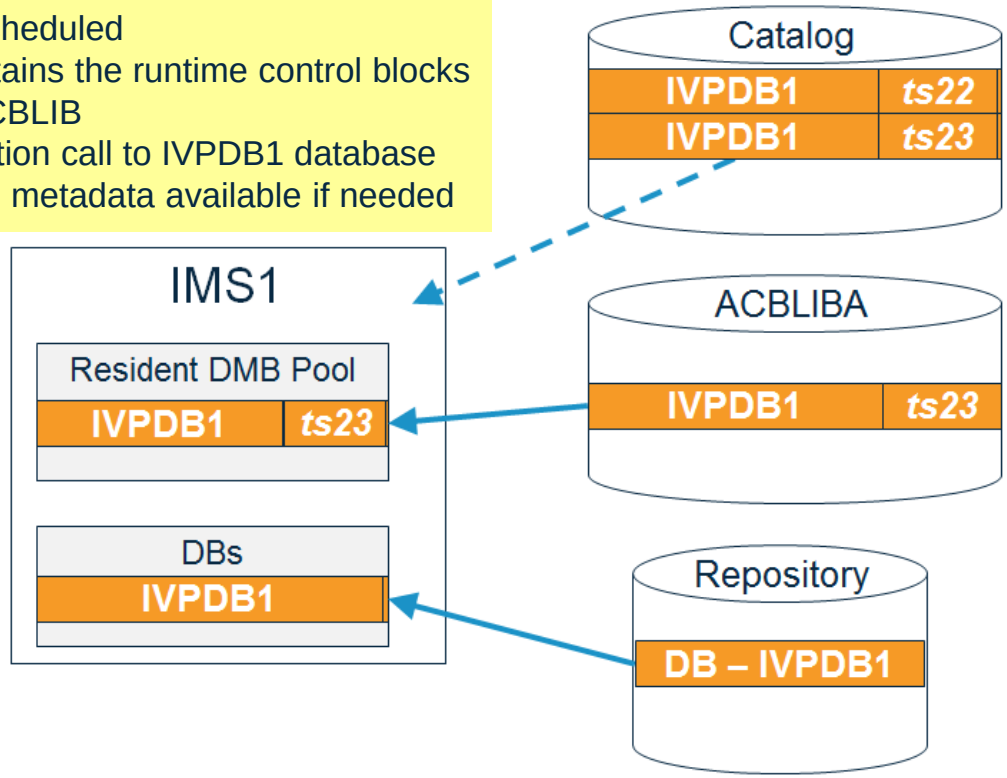
- The IMS catalog contains metadata about IMS program and database resources, and application COPYBOOKs / INCLUDEs
  - Includes all program and database related information defined to an IMS system, including databases, fields, segments, data types, and more
  - Changes to these IMS resources are reflected in the catalog metadata
- IMS PHIDAM / OSAM HALDB database with 4 Data Set Groups
- Using the IMS Catalog and ACBLIB
  - IMS database and program resource definitions start with DBD and PSB macro source coding, followed by DBDGEN and PSBGEN
  - Followed by “ACBGEN & Catalog Populate Utility” processing
    - Catalog and active ACB member are kept in sync via the ACBGEN process
  - IMS runtime database and program control blocks obtained from ACBLIB
  - IMS metadata is available from the catalog



IMS Resource Definitions Today

- **IMS Catalog**
  - Contains multiple instances & versions of database, program, and application metadata
- **IMS ACBLIB**
  - Contains the active ACB control blocks defining the DBD and PSB resources
    - Used by IMS to load the Application Control Blocks at runtime
- **IMS DRD Repository**
  - Defines the resources available for use in the IMS system

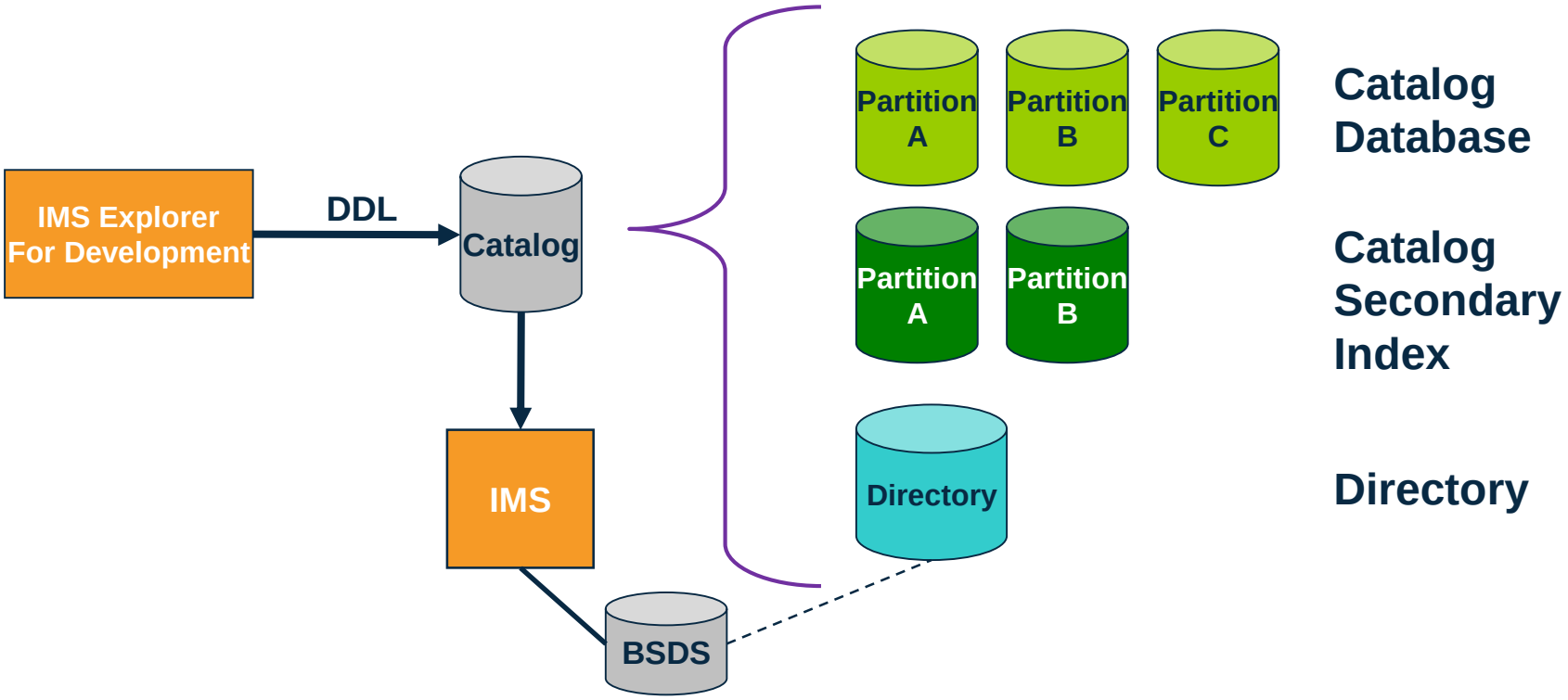
- PSB Scheduled
- IMS obtains the runtime control blocks from ACBLIB
- Application call to IVPDB1 database
- Catalog metadata available if needed



IMS Management of ACBs Tomorrow

- In IMS14, customers may optionally enable IMS management of Application Control Blocks (ACBs) for databases and program views
  - Use **Data Definition Language (DDL)** to affect IMS database and schema changes
  - IMS Catalog updated automatically when SQL DDL statements are used
    - If directed, IMS can activate changes to the database and program view definitions automatically
    - Changes not activated automatically are stored within IMS for later activation
      - » **IMPORT DEFN SOURCE(CATALOG) command**
  - SQL DDL statements can be submitted to IMS through products such as the IMS Explorer for Development, Java programs, or other DDL utilities
    - No longer need to code DBD and PSB macros
    - No longer need to generate DBDs, PSBs, and ACBs with utilities
    - No need to use Online Change or recycle an IMS system to activate an ACB
  - **ACBMGMT=CATALOG** in the <CATALOG> section of DFSDFxxx proclib member
    - IMS can build, activate, and load ACBs into memory dynamically from the catalog when database and program view definitions are created via SQL DDL statements

IMS 14 Catalog



- An IMS Catalog is made up of several components
  - A Catalog HALDB Database (many partitions, 4 Data Set Groups)
  - A Secondary Index
  - Directory Dataset(s)

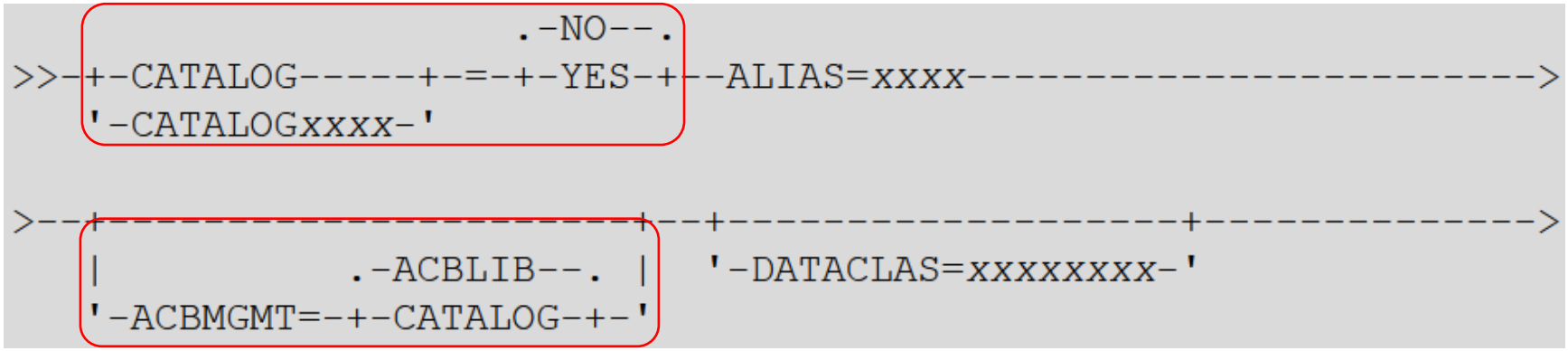
IMS Management of ACBs

- ACBs are stored in the IMS Directory which is an extension of the Catalog
- The Directory has functionality similar to an ACBLIB
  - Directory records have a format much like the ACBs in an ACBLIB
  - The Directory is kept in sync with the Catalog
- IMS will self-manage the Directory
  - A Directory data set is automatically allocated as a PDSE
    - another Directory data set is allocated when the PDSE becomes full
  - PDSEs are named using an extension of the catalog database data set name
    - <HALDB data set prefix>.DI<suffix>
- IMS will maintain a Boot Strap Data Set (BSDS) containing information about the Directory data set(s)
- When IMS ACB management is enabled
  - IMS will use the Directory to indicate which members are active in the IMS catalog
  - IMS will reference the Directory to get the runtime application control blocks
- 1:1 relationship between ACBs in the Directory and the Catalog
  - If the Catalog is shared then Directory data sets are shared
  - If the Catalog is non-shared then Directory data sets are non-shared

IMS Management of ACBs Directory

- PDSE Benefits
  - Automatic reuse of space
    - No manual compression required
  - Allows up to 123 extents
  - PDSE directory automatically extends
    - No “out of directory block” errors
  - Maximum size of a PDSE member is 15,728,639 records
  - Maximum number of PDSE members is 524,236
  - All updates to a PDSE are atomic, unlike a traditional PDS
    - Canceled jobs or system crashes will not corrupt a PDSE
  - Can be shared across a SYSPLEX

Enabling IMS Catalog ACB Management



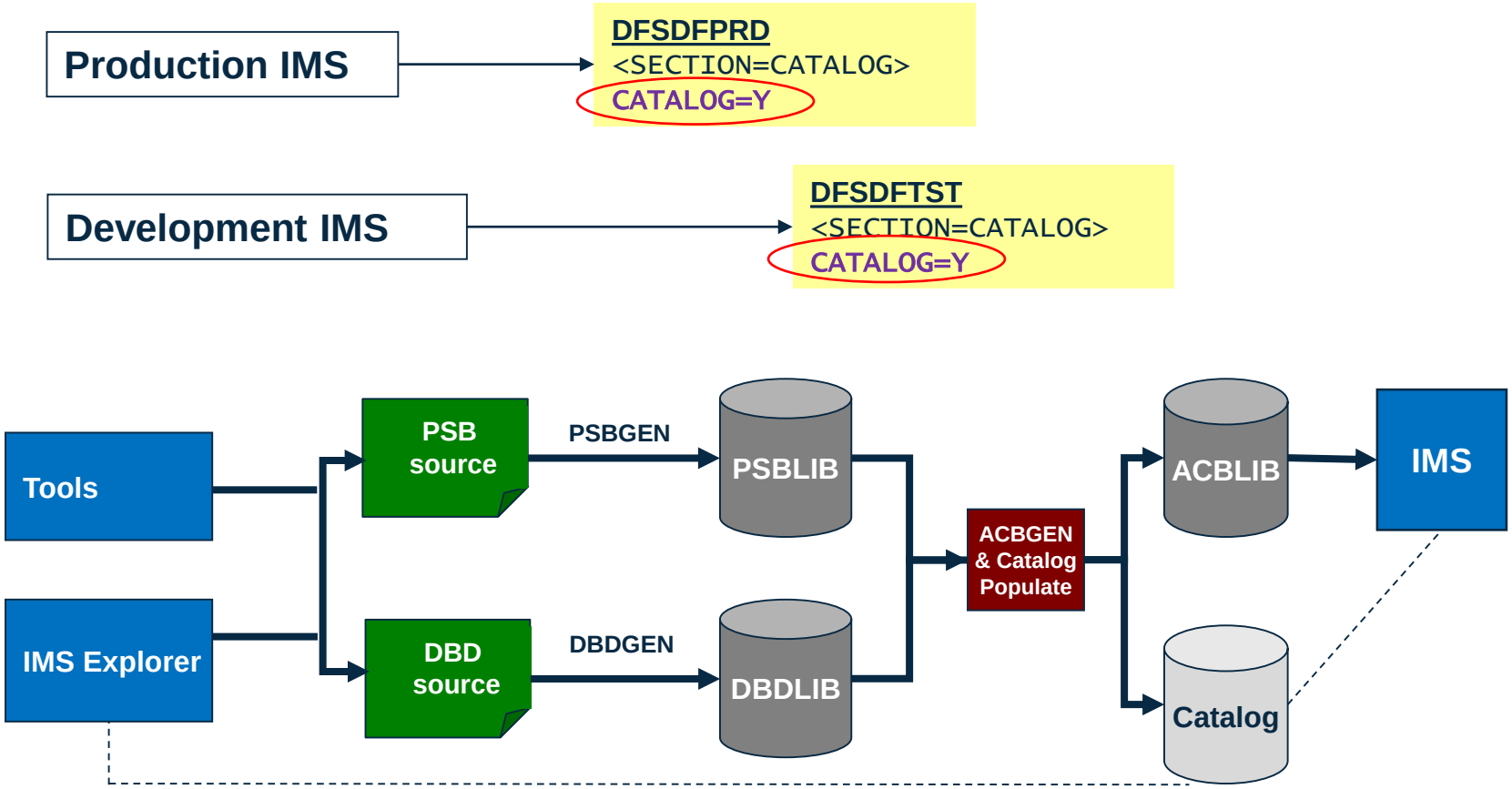
- In the CATALOG section of the DFSDFxxx PROCLIB member
  - Enable the IMS Catalog
    - CATALOG=**NO**
      - Specifies the IMS catalog is disabled
      - Default value and value if CATALOG section is omitted
    - CATALOG=YES
      - Specifies the IMS catalog is enabled
      - Catalog runtime resources must be included in the IMS system
        - » DFSCP000, DFSCD000, and DFSCX000

Enabling IMS Catalog ACB Management

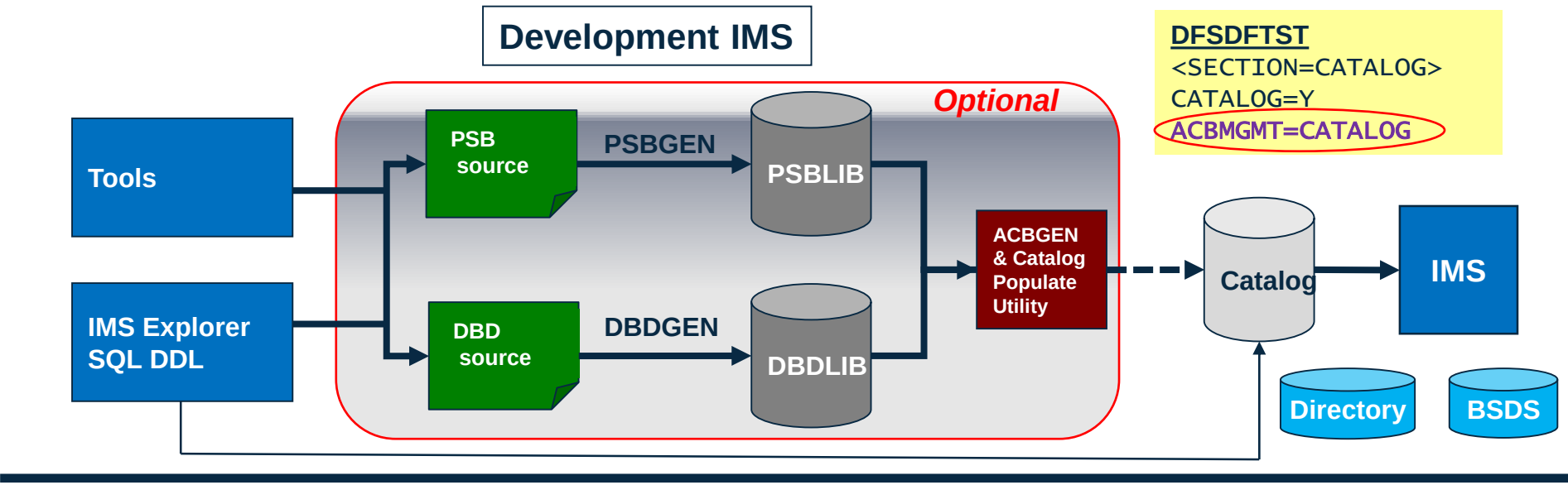
- Enable IMS Catalog ACB management
  - ACBMGMT=**ACBLIB**
    - IMS catalog is enabled and populated
    - User manages the runtime control blocks
      - » gen utilities, ACBLIB, and OLC
    - IMS catalog will be used by IMS to retrieve metadata
    - Default option if IMS catalog is enabled
  - ACBMGMT=CATALOG
    - IMS catalog must be enabled and populated using DFS3PU00 utility with MANAGEDACBS=SETUP
    - IMS manages runtime application control blocks using the IMS catalog
    - IMS will abend if the internally managed directory data sets are not found
    - DBD, PSB and ACB libraries no longer required

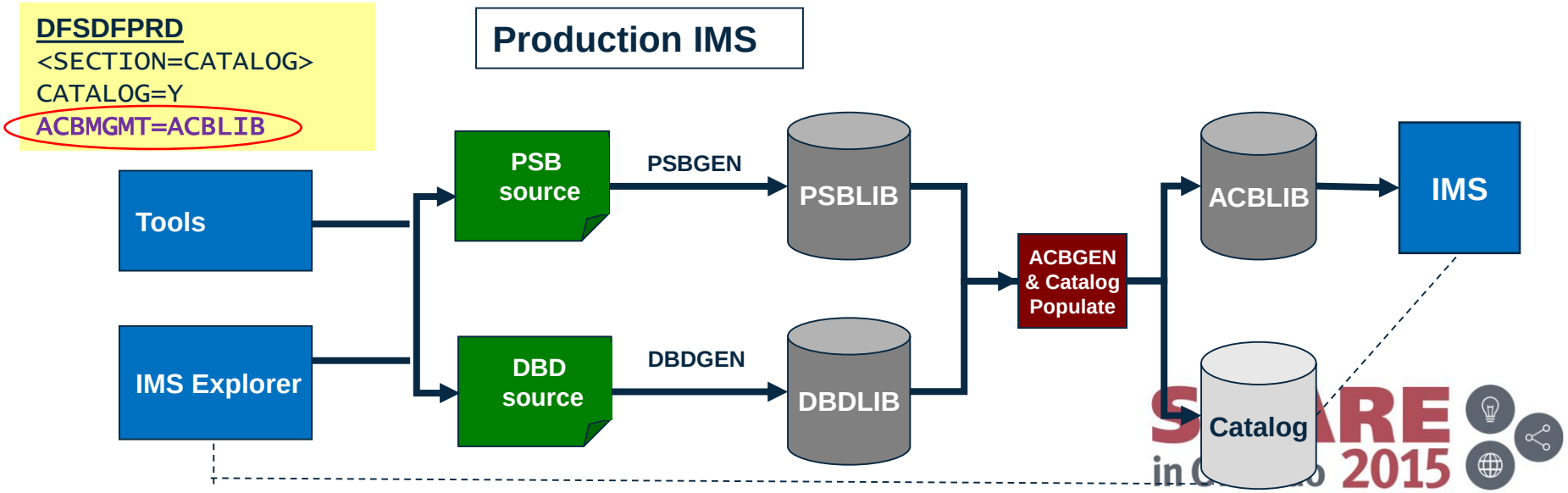
```
<SECTION=CATALOG>                                              /*Section Header*/
CATALOG=Y                                                    /*Enable IMS catalog*/
ALIAS=DFSC                                          /*Use standard catalog prefix DFSC*/
ACBMGMT=CATALOG           /*IMS manages the runtime application control blocks*/
RETENTION=(INSTANCES=5,DAYS=365)                             /*Retention criteria*/
```

Pre-IMS 14: IMS with Catalog and ACBLIB

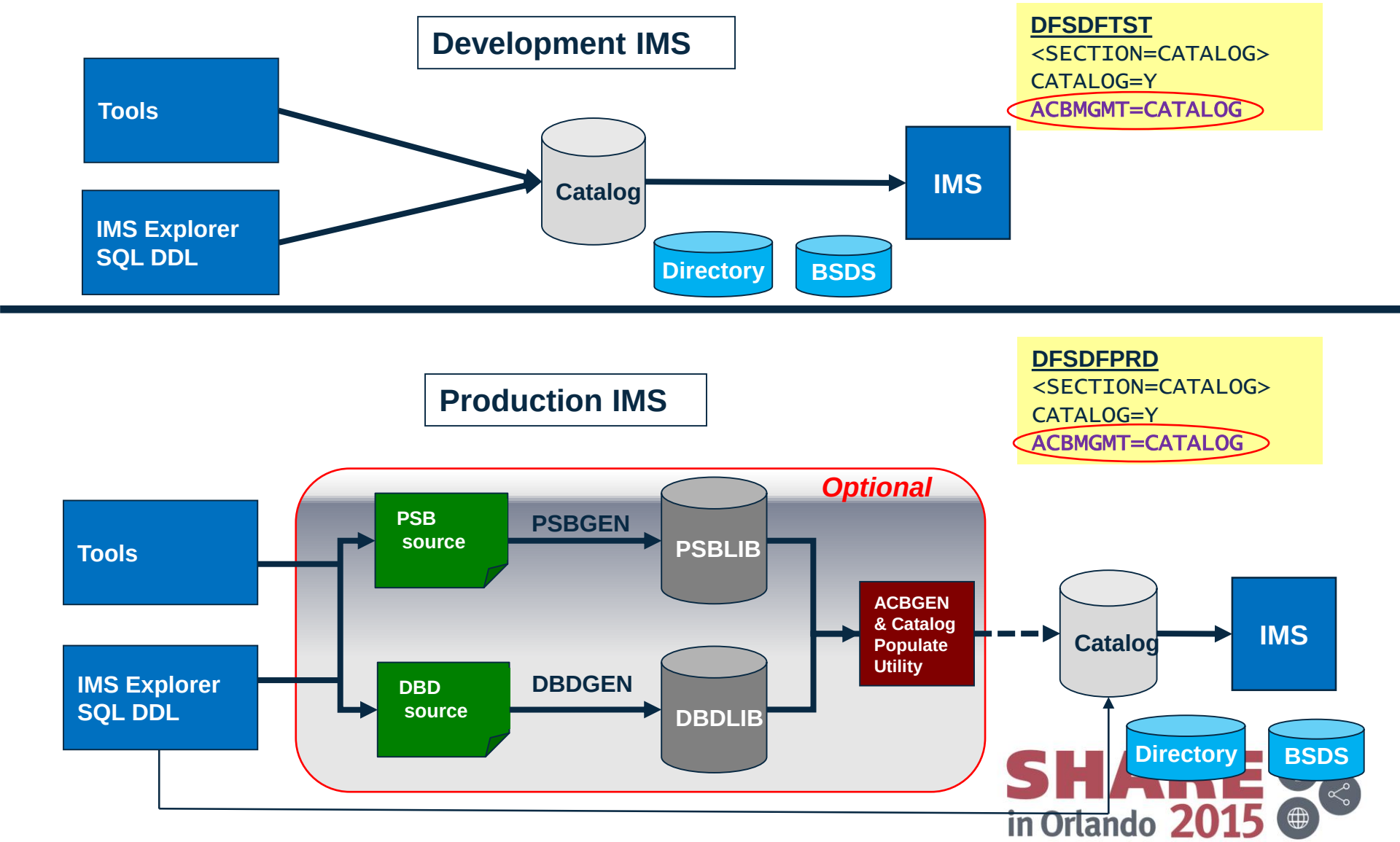


IMS 14: Transition to IMS Catalog Managed ACBs

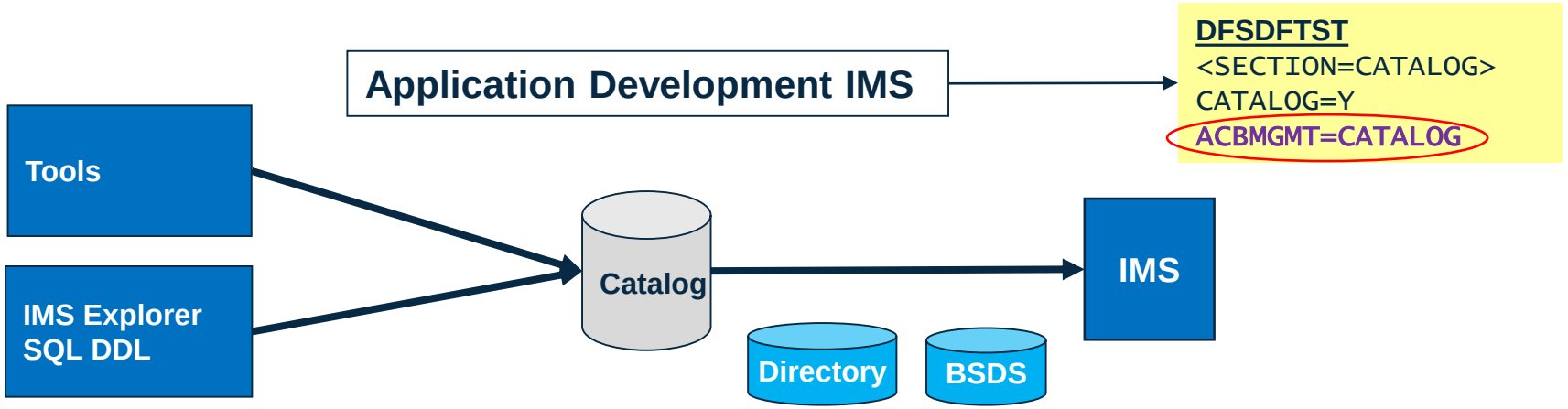


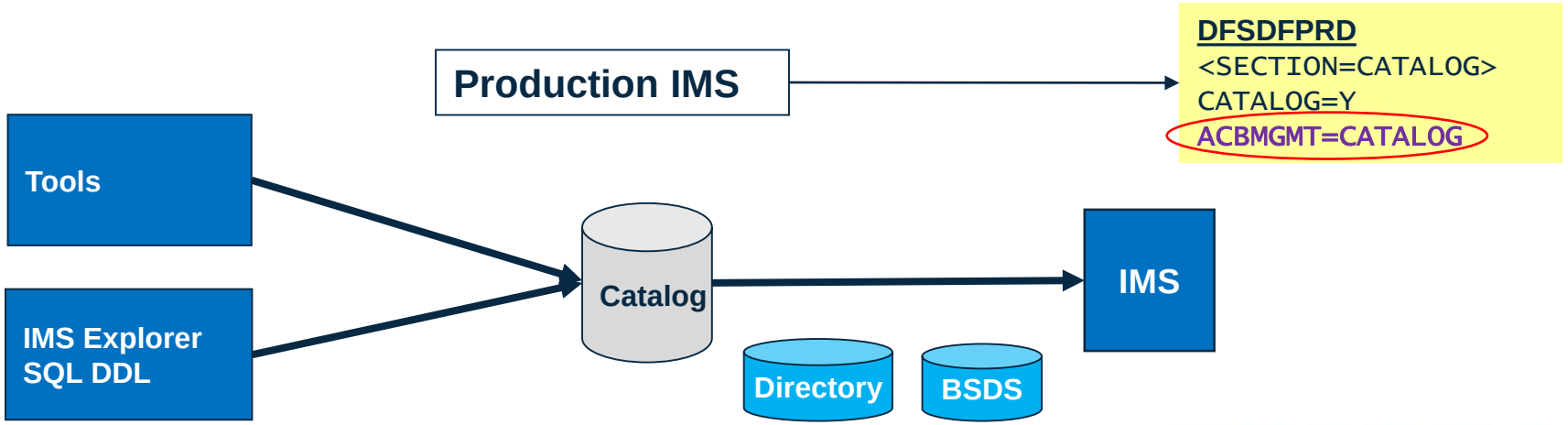


IMS 14: Transition to IMS Catalog Managed ACBs



IMS 14: Total Transition to IMS ACB Management

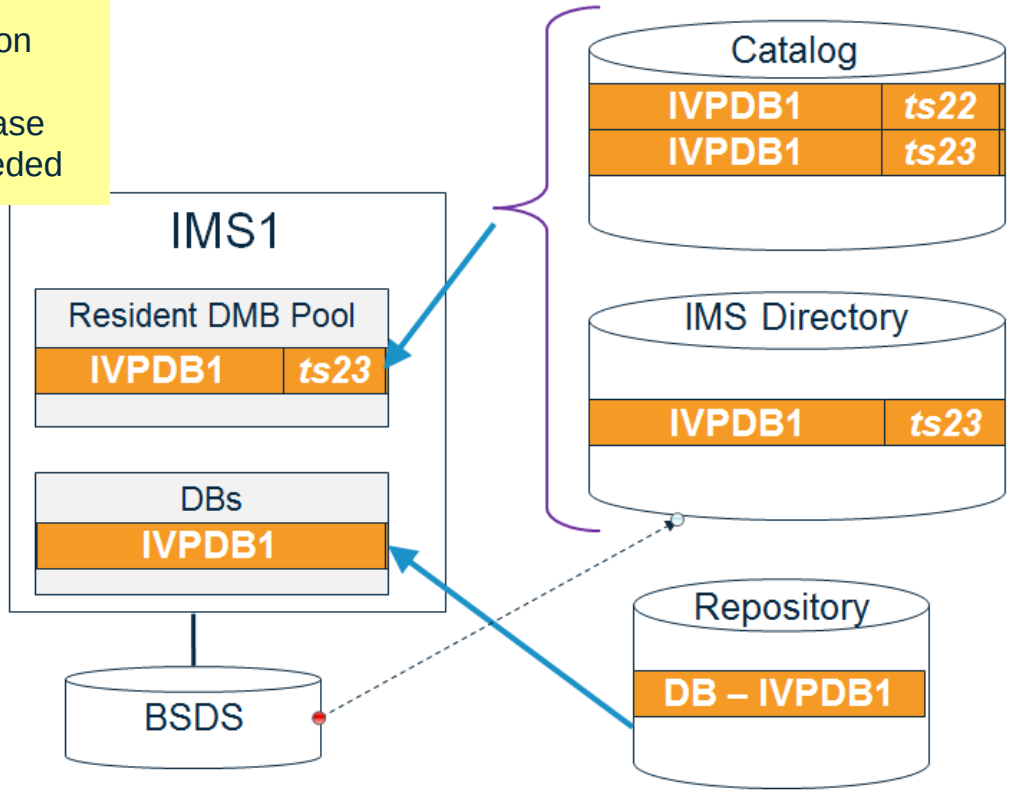




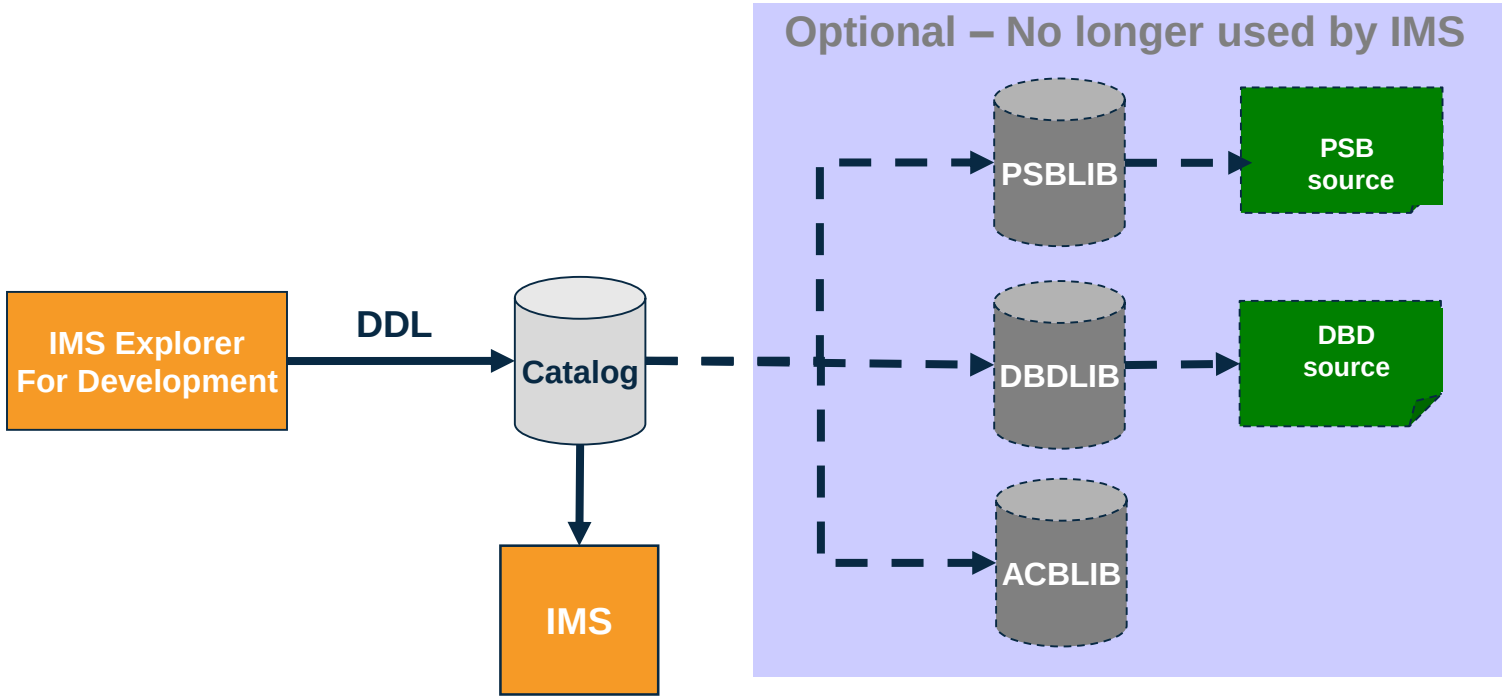
Running with IMS Catalog Managed ACBs

- IMS databases and programs are defined in the Catalog
  - Defines the active instance of database, program, and application metadata
  - Contains additional instances of database, program, and application metadata
  - Contains the application control blocks (ACBs)
- DRD Repository
  - Defines the resources available for use in an IMS system

- PSB Scheduled
- IMS obtains the runtime application control blocks from the Catalog
- Application call to IVPDB1 database
- Catalog metadata available if needed



IMS ACB Management Advantage



- Create DDL statements to build or modify IMS database or program resources
- IMS processes SQL DDL statements
- Activation of some DDL commands can happen automatically or commands are held in pending activation state until an IMPORT command is entered
  - **IMPORT DEFN SOURCE(CATALOG)**
- IMS application control blocks will be loaded from the catalog at runtime

Maintaining DBDLIB, PSBLIB and ACBLIB

- **Two methods to retain and maintain DBD, PSB and ACB libraries**
  - **Run the ACB Generation & Catalog Populate utility (DFS3UACB)**
    - Builds the application control blocks in ACBLIB
    - Updates the IMS catalog metadata and flags the active resources
    - Activates the ACBs by loading them into the IMS directory

    OR

  - **Run the ACB Maintenance utility, DFSUACB0**
    - Builds the application control blocks in ACBLIB

    AND
  - **Run the IMS Catalog Populate utility, DFS3PU00**
    - Updates the IMS catalog metadata and flags the active resources
    - Activates the ACBs by loading them into the IMS directory

Architectural Considerations

- Dynamic Database Definition function is disabled by default
- IMS catalog must be defined, populated and enabled
- IMS managed ACBs must be enabled
- IMS directory data sets and the BSDS can be built by IMS
- Internally created tasks are used to update the catalog
  - Number of PSTs may need to be increased if getting near your maximum
- Increased usage of 64 bit storage with the function enabled
  - Improved performance
- Internal construction of DMBs and PSBs are in a new code path
  - Any customized exits used in the generation of these control blocks ?
- Requires CSL - SCI and OM
- DBDLIB, PSBLIB and ACBLIB will no longer be maintained by IMS
- DBDGEN, PSBGEN and ACBGEN can be manually maintained

Operational Considerations

- Operational Characteristics
  - IMS will manage the directory
  - Databases and programs can be created and updated through DDL
  - No longer need to maintain …
    - DBD source, PSB source, DBDLIB, PSBLIB, and ACBLIB
    - Migration / Coexistence path will allow for continued usage of DBD and PSB source during transition
- Operational Considerations
  - Catalog should be updated through DDL
  - Catalog can still be updated with the Catalog Populate utility
    - Allows support for existing path of source to catalog via DBDGEN / PSBGEN / ACBGEN
  - Reorganize the IMS catalog database with OLR
  - Image Copy the IMS catalog on a regular basis
  - Include IMS catalog components in a Global Mirroring consistency group
  - Audit data will be collected for IMS catalog database updates

Performance Considerations

- Performance Characteristics
  - Performance has not been determined yet
    - Expectation is that IMS performance should not be impacted by catalog
  - Few differences in using catalog with directory vs. ACBLIB during IMS initialization
    - IMS must load control blocks directly into memory from DASD
  - Increased usage of 64 bit storage to cache control blocks
    - Eliminates I/O to directory during normal scheduling after caching
    - Reduces I/O to directory during the activation of changes
  - IMS automatically executes steps to change resources while processing DDL
    - Activation process may take longer since IMS does more work
- Performance Considerations
  - Impact of DDL change is similar to impact of DBD or PSB source change

Setup / Migration Considerations

- IMS catalog sharing
  - An IMS 14 catalog can be shared with IMS 12 or IMS 13
  - Older IMS will not process new fields in the catalog
- Sharing catalog with directory on previous releases
  - Can be done but not recommended
    - Difficult to keep resources in sync
    - IMS 14 catalog updated by DDL
    - IMS 12 & 13 source DBD / PSB updates followed by DBD, PSB and ACBGEN
  - Recommendation
    - Enable catalog, migrate to IMS 14 then enable IMS Managed ACBs
- Currently no plans for IMS 12 or 13 coexistence APARs
- DFS3UACB utility should not be needed after migration to DDL
  - Use during migration from DBD / PSB source to DDL

Migration Considerations

- Use of the new IMS ACB management feature is optional
- Define HALDB Catalog database, install/gen catalog DBD/PSB/ACBs, and enable the IMS catalog and IMS ACB management
  - DFSDFxxx PROCLIB member <SECTION=CATALOG>
    - CATALOG=YES
    - ACBMGMT=CATALOG
  - Run the Catalog Populate utility with MANAGEDACBS=SETUP
    - Creates a new IMS directory and corresponding BSDS required by IMS to manage ACBs
    - Loads the IMS directory with the ACBs contained in the concatenated ACBLIBs
  - IMS obtains runtime ACB control blocks from the IMS catalog
  - No longer need to gen ACB control blocks using utilities
  - No longer need to use Online Change to in-effect ACB changes
  - DBD, PSB and ACB libraries can be retired
- Start adding and changing database definitions dynamically using DDL