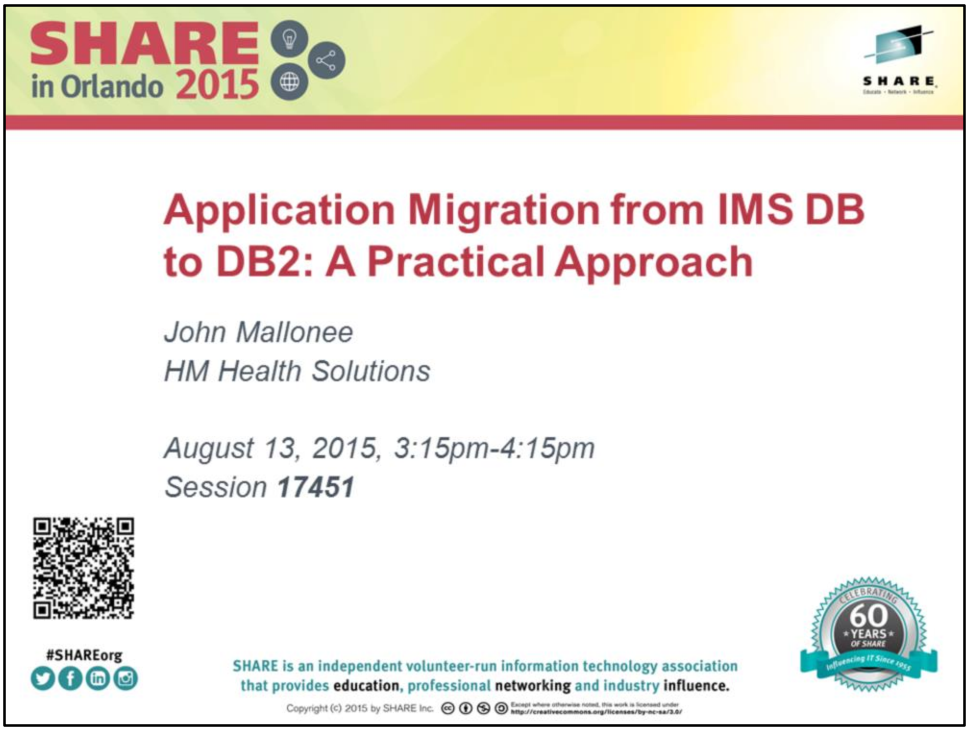


SHARE in Orlando 2015

SHARE Educate · Network · Influence

Application Migration from IMS DB to DB2: A Practical Approach

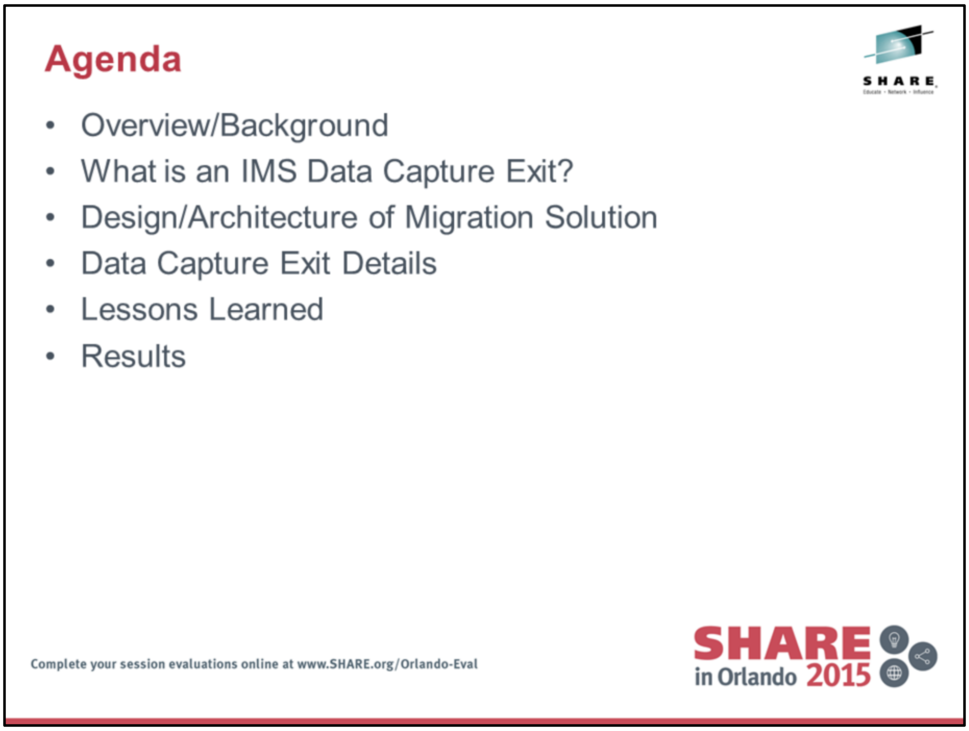
John Mallonee
HM Health Solutions

August 13, 2015, 3:15pm-4:15pm
*Session **17451***

#SHAREorg

CELEBRATING
60 YEARS OF SHARE
Influencing IT Since 1955

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



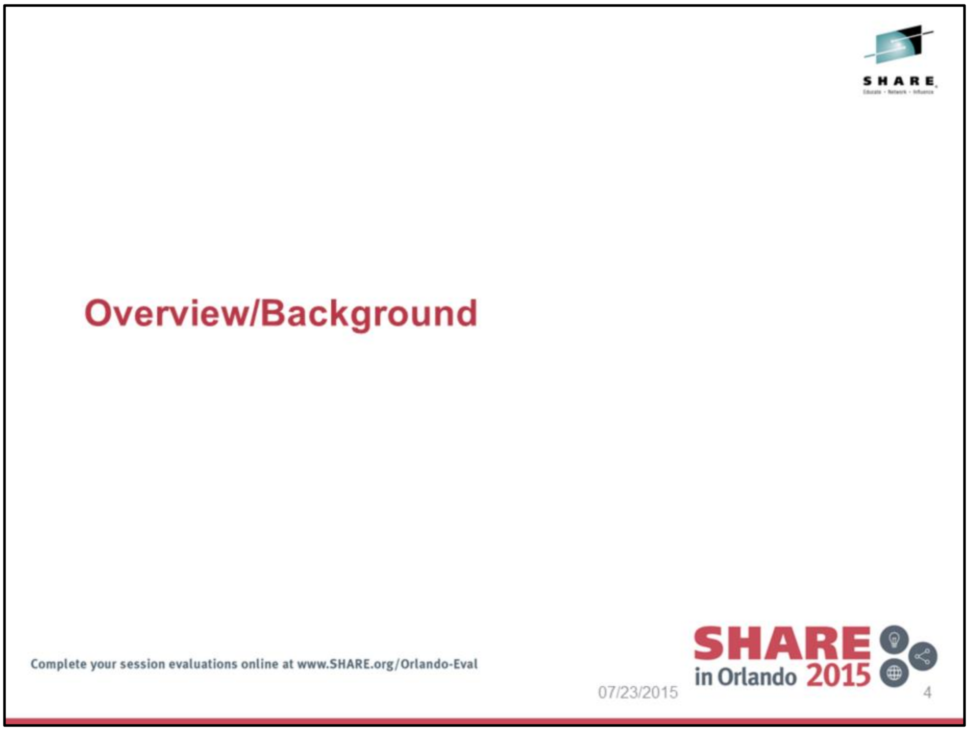
Agenda

- Overview/Background
- What is an IMS Data Capture Exit?
- Design/Architecture of Migration Solution
- Data Capture Exit Details
- Lessons Learned
- Results

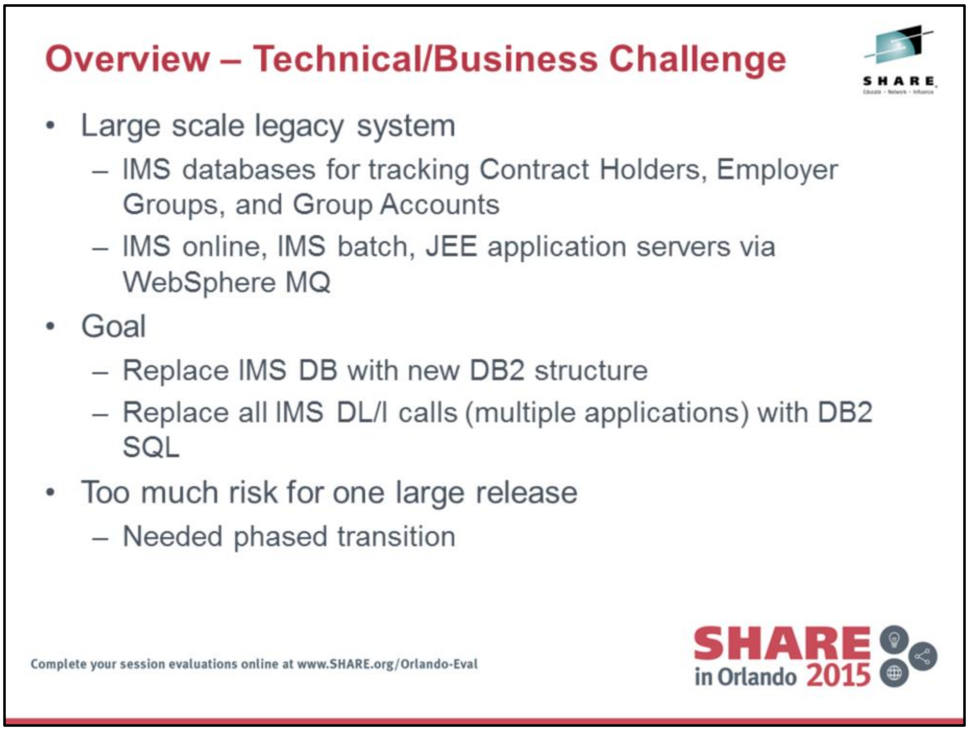
Complete your session evaluations online at www.SHARE.org/Orlando-Eval

SHARE in Orlando 2015

Are you looking to migrate from a legacy IMS database to DB2? There are practical approaches that allow for an incremental transition and minimize the risk of a "big bang" conversion. This session will demonstrate how we used an IMS Data Capture exit routine written in COBOL to synchronize data between IMS and DB2 during an enterprise data migration. It will present the overall business need for the synchronization, technical details on the creation of the exit routine and integration with our enterprise applications, environmental considerations, and practical results from the execution in the test and production environments.

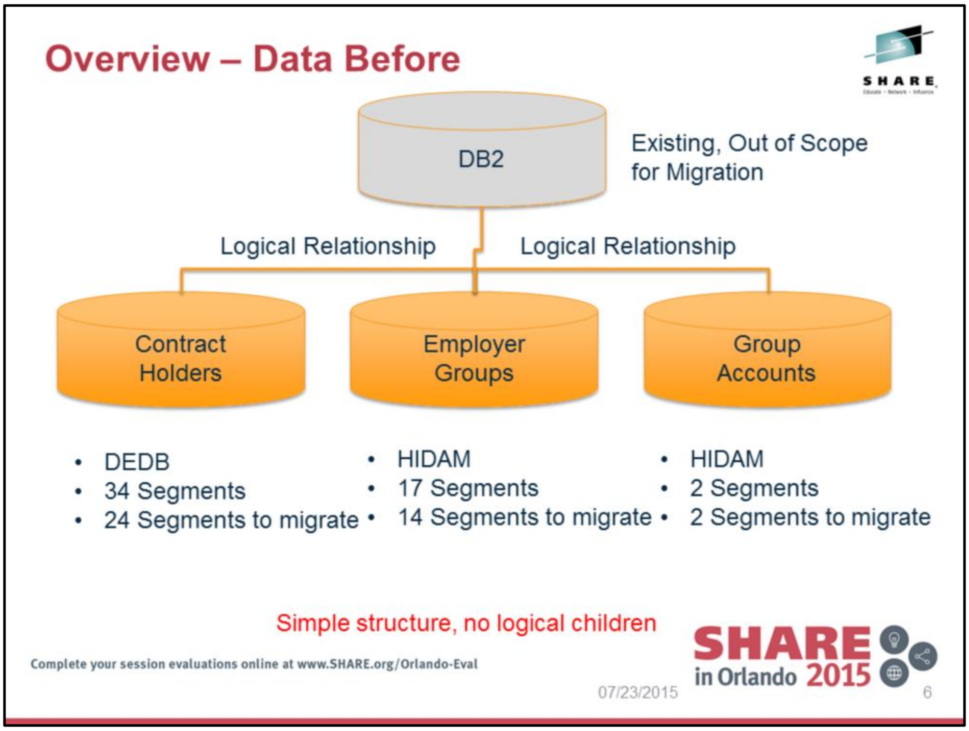


Overview/Background



Overview – Technical/Business Challenge

- Large scale legacy system
  - IMS databases for tracking Contract Holders, Employer Groups, and Group Accounts
  - IMS online, IMS batch, JEE application servers via WebSphere MQ
- Goal
  - Replace IMS DB with new DB2 structure
  - Replace all IMS DL/I calls (multiple applications) with DB2 SQL
- Too much risk for one large release
  - Needed phased transition



Overview – Data Before

DB2

Existing, Out of Scope for Migration

Logical Relationship

Logical Relationship

Contract Holders

- DEDB
- 34 Segments
- 24 Segments to migrate

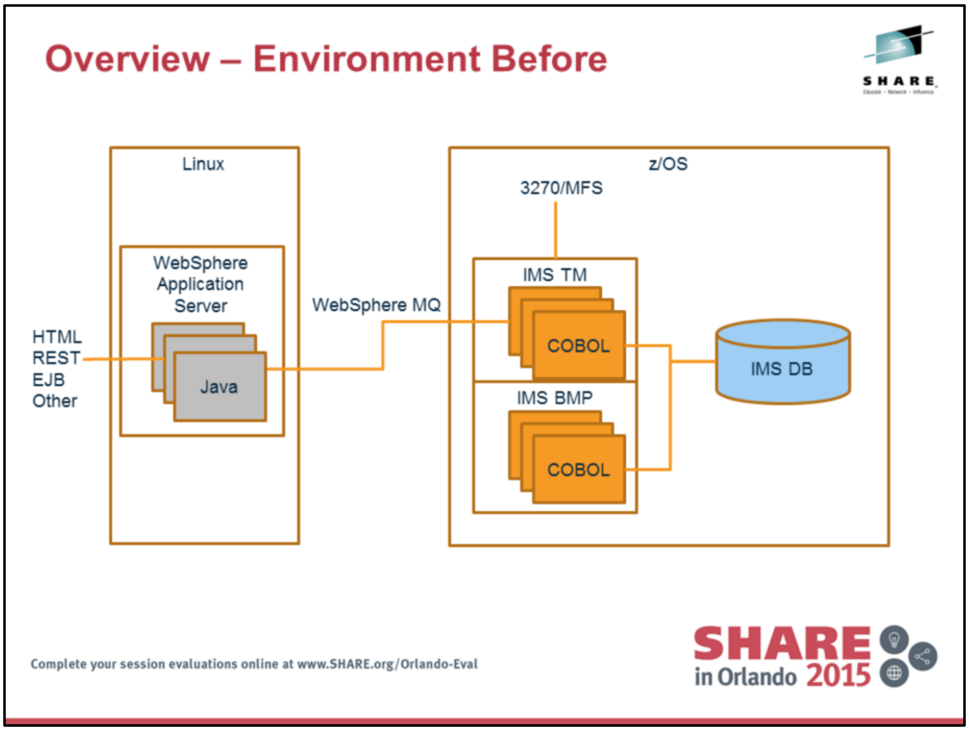
Employer Groups

- HIDAM
- 17 Segments
- 14 Segments to migrate

Group Accounts

- HIDAM
- 2 Segments
- 2 Segments to migrate

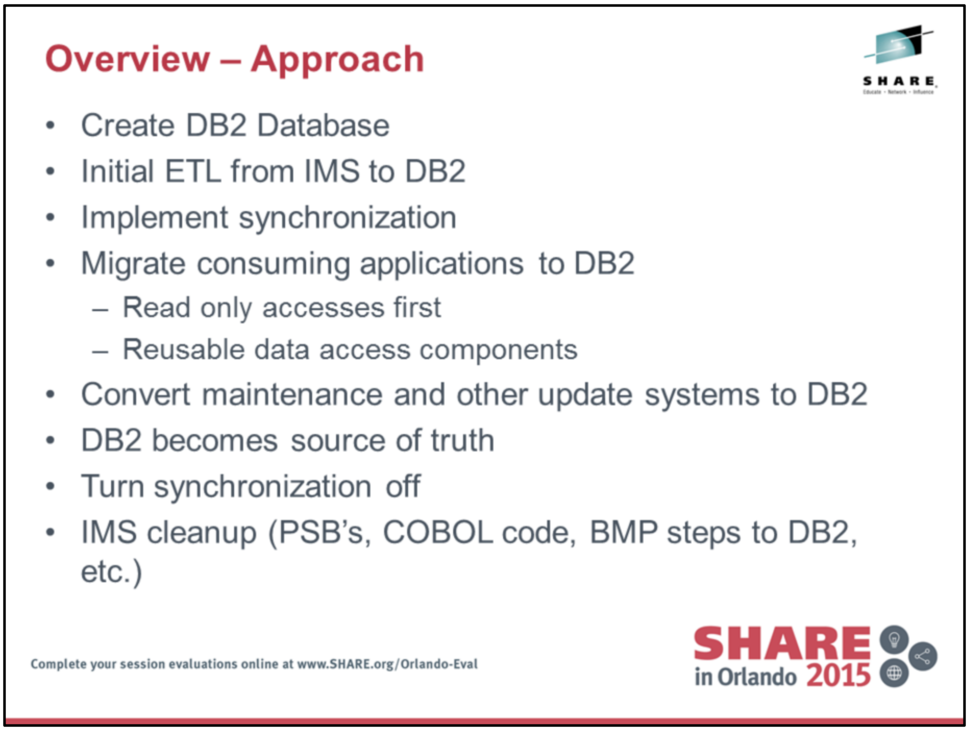
Simple structure, no logical children



Overview – Environment Before

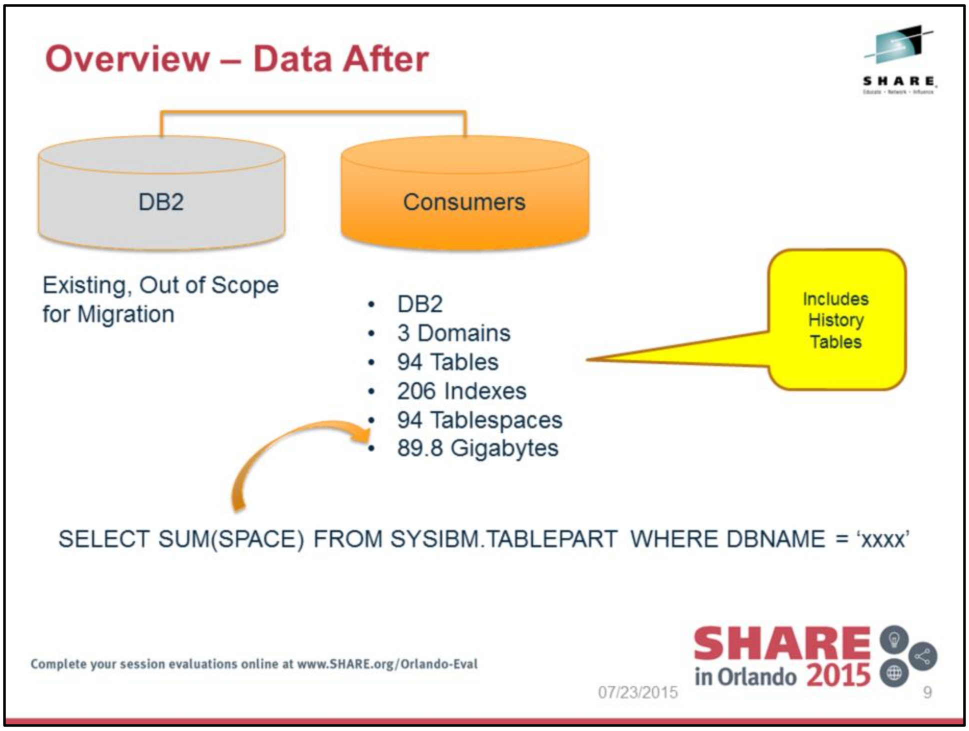
Complete your session evaluations online at www.SHARE.org/Orlando-Eval

SHARE in Orlando 2015



Overview – Approach

- Create DB2 Database
- Initial ETL from IMS to DB2
- Implement synchronization
- Migrate consuming applications to DB2
  - Read only accesses first
  - Reusable data access components
- Convert maintenance and other update systems to DB2
- DB2 becomes source of truth
- Turn synchronization off
- IMS cleanup (PSB's, COBOL code, BMP steps to DB2, etc.)



Overview – Data After

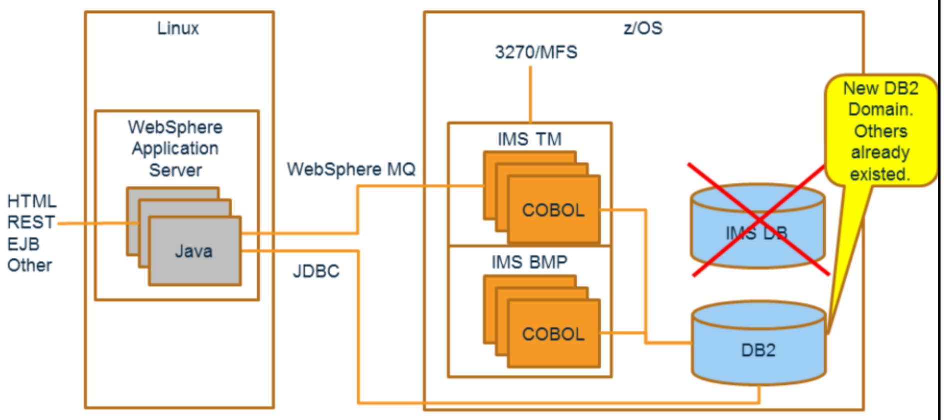
Existing, Out of Scope for Migration

- DB2
- 3 Domains
- 94 Tables
- 206 Indexes
- 94 Tablespaces
- 89.8 Gigabytes

Includes History Tables

```
SELECT SUM(SPACE) FROM SYSIBM.TABLEPART  WHERE DBNAME = 'xxxx'
```

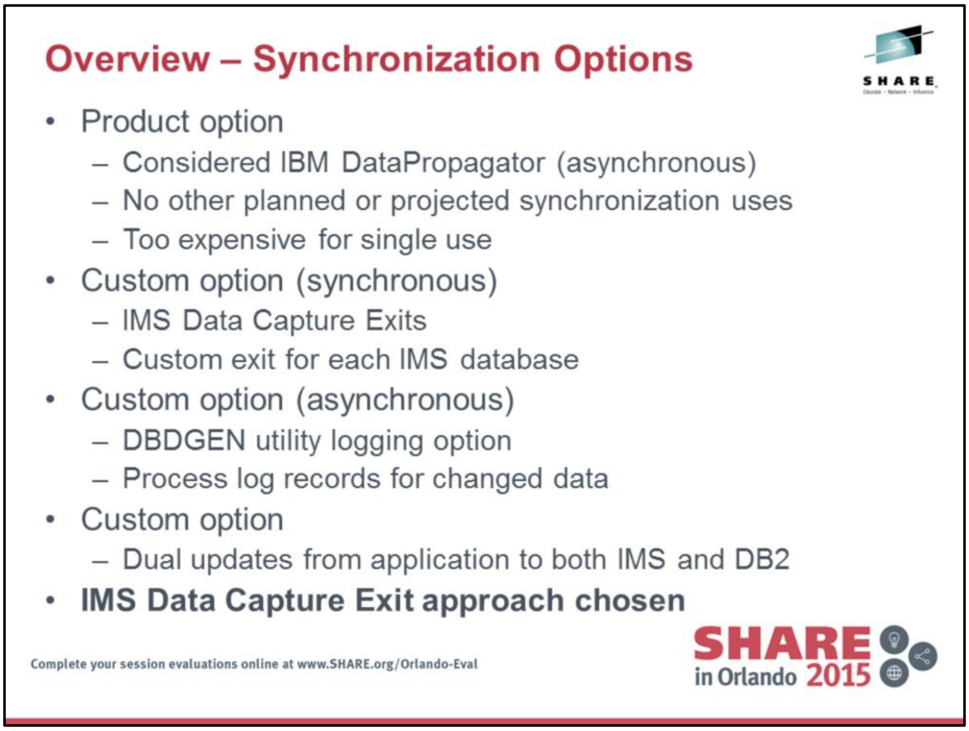
Overview – Environment After



New DB2 Domain. Others already existed.

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

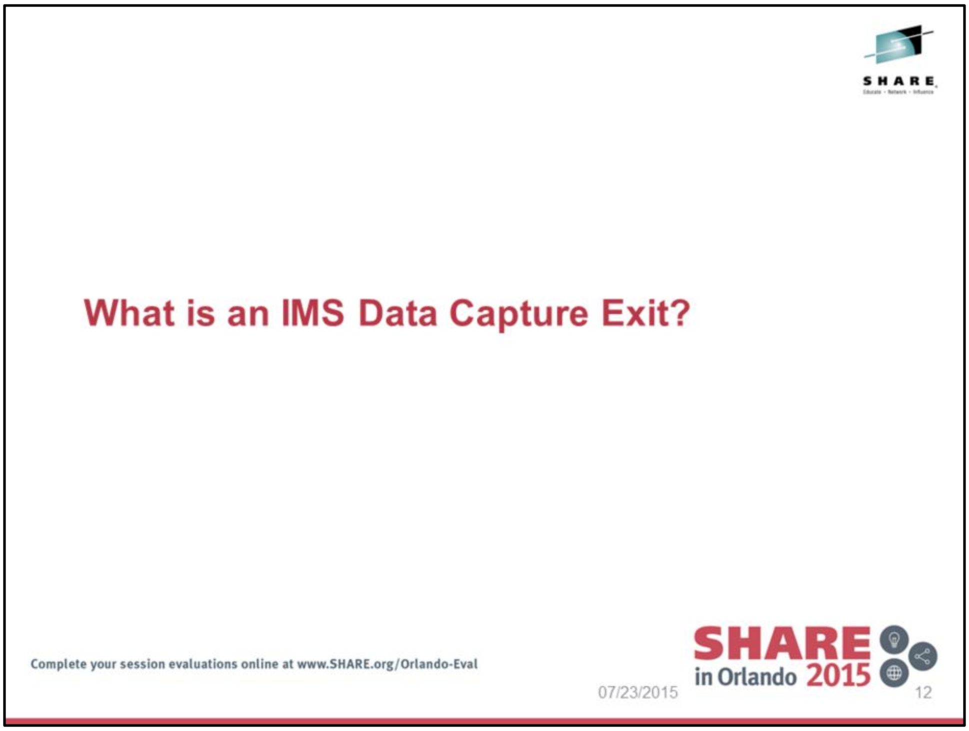
SHARE in Orlando 2015



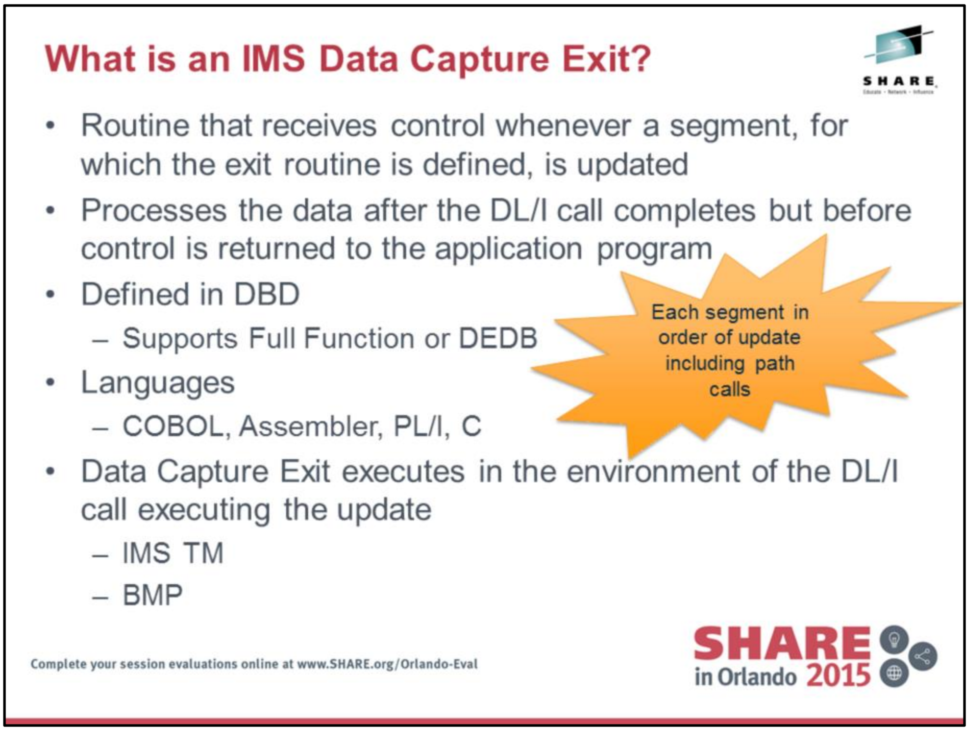
Overview – Synchronization Options

- Product option
  - Considered IBM DataPropagator (asynchronous)
  - No other planned or projected synchronization uses
  - Too expensive for single use
- Custom option (synchronous)
  - IMS Data Capture Exits
  - Custom exit for each IMS database
- Custom option (asynchronous)
  - DBDGEN utility logging option
  - Process log records for changed data
- Custom option
  - Dual updates from application to both IMS and DB2
- **IMS Data Capture Exit approach chosen**

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



What is an IMS Data Capture Exit?

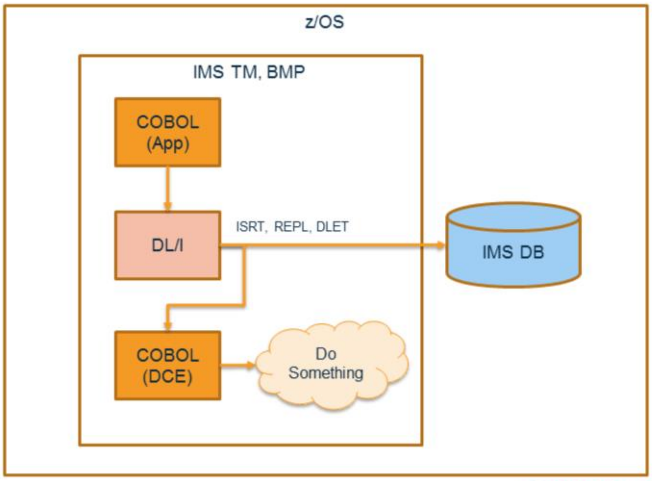


What is an IMS Data Capture Exit?

- Routine that receives control whenever a segment, for which the exit routine is defined, is updated
- Processes the data after the DL/I call completes but before control is returned to the application program
- Defined in DBD
  - Supports Full Function or DEDB
- Languages
  - COBOL, Assembler, PL/I, C
- Data Capture Exit executes in the environment of the DL/I call executing the update
  - IMS TM
  - BMP

Each segment in order of update including path calls

What is an IMS Data Capture Exit?



z/OS

IMS TM, BMP

COBOL (App)

DL/I

ISRT, REPL, DLET

IMS DB

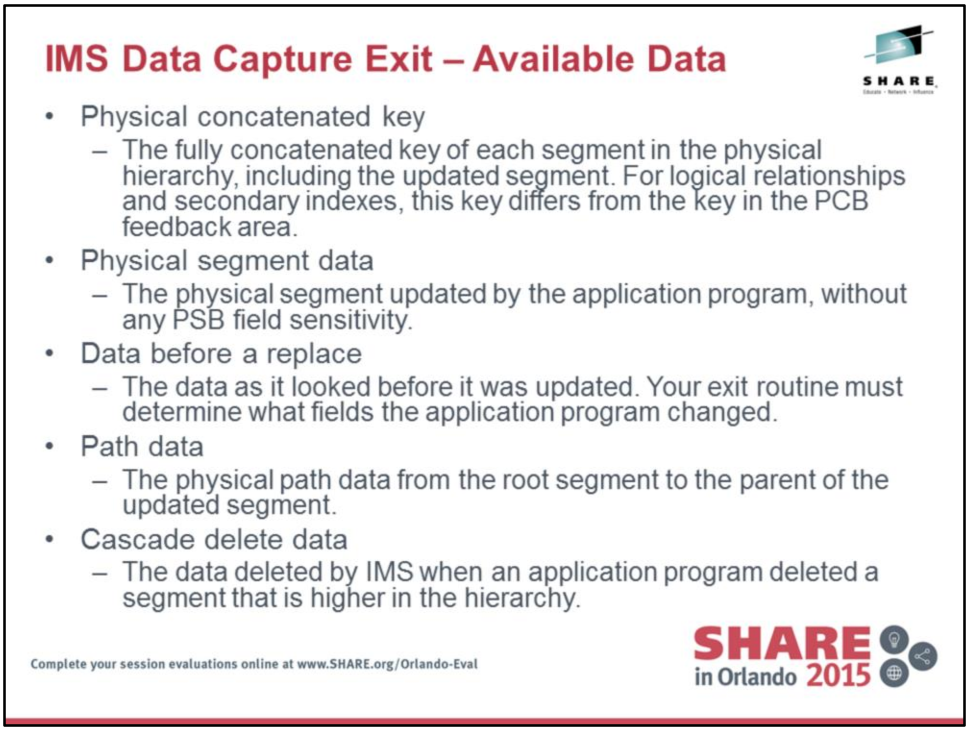
COBOL (DCE)

Do Something

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

SHARE in Orlando 2015

07/23/2015

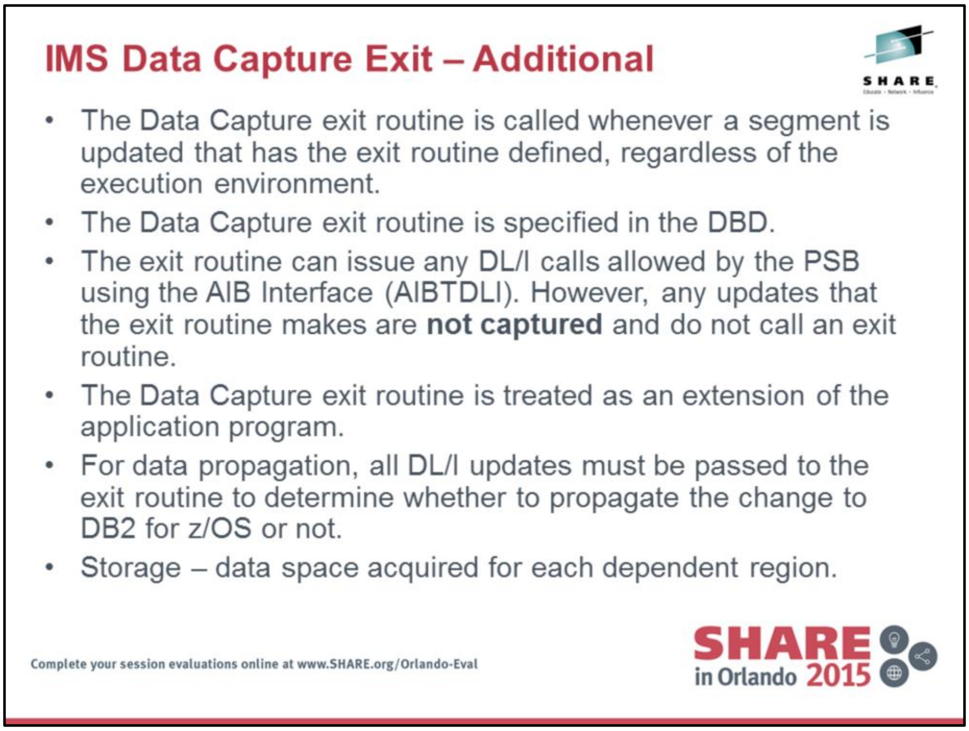


IMS Data Capture Exit – Available Data

- Physical concatenated key
  - The fully concatenated key of each segment in the physical hierarchy, including the updated segment. For logical relationships and secondary indexes, this key differs from the key in the PCB feedback area.
- Physical segment data
  - The physical segment updated by the application program, without any PSB field sensitivity.
- Data before a replace
  - The data as it looked before it was updated. Your exit routine must determine what fields the application program changed.
- Path data
  - The physical path data from the root segment to the parent of the updated segment.
- Cascade delete data
  - The data deleted by IMS when an application program deleted a segment that is higher in the hierarchy.

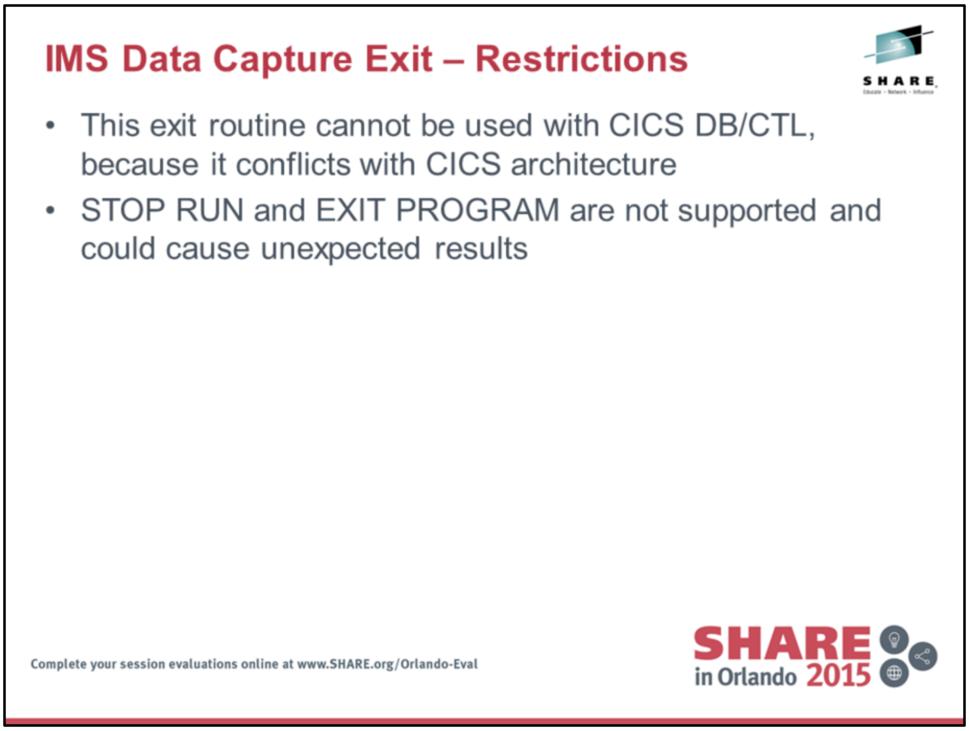
Complete your session evaluations online at www.SHARE.org/Orlando-Eval

SHARE in Orlando 2015



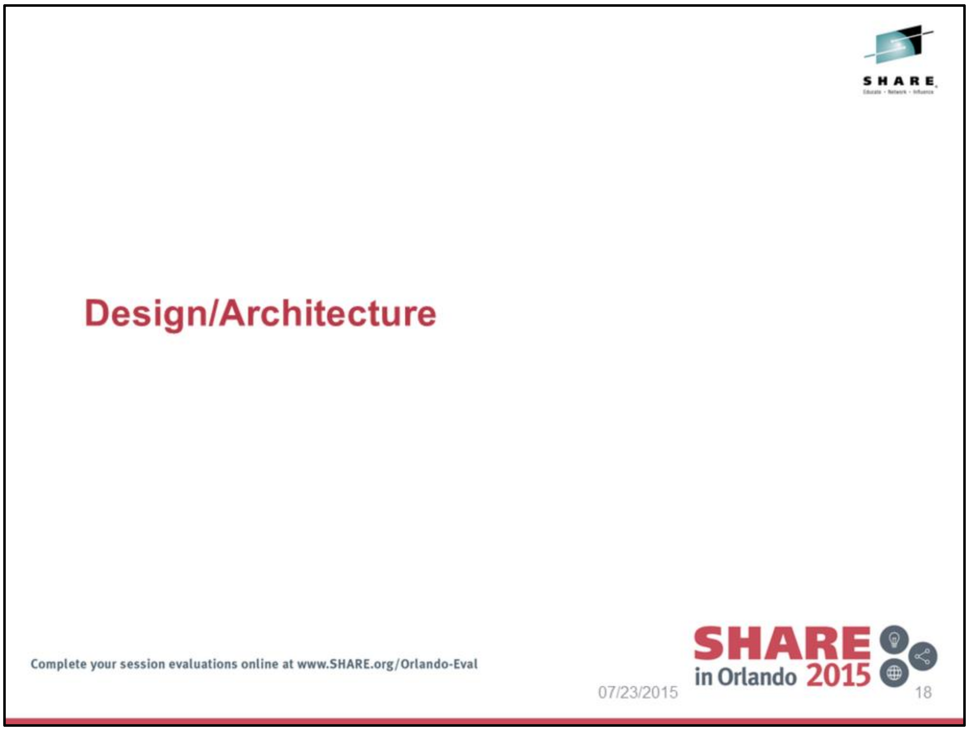
IMS Data Capture Exit – Additional

- The Data Capture exit routine is called whenever a segment is updated that has the exit routine defined, regardless of the execution environment.
- The Data Capture exit routine is specified in the DBD.
- The exit routine can issue any DL/I calls allowed by the PSB using the AIB Interface (AIBTDLI). However, any updates that the exit routine makes are **not captured** and do not call an exit routine.
- The Data Capture exit routine is treated as an extension of the application program.
- For data propagation, all DL/I updates must be passed to the exit routine to determine whether to propagate the change to DB2 for z/OS or not.
- Storage – data space acquired for each dependent region.

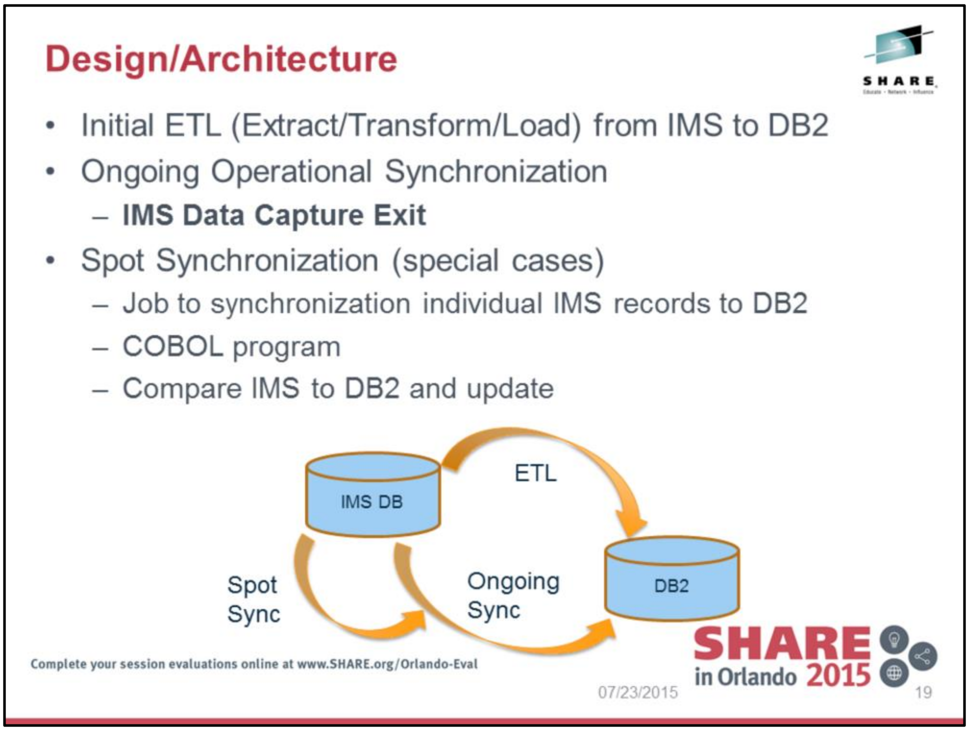


IMS Data Capture Exit – Restrictions

- This exit routine cannot be used with CICS DB/CTL, because it conflicts with CICS architecture
- STOP RUN and EXIT PROGRAM are not supported and could cause unexpected results



Design/Architecture

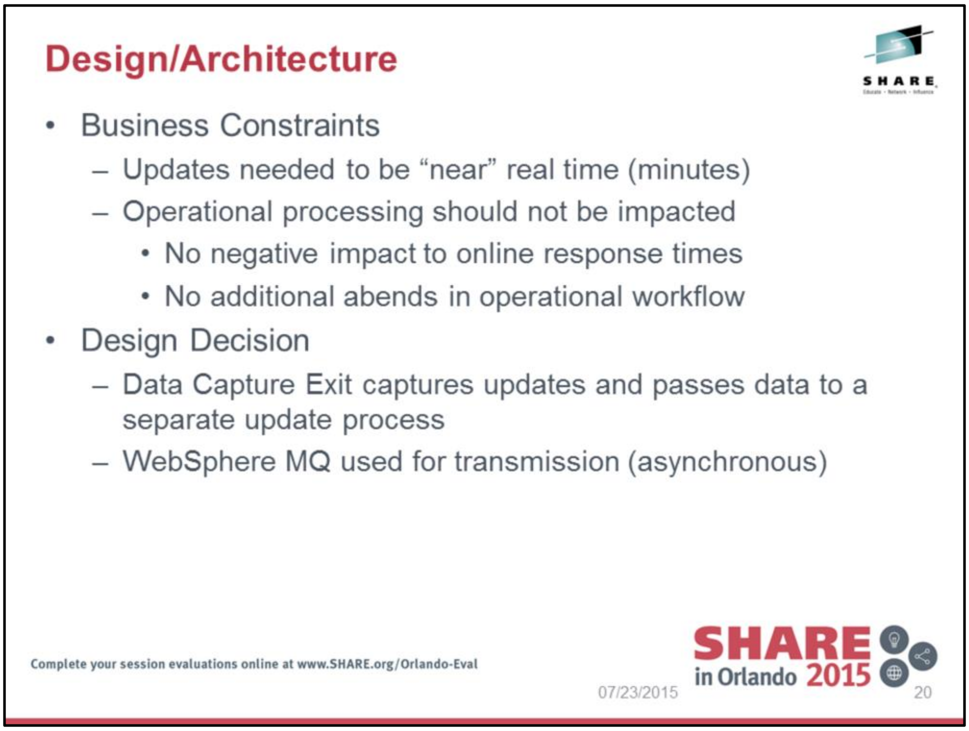


Design/Architecture

- Initial ETL (Extract/Transform/Load) from IMS to DB2
- Ongoing Operational Synchronization
  - **IMS Data Capture Exit**
- Spot Synchronization (special cases)
  - Job to synchronization individual IMS records to DB2
  - COBOL program
  - Compare IMS to DB2 and update

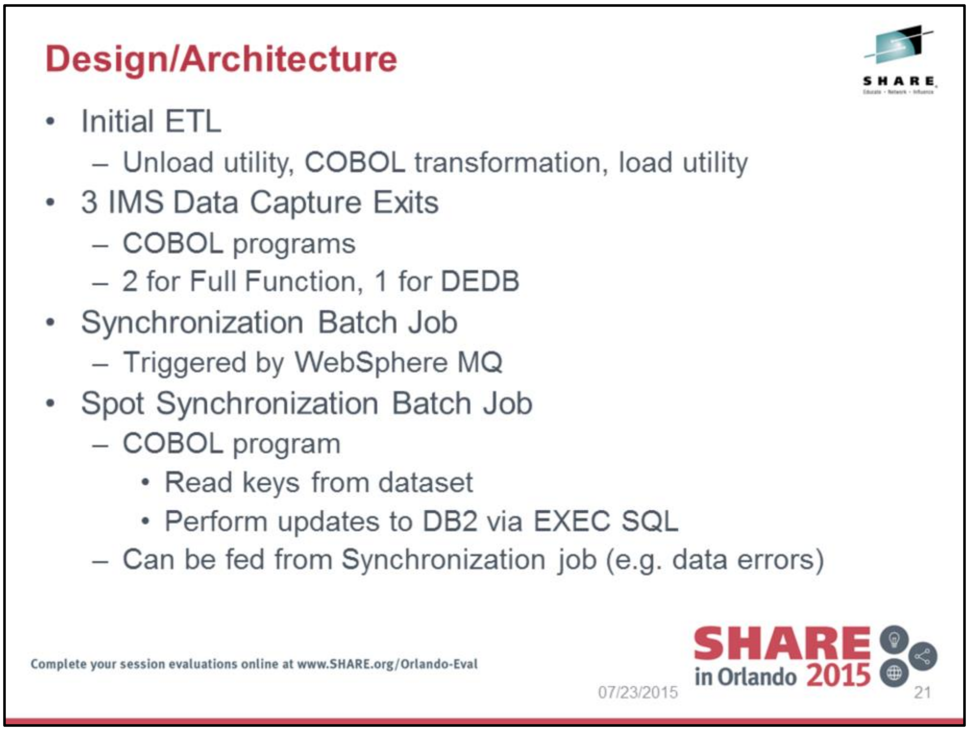
Complete your session evaluations online at www.SHARE.org/Orlando-Eval

SHARE in Orlando 2015



Design/Architecture

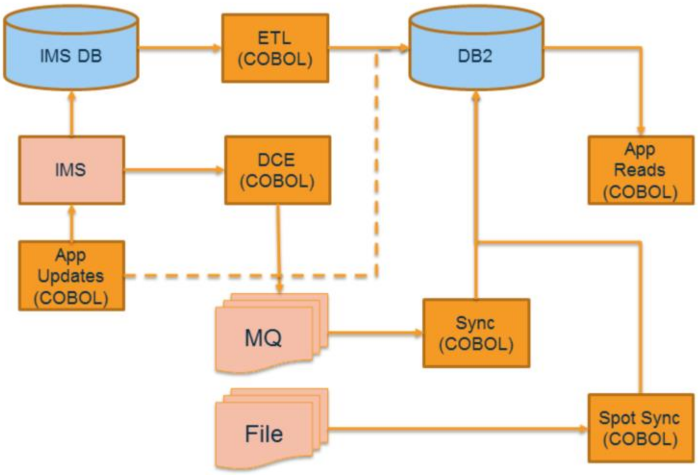
- Business Constraints
  - Updates needed to be “near” real time (minutes)
  - Operational processing should not be impacted
    - No negative impact to online response times
    - No additional abends in operational workflow
- Design Decision
  - Data Capture Exit captures updates and passes data to a separate update process
  - WebSphere MQ used for transmission (asynchronous)



Design/Architecture

- Initial ETL
  - Unload utility, COBOL transformation, load utility
- 3 IMS Data Capture Exits
  - COBOL programs
  - 2 for Full Function, 1 for DEDB
- Synchronization Batch Job
  - Triggered by WebSphere MQ
- Spot Synchronization Batch Job
  - COBOL program
    - Read keys from dataset
    - Perform updates to DB2 via EXEC SQL
  - Can be fed from Synchronization job (e.g. data errors)

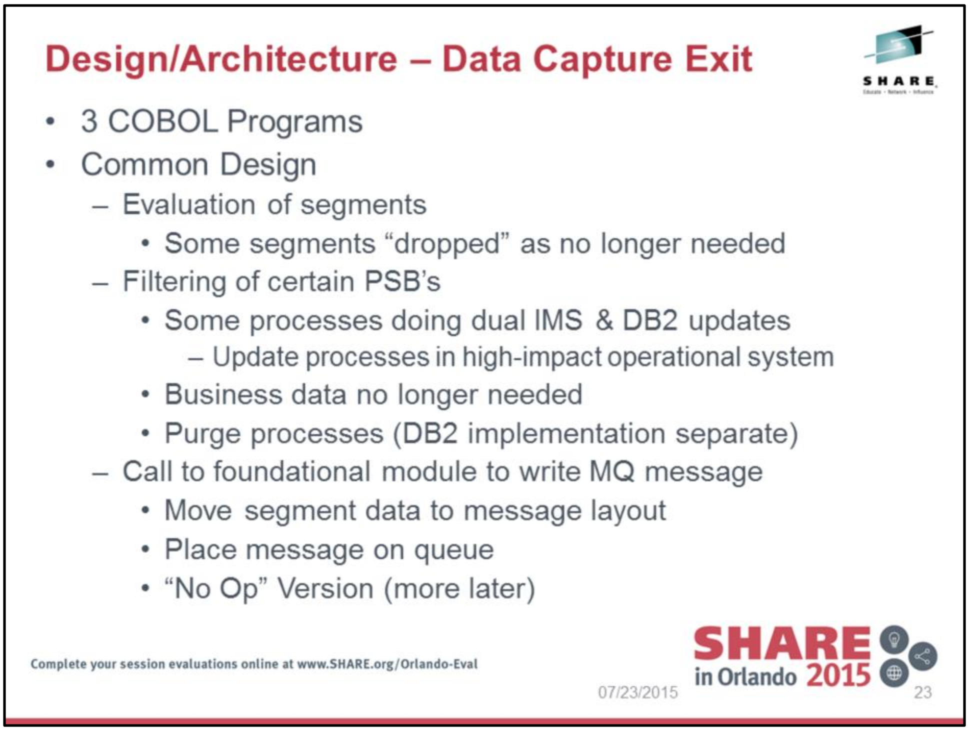
Design/Architecture



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

07/23/2015

SHARE in Orlando 2015



Design/Architecture – Data Capture Exit

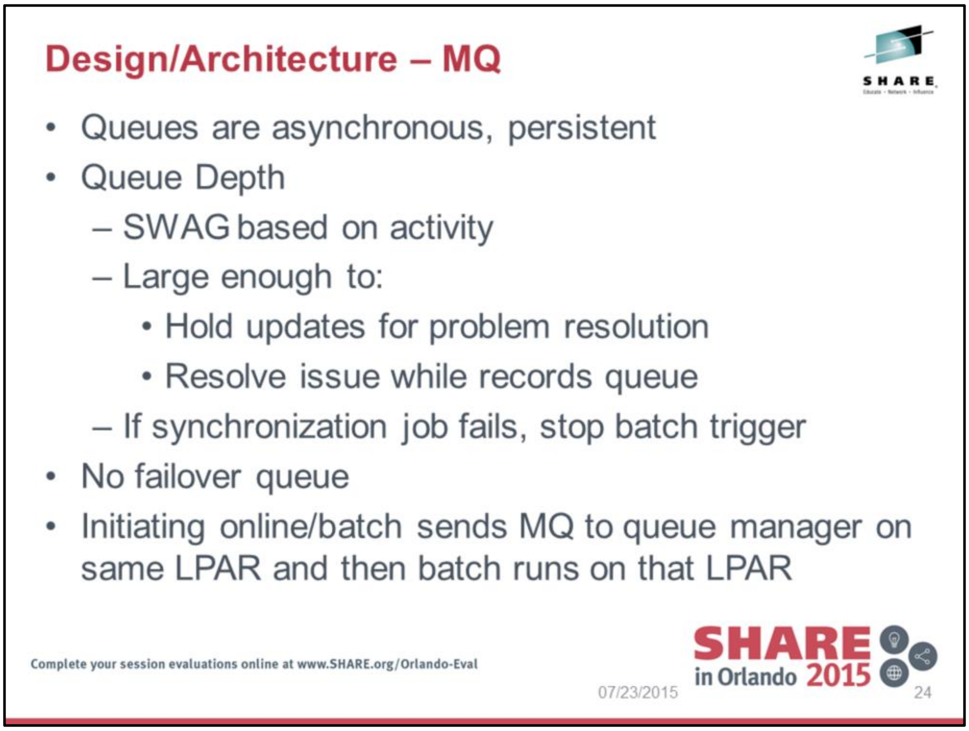
- 3 COBOL Programs
- Common Design
  - Evaluation of segments
    - Some segments “dropped” as no longer needed
  - Filtering of certain PSB’s
    - Some processes doing dual IMS & DB2 updates
      - Update processes in high-impact operational system
    - Business data no longer needed
    - Purge processes (DB2 implementation separate)
  - Call to foundational module to write MQ message
    - Move segment data to message layout
    - Place message on queue
    - “No Op” Version (more later)

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

07/23/2015

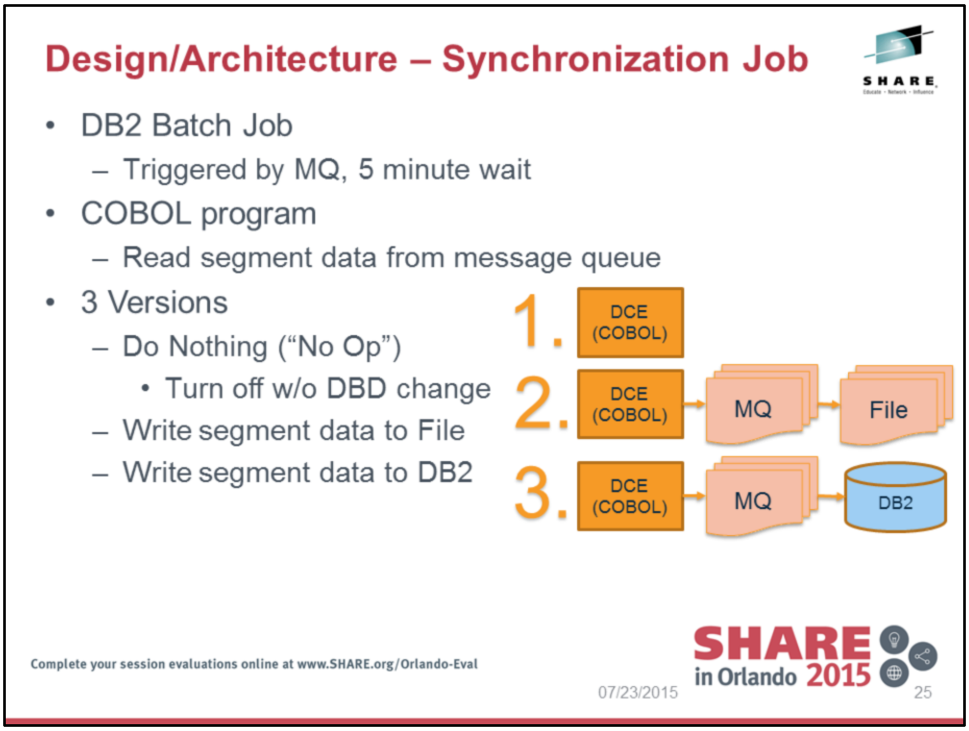
SHARE in Orlando 2015

23



Design/Architecture – MQ

- Queues are asynchronous, persistent
- Queue Depth
  - SWAG based on activity
  - Large enough to:
    - Hold updates for problem resolution
    - Resolve issue while records queue
  - If synchronization job fails, stop batch trigger
- No failover queue
- Initiating online/batch sends MQ to queue manager on same LPAR and then batch runs on that LPAR



Design/Architecture – Synchronization Job

- DB2 Batch Job
  - Triggered by MQ, 5 minute wait
- COBOL program
  - Read segment data from message queue
- 3 Versions
  - Do Nothing ("No Op")
    - Turn off w/o DBD change
  - Write segment data to File
  - Write segment data to DB2

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

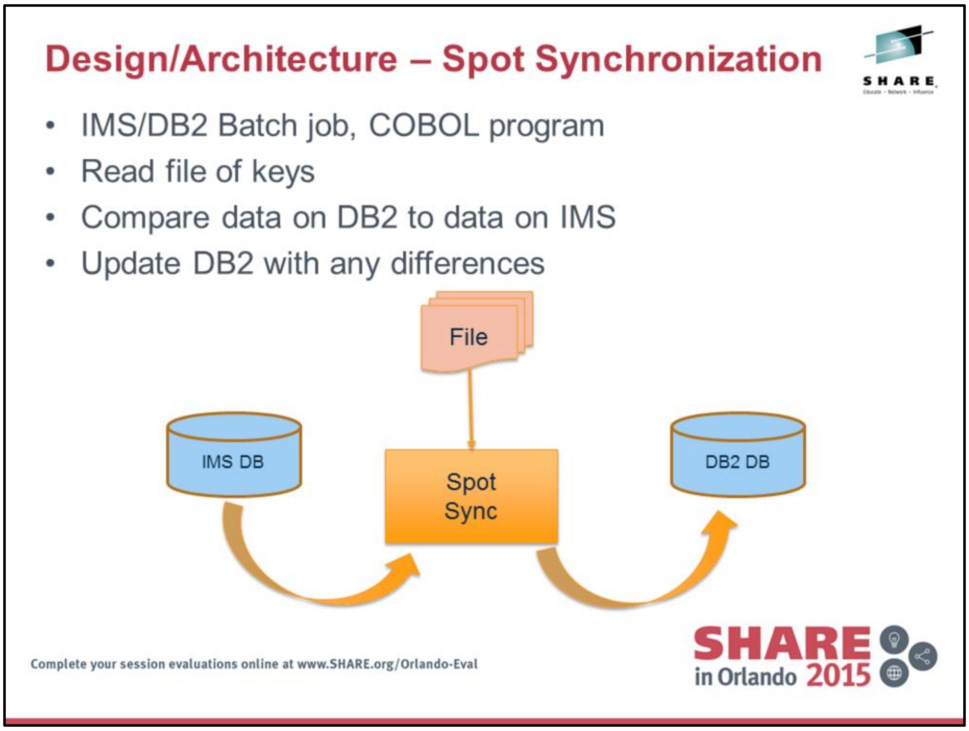
SHARE in Orlando 2015

07/23/2015

25

The "No Op" version allowed the synchronization process to be turned off without the need for a DBD change. The Data Capture Exits would still fire, but would not feed records to DB2.

The DB2 batch job was triggered via WebSphere MQ when message received on the associated queue. A 5-minute wait was added to keep the job resident if no messages were received to avoid constant job scheduling.

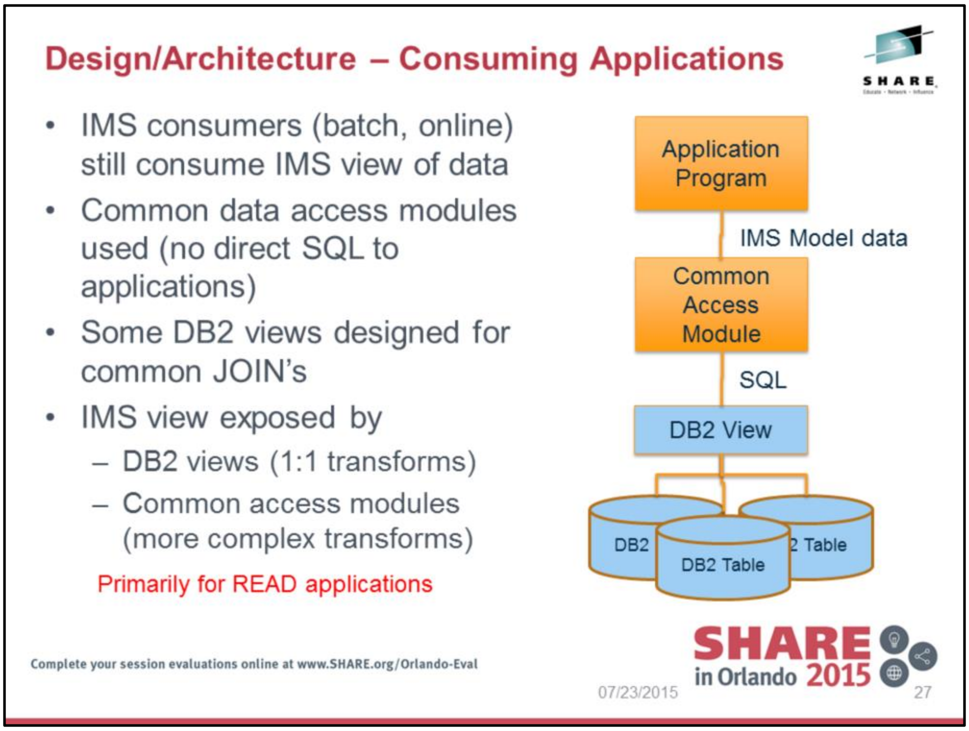


Design/Architecture – Spot Synchronization

- IMS/DB2 Batch job, COBOL program
- Read file of keys
- Compare data on DB2 to data on IMS
- Update DB2 with any differences

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

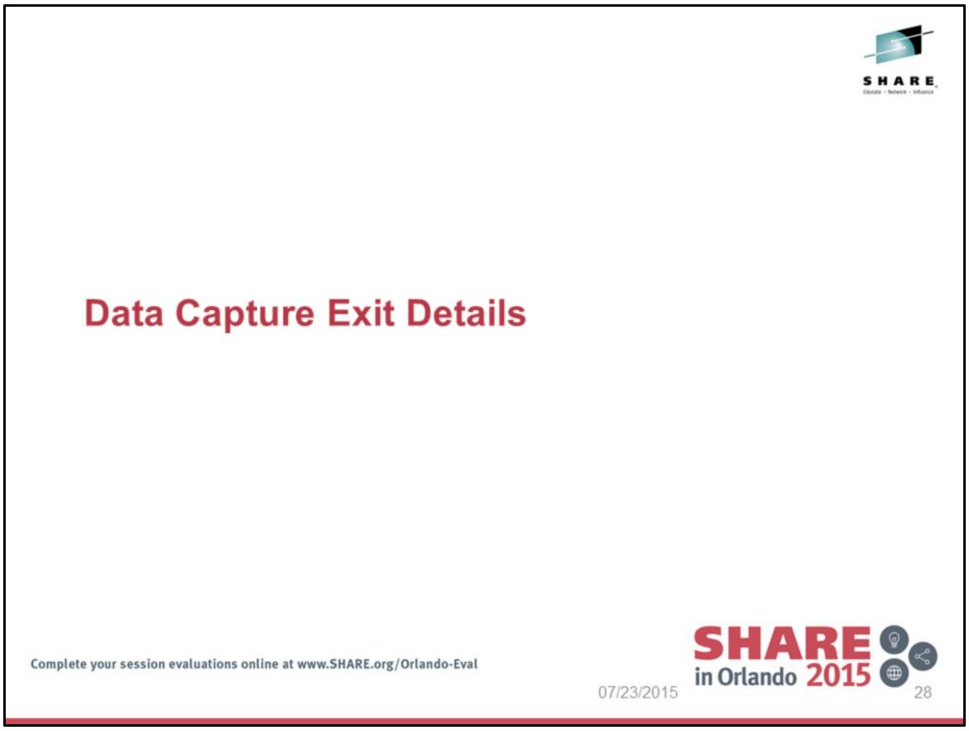
SHARE in Orlando 2015



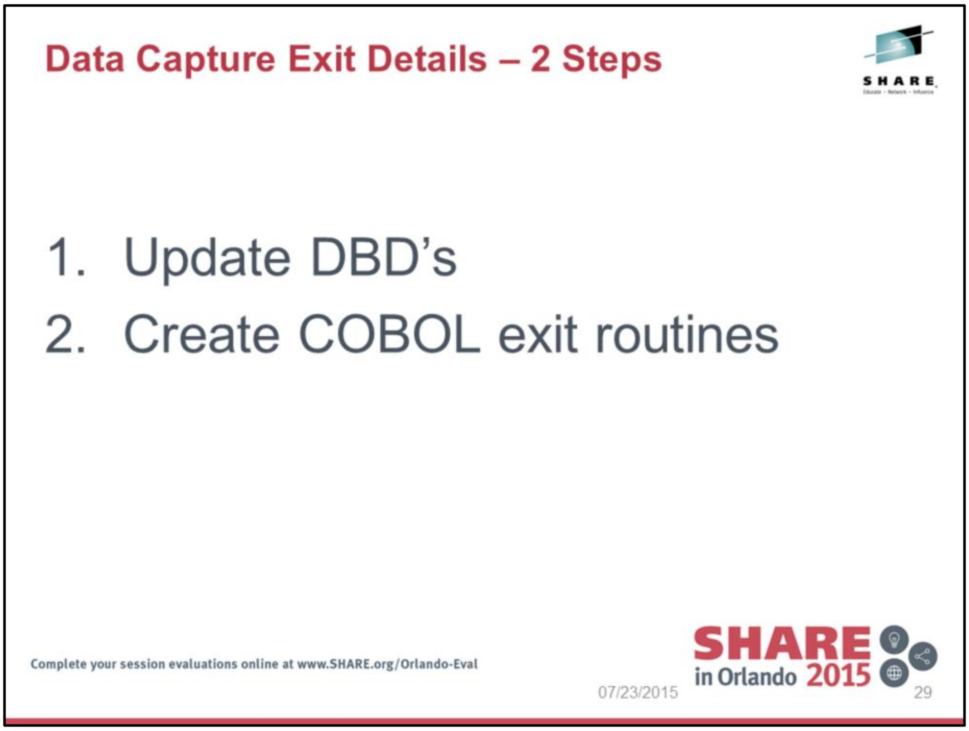
Design/Architecture – Consuming Applications

- IMS consumers (batch, online) still consume IMS view of data
- Common data access modules used (no direct SQL to applications)
- Some DB2 views designed for common JOIN's
- IMS view exposed by
  - DB2 views (1:1 transforms)
  - Common access modules (more complex transforms)

Primarily for READ applications

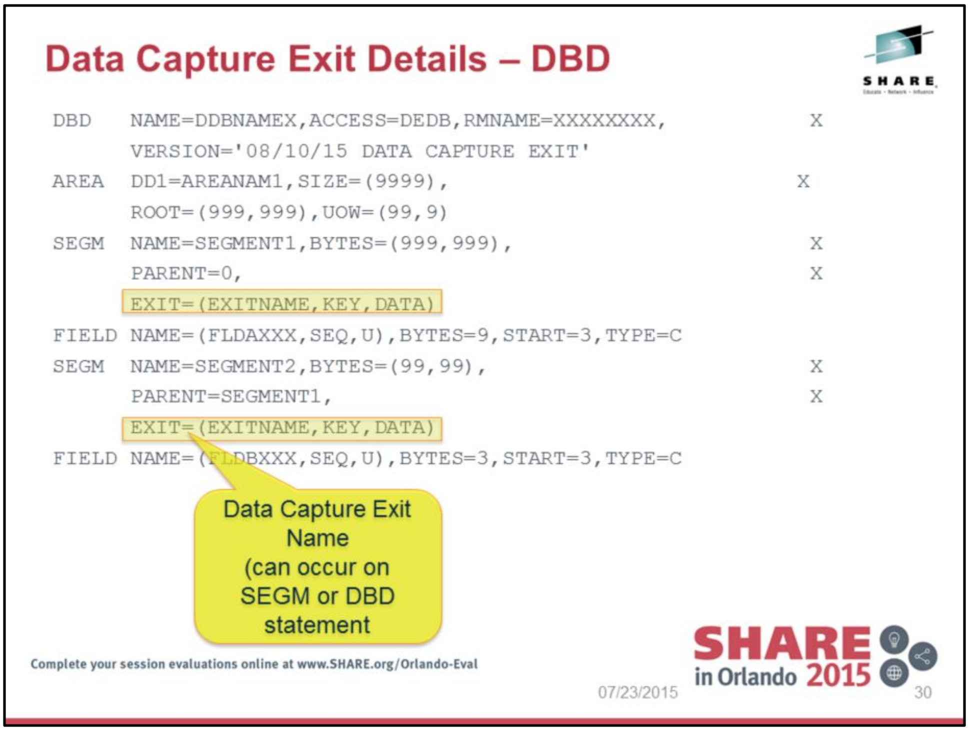


Data Capture Exit Details



Data Capture Exit Details – 2 Steps

1. Update DBD's
2. Create COBOL exit routines



Data Capture Exit Details – DBD

```
DBD   NAME=DDBNAMEX,ACCESS=DEDB,RMNAME=XXXXXXXX,            X
      VERSION='08/10/15 DATA CAPTURE EXIT'
AREA  DD1=AREANAM1,SIZE=(9999),                            X
      ROOT=(999,999),UOW=(99,9)
SEGM  NAME=SEGMENT1,BYTES=(999,999),                        X
      PARENT=0,                                             X
      EXIT=(EXITNAME,KEY,DATA)
FIELD NAME=(FLDAXXX,SEQ,U),BYTES=9,START=3,TYPE=C
SEGM  NAME=SEGMENT2,BYTES=(99,99),                          X
      PARENT=SEGMENT1,                                      X
      EXIT=(EXITNAME,KEY,DATA)
FIELD NAME=(FLDBXXX,SEQ,U),BYTES=3,START=3,TYPE=C
```

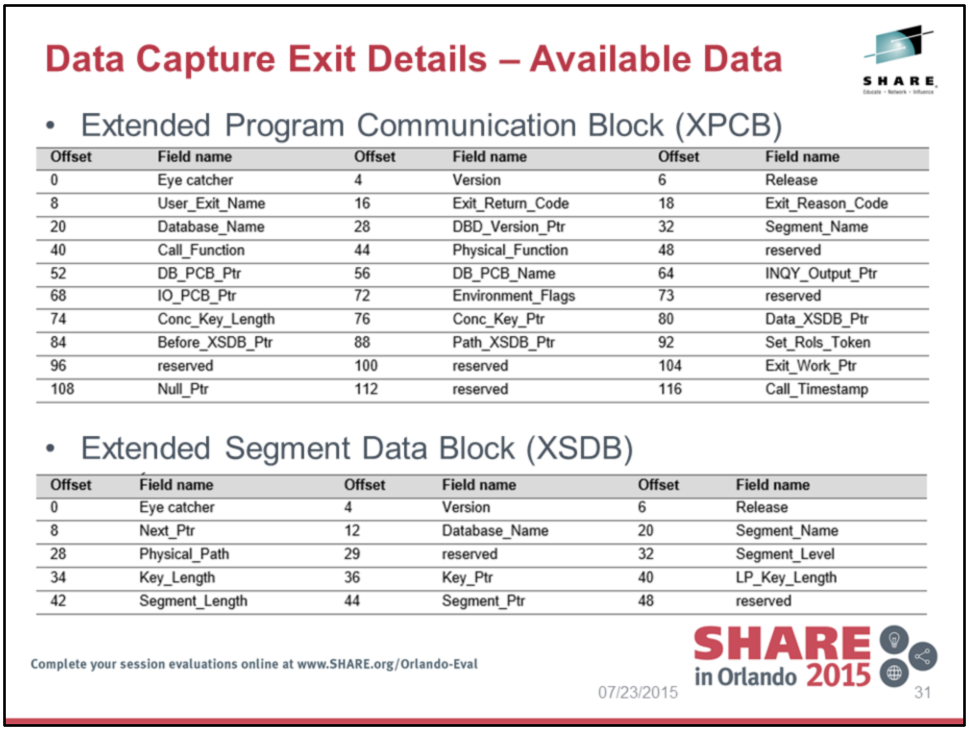
Data Capture Exit Name
(can occur on SEGM or DBD statement

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

07/23/2015

30

The EXIT can be defined on the SEGM (segment) statement or DBD statement. If specified on the DBD statement, the exit occurs for ALL segment types on the database.



Data Capture Exit Details – Available Data

- Extended Program Communication Block (XPCB)

Offset	Field name	Offset	Field name	Offset	Field name
0	Eye catcher	4	Version	6	Release
8	User_Exit_Name	16	Exit_Return_Code	18	Exit_Reason_Code
20	Database_Name	28	DBD_Version_Ptr	32	Segment_Name
40	Call_Function	44	Physical_Function	48	reserved
52	DB_PCB_Ptr	56	DB_PCB_Name	64	INQY_Output_Ptr
68	IO_PCB_Ptr	72	Environment_Flags	73	reserved
74	Conc_Key_Length	76	Conc_Key_Ptr	80	Data_XSDB_Ptr
84	Before_XSDB_Ptr	88	Path_XSDB_Ptr	92	Set_Rols_Token
96	reserved	100	reserved	104	Exit_Work_Ptr
108	Null_Ptr	112	reserved	116	Call_Timestamp

- Extended Segment Data Block (XSDB)

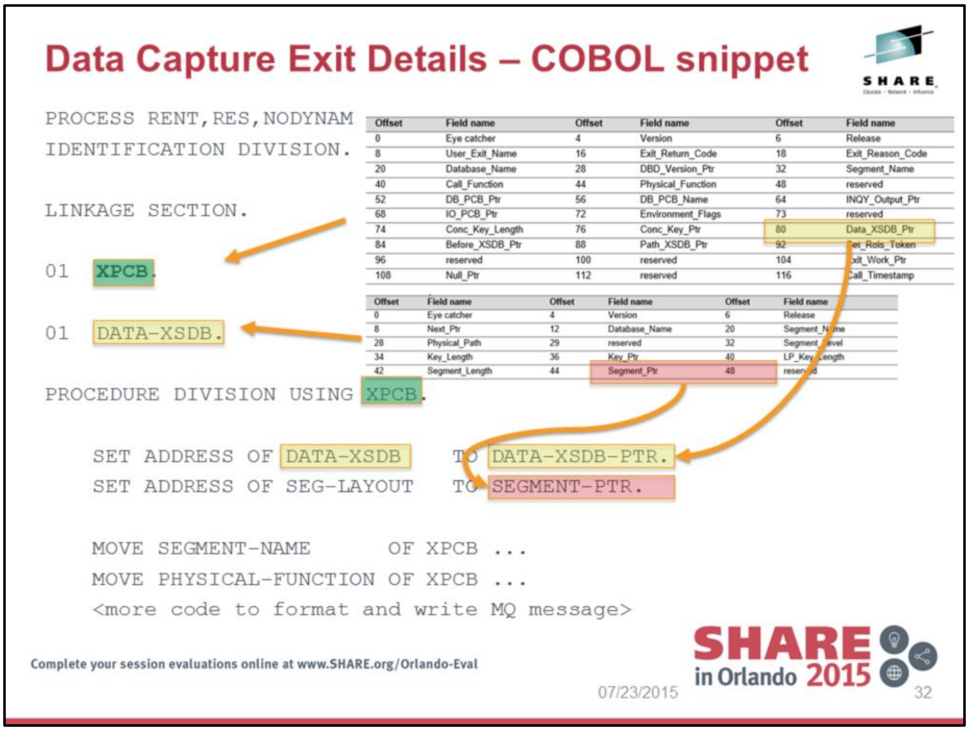
Offset	Field name	Offset	Field name	Offset	Field name
0	Eye catcher	4	Version	6	Release
8	Next_Ptr	12	Database_Name	20	Segment_Name
28	Physical_Path	29	reserved	32	Segment_Level
34	Key_Length	36	Key_Ptr	40	LP_Key_Length
42	Segment_Length	44	Segment_Ptr	48	reserved

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

SHARE in Orlando 2015

07/23/2015

31



Data Capture Exit Details – COBOL snippet

```
PROCESS RENT,RES,NODYNAM
IDENTIFICATION DIVISION.

LINKAGE SECTION.

01  XPCB.

01  DATA-XSDB.

PROCEDURE DIVISION USING XPCB.

    SET ADDRESS OF DATA-XSDB    TO DATA-XSDB-PTR.
    SET ADDRESS OF SEG-LAYOUT   TO SEGMENT-PTR.

    MOVE SEGMENT-NAME      OF XPCB ...
    MOVE PHYSICAL-FUNCTION OF XPCB ...
    <more code to format and write MQ message>
```

Offset	Field name	Offset	Field name	Offset	Field name
0	Eye catcher	4	Version	6	Release
8	User_Exit_Name	16	Exit_Return_Code	18	Exit_Reason_Code
20	Database_Name	28	DBD_Version_Ptr	32	Segment_Name
40	Call_Function	44	Physical_Function	48	reserved
52	DB_PCB_Ptr	56	DB_PCB_Name	64	INQY_Output_Ptr
68	IO_PCB_Ptr	72	Environment_Flags	73	reserved
74	Conc_Key_Length	76	Conc_Key_Ptr	80	Data_XSDB_Ptr
84	Before_XSDB_Ptr	88	Path_XSDB_Ptr	92	Set_Rols_Token
96	reserved	100	reserved	104	Exit_Work_Ptr
108	Null_Ptr	112	reserved	116	Call_Timestamp

Offset	Field name	Offset	Field name	Offset	Field name
0	Eye catcher	4	Version	6	Release
8	Next_Ptr	12	Database_Name	20	Segment_Name
28	Physical_Path	29	reserved	32	Segment_Level
34	Key_Length	36	Key_Ptr	40	LP_Key_Length
42	Segment_Length	44	Segment_Ptr	48	reserved

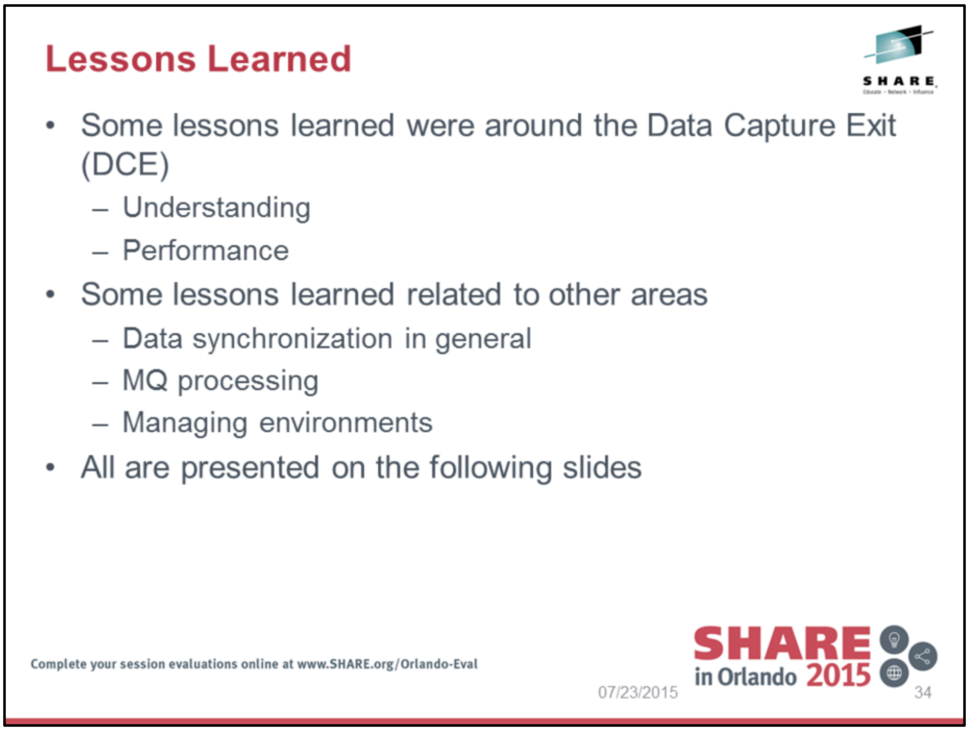
Complete your session evaluations online at www.SHARE.org/Orlando-Eval

SHARE in Orlando 2015

07/23/2015

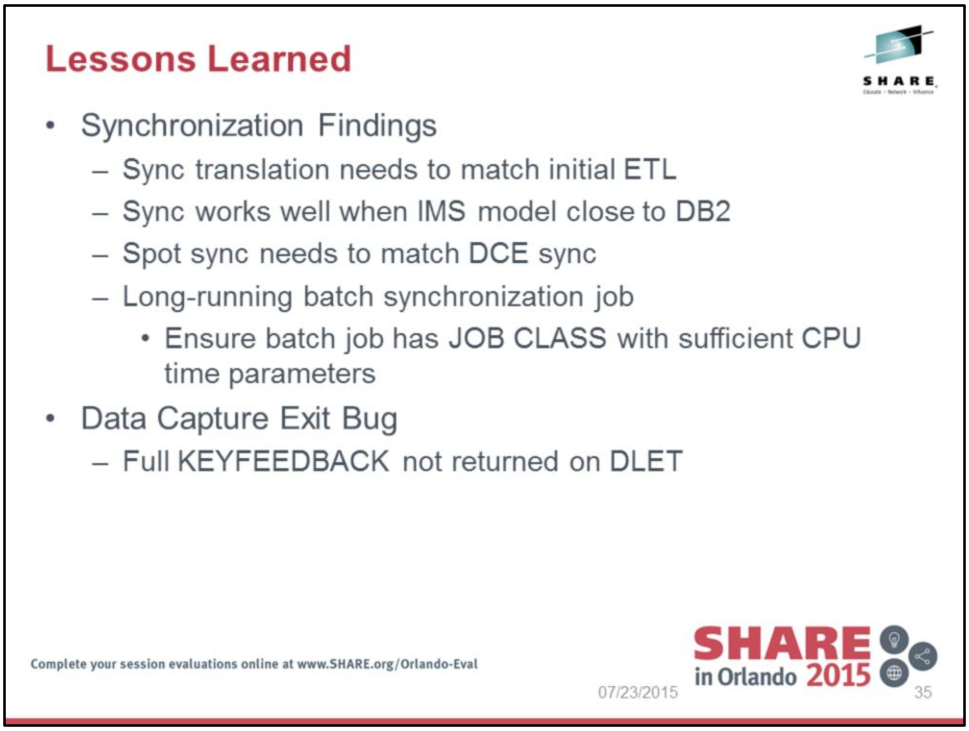
32

Lessons Learned



Lessons Learned

- Some lessons learned were around the Data Capture Exit (DCE)
  - Understanding
  - Performance
- Some lessons learned related to other areas
  - Data synchronization in general
  - MQ processing
  - Managing environments
- All are presented on the following slides



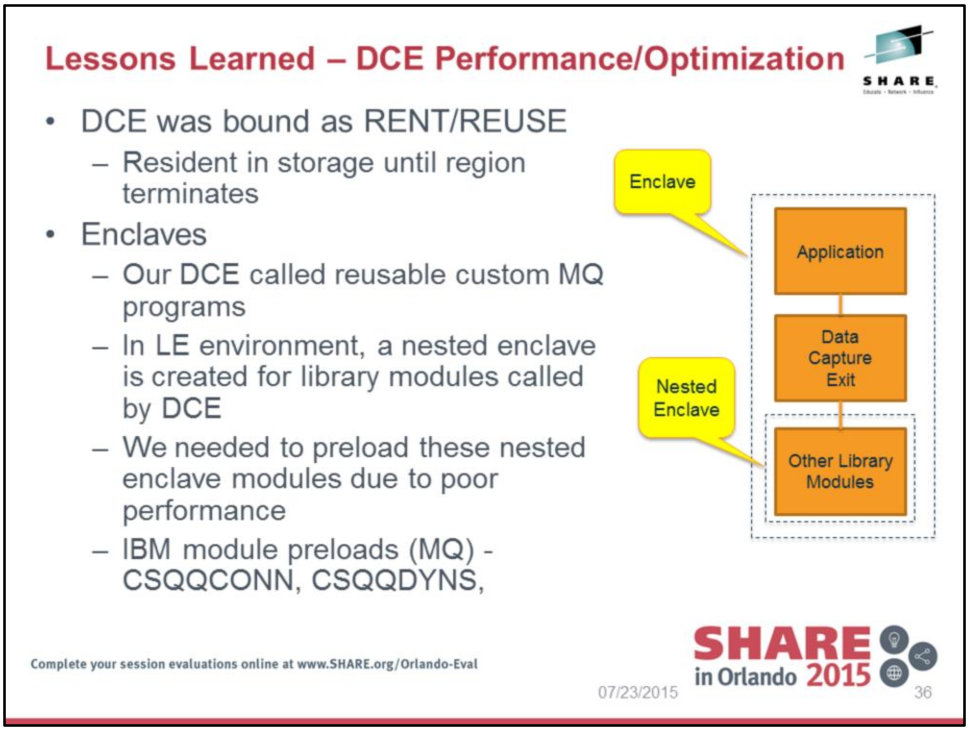
Lessons Learned

- Synchronization Findings
  - Sync translation needs to match initial ETL
  - Sync works well when IMS model close to DB2
  - Spot sync needs to match DCE sync
  - Long-running batch synchronization job
    - Ensure batch job has JOB CLASS with sufficient CPU time parameters
- Data Capture Exit Bug
  - Full KEYFEEDBACK not returned on DLET

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

07/23/2015

SHARE in Orlando 2015



Lessons Learned – DCE Performance/Optimization

- DCE was bound as RENT/REUSE
  - Resident in storage until region terminates
- Enclaves
  - Our DCE called reusable custom MQ programs
  - In LE environment, a nested enclave is created for library modules called by DCE
  - We needed to preload these nested enclave modules due to poor performance
  - IBM module preloads (MQ) - CSQQCONN, CSQQDYNS,

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

SHARE in Orlando 2015

07/23/2015

36

Nested Enclave:

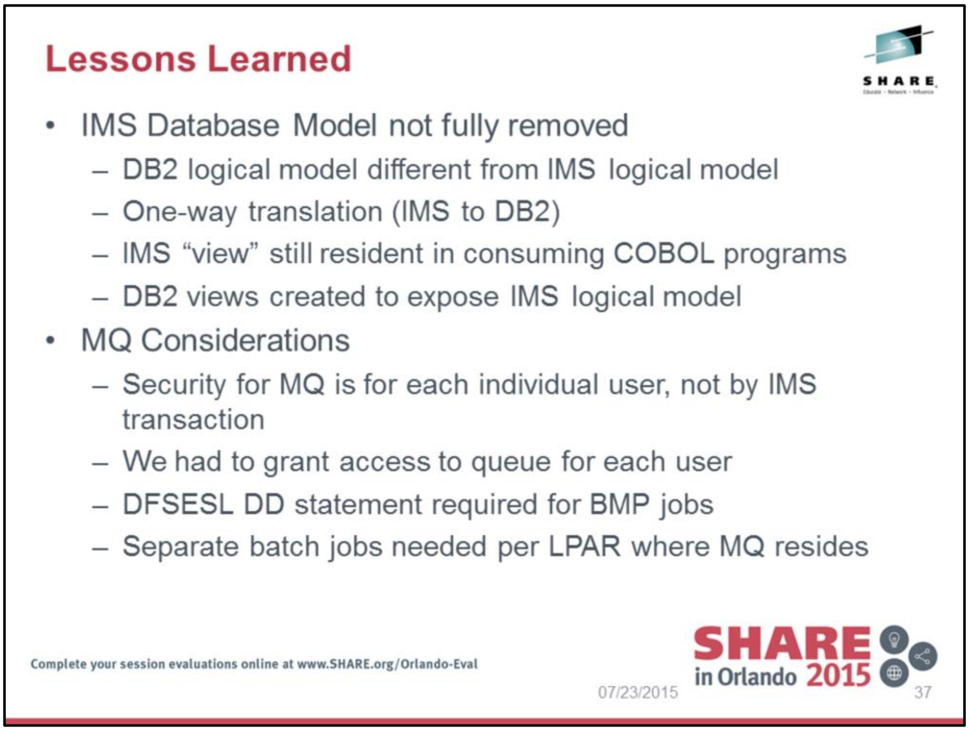
- IGZERREO - NESTENC option changed from NO to YES

From IBM Knowledge Center:

Loading the routine

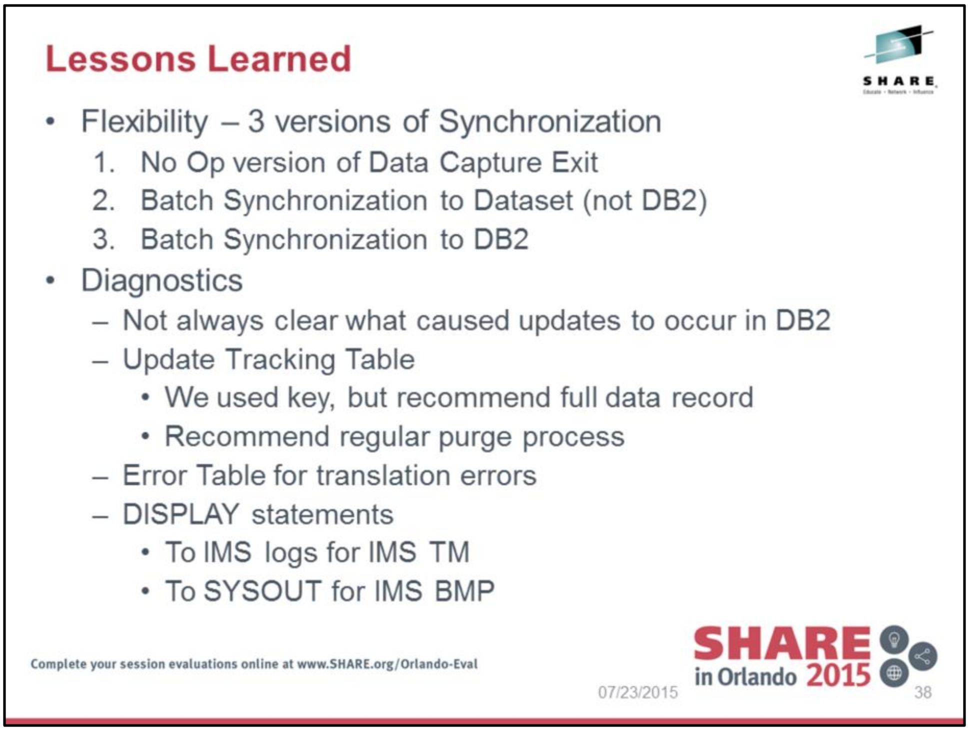
IMS loads the exit routine the first time IMS calls it; preloading the exit routine is not necessary. However, runtime library routines used by high-level languages should be preloaded. After abnormal termination in an IMS Fast Path region (IFP) or in a message processing region (MPP), the exit routine is deleted and must be reloaded. The exit routine must be reloaded when:

- A pseudo or standard abend of the application that is running in the region occurs (regardless of whether the region itself abends along with the application).
- The data capture routine gets an XPCB return code of 16.



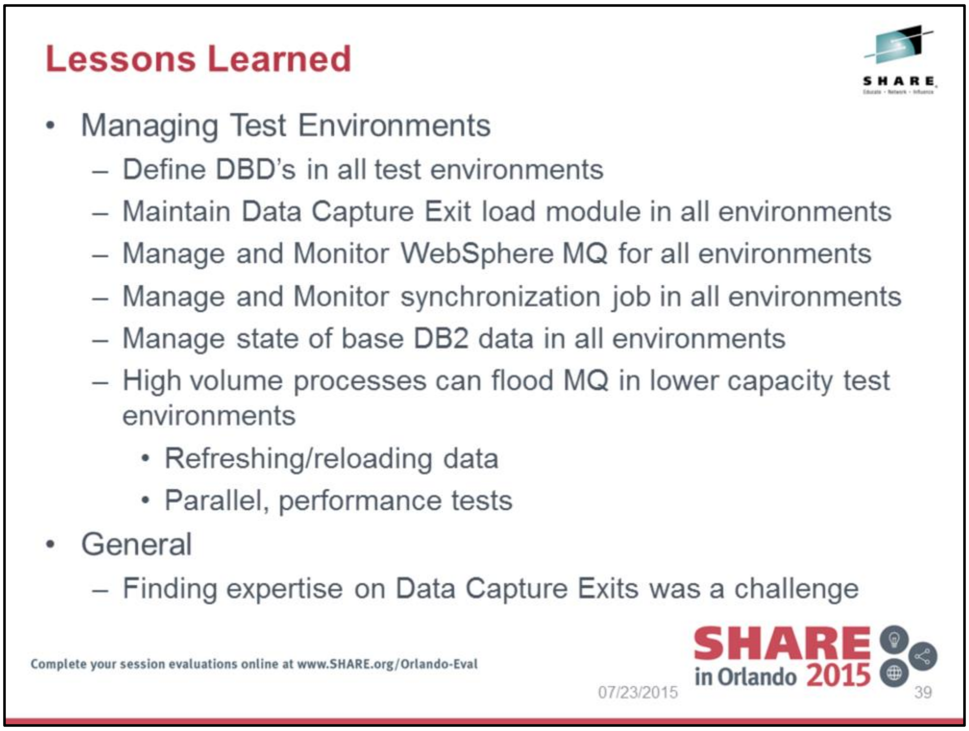
Lessons Learned

- IMS Database Model not fully removed
  - DB2 logical model different from IMS logical model
  - One-way translation (IMS to DB2)
  - IMS “view” still resident in consuming COBOL programs
  - DB2 views created to expose IMS logical model
- MQ Considerations
  - Security for MQ is for each individual user, not by IMS transaction
  - We had to grant access to queue for each user
  - DFSESL DD statement required for BMP jobs
  - Separate batch jobs needed per LPAR where MQ resides



Lessons Learned

- Flexibility – 3 versions of Synchronization
  1. No Op version of Data Capture Exit
  2. Batch Synchronization to Dataset (not DB2)
  3. Batch Synchronization to DB2
- Diagnostics
  - Not always clear what caused updates to occur in DB2
  - Update Tracking Table
    - We used key, but recommend full data record
    - Recommend regular purge process
  - Error Table for translation errors
  - DISPLAY statements
    - To IMS logs for IMS TM
    - To SYSOUT for IMS BMP



Lessons Learned

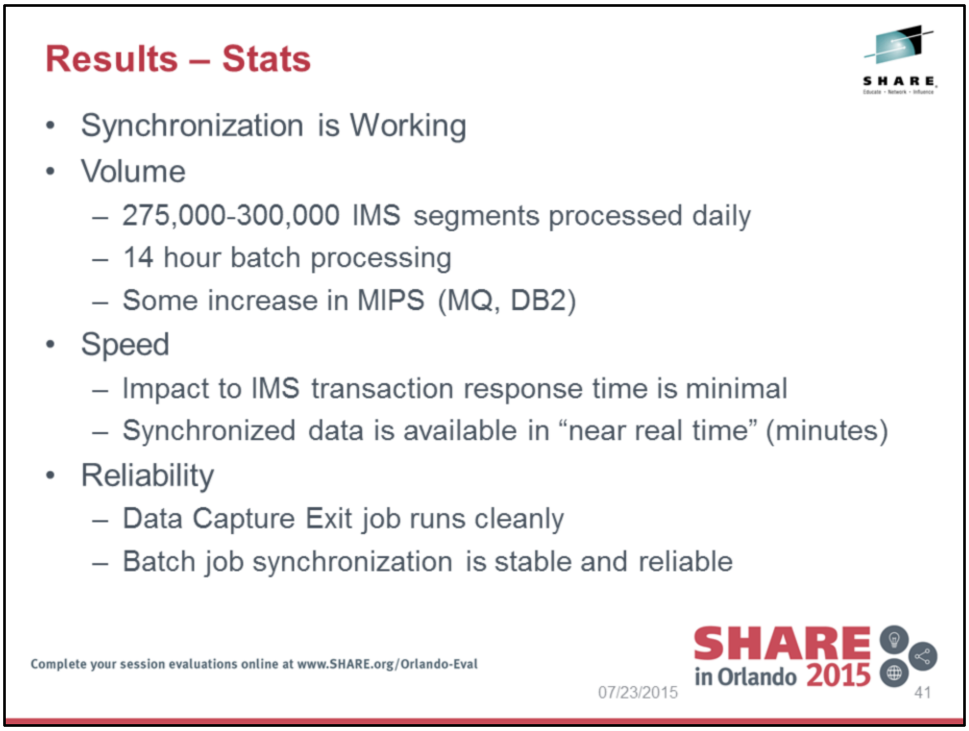
- Managing Test Environments
  - Define DBD's in all test environments
  - Maintain Data Capture Exit load module in all environments
  - Manage and Monitor WebSphere MQ for all environments
  - Manage and Monitor synchronization job in all environments
  - Manage state of base DB2 data in all environments
  - High volume processes can flood MQ in lower capacity test environments
    - Refreshing/reloading data
    - Parallel, performance tests
- General
  - Finding expertise on Data Capture Exits was a challenge

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

07/23/2015

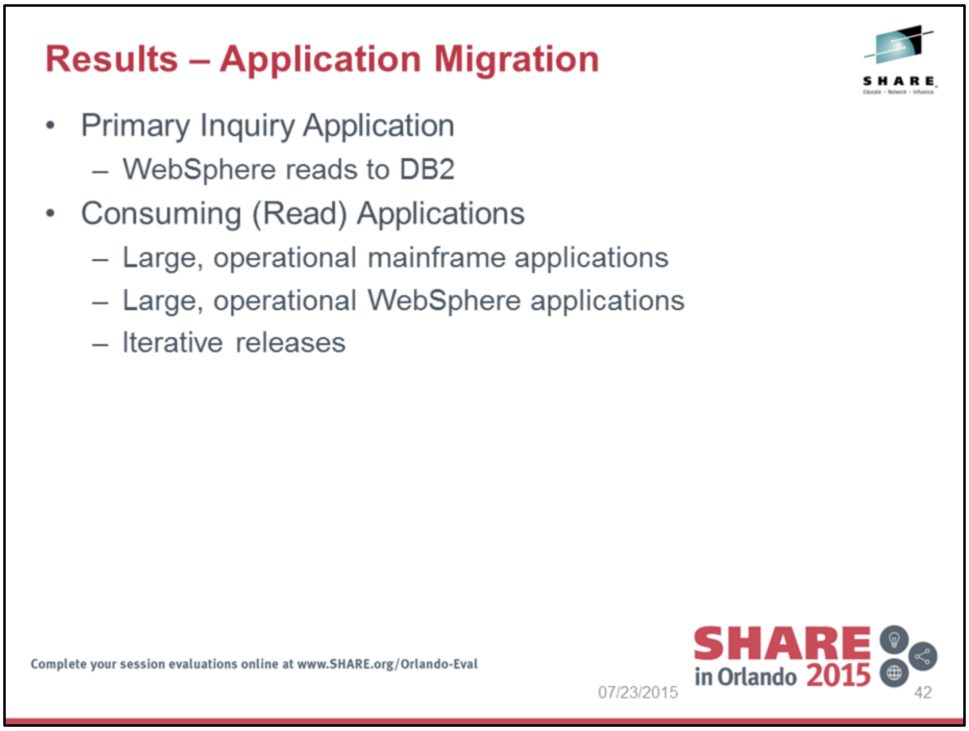
SHARE in Orlando 2015

Results



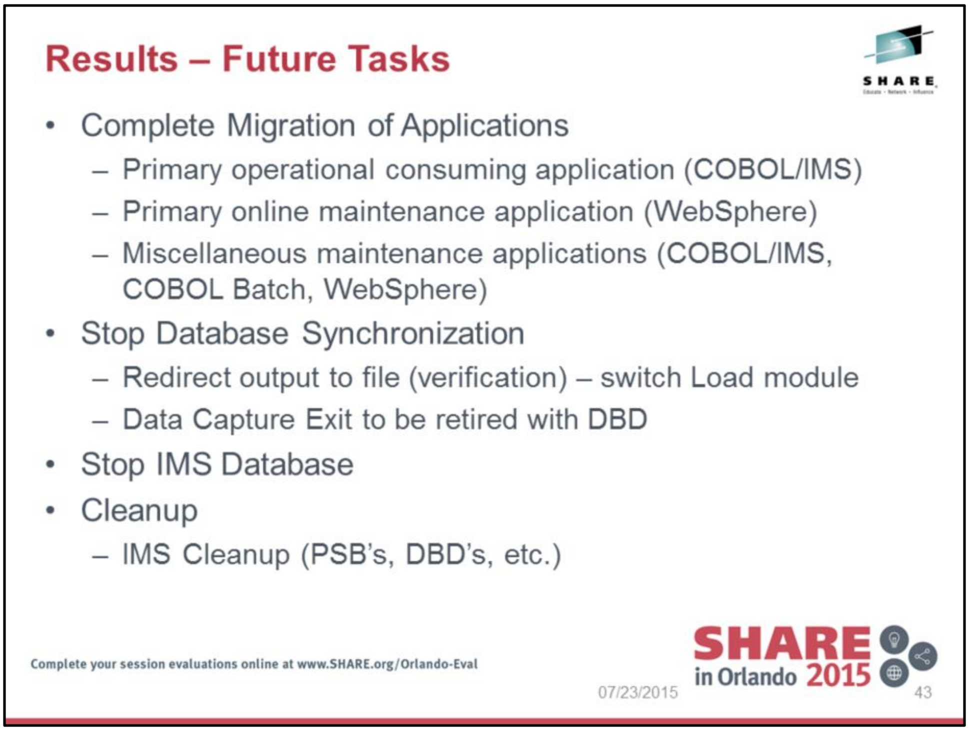
Results – Stats

- Synchronization is Working
- Volume
  - 275,000-300,000 IMS segments processed daily
  - 14 hour batch processing
  - Some increase in MIPS (MQ, DB2)
- Speed
  - Impact to IMS transaction response time is minimal
  - Synchronized data is available in “near real time” (minutes)
- Reliability
  - Data Capture Exit job runs cleanly
  - Batch job synchronization is stable and reliable



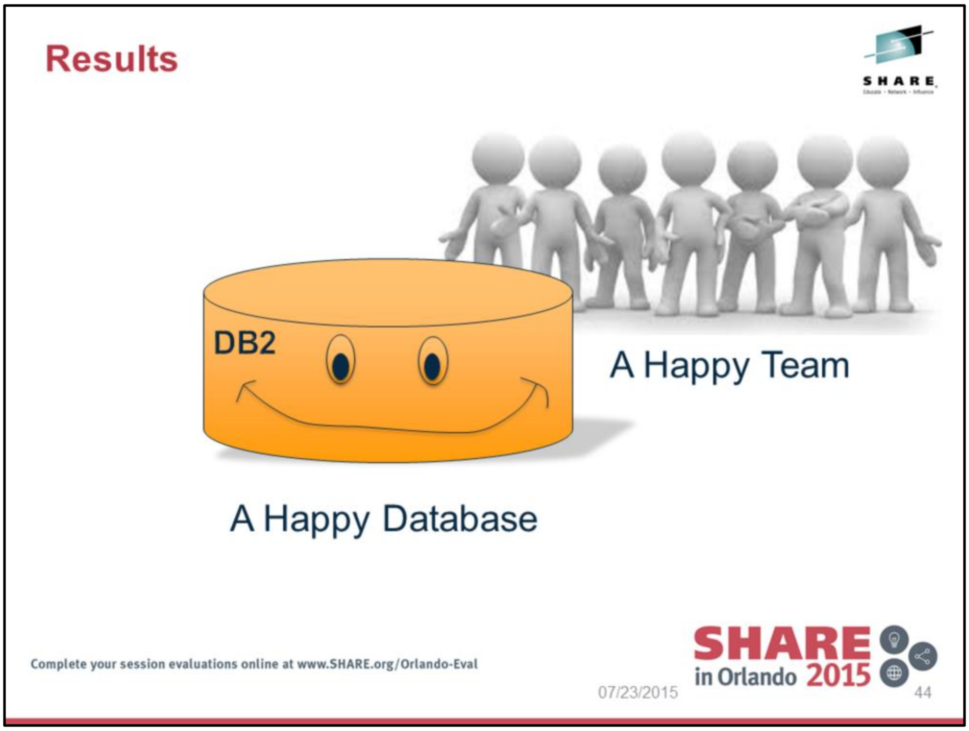
Results – Application Migration

- Primary Inquiry Application
  - WebSphere reads to DB2
- Consuming (Read) Applications
  - Large, operational mainframe applications
  - Large, operational WebSphere applications
  - Iterative releases



Results – Future Tasks

- Complete Migration of Applications
  - Primary operational consuming application (COBOL/IMS)
  - Primary online maintenance application (WebSphere)
  - Miscellaneous maintenance applications (COBOL/IMS, COBOL Batch, WebSphere)
- Stop Database Synchronization
  - Redirect output to file (verification) – switch Load module
  - Data Capture Exit to be retired with DBD
- Stop IMS Database
- Cleanup
  - IMS Cleanup (PSB's, DBD's, etc.)



Results

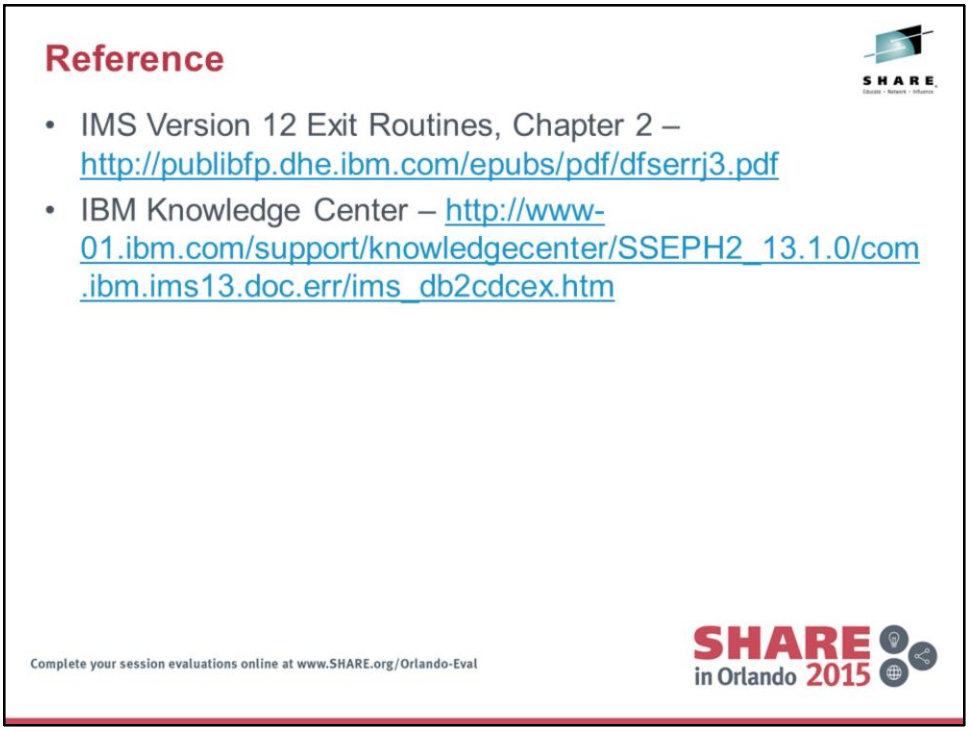
DB2

A Happy Team

A Happy Database

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

SHARE in Orlando 2015



Reference

- IMS Version 12 Exit Routines, Chapter 2 – http://publibfp.dhe.ibm.com/epubs/pdf/dfserrj3.pdf
- IBM Knowledge Center – http://www-01.ibm.com/support/knowledgecenter/SSEPH2_13.1.0/com.ibm.ims13.doc.err/ims_db2cdcex.htm