Virtual IMS user group: Newsletter 80

Welcome to the Virtual IMS user group newsletter. The Virtual IMS user group at itech-ed.com/virtualims is an independently-operated vendor-neutral site run by and for the IMS user community.

Virtual IMS user group presentation

The latest webinar from the Virtual IMS user group was entitled, "From Legacy to Infinity & Beyond". It was presented by Dougie Lawson.

Until 2014 Dougie was a senior software specialist in IMS with IBM Global Services in the United Kingdom, working in software support for the UK and Ireland and with the European software support teams.

He has thirty-nine years of experience in the IT field and thirty-eight of those working with IMS. His areas of expertise include Assembler Programming, IMS, Db2, Linux, COBOL, PL/I, C/C++, and z/OS.

- IMS.PROCLIB(CSLRI002)

```
ARMRST=N,
RMNAME=RM2,
IMSPLEX(NAME=PLEX2)
<SECTION=REPOSITORY>
REPOSITORY=(NAME=IMSREPO,TYPE=IMSRSC,GROUP=R002REPO,
AUDITACCESS=NOAUDIT)
```
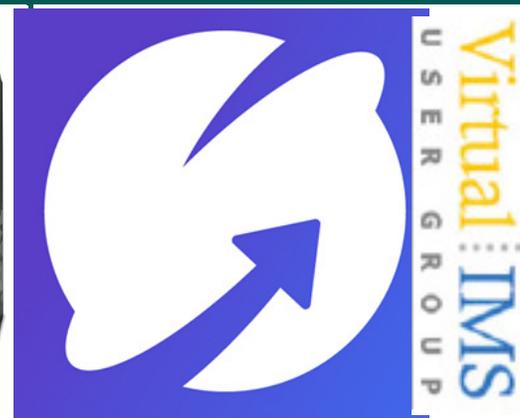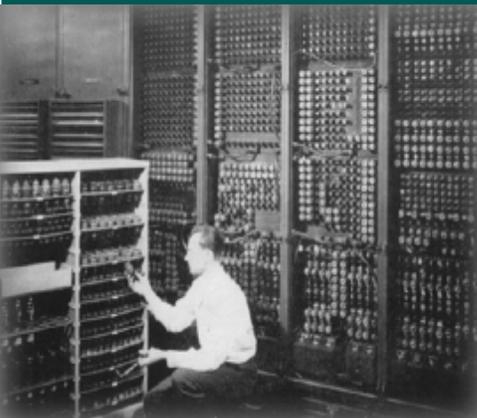
Figure 1: Resource Manager PROCLIB updates

Before joining IBM in 1994, Dougie worked as a senior systems programmer for a large bank in the UK, where he was responsible for their IMS and Db2 systems and related products.

He's now roaming the world for BMC Software as a Database Specialist finding lots of strange, lonely, and forgotten IMS and Db2 systems to play with.

Contents:

```
// JCLLIB ORDER=IMS.SDFSPROC
//ACB3BMP EXEC PGM=DFS3UACB,REGION=0M
//STEPLIB  DD   DSN=IMS.SDFSRESL,DISP=SHR
//PROCLIB  DD   DSN=IMS.PROCLIB,DISP=SHR
//DFSRESLB DD   DSN=IMS.SDFSRESL,DISP=SHR
//SYSPRINT DD   SYSOUT=*
//SYSOUT   DD   SYSOUT=*
//SYSABEND DD   SYSOUT=*
//IMS      DD   DSN=IMS.PSBLIB,DISP=SHR
//         DD   DSN=IMS.DBDLIB,DISP=SHR
//IMSACB   DD   DSN=IMS.ACBLIB,DISP=OLD
//SYSUT3   DD   UNIT=SYSDA,SPACE=(80,(100,100))
//SYSUT4   DD   UNIT=SYSDA,
//              SPACE=(256,(100,100)),
//              DCB=KEYLEN=30
//SYSIN    DD   *
         BUILD PSB=DFSIVP33
/*

//IMSACB01 DD   DSN=*.IMSACB,DISP=OLD
//ACBCATWK DD   SPACE=(CYL,(1,1)),UNIT=SYSDA
//SYSINP   DD   DUMMY
//DFS3PPRM DD *
BMP,DFS3PU00,DFSCP001,,,N00000,,,,,,,,,IMS,,,,,'',,,
/*
//
```

Figure 2: Catalog Populate Utility BMP JCL

Dougie Lawson started off by looking at where most sites are today with their IMS systems and then focused on all the new things that are available with IMS.

He first looked at the IMS Catalog, which was new with IMS V12. It holds IMS meta data in an OSAM HALDB. Database Recovery Control (DBRC) is optional but highly recommended. Backups and recovery work just like every other OSAM HALDB. A Catalog is required for Java, SQLIMS, and managed ACBs. For sites that use source code management systems, like Endevor, they need radical changes because of ACBGEN changes. Using a Catalog requires PROCLIB updates in IMS.

Dougie then explained why we need a Catalog. He said that Java was available in IMS V7.1 (2001) and the system Jar files were supplied with IMS base code (and still are). DL/I databases needed to be mapped to a Java object to read the DBD and create a metadata Jar file. The Jar file had to be distributed to the client system running the Java program. If the IMS DBA made a change to the database or there's an ACBGEN to promote to production with Online Change, that meant the Jar file had to be rebuilt and then redistributed to all clients. Users just had to hope that the Jar file in use matched the ACBs running in IMS. There was no checking. The system was flawed and didn't scale well.

The Catalog solves the Jar file problem. System Jar files can now query IMS for metadata. The IMS DL/I GUR call gets Catalog records in XML. The XML is processed by system Jar files to dynamically create the Java object for the database. If the database changes, the Catalog gets updated, and the clients get the database structure as it is right now. The DL/I Model Utility is gone. There's no need to distribute metadata Jar files and keep them current with the ACB used by IMS.

Dougie showed some examples of updates to PROCLIB members. Figure 1 shows Resource Manager PROCLIB updates.

The Catalog Populate Utility runs like ACBGEN and builds a staging ACBLIB as before. It can run as DLIBATCH or can run as BMP. Dougie Lawson described how this

```
SEGM      NAME=A1111111,                                           C
          EXTERNALNAME=PHONEBOOK,                                  C
          PARENT=0,                                                C
          BYTES=40,                                                C
          RULES=(LLL,LAST)

FIELD     EXTERNALNAME=FIRSTNAME,NAME=FNAME,                       C
          BYTES=10,                                                C
          START=11,                                                C
          TYPE=C,                                                  C
          DATATYPE=CHAR
```

Figure 3: DBD changes

can be done. Figure 2 shows the Catalog Populate Utility BMP JCL.

There are ACBGEN changes. Most sites will have ACBGEN built into their source manager, eg SCLM, Endevor, Zigi (aka Easy Github for z/OS). The ACBGEN may run on a development system, whereas the Catalog runs on the production system. That means the process for importing will have to change. The source manager supplier can help with this.

IMS Open Database Manager (ODBM) gives an open interface to IMS databases. It can be used by Db2 stored procedures, and will be used by IMS Connect for remote database connectors, eg Java DL/I,

JDBC, and DRDA native clients. Again, Dougie gave examples of updates for the ODBM PROCLIB and the ODBM PZP module.

There are IMS Connect changes for ODBM. IMS Connect gives us: transaction processing; TCP/IP MSC links; TCP/IP CICS ISC links; callout processing; and command processing. It's possible to add: ODBM for database processing; Java DL/I support; DRDA support; and JDBC support. Again, the user group were shown example PROCLIB updates.

Figure 3 shows DBD changes. For example, DBD needs external names. They're the names for the 'table' and 'columns'. The IBM-supplied DFSIVD2 (aka

IVPDB2) does have the new parameters.

There are also PSB changes. It needs to have a PCBNAME, which is used for the SQL statement FROM clause. The IBM-supplied sample DFSIVP2 doesn't have a PCBNAME. The Assembler label field is mapped to PCBNAME=xxxx if PCBNAME is not coded.

When it comes to IMS Java programming, JDBC is ideal for programmers who know SQL but don't know IMS. On the other hand, Java DL/I is more familiar territory for COBOL/IMS programmers. The IMS call compare trace (/TRA SET ON PSB sample1 COMP) will capture DL/I database calls from Java programs.

```
010000*
010100    EXEC SQLIMS
010200       DECLARE TELE1 CURSOR FOR DYSQL
010300    END-EXEC.
010400*
010500    MOVE "SELECT LASTNAME, FIRSTNAME FROM PCB01.PHONEBOOK"
010600         TO SQL-STMT-TEXT.
010700    MOVE 198 TO SQL-STMT-LENGTH.
010800*
010900    EXEC SQLIMS
011000       PREPARE DYSQL FROM :SQL-STATEMENT
011100    END-EXEC.
011200 *
011300    EXEC SQLIMS
011400       DESCRIBE OUTPUT DYSQL INTO :SQLIMSDA
011500    END-EXEC.
011600*
```

Figure 4: COBOL with SQLIMS

Dougie said that JDBC is easy and familiar, and it uses SQL just like Db2 JDBC. There are some differences in the SQL language – the IMS books list the allowed SQL language elements. IMS turns the SQL into DL/I calls. The hierarchy looks like SQL tables with foreign keys. There is support for a type-2 (local) or type-4 (remote) connection. With IMS Connect, it's always going to be a type-4 connection.

Dougie showed the user group the code to create an IMS connection, as well as an example of IMS Java programming using JDBC and Java DL/I. He did say that Java DL/I needs an understanding of the IMS hierarchy and an understanding of IMS segment search arguments (SSAs). It may need an understanding of command codes because it can issue some IMS system services call. It can use an AIB or PSB, though usually it's AIB.

Next, Dougie showed an example of COBOL with SQLIMS saying that it looks a lot like Db2 SQL. There's no static SQL, everything is dynamic SQL. It follows the formula: prepare, describe, execute (see Figure 4). All calls run with an AIB, and errors are passed back in SQLCODE and SQLSTATE. There are some new SQL codes just for IMS.

When using batch programs with ODBM, all calls use AIB. The interface is AERTDLI not AIBTDLI. ABENDs in batch AERTDLI programs don't cause ABENDU0113 in IMS. Again, the user group were shown example code.

When using Db2 stored procedures with ODBM, the connection to ODBM is with an CIMS INIT call. A PSB is allocated with an APSB call. You have to use a DPSB when you want to change a PSB. And users call CIMS to terminate a thread. TERM means terminates this DB thread, TALL means terminate all DB threads and destroy the DRA environment. All calls use an AIB. The interface

is AERTDLI with AIB, not AIBTDLI, which is just like ODBM batch. An ABEND in a stored procedure won't cause ABENDU0113 in IMS.

A copy of Dougie Lawson's presentation is available for download from the Virtual IMS user group Web site at itech-ed.com/virtualims/presentations/IMSInfinityAug21.pdf.

You can see and hear the whole user group meeting at https://youtu.be/n-qZ36-_bWI.

Meeting dates

The following meeting dates have been arranged for the Virtual IMS user group:

- On 12 October, Karen N Tischer, dba IMS Education & Consulting, Instructor/Consultant will be giving an "Overview of Fast Path DEDBs".

- The following meeting will be on 7 December, when Haley Fung, IBM Offering Manager for IMS, will be discussing "IMS and Ansible".

IMS articles and blogs

Performance Enhancement for IMS TM and Db2 Customers Using RACF by Sanjay Kaliyur in the IMS part of the IBM Z and LinuxONE Community (2 July 2021). You can find the article at `https://community.ibm.com/community/user/ibmz-and-linuxone/blogs/sanjay-kaliyur1/2021/06/15/performance-enhancement-ims-tm-and-db2`.

.

Like us on Facebook

#VirtualIMS

About the Virtual IMS user group

The Virtual IMS user group was established as a way for individuals using IBM's IMS hierarchical database and transaction processing systems to exchange information, learn new techniques, and advance their skills with the product

The Web site at https://itech-ed.com/virtualims provides a central point for coordinating periodic meetings (which contain technically-oriented topics presented in a webinar format), and provides articles, discussions, links, and other resources of interest to IBM IMS practitioners. Anyone with an interest in IMS is welcome to join the Virtual IMS user group and share in the knowledge exchange.

To share ideas, and for further information, contact trevor@itech-ed.com.

The Virtual IMS user group is free to its members.