



# How to Use IMS Log Records to Investigate Transaction Response Time

Loc Tran – April 12, 2022



# Agenda

- **Why IMS writes log records?**
- **IMS log record component**
- **IMS DFSERA10 program**
- **How to tie log records to a transaction?**
- **Full-Function**
- **Message Switch**
- **Fast Path**
- **Demo**

# IMS Log Record

## Why does IMS write log records?

- **Database Recovery**
  - Undo/redo database changes
  - Forward recovery
- **Data Communication Recovery**
  - Restore terminal status
  - Restore queues
  - Etc.
- **Statistics**
  - Provide statistical data for IMS usage and performance

# IMS Log Record Component

What are the components involved?

- ILOG macro

```
      ILOG  FUNC=WRT ,          CALL IMS  LOGGER
           DECB=(R5) ,
           RCD=(R10) ,
           SCD=(R11) ,
           WAITYPE=IWAIT

+* ,FUNC=WRT
+      MVI  4(1) ,ILOGFWRT      WRITE FUNCTION FOR DECTYPE  @BIH8I
+      MVI  5(1) ,ILOGOLDS      SET LOG DATA SET TYPE      @BIH8I
+      LA   15,0( ,R10)         CLEAR HIGH ORDER BIT       @BIAA4
+      ST   15,12(1)           STORE DATA ADDRESS IN DECB        @BIAA4
+* ,WAITYPE PARAMETER IGNORED
+      LLGF 15,SCDREENT-SCD(R11)  LOAD  LOGGER ENTRY ADDR
+      BASSM 14,15              BRANCH TO LOGICAL  LOGGER
```

# IMS Log Record Component (cont'd)

What are the components involved?

- **IMS Log Manager**
  - DFSFLLG0
- **OLDS/SLDS**
- **DBRC/RECON**

# IMS Log Record Component (cont'd)

What are the components involved?

- ILOGREC macro
  - Log record DSECT

```
*****
* THE FOLLOWING NAMING CONVENTIONS ARE USED (MOSTLY):
*   DFSLOG## - FOR COPY FILES
*             (## IS THE NUMBER OF THE LOG RECORD)
*   QLOG**** - FOR MACROS
*             (**** DESCRIBES THE USE OF THE LOG REC)
*****
CURRENT ANOP
          AIF ('&RTYPE' NE 'DSECT') .NOLOGRC
          DFSLOGRC
NOLOGRC ANOP
          AIF ('&RECID' EQ 'ALL') .L01
          AIF ('&RECID' EQ '00') .L00 @PQ85916
          AIF ('&RECID' EQ '01') .L01
          AIF ('&RECID' EQ '02') .L02
          AIF ('&RECID' EQ '03') .L01
          AIF ('&RECID' EQ '04') .L04 @BOR0373
          AIF ('&RECID' EQ '06') .L06
          AIF ('&RECID' EQ '07') .L07
          AIF ('&RECID' EQ '08') .L08
-----
L01 MEXIT
     ANOP
     AIF ('&RTYPE' EQ 'DS') .L01A @B021083
     QLOGMSGP
     AGO .L01B
L01A ANOP
     QLOGMSGP DSECT= , DSECTD=
L01B AIF ('&RECID' EQ 'ALL') .L02
L02 MEXIT
     ANOP
```

# IMS DFSERA10 program

## What does it do?

- **The ability to select certain IMS log records out of the SLDS by**
  - Record type
  - Value in log record offset
  - Print/Copy
  - By record sequence range

```
CONTROL CNTL SKIP=305800,STOPAFT=306000
```

```
OPTION COPY OFFSET=105,FLDTYP=C,VALUE=MBL62A04,FLDLEN=8,COND=E
```

```
OPTION PRINT OFFSET=6,FLDTYP=X,VALUE=D7F1F0F0F0F4F3C5,FLDLEN=8
```

# IMS DFSERA10 program (cont'd)

An example to extract log records for Full-function and Fast Path transactions from a SLDS

```
OPTION COPY OFFSET=5,FLDTYP=X,VALUE=01,FLDLEN=1
OPTION COPY OFFSET=5,FLDTYP=X,VALUE=03,FLDLEN=1
OPTION COPY OFFSET=5,FLDTYP=X,VALUE=07,FLDLEN=1
OPTION COPY OFFSET=5,FLDTYP=X,VALUE=08,FLDLEN=1
OPTION COPY OFFSET=5,FLDTYP=X,VALUE=31,FLDLEN=1
OPTION COPY OFFSET=5,FLDTYP=X,VALUE=33,FLDLEN=1
OPTION COPY OFFSET=5,FLDTYP=X,VALUE=35,FLDLEN=1
OPTION COPY OFFSET=5,FLDTYP=X,VALUE=36,FLDLEN=1
OPTION COPY OFFSET=5,FLDTYP=X,VALUE=56FA,FLDLEN=2
OPTION COPY OFFSET=5,FLDTYP=X,VALUE=59,FLDLEN=1
OPTION COPY OFFSET=5,FLDTYP=X,VALUE=FA,FLDLEN=1
```



# How to tie IMS log records to a transaction?

A transaction has multiple log records that signify the major events

- **Old method**
  - The use of DRRN – Disk Relative Record Number
- **New method**
  - UOW – Unit-of-work
    - IMSID + STCK
    - IMSID + Recovery Token

# Full-Function Log Records

## One Full-Function transaction

A	Log Code	Type	Sequence No.	Timestamp	Time difference between records
—	01	input msg	00000001ACD20	2022.095 07:15:22.177770	+ .000000
—	35	msg enqueue	00000001ACD21	2022.095 07:15:22.177786	+ .000015
—	08	pgm start	00000001ACD22	2022.095 07:15:22.179083	+ .001297
—	5607	recovery	00000001ACD23	2022.095 07:15:22.179084	+ .000000
—	31	msg get	00000001ACD24	2022.095 07:15:22.179108	+ .000023
—	5050	DB update	00000001ACD26	2022.095 07:15:22.201429	+ .022320
—	03	output msg	00000001ACD27	2022.095 07:15:22.202600	+ .001171
—	35	msg enqueue	00000001ACD28	2022.095 07:15:22.202606	+ .000005
—	37	msg xfer	00000001ACD29	2022.095 07:15:22.202614	+ .000007
—	37	msg xfer	00000001ACD2A	2022.095 07:15:22.202621	+ .000007
—	33	msg free	00000001ACD2B	2022.095 07:15:22.202632	+ .000010
—	31	msg get	00000001ACD2C	2022.095 07:15:22.202657	+ .000025
—	5612	recovery	00000001ACD2D	2022.095 07:15:22.202953	+ .000295
—	5607	recovery	00000001ACD2E	2022.095 07:15:22.202955	+ .000002
—	5612	recovery	00000001ACD2F	2022.095 07:15:22.203144	+ .000188
—	FA	Mainview	00000001ACD30	2022.095 07:15:22.203184	+ .000040
—	F9	Mainview	00000001ACD31	2022.095 07:15:22.203192	+ .000007
—	07	pgm end	00000001ACD32	2022.095 07:15:22.203199	+ .000007
—	36	msg dequeue	00000001ACD33	2022.095 07:15:22.204421	+ .001221
—	33	msg free	00000001ACD34	2022.095 07:15:22.204433	+ .000011
***** Bottom of Data *****					

# Full-Function Log Records (cont'd)

IMS can produce a X'56FA' statistical record for a transaction

- **Mapped by DSECT DFSETPCP**
- **DFSDFxxx of IMS PROCLIB**
  - TRANSTAT=y/n in the Diagnostics Statistics section
  - For all transactions
- **UPDATE command**
  - UPDATE TRANDESC with TRANSTAT()
  - For a particular transaction



# Message Switch Log Records

One transaction switched to 2 new transactions

Log A	Code	Type	Sequence No.	Timestamp	Time difference between records
—	01	input msg	00000001ACD48	2022.095 07:16:31.688893	+ .000000
—	35	msg enqueue	00000001ACD49	2022.095 07:16:31.688919	+ .000025
—	08	pgm start	00000001ACD4A	2022.095 07:16:31.689880	+ .000961
—	5607	recovery	00000001ACD4B	2022.095 07:16:31.689881	+ .000000
—	31	msg get	00000001ACD4C	2022.095 07:16:31.689900	+ .000019
—	03	output msg	00000001ACD4D	2022.095 07:16:31.700141	+ .010241
—	35	msg enqueue	00000001ACD4E	2022.095 07:16:31.700149	+ .000007
—	37	msg xfer	00000001ACD4F	2022.095 07:16:31.700155	+ .000006
—	37	msg xfer	00000001ACD50	2022.095 07:16:31.700161	+ .000005
—	33	msg free	00000001ACD51	2022.095 07:16:31.700171	+ .000010
—	5612	recovery	00000001ACD52	2022.095 07:16:31.700371	+ .000200
—	5607	recovery	00000001ACD53	2022.095 07:16:31.700371	+ .000000
—	31	msg get	00000001ACD54	2022.095 07:16:31.700385	+ .000013
—	FA	Mainview	00000001ACD55	2022.095 07:16:31.700416	+ .000030
—	03	output msg	00000001ACD56	2022.095 07:16:31.700550	+ .000133
—	35	msg enqueue	00000001ACD57	2022.095 07:16:31.700555	+ .000004
—	37	msg xfer	00000001ACD58	2022.095 07:16:31.700558	+ .000003
—	37	msg xfer	00000001ACD59	2022.095 07:16:31.700562	+ .000003
—	33	msg free	00000001ACD5A	2022.095 07:16:31.700566	+ .000004
—	5612	recovery	00000001ACD5B	2022.095 07:16:31.700775	+ .000208
—	5607	recovery	00000001ACD5C	2022.095 07:16:31.700775	+ .000000
—	31	msg get	00000001ACD5D	2022.095 07:16:31.700779	+ .000004
—	FA	Mainview	00000001ACD5E	2022.095 07:16:31.700790	+ .000010

# Message Switch Log Records (Cont'd)

One transaction switched to 2 new transactions

```

_ 03  output msg      000000001ACD5F 2022.095 07:16:31.700817 + .000027
_ 35  msg enqueue    000000001ACD60 2022.095 07:16:31.700821 + .000003
_ 37  msg xfer       000000001ACD61 2022.095 07:16:31.700822 + .000001
_ 37  msg xfer       000000001ACD62 2022.095 07:16:31.700825 + .000003
_ 33  msg free       000000001ACD63 2022.095 07:16:31.700831 + .000005
_ 5612 recovery    000000001ACD64 2022.095 07:16:31.700834 + .000003
_ 5607 recovery    000000001ACD65 2022.095 07:16:31.700834 + .000000
_ 5612 recovery    000000001ACD66 2022.095 07:16:31.700888 + .000053
_ FA  Mainview      000000001ACD67 2022.095 07:16:31.700892 + .000003
_ F9  Mainview      000000001ACD68 2022.095 07:16:31.700896 + .000003
_ 07  pgm end        000000001ACD69 2022.095 07:16:31.700901 + .000005
_ 31  msg get        000000001ACD6A 2022.095 07:16:31.700960 + .000058
_ 36  msg dequeue    000000001ACD6B 2022.095 07:16:31.703307 + .002347
_ 33  msg free       000000001ACD6C 2022.095 07:16:31.703315 + .000007
```



# Fast Path Log Records

Fast Path log records – X'59xx'

Log	Type	Sequence No.	Timestamp	Time difference between records
5901 FP inpt msg		00000001AC189	2022.090 06:35:52.659750	+ .000000
5903 FP out msg		00000001AC18A	2022.090 06:35:52.659751	+ .000000
5950 FP DB upd		00000001AC18B	2022.090 06:35:52.659753	+ .000002
5950 FP DB upd		00000001AC18C	2022.090 06:35:52.659754	+ .000000
5937 FP syncpnt		00000001AC18D	2022.090 06:35:52.659809	+ .000055
5936 FP msg deq		00000001AC18F	2022.090 06:35:52.660321	+ .000511
5612 recovery		00000001AC190	2022.090 06:35:52.660533	+ .000211
FA Mainview		00000001AC1AF	2022.090 06:36:00.369868	+ 7.709335

\*\*\*\*\* Bottom of Data \*\*\*\*\*

# Log Record Macros

## Log record macros for reference

01 – QLOGMSGP

03 – QLOGMSGP

07 – DFSLOG07

08 – DFSLOG08

31 – QLOGGETU

33 – QLOGFREE

35 – QLOGENQU

56 – DFSETPCP

5901 – DBFLGRIM

5903 – DBFLGROM

5936 – DBFLGRDQ

5937 - DBFLGSYN

**D E M O**



