


MQPERF1

**MQ Statistics and MP1B –
More detail on some
traditional areas**



An
IBM Washington
Systems Center **Wildfire**
Workshop



MP1B

- This SupportPac processes the MQ SMF data into:
 - Report format
 - CSV files
- Format of the output has changed over the years
- Originally owned and managed by Colin Paice (now retired)
- Now kept by Gwydion Tudur



MP1B – Contents

- Three programs
 - MQSMF
 - Formats the MQ SMF records into reports and CSV files
 - Produces messages about issues spotted in the SMF data
 - Very helpful for first time users
 - Source is not delivered
 - OEMPUT
 - Sample MQ program to exercise most of the APIs and track costs
 - Very helpful after upgrades, new implementations, big changes to infrastructure
 - Source not delivered
 - MQCMD
 - Sample program to issue MQ commands and capture response

MQSMF – the SMF Post processing Program

- This is the place we all start!
- The output from MQSMF looks like there (there are more files)

```
SDSF JOB DATA SET DISPLAY - JOB ELKINS3Q (JOB06020)      LINE 1-18 (53)
COMMAND INPUT ==>                                         SCROLL ==> CSR
NP   DDNAME   StepName ProcStep DSID Owner   C Dest           Rec-Cnt Page
    JESMSG LG  JES2      2 ELKINSC 0 LOCAL           20
    JESJCL    JES2      3 ELKINSC 0 LOCAL           70
    JESMSG    JES2      4 ELKINSC 0 LOCAL          165
    MESSAGE   S1        102 ELKINSC 0 LOCAL          430
    BUFF      S1        103 ELKINSC 0 LOCAL         49,415
    BUFFCSV   S1        104 ELKINSC 0 LOCAL          9,722
    DATA     S1        105 ELKINSC 0 LOCAL          1,526
    CF        S1        106 ELKINSC 0 LOCAL           49
    CFCSV     S1        107 ELKINSC 0 LOCAL           13
    DB2       S1        108 ELKINSC 0 LOCAL          2,507
    EOJ       S1        109 ELKINSC 0 LOCAL           407
    LOCK      S1        110 ELKINSC 0 LOCAL           246
    LOG       S1        111 ELKINSC 0 LOCAL          1,549
    LOGCSV    S1        112 ELKINSC 0 LOCAL           32
    MSGM      S1        113 ELKINSC 0 LOCAL          2,836
    MSGMCSV   S1        114 ELKINSC 0 LOCAL           811
    SMDS      S1        116 ELKINSC 0 LOCAL         15,414
    TASKSUM   S1        117 ELKINSC 0 LOCAL           24
```

MP1B – the Message Output

- When you begin looking at the MQ SMF data, looking at the MESSAGE output file can be helpful
 - The messages produced may vary based on the settings used
 - As an example these messages indicate bufferpool stress that needs investigation

```
MQQPST00W MPX1,QML3,2018/09/05,16:03:04,VRM:900, BP 10 Many(5979) buffers writte
MQQPST02S MPX1,QML3,2018/09/05,16:03:04,VRM:900, BP 10 Filled many(373) times.
l may be too small
MQQPST04E MPX1,QML3,2018/09/05,16:03:04,VRM:900, BP 10 Many (6022) pages read fr
ffer pool may be too small
MQQJST11W MPX1,QML5,2018/09/05,16:03:05,VRM:905, logging rate is low 0 < 50 MB/
```

MP1B – SYSPRINT

- This output file has the MQSMF runtime characteristics

```
Compiled Aug 12 2015 16:50:51.  
buffer: SMF_Interval_time 1  
buffer: Debug 1  
buffer: Detail 20  
kw UseTaskLocalTime      0 -Convert task start times to local time  
kw SMDSWriteTime         1000 -SMDS average Write time theshold in microseconds  
kw SMDSReadTime          1000 -SMDS average Read time theshold in microseconds
```

- A copy of some of the serious messages

```
6021 MQTASK36W Did BP 11 fill up for puts to ELKINSC.TEST.PS11  
6023 MQTASK16E long latch wait      1854, Name BMXL3 |CFMTODO |SRH1_L1  
6023 MQTASK15W longest latch wait address 0000000013bbbb10 000000001854  
6075 MQTASK16S long latch wait      66039, Name BMXL3 |CFMTODO |SRH1_L1  
6075 MQTASK15S longest latch wait address 0000000013bbbb10 0000000066039  
6075 MQTASK36W Did BP 20 fill up for puts to ELKINSC.TEST.PS20  
6129 MQTASK16E long latch wait      1443, Name BMXL2 |RMCRMST |RLMARQC  
6129 MQTASK16S long latch wait      54356, Name BMXL3 |CFMTODO |SRH1_L1  
6129 MQTASK15S longest latch wait address 0000000013bbbb10 0000000054356
```


MP1B – SYSPRINT Continued

- Finally the number, type and subtype of the MQ SMF records processed

```
Summary of MQ SMF records and subtypes found
=====
SMF type 115 subtype 1, record count 405 System statistics(1)
SMF type 115 subtype 2, record count 405 System statistics(2)
SMF type 115 subtype 5, record count 404 Storage statistics
SMF type 115 subtype 6, record count 404 Storage detail statistics
SMF type 115 subtype 7, record count 405 Storage summary statistics
SMF type 115 subtype 201, record count 405
SMF type 115 subtype 215, record count 405 Buffer manager extension
SMF type 115 subtype 231, record count 401 Chinit statistics
SMF type 116 subtype 0, record count 66 Accounting class(1)
SMF type 116 subtype 1, record count 3356 Accounting class(3)
SMF type 116 subtype 10, record count 78 Channel accounting data
***** BOTTOM OF DATA *****
```

MP1B – SMF Data output

- Output files ending in CSV are the comma separated values form of the data

```
MVS,QM,Date,Time,BP,size,lowest_free,highest_used,used_now,%hfull,SOS,#_sync_wri
ites,#_write_I/Os,Location,PageClas
MPX1,QML3,2018/09/05,14:03:28, 0,50000,49983, 17, 17, 0, 0, 0,
MPX1,QML3,2018/09/05,14:03:28, 1,20000,19999, 1, 1, 0, 0, 0,
MPX1,QML3,2018/09/05,14:03:28, 2,50000,49996, 4, 4, 0, 0, 0,
MPX1,QML3,2018/09/05,14:03:28, 3,20000, 4793,15207,15207, 76, 0, 0,
```

- Output files without the CSV are the report format

```
Buffer statistics
MPX1,QML3,2018/09/05,14:03:28,VRM:900,
  From 2018/09/05,14:02:27.567654 to 2018/09/05,14:03:28.385141, duration    61
= BPool    0, Size    50000,%full now  0, Highest %full  0, Disk reads    0
> 00 Buffs    50000 Low    49983 Now    49983 Getp     10 Getn     0
  00 Rio       0 STW       8 TPW       0 WIO       0 IMW     0
  00 DWT       0 DMC       0 STL       0 STLA      0 SOS     0
  00 Below the bar PAGECLAS 4KB
```




MP1B – A couple of warnings

- Each time MQ adds or alters the SMF data, MP1B must be updated to reflect those changes
 - Historically that has often meant that there can be a year (or more) between a new version of MQ and a new version of MP1B
 - HINT: It does no one any good for me to complain about this, customers need to!
 - Updated versions of MP1B continue to work with older (supported) MQ versions
- Each new release of MP1B may change the layout and/or add output files for new SMF data
 - Occasionally this has happened between releases as well
 - You may see messages in SYSPRINT indicating missing DD statements when using a new version of MP1B

Looking at the Message Manager Output

- The Message Manager data does not usually indicate problems
 - It is helpful to track trends of use and capacity planning
 - Are the total MQPUT & MQPUT1 requests rising, and if so what is the rate?
 - Are my peak processing periods changing?
 - Are new verbs being used?
 - It is helpful when you inherit a queue manager and are unsure how it is being used.

```
MPX1,QML3,2018/11/13,13:13:09,VRM:900,  
From 2018/11/13,13:11:42.889304 to 2018/11/13,13:13:09.322980, duration      86 seconds.  
MQOPENS      35,  MQCLOSEs      34,  MQGETs    1640706,  MQPUTs      102  
MQPUT1s    1640625,  MQINQs      53,  MQSETs        0,  C ALL H        0  
MQSUBs        0,  MQSUBRQs      0,  MQCBs        34  
MQCTLs      242,  MQSTATs      0,  Publish        0  
MPX1,QML3,2018/11/13,13:13:09,VRM:900, Get rate 19077/sec Put+put1 rate 19078/sec
```



Message Manager CSV output

- The CSV output file contains the same data, but in a CSV format
 - Can be downloaded and used to create charts, etc.
 - Typically we look at the
 - Total PUTs (MQPUT + MQPUT1)
 - Check the ratio of MQPUT1 to MQPUT
 - MQPUT1 Abuse can be a CPU hog
 - Check the ratio of MQOPEN to MQPUTs
 - Some applications SHOULD be using MQPUT1 to save CPU
 - Total API Requests
 - Total GETs/PUTs
 - Excessive MQGETs can indicate polling style applications
 - MQGET with an appropriate wait interval can...you guessed it save CPU!!!
 - 'New' verbs
 - Some MQ for z admins have been surprised to discover they are doing Pub/Sub!
 - No indication in the statistics of Selector (Message Properties) use

Buffer Manager Output

- The Buffer Manager report has groups like this for each bufferpool defined to the queue manager:

```
= BPool 10, Size 1000,%full now 81, Highest %full 81, Disk reads 0
> 10 Buffs 1000 Low 189 Now 189 Getp 1 Getn 0
  10 Rio 0 STW 0 TPW 0 WIO 0 IMW 0
  10 DWT 0 DMC 0 STL 0 STLA 0 SOS 0
  10 Below the bar PAGECLAS 4KB
```



Buffer Manager CSV

- The buffer manager CSV file contains the same data in a CSV format
 - Useful for spreadsheet processing



So how do I know when I have a problem with Bufferpools?

- Even if the bufferpools are above the bar, there can still be issues
- Tuning is a matter of I/O avoidance when possible
- **Short On Storage**
 - The SOS (QPSTSOS) value is greater than zero
 - Zero free pages are available
 - Access to the bufferpool is suspended, until pages are freed
- **Immediate Write**
 - The immediate write count (QPSTIMW) is greater than zero
 - This is the number of actual write operations
- **Synchronous Write Threshold**
 - The synchronous write threshold (QPSTDMC) is greater than zero
 - The bufferpool is at 5% free pages or fewer
 - This is the number of time the condition is hit during the interval



So how do I know when I have a problem with Bufferpools?

- **Asynchronous write threshold reached**
 - The asynchronous write threshold (QPSTDWT) count is not zero
 - The threshold is at 15% free pages
 - For an on-line bufferpool, this can be an issue as I/O avoidance is the name of the game
 - For a batch bufferpool, this is expected




So how do I anticipate a problem with Bufferpools?

- Make sure there is adequate head room in the bufferpool
 - If an online bufferpool is approaching the asynchronous write threshold (15% free pages) regularly, then it may need to be expanded or queues moved
 - One 'bad day' for a single queue could mean trouble for any application using that pool
 - If ANY bufferpool is approaching the synchronous write threshold (5% free pages) regularly, it needs to be expanded or queues moved
 - A lightly loaded bufferpool is a happy bufferpool!
- Planning tips:
 - Look out for a change in message sizes
 - Sometimes not well communicated from application development
 - Does not typically cause an error
 - Look out for changes in usage patterns
 - Adding new locations
 - Connecting client application directly to z/OS
 - New applications

Log Manager Report

- The log manager report is incredibly detailed
- Just a part of the report from one interval looks like this

```
MPX1,QML1,2018/11/12,08:39:18,VRM:900,
  From 2018/11/12,08:38:08.821259 to 2018/11/12,08:39:18.054993, duration      69 seconds.
  Wait for buffers(should be 0):          0 out of          0, 0%
  Total Number of pages written:           4
  Number of pages written/sec:             0
  Amount of data written/sec:             0 MB/Sec
  Total Number of write requests:          4
  Number of write requests/sec:           0
  Pages written per I/O:                   1
  Total number of read requests:           0
  Write_Wait      0, Write_Nowait          26, Write_Force      0, WTB      0
  Read_Stor      0, Read_Active            0, Read_Archive     0, TVC      0
  BSDS_Reqs     0, CIs_Created             0, BFWR              4, ALR      0
  ALW           0, CIs_Offload             0, LLCheckpoints      0
  Read delayed   0, Tape Lookahead         0, Lookahead Mount    0
  Write_Susp     4, Write_Reqs             4, CI_Writes          4
  Write_Serl     0, Write_Thrsh            0, Buff_Pagein        0
```



Log Manager Report – the fields and issues

- Wait for buffers – QJSTWRF – documented as ‘should be zero’
 - However, for many busy queue managers that count is almost never zero for a peak period
 - I (Lyn Elkins) tend not to take this so seriously, unless the number exceeds 100
- Reads
 - Reading the buffers, active and archive logs indicates that transactions are backing out work
 - And while logs are being read, there is no writing going on!
 - If this is rare, it can safely be ignored
 - If it is typical, investigation required
 - May need to gather the task accounting data to determine what task is doing this
 - Recently saw an issue with a new client attached application that was experiencing regular channel time outs
- Checkpoints
 - LOGLOAD – the queue manager attribute that controls the number of log records written before taking an internal checkpoint
 - Prior to V9, the number of checkpoints was only the count of those instances
 - Post V9, the number of checkpoints is the LOGLOAD generated checkpoints AND the log switch generated checkpoints
- Buff_Pagein – QJSTBPAG – number of times a log buffer had to be paged in
 - This is an indication that there is CPU constraint in the LPAR


QMGR Health – Log Manager – example from MQSMFCSV

- This view of the log manager data is emphasizing the number of READs, an indication of applications backout out an inflight transaction
 - In this sample, there were both buffer reads and active log reads
 - Need to look into applications to see why this is being done so often
 - Also examine high number of checkpoints

INTERVAL_DURATION	UNAVAILABLE_BUFFER_COUNT	LOG_READ_OUT_PUT_BUFFER	LOG_READ_ACTIVE_LOG	LOG_READ_ARCHIVE_LOG	TOTAL_LOG_READS	TAPE_CONTENTION_DELAYS	CHECKPOINTS	LOG_CI	MB_PER_SECOND
1795	0	623	4461	0	5084	0	10	2821634	6.14
1789	0	417	3337	0	3754	0	9	2825604	6.17
1796	0	540	2638	0	3178	0	12	3453542	7.51
1792	0	511	2307	0	2818	0	10	2972254	6.48
1789	0	449	2082	0	2531	0	10	2818718	6.15
1773	0	392	1952	0	2344	0	12	3445866	7.59
1798	0	424	1835	0	2259	0	10	3061346	6.65
1787	0	518	1725	0	2243	0	8	2460906	5.38
1797	0	381	1824	0	2205	0	14	4037442	8.78
1797	0	581	1597	0	2178	0	9	2778470	6.04
1791	0	306	1841	0	2147	0	11	3259292	7.11

Log Manager Report - Continued

```
____, __ write requests,      CIs, Average I/O , After I/O  , pages/I/O
      time in uSec, time in uSec,
Log 1, 1 page      4,      4,      269,      1,      1 MPX1,QML1,2018/11/12,08:39:18,VRM:900,
Standard deviation of first log, 1 page per I/O, response time +-      154
Log 1, 1 page Longest I/O      275 at 2018/11/12,12:38:08.825946 UTC
Log 1, 1 page Longest Request      277 at 2018/11/12,12:38:08.825946 UTC
Log write rate      0MB/s per copy      MPX1,QML1,2018/11/12,08:39:18,VRM:900,
Logger I/O busy : 0.00%
Logger task busy: 0.00%
```

Log Manager Report – the fields and issues

- Logger Task Busy
 - Information about how many microseconds the logger task was idle was added in V8, QJSTSLPTU
 - The task busy percentage is calculated based on the interval duration and QJSTSLPTU
 - If consistently above 95%, then workload should be examined
- Longest I/O time
 - This is the longest time it took to complete an I/O to the log
 - If this number is out of line with expectation and history, it should be investigated.

QMGR Health – More on Log I/O – example from MQSMFCSV

- In this example there are a number of very long log I/O response times
- Reported in microseconds, this is going as high as 1.4+ seconds – impacting persistent message speed.
- The average I/O time over the week was around 65,000 microseconds – so these are well out of line
- Not shown here was the log number, the long I/Os were not always on the same log - but many were
- This needs to be investigated by the folks who manage the I/O subsystem.

MB_PerS econd	IO_Total_ Time_1_1 _us	IO_Total_ Suspend_ Time_1_1 _us	IO_Max_ Duration_ 1_1_us
0.18	10597717	11065671	1434308
0.15	12716669	13221591	1390594
0.1	8684189	8952939	1340836
0.2	8898824	9208455	1003748
0.12	9603396	10015326	975297
0.14	13584386	14137813	958608
0.2	12086895	12575309	943756
0.13	8770048	9094084	933591
0.18	16134054	16813463	919142
0.17	11199830	11660753	842476
0.2	12952539	13415911	834920
0.2	14622726	15273655	781723
0.15	10658564	11090891	679185
0.11	9726687	10149635	677612
0.21	19876884	20659627	513114
0.21	13723385	14223804	447616
0.22	14733487	15284274	445473
0.19	14649385	15139950	433519



Summary

- MP1B is the first tool in looking at the SMF data
- For many queue managers regular examination of the 'big three' reports can be a leading indicator of tuning work that may be necessary, or tell you where the problems are
- But these are not the only reports that may be needed!