MQ on z/OS Performance deep dive

Colin Paice IBM Hursley
PAICE@UK.IBM.COM

Why are you here?

- You think you have a MQ performance problem
  - What information can you get from the MQ SMF data
- Audience
  - MQ sysprogs, Administrators

Agenda

- Collecting MQ SMF data
- Overall MQ activity
- Logging
- Buffer pools
- CF activity
- Activity by application task and by queue
- Chinit and channels

What can I use to process the SMF data?

- Tivoli Decision Support (TDS) from IBM
  - Stores data in DB2 tables
- MXG & SAS
  - Very common
- MP1B
  - SupportPac (Free)
  - Has 'expert' built in and can make observations on your data

Notes

- MP1B
  - Link http://www-01.ibm.com/support/docview.wss?uid=swg24005907
  - Instructions and JCL for extracting and processing SMF data
- MP16 Capacity planning and tuning document
  - http://www-01.ibm.com/support/docview.wss?uid=swg24007421
- ColinPaice blog on DeveloperWorks
  - Useful topics
  - Use search engine to find it

MP1B

- C program
- Output dataset
  - //LOG human readable
  - //LOGCSV Comma Separated Values, suitable for spread sheet
  - //TASK tasks and queues in detail
  - //QALL Queue information collected by queue
- Input
  - //SMFIN
  - //SYSIN – selection parameters including
    - Queue manager
    - Queue
    - Detail level

Notes on using MP1B

- Step 1: Run against all SMF data
  - Make //TASK DD DUMMY
  - Look at the 'problems' identified
- Step 2: Select records of interest include...
  - Jobname MYJOB
  - QM MQPA
  - Queue MYQUEUENAME
  - Channel MQPA_MQPC
  - CommitET 5000
  - FirstRecord 1500
  - LastRecord 1501
- Use a spread sheet on *CSV files

How to mine the data from reports

```
  E41K,QMI2,2015/01/06,00:00:00,VRM:710,
   From 2015/01/05,23:45:00.008321 to 2015/01/06,00:00:00.009132, duration   90
= BPool   0, Size 20000,%full now  5, Highest %full  5, Disk reads     0
 > 00 Buffs    20000  Low    18930  Now    18930  Getp   744466  Getn    36986
   00 Rio          0  STW   655777  TPW        0  WIO         0  IMW         0
   00 DWT          0  DMC        0  STL        0  STLA        0  SOS         0
= BPool   1, Size 20000,%full now 45, Highest %full 45, Disk reads     0
 > 01 Buffs    20000  Low    10848  Now    10848  Getp     3297  Getn      747
   01 Rio          0  STW     2146  TPW        0  WIO         0  IMW         0
   01 DWT          0  DMC        0  STL        0  STLA        0  SOS         0
```

- X ALL; f 'RIO ' ALL;delete all x

```
 00 Rio _________0  STW   655777  TPW        0  WIO         0  IMW         0
01 Rio          0  STW     2146  TPW        0  WIO         0  IMW         0
02 Rio          0  STW      117  TPW        0  WIO         0  IMW         0
03 Rio          0  STW     2345  TPW        0  WIO         0  IMW         0
00 Rio          0  STW   394032  TPW        0  WIO         0  IMW         0
01 Rio          0  STW        0  TPW        0  WIO         0  IMW         0
```

- COLS
- Sort 12 22 D

How to mine the data

```
01 Rio      58570  STW   126317  TPW    75743  WIO     21053  IMW      2823
01 Rio      36571  STW    76973  TPW    28293  WIO      7579  IMW       674
01 Rio      26017  STW    48450  TPW    13876  WIO      3799  IMW       439
01 Rio      20386  STW   130132  TPW    29001  WIO      7596  IMW       460
01 Rio      16750  STW    75283  TPW    35325  WIO      9843  IMW      1349
01 Rio      10986  STW    81608  TPW    32840  WIO      8837  IMW       836
01 Rio        361  STW     6887  TPW        0  WIO         0  IMW         0
01 Rio        141  STW    28765  TPW        0  WIO         0  IMW         0
01 Rio         23  STW    30012  TPW     9589  WIO      2479  IMW       109
01 Rio          1  STW     7811  TPW        0  WIO         0  IMW         0
00 Rio          0  STW   655777  TPW        0  WIO         0  IMW         0
01 Rio          0  STW     2146  TPW        0  WIO         0  IMW         0
02 Rio          0  STW      117  TPW        0  WIO         0  IMW         0
```

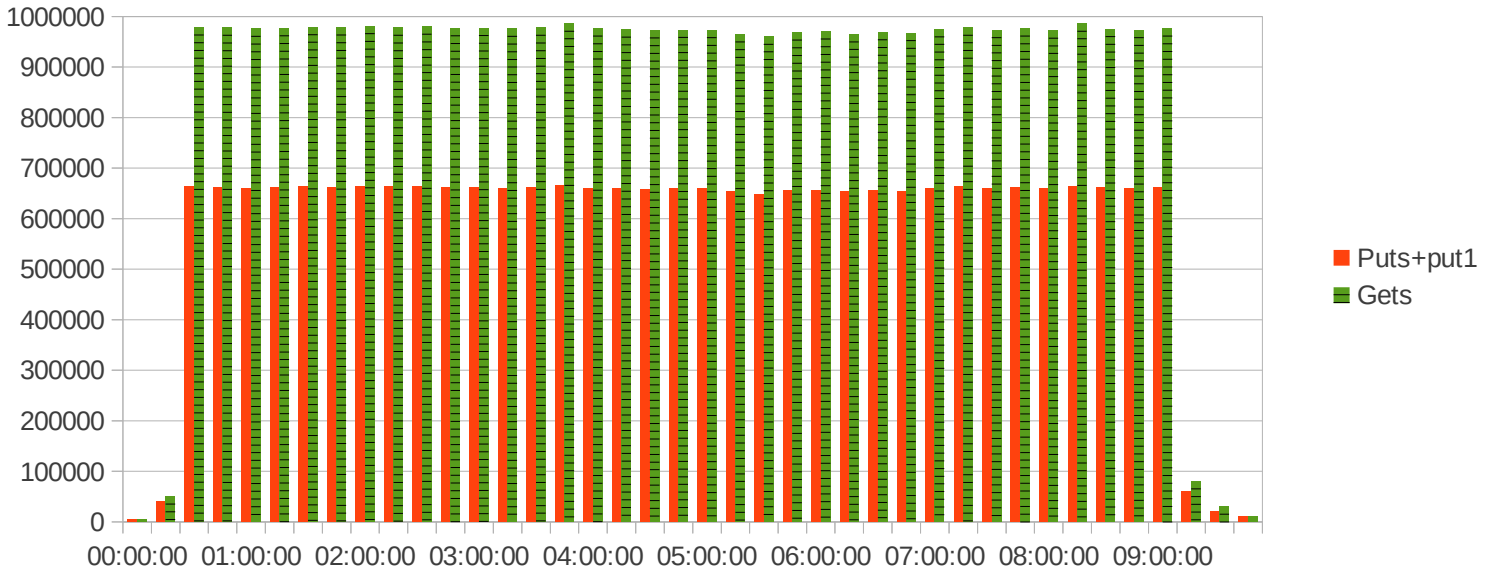
- Cancel from edit session
- Re edit the file. Find '`Rio  58570`'

Agenda

- Collecting MQ SMF data
- Overall MQ activity
- Logging
- Buffer pools
- CF activity
- Activity by application task and by queue
- Chinit and channels

What's going on in my queue manager?

- Long term monitoring – plot peak hour over year
- Number of MQPUT, MQPUT1 and MQGET requests



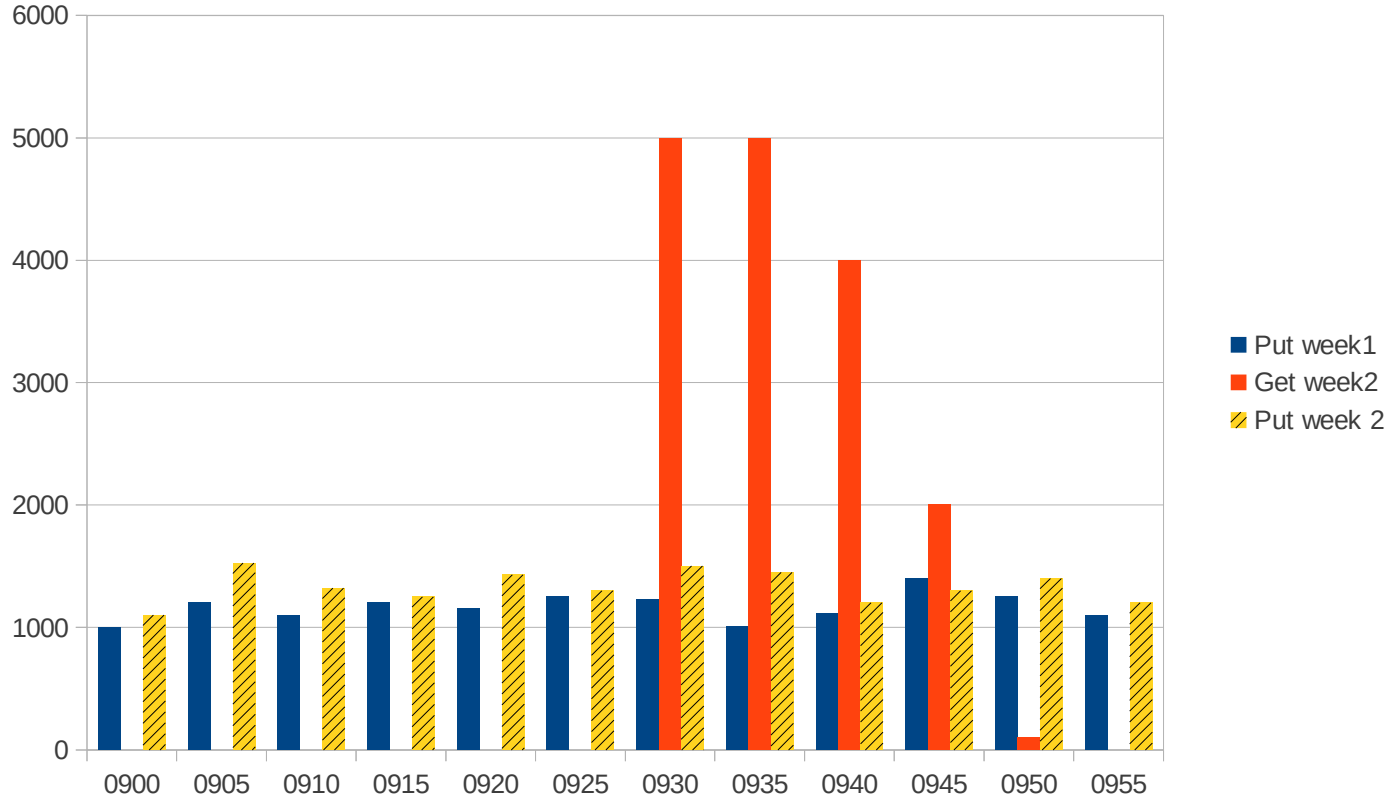
- Number of gets > puts+put1s
  - Servers waiting for messages
  - Application getwait for reply

Notes

- You should take the peak hour in the week, such as Monday morning 0800 to 0900 and plot the data on a week by week.
- This data was taken from a test system.
  - You can see the start up a long test and the shut down. In normal use this graph will have more variation
- A useful metric is the number of puts and gets
  - The number of puts is the measure of work going through
  - The number of gets can vary. For example too many servers will lead to more gets not getting messages.
  - You would expect a 2 * get for a get from the reply queue.
  - Under the covers the first get – has no message found and waits. When the reply arrives, the MQGET is retried and the message retrieved

What is your MQ usage profile

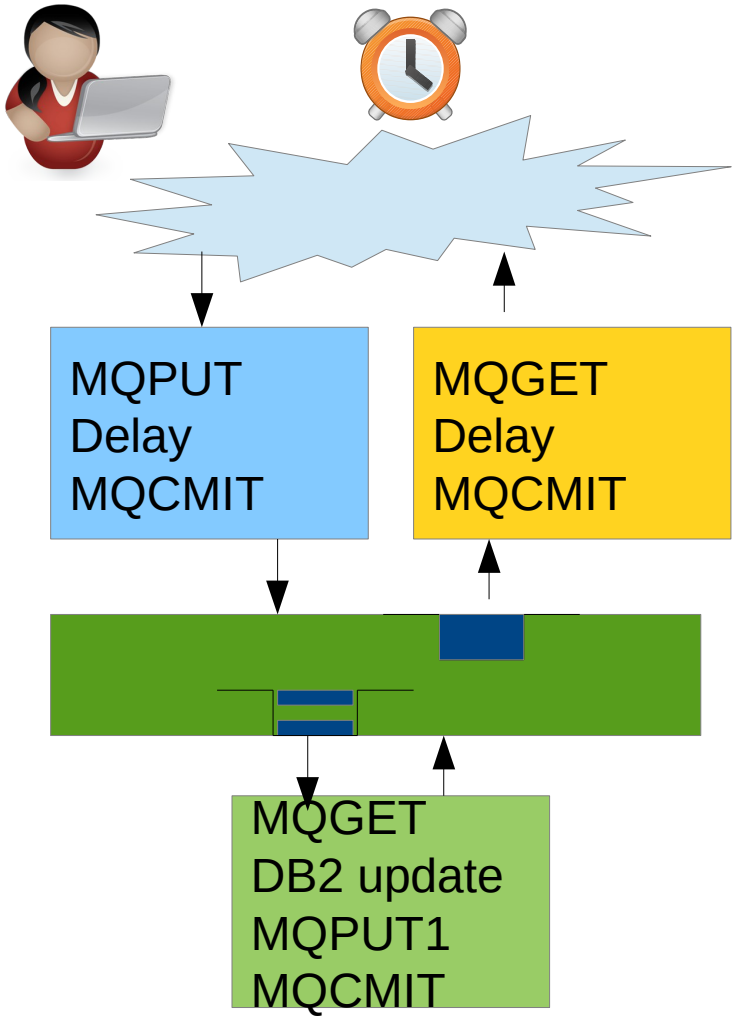
- Week 2 has higher than week 1
- Get jobs are not running at start
-



Notes

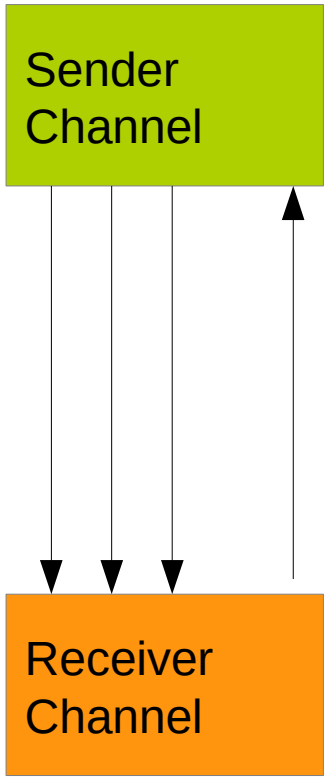
- This shows two things
- Week 2 has a higher put rate than week 1
- The gets were started much later than the puts
  - It is better for the get jobs (server transactions) to start before the puts, so the queue does not build up

Where is time spent within MQ application?



- Delay before MQPUT
- Long MQPUT request
- Long MQCMIT
- Delay before MQGET
- Long MQGET
- Long Database update
- Long MQPUT1
- Long MQCMIT
- Delay before MQGET
- Long MQGET
- Delay before MQCMIT
- Long MQCMIT

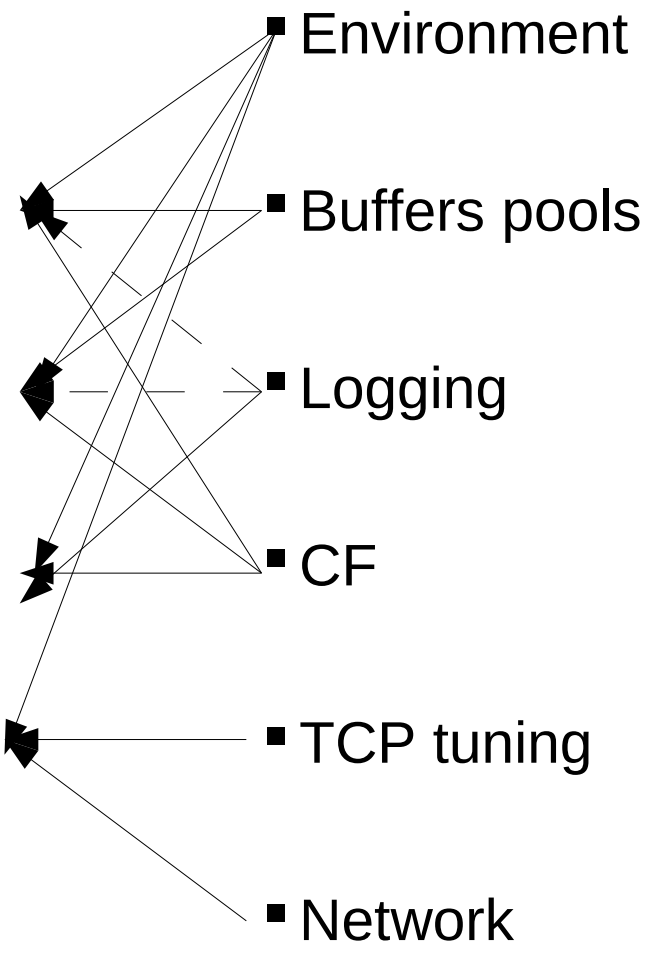
Where is time spent for channel?



- Message not available
- Loop
  - MQGET
  - TCP Send
- Put out of syncpoint = Put + commit
- TCP Receive
- MQCMIT
- Network delay
- Wait for data
- TCP Receive
- MQPUT
- MQCMIT
- TCP Send
- Network delay

What affects the requests

- Long MQPUT request
- Long MQGET
- Long MQCommit
- Send/Receive



- Environment
- Buffers pools
- Logging
- CF
- TCP tuning
- Network

Notes

- Environment: Lack of CPU either not enough engines or WLM priorities; Paging; Not enough CPU in the CF, paths to the CF are slow
- Buffer pool filling up affects Puts and gets (not commit)
- Logging performance affects Commits, Out of syncpoint puts and gets
- CF performance affects Puts , gets, Commits (during commit there are CF requests to move data from uncommitted to committed)
- TCPIP tuning parameters - such as buffer size Send 'I am full please wait', Receive – no data received
- Network, Transmission of data and Path the data takes

The areas where delays can occur

- Delays before MQ Requests
  - Lack of CPU, insufficient priority
  - Other application work
  - Use RMF and CICS/IMS/WAS statistics
- Duration of MQ puts and gets
  - Buffer pool full
  - Could be doing I/O to page sets, CF
- Duration of commit
  - Amount of data written per second, speed of disk I/O
- Network delay

First checks

- Logs
  - Are the logs busy?
- Buffer pool
  - Are the buffer pools filling up?
- CF
  - Check CF response time
- CHINIT
  - Display the channels with problems

Agenda

- Collecting MQ SMF data
- Overall MQ activity
- Logging
- Buffer pools
- CF activity
- Activity by application task and by queue
- Chinit and channels

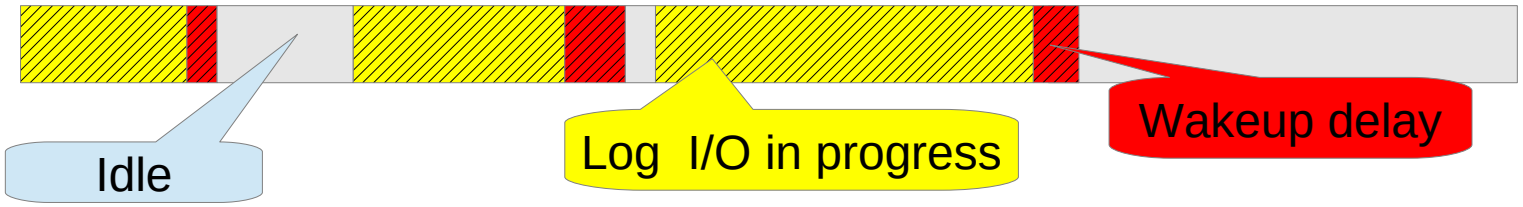
Log data

- Key metrics
  - Running out of log buffers
    - Data merged to log buffers before writing to datasets
  - How busy was the logging task?
    - Total time doing I/O in measurement period
    - Busy > 95%
  - Pages per I/O
    - The busier the system – more pages written per I/O
    - Do I need to stripe my logs?
  - I/O response time
    - 500 uSeconds is ok for DASD
    - 1000 uSeconds is ok for Mirrored DASD

Log response time -The I/O journey

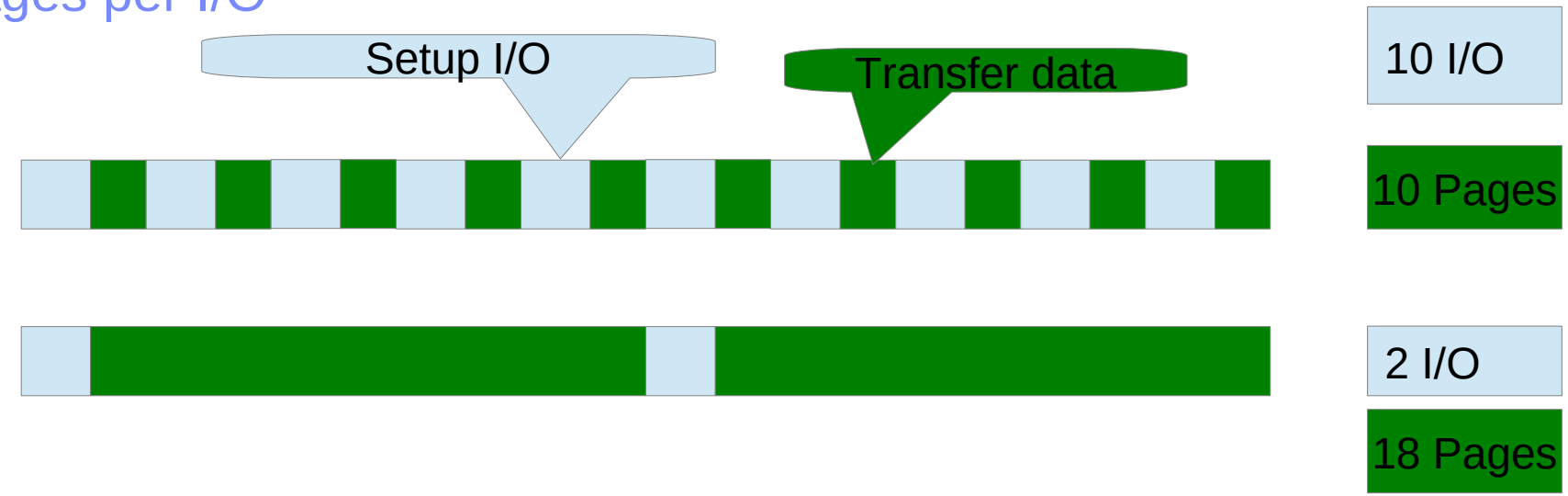
- MQ logging task issues request and waits
- Log request
  - Issue log write request and wait
    - Dual logging in parallel (write write – wait wait)
  - Z/OS issues the I/O – could be delayed because device is busy
  - Down the channel to the DASD
  - Send the data <<< This is the transfer of data
    - (DASD subsystem sends data to mirrored DASD)
  - I/O completes – posts MQ logging task
  - MQ logging task wakes up
- Response time depends on number of pages
  - Just 1 page – should be constant
  - If > 1 pages, depends on number of pages

What is log I/O busy? What % doing I/O in interval



- Elapsed time spend doing log I/O (40 seconds) + wakeup delay (0.5) = 40.5
- Interval 60 seconds
- % busy = 40.5 /60 = 68% busy.
- A “Busy” logging task > 95%
  - “Busy” = 50% could be busy for 10 minutes and idle for 10 minutes!
- There is spare capacity “in these gaps”

Pages per I/O



- Log task is 95% busy
- 'Set up' includes within z/OS and on hardware – down channels to DASD
- Writing more pages per I/O is more efficient
  - 10 I/O – one page per I/O
  - 2 I/O - wrote 9 pages per I/O
- Maximum 128 pages per I/O

Logging – what do I check?

- Did it run out of log buffers – how many requests had to wait?
  - 0 is best - small % ok.
- How many pages per I/O?
  - If small – logging can do more work
- Is logging task busy?
- Disk response time
- If writing multiple pages per I/O
  - Benefit from log striping. Write across 4 volumes in parallel
  - Use LISTCAT LEVEL(SCENDATA.MQPB) DATA ALL
    - ATTRIBUTES: STRIPE-COUNT----------4

Log data

```
I900,MQI3,2015/02/05,10:08:44,VRM:710,

  From 2015/02/05,09:38:44.273136 to 2015/02/05,10:[illegible].44.128155,
                                          duration  1800 seconds.
   Wait for buffers(should be 0):                  0 out of     14795
   Total Number of pages written:              81523
   Number of pages written/sec:                   45
   Amount of data written/sec:                     0 MB/Sec
   [illegible]ite requests:                    81424
   [illegible]quests/sec:                         45
   [illegible]I/O:                                 1
   [illegible]ead requests:                        0
   __,__,__ write requests,        CIs, Average I/O ,  ter I/O   , pages/IO
                                        time in uSec, time in uSec,
   Log 1, 1 page      81335,      81335,          621,             8,        1
   Log 1,>1 page         89,        188,          629,             9,      2.1


  Log write rate       0MB/s per copy
  Logger I/O busy : 2.85%
```

0 is good

OK response time

Single logging
Pages per I/O

Mostly 1 page /IO

0 is Low

Logging not busy

Log data

```
E41K,QMI2,2015/01/06,00:00:00,VRM:710,
  From 2015/01/05,23:45:00.008321 to 2015/01/06,00                    uration   900

   Wait for buffers(should be 0):                  0
   Total Number of pages written:               6296
   Number of pages written/sec:                    6
   Amount of data written/sec:                     0 MB/Sec
   Total Number of write requests:              5442
   Number of write requests/sec:                   6

   Pages written per I/O:                          1
   Total number of read requests:                  0
   _____,__ write requests,          CIs, Average I/O , After I/O   , pages/IO
                                          time in uSec, time in uSec,
   Log 1, 1 page          2676,          2676,          1753,              4,         1
   Log 1,>1 page            45,           472,          2983,              4,        10
   Log 2, 1 page          2676,          2676,          1729,            353,         1
   Log 2,>1 page            45,           472,          2887,            396,         9
Log write rate          0MB/s per copy
Logger I/O busy : 0.64%
```

0 is good

'High' response time
Mirrored DASD over 12KM

Mostly 1 page /IO

0 is Low

Dual logging

Logging not busy

Log data – IBM – MB Messages

```
                                          duration    55 seconds.
Wait for buffers(should be 0):                2234
Total Number of pages written:             2885274
Number of pages written/sec:                 52459
Amount of data written/sec:                    204 MB/Sec
Total Number of write requests:              26414
Number of write requests/sec:                  480
Pages written per I/O:                         109
Total number of read requests:                1378
  _____,__ write requests,        CIs, Average I/O , Aft  I/O   , pages/IO
                                       time in uSec, time in uSec,
  Log 1, 1 page          175,         175,          849,              2,         1
  Log 1,>1 page        13032,     1442462,         2473,              2,       110
  Log 2, 1 page          175,         175,          773,            143,         1
  Log 2,>1 page        13032,     1442462,         2584,             35,         0
 Log write rate      102MB/s per copy
 Logger I/O busy : 62.37%
```

Ran out of buffers

Some log read

OK response time

Many pages /IO

102 is OK

Logging not busy

- Achieved 350 MB/Second per log at Hursley
  - Tuned workload - logger 95%+ busy
- Single mover channel

My logs are busy and I have multiple pages per I/O – now what?

- Do you have multiple pages per I/O?
  - Ensure logs are striped to get parallel I/O
  - Be careful allocating striped logs a log of 1000 Cyl, 4 stripes->4000cyl
- Check log size
  - 3GB pre V8 (when using log archives on disk)
  - 4GB V8
- Archive log placement
  - Try to avoid active log volumes

Agenda

- Collecting MQ SMF data
- Overall MQ activity
- Logging
- Buffer pools
- CF activity
- Activity by application task and by queue
- Chinit and channels

Buffer pools

- Two patterns
  - Buffer pool never fills up
    - Queue depths low
  - Buffer pool always full
    - Deferred processing
- If buffer pool > 95% busy
  - All changes written synchronously to page set
    - 12 KB message – 3 pages – 3 * I/O +3 * log forces
    - 6 uSeconds -> 6000 uSeconds elapsed time
- Buffer pool > 85% busy
    - Back ground task asynchronously moves change pages to page set
- Try to keep performance critical buffer pools < 85% busy
- Checkpoint
  - Writes pages out to page set

Notes

- With full buffer pools (> 95% full) each page has to do disk I/O to the page set. For persistent messages to maintain integrity the contents are written to the log data sets as well. This is a log force requiring log I/O
- In this situation the performance of puts and gets will be dependant on the speed of the DASD. With increased message rate, there will be more contention for the page sets and so the response time will increase with load.

Buffer pool data

```
I900,MQI4,2015/02/05,10:25:21,VRM:710,
  From 2015/02/05,09:55:27.978944 to 2015/02/05,10:25:21.129407, durati n  1793
= BPool   0, Size 50000,%full now  0, Highest %full  0, Disk reads     0
 > 00 Buffs    50000  Low    49943  Now    49943  Getp    25104  Getn     1396
   00 Rio          0  STW    24983  TPW        0  WIO         0  IMW         0
   00 DWT          0  DMC        0  STL        0  STLA        0  SOS         0
= BPool   1, Size 20000,%full now  3, Highest %full  3, Disk reads     0
 > 01 Buffs    20000  Low    19216  Now    19216  Getp    34957  Getn     6898
   01 Rio          0  STW    34413  TPW        0  WIO         0  IMW         0
   01 DWT          0  DMC        0  STL        0  STLA        0  SOS         0
```

0 is good

During period

Measure of activity

< 85%

- Did buffer pool fill up?
- Was there any read activity?
- Getn – count of new 4KB pages requested – measure of activity

Buffer pool data

```
E11K,QME1,2015/01/06,13:15:00,VRM:710,
= BPool   3, Size 40000,%full now  0, Highest %full  0, Disk reads     0
 > 03 Buffs    40000  Low    39852  Now    39908  Getp    33101  Getn      207
   03 Rio          0  STW    12143  TPW      260  WIO       112  IMW        56
   03 DWT          0  DMC        0  STL        5  STLA        0  SOS         0
= BPool   4, Size 20000,%full now 12, Highest %full 12, Disk reads     0
< BPool   4, Pages written/sec     4, Pages read/sec                    0
 > 04 Buffs    20000  Low    17480  Now    17480  Getp     9705  Getn     5671
   04 Rio          0  STW    14183  TPW     3739  WIO       951  I
   04 DWT          0  DMC        0  STL     5671  STLA        0  SOS         0
= BPool   5, Size 40000,%full now  7, Highest %full 59, Disk reads 65650
< BPool   5, Pages written/sec    54, Pages read/sec                   72
 > 05 Buffs    40000  Low    16321  Now    36919  Getp   290044  Getn     7511
   05 Rio      65650  STW   185187  TPW    48842  WIO     12255  I
   05 DWT          0  DMC        0  STL    72980  STLA        0  SOS         0
= BPool   6, Size 70000,%full now 75, Highest %full 85, Di  reads 415292
< BPool   6, Pages written/sec   216, Pages read/sec                  461
 > 06 Buffs    70000  Low    10487  Now    17492  Ge  13601       Getn   113034
   06 Rio     415292  STW  1744166  TPW   194551  W     486       IMW        69
   06 DWT          5  DMC        0  STL   471664  S                SOS         0
```

Not used?

Low usage

Many reads

Many reads

< 85%

hit 85%

- Can you make BP6 bigger? Can you make it big enough?

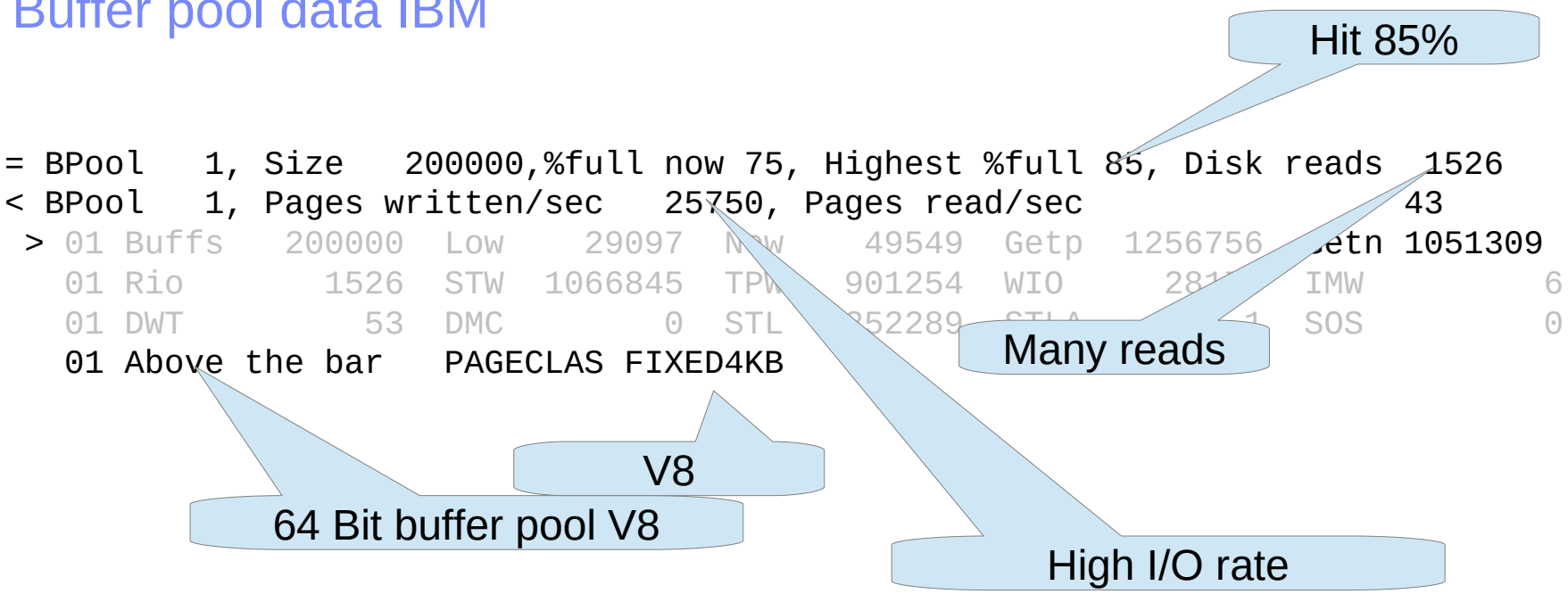
Buffer pool data

hit 85%

```
SYJ ,MQOJ,2015/02/17,04:17:28,VRM:710,
From 2015/02/17,03:47:44.740548 to 2015/02/17,04:17:28.3   08, duration  1784 seconds
= BPool   0, Size     1000,%full now 21, Highest %full 2    Disk reads     0

= BPool   1, Size     1000,%full now  1, Highest %full      Disk reads     0

= BPool   2, Size     1000,%full now  0, Highest %full  0,  Disk reads     0

= BPool   3, Size      500,%full now 79, Highest %full 85, Disk reads  7710
< BPool   3, Pages written/sec       4, Pages read/sec                   4
```

- Buffer pools look small (default?)
  - Recommend making BP3 bigger

Buffer pool data IBM



- High write rate – V8 improvements
  - –Aim to keep buffer pool < 85% full
  - –Parallel I/O writing 64 pages at a time
  - –Check your DASD for contention
- Because of high I/O rate - Used V8 FIXED4KB

Buffer pool FIXED4KB

- Need to page fix each page before each I/O
- If FIXED4KB buffer pool, page fix at buffer pool creation
  - Eliminate need for page fix for each I/O
    - Trade off CPU and real storage
  - Reduced CPU overall
  - Buffer pool is fixed – so real storage used
    - Even if buffers are not used.
- Typical QREP type behaviour – sending MB of data across a channel, and buffer pool fills up
  - Use MQ as elastic container

Can I make my buffer pool bigger?

- CSQY220I MQPC CSQSCTL Queue manager storage usage:local storage: used 401MB, free 1071MB: above bar: used 297MB, free >10GB
  - CSQY221I: Queue manager is short of local storage
- free 1071MB: You have spare storage in the region for buffer pools < 31
- ALTER BUFFPOOL(4) BUFFERS(....)
  - CSQP051I MQPC Insufficient storage for buffer pool 4 request
- Recommend %used < %80. (So about 100MB free)
  - Make changes carefully – wait for a day
- Not enough space? Can you resize other buffer pools?
- Still not enough space ? Go to V8

Notes

- The CSQY220I messages is produced hourly, or more frequently if there is a significant change to the amount of virtual storage used. It reports in use, and free storage.
- The queue manager needs some storage for its control blocks, so you cannot use 100% for buffer pools.
- If the queue manager detects it is running out of virtual storage it produces message CSQY221I. You should monitor this, and consider freeing space from a buffer pool.
- The system will not let you allocate more buffers than available storage. You will get message CSQP051I.
- It is a best practice to keep % storage used to be less than 80% of the total available.
- If your important buffer pools become > 85% full. Consider transferring storage from other buffer pools. If a buffer pool is always > 85% busy – making it smaller make not make a big difference.

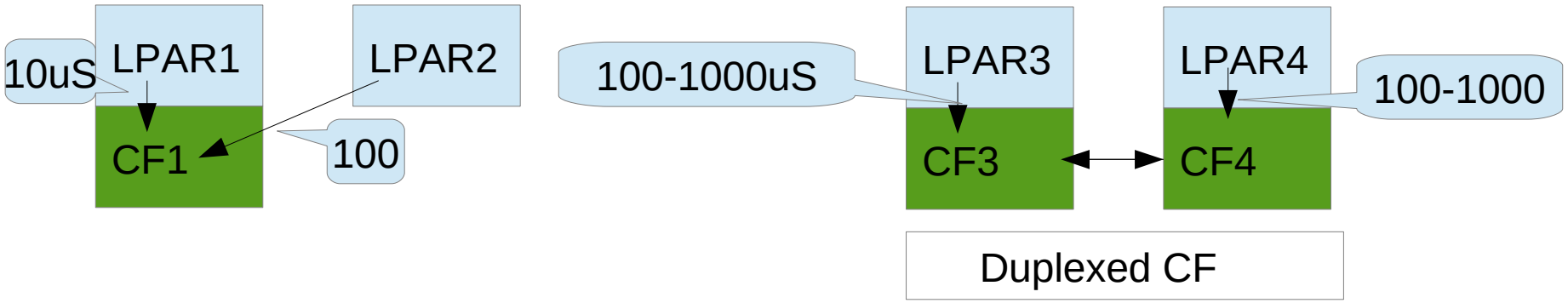
What to do now

- Collect Buffer Pool stats from a good day
  - Which buffer pools are over 85%
  - What read/write activity rate is there by buffer pool
- Identify which queues are causing I/O to the page sets at put/get time
  - Accounting class(3) and //PSIDQIO

Agenda

- Collecting MQ SMF data
- Overall MQ activity
- Logging
- Buffer pools
- CF activity
- Activity by application task and by queue
- Chinit and channels

Coupling facility



- Synchronous requests – one long instruction - 10 uSeconds
- Asynchronous – Issue request, suspend, wait, resume 800 uSecs
- Use RMF
  - Ratio of Sync and Async requests
  - Response times of Sync/ASync
- Ratio of Sync:Async can change with other work on CF
  - Monitor %CF busy, I/O response times and delays

Coupling facility RMF report – similar data from MQ reports

Typical times – same machine

Type	% requests	Response time	Std Dev
Sync	99.1	7	40
Async	0.9	832	580

Test LPAR – shared processors

Type	% requests	Response time	Std Dev
Sync	22	195	325
Async	78	250	400

Notes

- The response time can vary depending on
  - Is the CF is within the same processor
  - Does it have dedicated engines or are they shared
  - How busy the CF is
- These numbers are on the same physical machine with the a CF running within an LPAR on the machine.
  - First example had dedicated engine
  - Second example had shared engines
- Std Dev is Standard Deviation – a measure of the spread
  - Most request are within the mean +- 2 * Std Dev, so in first case the synchronous request could have been up to 87 microseconds

Coupling facility RMF report production data

- Response time different from different LPARs

LPAR	Type	% requests	Time
LPAR1	Sync	2	50
	Async	12	810
LPAR2	Sync	20	21
	Async	33	711
LPAR3	Sync	1	396
	ASync	32	581

Notes

- This shows 3 LPARs on 2 physical boxes
- It shows there can be long time differences depending on the physical box.
- If the requests are mainly asynchronous, then the CF code will regularly try a synchronous request in case it finds it is more efficient to switch modes. And the same if mainly Synchronous requests.
  - So you will usually get a mixture of Sync and Asynch request

Agenda

- Collecting MQ SMF data
- Overall MQ activity
- Logging
- Buffer pools
- CF activity
- Activity by application task and by queue
- Chinit and channels

What's happening with my queues?

- Start trace(A) class(3)
- Can produce lots of data
  - Turn on for few minutes
  - You can use ACCTQ to limit queues reporting data
- It can be expensive
  - Every CICS transaction
- It can report
  - How much CPU was used by each queue
  - How much CPU used by MQ verb
  - Why was it delayed?
  - How many message processed

What's happening with my queues?

- Deductions from the data - save you work! Look here first
  - Details in //SYS0001
  - Summarised in //TASKSUM
- Typical questions
  - Which are my hottest queue – using most CPU
  - Where are they spending time
  - Are my messages delayed
- Task and message data details in //TASK
- Queue data summarised in //QALL
- Which queues caused page set I/O? in //PSIDQIO

Notes

- MP1B has rules to print out typical analysis and eliminate the need for your to lood at detailed data. For example
  - If the average commit time is 20 milliseconds – this is very long and should be pointed out to you, so you can see if this is a logging problem, or your application is logging MB of data.
  - It is unusual that applications do get specific and get next within an application, so if both are detected it produces a warning.
- If you have 100,000 CICS transactions in the SMF data, you may get 100,000 warnings!
  - In //TASKSUM you get message xxx was produced 100,000 times. The peak number of … is yyyy.
  - This is a really good quick way to see what the transactions are doing
  –

Is anything wrong?

- //TASKSUM
  - Record# Count Value Message
  - 119226 1 265741 MQTASK12S long commit time I,IME1 ,"EPSTR001",
  - 47247 4652 1 MQTASK33W Get persistent messages out of syncpoint? EPS.EPS.NOTI.TECH
- Long(est) commit of 265741 microseconds!
  - Happened 1 time
  - You can select specific record range in MP1B
    - Firstrecord 119226
    - Lastrecord 119226
    - Detail 20
- Record 47247
  - Happened 4652 times – Spotted 1 message out of syncpoint.

Is anything wrong? - Customer 2

- //TASKSUM
  - Record# Count Value Message
  - 1558999 23 116176 MQQIST01W E11K,QME1 QIST read ahead messages > 0
  - 1206449 17 312 MQQPST08I E11K,QME1 BP 4 Read rate 312 pages per second
  - 1567789 6 561076 MQTASK01E Queue not indexed QIJ.LSB.CRO.03
  - 1758334 2 16331 MQTASK08E Long Put time due to logging SYSTEM.CHANNEL.SYNCQ
  - 5255 5194 0 MQTASK12I Get Specific and get any SA8.CPTE.TOPBRE.01
- 1567789 6 561076 MQTASK01E Queue not indexed QIJ.LSB.CRO.03
  - 6 jobs/transactions with sequential queue access
  - Highest number of messages skipped 561076
    - Record 1567789 in the file

Is anything wrong? - Customer 3

- //TASKSUM
  - SYJ ,MOOJ,2015/02/17,00:18:09,VRM:710,
  - Record# Count Value Message
  - 20270 1 38638 MQTASK08E Long Put time due to logging MQOJ.TRQ.FRROMCS1.T1
  - 744542 238 239 MQTASK01E Queue not indexed MQOJ.LOQ.MQOJ.IRDPRRCFCOMG
  - 732902 3 500 MQTASK02W High percent of no msg found MQOJ.TRQ.MQOD.T1
  - 712547 104 1 MQTASK12E Get Specific and get any BBSMVMQS.REPLY.MQOJ
  - 803264 1 12002 MQTASK14W long commit time M,"MQOJ.FRRODIC1.T1 ","BL30106AOS",
  -
- MQTASK08E Long Put time due to logging MQOJ.TRQ.FRROMCS1.T1
  - Is this put out of syncpoint?
  - 38 ms is a long time – check logging
- Two messages about logging- go and check the logging

What detailed information is there about a queue?

What is reported on a queue

- Queue name
- Time
  - CPU
  - Elapsed
    - Doing page set I/O
    - Logging
    - CF requests (sync/async)
- Generated messages (eg trigger)
- Message size
- Time on Queue
- Get count – total, valid

Example queue output summarised by queue

```
Open name                                    EPS.EPS.NOTI.TECH.C01S.EM
Base name                                    EPS.EPS.NOTI.TECH.C01S.QL
Queue type:QAlias                            EPS.EPS.NOTI.TECH.C01S.QL
CF structure name       EPS                  EPS.EPS.NOTI.TECH.C01S.QL
Get count                             9312   EPS.EPS.NOTI.TECH.C01S.QL
Get avg elapsed time                 1184 uS EPS.EPS.NOTI.TECH.C01S.QL
Get avg CPU time                      165 uS EPS.EPS.NOTI.TECH.C01S.QL
Get valid destructive                 9304   EPS.EPS.NOTI.TECH.C01S.QL
Put count                             9316   EPS.EPS.NOTI.TECH.C01S.QL
Put avg elapsed time                  163 uS EPS.EPS.NOTI.TECH.C01S.QL
Put avg CPU time                      130 uS EPS.EPS.NOTI.TECH.C01S.QL
Put1 count                             26    EPS.EPS.NOTI.TECH.C01S.QL
Put1 avg elapsed time                 145 uS EPS.EPS.NOTI.TECH.C01S.QL
Put1 avg CPU time                     122 uS EPS.EPS.NOTI.TECH.C01S.QL
Put1 num persistent                    26    EPS.EPS.NOTI.TECH.C01S.QL
inq count                               8    EPS.EPS.NOTI.TECH.C01S.QL
inq avg elapsed time                   25 uS EPS.EPS.NOTI.TECH.C01S.QL
inq avg CPU time                       25 uS EPS.EPS.NOTI.TECH.C01S.QL

Total Queue elapsed time    15747102 uS   EPS.EPS.NOTI.TECH.C01S.QL
Total Queue CPU used         4615974 uS   EPS.EPS.NOTI.TECH.C01S.QL
```

Notes

- The queue named used in the application was not the real queue used, because the OPEN queue name is different from the BASE name. We can see this is a Queue Alias.
- Usually the time for an MQPUT(1) and an MQGET should have elapsed time close to the CPU time. If not then there will be some delays – perhaps by latches, pageset I/O or log forces. So compare these figures.
  - The put1 figures look OK
  - The get figures are very different – you need to find out why.
- At the bottom is the total elapsed time and CPU time is listed. You can select lines in the file with 'Total Queue CPU', delete the other lines, then sort this file and see the queues which use lots of CPU.
-
- If you have queues with ACCTQ(NO) then data will not be displayed

Which queue use most CPU?

- //QALL
  - –X all; f 'Total Queue CPU used' all; delete all x
  - –Sort 35 50 d
- Example

4615974 uS **EPS.EPS.NOTI.TECH.C01S.QL**
3071565 uS EPS.EPS.IDIS.DDNB.QL
2896329 uS COB.COB.ECH1.PTRT.QL
2439066 uS COB.COB.ECH1.PTRB.QL
1415454 uS EPS.EPS.NOTI.TECH.C00S.QL
822096 uS MCF.HOP.FFIN.XCT0.AC1S.QL

Queue EPS.EPS.NOTI.TECH.C01S.QL is interesting

- List just the jobs with specific queue name
  - –MP1B: Queuename EPS.EPS.NOTI.TECH.C01S.QL
- Detail 20 to get maximum information

```
Get count                    9312
Get avg elapsed time         1184 uS
Get avg CPU time              165 uS
Get suspended time           1003 uS
Get log force count          9304
Get log force elapsed time   1004
Get TOQ average            386753 uS
Get TOQ maximum           6353112 uS
Get valid count              9304
Get valid destructive        9304

Get size average           32754 bytes
Get Dest-Next                9304
Get Browse-Next                 8
```

- Get out of Syncpoint!

Notes

- We can see a big difference in the elapsed and CPU time of the get calls.
- We can see

Which queue use most CPU? Different customer

- //QALL
  - –Same as individual task queue data
  - –Include 'Total Queue CPU used' records, delete all x
  - –Sort 35 50 d
- Example

```
2398998709 uS    QIJ.LSB.CRO.02
1731338272 uS    QSO.ASYN.PLANET.01
 612038927 uS    QSO.MOS.PP01A
 227497532 uS    QIJ.LSB.CRO.03   <<  queue was not indexed
  77230604 uS    SW6.WFSF.ALIM.02
  76540107 uS    SW6.WFSF.ALIM.01

  72739721 uS    QPASA.CE383B7BA9EF4F5A
```

What about Queue QIJ.LSB.CRO.03

```
Queue indexed by NONE                          QIJ.LSB.CRO.03
Get count                    561078            QIJ.LSB.CRO.03
Get avg elapsed time            294 uS         QIJ.LSB.CRO.03
Get avg CPU time                 70 uS         QIJ.LSB.CRO.03
Get suspended time              190 uS         QIJ.LSB.CRO.03
Get pageset total count       26264            QIJ.LSB.CRO.03
Get avg pageset elapsed time     34 uS         QIJ.LSB.CRO.03
Get total empty pages       4164521            QIJ.LSB.CRO.03
Get TOQ average         558 073 597 uS         QIJ.LSB.CRO.03
Get valid count              561077            QIJ.LSB.CRO.03
Get valid destructive        561077            QIJ.LSB.CRO.03
Get size maximum                156 bytes      QIJ.LSB.CRO.03
Get Dest-Specific            561078            QIJ.LSB.CRO.03
Get persistent count         561077            QIJ.LSB.CRO.03
Curdepth maximum             561076            QIJ.LSB.CRO.03
```

- Queue get specific – queue is not indexed
  - Many empty pages skipped
  - Page set I/O occurred
- Report: 1567789 6 561076 MQTASK01E Queue not indexed QIJ.LSB.CRO.03

Are my messages being delayed – how long do my messages exist for (Time On Queue)?

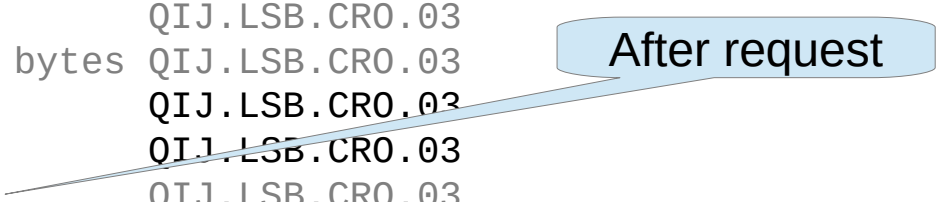
- Time on Queue = Time between put and get.
  - T0 PUT
  - T1 Commit
  - T2 GET
- Time On Queue (TOQ) = T2 – T0
  - A large TOQ – is this
    - (T1 – T0) being large and (T2 – T1) small.
      - Channel with Batchint > 0
      - Long duration Units Of Work
    - T1 – T0 small, T2-T1 large.
      - Triggered transaction?
      - Transaction busy before doing get
      - Not enough servers

Are my messages being delayed - single task

- Time between put and get.
  - Time 0 Put A
  - 1 Put B
  - 10 Commit
  - 11 Get A then lots of DB2 updates
  - 100 Get B ….
- Minimum TOQ 11 – message A
- Maximum TOQ 99 – message B

What about Queue TOQ for QIJ.LSB.CRO.03

```
Queue indexed by NONE                              QIJ.LSB.CRO.03
Get count                          561078          QIJ.LSB.CRO.03
Get avg elapsed time                  294 uS       QIJ.LSB.CRO.03
Get avg CPU time                       70 uS       QIJ.LSB.CRO.03
Get suspended time                    190 uS       QIJ.LSB.CRO.03
Get pageset total count             26264          QIJ.LSB.CRO.03
Get pageset elapsed time               34 uS       QIJ.LSB.CRO.03
Get total empty pages              4164521         QIJ.LSB.CRO.03
Get TOQ average          558 073 597 uS             QIJ.LSB.CRO.03
Get valid count                    561077          QIJ.LSB.CRO.03
Get valid destructive              561077          QIJ.LSB.CRO.03
Get size maximum                      156 bytes    QIJ.LSB.CRO.03
Get Dest-Specific                  561078          QIJ.LSB.CRO.03
Get persistent count               561077          QIJ.LSB.CRO.03
Curdepth maximum                   561076          QIJ.LSB.CRO.03
```



- Time on Queue 558 seconds!
  - This application is draining a deep queue

Agenda

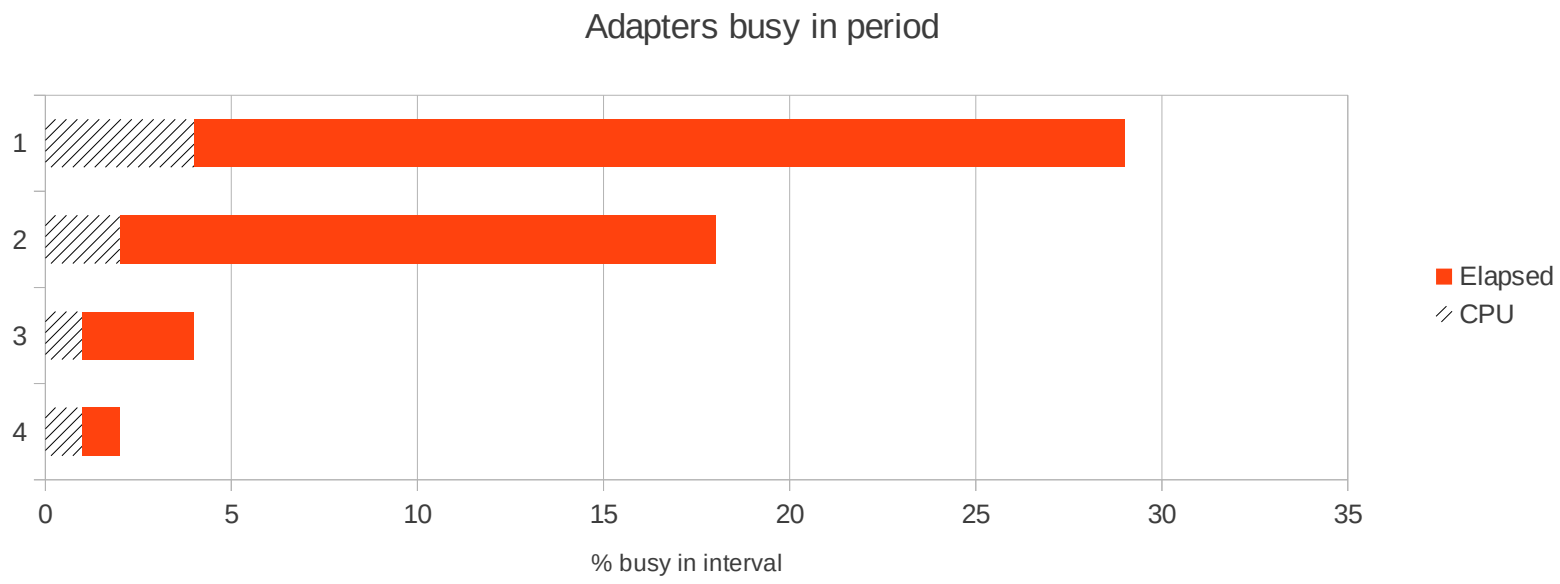
- Collecting MQ SMF data
- Overall MQ activity
- Logging
- Buffer pools
- CF activity
- Activity by application task and by queue
- Chinit and channels

What can you do with the CHINIT SMF data?

- New in V8
- Statistics
  - Maximum number of channels used in interval
  - Adapter, Dispatcher, SSL, DNS server; TCB usage
- Accounting
  - Like Display Channel Status + more fields
- Low cost
  - Records produced when stats collected
  - Or if channel ends in the interval

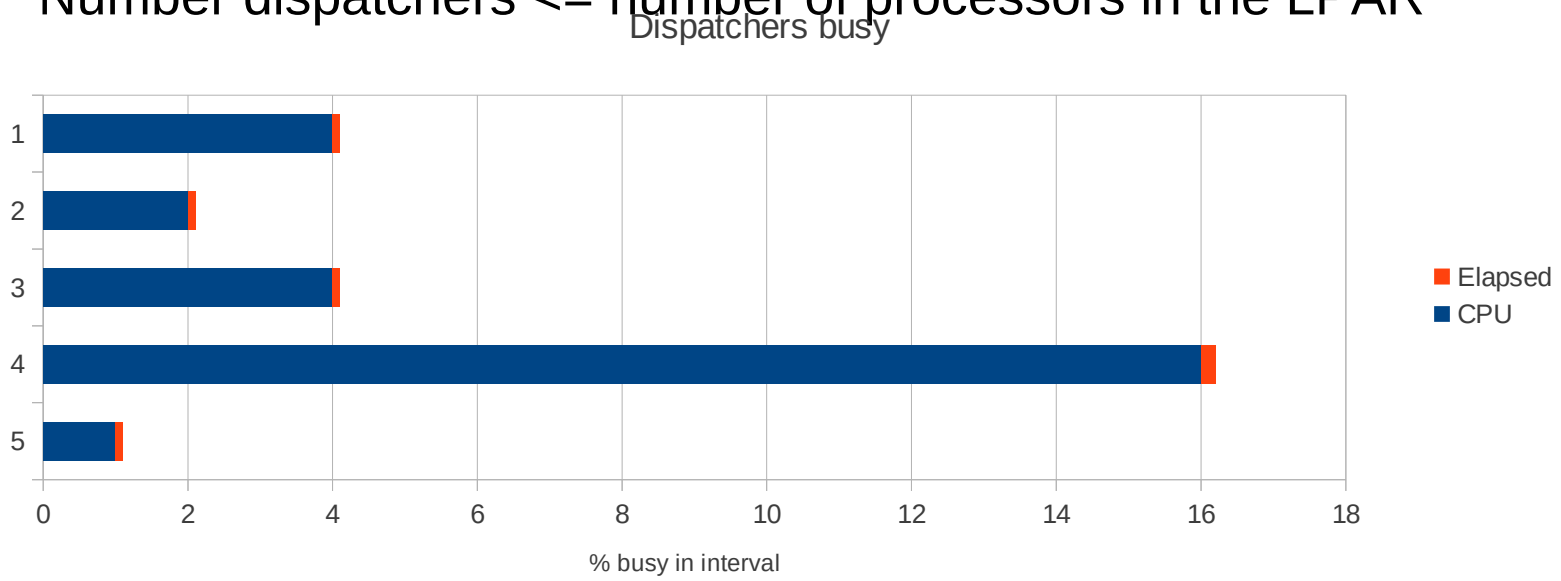
Chinit Statistics - Adapters

- Executes MQ requests, so can have large elapsed time and little CPU
- Pool of TCBs, first free used
- If all used then need more!
- Check you have some unused adapters



Chinit Statistics - Dispatchers

- There is Channel to Dispatcher affinity
  - Uses the same dispatcher for duration of the connection
  - Stop and restart channel may use a different dispatcher
- Do you have a hot dispatcher (> 80% busy?)
  - Few channels
  - Add more dispatchers
  - Stop and restart channels
  - Can vary from day to day – depends on start order
- Number dispatchers <= number of processors in the LPAR



Channel accounting data – sender channel

Records prefixed with
`Channel Address  eg  TO_MVSCA  winvsca.hursley.ibm.com`
Data records

```
..Connection name                winmvsca.hursley.ibm.com
..Batch size                                   50
..Messages/batch                             48.8
..Number of messages                       33,896
..Number of persistent messages            33,896
..Message data                        120,263,008   114 MB
..Persistent message data             120,263,008   114 MB
..Total bytes sent                    120,266,176   114 MB
..Total bytes received                     19,996    19 KB

..Bytes sent/Batch                        173,044   168 KB

..Bytes received/second                       137   137  B/sec
..Bytes sent/second                       829,421   809 KB/sec

..Net time average                            116 uSec
..Net time min                                 71 uSec
..Net time max                             14,481 uSec
```

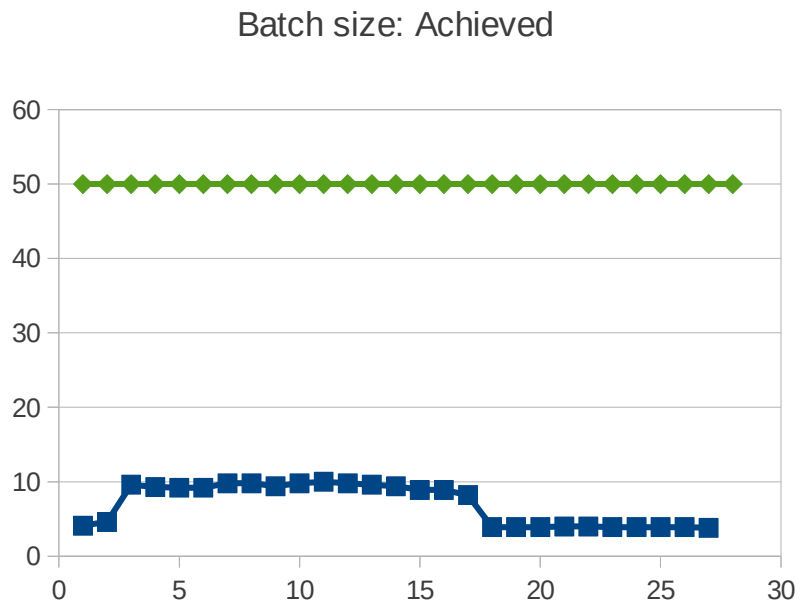
What can you do with the Channel accounting

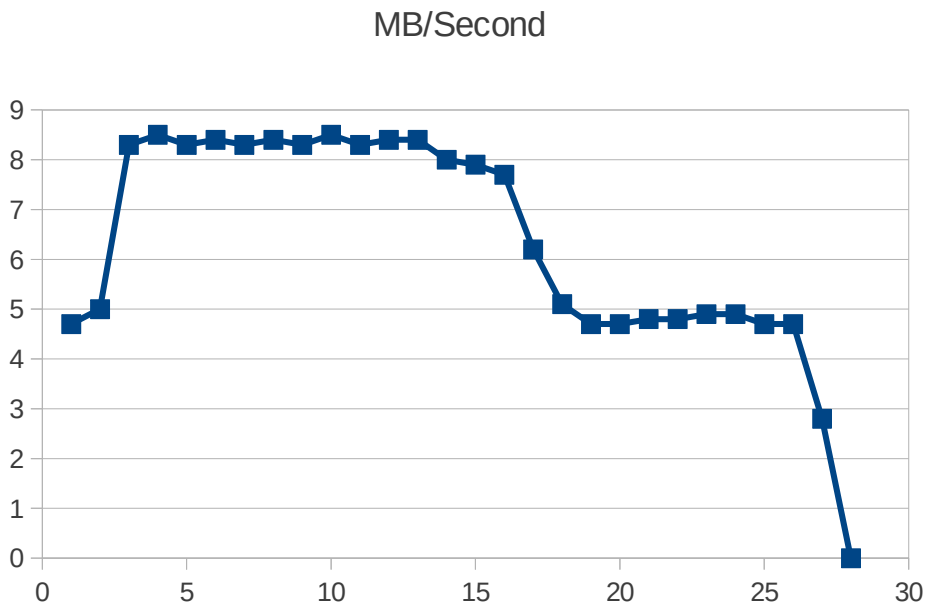
- What are my hot channels – use ispf editor
  - Bytes received/second, Bytes sent/second
- What is the nettime
  - Net time max,average,min
  - When was maximum ? *Net time max date&time*
- Does my channel have more capacity?
  - Number of full batches
  - Xmitq empty count – queue was empty
  - XBATCHSZ & batch size
  - Bytes sent/Batch (Batchint)

Notes

- You can edit the output and include only the total bytes row, then sort on the number column to see channel with highest thoughput
  - May want to do it on rate – in case channel started and stopped in the interval
- Max nettime
  - Can your correlate this to when you have a slow down?
- `Messages/batch` the XBATCHSZ, tells you if you have spare capacity. If = batchsize it was always busy.
  - May have hit batchlim – so xbatchsz will be smaller than batch size

Chinit: channel accounting





What's happening to my Channel?

- Ping
  - //RUN1GB EXEC PGM=IKJEFT01,REGION=0M
  - //SYSTSPRT DD SYSOUT=*
  - //SYSTSIN   DD *
  - ping winmvsac (count 1000 verbose length 32768
  - /*
- Across the Hursley site
  - Packets: Sent=1000, Received=1000, Lost=0 (0% loss)
  - Approximate round trip times in milliseconds:
  - Minimum=1.03 ms, Maximum=10.19 ms, Average=1.13 ms, StdDev=0.50 ms
- Hursley to Mainz in Germany
  - 64 bytes Minimum=22.37 ms, Maximum=32.30, Average=24.08
- UK to the US
  - 64 bytes Minimum=94.39 ms, Maximum=95.54, Average=94.71

What's happening to my Channel Network

- NETSTAT
  - –NETSTAT ALL (CLIE Q02CHIN
  - –NETSTAT ALL (ORT 1.2.3.4+5678
- Output – from start of connection
  - CongestionWindow: 0000065536 >= 64K is good
  - SndWnd: 0000065536 >= 64K is good
  - MaxSndWnd: 0000065536 >= 64K is good
  - Round-trip information: Smooth trip time: 0.000
  - SmoothTripVariance: 1.000
  - ReceiveBufferSize: 0000065536 > 64K is good
  - SendBufferSize: 0000065536 > 64K is good
  - ReceiveDataQueued: 0000000000 ? Problem
  - SendDataQueued: 0000000000 May be transient
  - ReXmt: 0000000000 = 0 is good
  - ReXmtCount: 0000000000 = 0 is good
  - DupACKs: 0000000000 = 0 is good

Current remote end buffer available

0.00 0 ms is good

Lost packets

Notes

- Use NETSTAT ALL (CLIENT MQ02CHIN to list all the connections for the chinit. Good idea to do it in batch because of potential large number of connections
- Then use NETSTAT ALL (IPPORT 1.2.3.4+5678 for IP address+port
- Netstat data is from start of connection
- Send window is the buffer size at the remote end. 64KB is good
- If there is congestion this can go down, so compare with max send size
- Round trip time is as seen by TCPIP
  - Variance tells you how much variation there is (eg range)
- ReXMT is the number of packets dropped and retransmitted. If this is 0 it is good. If > 0 then this indicates there was a problem at some time. Wait for a period and try again. If numbers go up – this indicates a problem
-

Agenda

- Collecting MQ SMF data
- Overall MQ activity
- Logging
- Buffer pools
- CF activity
- Activity by application task and by queue
- Chinit and channels
- Summary

What to do when you get back to the office...

- Collect SMF statistics regularly
  - Collect accounting for short period
- Use SMF data
  - When you think you have a problem
  - Is my workload growing?
  - Health check to see if applications are well behaved.
- Look at log stats
- Are buffer pools filling up
- CF response time
- What about my channels; What are high use channels
  - Adapters?
- Look at the 'problem' messages produced by MP1B

Thank you

Any questions?