IBM MQ and Db2 – MQ Verbs in Db2 & Q-Replication 24657

Lyn Elkins
elkinsc@us.ibm.com

Agenda

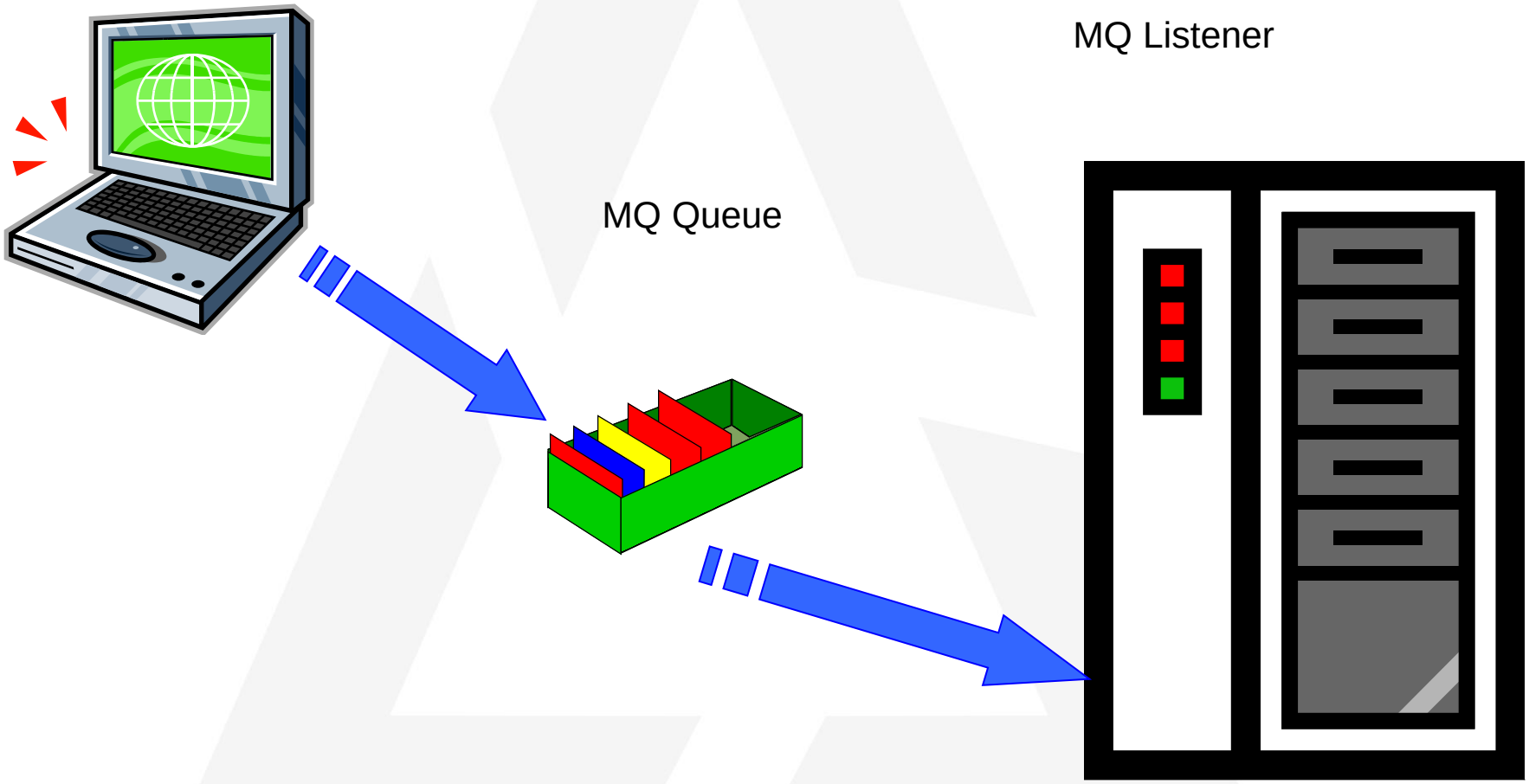
- The MQ Functions in Db2

- QREP – aka InfoSphere Replication Server
  - What it is and how it works
  - Potential bottlenecks
  - Performance Improvements
  - What's been done in MQ for QREP
  - Options and Requirements
- The last words

Db2 and MQ Functions

- MQListener
- MQSEND
- MQREAD
- MQRECEIVE

MQListener



MQListener

- The Db2 MQListener is a combination of Db2 and IBM MQ functionality
  - Db2 Process will listen for messages on a queue
  - Invoke a stored procedure to act on the message
  - Possibly send a reply
- Often used to send updates asynchronously, when the data is ‘preformatted’
  - ‘Near realtime’ updates with all the advantages of queuing!
  - Growth in usage for event and audit trail collections

MQListener - continued

- The MQListener on z/OS:
  - Provides both 1 and 2 phase commit for resources
  - Runs in USS
  - Configured via database updates
    - SYSMQL.LISTENERS table
- From an MQ perspective:
  - Standard IBM MQ administration
  - Good recommendations for DLQ, Backout queue and threshold

MQListener - continued

- The Listeners table entries include:
  - Configuration name
  - Queue manager
  - Input queue
  - Node – unused
  - Schema – the schema name for the stored procedure to be invoked
  - Name – the name of the stored procedure
  - Type – unused
  - Instances – number of instances of the procedure that can be invoked
  - Wait – the number of milliseconds that the listener should wait before checking for a new message

Db2 IBM MQ Functions

- Db2 MQ Services
- Db2 MQ Policies
- MQSEND
- MQREAD
- MQRECEIVE

MQ Service Definitions in Db2

- Typically maintained by the Db2 admin
- Defines the MQ resources that the Db2 MQ functions use
  - This is the connection between Db2 and IBM MQ
- Hold the following information:
  - Service Name
  - Queue manager
  - Queue name
    - This field is called ‘Input queue’
  - CCSID
  - Encoding
  - Descriptions

MQ Policy Definitions in Db2

- Typically maintained by the Db2 admin
- Defines the MQ quality of service for the Db2 MQ functions
- Holds the following information:
  - Policy Name
  - Send Values:
    - Priority
    - Persistence
    - Expiry
    - Retry Count
    - Retry Interval
    - Message Type
    - Reply to Queue
    - Reply to Queue Manager
  - Receive Values:
    - Wait interval
    - Accept Truncated Message
    - Open Shared
  - Syncpoint

MQSEND

- MQSEND is a SQL-centric way of putting messages on a queue
- The parms include:
  - The Send Service
  - The Service Policy
  - Message Data
  - Correlid

MQREAD and MQRECEIVE

- MQREAD and MQRECEIVE are SQL-centric way of retrieving messages from a queue
- The parms include:
  - The Send Service
  - The Service Policy
  - CORRELID for MQRECEIVE
- MQREAD and MQREADCLOB are non-destructive MQGET operations
- MQRECEIVE and MQRECEIVECLOB are destructive MQGET operations

Another creative use of Db2 and MQ

SHARE
EDUCATE • NETWORK • INFLUENCE

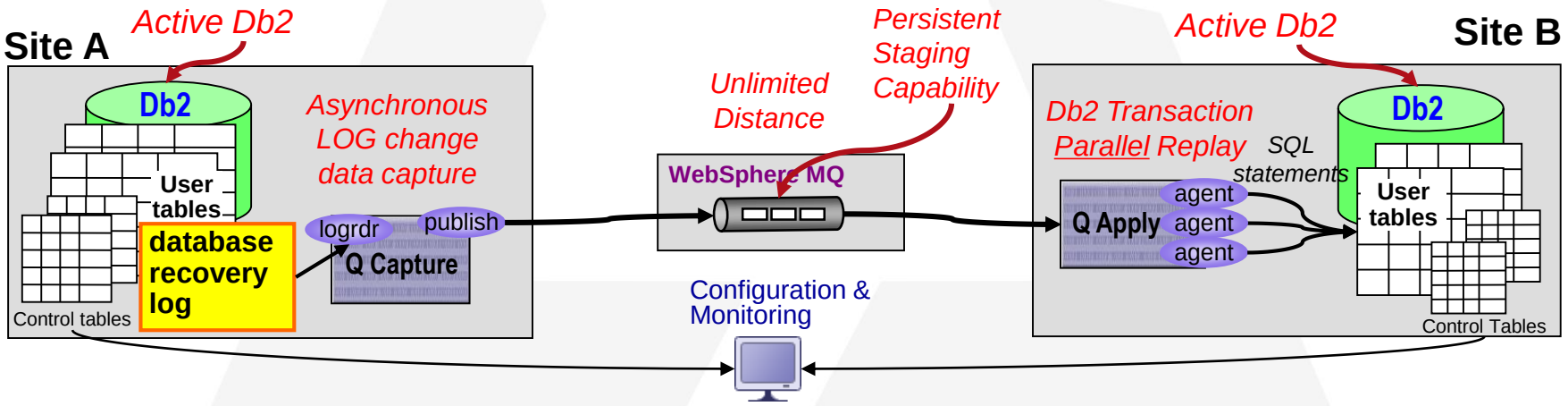
- A large high volume customer wanted to reduce CPU costs, but wanted to use client connections to z/OS hosted services
  - Before there was the VUE version of MQ
    - Before the Client Attach Feature was retired even!
- The Db2 Connect code was modified to be offload-able
  - A reduction in GP CPU costs onto zIIPs or zAPPS or whatever
- The customer used Db2 connect to invoke a Db2 Stored procedure, using the MQ functions instead of the MQ Client
  - Saving them significant GP CPU costs – even today because the CHIN does not have any offload-able code`

Q Replication

- What is it?
- The MQ-centric view
- Where have we (IBM) seen issues
- What's been done to help
- What's new in MQ V8 that may help QREP
- Additional information

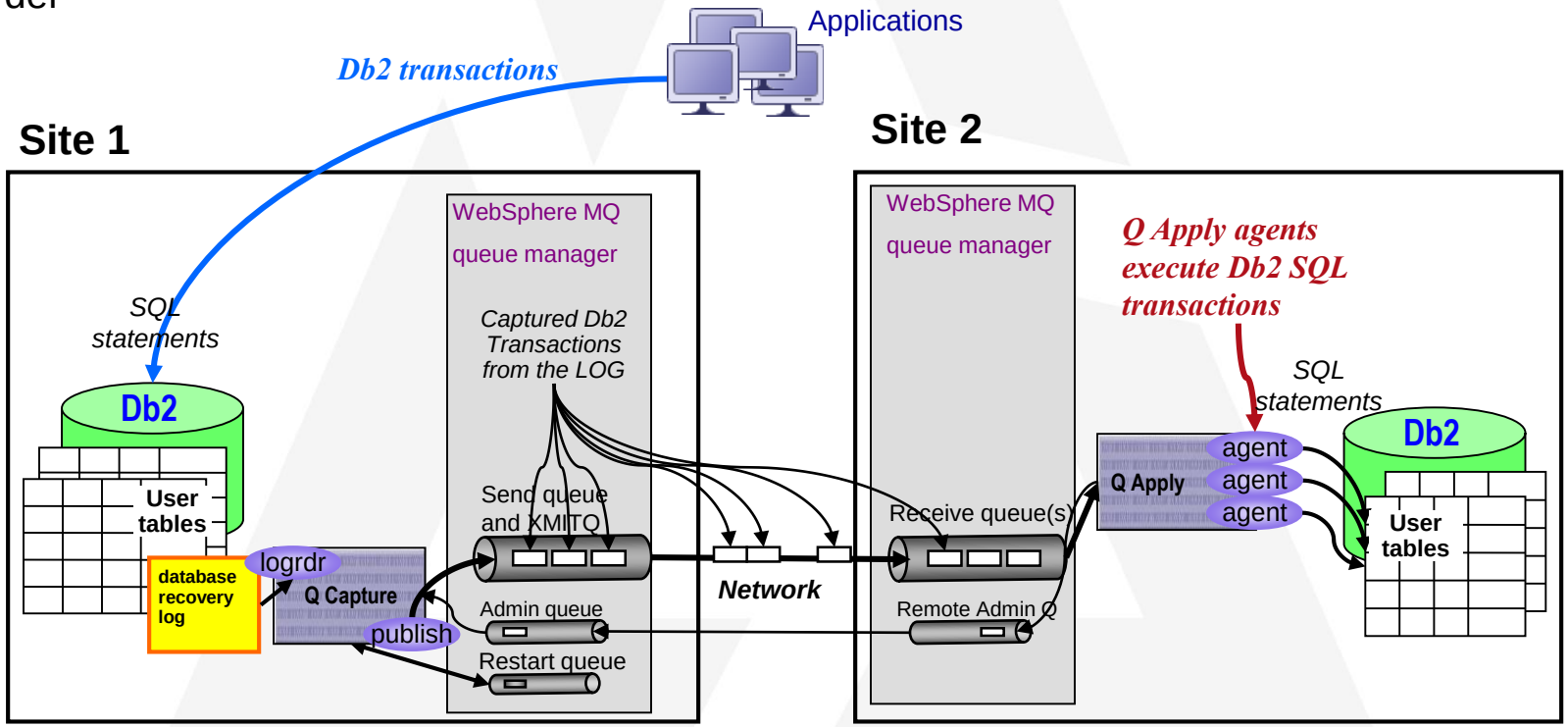
Q Replication Technology

- Part of the *InfoSphere Data Replication* product
- A software-based asynchronous replication solution
  - For Relational Databases
  - For selected tables/columns/transactions/operations
  - Changes are captured from the database recovery log; transmitted as (compact) binary data; and then applied to the remote database(s) using SQL statements. Technology is ***log-capture/transaction-replay***
- **Each Db2 is ACTIVE** and can be used for READ/WRITE operations

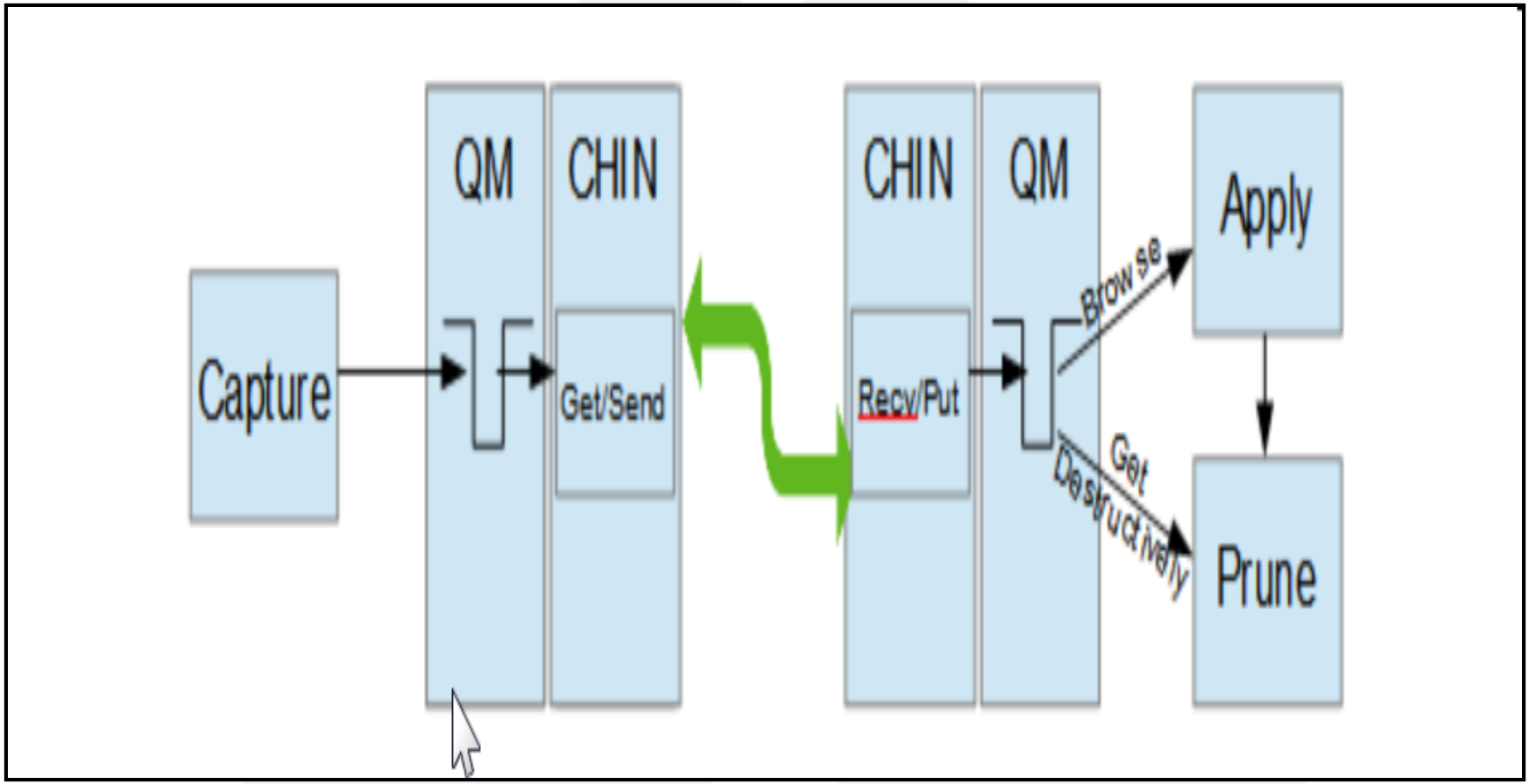


Q Replication Process

- Capture and Apply are Db2 applications
- Parallel Capture program publishes captured database *transactions* as compact MQSeries messages, only the data is sent
  - One MQ message = One captured Db2 transaction
- Apply program replays transactions by rebuilding SQL statements
- Each component runs independently and can be individually stopped/restarted at any time, in any order



Q Replication – the MQ View



Q Replication Potential Bottlenecks

- Global potential Bottlenecks:
  - CPU
  - I/O Speed

- MQ potential bottlenecks
  - MQ logging –
    - Log files are limited to 4G
    - Log files should be striped across 4 volumes
    - Building efficient messages sizes, helps logging as well.
  - MQ channel (single TCP/IP connection) for OLTP transactions (small messages).
    - Create good consistency groups, using multiple queues (QREP Version 10.2).
  - MQ bufferpool filling up. Reading or writing to the pageset can reduce replication throughput by 4 times.

Q Replication Potential Bottlenecks

- QREP potential bottlenecks:
  - Message size - Use QREP batching to build efficient message sizes.
    - Fewer larger messages can be more efficient
  - Capture publish thread might become CPU bound
    - This thread includes MQPUT and COMMIT elapsed times.
  - Workload contention at the target
    - Locking and/ or dependency of transactions updating the same rows
    - Can be caused by applications running at the target or by multiple apply agents applying transactions in parallel.
  - Apply program limited to one data sharing member.

MQ Enhancements to support Queue Replication

- Sequential Read Ahead
  - For small messages (< 120K)
  - Part of Base MQ as of V7.1
  - Pre-stage messages from deep queues for a pageset into the MQ buffer pool
  - RECOVER QMGR(TUNE READAHEAD ON)
- Large Message Read Ahead
  - Part of Base MQ as of V8
  - Parallelizes the disk read operations for large messages
  - RECOVER QMGR(TUNE RAHGET ON)
  - Also requires non-serialized pageset access
    - Contact support for more info

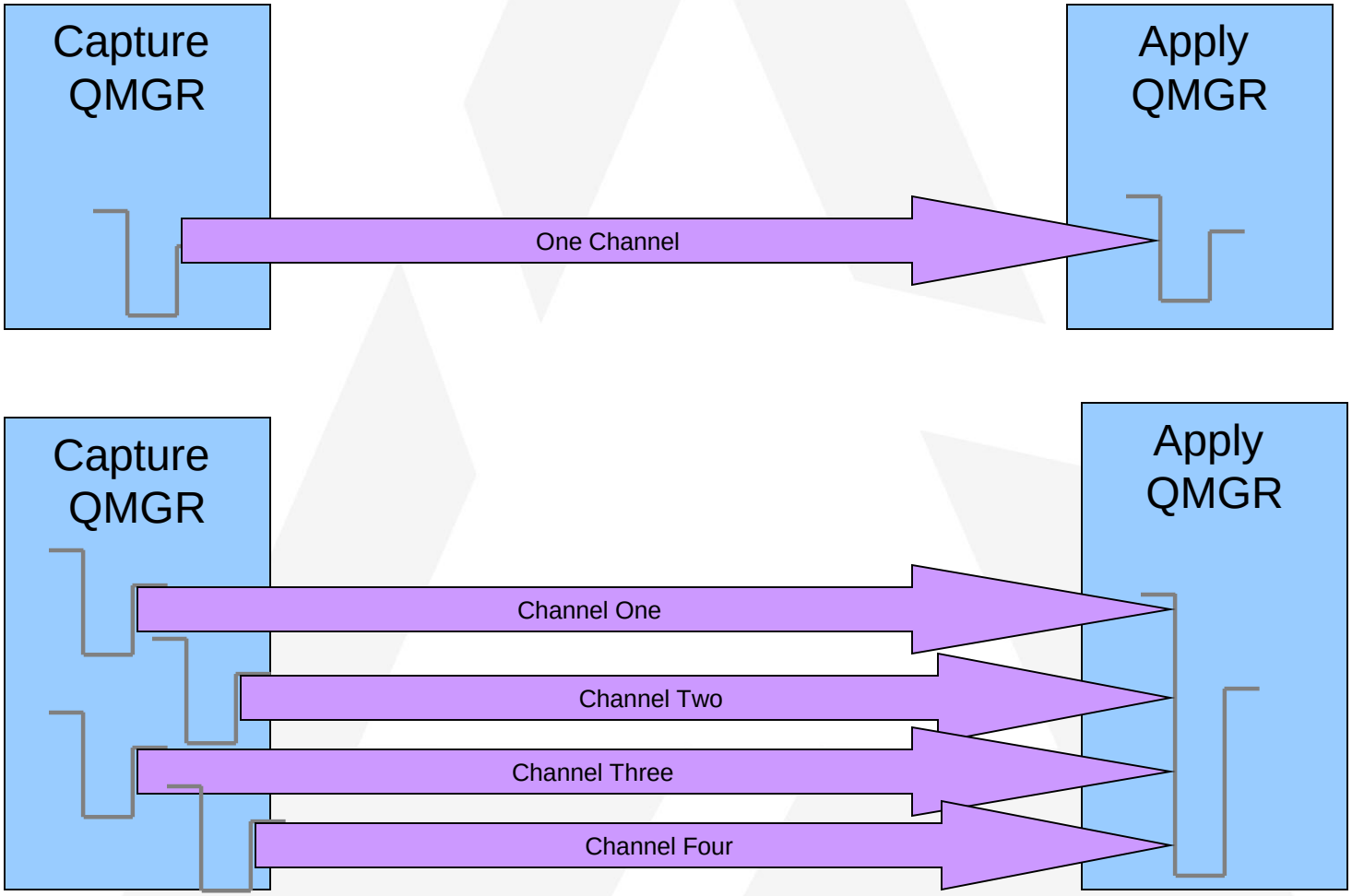
Warning

- If you are using MQ version 8, please make sure you have PI53268 applied: https://www-01.ibm.com/support/docview.wss?uid=swg1PI53268

Queue Replication – enhancements for performance

- **Using parallel send queues (z/OS)**
  - **Can reduce end-to-end replication time under some circumstances**
    - Check for growing transmission queue depths
    - Growing latency
  - Can use up to 4 parallel send queues on Capture
  - Each capture queue:
    - Can use a different MQ resource pool
    - Can be transported across an individual channel
    - Can make better use of network bandwidth than a single stream of small messages

Before and After



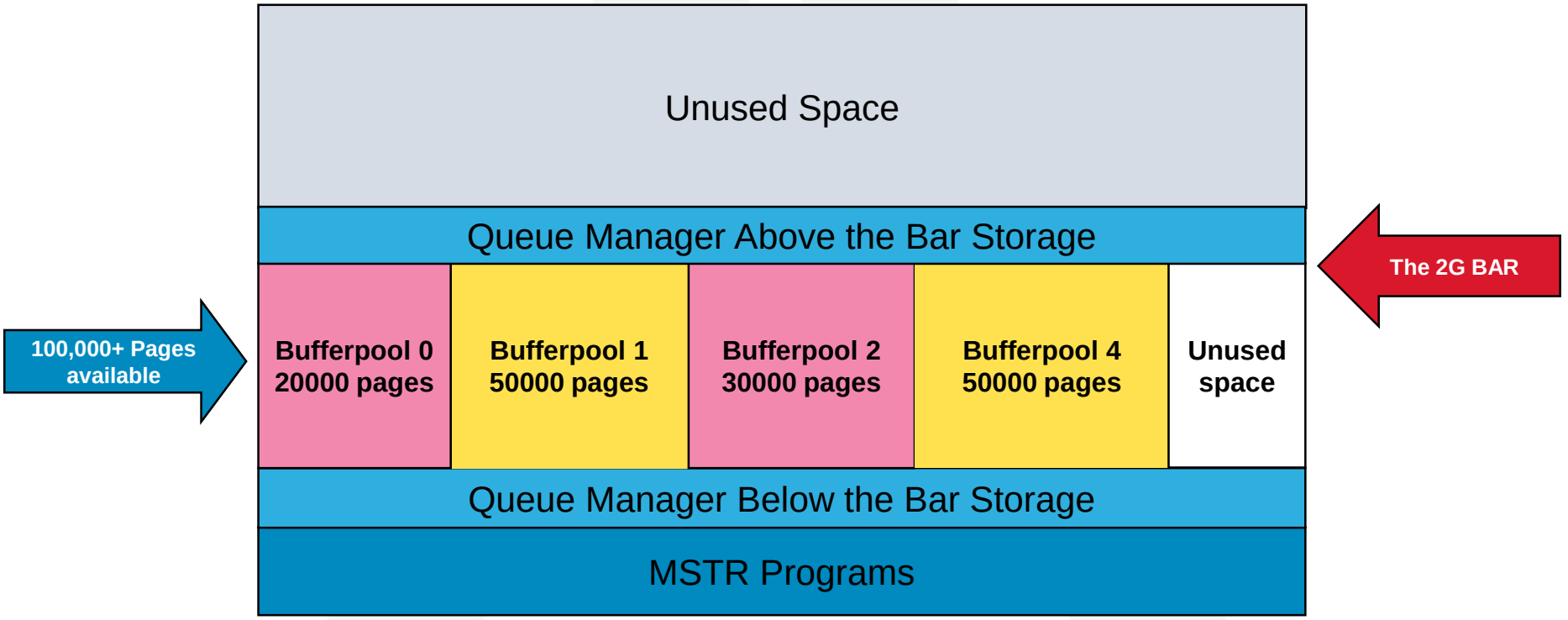
Queue Replication – Misc. notes

- The QREP development team has also looked at:
  - Using MQ log compression
    - Very data dependant
    - CPU consumption is often very high for no benefit
- If the queue manager is dedicated to QREP:
  - Setting the maximum number of channels (MAXCHL) low can benefit throughput
  - The CHIN associates channels with TCBs, trying to spread the channels evenly. If the MAXCHL is high and the actual number of running channels low, all channels may end up running on a single TCB.

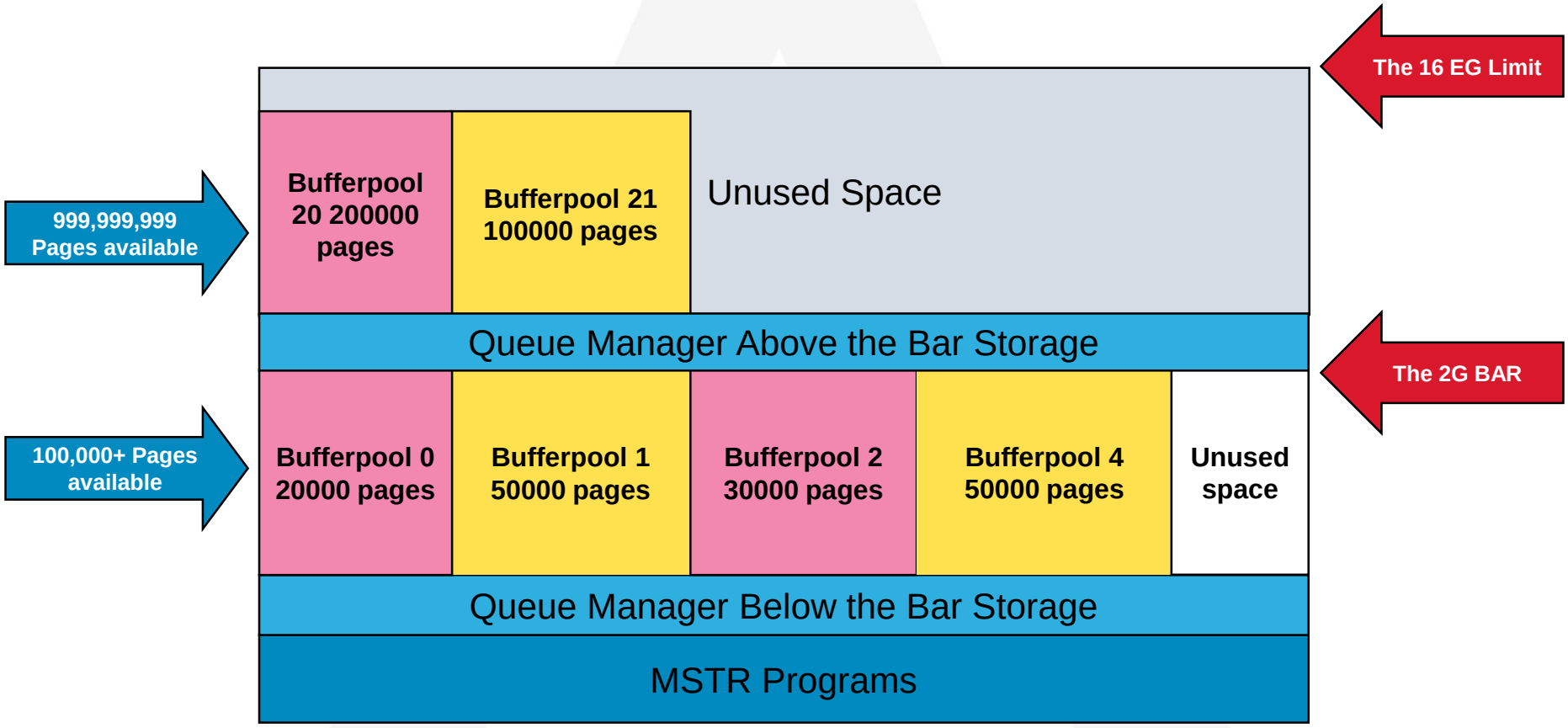
MQ V8 – Buffers above the bar

- A key enhancement to MQ was the addition of buffers above the 32-bit bar
  - This will allow larger bufferpools
    - Can hold a complete 64G queue in memory
    - Reduces the possibility of pageset I/O
    - Reduces the possibility of buffer thrashing
  - Performance numbers for QREP-style application looked good when there is enough memory to back up the above the bar pages.
  - See the MP1J performance report

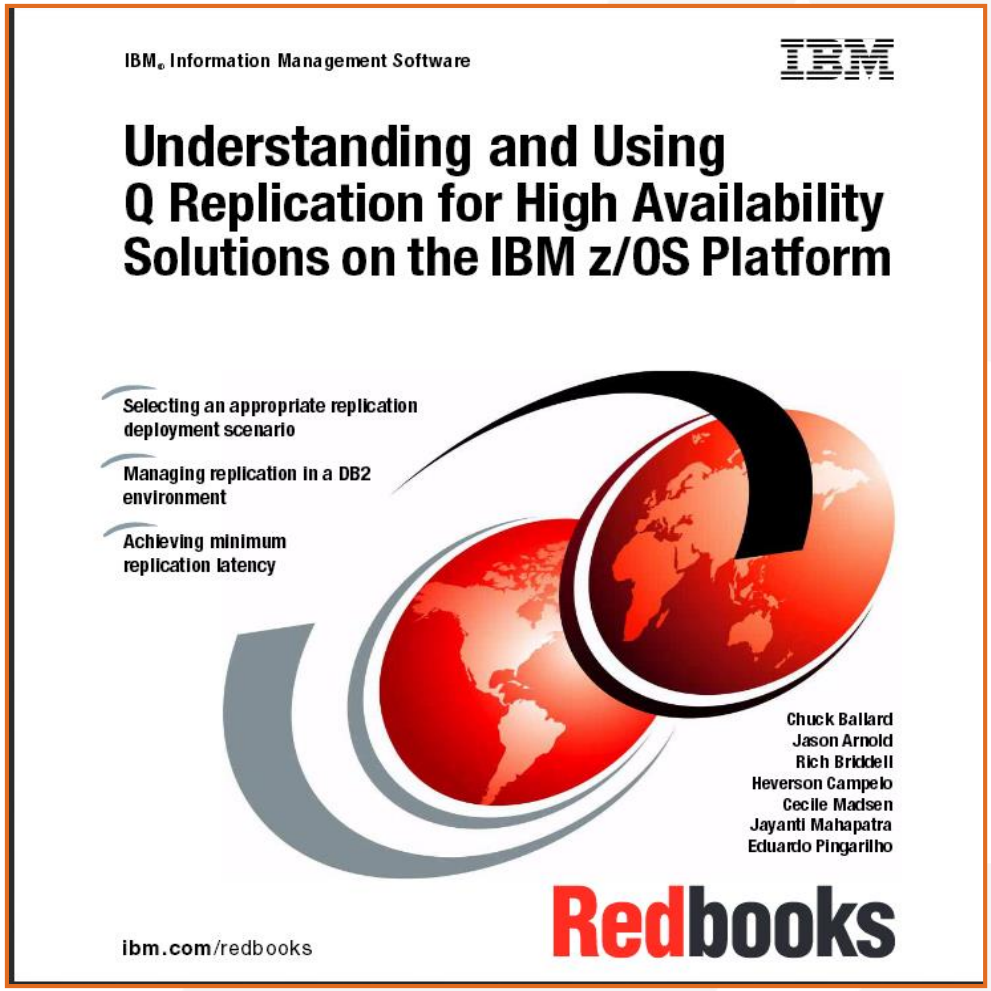
MQ Address Space prior to V8



MQ Address Space after V8



More Information



Note that this is publication SG24-8154-00. There was an earlier Redbooks publication by this same name for the replication server at V8.

More Information

- Redbooks and Redpapers:
  - http://www.redbooks.ibm.com/redpapers/pdfs/redp4947.pdf
  - http://www.redbooks.ibm.com/abstracts/sg248154.html?Open
- MP16 – the 'bible' for MQ on z/OS:
  https://github.com/ibm-messaging/mqperf/blob/gh-pages/mp16.pdf
- QREP InfoCenter:
  https://www.ibm.com/support/knowledgecenter/SSDP5R_9.7.5/com.ibm.swg.im.iis.prod.repl.nav.doc/topics/iiyrqcncreplepovu.html

And before I go…..

- One last word on MQ and Db2
  - If you want to use queue sharing, you still have to have Db2 data sharing even if you use Shared Message Data Sets for message offloads
  - MQ still uses the Db2 DSG for information about shared objects

Shameless Promotion

SHARE
EDUCATE • NETWORK • INFLUENCE

- The WSC offers TWO MQ Wildfire Workshops:
  - MQPERF – MQ for z/OS Performance and SMF Evaluation
    - 2 Days
  - MQ for z/OS- From set-up to Security
    - 2.75 days

- Ask your IBM rep for a schedule or to get one scheduled near you soon!

MQ & ACE/IIB Sessions (Room 101A unless stated)

Day	Monday	Tuesday	Wednesday	Thursday	Friday
8:30			24025: Lab: MQ for z/OS Hands on Labs Room 102B Lyn Elkins, Mark Taylor, & Mitch Johnson	24586: Better MQ z/OS Performance Through Understanding the Internals [z/OS] Lyn Elkins	24772: CICS and MQ - New and Vintage Room 101B Lyn Elkins
10:00	24590: Introduction to IBM MQ - Enterprise Messaging That Makes Your Life Easier [z/OS & Distributed] Neil Johnston	24585: An Introduction to and Comparison of the Different APIs Supported by MQ [z/OS & Distributed] Neil Johnston	24593: New Approaches to MQ High Availability and Disaster Recovery [Distributed] Room 101C Mark Taylor	24655: Gluing Together IBM MQ with WebSphere Liberty and WebSphere Application Server Traditional Using JMS 2.0 [Z/OS & Distributed] Room 103B Mitch Johnson	24591: MQ SMF: Bad Examples of Untuned, Unexamined Queue Managers – and Other Heartbreak [Z/OS] Lyn Elkins
11:15	24584: A technical introduction to App Connect Enterprise and IBM Integration Bus (on z/OS) [z/OS & Distributed] David Coles	24652: What's New in Logging in IBM MQ on the Distributed Platform Mark Taylor	24651: Secu terprise: Understandi rity through Scenarios a S & Distributed] Neil Johnst		24588: Getting the Best from Your MQ Shared Queues [z/OS] Neil Johnston
1:45 PM	24598: What's New in the Messaging Family - MQ v9.1 and More [z/OS & Distributed] Mark Taylor		24595: Run gration Bus on z/OS - W low [z/OS] David Coles	24596: Using Open Source Solutions to Monitor Your Queue Managers Room 103B Mark Taylor	
3:15 PM	24597: What's New in IBM App Connect Enterprise and IBM Integration Bus [z/OS & Distributed] David Coles	24587: Everything You Wanted to Know about SSL/TLS Principles but Were Unsure Who to Ask [z/OS & Distributed] Neil Johnston	24657: MQ & DB2 - MQ Verbs in DB2 & InfoSphere Data Replication (Q Replication) Performance [z/OS] Room 103B Lyn Elkins	24656: Hybrid Messaging & Integration - IBM App Connect, IBM App Connect Enterprise and IBM Event Streams David Coles	
4:30 PM	24654: Making MQ Application Development Easier [z/OS & Distributed] Mark Taylor	24589: Implementing RESTful Services Using IBM MQ and z/OS Connect or IBM Integration Bus and API Connect Mitch Johnson & David Coles	24594: New Bridges - Connecting MQ to the Rest of the World [z/OS & Distributed Room 103B Mark Taylor	24653: IBM MQ: Are z/OS & Distributed Platforms Like Oil & Water? [z/OS & Distributed] Room 101C Lyn Elkins & Mark Taylor	

You Are Here

Thank you!

SHARE
EDUCATE • NETWORK • INFLUENCE

- Remember this was session 24657