IBM Hybrid Cloud

MQ for HPE NonStop v8

Moving your business from MQ v5.3 to a solution based on a highly-available MQ v8

GTUG – Leipzig

May 2018

John Kinchen
David Ward
MQ Development
davidward@us.ibm.com

Important Disclaimer

IBM's statements regarding its plans, directions and intent are subject to change or withdrawal without notice at IBM's sole discretion. Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision. The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

- **Content Authority**. The workshops, sessions and materials have been prepared by IBM or the session speakers and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While efforts were made to verify the completeness and accuracy of the information contained in this presentation, it is provided AS-IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other materials. Nothing contained in this presentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.
- **Performance**. Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.
- **Availability**. References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates.

Trademark Statement

- IBM and the IBM logo are trademarks of International Business Machines Corporation, registered in many jurisdictions. Other marks may be trademarks or registered trademarks of their respective owners.
- Intel, and Itanium, are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
- Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
- Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.
- Other company, product and service names may be trademarks, registered marks or service marks of their respective owners.
- References in this publication to IBM products and services do not imply that IBM intends to make them available in all countries in which IBM operates.
- A current list of IBM trademarks is available on the web at "Copyright and trademark information" ibm.com/legal/copytrade.shtml"

IBM MQ for HPE NonStop Server Early/Beta program

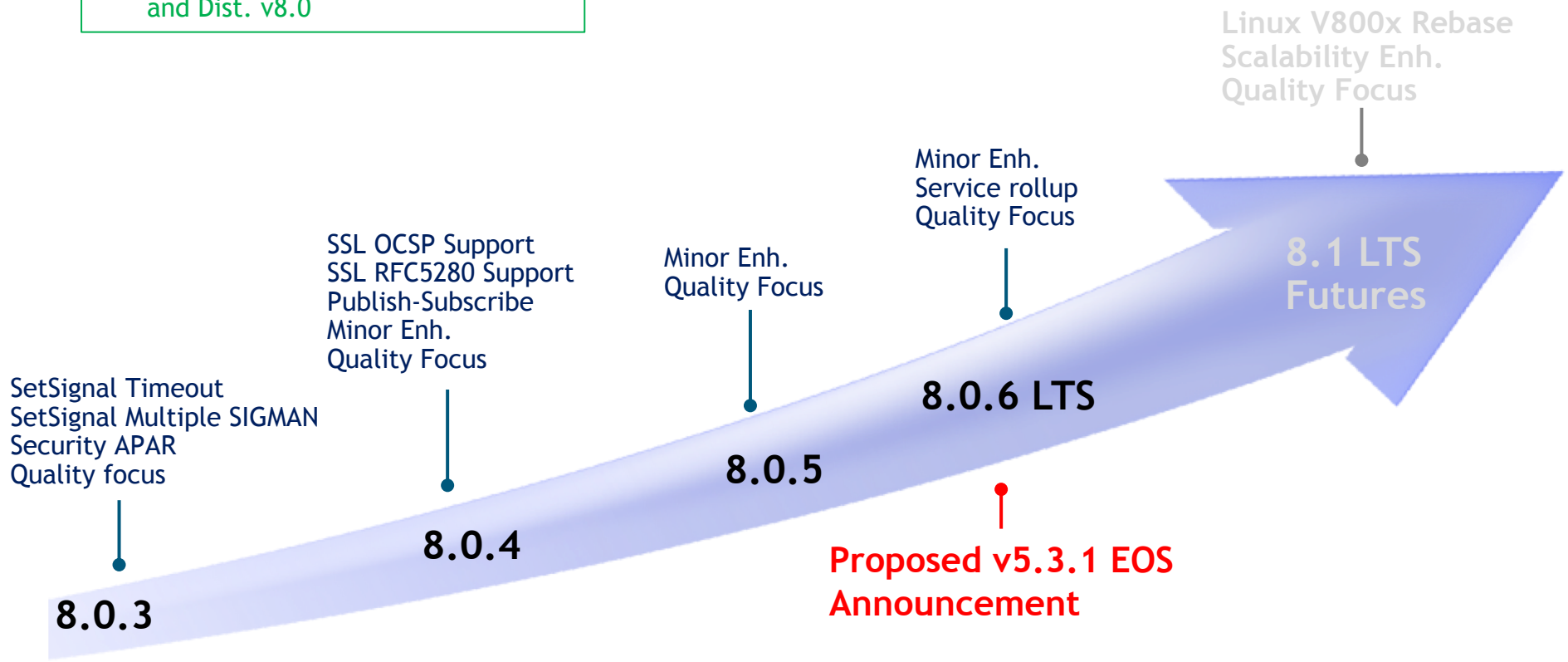
- **Beta program for next version of MQ on HPE NonStop Server:**
  - Opportunity to try your applications with the Beta code
  - Every MQ v8 CD release is available shortly after GA with Beta licenses, T&Cs
  - Update and discussion calls with the development team for Beta participants
  - Provide feedback to the product team
  - Support for any questions
  - Advance information to help with your planning
  - Invite to any beta program workshops/education events
- **Joining the Beta program:**
  - Nomination from either your local IBM contact or the Beta program manager
  - IBM asks you to accept standard Beta program terms and conditions (including confidentiality terms).
  - Any questions on the Beta program
    - Please ask the Beta program manager
    - Email : **pete_murphy@uk.ibm.com**

MQ for HPE NonStop Roadmap

V8.0.0 - Shipped Jun/17:
- Available on both NonStop x & I
- Continuous Delivery Model
- Production Ready
- Delivered majority of NonStop v5.3 and Dist. v8.0

V8.0.1 - Shipped in Jan/18:
- Partitionable Q Files
- ALTMQFLS/RUNNSCNF Enh.
- Migration Utility Enh.

V8.0.2 - Shipped in Mar/18:
- Security APARs
- Minor Usability Enh.
- Stability/Perf. improvements



MQ v8.0.3

3rd continuous delivery of MQ V8.0.0:

- Contains all function/fixes from previous CD releases

New Function

- **MQGET SET_SIGNAL IPC timeout**
  - Configurable IPC close timeout
- **Multiple MQGET SET_SIGNAL support**
  - Ability to configure SSMGR's by queue
- **SSL Enhancements**
  - SSL vulnerability fix CVE-2018-0739 (DoS)
  - SSL handshake rejected for disabled ciphers
- **Migration Utility Enhancements:**
  - Improved Cluster Handling
  - Support for V531 QM without OAM
  - Weak ciphers
  - Option to ignore SETMQAUTH errors
- **RUNNSCNF Utility Enhancements**
  - Support for protected variables
- **Installer Enhancements**
  - Usability improvements
  - Fixpack Installer

Stability

- **At least one customer in production**
- **Number of other customers are already running pre-production workload on MQ V8.0 in PoC environments**
- **Continuous & extended testing**
- **Stability & Performance fixes**
- **The following APARs are included in 8.0.3 <APAR List>**

Platforms

- **Supports NSX & NSI**
  - L-Series & J-Series (same RVUs)

Feeling adventurous!

- **Pub/Sub support**
- **MQ V8 has been tested against vNS**

Not officially supported…yet!

MQ v8.0.3: MQGET Multiple SSMGR processes

- **SSMGR manages IPC notifications to apps using MQGET with MQGMQ_SET_SIGNAL**
- **runnscnf extended to support the SetSignalManager class**
  - class SetSignalManager
    - PrimaryCPU
    - Priority
    - ProcessName
    - InactivityTimeout (1 min to many years)
    - CloseOnTimeout (true or false)
- **Each queue can be associated with a SetSignalManager object**
- **CloseOnTimeout can be used to simulate MQ 5.3 behavior**
  - Apps can be opened once and not closed
  - IPC Open does not happen until a signal needs to be delivered
    - (different from MQ 5.3 which opens the app immediately)

IBM MQ: CD vs LTS

Standard IBM definition:

Continuous Delivery

New CD versions of MQ are released approximately every 4 months, incrementally introducing new product capabilities.

Intended for those that can actually integrate new function

Long-term Support

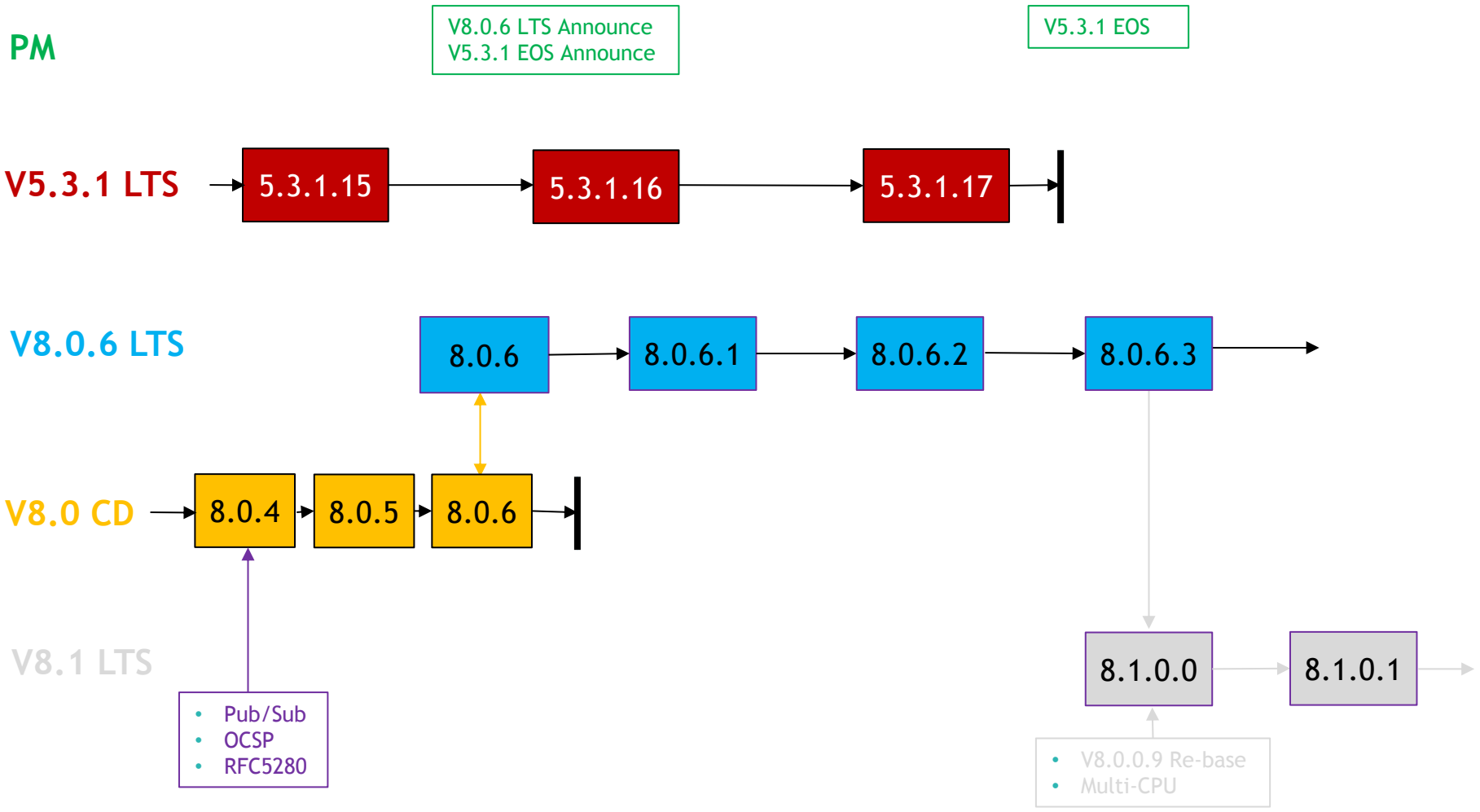
Approximately every 2 years a new LTS version is released, rolling up many of the CD capabilities into a release with 5+3 support attached.

Required for those looking for fixed function

HPE NonStop perspective:

- All CD releases from v800 onwards are production-ready and PMRs are accepted against any release
- Product is still maturing and IBM recommends customers move to the latest release.
- Fixes will be provided for CD and CD-1 releases by default
  - *BUT customers can request an ifix for an earlier CD release* (LTS equiv.: back-level APAR fix)
- Each CD release WILL contain cumulative service fixes (LTS equiv.: fixpacks)
- Each CD release MAY contain new function
- LTS version be available when V8.0 function is considered *fixed* with 5+3 support attached.

MQ for HPE NonStop Release & Support Timeline



Working with MQ V8.0 in a Multi-CPU Environment - Today

- There is no Q Server concept in V8.0 so a single QM cannot share load across multiple CPUs
- Load-balancing in V8.0:
  - Customer apps & configs would need to change
  - Faster HPE machines may reduce/remove need
  - It is not seen as a priority to most customers that IBM has talked to already

QMs can run in any CPU

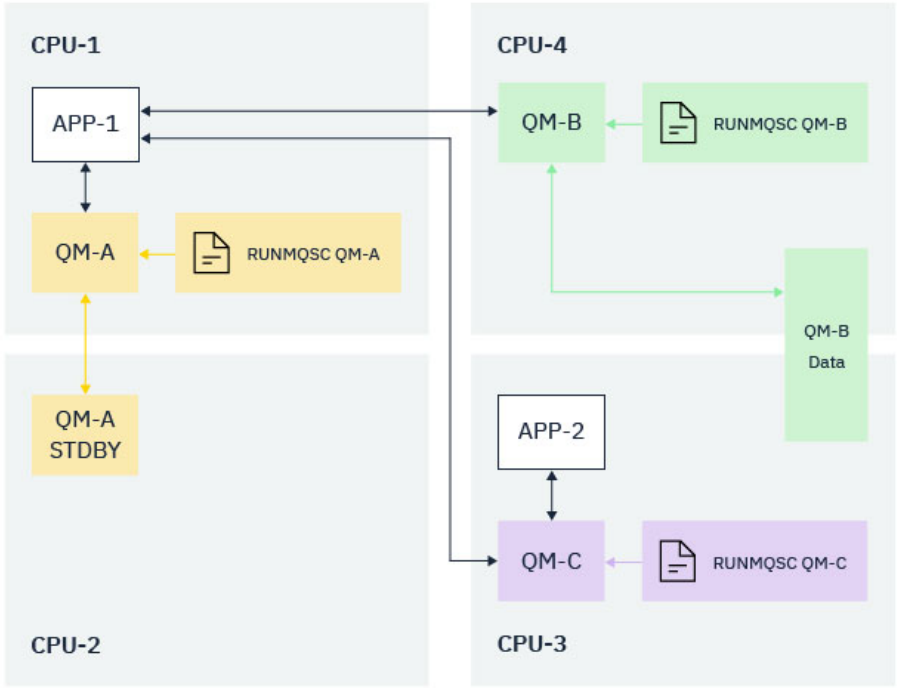
Standby QMs can run in any CPU

MQ Tooling can be run any CPU

Applications can run in any CPU

Q data can be split across multiple CPUs

No additional MQ licensing costs!!



Scalability and Multi-CPU

Today:

- MQ v8 QM runs in a single NonStop CPU
- Multiple QMs can be started in all CPUs
- MQ Tooling can run any CPU
- Application programs can run in any CPU

- *BUT: Unlike MQ 5.3, a single QM workload cannot be distributed across multiple CPUs*

Future:

- Feasibility study under way to consider ALL scalability options for MQ on HPE NonStop
- Some possible solutions involve application and/or configuration changes.
- Any solution is unlikely before 2H19

How you can help?:

- Some customers are less concerned with scalability across multiple CPUs but are more concerned with simple performance.
- Other customers have the opposite perspective

- *Extending the current MQ v8 product to support more scalability is a large-scale effort*
- *If this functionality impacts your business, please reach out to IBM now!*

Other Scalability Enhancements

- Multiplexed client performance:
  - Multiple Client conversations (e.g. threads) can use the same TCP/IP socket (channel instance):
  - Enabled via new SHARECNV(n) Channel attribute
  - Heartbeating, Read-Ahead & Asynchronous Consume support
  - SHARECNV (n > 1) reduces number of sockets & channel instances needed
- Multiple cluster transmission queues:
  - Separation of message traffic
  - Management of messages
  - System monitoring

Coming later this year

- Routed Pub/Sub in Clusters

Non-Persistent Messages (NPM) & Fault Tolerance

MQ v5.3.1:

- Stores NPM in the Queue Server which is a HPE NonStop process pair
- Primary QS & Backup QS are running in different CPUs
- NPMs will be "*checkpointed*" to the Backup QS by default
- Checkpointing can be disabled to improve performance, but NPMs will then be lost

Provides resilience to NPM loss out of the box

MQ v8.0:

- Based on the IBM V8.0 Linux product design & architecture
- NPM trades away most of MQ's assurances re: message loss vs higher performance

No resilience to NPM loss in the event of CPU failure

IBM is currently investigating NPM options going forward

MQ Security

MQ Security: Channel CIPHERs

SSL 3 and TLS 1.0 Cipherspecs

DES_SHA_EXPORT

DES_SHA_EXPORT1024

NULL_MD5

NULL_SHA

RC2_MD5_EXPORT

RC4_56_SHA_EXPORT1024

RC4_MD5_EXPORT

RC4_MD5_US

RC4_SHA_US

TRIPLE_DES_SHA_US

TLS_RSA_WITH_DES_CBC_SHA

TLS_RSA_WITH_AES_128_CBC_SHA

TLS_RSA_WITH_AES_256_CBC_SHA

TLS_RSA_WITH_3DES_EDE_CBC_SHA

TLS 1.2 Cipherspecs

TLS_RSA_WITH_NULL_SHA256

TLS_RSA_WITH_AES_128_CBC_SHA256

TLS_RSA_WITH_AES_128_GCM_SHA256

TLS_RSA_WITH_AES_256_CBC_SHA256

TLS_RSA_WITH_AES_256_GCM_SHA384

ECDHE_RSA_AES_256_GCM_SHA384

ECDHE_RSA_AES_128_CBC_SHA256

ECDHE_RSA_AES_128_GCM_SHA256

ECDHE_RSA_AES_256_CBC_SHA384

ECDHE_ECDSA_AES_128_CBC_SHA256

ECDHE_ECDSA_AES_256_CBC_SHA384

ECDHE_ECDSA_AES_128_GCM_SHA256

ECDHE_ECDSA_AES_256_GCM_SHA384

MQ Security – SSL Multiple Certificates

- **Configurable default certificate label for qmgrs and clients**
  - Instead of *cert.pem*
  - `ALTER QMGR CERTLABL('mycertificatename')`
  - Which uses *mycertificatename.pem* instead

- **Channel-level certificates**
  - To support different business partners using different certificate authorities.
  - For queue managers and C clients

  - `ALTER CHANNEL … CERTLABL('Thischannelcertificate')`

- **Both ends must be V8 or later**
- **TLS 1.2 only**

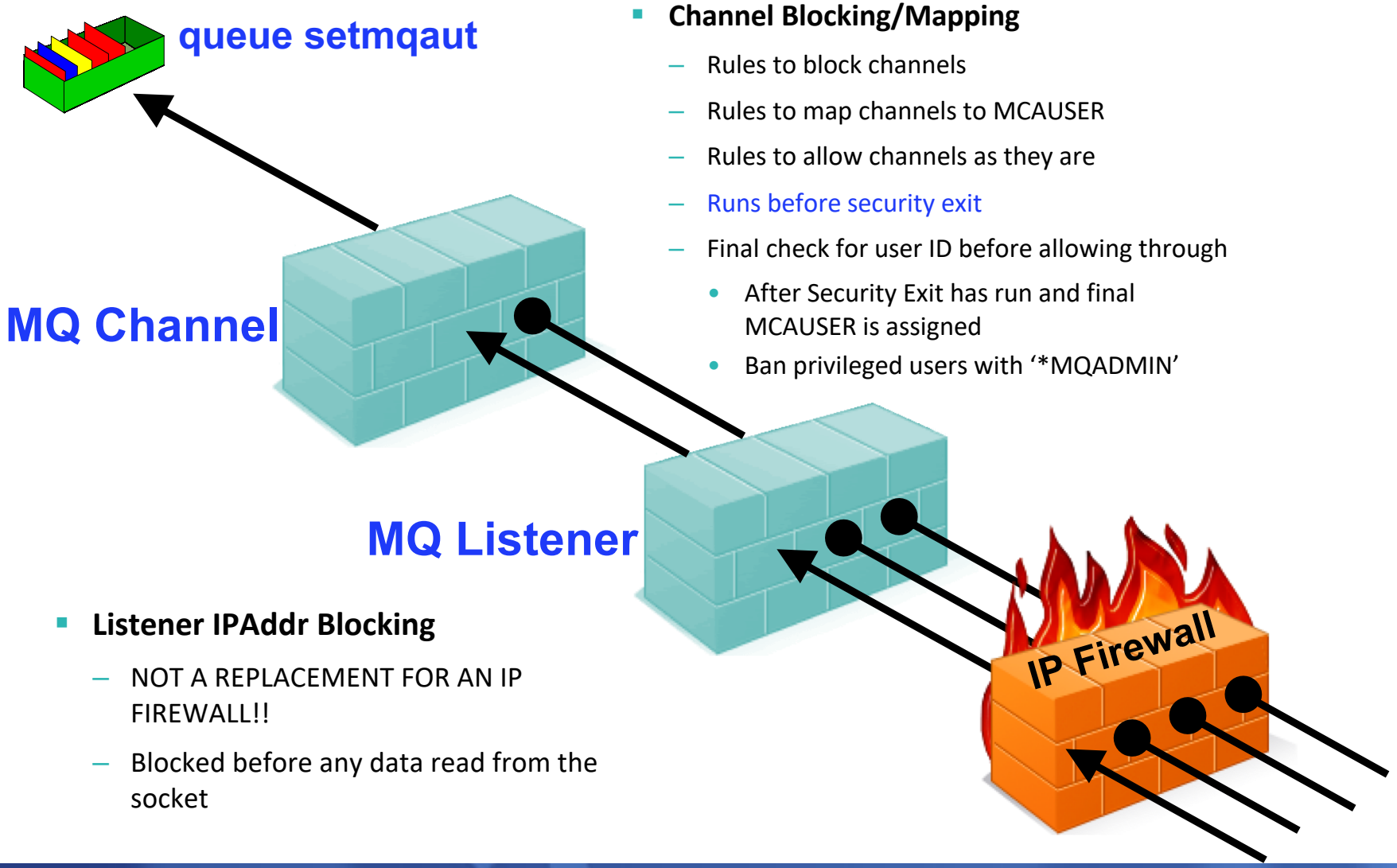
MQ Security: SSL Handshake Reject (8.0.3 and later)

- **Inbound SSL channel connections**
- **Previously,**
  - SSL handshake would complete then protocol and cipher would accepted or rejected
- **Now,**
  - SSL handshake only completed for enabled protocols and ciphers
- **SSL3 protocol and ciphers disabled by default**
- **TLS 1.0 protocol and ciphers can be disabled**
- **Use TLS 1.2 !**

MQ Security - CHLAUTH

- **Set rules (via MQSC, PCF or MQ Explorer) to permit/deny inbound connections**
  - Inbound clients
  - Inbound message channels
- **CHLAUTH Rules can**
  - Allow a connection
  - Allow a connection and assign an MCAUSER
  - Block a connection
  - Block privileged access
  - Control SSL Peer Name matching
- **CHLAUTH Rules can use any of the following identifying data**
  - IP address
  - SSL/TLS DN
  - Client userid
  - Remote queue manager name

MQ Security: CHLAUTH Access Blocking Points



- **Channel Blocking/Mapping**
  - Rules to block channels
  - Rules to map channels to MCAUSER
  - Rules to allow channels as they are
  - Runs before security exit
  - Final check for user ID before allowing through
    - After Security Exit has run and final MCAUSER is assigned
    - Ban privileged users with '*MQADMIN'

- **Listener IPAddr Blocking**
  - NOT A REPLACEMENT FOR AN IP FIREWALL!!
  - Blocked before any data read from the socket

MQ Security - CONNAUTH

- **Allows an MQ administrator to require applications to provide a user ID and password when connecting.**
- **Both local and client connections are supported.**

- **ALTER QMGR CONNAUTH(DAVES.CONNAUTH.USE.PW)**
- **DEFINE AUTHINFO(DAVES.CONNAUTH.USE.PW)**
- **AUTHTYPE(IDPWOS)**
- **CHCKCLNT(REQUIRED)**
- **CHCKLOCL(OPTIONAL)**
- **REFRESH SECURITY TYPE(CONNAUTH)**

- ***Program example: see amqsput0.c sample source***

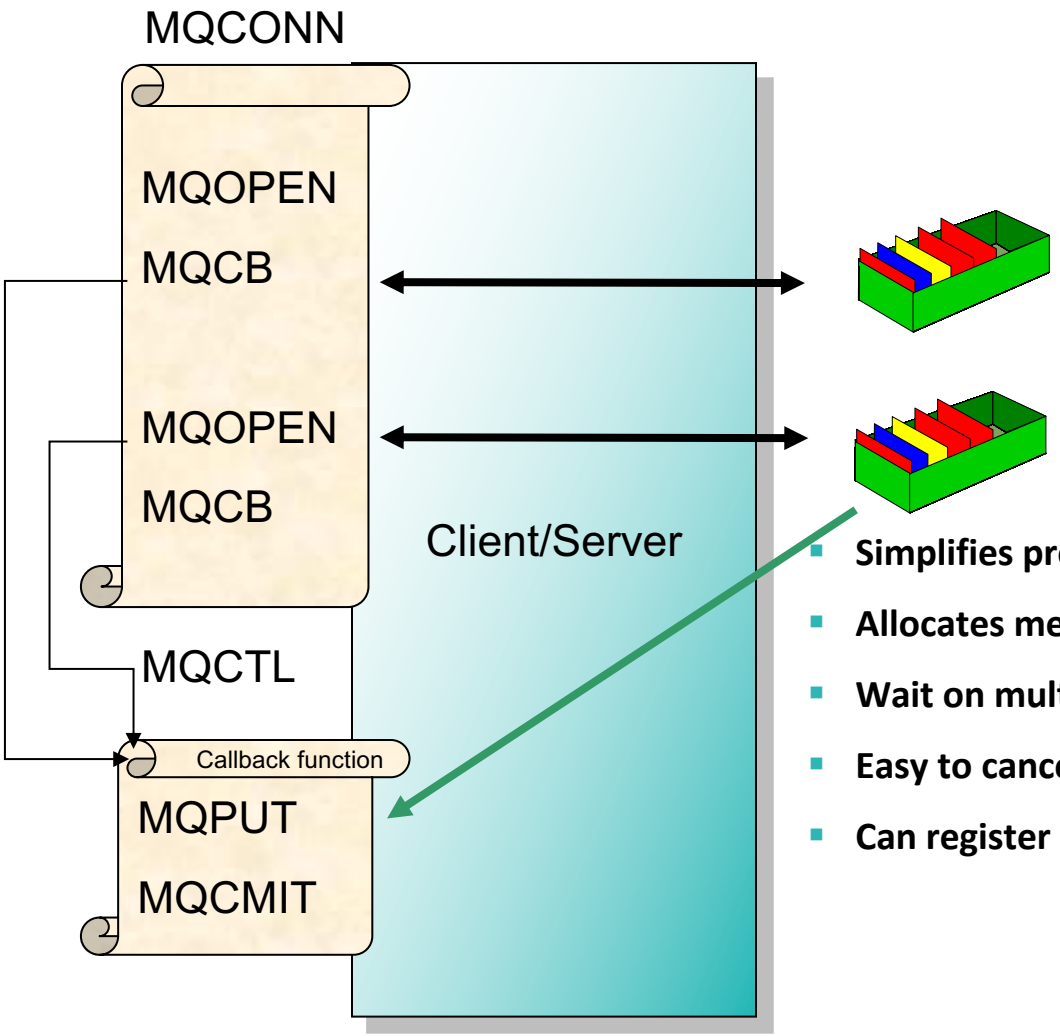
MQI Asynchronous Consume

- **A message-driven function or routine called by the Queue Manager when a message is ready to be delivered**
- **No MQGET is needed and no buffer need be provided by the application**
- **Fewer resources allocated waiting for a message to arrive**
- **Enabled using the MQCB and MQCTL calls**

MQI Asynchronous Consume

- ***Asynchronous consume allows the application to register an interest in messages of a certain type and identify a call-back routine that should be invoked when a message arrives.***
- **Simplifies programming**
  - The application can continue to do whatever else it needs to without needing to tie up a thread sitting in a MQGET call.
- **Manages message buffers**
  - The application does not need to 'guess' the size of the next message and provide a buffer just large enough. The system will pass the application a message already in a buffer.
- **Wait on multiple queues**
  - The application can register an interest in any number of queues. This is very much simpler than using MQGET where one generally ends up polling each queue in turn.
- **Easy to cancel**
  - The application can use either MQCTL or MQCB to stop consuming from a queue at any time. This is awkward to achieve when an application is using MQGET.
- **Can register an event handler**
  - The application is notified of events such as when the queue manager is quiescing or when a communications failure occurs.

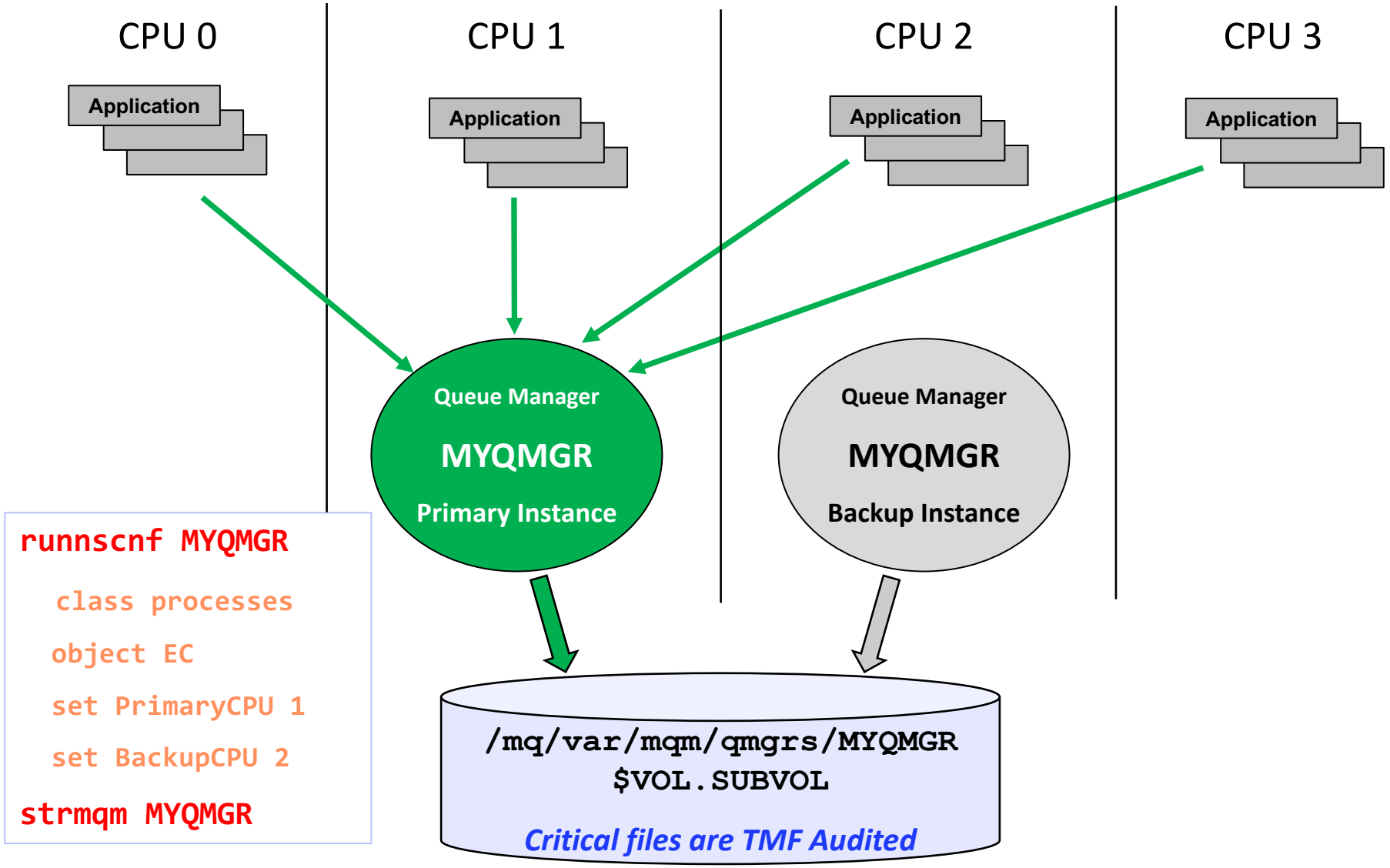
MQI Asynchronous Consume



- **Simplifies programming**
- **Allocates message buffers**
- **Wait on multiple queues**
- **Easy to cancel**
- **Can register an event handler**

Resilience

High-availability NonStop Queue Manager (HANSQM)




```
runnscnf MYQMGR
   class processes
   object EC
   set PrimaryCPU 1
   set BackupCPU 2
strmqm MYQMGR
```

High-availability NonStop Queue Manager (HANSQM)

- **Configured using runnscnf**
- **No special strmqm options**
- **Primary and Backup CPUs can "float" or be fixed**
- **Relevant runnscnf attributes**
  - PrimaryCPU and BackupCPU — CPU of Primary and Backup instance
  - AllowedCPUs — Comma separated list of CPUs
  - ProcessName — EC process name
- `dspmq –x`

```
QMNAME(DAVE)                              STATUS(Running)
PROCESS($Z1CW) PRIMARY(2,1122) INSTANCE(1)
PROCESS($Z1CW) BACKUP(0,463) INSTANCE(2)
```

- **Setting PrimaryCPU = BackupCPU causes the queue manager to run without a backup instance.**

Resilience: Non-persistent Messages

- **The current MQ v8 NonStop product does not harden NPM's**
- **Presently, the QOS for NPM's is the same MQ on Linux**
  - NPM's may/will be lost if the OS or hardware fails or the queue manager fails.
- **Likely: MQ v8.0.4 will add support for hardening non-persistent messages.**

MQ Disaster Recovery: Thoughts and Considerations

MQ DR: Hard and Soft Requirements

- **Hard requirement: After system outage application functionality must be restored**
- **Hard requirement: Local transactional consistency**
- **Soft requirement: No impact on application design**
- **Soft requirement: Low impact on resource usage**
- **Soft requirement: Low impact on transaction duration**

MQ DR: The MQ value proposition

Assured, one-time message delivery

MQ DR: Thoughts for Distributed Systems

- **MQ per se is a distributed transactional system – across platform boundaries**
- **Local transactional consistency is not**
- **Consequences for DR: Asynchronous DR solutions losing "on the fly" data can cause message loss or duplication**
- **When you plan your DR solution, consider consequences for distributed transactions!**
- **The only "perfect" solution is a synchronous solution – at a cost**

MQ DR: v8 Considerations

- **All MQ application data objects (queues and metadata) audited**
- **With MQ 5 per default this was not the case!**
- **RDF and Shadowbase will work**
- **Solution must replicate creation and deletion of files**
- **There are still some OSS files that need to synched by other methods**

MQ DR: v8 Procedures

- **Do NOT create queue managers on both systems!**
- **Replication and synch mechanisms will create/delete queue managers**
- **Creation and deletion of queue managers will need OSS synch**
- **Proposed future plans include**
  - Some key OSS files will be relocated to Guardian Enscribe under TMF audit control
  - Goal is to make designing DR solutions for MQ easier
- **After DR takeover channel resets likely required (even with synch replication)**

MQ Knowledge Center

- **MQ v8 downloadable PDF documents**

ftp://public.dhe.ibm.com/software/integration/wmq/docs/V8.0/PDFs/

- **MQ v8 NonStop product readme**

http://www.ibm.com/support/docview.wss?uid=swg27049906

- **MQ v8 NonStop documentation**

https://www.ibm.com/support/knowledgecenter/SSFKSJ_8.0.0/com.ibm.mq.hpnss.doc/intro.htm

Thank you

- **Questions ?**

- **davidward@us.ibm.com**