

WMQ Clustering

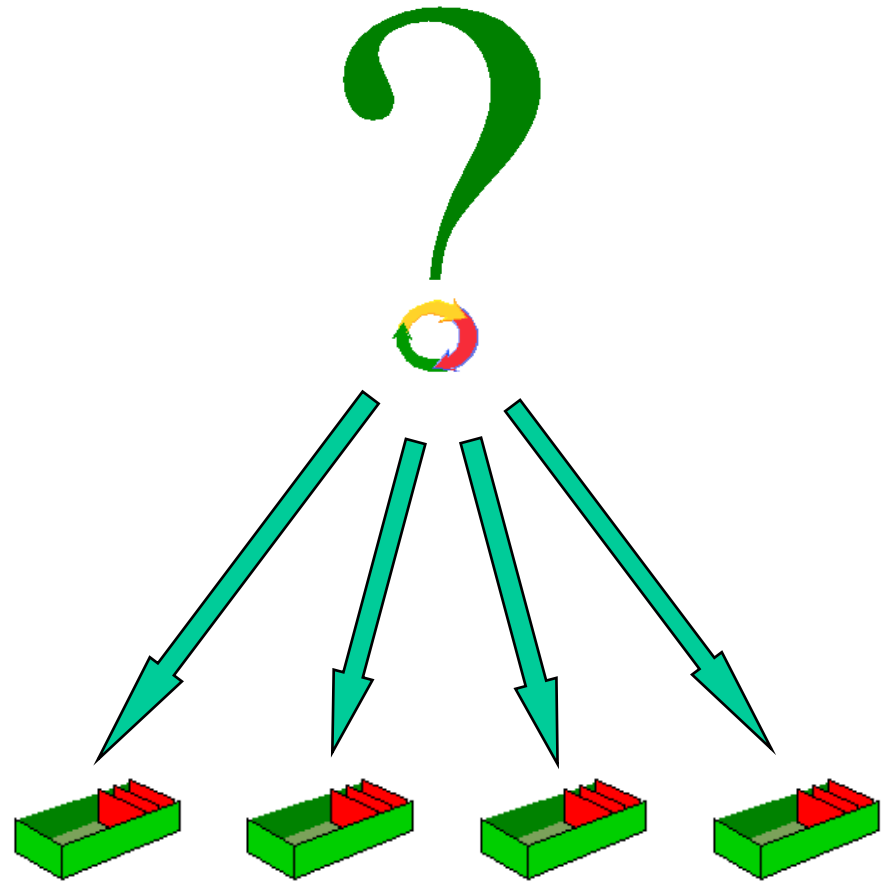
Judy Zhang (张娟)

juanz@cn.ibm.com

IBM CDL MQ L2 support , July 2011

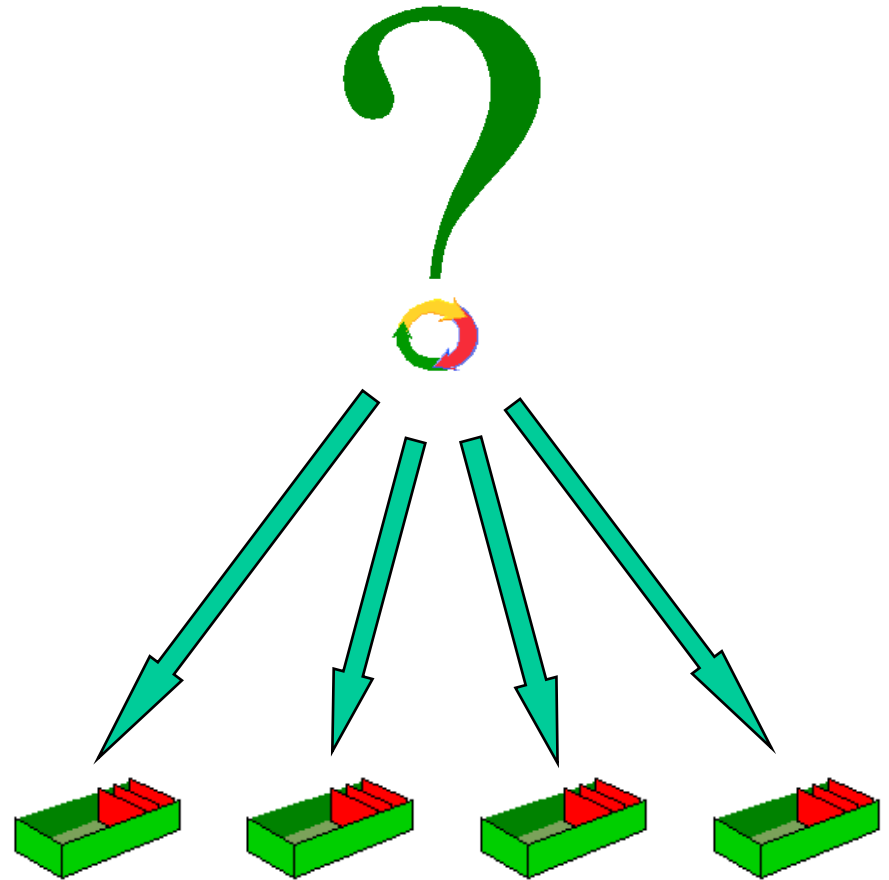
Agenda

- Clustering overview
- How cluster works
- Cluster configuration
- Workload balancing
- Best practice
- Clustering and sharing (QSG z/OS only)



Agenda

- Clustering overview
- How cluster works
- Cluster configuration
- Workload balancing
- Best practice
- Clustering and sharing (QSG z/OS only)



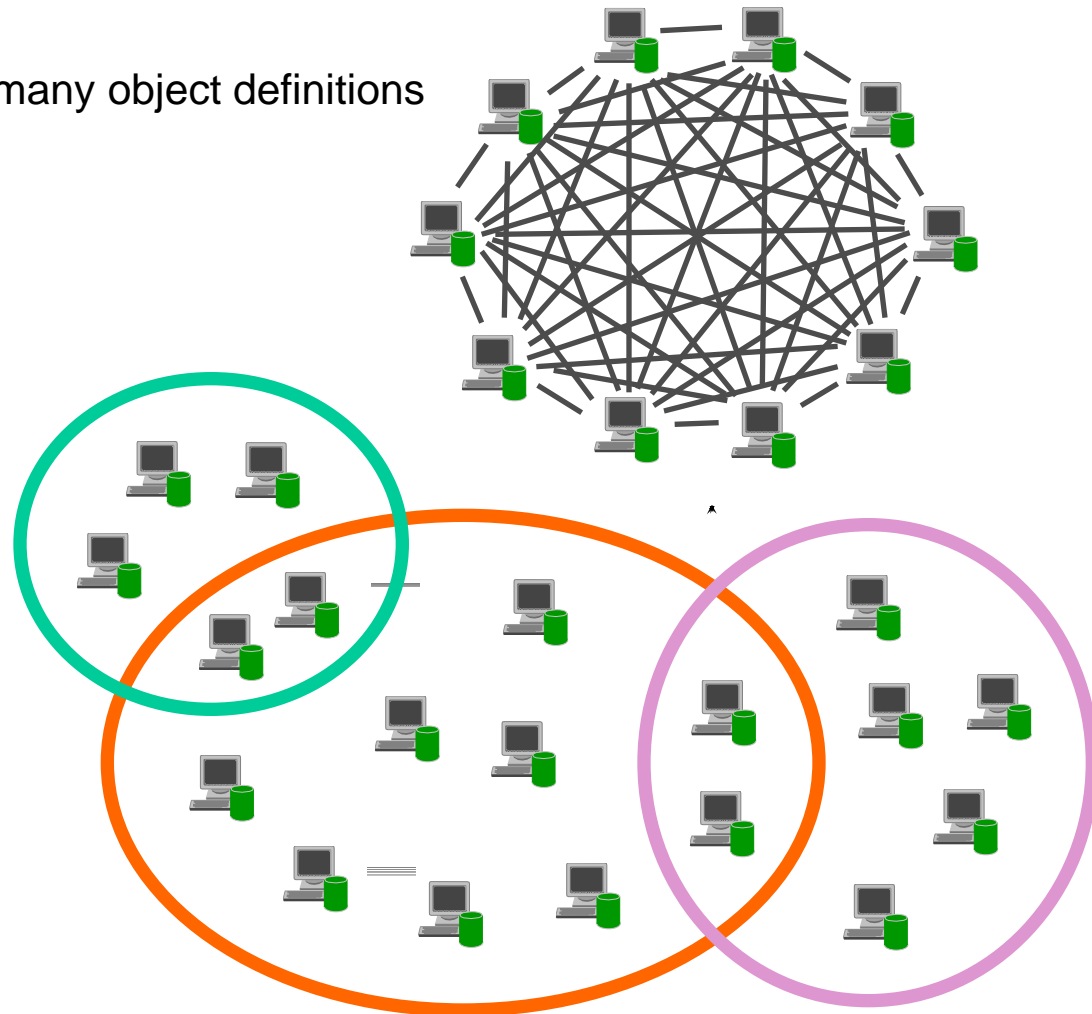
What is Clustering ?

Clustering is a way to logically group WebSphere MQ queue managers so that you have:

- reduced system administration
- increased availability
- workload balancing
- **Repository**
- **Repository queue manager**
- **Full repository and partial repository**
- **Cluster-receiver channel/Cluster-sender channel**
- **Cluster transmission queue**

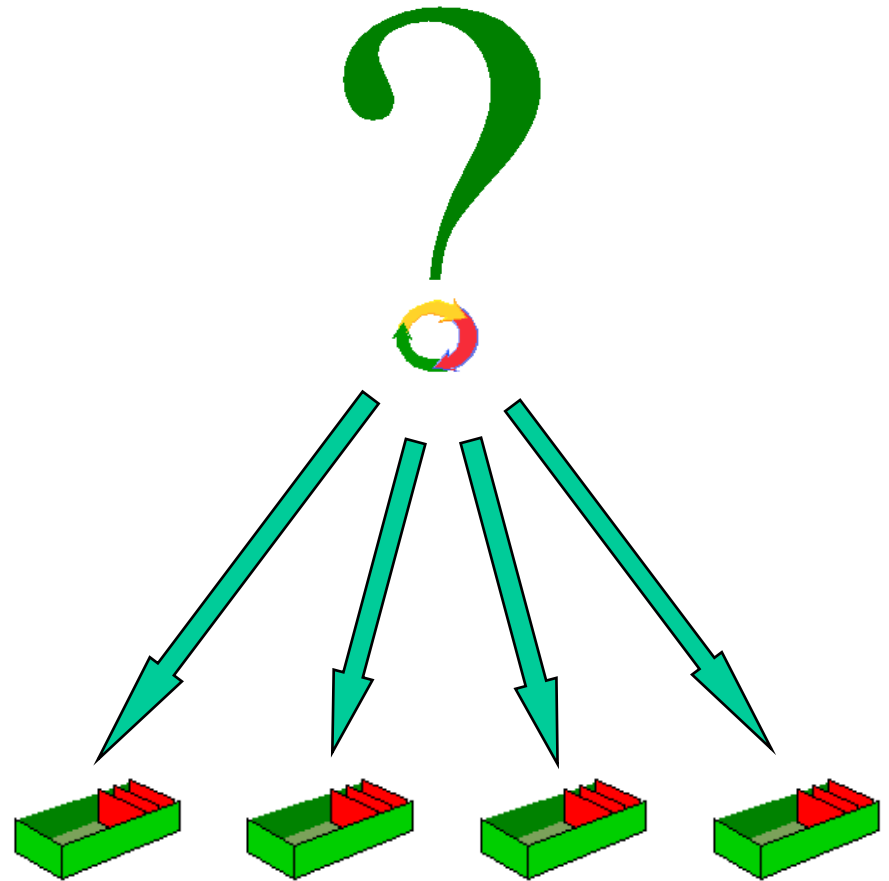
How benefit from Clustering

- Simplified administration
 - ▶ Large WMQ networks require many object definitions
 - Channels
 - Transmit queues
 - Remote queues
- Workload balancing
 - ▶ Spread the load
 - ▶ Route around failures
- Flexible connectivity
 - ▶ Overlapping clusters
 - ▶ Gateway Queue managers

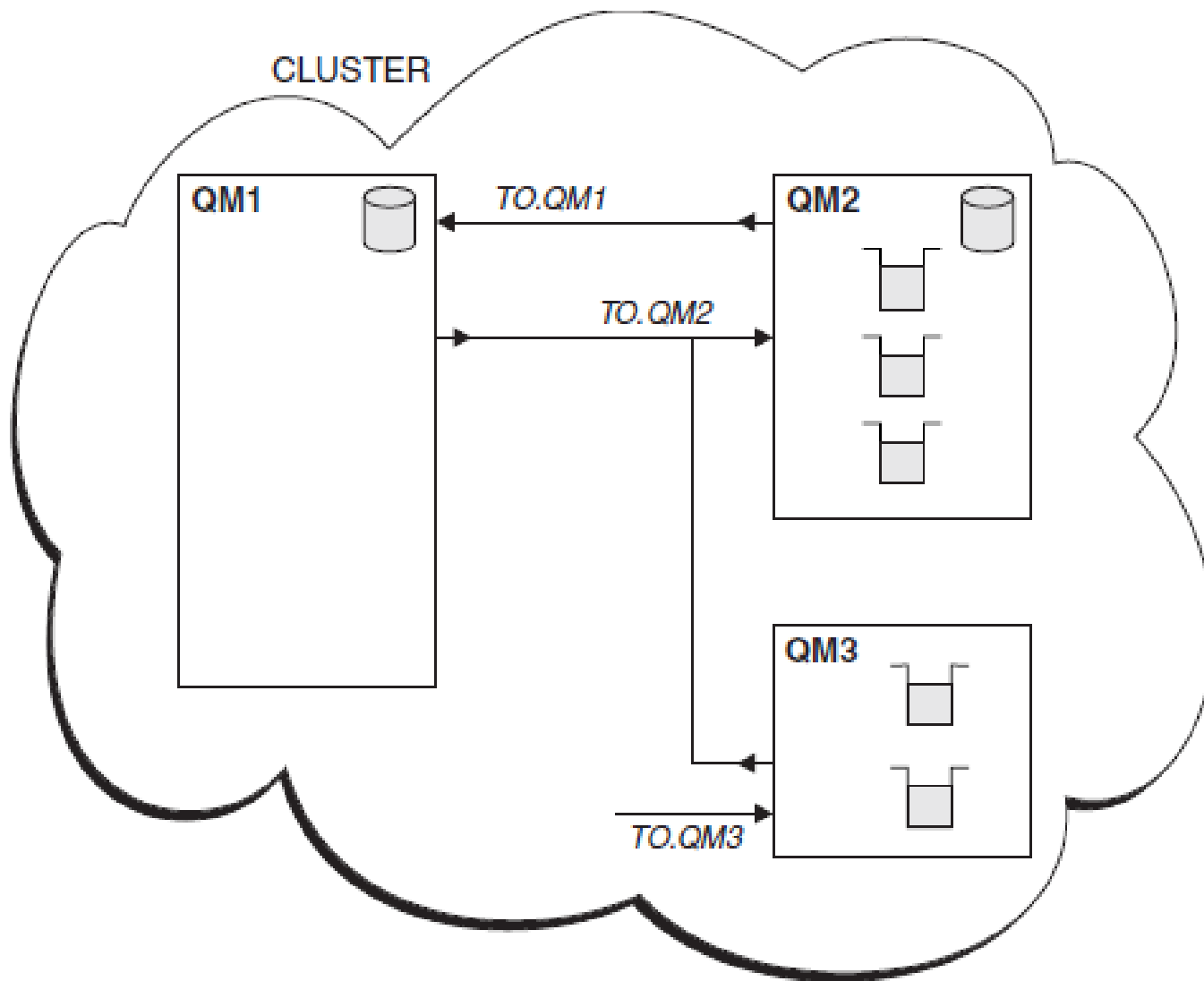


Agenda

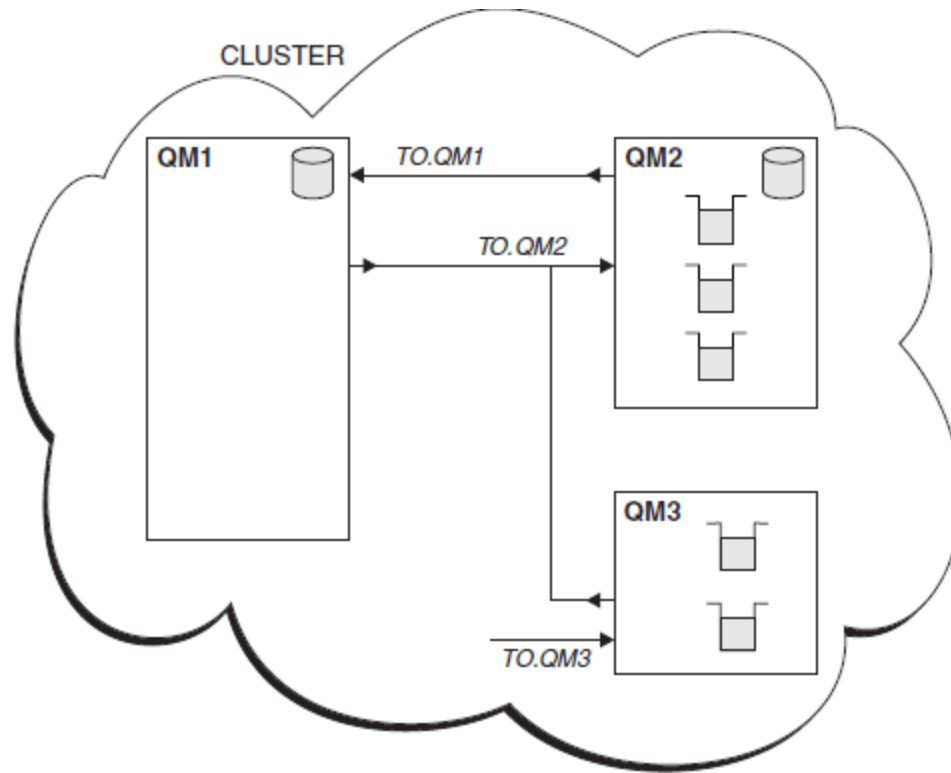
- Clustering overview
- How cluster works
- Cluster configuration
- Workload balancing
- Best practice
- Clustering and sharing (QSG z/OS only)



How Cluster looks like

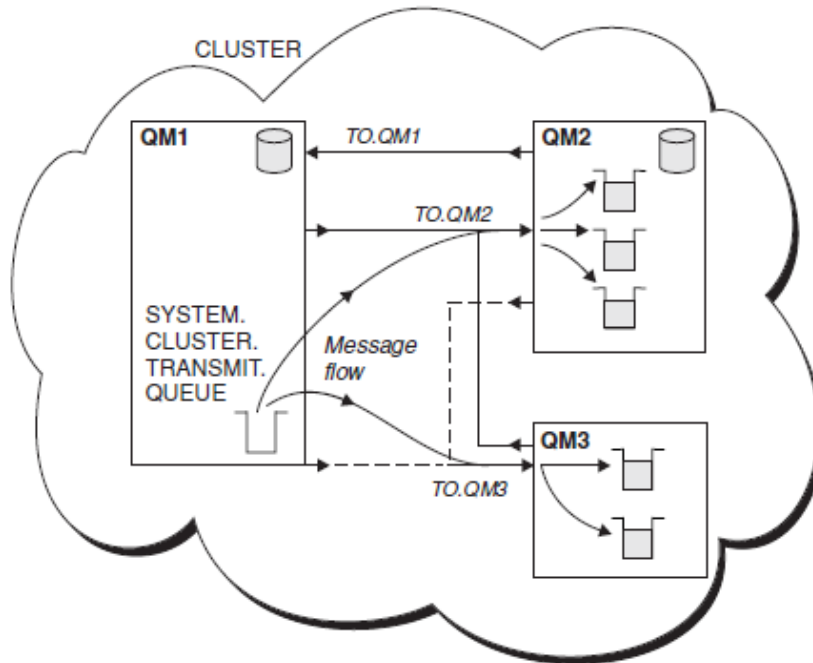


What make Cluster work



- Queue managers and repositories
- Queues
 - ▶ `DEFINE QLOCAL(Q1) CLUSTER(SALES)`
- Cluster transmission queue
- Cluster channels
- Auto-definition of remote queues
- Auto-definition of channels
- System queues:
 - ▶ `SYSTEM.CLUSTER.REPOSITORY.QUEUE`
 - ▶ `SYSTEM.CLUSTER.COMMAND.QUEUE`
 - ▶ `SYSTEM.CLUSTER.TRANSMIT.QUEUE`

How clustering work



- Defining a cluster-sender channel has the effect of introducing a queue manager to one of the full repository queue managers.
- The full repository queue manager updates the information in its full repository accordingly
- Then it automatically creates a cluster-sender channel back to the queue manager.
- and sends the queue manager information about the cluster.
- Thus a queue manager learns about a cluster and a cluster learns about a queue manager

How clustering work (cont.)

- In the figure on the prior page, QM1 and QM2 are full repositories.
- The solid lines represent manually defined CLUSRCVR and CLUSSDR channels.
- The two dashed lines represent cluster-sender channels that have been created automatically.
- QM2 and QM3 host some clustered queues that can be put to by any other queue manager in the cluster.
- All queue managers have a `SYSTEM.CLUSTER.TRANSMIT.QUEUE` from which they can send messages to any other queue manager in the same cluster.

Cluster Architecture Overview - Queues

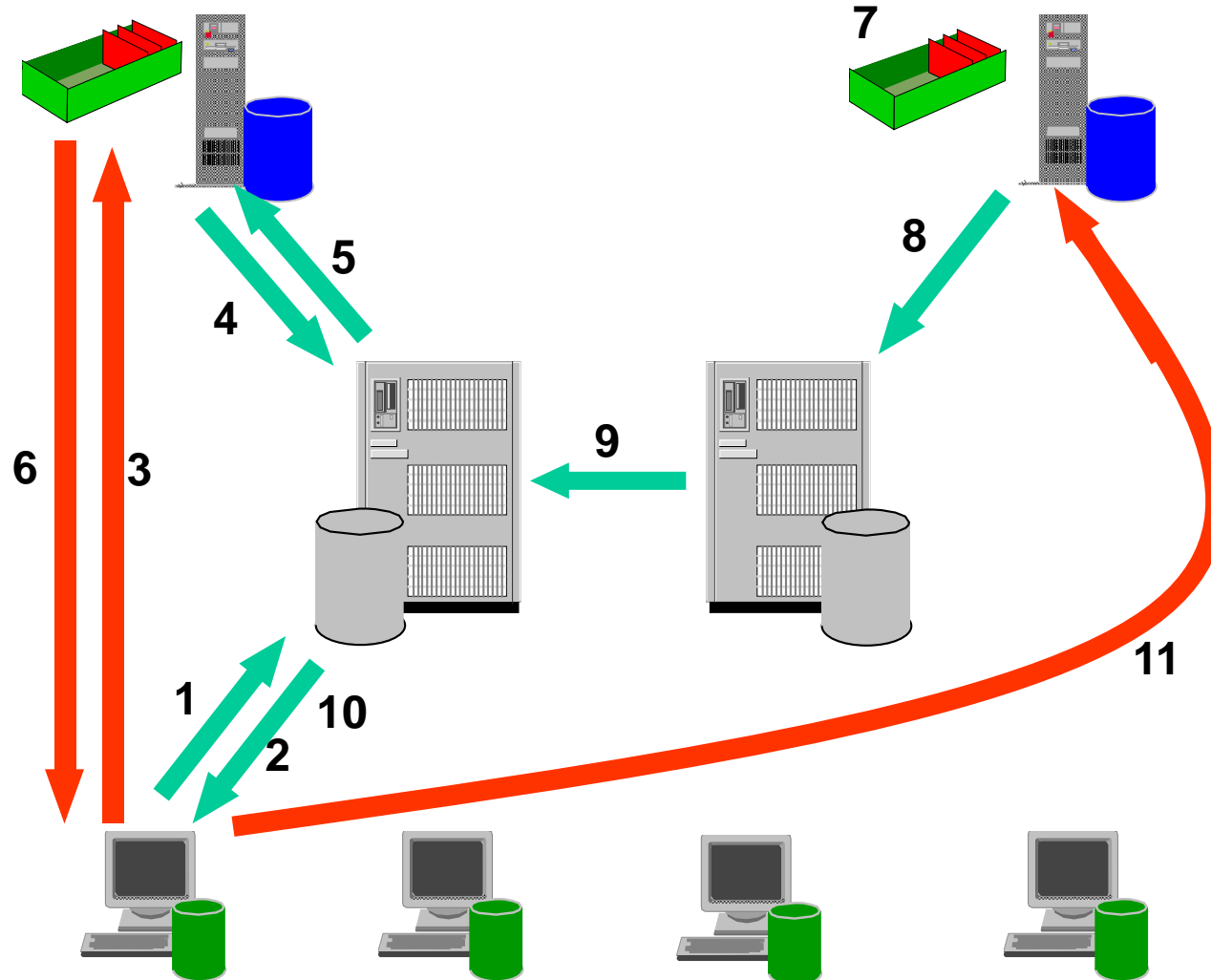
Partial Repository QMs

+ Reply applications

Full Repository QMs

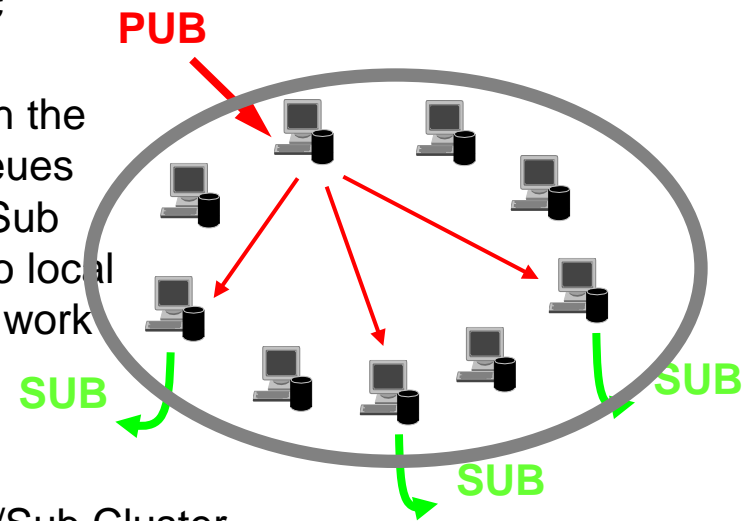
Partial Repository QMs

+ Request applications

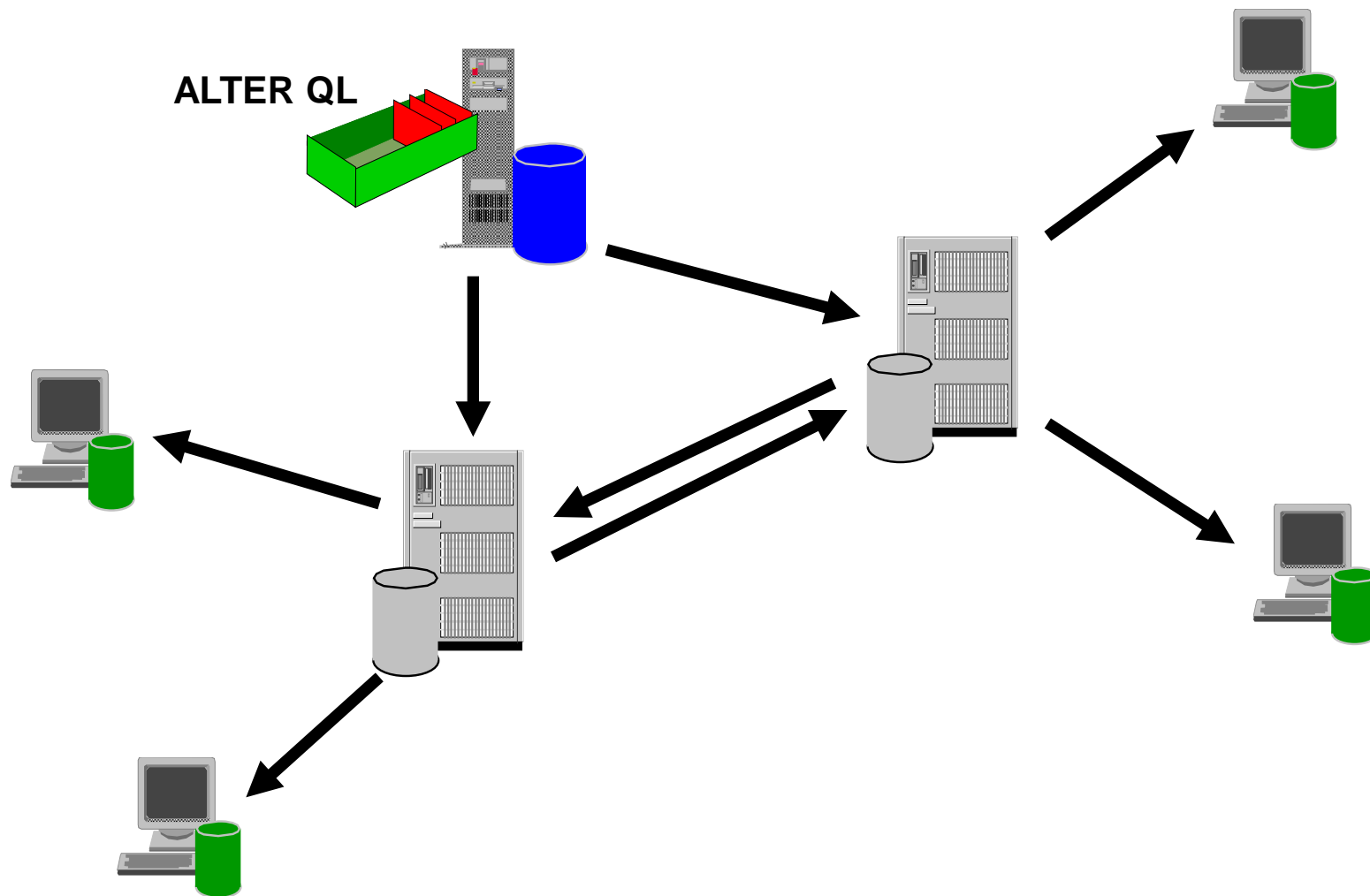


Cluster Architecture Overview - Pub/Sub(New in V7)

- Builds on traditional 'point to point' clustering's Simplified administration
 - ▶ Non-disruptive add/remove of qmgrs
- Point of administration becomes 'Topic' rather than 'Queue'
 - ▶ Information about topics passed through the cluster in same way we just saw for queues
 - ▶ Required for publishing to remote Pub/Sub Cluster queue managers (as opposed to local pub/sub where you can sometimes just work with Topic Strings)
- Cluster queues
 - ▶ Can be used in traditional way in a Pub/Sub Cluster

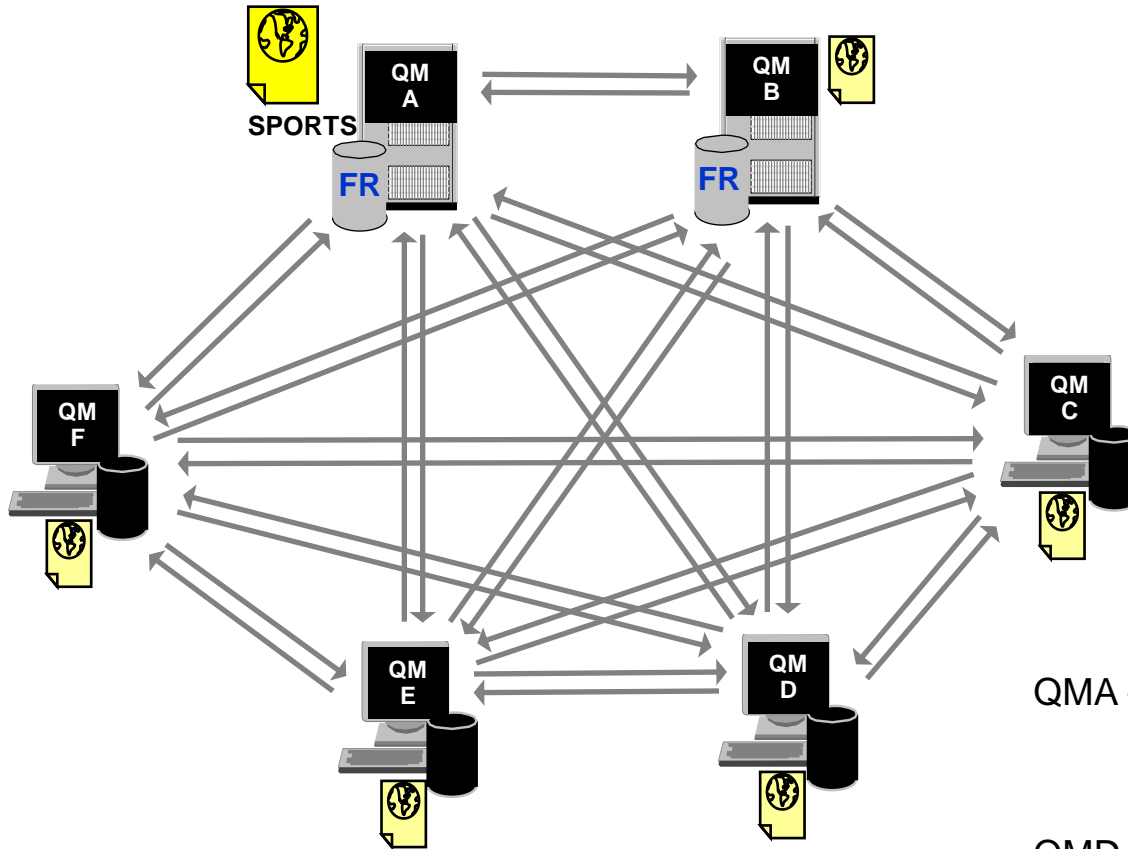


Change Propagation - Queues



Change Propagation - Topics

QMA - **DEFINE TOPIC(SPORTS) TOPICSTR(/global/sports) CLUSTER(DEMO)**



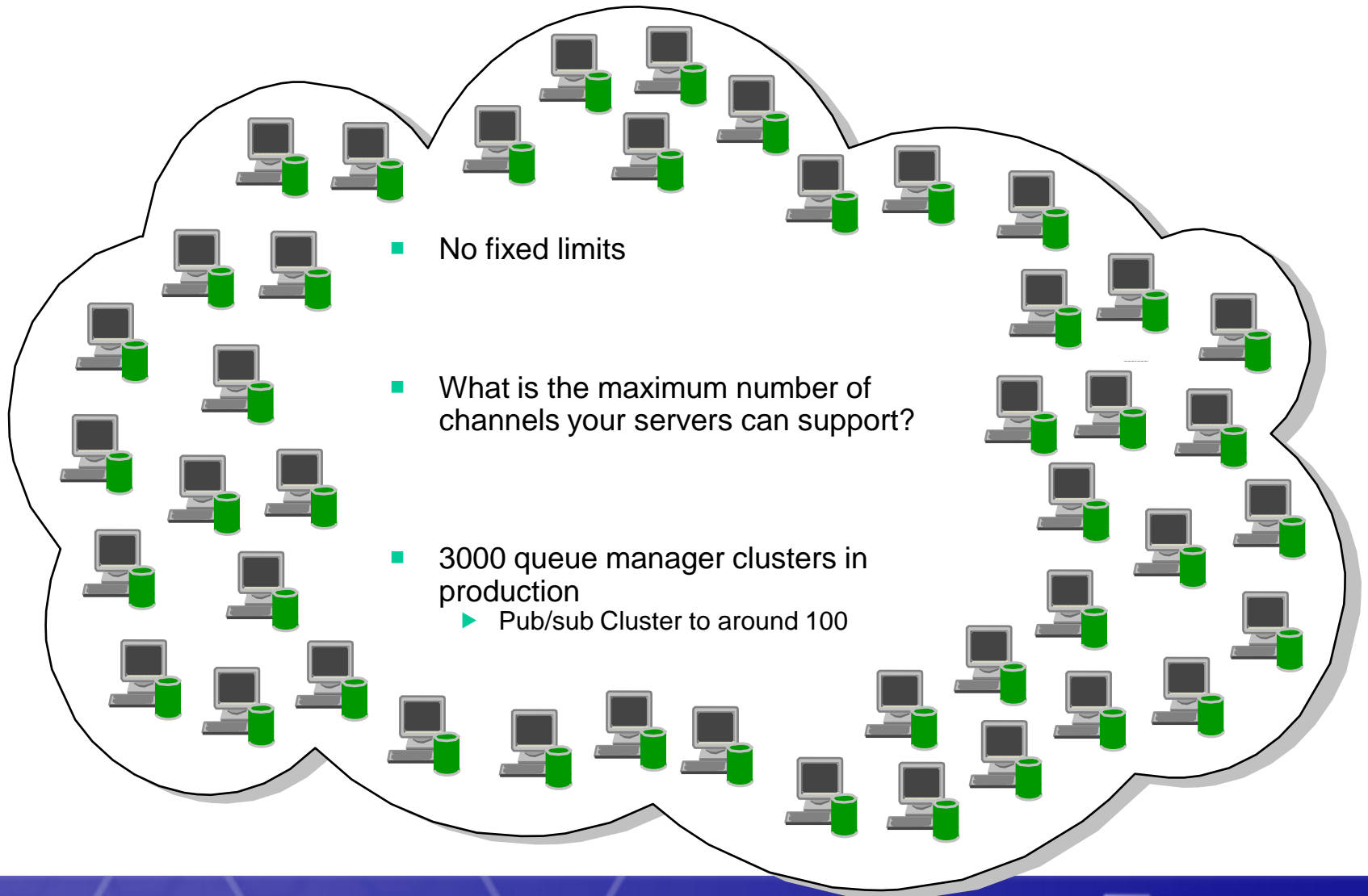
QMA - **DIS TOPIC(SPORTS) TYPE(ALL)**

- Local topic
- Cluster topic

QMD - **DIS TOPIC(SPORTS) TYPE(ALL)**

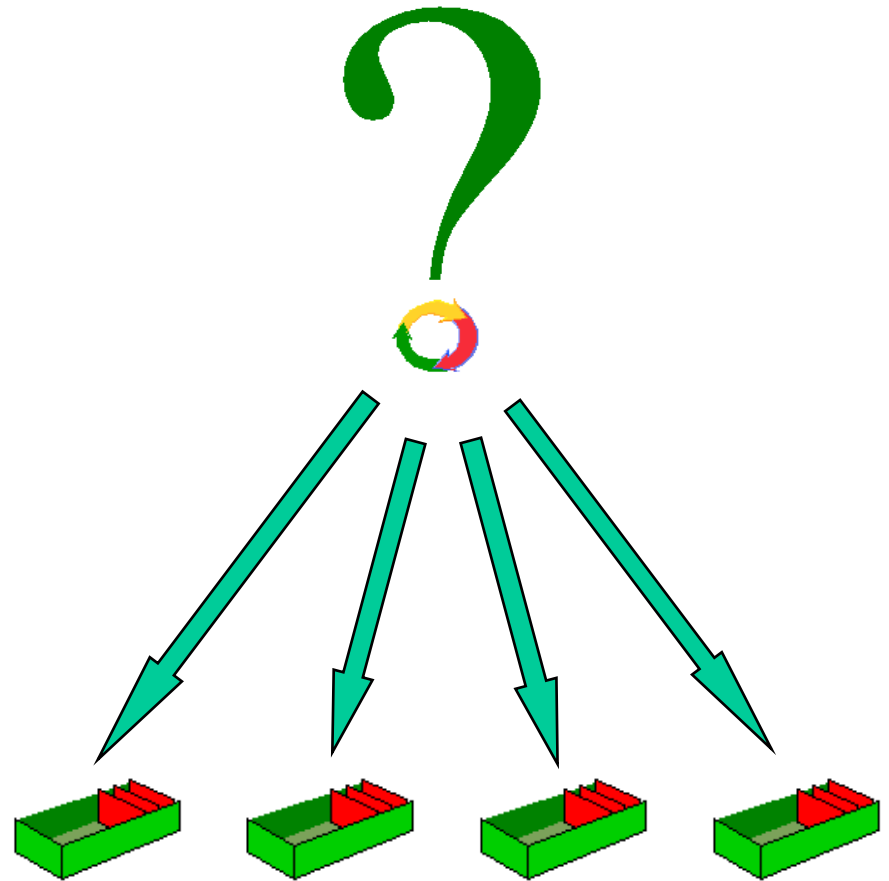
- Cluster topic

How big can a cluster be?



Agenda

- Clustering overview
- How cluster works
- Cluster configuration
- Workload balancing
- Best practice
- Clustering and sharing (QSG z/OS only)



Resource definition for a Partial Repository QM1

```
DEFINE CHANNEL(TO.QM1) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)  
CONNAME(MACHINE1.IBM.COM) CLUSTER(DEMO)
```

- CLUSRCVR definition provides the information to the cluster that allows other queue managers to automatically define sender channels

```
DEFINE CHANNEL(TO.QM2) CHLTYPE(CLUSSDR) TRPTYPE(TCP)  
CONNAME(MACHINE2.IBM.COM) CLUSTER(DEMO)
```

- CLUSSDR definition must direct the queue manager to a Full Repository where it can find out information about the cluster

```
DEFINE QLOCAL(PAYROLLQ) CLUSTER(DEMO)
```

- Queues can be advertised to the cluster using the CLUSTER() keyword

```
DEFINE TOPIC(SPORTS) TOPICSTR(/global/sports) CLUSTER(DEMO)
```

- ...and so can Topics

Resource definition for Full Repository QM2

```
ALTER QMGR REPOS(DEMO)
```

- A queue manager is made into a Full Repository for a cluster by using the REPOS keyword on the QMGR object

```
DEFINE CHANNEL(TO.QM2) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)  
CONNAME(MACHINE2.IBM.COM) CLUSTER(DEMO)
```

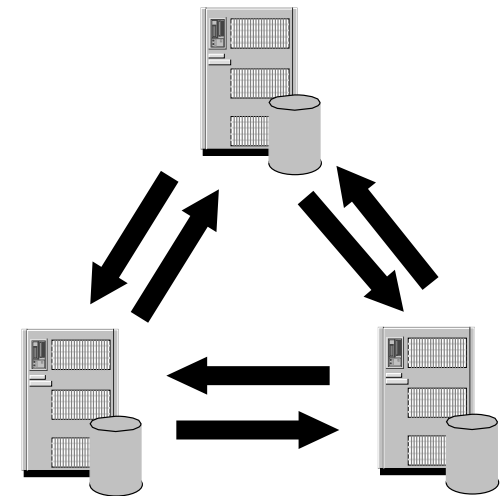
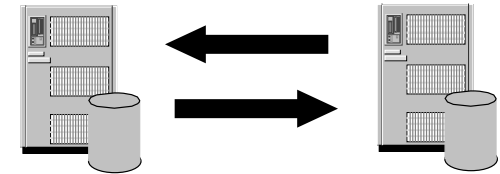
- CLUSRCVR definition provides the information to the cluster that allows other queue managers to automatically define sender channels

```
DEFINE CHANNEL(TO.QM3) CHLTYPE(CLUSSDR) TRPTYPE(TCP)  
CONNAME(MACHINE3.IBM.COM) CLUSTER(DEMO)
```

- CLUSSDR definition must direct the queue manager to another Full Repository for the cluster. When a Full Repository learns some information about the cluster it only forwards it to other Full Repositories for which it has a manually defined CLUSSDR channel

Considerations for Full Repositories (FRs)

- FRs should be highly available
 - ▶ Avoid single point of failure - have at least 2
 - ▶ Recommended to have exactly 2 unless you find a very good reason to have more
 - ▶ Put them on highly available machines
- FRs must be fully inter-connected
 - ▶ Using manually defined cluster sender channels
- If at least one FR is not available or they are not fully connected
 - ▶ Cluster definition changes via FRs will not flow
 - ▶ User messages between Partial Repositories over existing channels will flow



Checking initial definitions

- On the Full Repository display the new Partial Repository queue manager

```
DISPLAY CLUSQMGR(<partial repos qmgr name>)
```

- If the Partial Repository queue manager is not shown, the Full Repository has not heard from the Partial Repository
 - ▶ Check channel status from the Partial to the Full
 - ▶ Check CLUSRCVR and CLUSSDR channels CLUSTER name
 - ▶ Check for error messages

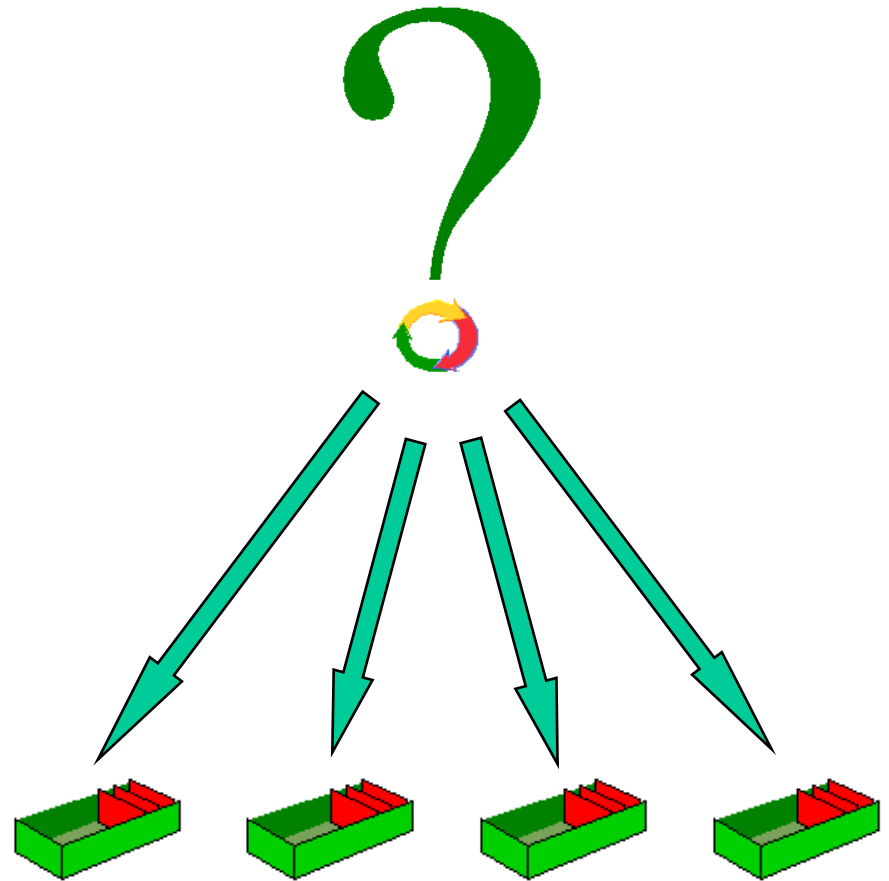
- On the Partial Repository display all cluster queue managers

```
DISPLAY CLUSQMGR(*)
```

- If the channel to the Full Repository has a queue manager name starting SYSTEM.TEMPQMGR, the Partial Repository has not heard back from the Full Repository
 - ▶ Check channel status from the Full to the Partial
 - ▶ Check for error messages

Agenda

- Clustering overview
- How cluster works
- Cluster configuration
- **Workload balancing**
- Best practice
- Clustering and sharing (QSG z/OS only)



Workload balancing

- When a cluster contains more than one instance of the same queue, workload balancing determines the best queue manager to route a message to.

Workload balancing

- MQ V6 and later version have additional parameters that can be use to influence the results of the algorithm.
 - Queues: CLWLPRTY, CLWLRANK, CLWLUSEQ
 - Queue Managers: CLWLUSEQ, CLWLMRUC
 - Channels: CLWLPRTY, CLWLRANK,CLWLWGHT, NETPRTY
- You can write your own cluster workload user exit

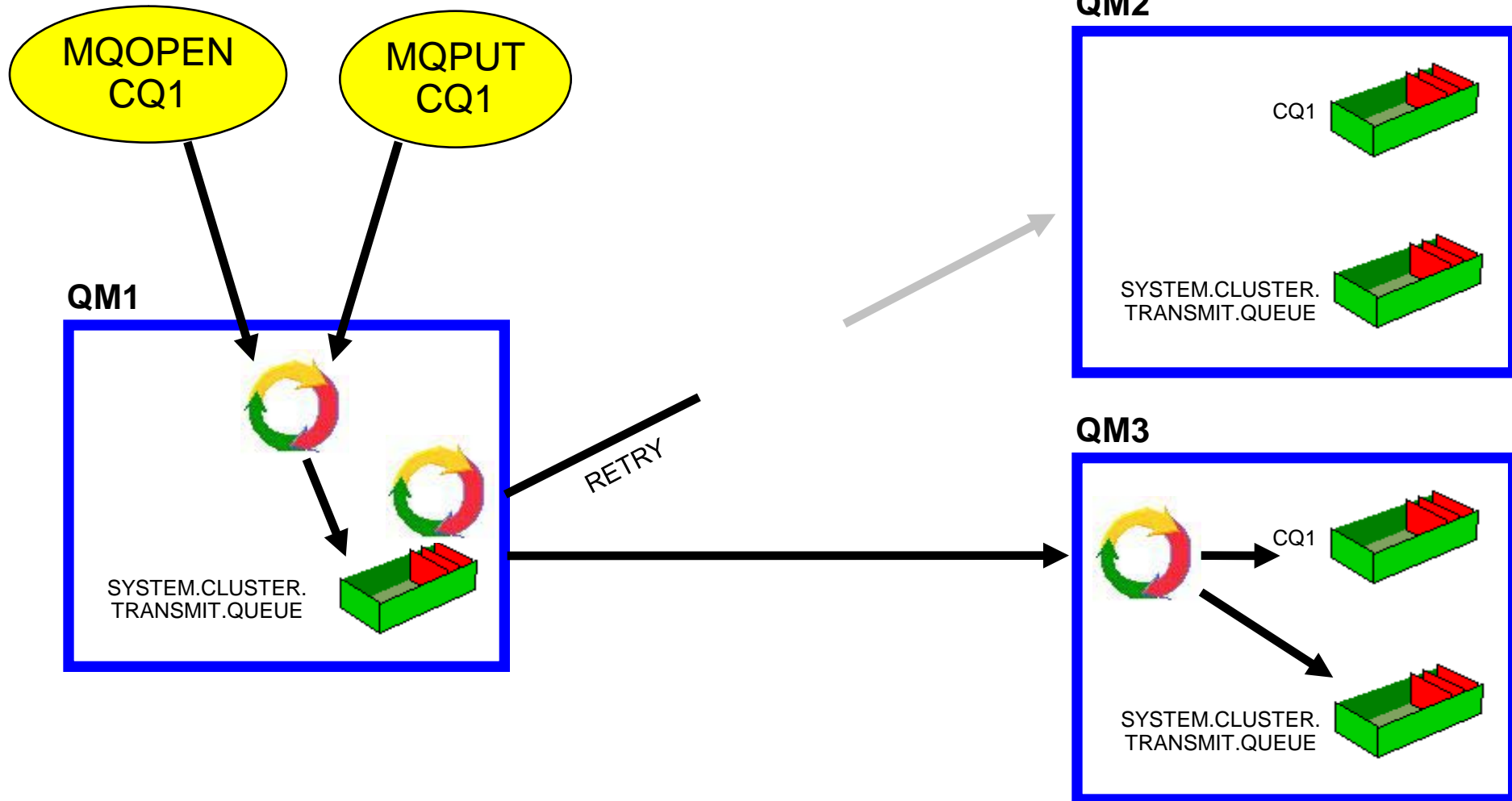
Workload balancing

- For workload balancing to occur:
 - open the queue with the MQOO_BIND_NOT_FIXED open option, or
 - open with the default MQOO_BIND_AS_Q_DEF and with DEFBIND(NOTFIXED) set in the queue definition. DEFBIND(OPEN) is the default.

Workload balancing

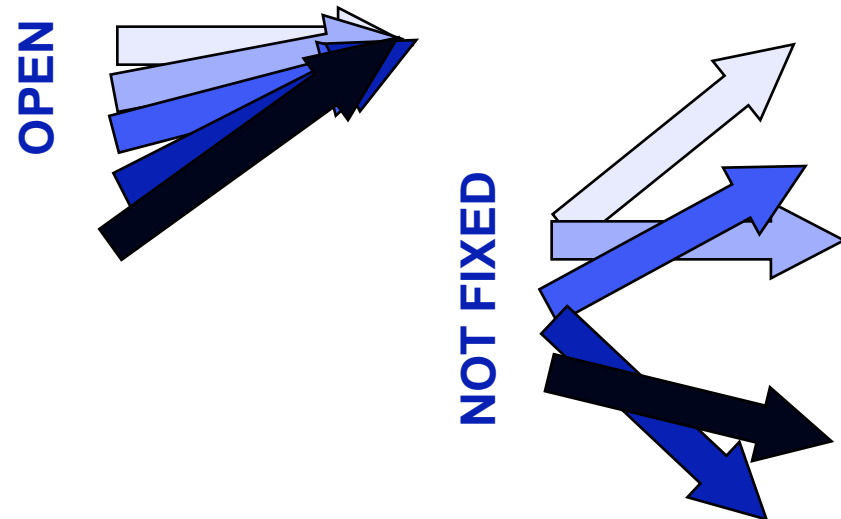
- Leave MQMD.ObjectQMgrName blank to allow the queue manager to choose the queue instance. To force the message to a specific instance of the clustered queue, specify that queue manager's name in ObjectQmgrName.

When does workload balancing occur?

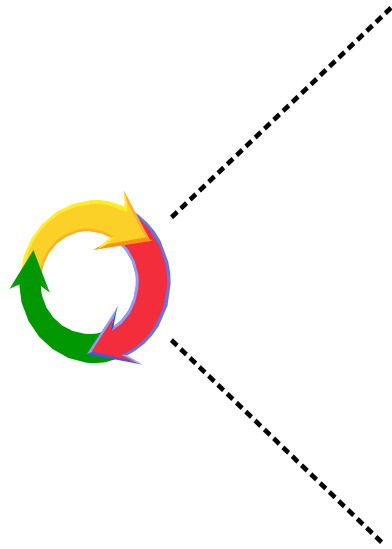


Workload balancing - Bind Options

- Bind on open
 - ▶ Messages are bound to a destination chosen at MQOPEN
 - ▶ All messages put using open handle are bound to same destination
- Bind not fixed
 - ▶ Each message is bound to a destination at MQPUT
 - ▶ Workload balancing done on every message
 - ▶ Recommended - No affinities are created (SCTQ build up)
- Application options
 - ▶ MQOO_BIND_ON_OPEN
 - ▶ MQOO_BIND_NOT_FIXED
 - ▶ MQOO_BIND_AS_Q_DEF (Default)
- DEFBIND Queue attribute
 - ▶ OPEN (Default)
 - ▶ NOTFIXED



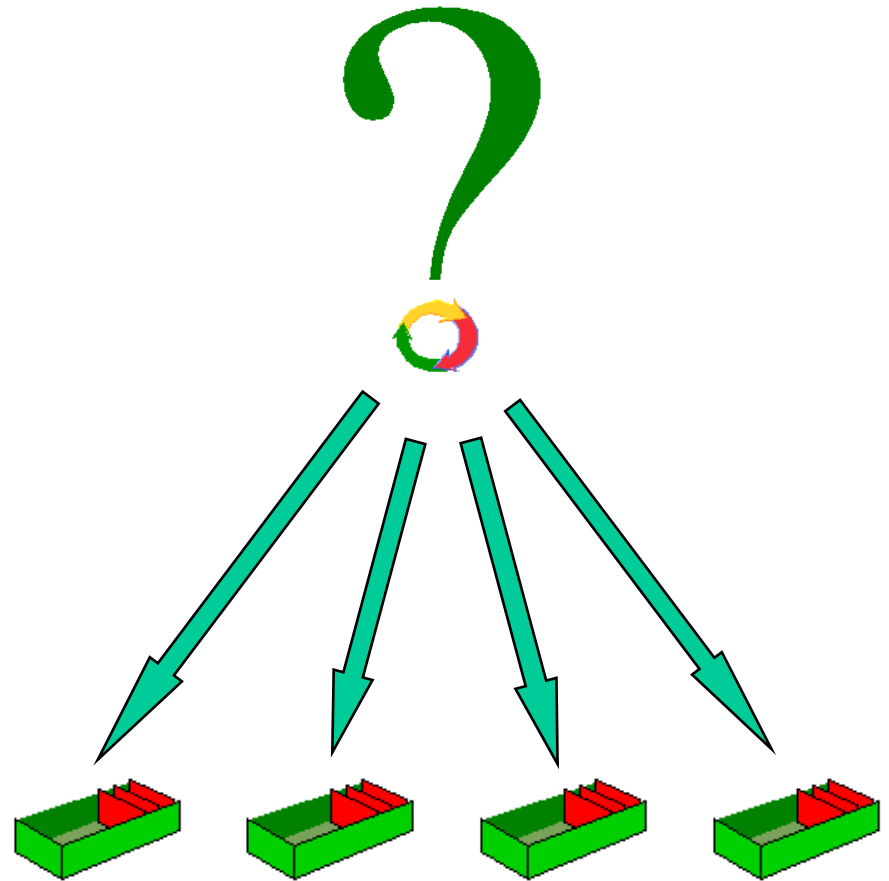
Cluster Workload Algorithm



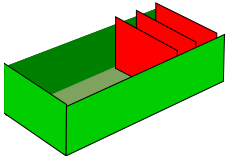
- Queue PUT(ENABLED/DISABLED)
- Local instance (CLWLUSEQ)
- Channel rank (CLWLRANK)
- Queue rank (CLWLRANK)
- Channel Status
 - INACTIVE, RUNNING
 - BINDING, INITIALIZING, STARTING, STOPPING
 - RETRYING
 - REQUESTING, PAUSED, STOPPED
- Channel net priority (NETPRTY)
- Channel priority (CLWLPRTY)
- Queue priority (CLWLPRTY)
- Most recently used (CLWLMRUC)
- Least recently used with channel weighting (CLWLWGHT)

Agenda

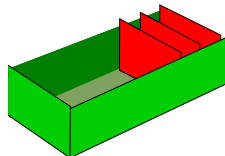
- Clustering overview
- How cluster works
- Cluster configuration
- Workload balancing
- Best practice
- Clustering and sharing (QSG z/OS only)



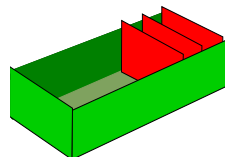
The SYSTEM.CLUSTER queues



- **SYSTEM.CLUSTER.COMMAND.QUEUE**
 - ▶ Holds inbound administrative messages
 - ▶ IPPROCS should always be 1
 - ▶ CURDEPTH should be zero or decrementing
 - ▶ If not, check repository task and error messages

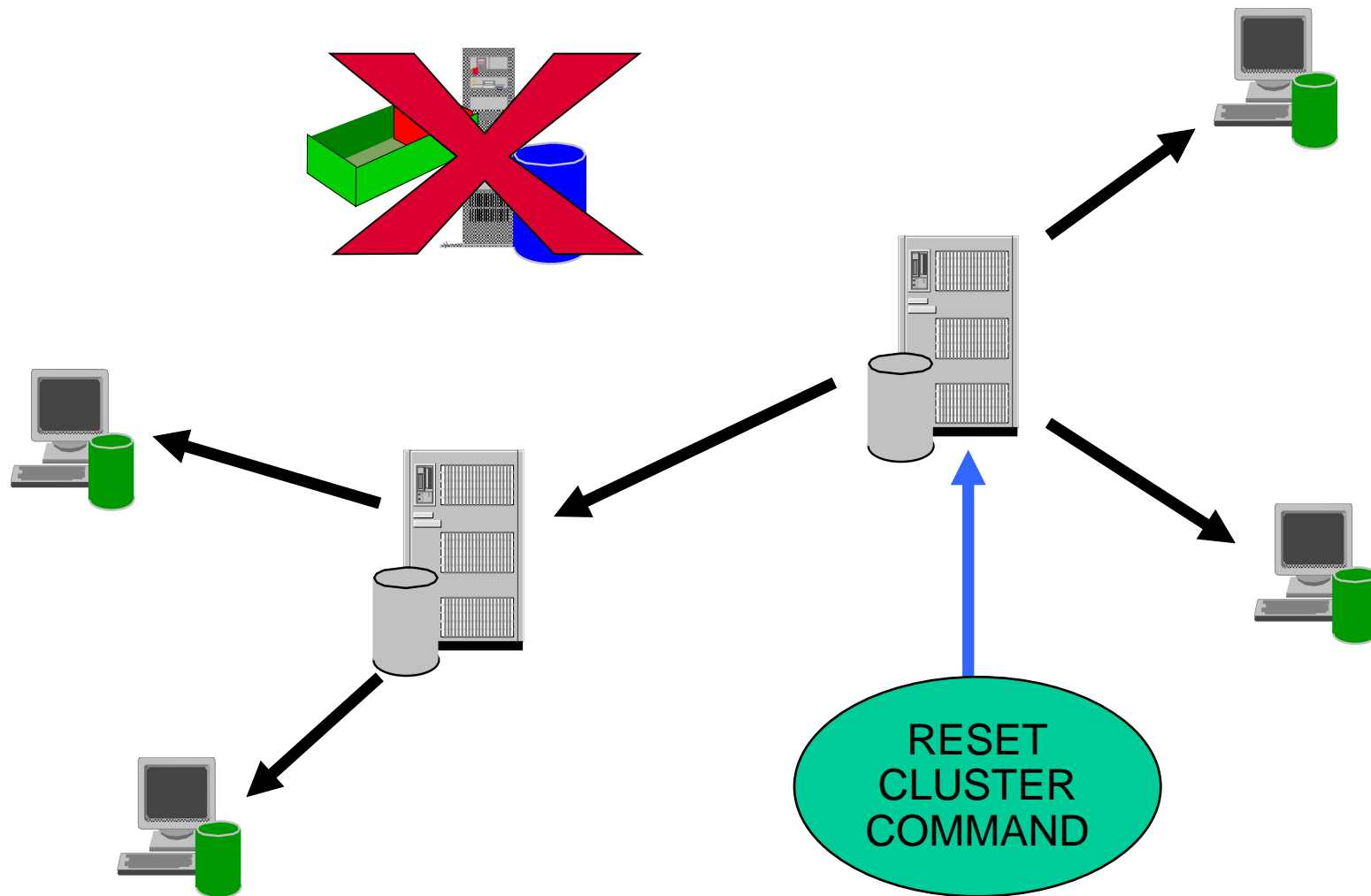


- **SYSTEM.CLUSTER.REPOSITORY.QUEUE**
 - ▶ Holds hardened view of repository cache
 - ▶ CURDEPTH should be greater than zero
 - ▶ CURDEPTH varies depending on checkpoints. This is not a problem.



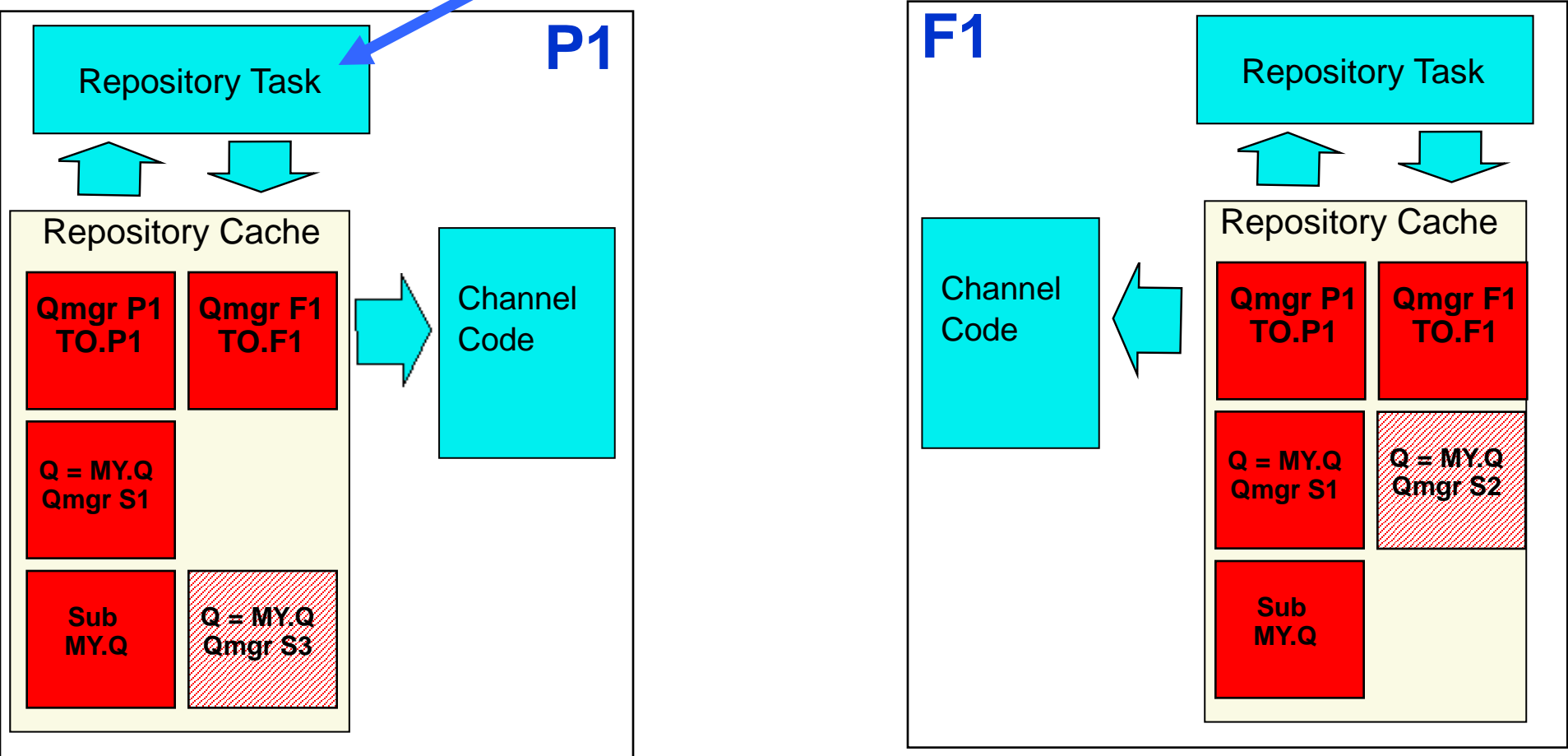
- **SYSTEM.CLUSTER.TRANSMIT.QUEUE**
 - ▶ Holds outbound administrative messages
 - ▶ Holds outbound user messages
 - ▶ CorrelId in MQMD added on transmission queue will contain the name of the channel that the message should be sent down

Incorrectly deleted queue managers



Refreshing repository information

REFRESH CLUSTER COMMAND



Refreshing repository information

- Messages have been removed from either the `SYSTEM.CLUSTER.COMMAND.QUEUE`, or from another manager's `SYSTEM.CLUSTER.TRANSMIT.QUEUE`, where the destination queue is `SYSTEM.CLUSTER.COMMAND.QUEUE` on the queue manager in question.
- Issuing a `REFRESH CLUSTER` command has been recommended by IBM® Service.
- The `CLUSRCVR` channels were removed from a cluster, or their `CONNAMES` were altered on two or more Full Repository queue managers while they could not communicate.
- The same name has been used for a `CLUSRCVR` channel on more than one queue manager in a cluster, and as a result, messages destined for one of the queue managers have been delivered to another. In this case, the duplicates should be removed, and then a `REFRESH CLUSTER` command should be issued on the single remaining queue manager that has the `CLUSRCVR` definition.
- `RESET CLUSTER ACTION(FORCEREMOVE)` was issued in error.
- The queue manager has been restarted from an earlier point in time than it last finished, (for example, by restoring backed up data.)

Implementation recommendations

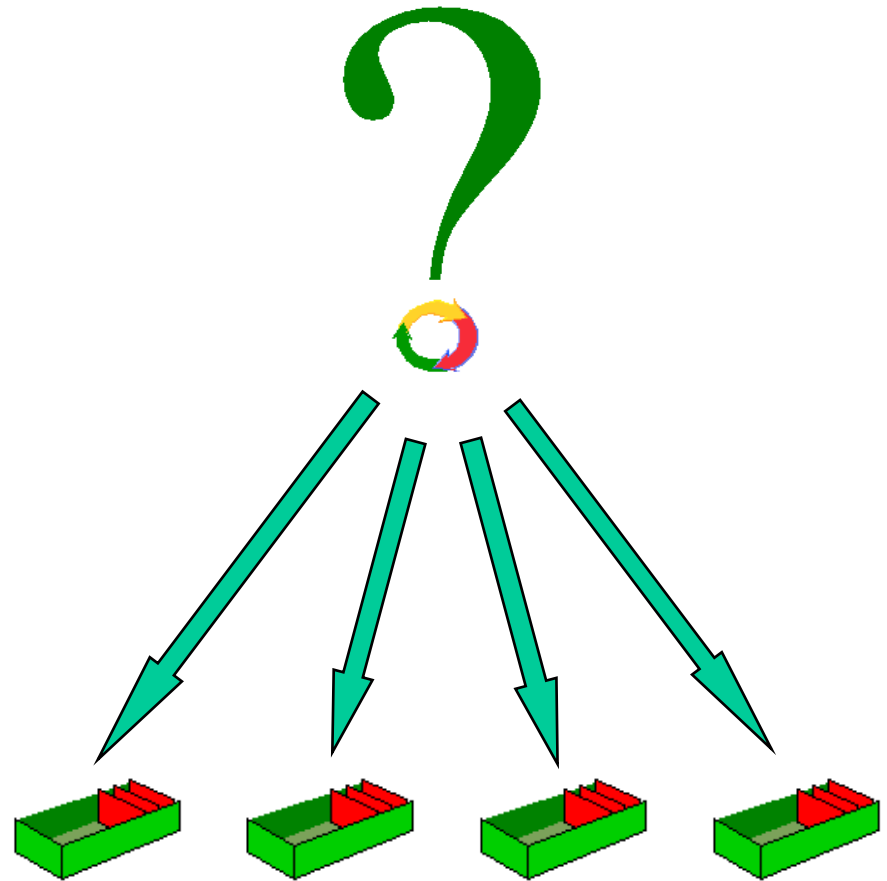
- Be clear about your requirements
 - ▶ Reduced system administration?
 - ▶ Workload balancing?
- Read the Queue Manager Clusters manual
 - ▶ The tasks sections are useful – Especially how to remove a queue manager from a cluster
- Helps to experiment in a development/test environment
 - ▶ Before using commands (especially REFRESH CLUSTER) read the whole section for that command in the MQSC manual
- Usage notes contain important info
- Naming conventions
 - ▶ No one recommended convention - consistency is the key
- Document processes for production system changes
 - ▶ Add/remove queue manager to/from cluster
 - ▶ Take queue manager offline for maintenance
- Have two Full Repositories
 - ▶ Be careful about connecting them
 - ▶ Consider where you should host them

Implementation recommendations (continued)

- Be careful defining cluster topics in existing clusters
- Monitor the SYSTEM.CLUSTER.TRANSMIT.QUEUE
 - ▶ Per channel CURDEPTH (V6) – DIS CHSTATUS XQMSGSA
- When debugging check that channels are healthy
 - ▶ Definition propagation: Path from one QM to another via FRs
 - ▶ Application: Path from application to queue manager hosting queue.
- Consider how you will administrate and debug
 - ▶ Monitor CSQX4.../AMQ94.. messages
 - ▶ MQRC_UNKNOWN_OBJECT_NAME
 - ▶ “Where’s my message?” (when there are now n cluster queues)
- Bind-not-fixed gives better availability
 - ▶ By default, bind-on-open is used
- Further Information in the Infocenter
 - ▶ Queue Manager Clusters
 - ▶ Script (MQSC) Command Reference
 - ▶ Publish/Subscribe User’s Guide

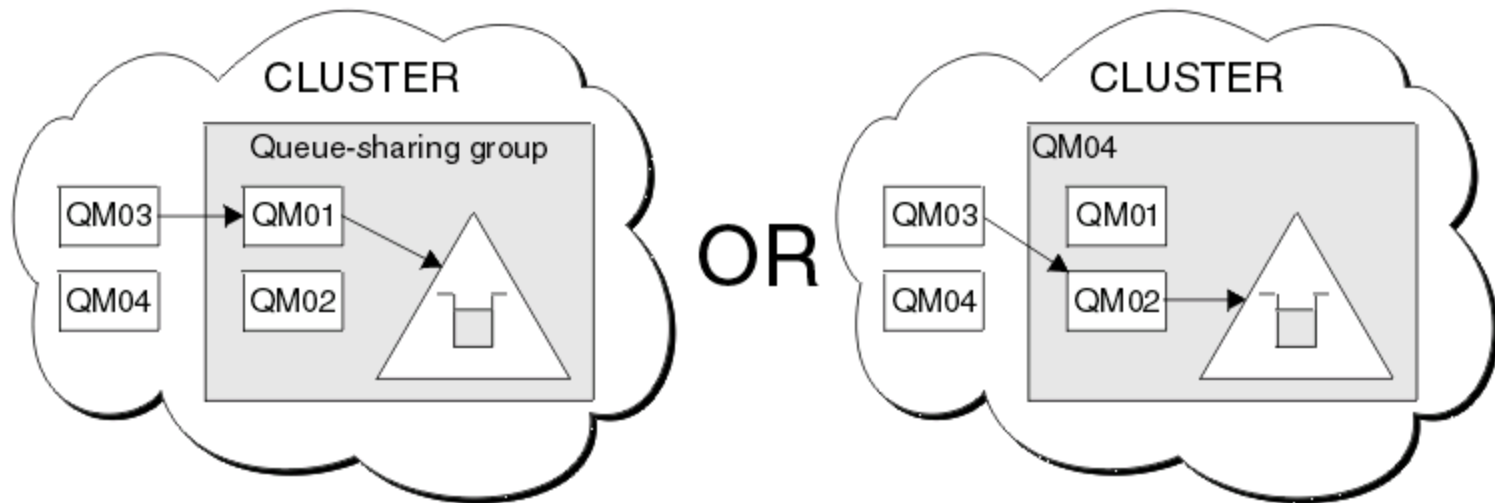
Agenda

- Clustering overview
- How cluster works
- Cluster configuration
- Workload balancing
- Best practice
- Clustering and sharing (QSG z/OS only)



Clustering and Sharing(QSG)

- You can define shared queues to be cluster queues. Such a queue is advertised as being hosted by each queue manager in the QSG which is also a member of the cluster.



Clustering VS. Sharing

Consider using clustering for the following reasons:

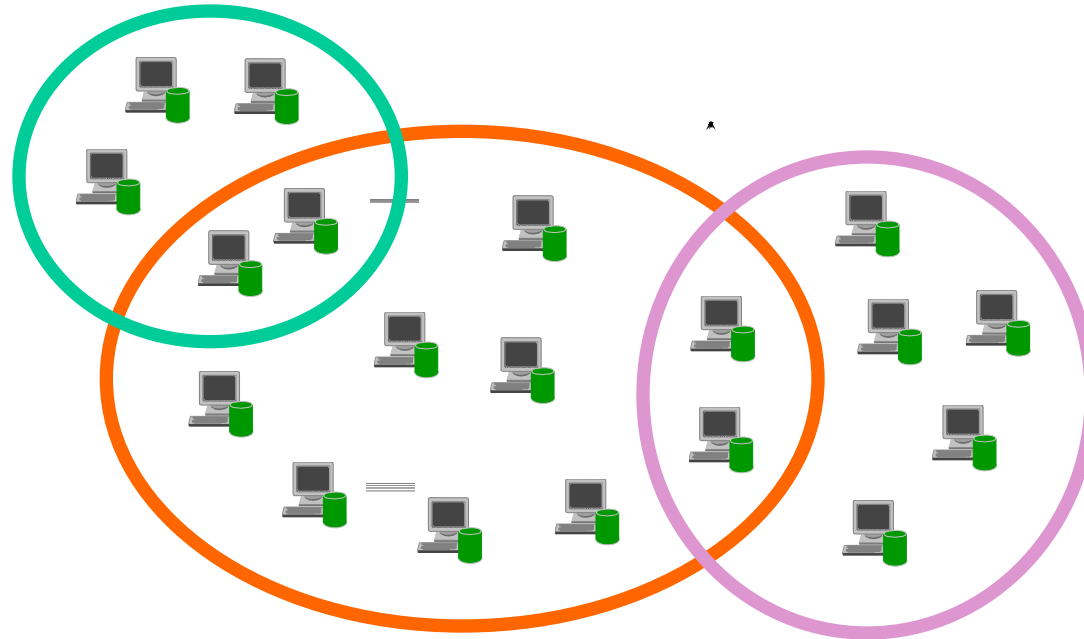
- You have several queue managers on different platforms and you want them to participate in the workload balancing and availability.
- You wish to simplify the channel definitions between related queue managers.

Clustering VS. Sharing

- considerations exploiting the Parallel Sysplex, the following recommendations presumes that a Parallel Sysplex is provided and feasible:
 - ▶ Establish one or more QSGs.
 - ▶ Implement a front-end routing mechanism to distribute work dynamically between the queue managers using the generic port concept.
 - ▶ Enable the applications to run several clones in parallel and support continuous availability. You probably need this regardless of whether you are using sharing and/or clustering. With respect to WebSphere MQ, you may not have to change your code at all.
 - ▶ Implement shared inbound and shared outbound channels according to your needs.

Summary

- Clustering overview
- How cluster works
- Cluster configuration
- Workload balancing
- Best practice
- Clustering and sharing (QSG z/OS only)



Further Clustering sessions

■ **Advanced Clustering**

- ▶ Pub Sub in more detail
- ▶ Routes in/out of clusters
- ▶ Workload balancing
- ▶ Security
- ▶ (A little) on HA and DR

■ **Best Practice on Clustering**

- ▶ Conventions
- ▶ Topologies
- ▶ Full repositories
- ▶ Migration (platform, host)
- ▶ Migration (release, fixpack)
- ▶ Staying available
- ▶ Security
- ▶ Monitoring

OpenMic session introduction / Survey

- ✓ Frequency: Bi-Monthly
- ✓ Products:
 - ▶ **WebSphere MQ**
 - ▶ **WebSphere Message Broker**
 - ▶ **WebSphere Application Server**
- Candidate topics survey for year 2011
 - **MQ**
 - ▶ **MQ PUB/SUB**
 - ▶ **MQ Cluster**
 - ▶ **Deploy & running MQ client on WAS.**
 - **WAS**
 - ▶ **WAS introduction**
 - ▶ **Advanced security configuration Enhanced Admin**
 - ▶ **Session management**
 - ▶ **Data source and connection pooling**

Q & A

