MQ Pub/Sub: direct routing clusters and proxy subscriptions

http://www-01.ibm.com/support/docview.wss?uid=swg27050262

Angel Rivera (rivera@us.ibm.com)
IBM MQ Distributed Level 2 Support
Date last updated: 03-Oct-2017

ON DEMAND BUSINESS™

Related presentations

This presentation is one of a series.
For the complete list, please see:

https://developer.ibm.com/answers/questions/40207
4/mq-pubsub-training-presentations.html
MQ Pub/Sub: training presentations

Related zip file

This techdoc has 1 zip file with files that are discussed in this presentation:

QMPS-pub-sub-direct-routing-clusters.zip

The included files contain information after the clustered topic was created in PR 4 and the subscriber was created in PR 7.
The runmqsc* files contain a variety of commands to display the topics, etc.

runmqsc_CDPR4_after_subscribe.txt => PR 4
runmqsc_CDPR7_after_subscribe.txt => PR 7
amqrfdm_CDRPR6.stdout => cluster cache in PR 6
amqrfdm_CDRPR7.stdout => cluster cache in PR 7

Agenda

- Review of using **clustered queues in a cluster**

- Adding a **clustered topic**

- Notice impact that in **direct route clusters:**
the auto defined cluster-senders are created in all
partial repositories!

- **Proxy subscriptions** are briefly explained.

Tutorials for creating a cluster

The cluster for this presentation was based on the following tutorials:

http://www-01.ibm.com/support/docview.wss?uid=swg27038687
Cluster setup and basic usage of clustered queues and topics in MQ 7

The above WSTE is based on these tutorials:

http://www.ibm.com/support/docview.wss?uid=swg27037038
Setup of a cluster and basic usage of clustered queues in MQ 7

http://www.ibm.com/support/docview.wss?uid=swg27038974
Basic usage of clustered topics in WebSphere MQ 7

Initial topology of Direct Routing cluster

There are 2 hosts.

Each host has 4 queue managers:
 - 1 Full Repository (FR)
 - 3 Partial Repositories (PR)

Each FR is connected to the 6 PRs
Each PR is connected to the 2 FRs

There are no connections between the PRs.
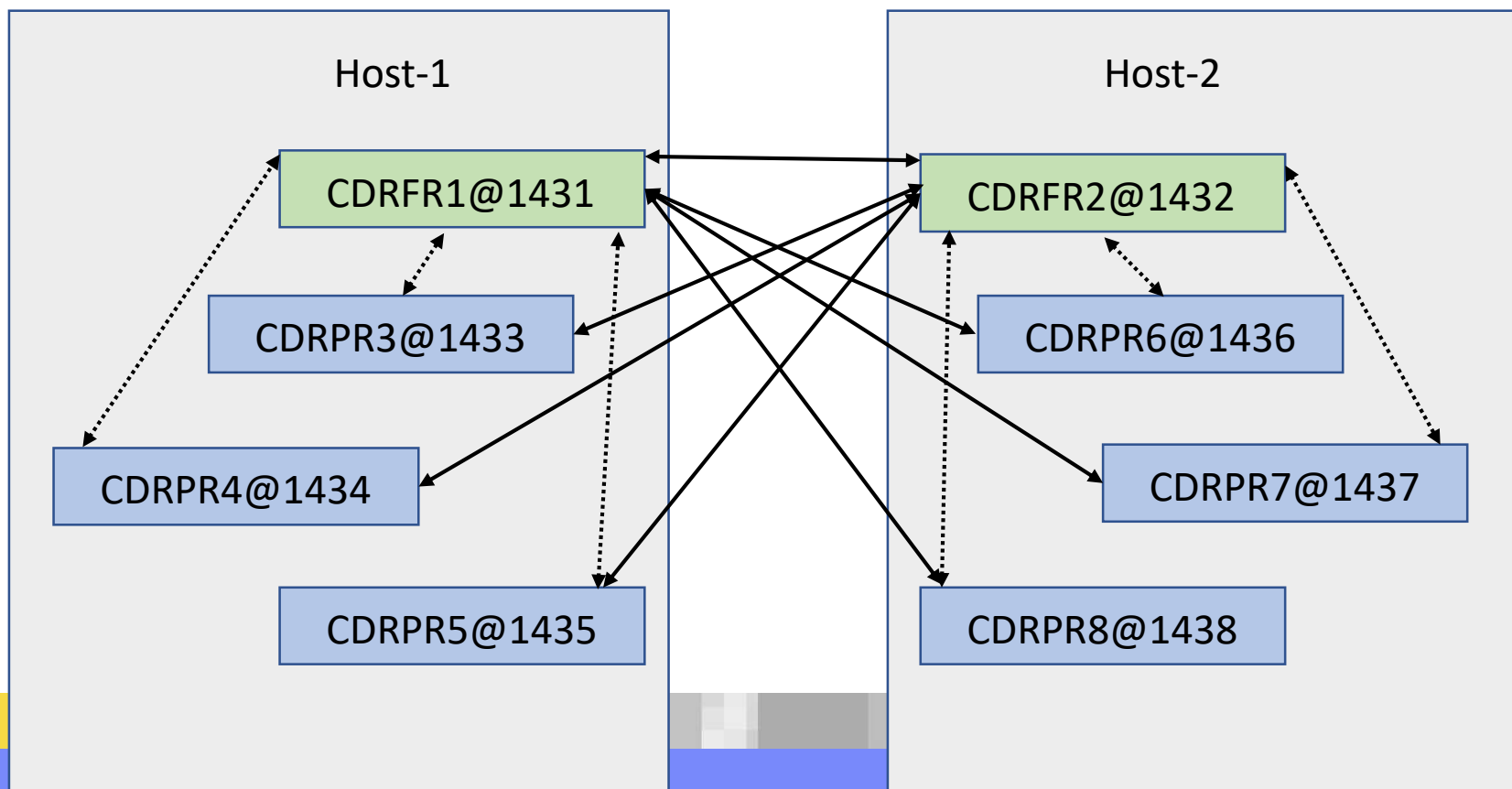There are no clustered queues, nor clustered topics

Initial topology of Direct Routing cluster

Cluster-sender and cluster-receiver channels that connect the queue managers.
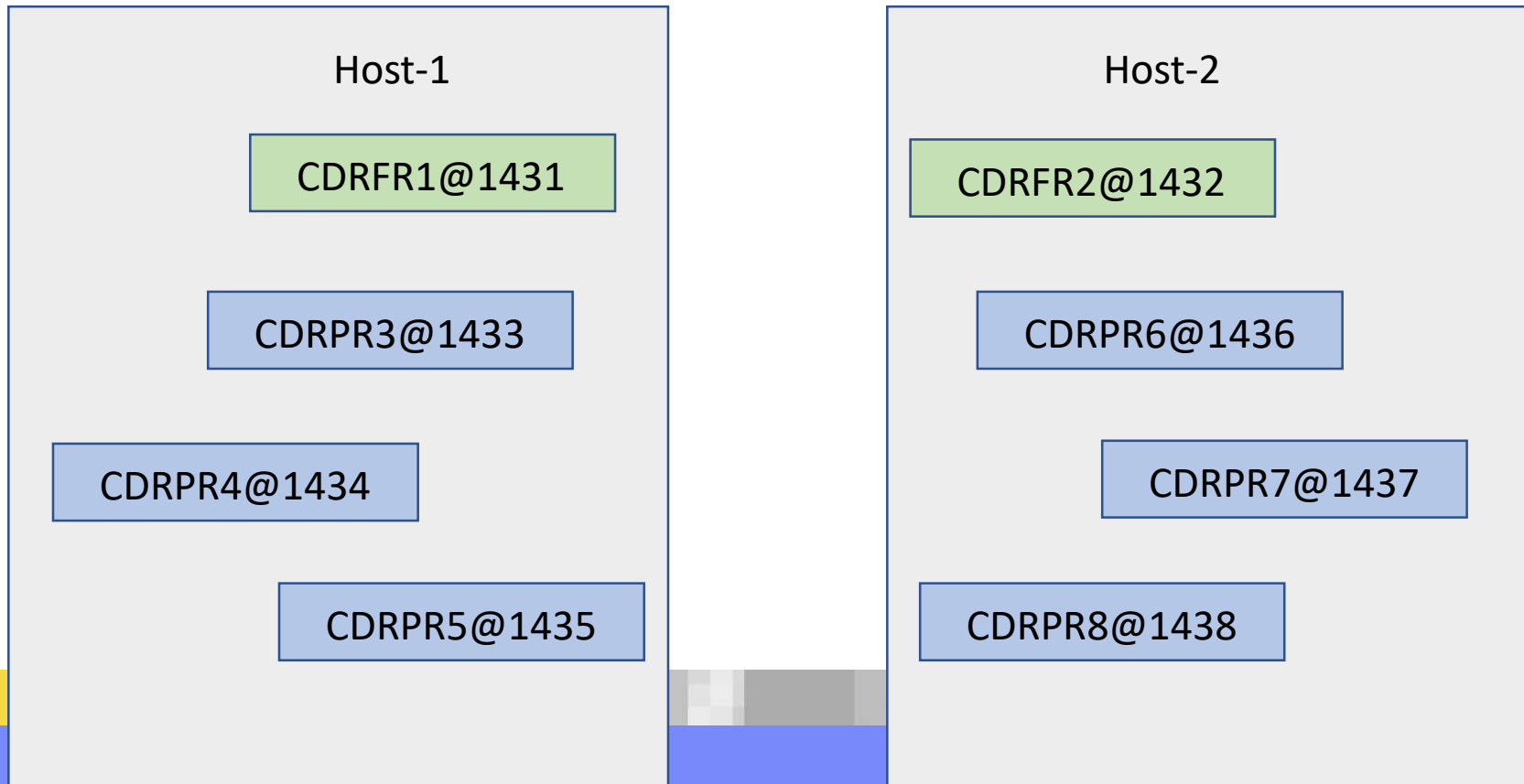
Dotted line: manually created   Solid line: automatically created

Simpler view of the topology

To simplify the view, let's hide the cluster channels for the infrastructure.
Only NEW channels will be shown later on.

Host-1

CDRFR1@1431

CDRPR3@1433

CDRPR4@1434

CDRPR5@1435

Host-2

CDRFR2@1432

CDRPR6@1436

CDRPR7@1437

CDRPR8@1438

PR 7 only has CS channels to the FRs

The PR number 7 (CDRPR7) will be used later on.

Currently, it has ONLY 2 cluster-sender channels, one to each of the FRs:

- TO.CDRFR2 was manually created when PR was added to cluster.
- TO.CDRFR1 was automatically created.

Both channels are shown in the tab **"Cluster-sender Channels"**, in the folder "Queue Manager Clusters" of the MQ Explorer (next page)

PR 7 only has CS channels to the FRs

PR 7 only has CS channels to the FRs

The MQ Explorer, under the **Channels folder** for the queue manager, shows ONLY the manually created channels.

The automatically created channels are NOT shown. In this case, TO.CDRFR1 is NOT shown.

Notes: runmqsc shows manual channels

To narrow the scope of the output list, the SYSTEM* channels were removed.

Notice that **only the 2 channels that were manually created are shown.**

The cluster-sender channels that are created automatically are NOT shown (for example, TO.CDRFR1 for FR1)

runmqsc CDRPR7
display channel(*)
   1 : display channel(*)
AMQ8414I: Display Channel details.
  CHANNEL(TO.CDRFR2)   **CHLTYPE(CLUSSDR)**
AMQ8414I: Display Channel details.
  CHANNEL(TO.CDRPR7)   CHLTYPE(CLUSRCVR)

notes

Notes: amqrfdm shows automatic channels

The utility amqrfdm is used to view the cluster cache. (More details later on)
It is invoked by runmqras (when using "-section cluster").
Issue the following to manually run the utility (include the flag –d):

amqrfdm -m CDRPR7 -d > amqrfdm_CDRPR7.stdout

The following entry is for the auto cluster sender channel to the other FR: CDRFR1
To keep listing brief, some lines were removed.

Qm(CDRFR1   ) Live   Seq(1505574418)
 Channel(TO.CDRFR1   ) Stopped   ChlSeq(26) DestSeqFactor(0)
 CLWLChannelRank(0) CLWLChannelPriority(0) CLWLChannelWeight(50)
 XmitQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE   )
 Conname('angelito.raleigh.ibm.com(1431)')
Desc(   )
 UUID(CDRFR1_2017-09-13_13.09.49   )
 QMFlags(2272: Repos CLUSSDR Auto Joined InUse Refresh)
 State: Flags(0) MsgId(414D51204344525250237202020202020C400BD59A4AD4721)
 Product: MQMM   Version: 09000300
 ChosenCount(8)
Cluster(CLUSTERDR   ) Live   Seq(1505574418)
 Exp(Mon 16 Oct 2017 10:50:14 AM GMT) Raw(x59E48EE6)
 Upd(Sat 16 Sep 2017 10:50:14 AM GMT) Raw(x59BD01E6)
 QMFlags(2032: Repos CLUSSDR Auto Joined)
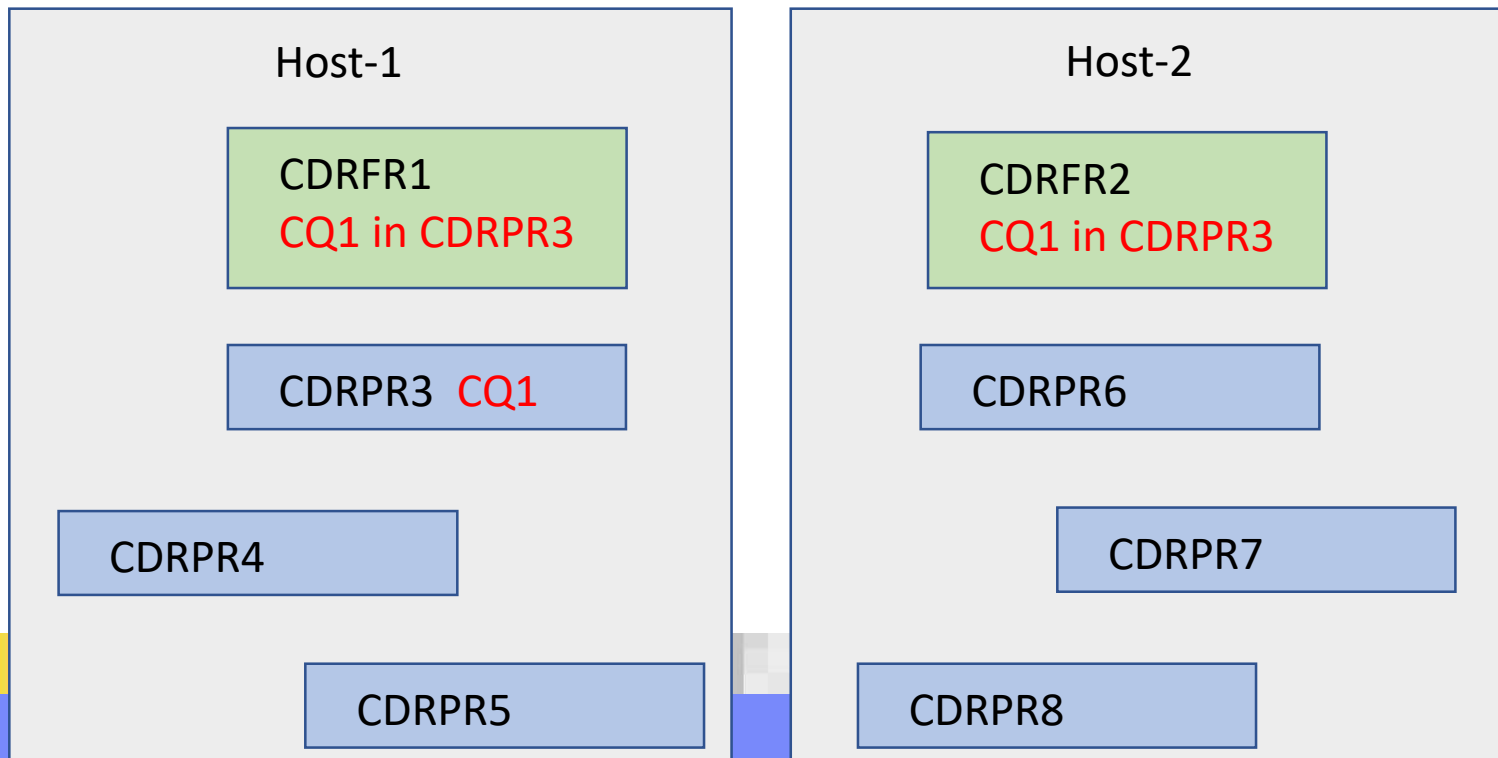
Adding a clustered QUEUE in PR 3

Let's add a local queue CQ1 in PR 3: CDRPR3
define qlocal(CQ1) cluster(CLUSTERDR)
It is made a **clustered queue**. The FRs know it.
But the other PRs DO NOT know about it yet.

Host-1	Host-2
CDRFR1 CQ1 in CDRPR3	**CDRFR2** CQ1 in CDRPR3
CDRPR3 CQ1	CDRPR6
CDRPR4	CDRPR7
CDRPR5	CDRPR8

FRs know the local CQ1 is in PR 3

- The FRs know that the local CQ1 for the clustered queue CQ1 is in PR 3 (CDRPR3)

Put done to clustered queue in another PR

A PUT application connects to PR 7 (CDRPR7) and wants to put a message into Clustered Queue CQ1

Host-1

CDRFR1
CQ1 in CDRPR3

CDRPR3 CQ1

CDRPR4

CDRPR5

Host-2

CDRFR2
CQ1 in CDRPR3

CDRPR6

CDRPR7

CDRPR8

PUT
Clustered
CQ1

16
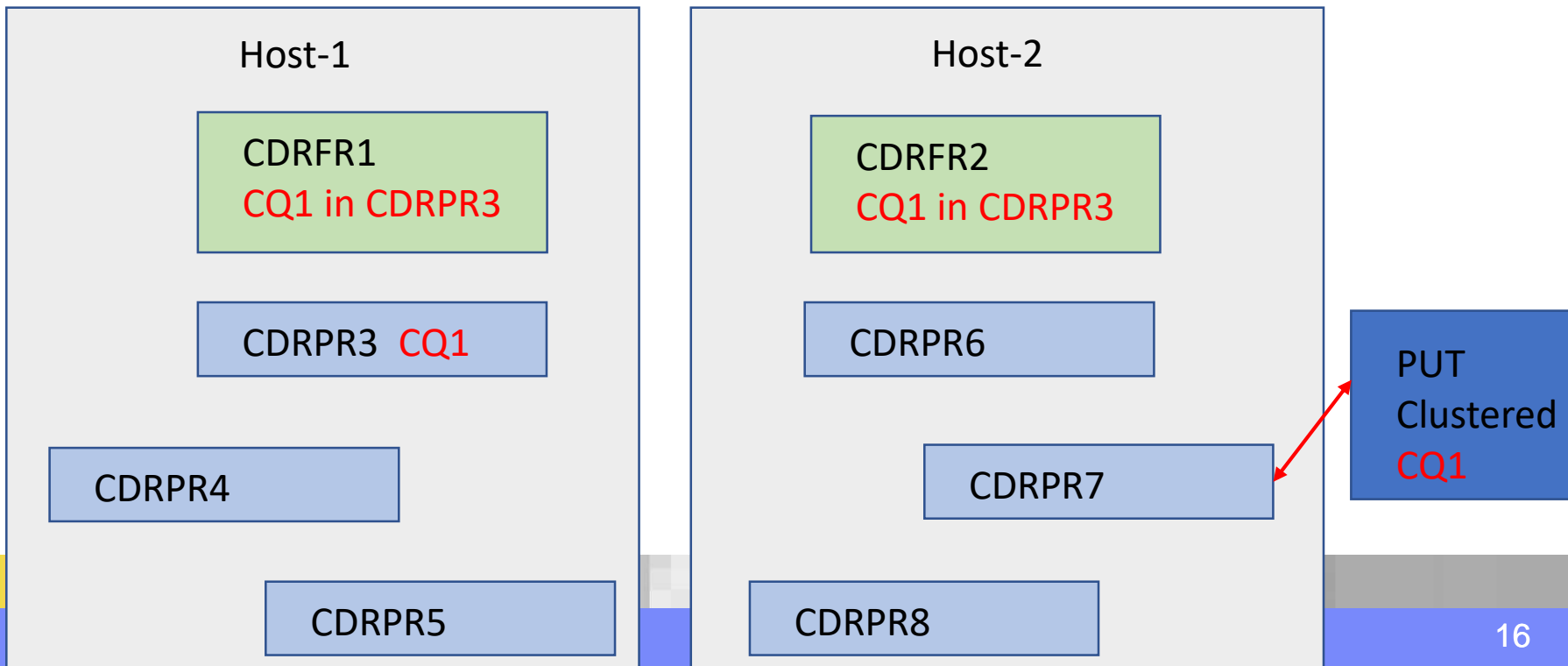
Put done to clustered queue in another PR

Initially CDRPR7 does not know about CQ1

Nor in the list of
Local queues

Nor in the list of
Cluster Queues

Adding a clustered queue in a PR

CDRPR7 asks a FR if there is a clustered queue CQ1 and if so, asks where the local copy is located

Host-1

CDRFR1
CQ1 in CDRPR3

CDRPR3 CQ1

CDRPR4

CDRPR5

Host-2

CDRFR2
CQ1 in CDRPR3

CDRPR6

CDRPR7

CDRPR8

PUT Clustered CQ1

18
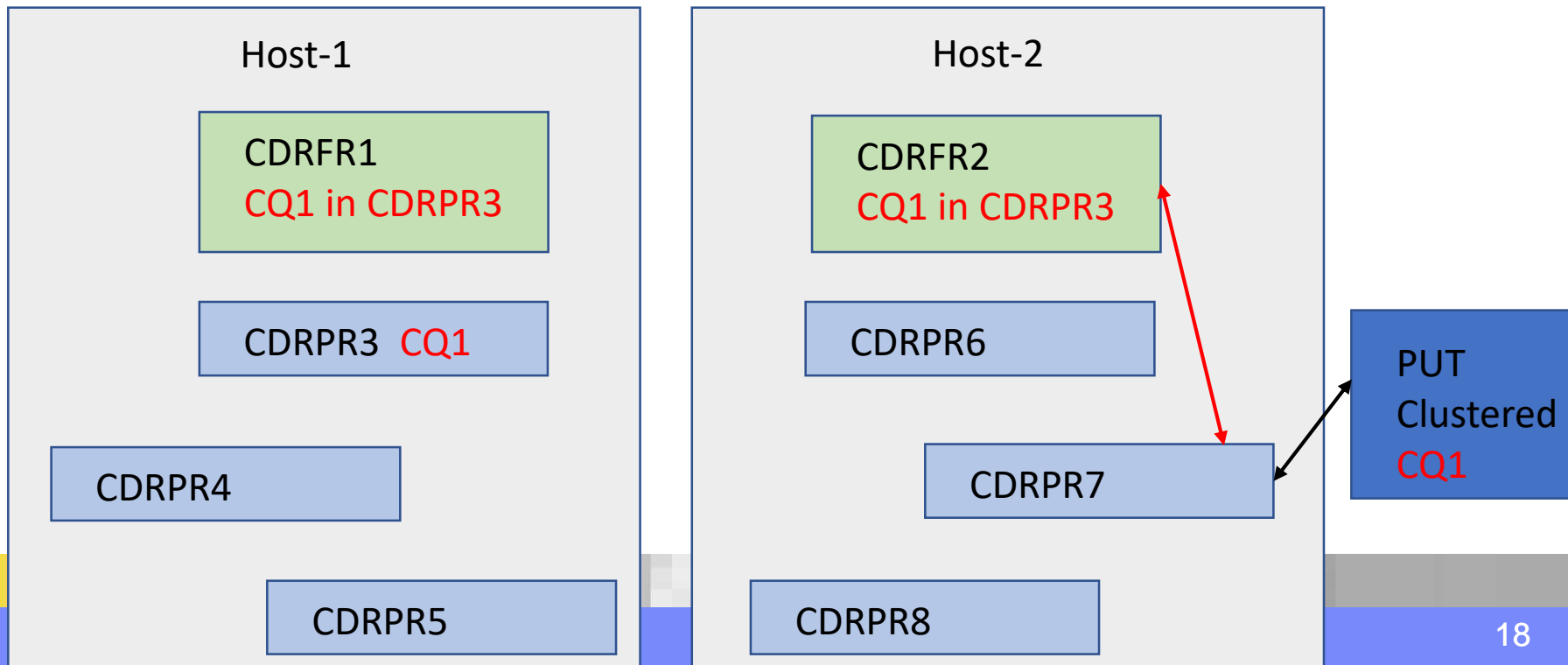
Adding a clustered queue in a PR

The FR indicates that local CQ1 is in CDRPR3.
The clustering code **creates automatically a cluster sender channel** from CDRPR7 to CDRPR3.
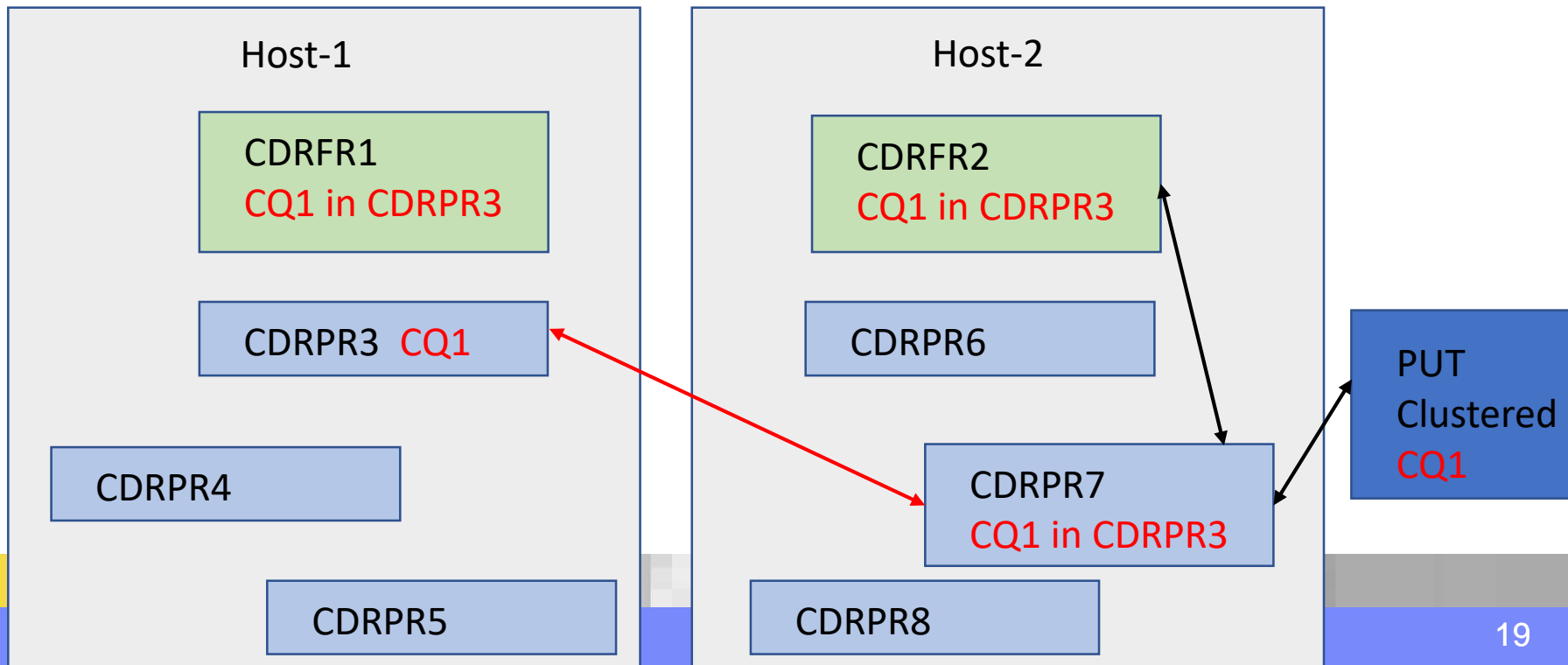
PR 7 has now a channel to PR 3

PR 7 now knows that the clustered CQ1 points to the local queue CQ1 in PR 3.

PR 7 has now a channel to PR 3

The "auto cluster-sender" channel TO.CDRPR3 was added to PR 7 to connect it to PR 3, in order to send a message to the queue CQ1.

PR 7 has now a channel to PR 3

The message that was placed in PR 7 was
transmitted to PR 3 and stored in
the local queue CQ1.
The number of messages in that queue went from
0 to 1
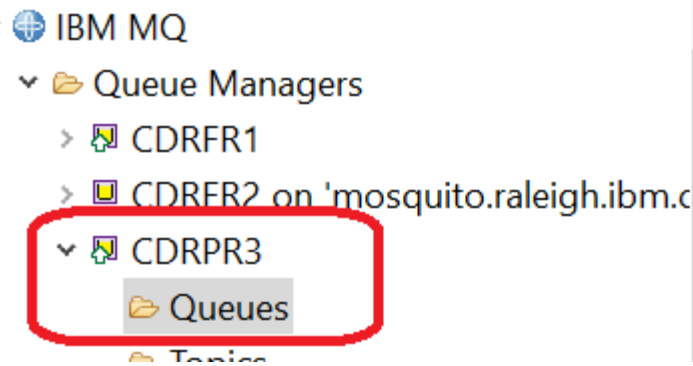
Queues, ONLY needed channel is created

- At this point ONLY the cluster sender from PR 7 to PR 3 was added automatically.
- No additional channels were created.

Host-1

| CDRFR1 |
| CQ1 in CDRPR3 |

| CDRPR3 CQ1 |

| CDRPR4 |

| CDRPR5 |

Host-2

| CDRFR2 |
| CQ1 in CDRPR3 |

| CDRPR6 |

| CDRPR7 |
| CQ1 in CDRPR3 |

| CDRPR8 |

23
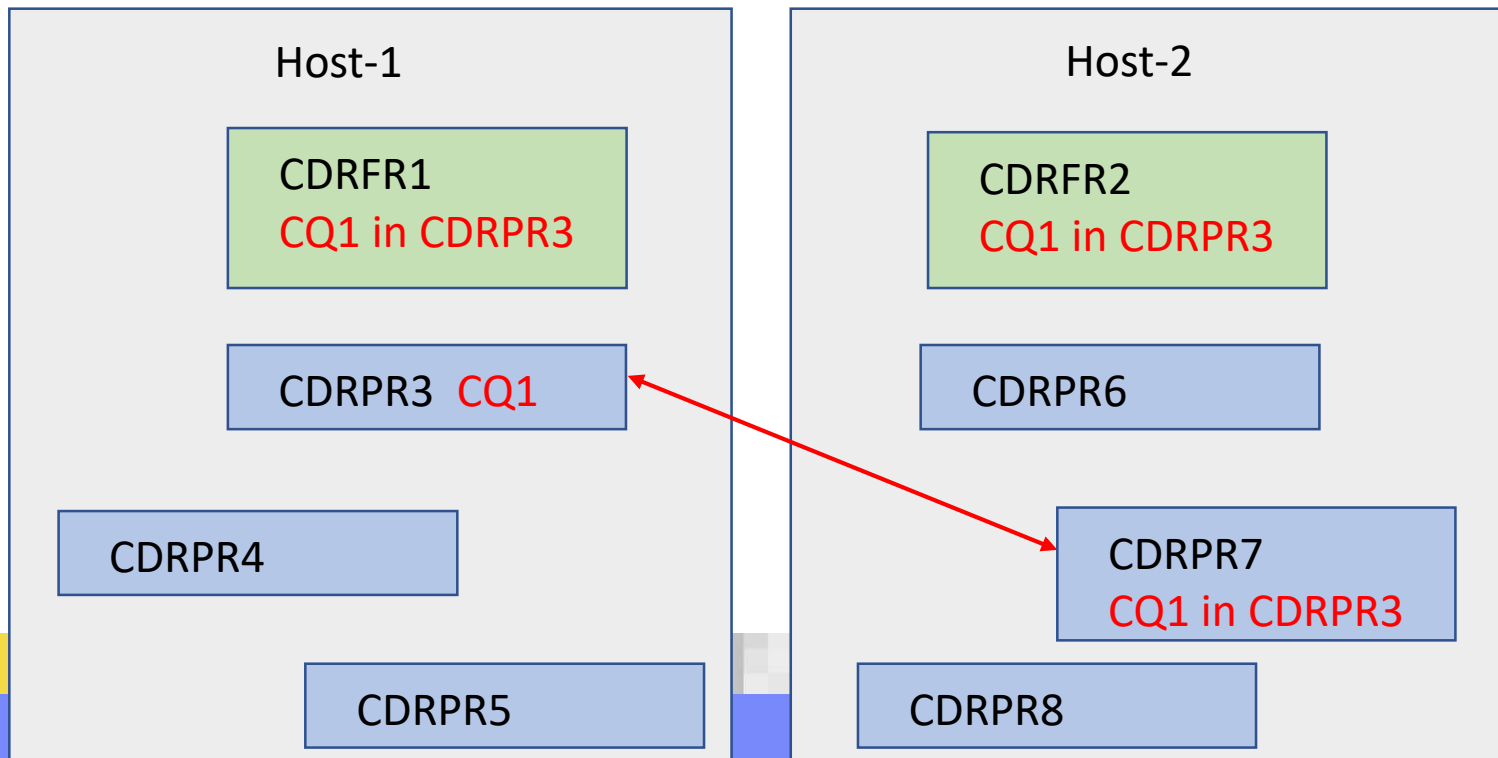
Extra: MQGET cannot be done indirectly

The PUT application for CQ1 is in CDRPR7.
Could we have a GET application for CQ1 also in
CDRPR7?   ANSWER: NO!!

Host-1

CDRFR1
CQ1 in CDRPR3

CDRPR3 CQ1

CDRPR4

CDRPR5

Host-2

CDRFR2
CQ1 in CDRPR3

CDRPR6

CDRPR7

CDRPR8

GET
Clustered
CQ1

PUT
Clustered
CQ1
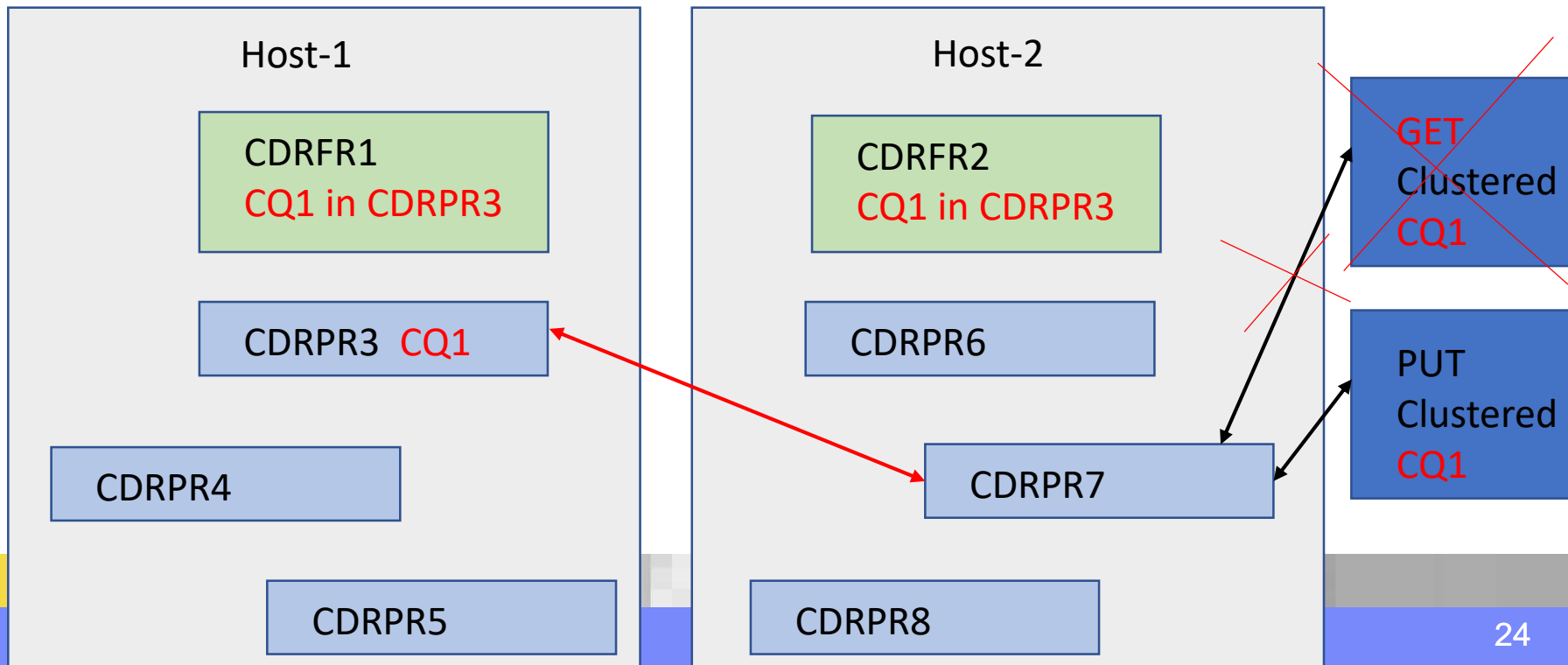
Extra: MQGET must be done directly local Q

The tutorial for setup of a cluster shows in page 44:
MQ GETs are OK only when using the local Q.
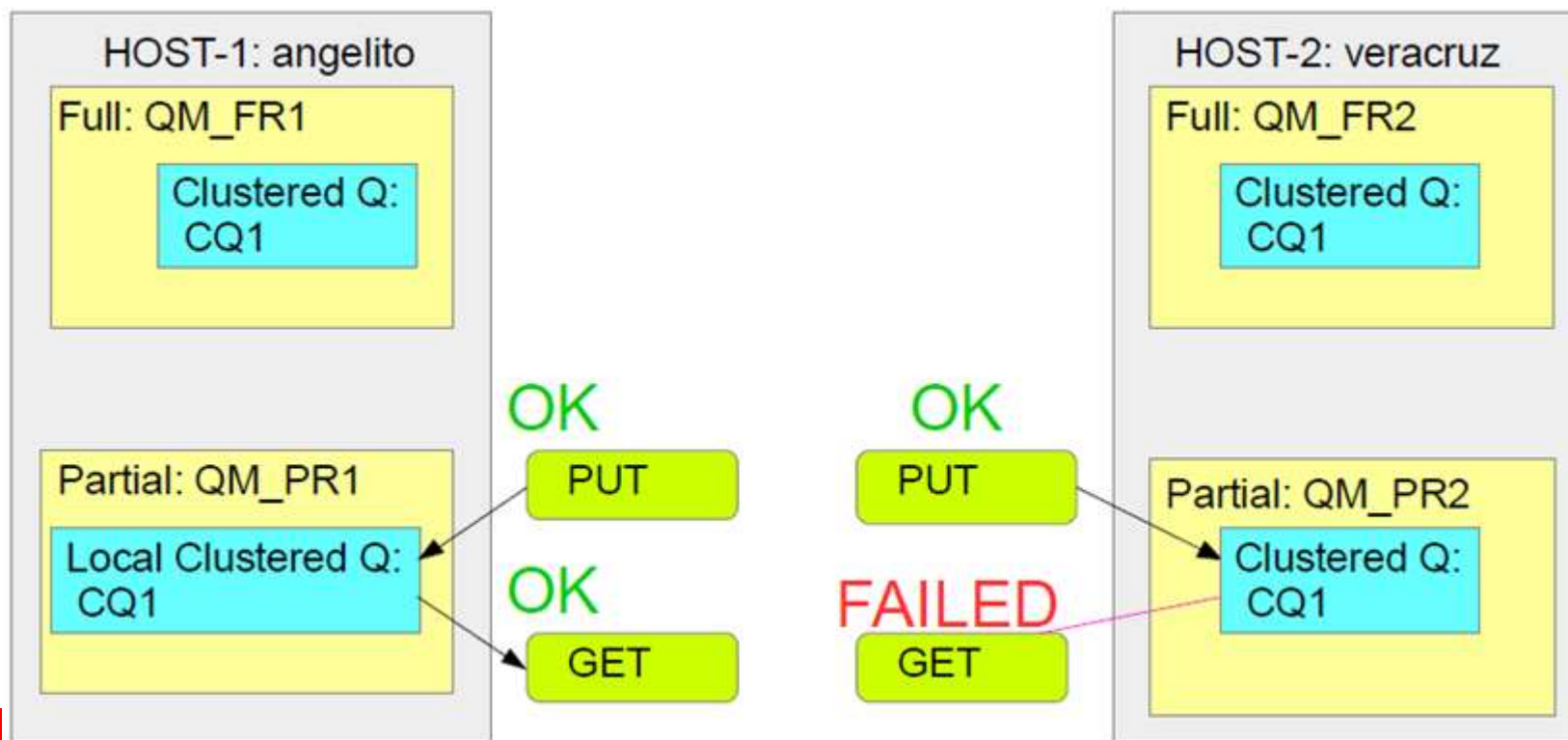
MQ cluster utility: amqrfdm to view cache

The MQ utility "**amqrfdm**" can be used to view the cluster cache for a queue manager.
Used by "runmqras" with **"-section cluster"**.
You can also run it manually (must specify **-d**).

Unix example: For the Partial Repository CDRPR7
amqrfdm -m CDRPR7 -d > amqrfdm_CDRPR7.txt

Windows example: For the Full Repository CDRFR1
**"C:\Program Files\IBM\MQ\bin64\amqrfdm.exe" -m CDRFR1
-d > "C:\temp\amqrfdm_CDRFR1.txt"**

Notes: Viewing the cluster cache, FR

* Windows example: For the Full Repository CDRFR1
"C:\Program Files\IBM\MQ\bin64\amqrfdm.exe" -m CDRFR1 -d > "C:\temp\amqrfdm_CDRFR1.txt"

Example for clustered queue CQ1:
It shows 2 entries:
The 1st is for the "clustered Queue" itself.
The 2nd, for the corresponding "local" queue, which is in CDRPR3.

```
Q(CQ1                                              ) Live    Seq(0)
 Offset: @11510
 ptr(880D2E90)
 Cluster(CLUSTERDR                                              )
 UUID(CDRPR3_2017-09-13_13.14.19                      )
 SubID(15E       7C3C49B9)
 Exp(10/13/2017 6:43:13 PM)  Raw(x59E10941)
 Upd(9/13/2017 6:43:13 PM)  Raw(x59B97C41)
 Flags(No Ack ClusQ )

QLOCAL(CQ1                                         ) 3 Live    Seq(1505322879)
 Offset: @113A0    Clusters: @11290
 ptr(880D2D20) clus(880D2C10)
 QDesc(                                                     )
 UUID(CDRPR3_2017-09-13_13.14.19                     )
 DefBind(0) DefPersistence(0) DefPriority(0) DefPutResponse(1) InhibitPut(0)
 CLWLQueuePriority(0) CLWLQueueRank(0)
 Cluster(CLUSTERDR                                    ) Live    Seq(1505322879)
```

Notes: Viewing the cluster cache, PR 7

Unix example: For the Partial Repository CDRPR7
amqrfdm -m CDRPR7 -d > amqrfdm_CDRPR7.txt

There are 2 highlights:
This PR knows that CQ1 is a clustered queue in PR 3, and knows the cluster sender channel to use.

1) The clustered queue CQ1 has a local in CDRPR3
```
QLOCAL(CQ1                                            )
 QDesc(                              )
 UUID(CDRPR3_2017-09-13_13.14.19                    )
```

2) To connect to CDRPR3 it is necessary to use the channel: TO.CDRPR3
```
Qm(CDRPR3                                        ) Live    Seq(1505574920)
 Channel(TO.CDRPR3            ) Running      ChlSeq(11) DestSeqFactor(0)
 CLWLChannelRank(0) CLWLChannelPriority(0) CLWLChannelWeight(50)
 XmitQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE                       )
 Conname('angelito.raleigh.ibm.com(1433)')
 Desc(                              )
 UUID(CDRPR3_2017-09-13_13.14.19                    )
 QMFlags(270: CLUSSDR Auto Joined InUse Refresh )
 Product: MQMM    Version: 09000300
 Cluster(CLUSTERDR                            ) Live    Seq(1505574920)
 QMFlags(30: CLUSSDR Auto Joined )
```

Now let's add a clustered topic

Let's define **a clustered topic object in PR 4** and **a subscriber in PR 7**

Notice that at this point, there are no cluster channels between these 2 PRs.

A cluster sender channel will be added automatically between these 2 PRs…
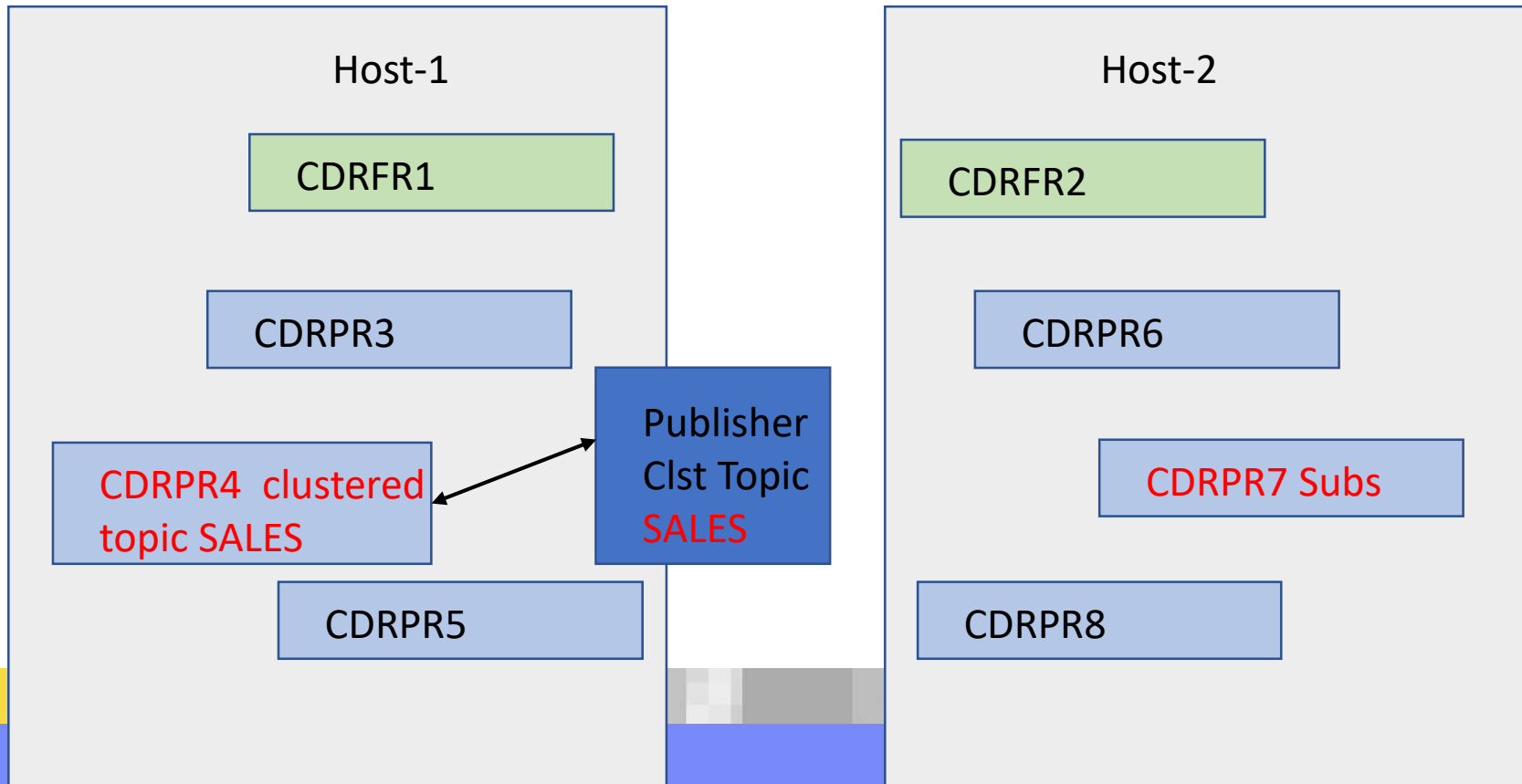but (spoiler alert!) a side effect is that: similar channels will be added between ALL the PRs!

Pub Topic in PR 4 and subscriber in PR 7

PR 4 will have a Clustered Topic and a Publisher
PR 7 will have a Subscriber
At this point, there is NO channel between them.

Host-1	Host-2
CDRFR1	CDRFR2
CDRPR3	CDRPR6
CDRPR4 clustered topic SALES ←→ Publisher Clst Topic SALES	CDRPR7 Subs
CDRPR5	CDRPR8

Now let's add a clustered topic

Adding a clustered topic in PR 4:

DEFINE TOPIC(SALES) TOPICSTR('sales') CLUSTER(CLUSTERDR)

Notice default: CLROUTE(DIRECT) => direct routing cluster

```
display topic(SALES)
     1 : display topic(SALES)
AMQ8633I: Display topic details.
    TOPIC(SALES)                      TYPE(LOCAL)
    TOPICSTR(sales)                   DESCR( )
    CLUSTER(CLUSTERDR)                CLROUTE(DIRECT)
    DURSUB(ASPARENT)                  PUB(ASPARENT)
    SUB(ASPARENT)                     DEFPSIST(ASPARENT)
    DEFPRTY(ASPARENT)                 DEFPRESP(ASPARENT)
    ALTDATE(2017-09-16)               ALTTIME(15.18.33)
    PMSGDLV(ASPARENT)                 NPMSGDLV(ASPARENT)
    PUBSCOPE(ASPARENT)                SUBSCOPE(ASPARENT)
    PROXYSUB(FIRSTUSE)                WILDCARD(PASSTHRU)
    MDURMDL( )                        MNDURMDL( )
    MCAST(ASPARENT)                   COMMINFO( )
    USEDLQ(ASPARENT)                  CUSTOM( )
```

Both FRs know about clustered topic

Now both FRs know that there is a clustered topic called SALES in CDRPR4

Pubs and Subs can be done from any PR

One feature of distributed Pub/Sub is that you can use Publishers in one or more PRs, and you can use Subscribers in one or more PRs

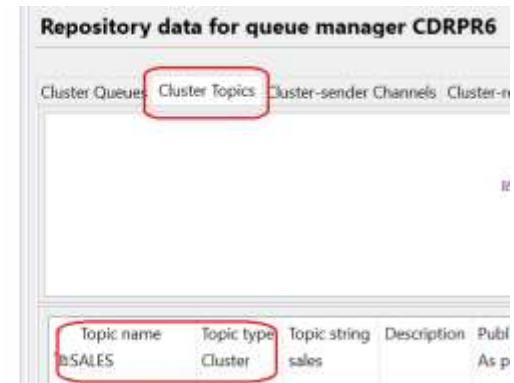In a "**direct route cluster**", **the knowledge about a clustered topic needs to be broadcasted to all PRs.**

In order to provide a minimal direct route between the PRs, **cluster sender channels will be created between all the PRs.**

All PRs know about the Clustered Topic

All the PRs receive notification about
the Clustered Topic SALES:

View of CDRPR6

View of CDRPR8
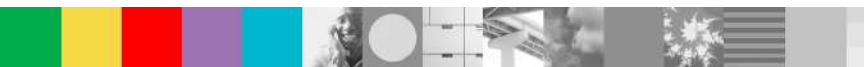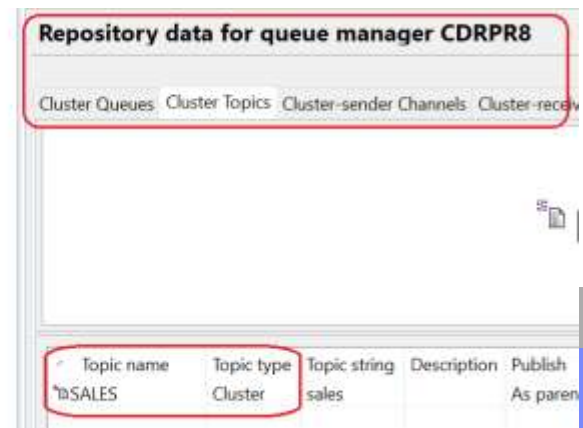
Notes: amqrfdm shows clustered topic

n o t e s

Partial view of the cluster cache for PR 6.

This PR is not going to be used in the scenario, but for that reason it is chosen for this example.

Notice that even though there are no plans to use the clustered topic SALES in this PR, this clustered topic is now included in the cluster cache for this PR.

Why? Because in the future a Publisher or a Subscriber could run in this PR.

** Clustered Topic Object: refers to PR 4

TOPIC(SALES) 2 Live   Seq(1505575405)

  UUID(CDRPR4_2017-09-13_13.15.47)

  DefPersistence(-1) DefPriority(-2) DefPutResponse(0) InhibitPub(0) InhibitSub(0)

  DurableSubs(0) NonPersMsgDlv(0) PersMsgDlv(0) PubScope(1) SubScope(1)

  ProxySub(2) Wildcard(2) Routing(0)

  Cluster(CLUSTERDR) Live   Seq(1505575405)

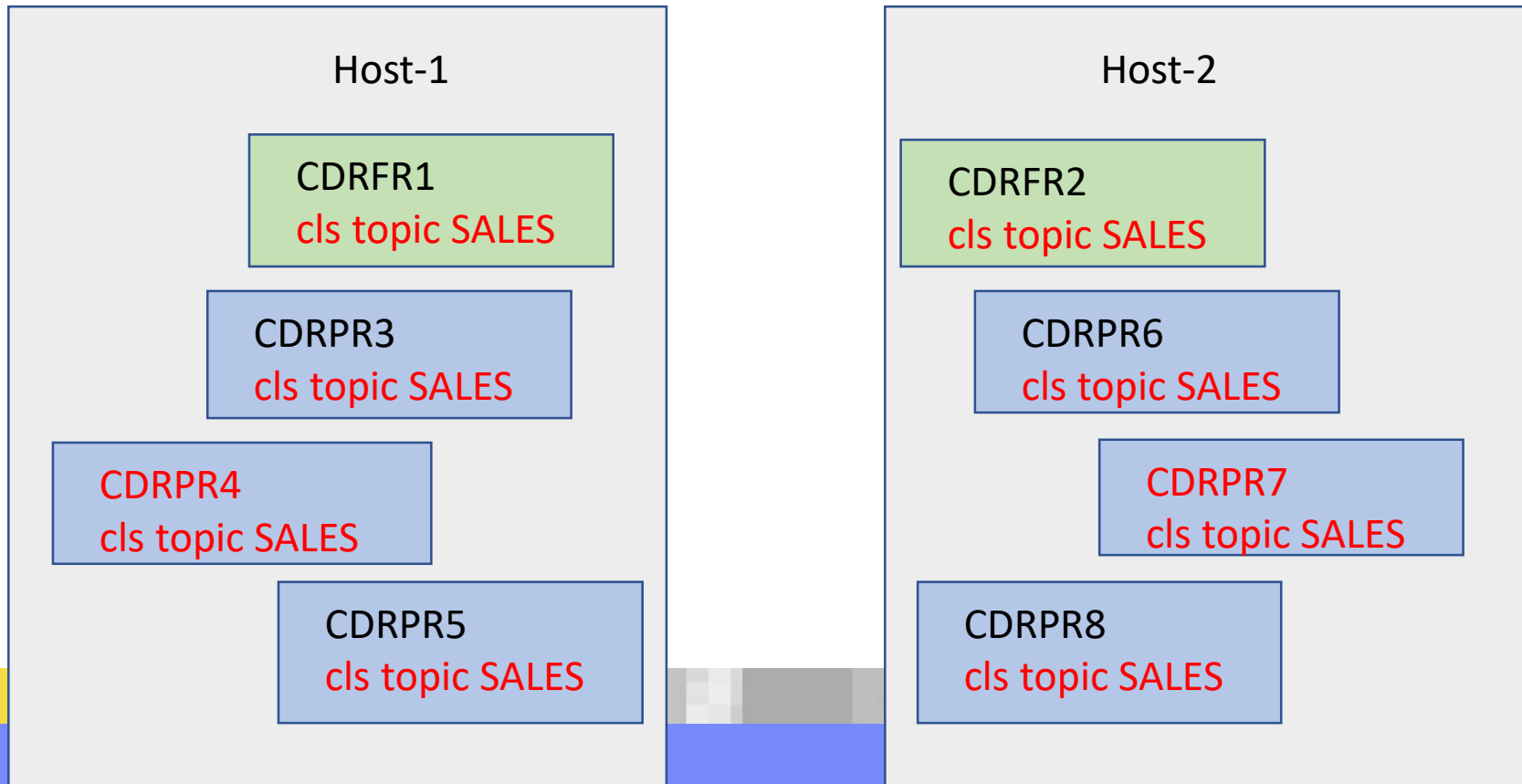** Topic String for the topic object

TOPSTR Topic(SALES)

  UUID(CDRPR4_2017-09-13_13.15.47)

  Str('sales')

All PRs know about the clustered topic

At this point, all the queue managers know the existence of the clustered topic SALES.

Host-1

CDRFR1
cls topic SALES

CDRPR3
cls topic SALES

CDRPR4
cls topic SALES

CDRPR5
cls topic SALES

Host-2

CDRFR2
cls topic SALES

CDRPR6
cls topic SALES

CDRPR7
cls topic SALES

CDRPR8
cls topic SALES

Cluster channels are created for ALL PRs !

In addition, **<u>auto cluster channels are created for all the PRs!!</u>** The following shows CDRPR4

Connections between all PRs

- There is now connection between all PRs.
- **It is called "direct routing cluster" because there is a <u>direct route</u> between all the queue managers.**
- To go from PR 4 to PR 7, no need to go thru PR 3.

Host-1	Host-2
CDRFR1 SALES in CDRPR4	CDRFR2 SALES in CDRPR4
CDRPR3	CDRPR6
CDRPR4 SALES	CDRPR7
CDRPR5	CDRPR8
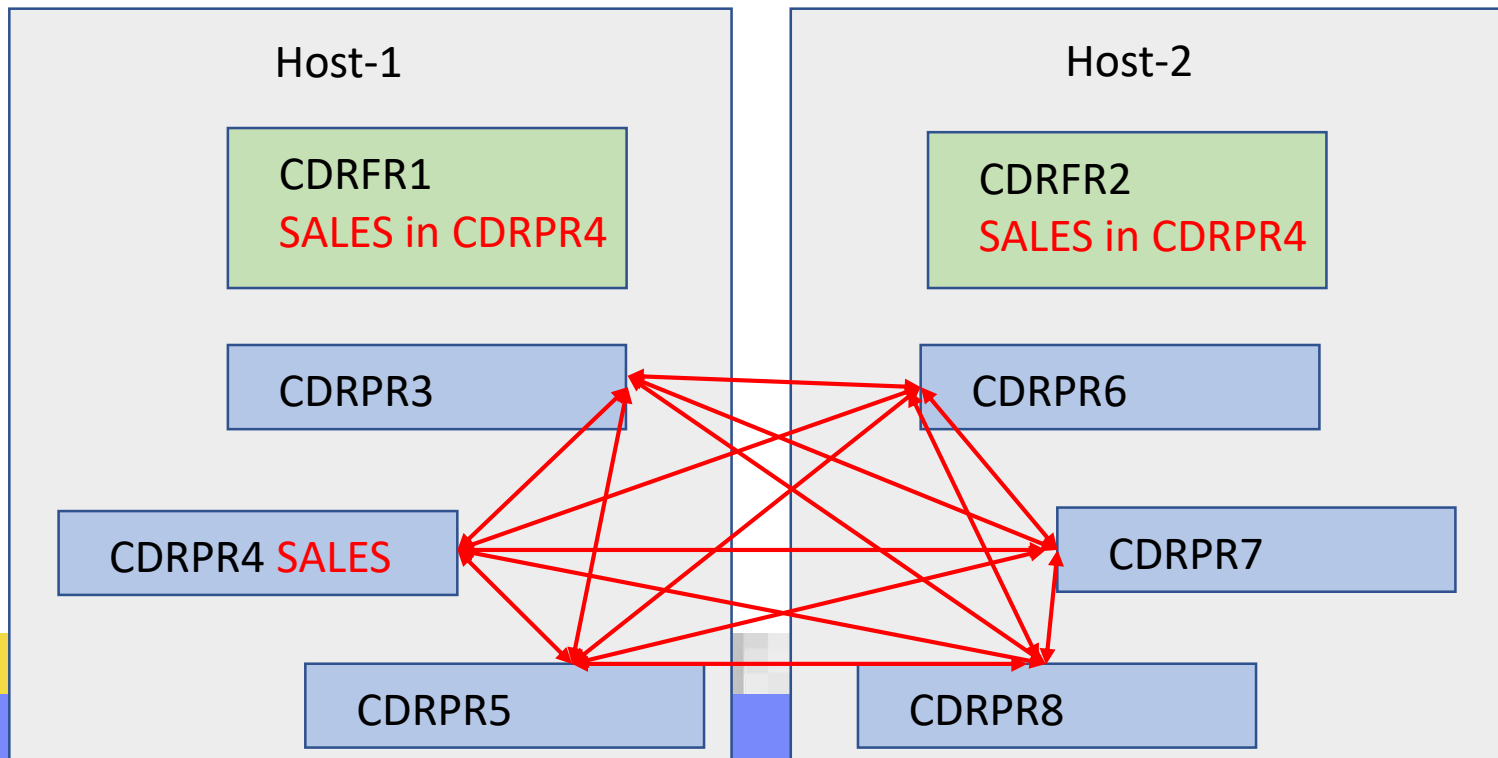
Notes: amqrfdm shows auto cluster-senders

n o t e s

Partial view of the cluster cache for PR 7, showing 2 cluster-senders.

This channel is expected between PR 7 and PR 4 (which has the clustered topic)
Qm(CDRPR4   ) Live   Seq(1505575405)
 Channel(TO.CDRPR4   ) Stopped   ChlSeq(15) DestSeqFactor(0)
 CLWLChannelRank(0) CLWLChannelPriority(0) CLWLChannelWeight(50)
 XmitQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE   )
 Conname('angelito.raleigh.ibm.com(1434)')
Desc(   )
 UUID(CDRPR4_2017-09-13_13.15.47   )
 QMFlags(270: CLUSSDR <u>Auto Joined</u> InUse Refresh)

But notice that other auto cluster-sender channels to other PRs are also created, such as to PR6
Qm(CDRPR6   ) Live   Seq(1505558079)
 Channel(TO.CDRPR6   ) Inactive   ChlSeq(18) DestSeqFactor(0)
 CLWLChannelRank(0) CLWLChannelPriority(0) CLWLChannelWeight(50)
 XmitQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE   )
 Conname('mosquito.raleigh.ibm.com(1436)')
Desc(   )
 UUID(CDRPR6_2017-09-13_07.39.27   )
 QMFlags(270: CLUSSDR <u>Auto Joined </u>InUse Refresh)

Create subscriber in PR 7

Let's create a subscriber SUBSALES in PR 7, which uses a provided queue.

DEFINE QLOCAL(QS2)

DEFINE SUB('SUBSALES2') TOPICSTR('sales') +
DESTCLAS(PROVIDED) DEST(QS2)

Create subscriber in PR 7

View in MQ Explorer of ALL the Subscribers for PR7.
Note: You must enable the option to see all
SYSTEM objects!

Publish in PR 4

We will publish in PR 4 and we expect the subscriber in PR 7 to receive a message.

C:\> **amqspub sales CDRPR4**
Sample AMQSPUBA start
target topic is sales
TEST-PUB
Sample AMQSPUBA end

Question: How does PR 4 know that it needs to send a message to a subscriber SUBSALES2 in PR 7?

Proxy Subscribers

When a subscriber is created for a clustered topic, each queue manager will broadcast to the other queue managers an entity called:

a proxy subscriber

This Proxy Subscriber will specify:

Topic String: sales

Destination queue manager: CDRPR7

Cluster Name: CLUSTERDR

All PRs know about the Proxy Subscriber

At this point, all the queue managers know the existence of the proxy subscriber in PR 7 (in addition to the clustered topic SALES)

Host-1

CDRFR1
cls topic SALES
proxy sub PR 7

CDRPR3
cls topic SALES
proxy sub PR 7

CDRPR4
cls topic SALES
proxy sub PR 7

CDRPR5
cls topic SALES
proxy sub PR 7

Host-2

CDRFR2
cls topic SALES
proxy sub PR 7

CDRPR6
cls topic SALES
proxy sub PR 7

CDRPR7
cls topic SALES
local sub PR 7

CDRPR8
cls topic SALES
proxy sub PR 7
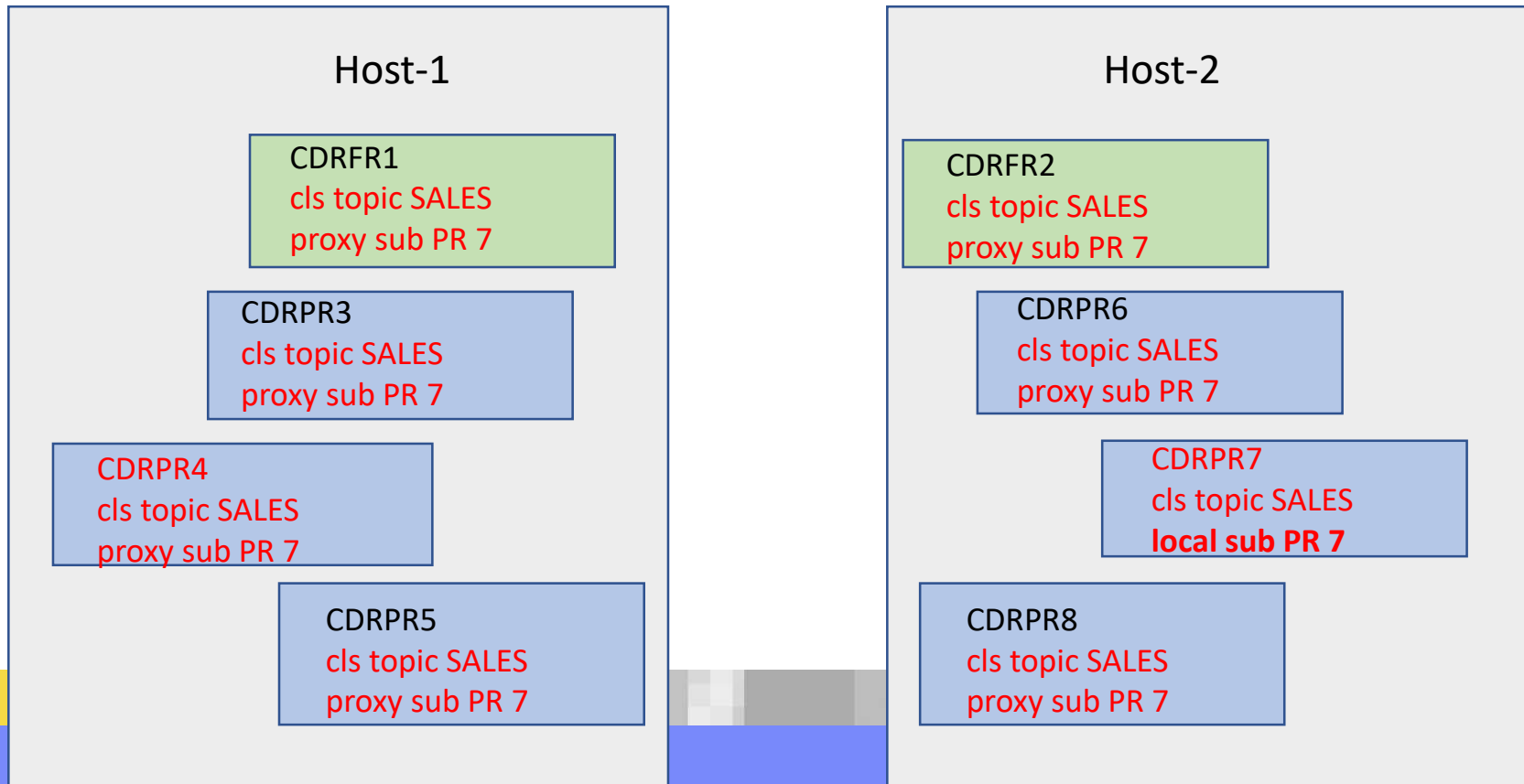
Proxy subscriber in PR 4 pointing to PR 7

Note: Enable to see the SYSTEM objects

Name: **SYSTEM.PROXY.CDRPR7 CLUSTERDR sales**

Topic String: sales

Destination name: SYSTEM.INTER.QMGR.PUBS

Destination queue manager: CDRPR7

Type: Proxy

IBM MQ
- Queue Managers
  - CDRFR1
  - CDRFR2 on 'mosquito.raleigh.ibm.com'
  - CDRPR3
  - CDRPR4
    - Queues
    - Topics
    - Subscriptions
    - Channels

Subscriptions

Filter: Standard for Subscriptions

	Subscription name	Topic name	Topic string
	CDRPR4 SYSTEM.BROKER.INTER.BROKER.CO...	SYSTEM.B...	SYSTEM.BR...
	SYSTEM.DEFAULT.SUB		
	SYSTEM.PROXY.CDRPR7 CLUSTERDR sales		sales

Composite view
(right columns) ➔

Subscriptions

Filter: Standard for Subscriptions

	Destination name	Destination queue manager	Type
e	SYSTEM.BROKER.INTER.BR...	CDRPR4	API
			Admin
	SYSTEM.INTER.QMGR.PUBS	CDRPR7	Proxy

Subscriber in PR 7 receives

Because of Proxy Subscriber, the published message is routed to PR 7, using the auto cluster sender channel between PR 4 and PR 7.

Message is received by subscriber in PR 7 (stored in provided queue QS2)

Notes: message received in Subs in PR 7

Confirming receipt in PR 7 of the published message in PR 4 (notice that ReplyToQMgr indicates the queue manager of origin, in this case CDRPR4).

mqm@mosquito: /home/mqm
$ amqsbcg QS2 CDRPR7
AMQSBCG0 - starts here
****

 MQOPEN - 'QS2'
 MQGET of message number 1, CompCode:0 Reason:0
****Message descriptor****
  StrucId : 'MD ' Version : 2
  Report   : 0 MsgType : 8
  Expiry   : -1 Feedback : 0
 Format : 'MQSTR   '
  Priority : 0 Persistence : 0
  MsgId : X'414D5120434452505237202020202020C400BD599BBC4721'
  CorrelId : X'414D5120434452505237202020202020C400BD59B1104821'
  ReplyToQ   : '   '
  ReplyToQMgr   : 'CDRPR4   '
****   Message   ****
 length - 8 of 8 bytes
00000000: 5445 5354 2D50 5542   '**TEST-PUB**   '

Channel active between PR 4 and PR 7

Notice that the channel between PR 4 and PR 7 became automatically active:

Topic status has indicators for proxy subs

The "Topic Status" view has indirect indicators of proxy subs.
For example, PR 6 does not have local Topic objects nor subscribers. But it knows about topic 'sales' and that there is a Subscriber (the one in PR 7).

Notes: subscription with subtype(PROXY)

PR 6 was not involved in the scenario (it has no publishers and no subscribers), but has "proxy subscriptions". In PR 6, we can issue the following command in runmqsc to show them:

<u>display sub(*) subtype(proxy) all</u>
AMQ8096I: IBM MQ subscription inquired.
  SUBID(414D51204344425250532362020202020203FFEBC596E217424)
  SUB(SYSTEM.PROXY.CDRPR7 CLUSTERDR sales)
  TOPICSTR(sales)   TOPICOBJ( )
  DISTYPE(RESOLVED)   DEST(SYSTEM.INTER.QMGR.PUBS)
  DESTQMGR(CDRPR7)   PUBAPPID( )
  SELECTOR( )   SELTYPE(NONE)
  USERDATA( )
  PUBACCT(0335303100000000000000000000000000000000000000000000000000000006)
  DESTCORL(000000000000000000000000000000000000000000000000)
  DESTCLAS(PROVIDED)   DURABLE(YES)
  EXPIRY(UNLIMITED)   PSPROP(MSGPROP)
  PUBPRTY(ASPUB)   REQONLY(NO)
  SUBSCOPE(ALL)   SUBLEVEL(1)
  SUBTYPE(PROXY)   VARUSER(FIXED)
  WSCHEMA(TOPIC)   SUBUSER(mqm)

n o t e s

One more test

Let's use a pair of PRs that have not been used yet.

PR 5 will have a Publisher on the existing clustered topic SALES

PR 8 will have a **non durable subscriber** (MQ Explorer, by using Test Subscription)

Baseline for PR 5

PR 5 knows about:

clustered topic SALES

Topic Status shows
that there is a Sub
for SALES
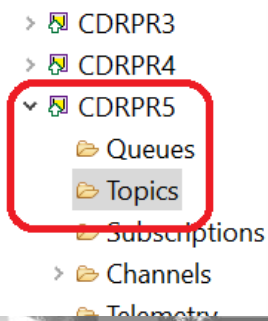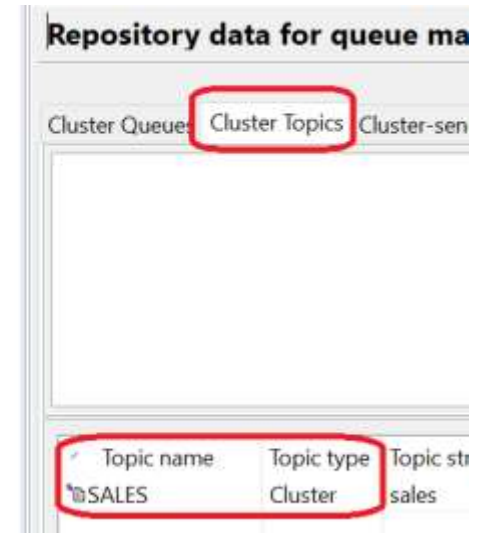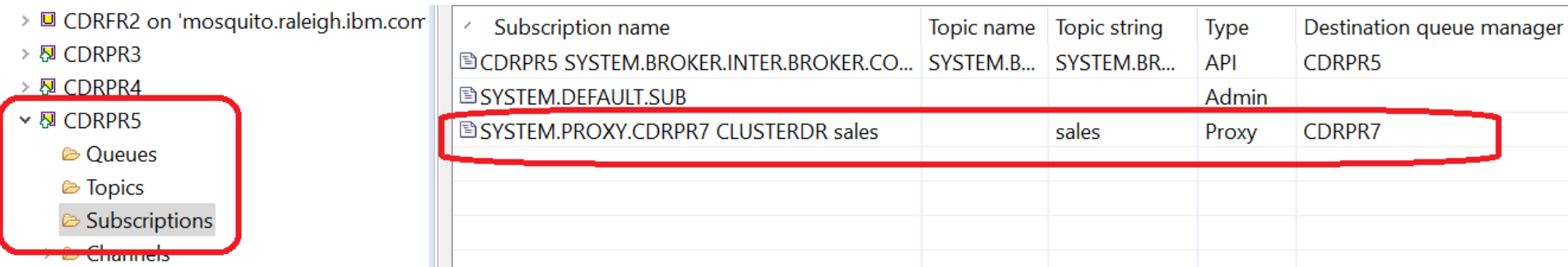
Baseline for PR 5

PR 5 knows about:
There is a Proxy Subscriber for
topic string 'sales' in PR 7

Subscription name	Topic name	Topic string	Type	Destination queue manager
CDRPR5 SYSTEM.BROKER.INTER.BROKER.CO...	SYSTEM.B...	SYSTEM.BR...	API	CDRPR5
SYSTEM.DEFAULT.SUB			Admin	
SYSTEM.PROXY.CDRPR7 CLUSTERDR sales		sales	Proxy	CDRPR7

- CDRFR2 on 'mosquito.raleigh.ibm.com
- CDRPR3
- CDRPR4
- CDRPR5
  - Queues
  - Topics
  - Subscriptions
  - Channels

Baseline for PR 5

PR 5 knows about:

The cluster sender to PR 8 exists, but it is inactive.

Baseline for PR 8

PR 8 knows about:
Clustered Topic SALES

MQ Explorer Test Subscription in PR 8

In PR 8, from Topic Status, select
"Test Subscription" on topic string 'sales'
Queue manager determines that is a clustered topic.
A Proxy Subscriber is sent TO ALL PRs!

MQ Explorer Test Subscription in PR 8

A Proxy Subscriber is sent TO ALL PRs!

Notice that now PR 5 shows a 2nd Proxy Subscriber, the one for PR 8.

Publish in PR 5, message received in PR 8

In PR 5 publish a message (Test Publish).
The queue manager knows from the Proxy Subs that needs to send a message to PR 8.
The subscriber in PR 8 receives it.

Publish in PR 5, message received in PR 8

The cluster sender from PR 5 to PR 8 becomes active.

Closing Test Sub in PR 8, refreshes cluster

Now let's close the Test Subscriber in PR 8.
This will cause a notification sent to all PRs to remove the corresponding Proxy Subscriber.
Notice that PR 5 shows now only 1 Proxy instead of 2.

Potential impact in large, busy clusters

Direct Route Clusters rely on MQ maintaining a high level of sharing of information and connectivity between every queue manager in the cluster.

For small and simple networks (that is, a small number of queue managers, and a fairly static set of publishers and subscribers) this could be acceptable.

However, when used in larger or more dynamic environments the overhead might be prohibitive.

The End

This is the end of the presentation.

THANKS!!