IBM Software Group

WebSphere MQ - Finding Your Way Around Channels on z/OS

WebSphere MQ L2 Team
20 March 2012

WebSphere® Support Technical Exchange

Agenda

- Introduce the presenters
- We will discuss channels as they relate to z/OS Websphere MQ in particular. 4 different aspects
  - Channel pairs, uniqueness of channel types
  - Channel parameters and best practices of those parameters
  - What can be learned from examining the CHIN log? In particular, in error conditions
  - Channels and the underlying transport.

WMQ - L2 Presenters

L2 Presenter	Topic
Mark Womack	General discussion of channels
Kenneth Langhorne	Channel parameters
Beverly Kingsley	Channel Initiator joblog / jeslog
Rick Armstrong	Channels and networking

Compatible Channel Types

- Sender with receiver
- Requester with server
- Requester with sender (for callback)
- Server with receiver (server used as a sender)
- Client-connection with server-connection
- Cluster-sender with cluster-receiver

Categories : Message and MQI

- Two types of MQI Channels
  - Client connection and Server connection
    - Bidirectional used to transfer MQI calls/responses
- Six types of Message Channels
  - Sender, Receiver, Server, Requester, Cluster Sender, Cluster Receiver
    - Unidirectional since messages go in one direction

SVRCONN and CLNTCONN

- Client Connection channel
  - Server Connection channel
    - Connection to single QMGR
    - Multiple instances allowed
    - No auto-define SVRCONN on z/OS
  - How should client connect to QSG
    - Back end availability
    - Front end availability

What's so funny about cluster channels

- REFRESH CLUSTER REPOS(YES)
  - Partials can't find the FR anymore
- CLUSSDR knows what CLUSRCVR knows
- The more full repositories the better, not !
  - Partials subscribe to
- Beware of message reallocation
  - Is my channel really hung or working as designed
- The effect of disaster recovery
  - A rose by any other name is a good idea

Server Channel

- Server channel can connect to remote receiver or requester
  - Receiver always passive so server initiates connect
  - Requester active at start then acts as receiver
  - Server can also initiate server-requester connect
    - If server knows remote connection name
- Server-Receiver vs. Sender-Receiver [if CONNAME present]
  - Fully-qualified servers synonymous to SDR-RCVR
  - Connections only initiated from non-RCVR end
  - Messages flow from SDR/SVR to RCVR

So why use SVR channels

- SVR-RCVR
  - No real advantage over SDR-RCVR pair
  - But if tied to RQSTR then dynamic host/IP can change because SVR CONNAME is optional
    - and the RQSTR initiates the connection
- Callback
  - RQSTR starts channel, SDR terminates it
  - SDR re-initiates connect then sends msgs from xmitQ

Parameters for Channels

- The channel parameters that I normally work with for channel concerns are normally either under the ALTER QMGR or the DEFINE CHANNEL command in the Information Center for 7.1

- Welcome to the IBM® WebSphere MQ Information Center
- The IBM(R) WebSphere(R) MQ Information Center contains the documentation for WebSphere MQ Version 7.1.

- WebSphere MQ > Reference > Administration reference > MQSC reference > The MQSC commands
- ALTER QMGR

- WebSphere MQ > Reference > Administration reference > MQSC reference > The MQSC commands
- DEFINE CHANNEL

- http://publib.boulder.ibm.com/infocenter/wmqv7/v7r1/index.jsp?topic=%2Fcom.ibm.mq.doc%2Fsc10600_.htm

Parameters to help channels

- ADOPTMCA -Specifies whether an orphaned instance of an MCA restarts immediately when a new inbound channel request matching the ADOPTCHK parameter is detected: .
- KAINT KeepAlive Interval - Specifies whether an orphaned instance of an MCA restarts immediately when a new inbound channel request matching the ADOPTCHK parameter is detected.
- RCVTIME Receive Time - The approximate length of time that a TCP/IP channel waits to receive data, including heartbeats, from its partner before returning to the inactive state. This parameter applies only to message channels and not to MQI channels
- TCPKEEP -Specifies whether the KEEPALIVE facility is to be used to check that the other end of the connection is still available. If it is unavailable, the channel is closed

Parameters to help channels

- BATCHLIM - The minimum amount of time, in milliseconds, that a channel keeps a batch open.
- NPMSPEED - The class of service for nonpersistent messages on this channel: FAST| NORMAL
- USEDLQ - This attribute determines whether the dead-letter queue (or undelivered message queue) is used when messages cannot be delivered by channels. NO| YES
- RESETSEQ - Pending reset sequence number. This is the sequence number from an outstanding request and it indicates a user RESET CHANNEL command request is outstanding.

Additional Channel Links

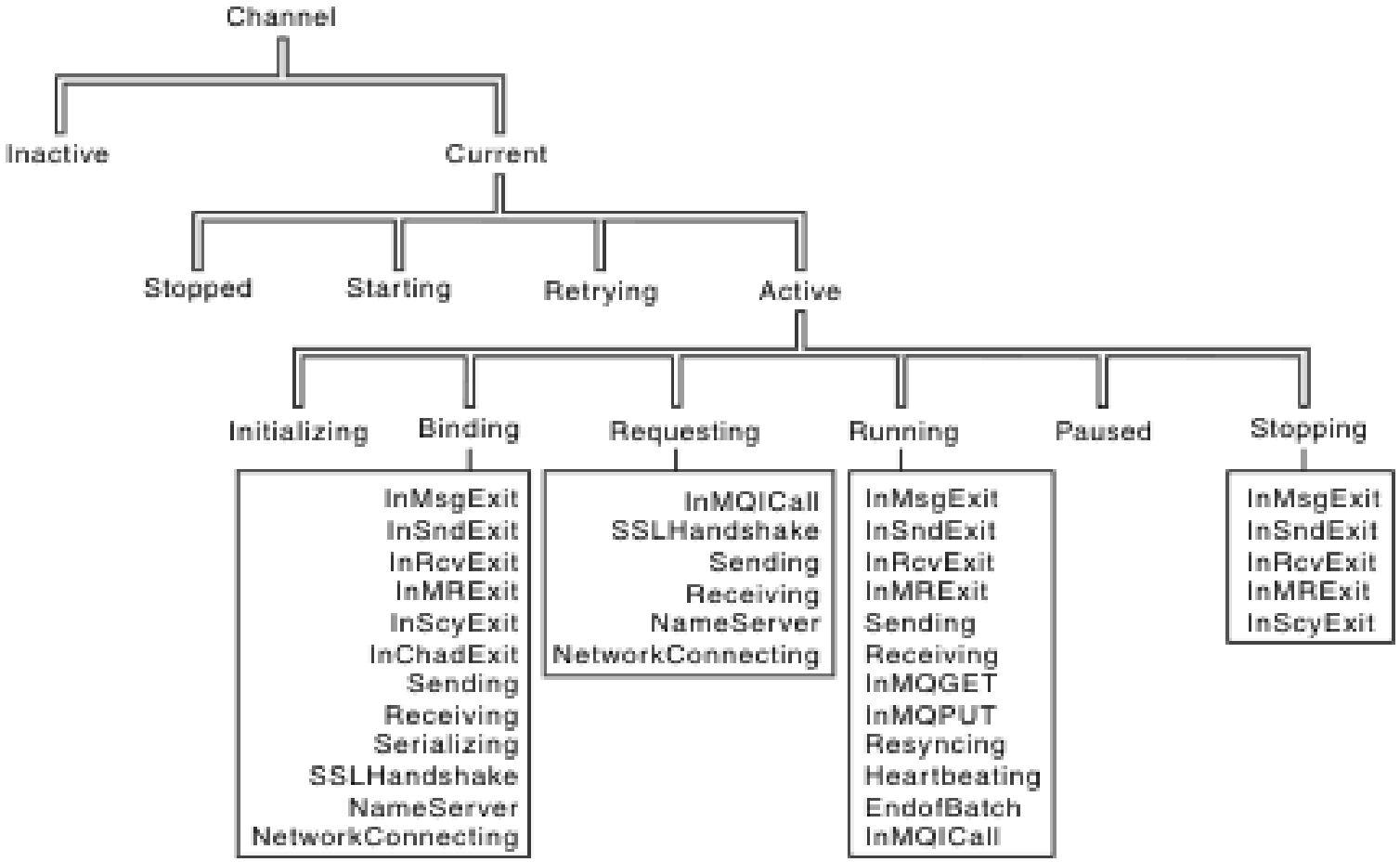
WebSphere MQ > Troubleshooting and support > Dealing with problems
Resolving problems with channels and DQM

WebSphere MQ > Configuring > Connecting applications using distributed queuing > Introduction to distributed queue management > Channel control function
In-doubt channels

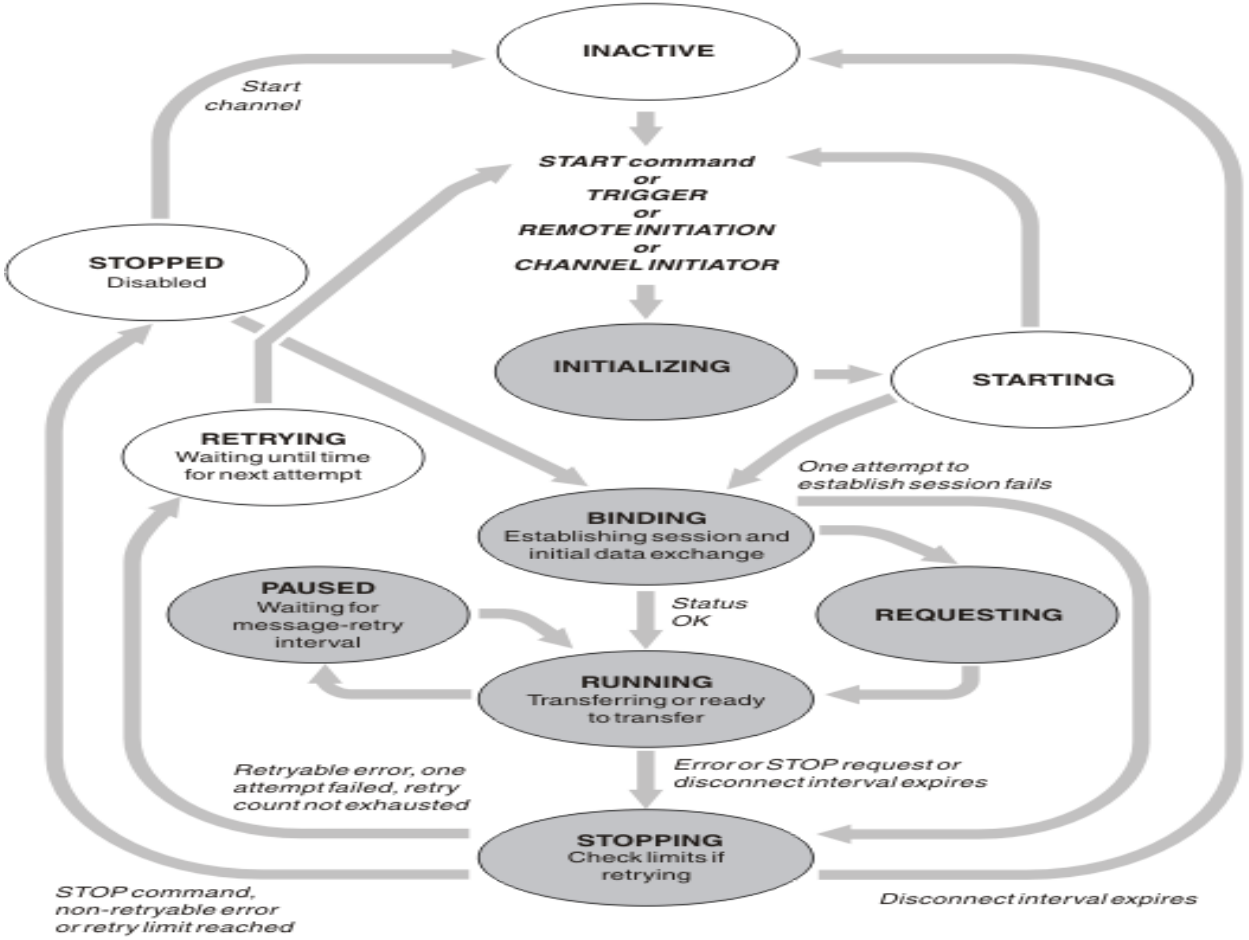
WebSphere MQ > Configuring > Connecting applications using distributed queuing > Introduction to distributed queue management > Channel control function
Channel states

http://publib.boulder.ibm.com/infocenter/wmqv7/v7r1/index.jsp

Channel States



Current and Active



Channel Initiator

- Distributed queuing – component of WMQ
  - Channel Initiator address space (z/OS)
  - Messages issued to the CHIN joblog
  - Almost all messages are CSQXnnny
    - 'X' is the component letter that represents the CHINIT
    - 'y' is one of: Error, Information
  - Channel Initiator Error messages
    - CSQX***E

CHINIT Joblog

- CHINIT initialization startup messages and parameters
  - CSQX000I - Version of MQ
  - CSQX001I - Chinit address space is starting
  - CSQX004I - Chinit is using (eg) 182 MB of local storage, (eg) 13338 MB are free
    - From MQ V7; indicates storage problem; similar to QMGR message CSQY220I
  - CSQX022I - Chinit startup complete (Note: CSQINPX CSQU012I on MSTR log)
  - CSQX034I - Chinit stopping because Queue Manager is stopping
  - Messages CSQX070I thru CSQX082I - Chinit parameters on QMGR attributes
  - Messages CSQX090I thru CSQX094I - Chinit parms alterable on QMGR
  - Message CSQX085I - Chinit LU62 QMGR attributes
    - Use ALTER QMGR

CHINIT joblog

- CHINIT error messages
  - CSQX20nE - An error in the transport layer; usually not an MQ error
    - Csect name, ConnId, channel name, return code, reason code
  - CSQX40nE - An error associated with clustering
  - CSQX50nE - Channel errors; various problems with channels
  - Most CSQX60nE - An error in System SSL (reported errors / return codes)
    - Usually not an MQ issue
- Finding MQ error messages in joblog / syslog
  - L2 has a REXX exec that finds 'blocks' of data line that contain a
  - search string - it can be used to extract all the 'error' messages
- while in ISPF browse/view/edit of a Joblog/Syslog file:
  - TSO %mqfindl CSQ****E .* (Request via email if you want a copy)

CHINIT line trace

- Wrap-around line trace for each channel
  - Included in the CHINIT address space
  - Kept in 4 KB buffers
    - Always active; cannot turn it off
    - Applies to TCP/IP and LU6.2 channels
    - Reduce the number of times a communication trace is required
  - Two ways to capture it:
    - Adding CSQSNAP DD to the CHINIT startup JCL
      - Start trace(chinit) class(4) ifcid(202)
      - Display chstatus(channel) saved
    - Take dump of CHINIT address space

Format the dump with the IPCS command
verbx csqxdprd 'subsys=ssid,chst=3

....where 'ssid' is MQ subsys ID

CHINIT line trace

- Contains last INBOUND and OUTBOUND flows for each running channel
- Ideal for channels that appears hung or if the problem is in MQ

```
* Line Trace *
**************
CSQXLTR           000000001F3B9000 Line trace
Line trace: First entry on 20120216
00:51:52.49 : INBOUND  :        54534820 0000002C 018B3000 00000000 | ..... |
                                00000000 00000111 03330000 0000002C | ..... |
                                00000000 00000000 00000000 00000000 | ..... |
                                00000000 00000000 00000000 00000000 | ..... |
                                00000000 00000000 00000000 00000000 | ..... |
                                00000000 00000000 00000000 00000000 | ..... |
                                00000000 00000000 00000000 00000000 | ..... |
                                00000000 00000000 00000000 00000000 | ..... |
                                20202020 20202020 20202020 20202020 | ..... |
                                20202020 20202020 414D512E 2A202020 | ..... |

00:51:52.49 : OUTBOUND :        54534820 0000002C 019B3000 00000000 | ..... |
                                00000000 00000111 03330000 0000002C | ..... |
                                00000000 00000000 00000000 00000000 | ..... |
                                00000000 00000000 00000000 00000000 | ..... |
                                00000000 00000000 00000000 00000000 | ..... |
                                00000000 00000000 00000000 00000000 | ..... |
                                00000000 00000000 00000000 00000000 | ..... |
```

CHINIT FFST

- First Failure Support Technology / FFST
  - Immediate notification under specific error conditions
  - CSQX053E issued at the same time indicating error info written to CSQSNAP data set
  - Snap dump FFST provides:
    - Customized dump
    - Symptom strings
    - Symptom records
  - Need CSQSNAP DD statement in the CHINIT startup JCL
    - Otherwise CSQX012E: csect-name Unable to open ddname data set
    - Some problems may be intermittent; so we recommend you include:
      - CSQSNAP DD to capture the failure at the time it occurs

CSQX526E

- CSQX526E: csect-name Message sequence error for channel channel-name, sent=msg-seqno expected=exp-seqno

Explanation

The local queue manager does not agree with the remote end on the next message sequence number for channel channel-name. The message is normally issued at both the sending and receiving end: at the sending end, msg-seqno and exp-seqno are unpredictable; at the receiving end, a message had sequence number msg-seqno but sequence number exp-seqno was expected.

System programmer response

Determine the cause of the inconsistency. It could be that the synchronization information has become damaged, or has been backed out to a previous version. If the problem cannot be resolved, the sequence number can be reset manually at the sending end of the channel using the RESET CHANNEL command. (For some queue managers, it may be necessary to issue the RESET CHANNEL command at the receiving end as well.)

CSQX526E contd

- Typically CSQX526E can be received if:
  - Multiple SENDER channels reference the same instance of a RECEIVER channel.
  - The SYSTEM.CHANNEL.SYNCQ contains invalid or duplicate messages (damaged).
  - If the expected sequence number is 1:
    - then the receiver has created a new SYNCQ Message as it did not recognise
    - SYNCQ cleared before a migrated version of MQ was brought up
  - Remedies action you can use to fix it:
    - Use the RESET CHANNEL command to reset the sequence number
    - Ensure that only one unique sending channel references a remote RECEIVER.
    - Use utility CSQ4SUTL to "clean up" redundant information from the SYSTEM.CHANNEL.SYNCQ.
    - CSQ4SUTL can be obtained from L2, and used under the guidance of, the Support Center.

CSQX526E documentation needed

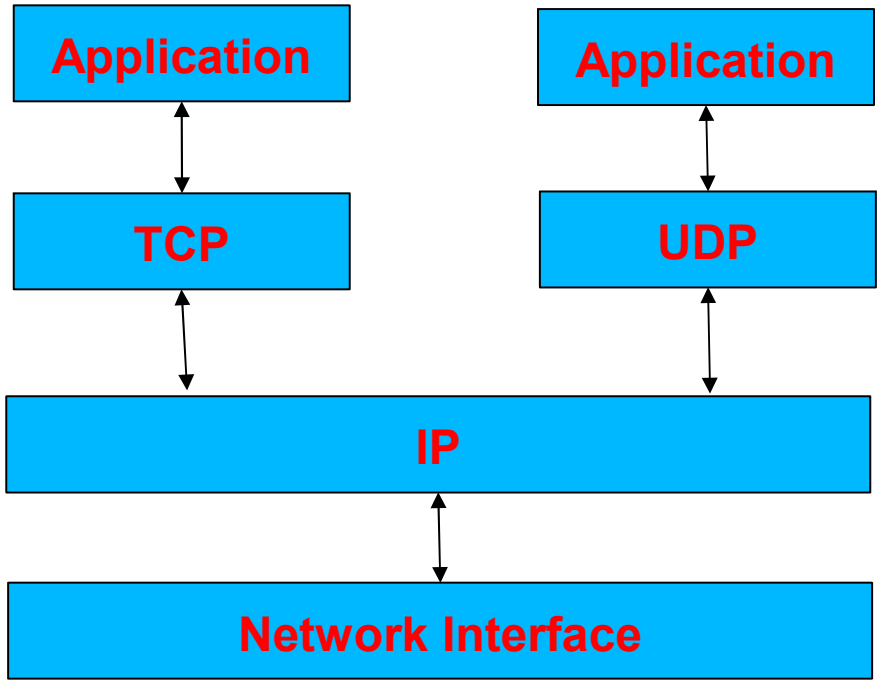
- Turn on MSTR and CHIN tracing by either
  - Setting TRACSTR and TRAXSTR to YES
  - Issuing command:
    - START TRACE(GLOBAL) DEST(RES) CLASS(*) RMID(*)

- SLIP trap on the CSQX526E message as follows:

```
SLIP SET,MSGID=CSQX526E,
JOBNAME=ssidCHIN,A=SVCD,
JOBLIST=(ssidMSTR,ssidCHIN),
DSPNAME=('ssidCHIN'.CSQXTRDS),
SDATA=(CSA,RGN,PSA,SQA,LSQA,TRT,SUM),
MATCHLIM=1,END
```

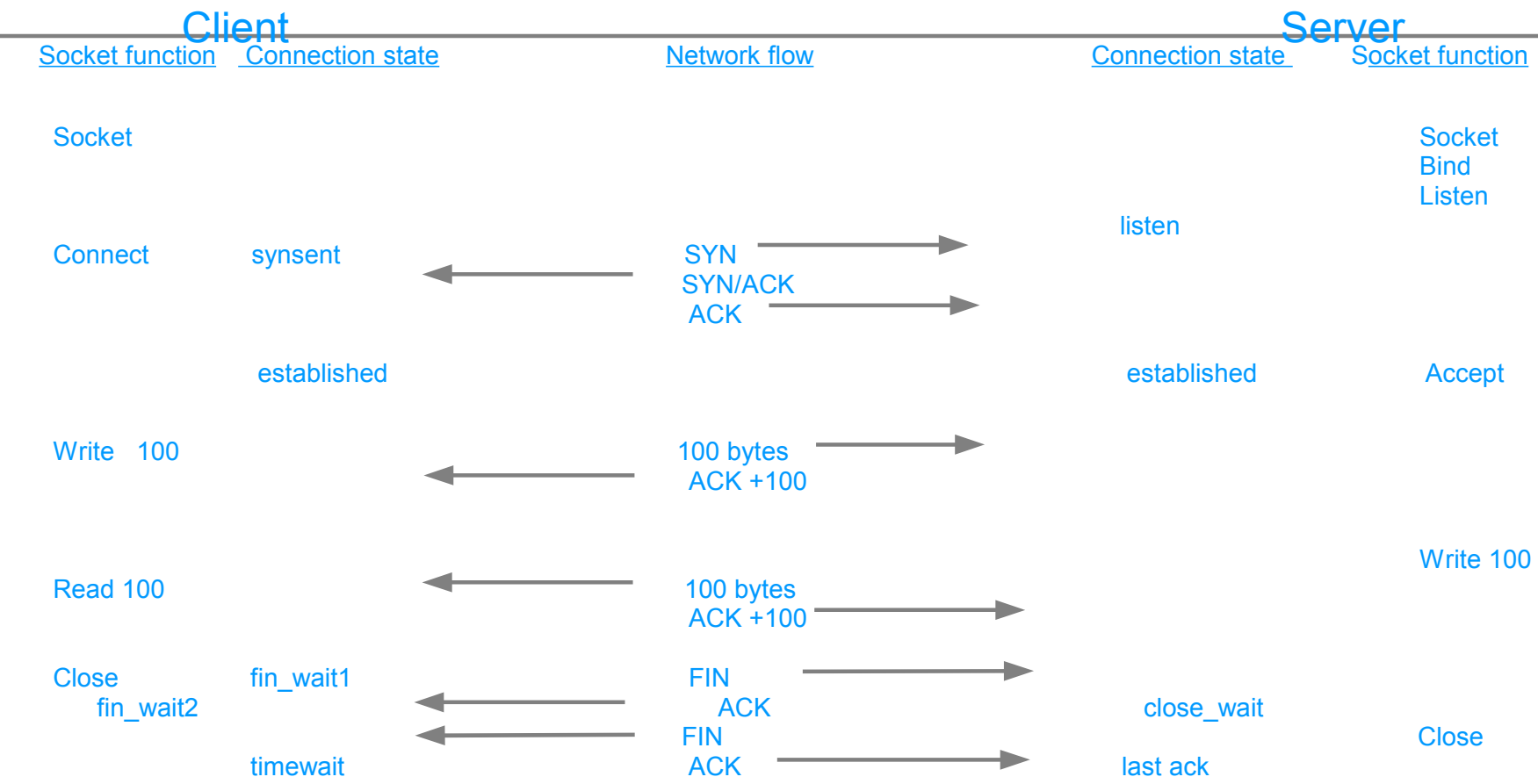
Channels and the transport

- TCP/IP or LU6.2 (APPC)?
- Majority of PMRs use TCP/IP transport



TCP/IP socket overview

Simple example of application socket calls client/server



Domain Nameserver issues

- CHIN will attempt a getnameinfo() call to associate client IP address to a hostname
- Delays in DNS response can cause CSQX014E and other problems
- Task is single threaded

- This is a DNS problem...
- Check Resolver settings to see what DNS(s) are configured
- Check RESOLVERTIMEOUT setting. 30 sec default
- Bypass
  - V6 – PK79874/UK47580
  - V7 – PK88882/UK47582
    - Requires Service parm

Domain Nameserver issues (cont)

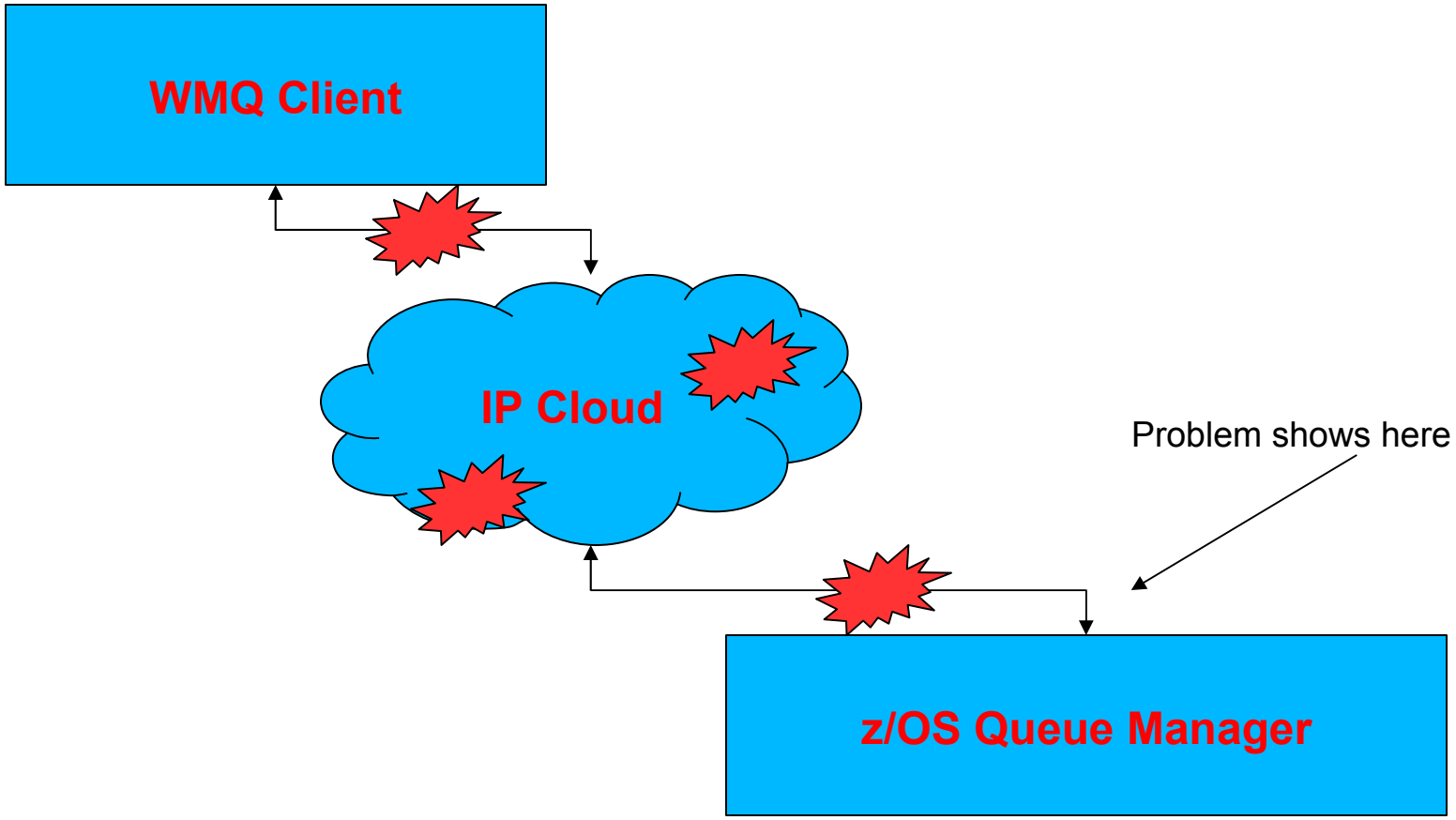
- Other causes of channels in SUBSTATE(NAMESERVER)
  - Delays opening and closing channels
  - Many CSQX500I and CSQX501I messages
- Ensure that no applications intercept WTO messaging.
- See Technotes:
  - http://www.ibm.com/support/entry/portal/Overview/Software/WebSphere/WebSphere_MQ
  - 1272913
  - 1585797
- Suppression of CSQX500I and CSQX501I
  - Requires a service parameter

Channel connection reset

- +CSQX208E +RTPH CSQXRESP Error receiving data, channel TEST.CHAN, connection 9.1.1.1 (9.1.1.1) (queue manager ????) TRPTYPE=TCP RC=00000461 (ECONNRESET) reason=00000000

- CSQX206E RTPH CSQXRCTL Error sending data, channel TEST.CHAN, connection raleigh (9.1.1.1) (queue manager RTPH) TRPTYPE=TCP RC=0000008C reason=76697242
  - 76697242 = JRCONNTCBNOTFOUND

Channel connection reset

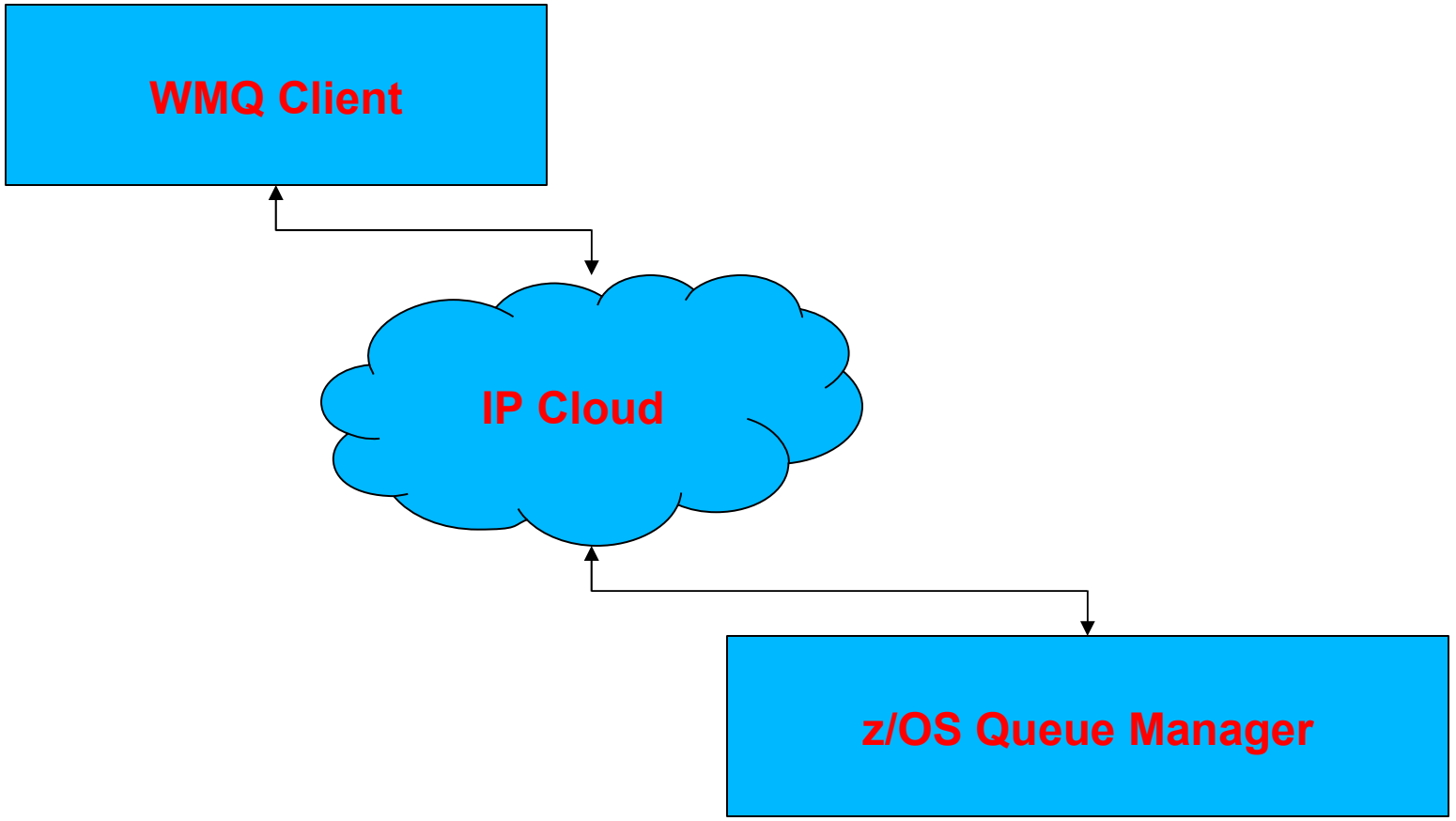


Channel connection reset

- The problem is reported on z/OS
- The reset is not generated from the reporting host channel
- The reset could come from the other host
- The reset could come from any intermediate firewalls

- May need to trace on both sides of the channel connection
- May need to trace intermediate hops along the connection

TCP/IP network overview



Summary

- URL Links to keep handy:
- WebSphere MQ Support Portal
  http://www.ibm.com/support/entry/portal/Overview/Software/WebSphere/WebSphere_MQ
- Channel Mustgather documents:
  - Channels
    http://www.ibm.com/support/docview.wss?rs=171&uid=swg21292922
  - Shared Channels
    http://www.ibm.com/support/docview.wss?rs=171&uid=swg21243378
  - Triggered Channels
    http://www.ibm.com/support/docview.wss?rs=171&uid=swg21291069

References and Useful links

- WebSphere MQ Library at:
  http://www.ibm.com/software/integration/wmq/library/
- Submit PMRs electronically using the SR tool at:
  http://www.ibm.com/software/support/probsub.html
- Migration PTFs:
  http://www.ibm.com/support/docview.wss?rs=171&uid=swg27006519
- You can search for known problems at the WebSphere MQ Support web site:
  http://www-947.ibm.com/support/entry/portal/Overview/Software/WebSphere/WebSphere_MQ
- Technote 1201593 - PSP Buckets - How to find them on the Web:
  http://www.ibm.com/support/docview.wss?rs=171&uid=swg21201593
- System Requirements for WebSphere MQ V7.0/7.0.1 on z/OS
  http://www.ibm.com/support/docview.wss?rs=171&uid=swg27011919
- Upgrade MQMESA, Subset HMS7010
  http://www.ibm.com/support/docview.wss?rs=171&uid=isg1_MQMESA_HMS7010

Additional WebSphere Product Resources

- Learn about upcoming WebSphere Support Technical Exchange webcasts, and access previously recorded presentations at:
  http://www.ibm.com/software/websphere/support/supp_tech.html
- Discover the latest trends in WebSphere Technology and implementation, participate in technically-focused briefings, webcasts and podcasts at:
  http://www.ibm.com/developerworks/websphere/community/
- Join the Global WebSphere Community:
  http://www.websphereusergroup.org
- Access key product show-me demos and tutorials by visiting IBM Education Assistant:
  http://www.ibm.com/software/info/education/assistant
- View a webcast replay with step-by-step instructions for using the Service Request (SR) tool for submitting problems electronically:
  http://www.ibm.com/software/websphere/support/d2w.html
- Sign up to receive weekly technical My Notifications emails:
  http://www.ibm.com/software/support/einfo.html

Connect with us!

1. **Get notified on upcoming webcasts**
   Send an e-mail to wsehelp@us.ibm.com with subject line “wste subscribe” to get a list of mailing lists and to subscribe
2. **Tell us what you want to learn**
   Send us suggestions for future topics or improvements about our webcasts to wsehelp@us.ibm.com
3. **Be connected!**
   Connect with us on Facebook
   Connect with us on Twitter

Questions and Answers