

Unit 13. WebSphere MQ clients

What this unit is about

This unit covers the ways WebSphere MQ clients can attach to a WebSphere server.

What you should be able to do

After completing this unit, you should be able to:

- Describe the various ways to connect a client to a queue manager
- Analyze client connection requirements in order to determine the best approach
- Explain the limitations of various client connection methods
- Propose methodologies to ensure security with client connections

How you will check your progress

- Checkpoint exercise and lab work

Unit objectives

After completing this unit, you should be able to:

- Describe the various ways to connect a client to a queue manager
- Analyze client connection requirements in order to determine the best approach
- Explain the limitations of various client connection methods
- Propose methodologies to ensure security with client connections

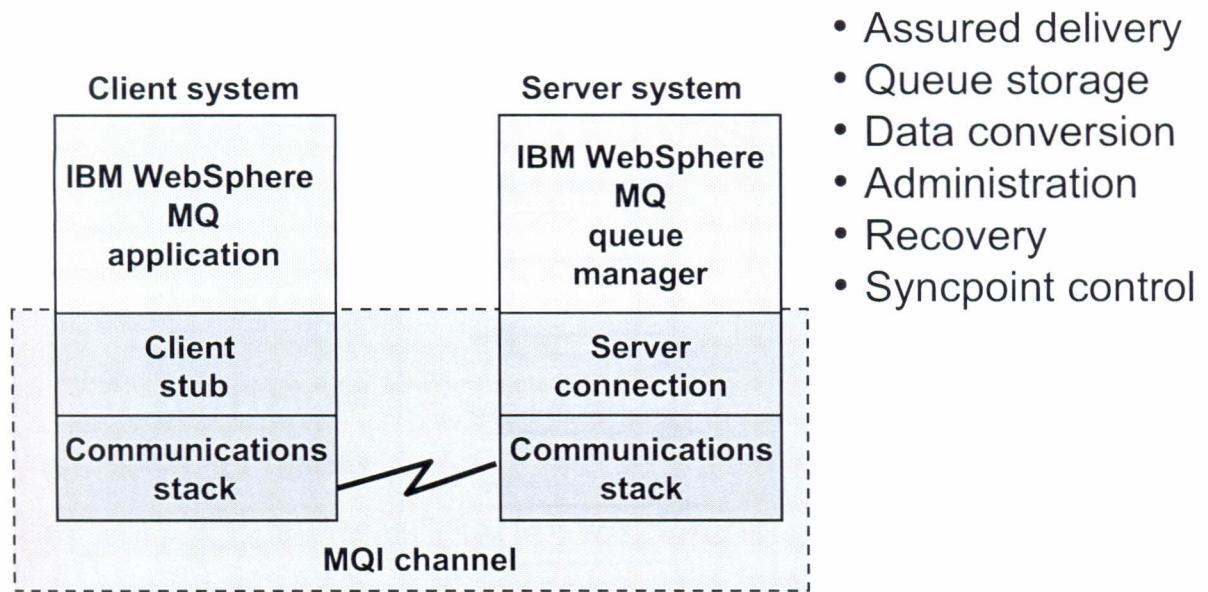
© Copyright IBM Corporation 2008

Figure 13-1. Unit objectives

WM203 / VM2032.0

Notes:

WebSphere MQ client



© Copyright IBM Corporation 2008

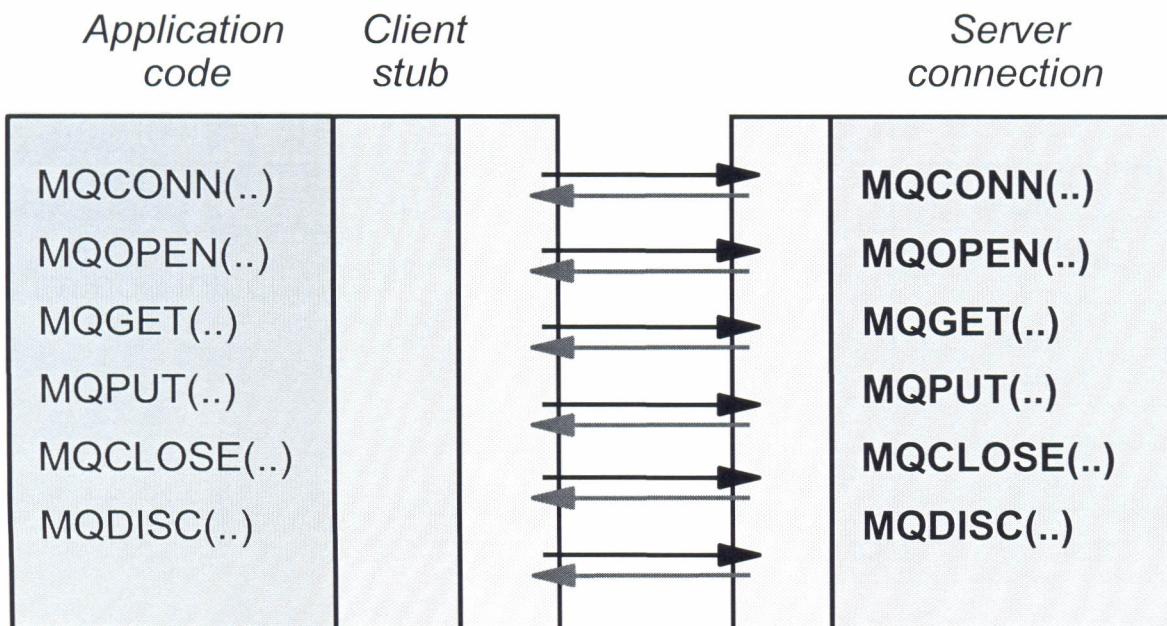
Figure 13-3. WebSphere MQ client

WM203 / VM2032.0

Notes:

/M2032.0

WebSphere MQ clients explained



© Copyright IBM Corporation 2008

Figure 13-4. WebSphere MQ clients explained

WM203 / VM2032.0

Figure 13-

Notes:

The full range of MQI calls and options is available to a WebSphere MQ client application, including:

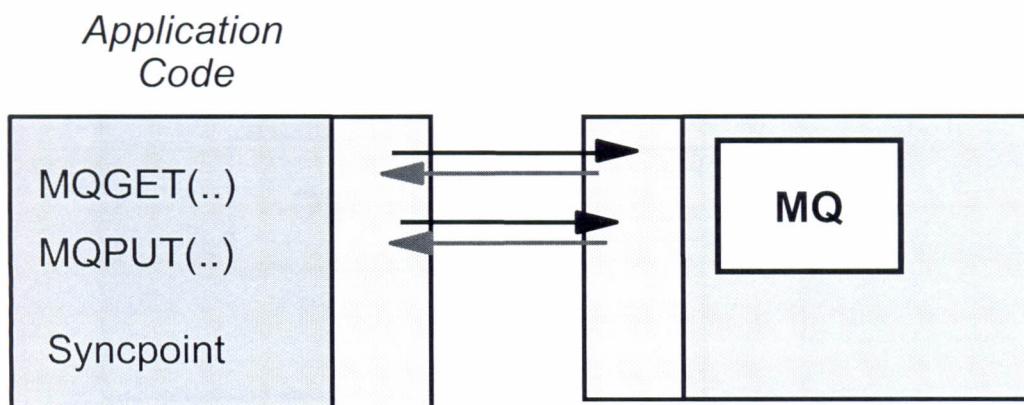
- The use of the `MQMO_CONVERT` option on the `MQGET` call. This option causes the application data of the message to be converted into the numeric and character representation in use on the client system. The server queue manager provides the typical level of support.
- A client application may be connected to more than one queue manager simultaneously. Each `MQCONN` call to a different queue manager returns a different connection handle. This behavior does not apply if the application is not running as a WebSphere MQ client.

The MQI stub which is linked with an application when running as a client is different from the one used when the application is not running as a client. An application receives the reason code `MQRC_Q_MGR_NOT_AVAILABLE` on an `MQCONN` call if it is linked with the wrong MQI stub.

Notes

A Web resource calls in
A Web Such a on the of work it may manag other.

Syncpoint control on a base client



- WebSphere MQ client application can participate in a local unit of work involving MQSeries resources
 - Uses the MQCMIT and MQBACK calls for this purpose
- WebSphere MQ client application cannot participate in a global unit of work involving WebSphere MQ resources

© Copyright IBM Corporation 2008

Figure 13-5. Syncpoint control on a base client

WM203 / VM2032.0

Notes:

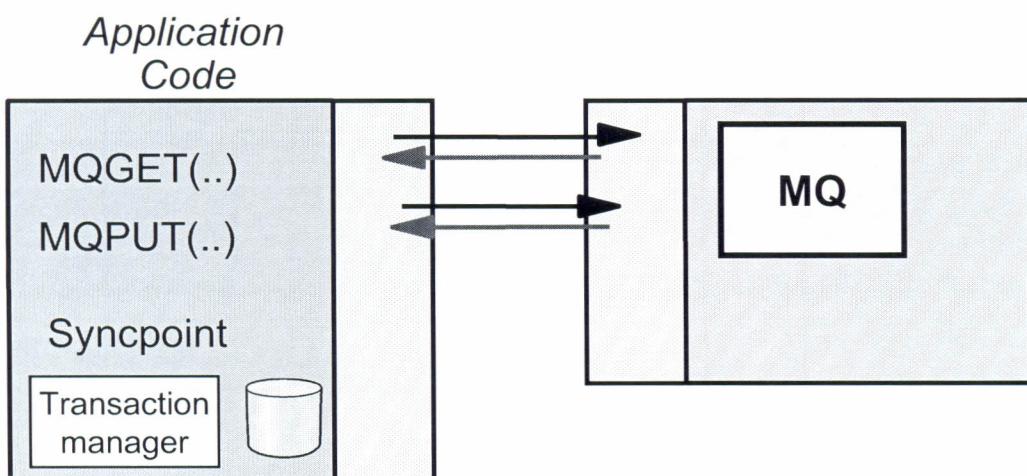
A WebSphere MQ client application can participate in a local unit of work involving the resources of the queue manager it is connected to. It uses the MQCMIT and MQBACK calls in the normal way.

A WebSphere MQ client application cannot however participate in a global unit of work. Such an application can issue calls to another resource manager or sync point coordinator, on the same system as the application or on another system, and may participate in a unit of work associated with that resource manager or sync point coordinator. At the same time, it may also be participating in a local unit of work involving the resources of the queue manager it is connected to. In which case, the two units of work are independent of each other.

VM2032.0
Figure 13-5. Syncpoint control on a base client
WM203 / VM2032.0

Notes:
A WebSphere MQ client application can participate in a local unit of work involving the resources of the queue manager it is connected to. It uses the MQCMIT and MQBACK calls in the normal way.
A WebSphere MQ client application cannot however participate in a global unit of work. Such an application can issue calls to another resource manager or sync point coordinator, on the same system as the application or on another system, and may participate in a unit of work associated with that resource manager or sync point coordinator. At the same time, it may also be participating in a local unit of work involving the resources of the queue manager it is connected to. In which case, the two units of work are independent of each other.

Extended transactional client



- Extended transactional client can participate in global unit of work
 - Transaction manager runs on client system
 - Transaction manager provides syncpoint processing

© Copyright IBM Corporation 2008

Figure 13-6. Extended transactional client

WM203 / VM2032.0

Notes:

A *WebSphere MQ extended transactional client* is a WebSphere MQ client with some additional function. This function allows a client application, within the same unit of work:

- To put messages to, and get messages from, queues that are owned by the queue manager to which it is connected
- To update the resources of a resource manager other than a WebSphere MQ queue manager

This unit of work must be managed by an external transaction manager that is running on the same system as the client application. The unit of work cannot be managed by the queue manager to which the client application is connected. This means that the queue manager can act only as a resource manager, not as a transaction manager. It also means that the client application can commit or back out the unit of work using only the application programming interface (API) provided by the external transaction manager. The client application cannot, therefore, use the MQI calls, MQBEGIN, MQCMIT, and MQBACK.

Also: Even though there is no extended transactional client that runs on i5/OS, a client application that is using an extended transactional client can still connect to a queue manager that runs on i5/OS.

Note, however, that an extended transactional client cannot connect to a queue manager on z/OS.

of

/ VM2032.0

ne
work:

ueue

ueue

ning on
the
queue
means
lication
ent
CK.

Corp. 2008

Client installation

- Method 1
 - Install from WebSphere MQ client CD-ROM supplied with product
- Method 2
 - For a client platform which is the same as the server platform
 - Select required client component during standard install
- Method 3
 - Install client component on the server system
 - Copy files of client component to client system
 - Modify CONFIG.SYS, AUTOEXEC.BAT, and so forth
- Method 4
 - Install from SupportPac

© Copyright IBM Corporation 2008

Figure 13-7. Client installation

WM203 / VM2032.0

Figure 13-

Notes:

WebSphere MQ product supplies a WebSphere MQ Client CD-ROM which contains the software, including an easy installation feature, for clients on the following platforms.

- AIX (includes Java client)
- HP-UX (includes Java client)
- Linux
- Solaris (includes Java client)
- Windows (includes Java client)

WebSphere MQ for Compaq OpenVMS and WebSphere MQ on UNIX systems other than AIX, HP-UX, and Sun Solaris include the files for the desktop clients and for a client on the same platform as the server platform.

Most WebSphere MQ products supply files for clients on the same platform as the server and for clients on other platforms. WebSphere MQ for iSeries, z/OS, and WebSphere MQ for Compaq Tru64 UNIX and VSE/ESA can supply files for clients on other platforms only.

Checkpoint questions

1. Which best describes an WebSphere MQ client?
 - a. A channel to a WebSphere MQ server
 - b. An API for WebSphere MQ
 - c. A customer who purchases WebSphere MQ
 - d. Libraries and communications allowing an application to use a remote WebSphere MQ server
2. True or false: An WebSphere MQ client can participate in global (XA) units of work.
1. Which are valid sources of WebSphere MQ clients?
 1. WebSphere MQ installation CDs
 2. Download from WebSphere MQ SupportPac site
 3. www.dodgeydownloads.com

© Copyright IBM Corporation 2008

Figure 13-8. Checkpoint questions

WM203 / VM2032.0

Notes:

ins the
ms.

other than
ent on the

ne server
here MQ
orms only.

Client overview and installation topic summary

Having completed this topic, you should be able to:

- Define what WebSphere MQ clients do
- Discuss client uses
- Discuss how to get WebSphere MQ clients
- Examine the client installation process

© Copyright IBM Corporation 2008

Figure 13-9. Client overview and installation topic summary

WM203 / VM2032.0

Notes:

Client definition methodologies topic objectives

After completing this topic, you should be able to:

- Define client connections
- Discuss management of client definitions

© Copyright IBM Corporation 2008

Figure 13-10. Client definition methodologies topic objectives

WM203 / VM2032.0

Figure 13

Notes:

Defining a MQI channel

- Use `DEFINE CHANNEL` command with parameters:

<code>CHLTYPE:</code>	<code>CLNTCONN</code> or <code>SVRCONN</code>
<code>TRPTYPE:</code>	<code>DECNET</code> , <code>LU62</code> , <code>NETBIOS</code> , <code>SPX</code> , or <code>TCP</code>
<code>CONNNAME</code> (<i>string</i>)	For a client connection only
<code>QMNAME</code> (<i>string</i>)	For a client connection only

- No operational involvement on an MQI channel
 - MQI channel starts when a client application issues `MQCONN` (or `MQCONNX`)
 - MQI channel stops when a client application issues `MQDISC`
- Four ways to configure for clients
 - Method 1: `MQSERVER` environment variable on client
 - Method 2: Client channel definition table
 - Method 3: Encapsulated connection object in Java (application code)
 - Method 4: External JNDI lookup for JMS (application code)

© Copyright IBM Corporation 2008

Figure 13-11. Defining a MQI channel

WM203 / VM2032.0

Notes:

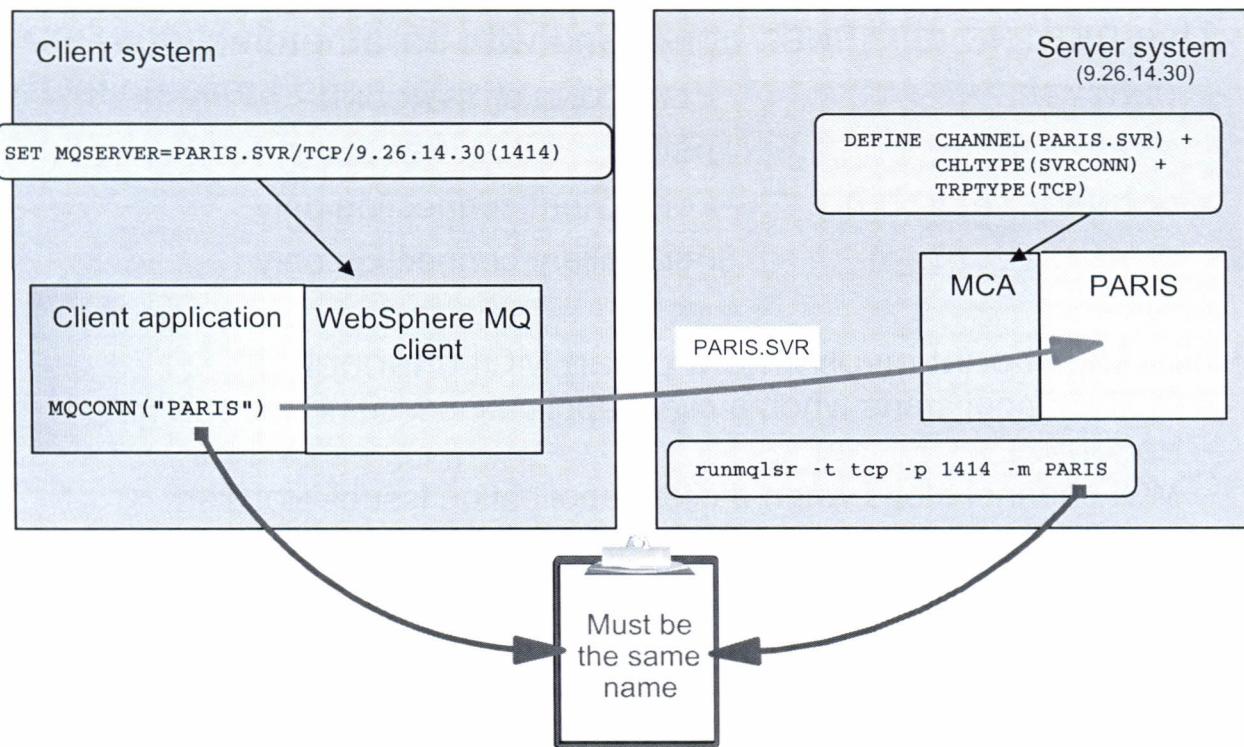
MQI channels do not have to be started. Run the client application which calls an `MQCONN` or `MQCONNX`.

An MQI channel can be used to connect a client to a single queue manager, or to a queue manager that is part of a queue-sharing group.

Do not forget the WebSphere MQ listener must be running on the server system.

Refer to the *WebSphere MQ Using Java* manual for further information regarding Java connection encapsulation and Java Naming and Directory Interface (JNDI) lookup.

Method 1 - Using MQSERVER



© Copyright IBM Corporation 2008

Figure 13-12. Method 1 - Using MQSERVER

WM203 / VM2032.0

Notes:

Method 1 is the easier of the two methods setting up a client. On the server, you use the **DEFINE CHANNEL** command to define a server connection. On the client system, you define a simple client connection by setting the environment variable **MQSERVER** to the following value:

`ChannelName/TransportType/ConnectionName`

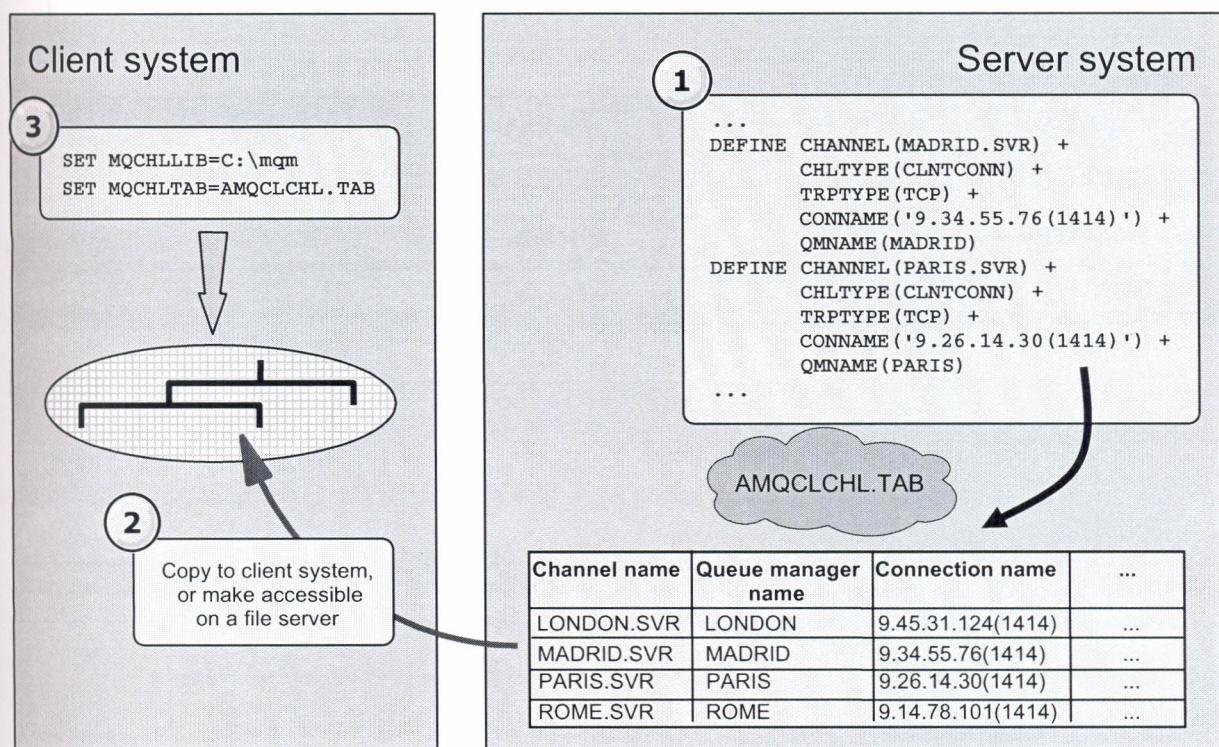
For example, on DOS or Windows:

`SET MQSERVER=VENUS.SVR/TCP/9.20.4.56`

For example, on a UNIX system:

`export MQSERVER=VENUS.SVR/TCP/'9.20.4.56'`

Method 2 - Using a client channel definition table



© Copyright IBM Corporation 2008

Figure 13-13. Method 2 - Using a client channel definition table

WM203 / VM2032.0

Notes:

Method 2 is a more complex method which involves a definition of a client connection and a server connection on the server system.

The client connection definition is stored in the *client channel definition table*, file name AMQCLCHL.TAB. This table must either be accessible from the client system by residing on a file server, or it must be copied to the client system.

On the client system, the environment variables MQCHLLIB and MQCHLTAB are then set in order to specify the location and the name of the client channel definition table.

For example, on DOS or Windows:

```
SET MQCHLLIB=C:\MQM
SET MQCHLTAB=AMQCLCHL.TAB
```

For example, you can set the environment variables on a UNIX system:

```
export MQCHLLIB=/mqmstop/qmgrs/QUEUENAME@ipcc
export MQCHLTAB=AMQCLCHL.TAB
```

WebSphere MQ z/OS platform is the only platform that cannot store the client channel definition table associated with the queue manager running on the server.

Channel §	Queue Manager §	Protocol §	Connection Name §
Stock.A §	Stock §	TCP/IP §	STOCKQMA §
Stock.B §	SALES §	NetBios §	STOCKQMB §
Stock.C §	STOCK §	LU6.2 §	STOCKDEF §
Stock.D §	SALES §	NetBios §	STOCKQMA §
Stock.E §	Stock §	TCP/IP §	STOCKDEF §
Stock.F §	SALES §	TCP/IP §	STOCKDEF §

A

CI
C
MC

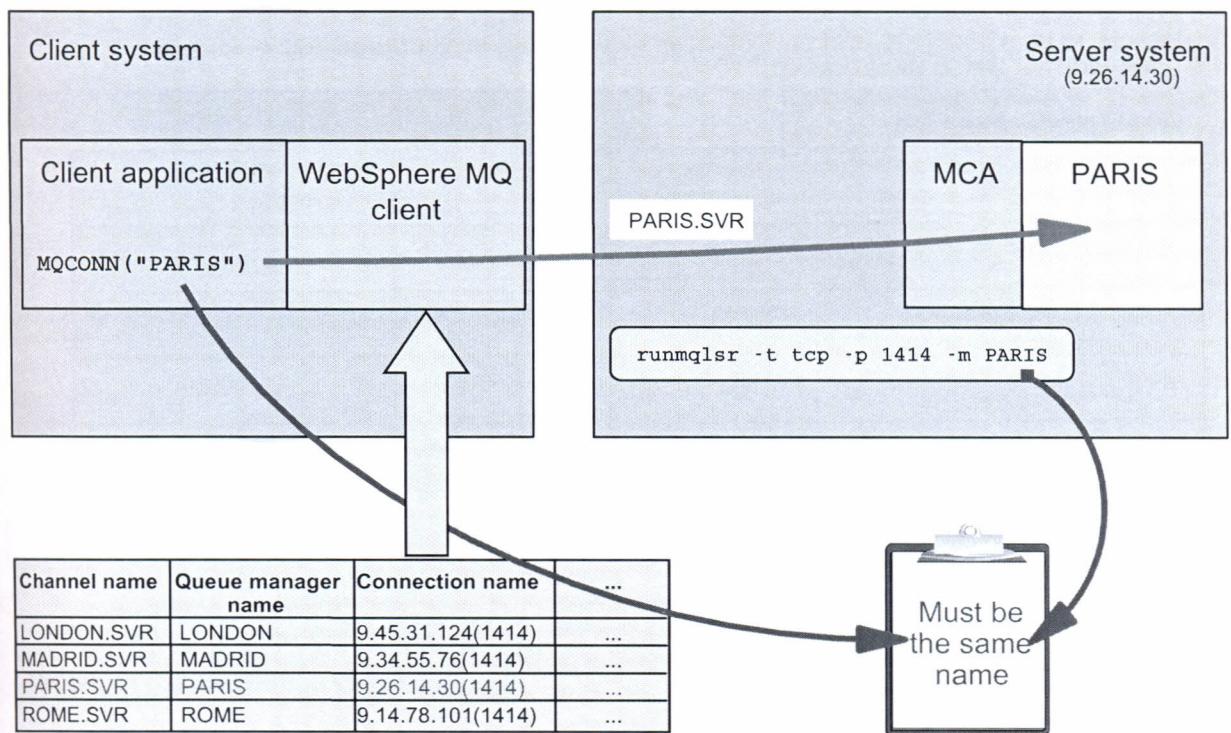
Char
LON
MAD
PAR
RON

Figure 13-

Notes

nel

Accessing a client channel definition table



© Copyright IBM Corporation 2008

Figure 13-14. Accessing a client channel definition table

WM203 / VM2032.0

Notes:

Weighted selection on CLNTCONN channels

Example definition:

```
DEF CHL NAME(SYDNEY.SVR) CHLTYPE(CLNTCONN) QMNAME(GRP1) CONNAME(9.22.44.55) +
CLNTWGHT(0) AFFINITY(PREFERRED)
```

Client Channel Definition Table:

Group Name	Channel Name	Queue Manager Name	Connection	Client Weighting	Affinity
GRP1	LONDON.SVR	LONDON	9.34.45.23(1414)	20	NONE
GRP1	MADRID.SVR	MADRID	9.56.34.12(1414)	50	PREFERRED
GRP1	PARIS.SVR	PARIS	9.3.65.127(1414)	30	NONE
GRP1	SYDNEY.SVR	SYDNEY	9.22.44.55(1414)	0	PREFERRED

Application 1: MQCONN("*GRP1")

First attempt: SYDNEY.SVR
Subsequent attempts: SYDNEY.SVR ✓

Application 2: MQCONN("*GRP1")

First attempt: SYDNEY X
Next attempt: 50% chance MADRID ✓
subsequently 100% MADRID

Application 3: MQCONN("*GRP1")

First attempt: SYDNEY X
Next attempt: 30% chance PARIS
subsequently 100% SYDNEY (if up)
Or MADRID, PARIS or LONDON

* Distribution not guaranteed

© Copyright IBM Corporation 2008

Figure 13-15. Weighted selection on CLNTCONN channels

WM203 / VM2032.0

Figure 13-16.

Notes:

WebSphere MQ V7 introduces 2 new parameters on the CLNTCONN channel definition for use in Client Channel Definition Tables. These two parameters: Client Weighting (CLNTWGHT) and Affinity (AFFINITY) allow load sharing and selection of a channel based on the weights assigned to the channel at definition time. The special channel weighting of 0 means to always use this channel definition. However if this connection is not available, then further matching channels are tried. If there are multiple matching channels with a weighting of 0, then these are tried first, in alphabetic order as with WebSphere MQ V6.

For non-zero client weightings, weights are summed, then a random number between 1 and that sum is generated. This is used to select the matching channel. In the example, a number between 1 and 20 would result in LONDON being selected. 21 to 70 results in MADRID and 71 – 99 in PARIS.

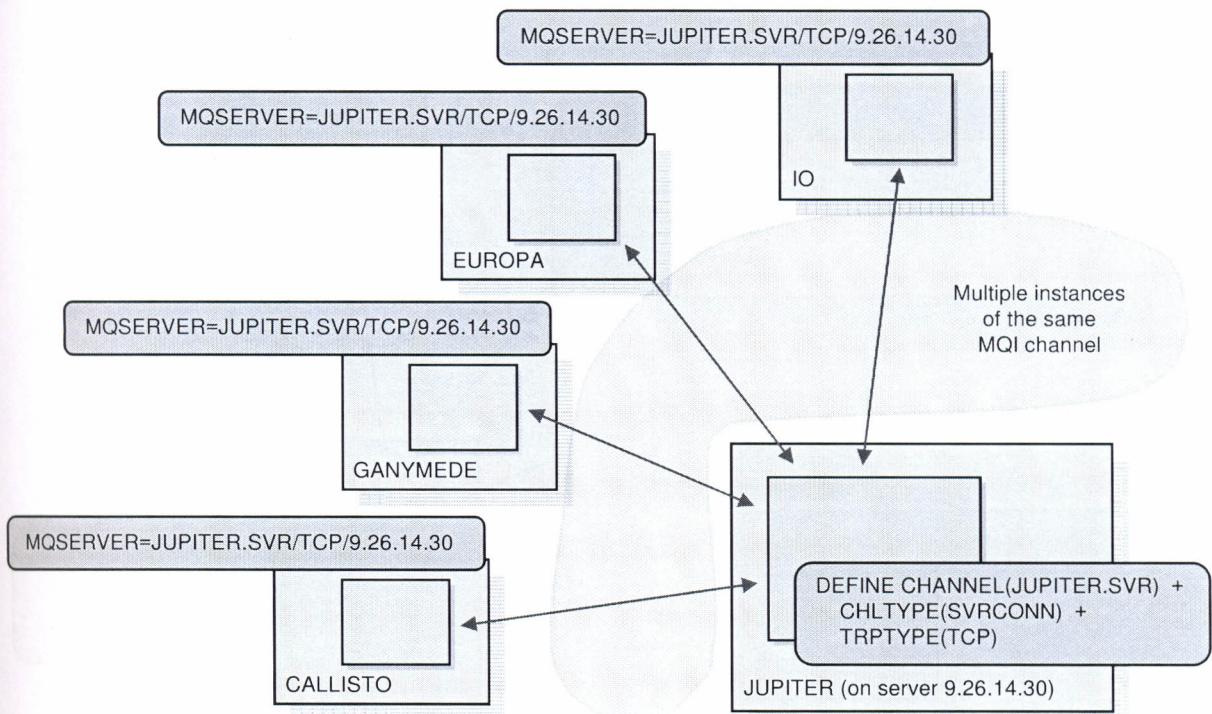
The channel affinity attribute is used so client applications that connect multiple times using the same queue manager name can choose whether to use the same client channel definition for each connection.

Notes:

Each of the channel.

- The sys
- On ser
- Ea JU

Channel instances



© Copyright IBM Corporation 2008

Figure 13-16. Channel instances

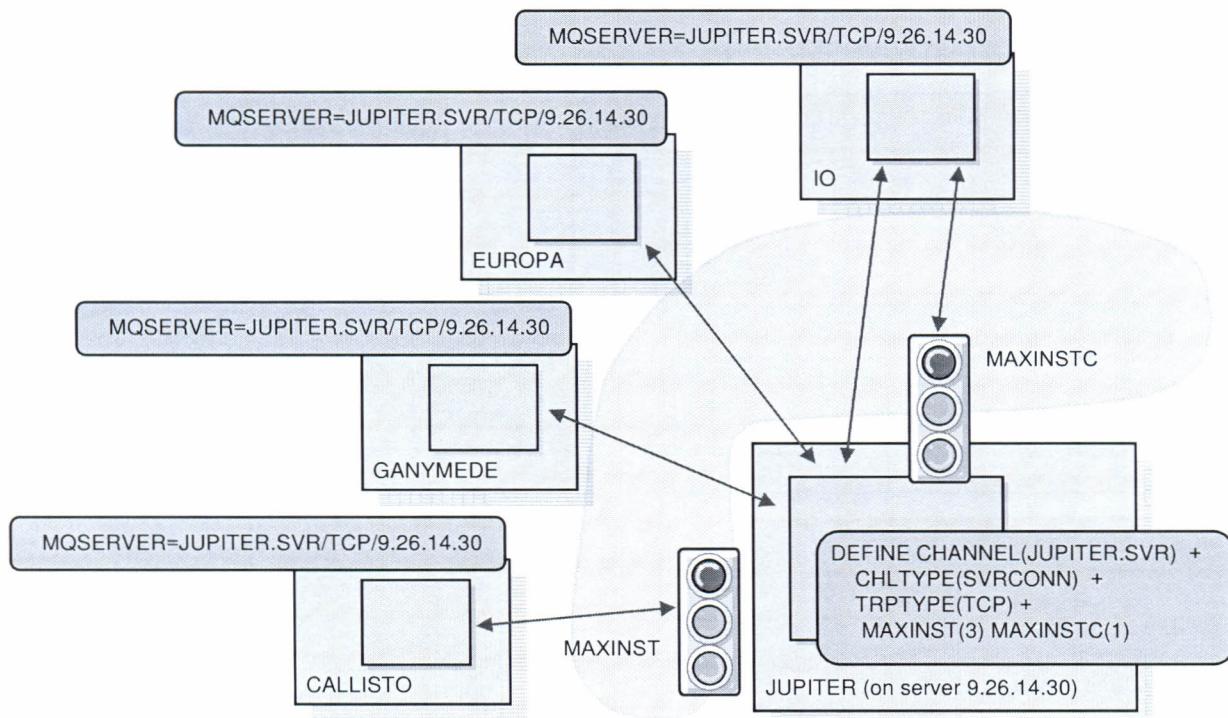
WM203 / VM2032.0

Notes:

Each of the four MQI channels shown on the visual is an *instance* of the same MQI channel. The main points to note are as follows:

- The environment variable MQSERVER has an identical value on each of the client systems with host names IO, EUROPA, GANYMEDE, and CALLISTO.
- On the server system, host name JUPITER, the queue manager requires just one server connection definition for the MQI channel JUPITER.SVR.
- Each client application is using a different instance of the MQI channel JUPITER.SVR.

Instance limits on SVRCONN channels



© Copyright IBM Corporation 2008

Figure 13-17. Instance limits on SVRCONN channels

WM203 / VM2032.0

Notes:

- Max Instances: MAXINST(0 – 999 999 999)
 - Max number of SVRCONN channels of this name that can be started
 - Parameter on SVRCONN channel definition
 - A value of zero prevents all client access on this channel.
 - If value is dynamically reduced:
 - no new channel instances are started
 - Running channel instances remain running
- MAXINSTC(0 – 999 999 999)
 - Limits the quantity of client connections from an individual client. For example: from individual IP addresses
 - A value of zero prevents all client access on this channel.
 - If value is dynamically reduced:

no new channel instances are started from this client
Running channel instances remain running

R) +

03 / VM2032.0

Notebook

total 10 pages

8 pages

8 pages

8 pages

8 pages

example: from

Auto-definition of channels

- Applies only to the end of a channel with type:
 - Receiver
 - Server connection
- Function invoked when an incoming request is received to start a channel but there is no channel definition
- Channel definition is created automatically using the model:
 - SYSTEM.AUTO.RECEIVER
 - SYSTEM.AUTO.SVRCONN
- Partner's values are used for:
 - Channel name
 - Sequence number wrap value

© Copyright IBM Corporation 2008

Figure 13-18. Auto-definition of channels

WM203 / VM2032.0

Figure 13

Notes:

The auto-definition of channels is supported on all WebSphere MQ platforms except for z/OS.

To enable the automatic definition of channels, the attribute *ChannelAutoDef* of the queue manager object must be set to `MQCHAD_ENABLED`. The corresponding parameter on the `ALTER QMGR` command is `CHAD (ENABLED)`.

Once a channel definition has been created in this way and stored, the definition may be used later as though it had always existed.

Channel auto-definition events can be enabled by setting the attribute *ChannelAutoDefEvent* of the queue manager object to `MQEVR_ENABLED`. The corresponding parameter on the `ALTER QMGR` command is `CHADEV (ENABLED)`.

The partner may be any queue manager or client; it does not have to be at Version 6 level.

Checkpoint questions

1. True or false: It is not possible to use the MQSERVER variable for LU6.2 protocol connections
2. True or false: CCDTs cannot be used with MQ Java applications.
3. True or false: A client channel with weighting = 0 is always selected.
4. Which of the following statements are true?
 - a. MAXINST limits the total number of client connections for a specific channel
 - b. MAXINSTC limits the total number of connections for a specific channel from an individual client
 - c. Affinity causes subsequent client connections from a specific client to return to the same channel.
 - d. All of the above

© Copyright IBM Corporation 2008

Figure 13-19. Checkpoint questions

WM203 / VM2032.0

Notes:

for

queue
the

ay be

onding

6 level.

Client definition methodologies topic summary

Having completed this topic, you should be able to:

- Define client connections
- Discuss management of client definitions

© Copyright IBM Corporation 2008

Figure 13-20. Client definition methodologies topic summary

WM203 / VM2032.0

Notes:

13.3.Application development implications

03 / VM2032.0

Application development implications topic objectives

After completing this topic, you should be able to:

- Describe WebSphere MQ client applications
- Discuss limitations of client applications
- Discuss V7 improvements for client applications

© Copyright IBM Corporation 2008

Figure 13-21. Application development implications topic objectives

WM203 / VM2032.0

WebS

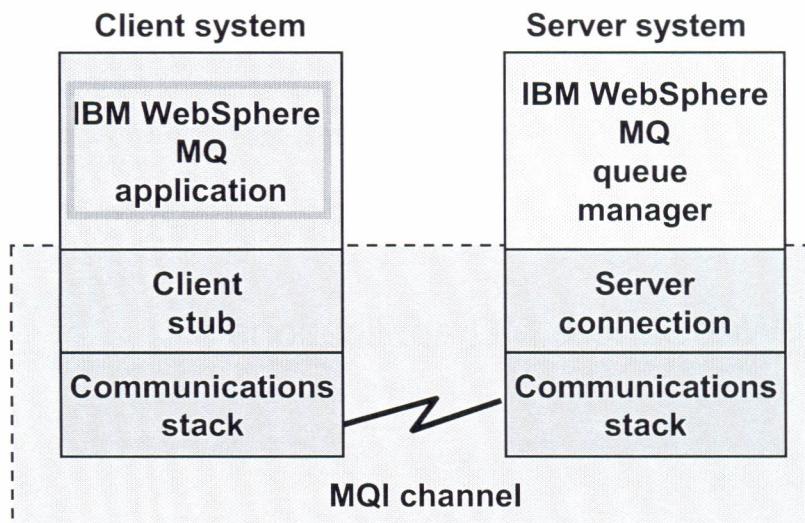
Figure 13-22.

Notes:

Notes:

A WebSp
other by u
an MQCC
client app
paramete
paramete

WebSphere MQ client applications



© Copyright IBM Corporation 2008

Figure 13-22. WebSphere MQ client applications

WM203 / VM2032.0

Notes:

A WebSphere MQ client application and a server queue manager communicate with each other by using an *MQI channel*. An MQI channel starts when the client application issues an MQCONN or MQCONNX call to connect to the queue manager and ends when the client application issues an MQDISC call to disconnect from the queue manager. The input parameters of an MQI call flow in one direction on an MQI channel and the output parameters flow in the opposite direction.

WebSphere MQ client limitations

- Remote to WebSphere MQ server
 - Reliance on network connectivity
- Client based trigger monitors
 - Client trigger monitors are client too
- Distribution of WebSphere MQ maintenance
- Control of WebSphere MQ applications

© Copyright IBM Corporation 2008

Figure 13-23. WebSphere MQ client limitations

WM203 / VM2032.0

MQS

- In V
100
- Usa
– ex
- Mat

Notes:

There are a few limitations when using WebSphere MQ clients:

- Clients use network connections. This introduces a dependency on network availability. This is even the case where the WebSphere MQ client and server are co-located.
- WebSphere MQ client trigger monitors are available. Trigger monitors work just like server-based trigger monitors but are located on the client side.
- Distribution of WebSphere MQ client maintenance can be an issue, particularly if control of the client computer is with another organization.
- Similarly, control of the client application might be outside the WebSphere MQ administrators grasp.

Figure 13-24

Notes:

The MQ channel
an alterna
such as
Only the
number)
limited.

In WebS
length (N
previous

To take a
channel
manag
4 MB, al
after We

MQSERVER enhancements

- In V7, maximum message length increases from 4 MB to 100 MB
- Usage example:
 - `export MQSERVER=SYSTEM.DEF.SVRCONN/TCP/1.2.3.4(1414)`
- Matching SVRCONN channel on server must specify 100 MB

© Copyright IBM Corporation 2008

Figure 13-24. MQSERVER enhancements

WM203 / VM2032.0

Notes:

The MQSERVER environment variable can be used to define a minimal CLNTCONN channel to allow a WebSphere MQ client program to connect to a queue manager. This is an alternative to using a Client Channel Definition Table (CCDT) or a programmatic method such as MQCONN in the MQI to define the channel.

Only the channel name, transport protocol, and the connection name (host name and port number) can be specified in the value of this variable, so the configuration parameters are limited.

In WebSphere MQ V7.0, a channel defined using MQSERVER has a maximum message length (MAXMSGL) of 100 MB, increased from the 4 MB maximum message length in previous versions of WebSphere MQ.

To take advantage of the increased maximum length the identically named SVRCONN type channel must also have its MAXMSGL altered to an appropriate value on the queue manager. The default value taken for MAXMSGL when an SVRCONN channel is defined is 4 MB, although this might have changed on the SYSTEM.DEF.SVRCONN channel object after WebSphere MQ was installed.

Compatibility

- V6 and V7 clients can connect to both V6 and V7 queue managers
- V6 clients cannot use new features of clients
 - Weighting, affinity
 - Reconnection
- V7 client channel definition tables are not compatible with V6 clients

© Copyright IBM Corporation 2008

Figure 13-25. Compatibility

WM203 / VM2032.0

Figure 13-2

Notes:

There are some limitations on the use of new features of V7 clients such as channel weighting and reconnection. These are invisible to the application program.

Be aware CCDTs generated with WebSphere V7 is incompatible with V6 clients.

Notes

Application development implications topic summary

Having completed this topic, you should be able to:

- Describe WebSphere MQ client applications
- Discuss limitations of client applications
- Discuss V7 improvements for client applications

© Copyright IBM Corporation 2008

Figure 13-26. Application development implications topic summary

WM203 / VM2032.0

Notes:

nnel

WebSphere MQ client security considerations topic objectives

After completing this topic, you should be able to:

- Review WebSphere MQ client security features
- Describe the use of channel security exits for clients
- Discuss use of SSL for client channels

© Copyright IBM Corporation 2008

Figure 13-27. WebSphere MQ client security considerations topic objectives

WM203 / VM2032.0

Notes:

- Access procedures
- Security – Client – Server – MQ
- No security – Variables – Services – MQ
- No security – MQ – MQ

Figure 13-28.

Notes:

When a WebSphere MQ client connects to the server, it needs to have access control information. The source for this is the client's configuration, which determines the connection to the server based on the connection structure defined in the configuration.

Access control for a WebSphere MQ client

- Access control based on user ID used by server connection process (*MCAUserIdentity* in MQCD)
- Security exits at both ends of the MQI channel
 - Client security exit can flow a user ID and password
 - Server security exit can authenticate the user ID and set *MCAUserIdentity*
- No security exit at the client end of the MQI channel
 - Value of logged in user ID flows to the server system
 - Server security exit can authenticate the user ID and set *MCAUserIdentity*
- No security exit at either end of the MQI channel
 - *MCAUserIdentity* has the value of *MCAUSER* if it is nonblank
 - *MCAUserIdentity* has the value of flowed user ID otherwise

© Copyright IBM Corporation 2008

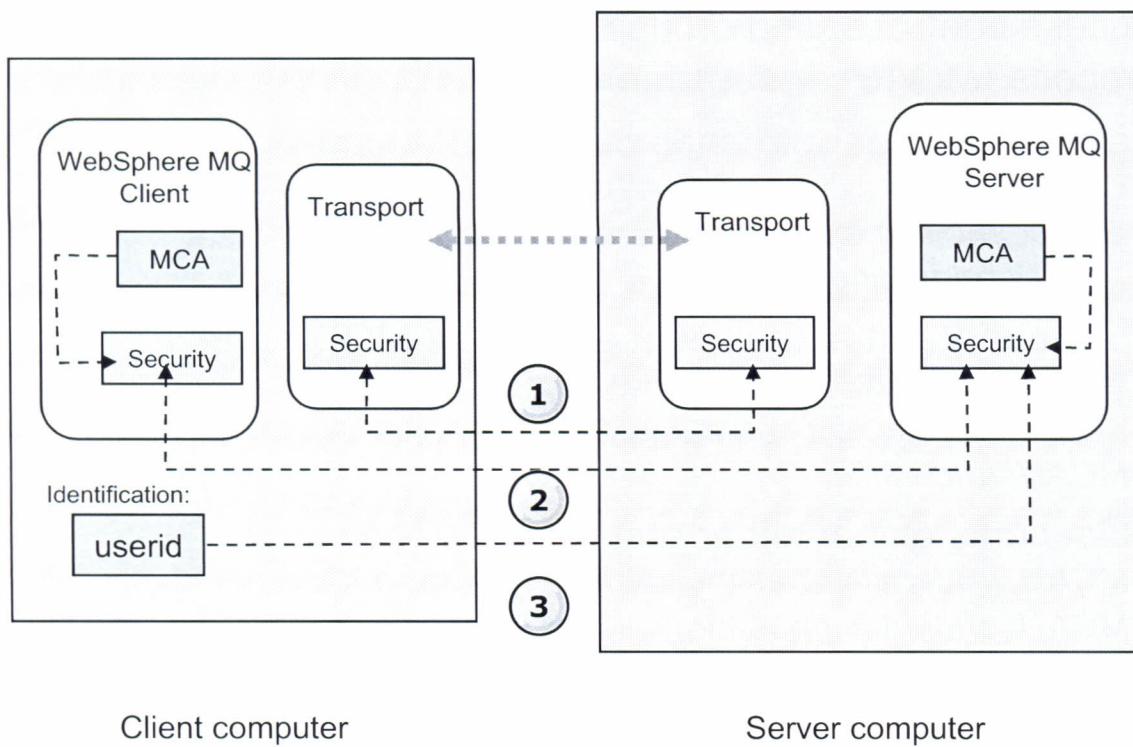
Figure 13-28. Access control for a WebSphere MQ client

WM203 / VM2032.0

Notes:

When a WebSphere MQ client application wants to access a **WebSphere MQ** object on the server queue manager (for example, connect to the queue manager, or open a queue), access control is based on a user ID used by the server connection process. The reason for this is that it is the server connection which actually issues the MQI calls. This user ID is determined by the value of the *MCAUserIdentity* field in the active channel definition at the server end of the MQI channel instance. The active channel definition is defined by the structure MQCD and is documented in *WebSphere MQ Intercommunication* manual.

WebSphere MQ client security



© Copyright IBM Corporation 2008

Figure 13-29. WebSphere MQ client security

WM203 / VM2032.0

Figure 13-3

Notes:

Security checks for WebSphere MQ Clients are:

1. Transport level checks, such as IP filtering, and so on
2. Custom channel exits, such as to perform security checks to custom specification
3. Default user ID propagation such as, WebSphere MQ passes the currently logged-on user ID that may be used by a server-side security exit to perform authentication.

Notes

WebSphere MQ client security considerations topic summary

Having completed this topic, you should be able to:

- Review WebSphere MQ client security features
- Describe the use of channel security exits for clients
- Discuss use of SSL for client channels

© Copyright IBM Corporation 2008

Figure 13-30. WebSphere MQ client security considerations topic summary

WM203 / VM2032.0

Notes:

Unit summary

Having completed this unit, you should be able to:

- Describe the various ways to connect a client to a queue manager
- Analyze client connection requirements in order to determine the best approach
- Explain the limitations of various client connection methods
- Propose methodologies to ensure security with client connections

What t

What y

How y

Refer

© Copyright IBM Corporation 2008

Figure 13-31. Unit summary

WM203 / VM2032.0

Notes: