

9.0

IBM MQ 的疑難排解及支援

IBM

附註

使用本資訊及其支援的產品之前，請先閱讀第 237 頁的『[注意事項](#)』中的資訊。

除非新版中另有指示，否則此版本適用於 IBM® MQ 9.0.0 版及所有後續版本與修訂版。

當您將資訊傳送至 IBM 時，您授與 IBM 非專屬權利，以任何其認為適當的方式使用或散佈資訊，而無需對您負責。

© Copyright International Business Machines Corporation 2007, 2023.

目錄

疑難排解和支援	7
疑難排解概觀.....	7
進行起始檢查.....	8
對 UNIX, Linux, and Windows 進行起始檢查.....	9
對 IBM i 進行起始檢查.....	16
對 z/OS 進行起始檢查.....	24
聯絡 IBM 支援中心.....	36
使用錯誤日誌.....	37
UNIX, Linux, and Windows 上的錯誤日誌.....	38
IBM i 上的錯誤日誌.....	41
IBM MQ classes for JMS 中的錯誤日誌.....	44
在 Multiplatforms 上暫停來自錯誤日誌的通道錯誤訊息.....	44
First Failure Support Technology (FFST).....	45
FFST: IBM MQ classes for JMS.....	46
FFST: IBM MQ for Windows.....	50
FFST: IBM MQ for UNIX 和 Linux 系統.....	52
FFST: IBM MQ for IBM i.....	54
使用追蹤.....	56
在 Windows 上使用追蹤.....	56
在 UNIX and Linux 系統上使用追蹤.....	58
在 IBM i 上搭配使用追蹤與 IBM MQ 伺服器.....	60
在 IBM i 上搭配使用追蹤與 IBM MQ 用戶端.....	63
在 z/OS 上使用追蹤來判斷問題.....	65
追蹤 Advanced Message Queuing Protocol (AMQP) 服務.....	75
追蹤 TLS: runmqakm 、 strmqikm 及 runmqckm 函數.....	77
追蹤 IBM MQ classes for JMS 應用程式.....	78
追蹤 IBM MQ classes for Java 應用程式.....	82
追蹤 IBM MQ 資源配接器.....	85
追蹤其他 IBM MQ Java 元件.....	87
使用 IBM MQ classes for Java 和 IBM MQ classes for JMS 來控制執行中處理程序中的追蹤.....	89
追蹤 IBM MQ .NET 應用程式.....	92
追蹤 IBM MQ Console 和 REST API.....	92
追蹤 IBM MQ Bridge to Salesforce.....	93
追蹤 IBM MQ Bridge to blockchain.....	94
啟用 LDAP 用戶端程式庫程式碼的動態追蹤.....	94
z/OS 上的問題判斷.....	95
IBM MQ for z/OS 效能限制.....	96
IBM MQ for z/OS 回復動作.....	97
IBM MQ for z/OS 異常終止.....	98
在 IBM MQ for z/OS 上產生的診斷資訊.....	101
IBM MQ for z/OS 的其他問題判斷資訊來源.....	102
CICS 的診斷輔助工具.....	103
IMS 的診斷輔助工具.....	104
Db2 的診斷輔助工具.....	104
疑難排解從 IBM MQ for z/OS 到 Product Insights 的連線.....	104
IBM MQ for z/OS 傾出.....	104
處理 z/OS 上的效能問題.....	122
處理 z/OS 上不正確的輸出.....	127
處理擷取通道起始程式 (CHINIT) 的 SMF 資料時的問題.....	132
DQM 中的問題判斷.....	133
來自通道控制的錯誤訊息.....	134
連線測試.....	134

無法傳送郵件的佇列考量.....	134
驗證檢查.....	135
不確定的關係.....	135
通道啟動協議錯誤.....	135
共用通道回復.....	135
當通道拒絕執行時.....	136
正在重試鏈結.....	137
資料結構.....	138
使用者結束程式問題.....	138
災難回復.....	138
通道切換.....	138
連線切換.....	139
用戶端問題.....	139
錯誤日誌.....	139
訊息監視.....	140
通道鑑別記錄疑難排解.....	140
指令疑難排解.....	141
分散式發佈/訂閱疑難排解.....	141
發佈/訂閱叢集的遞送: 行為注意事項.....	142
檢查 Proxy 訂閱位置.....	143
重新同步化 Proxy 訂閱.....	143
分散式發佈/訂閱網路中的迴圈偵測.....	144
Java 和 JMS 疑難排解.....	145
IBM MQ classes for JMS 問題疑難排解.....	145
JMS 提供者版本疑難排解.....	146
JMS 中的 PCF 處理.....	147
JMS 連線儲存區錯誤處理.....	148
疑難排解 JMSSC0108 訊息.....	151
WebSphere Application Server SystemOut.log 檔案中的 CWSJY0003W 警告訊息.....	153
包含方法 'xa_end' 失敗, 並顯示 errorCode '100' 錯誤的 J2CA0027E 訊息.....	154
2035 從 WebSphere Application Server 連接至 IBM MQ 時 MQRC_NOT_AUTHORIZED.....	155
IBM MQ 資源配接器的問題判斷.....	158
使用 IBM MQ 連線內容置換.....	159
疑難排解 IBM MQ.NET 問題.....	164
解決 IBM MQ MQI clients 的問題.....	165
多重播送疑難排解.....	166
在非多重播送網路上測試多重播送應用程式.....	166
為多重播送資料流量設定適當的網路.....	166
多重播送主題字串太長.....	166
多重播送主題拓撲問題.....	167
佇列管理程式叢集疑難排解.....	168
執行 REFRESH CLUSTER 時看到的應用程式問題.....	170
叢集傳送端通道持續嘗試啟動.....	172
DISPLAY CLUSQMGR show CLUSQMGR names starting SYSTEM.TEMP。.....	173
回覆碼 = 2035 MQRC_NOT_AUTHORIZED.....	174
嘗試開啟叢集中的佇列時, 回覆碼 = 2085 MQRC_UNKNOWN_OBJECT_NAME.....	174
嘗試開啟叢集中的佇列時, 回覆碼 = 2189 MQRC_CLUSTER_RESOLUTION_ERROR.....	175
傳回 code=2082 MQRC_UNKNOWN_ALIAS_BASE_Q 開啟叢集中的佇列.....	175
訊息未到達目的地佇列.....	176
放入叢集別名佇列的訊息會移至 SYSTEM.DEAD.LETTER.QUEUE.....	176
佇列管理程式叢集中佇列及通道的相關資訊已過期.....	177
叢集中沒有任何變更反映在本端佇列管理程式中.....	177
「DISPLAY CLUSQMGR」會顯示佇列管理程式兩次.....	178
佇列管理程式未重新結合叢集.....	178
叢集傳送端通道上設定的工作量平衡無法運作.....	179
已還原叢集中的過期資訊.....	179
錯誤地強制從完整儲存庫中移除叢集佇列管理程式.....	179
已刪除可能的儲存庫訊息.....	180




同時移動兩個完整儲存庫.....	180
叢集的不明狀態.....	181
叢集佇列管理程式失敗時發生的情況.....	181
儲存庫失敗時發生的情況.....	182
針對 MQPUT 停用叢集佇列會發生什麼情況.....	182
切換傳輸佇列時的潛在問題.....	183
佇列管理程式疑難排解.....	184
未遞送訊息疑難排解.....	185
TLS 疑難排解資訊.....	185
RDQM 配置疑難排解.....	193
RDQM HA 架構.....	193
RDQM HA 配置和錯誤範例.....	194
MQ Telemetry 疑難排解.....	202
Telemetry 日誌、錯誤日誌和配置檔的位置.....	202
追蹤遙測 (MQXR) 服務.....	204
解決問題: MQTT 用戶端未連接.....	206
解決問題: 已捨棄 MQTT 用戶端連線.....	207
解決問題: MQTT 應用程式中遺失訊息.....	208
解決問題: 遙測 (MQXR) 服務未啟動.....	209
解決問題: Telemetry 服務未呼叫 JAAS 登入模組.....	210
失敗後回復.....	213
磁碟機故障.....	214
佇列管理程式物件已損壞.....	215
損壞的單一物件.....	215
自動媒體回復失敗.....	215
z/OS 上的回復程序範例.....	215
注意事項.....	237
程式設計介面資訊.....	238
商標.....	238

IBM MQ 疑難排解及支援

如果您的佇列管理程式網路或 IBM MQ 應用程式有問題，請使用說明的技術來協助您診斷及解決問題。

如需疑難排解及支援的簡介，請參閱第 7 頁的『疑難排解概觀』。

您可以對平台進行一些起始檢查，以協助判斷部分一般問題的原因。請參閱適合您平台的主題：


-  第 9 頁的『對 UNIX, Linux, and Windows 進行起始檢查』
-  第 16 頁的『對 IBM i 進行起始檢查』
-  第 24 頁的『對 z/OS 進行起始檢查』

如需解決問題的相關資訊，請參閱第 8 頁的『進行起始檢查』。

如需解決 MQ Telemetry 問題的相關資訊，請參閱第 202 頁的『MQ Telemetry 疑難排解』。

如需使用通道鑑別記錄時解決問題的相關資訊，請參閱第 140 頁的『通道鑑別記錄疑難排解』。

IBM MQ 所產生的資訊可協助您尋找並解決問題。如需相關資訊，請參閱下列主題：

- 第 37 頁的『使用錯誤日誌』
- 第 56 頁的『使用追蹤』
-  第 95 頁的『z/OS 上的問題判斷』
- 第 45 頁的『First Failure Support Technology (FFST)』

如需在問題之後回復的相關資訊，請參閱第 213 頁的『失敗後回復』。

另請參閱「第 36 頁的『聯絡 IBM 支援中心』」。

如果 IBM MQ 元件或指令已傳回錯誤，且您想要將訊息寫入畫面或日誌的進一步相關資訊，您可以瀏覽訊息的詳細資料，請參閱 [訊息及原因碼](#)。

相關資訊

[疑難排解和支援參照](#)

疑難排解概觀

疑難排解是尋找並消除問題原因的程序。每當您使用 IBM 軟體時發生問題，只要您問自己「發生什麼事？」，疑難排解程序即會開始。

高階的基本疑難排解策略包括：

1. [第 7 頁的『記錄問題的症狀』](#)
2. [第 8 頁的『重建問題』](#)
3. [第 8 頁的『消除可能的原因』](#)

記錄問題的症狀

視您所擁有的問題類型而定，不論是應用程式、伺服器或工具的問題，您可能會收到一則訊息，指出發生錯誤。一律記錄您看到的錯誤訊息。雖然這聽起來很簡單，但錯誤訊息有時會包含一些程式碼，當您進一步調查問題時，這些程式碼可能會更有意義。您也可能會收到多個看起來類似但有細微差異的錯誤訊息。透過記錄每一個問題的詳細資料，您可以進一步瞭解問題的存在位置。

錯誤訊息來源：


- 問題視圖
- 本端錯誤日誌
- Eclipse 日誌

- 使用者追蹤
- 服務追蹤
- 錯誤對話框

重建問題

回想一下您所執行的步驟導致問題的原因。請重試這些步驟，以查看您是否可以輕鬆重建問題。如果您有一致可重複的測試案例，則更容易判斷需要哪些解決方案。

- 您是如何第一次注意到問題的？
- 你做了什麼讓你注意到問題的事嗎？
- 造成問題的處理程序是新程序，還是之前已順利運作？
- 如果此程序之前運作過，則有哪些變更？(變更可以指對系統所做的任何類型變更，範圍從新增硬體或軟體，到重新配置現有軟體。)
- 您所看到問題的第一個症狀是什麼？是否有其他症狀在同一時間出現？
- 其他地方也有同樣的問題嗎？只有一部機器發生問題，還是多部機器發生相同問題？
- 正在產生哪些可能指出問題所在的訊息？

 您可以在 [第 9 頁的『對 UNIX, Linux, and Windows 進行起始檢查』](#) 中找到這些問題類型的相關資訊。

消除可能的原因

刪除未造成問題的元件，以縮小問題的範圍。透過使用排除處理程序，您可以簡化問題，並避免在不負責的區域中浪費時間。請參閱本產品及其他可用資源中的資訊，以協助您進行排除處理程序。




進行起始檢查

您可以進行一些起始檢查，以回答您可能遇到的一般問題。





關於這項作業

請使用子主題中提供的資訊及一般建議，以協助您對平台執行起始檢查並更正問題。

程序

- 針對您的平台執行起始檢查：
 -  [第 9 頁的『對 UNIX, Linux, and Windows 進行起始檢查』](#)
 -  [第 24 頁的『對 z/OS 進行起始檢查』](#)
 -  [第 16 頁的『對 IBM i 進行起始檢查』](#)

系統管理者的提示

- 請檢查錯誤日誌，以取得適用於您作業系統的訊息：
 -  [第 38 頁的『UNIX, Linux, and Windows 上的錯誤日誌』](#)
 -  [第 41 頁的『IBM i 上的錯誤日誌』](#)
 -  [第 101 頁的『在 IBM MQ for z/OS 上產生的診斷資訊』](#)
- 檢查 `qm.ini` 的內容是否有任何配置變更或錯誤。
如需變更配置資訊的相關資訊，請參閱：
 -  [變更 UNIX, Linux, and Windows 上的配置資訊](#)

– **IBM i** [變更 IBM i 上的配置資訊](#)

– **z/OS** [在 z/OS 上自訂佇列管理程式](#)

- 如果您的應用程式開發團隊正在報告非預期的內容，您可以使用追蹤來調查問題。

如需使用追蹤的相關資訊，請參閱 [第 56 頁的『使用追蹤』](#)。

應用程式開發人員的提示

- 請檢查應用程式中 MQI 呼叫的回覆碼。

如需原因碼清單，請參閱 [API 完成及原因碼](#)。請使用回覆碼中提供的資訊來判斷問題的原因。請遵循原因碼之「程式設計師」回應小節中的步驟來解決問題。

- 如果您不確定應用程式是否如預期般運作，例如，您不確定傳遞至 MQI 或 MQI 之外的參數，則可以使用追蹤來收集 MQI 呼叫所有輸入及輸出的相關資訊。

如需使用追蹤的相關資訊，請參閱 [第 56 頁的『使用追蹤』](#)。如需在 MQI 應用程式中處理錯誤的相關資訊，請參閱 [處理程式錯誤](#)。

相關概念

[第 37 頁的『使用錯誤日誌』](#)

您可以使用各種錯誤日誌來協助進行問題判斷及疑難排解。

相關工作

[第 56 頁的『使用追蹤』](#)

您可以使用不同類型的追蹤，協助您判斷問題及進行疑難排解。

相關資訊

[疑難排解和支援參照](#)

ULW 對 UNIX, Linux, and Windows 進行起始檢查

在 UNIX, Linux, and Windows 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

關於這項作業

您問題的原因可能是：

- IBM MQ
- 網路
- 應用程式
- 您已配置為使用 IBM MQ 的其他應用程式

程序

- 請考量下列問題清單。

在您瀏覽清單時，請記下可能與問題相關的任何事項。即使您的觀察沒有立即提出原因，但如果您必須進行有系統的問題判斷工作，則稍後可能會有用。

- [第 10 頁的『IBM MQ 之前是否已順利執行?』](#)
- [第 10 頁的『自前次成功執行以來是否已進行任何變更?』](#)
- [第 11 頁的『是否有任何錯誤訊息或回覆碼來說明問題?』](#)
- [第 11 頁的『您可以重新產生問題嗎?』](#)
- [第 11 頁的『在「Windows」上建立或啟動佇列管理程式時，您是否收到錯誤碼?』](#)
- [第 11 頁的『問題是否僅影響遠端佇列?』](#)
- [第 12 頁的『您是否取得不正確的輸出?』](#)
- [第 14 頁的『您的部分佇列是否失敗?』](#)
- [第 14 頁的『您是否未收到來自 PCF 指令的回應?』](#)

- [第 15 頁的『之前是否已順利執行應用程式?』](#)
- [第 16 頁的『您的應用程式或系統執行緩慢嗎?』](#)
- [第 16 頁的『問題會影響網路的特定部分嗎?』](#)
- [第 16 頁的『問題是否在一天中的特定時間發生?』](#)
- [第 16 頁的『問題是否間歇性?』](#)

相關工作

[第 24 頁的『對 z/OS 進行起始檢查』](#)

在 z/OS 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

[第 16 頁的『對 IBM i 進行起始檢查』](#)

在 IBM i 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

[第 36 頁的『聯絡 IBM 支援中心』](#)

如果您在使用 IBM MQ 時遇到問題需要協助，您可以透過「IBM 支援中心網站」聯絡 IBM 支援中心。您也可以訂閱 IBM MQ 修正程式、疑難排解及其他新聞的相關通知。

相關資訊

[訊息及原因碼](#)

[PCF 原因碼](#)

[疑難排解和支援參照](#)

ULW IBM MQ 之前是否已順利執行?

如果 IBM MQ 之前未順利執行，則可能是您尚未正確設定它。請參閱 [安裝 IBM MQ](#)，並選取您企業用來檢查您是否已正確安裝產品的一或多個平台。

如果要執行驗證程序，請針對您企業使用的一或多個平台，參閱 [驗證 IBM MQ 安裝](#)。

另請參閱 [配置](#)，以取得 IBM MQ 後置安裝配置的相關資訊。

ULW 自前次成功執行以來是否已進行任何變更?

對 IBM MQ 配置所做的變更、維護更新，或對與 IBM MQ 互動的其他程式所做的變更，可能是您問題的原因。

當您考量最近可能已進行的變更時，請考量 IBM MQ 系統，以及它與之互動的其他程式、硬體及任何新應用程式。另請考量您不知道的新應用程式可能已在系統上執行的可能性。

- 您是否已變更、新增或刪除任何佇列定義?
- 您是否已變更或新增任何通道定義? 可能已對 IBM MQ 通道定義或應用程式所需的任何基礎通訊定義進行變更。
- 您的應用程式是否處理因您所做的任何變更而可能取得的回覆碼?
- 您是否已變更任何可能影響 IBM MQ 作業的作業系統元件? 例如，您是否已修改 Windows 登錄。

您已套用任何維護更新項目嗎?

如果您已將維護更新套用於 IBM MQ，請確認更新動作已順利完成，且未產生任何錯誤訊息。

- 更新有任何特殊指示嗎?
- 是否有執行任何測試來驗證已正確且完整套用更新?
- 如果 IBM MQ 還原至先前的維護層次，問題是否仍然存在?
- 如果安裝成功，請洽詢「IBM 支援中心」，以找出任何維護套件錯誤。
- 如果維護套件已套用於任何其他程式，請考量它對 IBM MQ 介面的影響。

ULW 是否有任何錯誤訊息或回覆碼來說明問題?

您可能會找到錯誤訊息或回覆碼，可協助您判斷問題的位置及原因。

IBM MQ 會使用錯誤日誌來擷取關於其專屬作業、您啟動的任何佇列管理程式，以及來自使用中通道的錯誤資料的訊息。請檢查錯誤日誌，以查看是否已記錄與您問題相關聯的任何訊息。

IBM MQ 也會在 Windows 應用程式事件日誌中記載錯誤。在 Windows 上，檢查 Windows 應用程式事件日誌是否顯示任何 IBM MQ 錯誤。若要開啟日誌，請從「電腦管理」畫面中展開「事件檢視器」，然後選取應用程式。

ULW 如需錯誤日誌的位置和內容的相關資訊，請參閱 [第 38 頁的『UNIX, Linux, and Windows 上的錯誤日誌』](#)

對於每一個 IBM MQ 訊息佇列介面 (MQI) 及 IBM MQ 管理介面 (MQAI) 呼叫，佇列管理程式或結束常式會傳回完成碼及原因碼，以指出呼叫成功或失敗。如果應用程式取得回覆碼，指出「訊息佇列介面 (MQI)」呼叫失敗，請檢查原因碼以進一步瞭解問題。

如需原因碼清單，請參閱 [API 完成及原因碼](#)。

回覆碼的詳細資訊包含在每一個 MQI 呼叫的說明內。

相關資訊

[診斷訊息: AMQ4000-9999](#)

[PCF 原因碼](#)

[傳輸層安全 \(TLS\) 回覆碼](#)

[WCF 自訂通道異常狀況](#)

z/OS [IBM MQ for z/OS 訊息、完成及原因碼](#)
[疑難排解和支援參照](#)

ULW 您可以重新產生問題嗎?

如果您可以重新產生問題，請考量重新產生問題的條件：

- 它是由指令或同等的管理要求所造成嗎？

如果作業是由另一個方法輸入，則該作業是否正常運作？如果指令在指令行上輸入，但未輸入，則指令可以運作，請檢查指令伺服器是否未停止，以及 SYSTEM.ADMIN.COMMAND.QUEUE 尚未變更。

- 是程式造成的嗎？它是在所有 IBM MQ 系統及所有佇列管理程式上失敗，還是只在部分系統上失敗？
- 當發生問題時，您可以識別任何似乎一律在系統中執行的應用程式嗎？如果是這樣，請檢查應用程式，以查看它是否有錯誤。

Windows 在「Windows」上建立或啟動佇列管理程式時，您是否收到錯誤碼？

如果 IBM MQ Explorer 或 amqmdain 指令無法建立或啟動佇列管理程式，指出發生權限問題，可能是因為執行 IBM MQ Windows 服務的使用者權限不足。

確保配置 IBM MQ Windows 服務的使用者具有 [IBM MQ Windows 服務所需的使用者權限](#)中說明的權限。依預設，此服務會配置成以 MUSR_MQADMIN 使用者身分執行。對於後續安裝，「準備 IBM MQ 精靈」會建立名為 MUSR_MQADMINx 的使用者帳戶，其中 x 是下一個可用的數字，代表不存在的使用者 ID。

ULW 問題是否僅影響遠端佇列？

檢查問題是否僅影響遠端佇列的事項。

如果問題只影響遠端佇列，請執行下列檢查：

- 請確認必要通道已啟動、可以觸發，且任何必要的起始程式都在執行中。
- 請檢查應該將訊息放入遠端佇列的程式是否未報告問題。

- 如果您使用觸發來啟動分散式佇列程序，請檢查傳輸佇列是否已設定觸發。此外，請檢查觸發監視器是否在執行中。
- 請檢查錯誤日誌，以取得指出通道錯誤或問題的訊息。
- 必要的話，請手動啟動通道。

ULW 您是否取得不正確的輸出？

在此區段中，不正確的輸出是指您的應用程式：未收到您預期它會接收的訊息；接收包含非預期或毀損資訊的訊息；接收您預期它不會接收的訊息，例如，指定給不同應用程式的訊息。

未到達佇列的訊息

如果預期訊息未送達，請檢查下列各項：

- 訊息是否已順利放入佇列？
 - 是否已正確定義佇列？例如，MAXMSGL 是否足夠大？
 - 是否已啟用佇列放置？
 - 佇列是否已滿？
 - 是否有另一個應用程式取得佇列的專用存取權？
- 您可以從佇列中取得任何訊息嗎？
 - 您需要取得同步點嗎？

如果正在同步點內放置或擷取訊息，則在確定回復單元之前，其他作業無法使用這些訊息。
 - 您的等待間隔是否足夠長？

您可以將等待間隔設為 MQGET 呼叫的選項。請確定您等待回應的時間足夠長。
 - 您是否在等待由訊息或相關性 ID (*MsgId* 或 *CorrelId*) 識別的特定訊息？

請確認您正在等待具有正確 *MsgId* 或 *CorrelId* 的訊息。成功的 MQGET 呼叫會將這兩個值設為所擷取訊息的值，因此您可能需要重設這些值，才能順利取得另一個訊息。

此外，請檢查您是否可以從佇列取得其他訊息。
 - 其他應用程式可以從佇列取得訊息嗎？
 - 您預期的訊息定義為持續性嗎？

如果沒有，且 IBM MQ 已重新啟動，則訊息已遺失。
 - 是否有另一個應用程式取得佇列的專用存取權？

如果您找不到佇列的任何錯誤，且 IBM MQ 正在執行中，請檢查您預期將訊息放入佇列的處理程序，以取得下列各項：

- 應用程式是否已啟動？

如果應該已觸發它，請檢查是否已指定正確的觸發選項。
- 應用程式已停止嗎？
- 觸發監視器是否在執行中？
- 是否已正確定義觸發程式程序？
- 應用程式是否正確完成？

在工作日誌中尋找異常結束的證明。
- 應用程式已確定其變更，還是已取消？

如果多個交易負責處理佇列，則它們可能會彼此衝突。例如，假設一個交易發出緩衝區長度為零的 MQGET 呼叫，以找出訊息的長度，然後發出指定該訊息 *MsgId* 的特定 MQGET 呼叫。不過，在此期間，另一個交易會針對該訊息發出成功 MQGET 呼叫，因此第一個應用程式會收到原因碼 MQRC_NO_MSG_AVAILABLE。預期在多伺服器環境中執行的應用程式必須設計成處理此狀況。

請考量可能已收到訊息，但您的應用程式無法以某種方式處理該訊息。例如，訊息預期格式的錯誤是否導致您的程式拒絕它？如果是的話，請參閱這個主題中的後續資訊。

包含非預期或毀損資訊的訊息

如果訊息中包含的資訊不是您應用程式所預期的，或已在某些方面毀損，請考量下列事項：

- 您的應用程式或將訊息放入佇列的應用程式是否已變更？

請確定所有變更同時反映在所有需要注意變更的系統上。

例如，訊息資料的格式可能已變更，在此情況下，必須重新編譯這兩個應用程式以取得變更。如果一個應用程式尚未重新編譯，另一個應用程式會出現毀損的資料。

- 應用程式是否將訊息傳送至錯誤佇列？

請檢查應用程式正在接收的訊息是否預期用於服務不同佇列的應用程式。必要的話，請變更安全定義，以防止未獲授權的應用程式將訊息放入錯誤佇列。

如果您的應用程式使用別名佇列，請檢查別名是否指向正確的佇列。

- 是否已正確指定此佇列的觸發資訊？

請檢查您的應用程式是否應該已啟動；或是否應該已啟動不同的應用程式？

如果這些檢查無法讓您解決問題，請檢查應用程式邏輯，包括傳送訊息的程式，以及接收訊息的程式。

使用分散式佇列時輸出不正確的問題

如果您的應用程式使用分散式佇列，請考量下列要點：

- IBM MQ 是否已正確安裝在傳送端和接收端系統上，以及是否已正確配置分散式佇列？
- 兩個系統之間是否有鏈結可用？

檢查這兩個系統是否可用，並連接至 IBM MQ。請檢查兩個系統之間的連線是否處於作用中。

您可以針對佇列管理程式 (PING QMGR) 或通道 (PING CHANNEL) 使用 MQSC 指令 PING，以驗證鏈結可運作。

- 傳送系統中是否有觸發設定？
- 您正在等待來自遠端系統之回覆訊息的訊息嗎？

檢查是否在遠端系統中啟動觸發。

- 佇列是否已滿？

如果是的話，請檢查訊息是否已放入無法傳送郵件的佇列中。

無法傳送郵件的佇列標頭包含原因或回饋碼，說明訊息無法放入目標佇列的原因。如需無法傳送郵件的佇列標頭結構的相關資訊，請參閱 [使用無法傳送的郵件 \(未遞送的訊息\) 佇列](#) 及 [MQDLH-無法傳送的郵件標頭](#)。

- 傳送端和接收端佇列管理程式之間是否不符？

例如，訊息長度可能超過接收端佇列管理程式所能處理的長度。

- 傳送及接收通道的通道定義是否相容？

例如，序號折返中的不符可能停止分散式佇列元件。如需分散式佇列作業的相關資訊，請參閱 [分散式佇列及叢集](#)。

- 是否涉及資料轉換？如果傳送端與接收端應用程式之間的資料格式不同，則需要進行資料轉換。當發出 MQGET 呼叫時，如果將格式辨識為其中一種內建格式，則會發生自動轉換。

如果無法辨識轉換的資料格式，則會採用資料轉換結束程式，以容許您使用自己的常式執行轉換。

如需資料轉換的進一步相關資訊，請參閱 [資料轉換](#)。

ULW 您的部分佇列是否失敗?

如果您懷疑只有一部分佇列發生問題，請檢查您認為有問題的本端佇列。

執行下列檢查：

1. 顯示每一個佇列的相關資訊。您可以使用 MQSC 指令 DISPLAY QUEUE 來顯示資訊。
2. 使用顯示的資料來執行下列檢查：
 - 如果 CURDEPTH 是在 MAXDEPTH，則不會處理佇列。請檢查所有應用程式是否正常執行。
 - 如果 CURDEPTH 不是 MAXDEPTH，請檢查下列佇列屬性以確定它們是正確的：
 - 如果正在使用觸發：
 - 觸發監視器正在執行中嗎？
 - 觸發深度是否太大？也就是說，它是否經常產生觸發事件？
 - 程序名稱是否正確？
 - 程序是否可用且可運作？
 - 佇列可以共用嗎？如果沒有，則另一個應用程式可能已開啟它以供輸入。
 - 是否已針對 GET 及 PUT 適當地啟用佇列？
 - 如果沒有應用程式從佇列取得訊息，請判斷原因。可能是因為應用程式需要啟動、連線已中斷，或 MQOPEN 呼叫因某些原因而失敗。

請檢查佇列屬性 IPPROCS 及 OPPROCS。這些屬性指出是否已開啟佇列以供輸入及輸出。如果值為零，則表示無法執行該類型的作業。值可能已變更；佇列可能已開啟，但現在已關閉。

您需要在預期放置或取得訊息時檢查狀態。

如果您無法解決問題，請聯絡「IBM 支援中心」以取得協助。

ULW 您是否未收到來自 PCF 指令的回應?

如果您已發出指令但未收到回應，則考量。

如果您已發出指令，但尚未收到回應，請考量下列檢查：

- 指令伺服器是否在執行中？

使用 dspmqcsv 指令來檢查指令伺服器的狀態。

 - 如果此指令的回應指出指令伺服器不在執行中，請使用 strmqcsv 指令來啟動它。
 - 如果指令的回應指出 SYSTEM.ADMIN.COMMAND.QUEUE，請啟用 MQGET 要求的佇列。
- 回覆是否已傳送至無法傳送郵件的佇列？

無法傳送郵件的佇列標頭結構包含說明問題的原因或回饋碼。如需無法傳送郵件的佇列標頭結構 (MQDLH) 的相關資訊，請參閱 [MQDLH-無法傳送郵件的標頭](#) 及 [使用無法傳送郵件 \(未遞送訊息\) 佇列](#)。

如果無法傳送郵件的佇列包含訊息，您可以使用提供的瀏覽範例應用程式 (amqsbcg)，利用 MQGET 呼叫來瀏覽訊息。範例應用程式會逐步執行具名佇列管理程式之具名佇列上的所有訊息，同時顯示具名佇列上所有訊息的訊息描述子及訊息環境定義欄位。
- 訊息是否已傳送至錯誤日誌？

如需進一步資訊，請參閱第 40 頁的『UNIX, Linux, and Windows 上的錯誤日誌目錄』。
- 是否已針對放置及取得作業啟用佇列？
- *WaitInterval* 夠長嗎？

如果 MQGET 呼叫已逾時，則會傳回完成碼 MQCC_FAILED 及原因碼 MQRC_NO_MSG_AVAILABLE。(如需 *WaitInterval* 欄位的相關資訊，以及 MQGET 的完成碼和原因碼，請參閱 [WaitInterval \(MQLONG\)](#)。)
- 如果您使用自己的應用程式，將指令放置到 SYSTEM.ADMIN.COMMAND.QUEUE，您需要取得同步點嗎？

除非您已從同步點排除要求訊息，否則您需要在接收回覆訊息之前取得同步點。

- 佇列的 MAXDEPTH 及 MAXMSGL 屬性設定是否足夠高？
- 您是否正確使用 *CorrelId* 和 *MsgId* 欄位？

在應用程式中設定 *MsgId* 及 *CorrelId* 的值，以確保您從佇列接收所有訊息。

請嘗試停止指令伺服器，然後重新啟動它，以回應所產生的任何錯誤訊息。

如果系統仍未回應，則問題可能在於佇列管理程式或整個 IBM MQ 系統。首先，請嘗試停止個別佇列管理程式，以隔離失敗的佇列管理程式。如果此步驟未顯示問題，請嘗試停止並重新啟動 IBM MQ，以回應錯誤日誌中產生的任何訊息。

如果重新啟動之後仍發生問題，請聯絡「IBM 支援中心」以取得協助。

ULW 之前是否已順利執行應用程式？

請利用這個主題中的資訊來協助診斷應用程式的一般問題。

如果問題似乎涉及某個特定應用程式，請考量應用程式之前是否已順利執行。

在對此問題回答 **是** 之前，請考量下列事項：

- 自前次順利執行應用程式之後，對應用程式進行了任何變更嗎？
如果是這樣，則錯誤可能位於應用程式的新部分或已修改部分的某個位置。請查看變更，並查看您是否可以找到問題的明顯原因。可以使用前一層次的應用程式重試嗎？
- 應用程式的所有功能以前是否已完全執行過？
問題是否發生在第一次使用之前從未呼叫過的應用程式部分時？如果是這樣，則可能是應用程式的該部分發生錯誤。嘗試找出應用程式失敗時執行的動作，並檢查程式該部分中的原始碼是否有錯誤。
如果先前多次順利執行程式，請檢查現行佇列狀態，以及發生錯誤時正在處理的檔案。它們可能包含一些異常資料值，用來呼叫程式中的稀有使用路徑。
- 應用程式會檢查所有回覆碼嗎？
您的 IBM MQ 系統是否已變更 (可能以次要方式)，讓您的應用程式不會檢查它因變更而收到的回覆碼。例如，您的應用程式是否假設可以共用它所存取的佇列？如果佇列已重新定義為專用，您的應用程式是否可以處理回覆碼，指出它無法再存取該佇列？
- 應用程式是否在其他 IBM MQ 系統上執行？
可能是這個 IBM MQ 系統的設定方式造成問題的不同嗎？例如，是否已定義具有相同訊息長度或優先順序的佇列？

在您查看程式碼之前，並根據撰寫程式碼的程式設計語言，檢查轉換器或編譯器和鏈結編輯器的輸出，以查看是否已報告任何錯誤。

如果您的應用程式無法轉換、編譯或鏈結編輯至載入程式庫，則如果您嘗試呼叫它，它也會無法執行。如需建置應用程式的相關資訊，請參閱 [開發應用程式](#)。

如果文件顯示每一個步驟都已完成且沒有錯誤，請考量應用程式的編碼邏輯。問題的症狀是否指出失敗的功能，因而指出錯誤的程式碼片段？請參閱下一節，以取得導致 IBM MQ 應用程式發生問題的一些常見錯誤範例。

一般程式設計錯誤

下列清單中的錯誤說明執行 IBM MQ 程式時所發生問題的最常見原因。請考量下列一或多個錯誤可能導致 IBM MQ 系統發生問題的可能性：

- 假設佇列可以共用，但實際上它們是專用的。
- 在 MQI 呼叫中傳遞不正確的參數。
- 在 MQI 呼叫中傳遞的參數不足。這可能表示 IBM MQ 無法設定應用程式要處理的完成碼及原因碼。
- 無法檢查 MQI 要求的回覆碼。
- 傳遞指定了不正確長度的變數。

- 以錯誤順序傳遞參數。
- 無法正確起始設定 *MsgId* 和 *CorrelId*。
- 無法在 MQRC_TRUNCATED_MSG_ACCEPTED 之後起始設定 *Encoding* 和 *CodedCharSetId*。

ULW 您的應用程式或系統執行緩慢嗎？

如果您的應用程式執行緩慢，則可能是在迴圈中，或等待無法使用的資源，或可能有效能問題。

您的系統運作可能已接近其容量限制。這種類型的問題可能在尖峰系統負載時間最差，通常是在上午中及下午中。（如果您的網路延伸到多個時區，則其他時間可能會出現尖峰系統負載。）

硬體限制可能導致效能問題。

如果您發現效能降低並不取決於系統負載，但有時會在系統負載較輕時發生，可能是設計不佳的應用程式所造成。這可能是只有在存取特定佇列時才會發生的問題。

如果效能問題持續存在，問題可能在於 IBM MQ 本身。如果您懷疑這樣做，請聯絡「IBM 支援中心」以取得協助。

應用程式效能變慢或在佇列（通常是傳輸佇列）上建置訊息的常見原因是有一或多個應用程式在工作單元之外寫入持續訊息；如需相關資訊，請參閱 [訊息持續性](#)。

ULW 問題會影響網路的特定部分嗎？

您可能能夠識別受問題影響的網路特定部分（例如，遠端佇列）。如果遠端訊息佇列管理程式的鏈結無法運作，則訊息無法流向遠端佇列。

檢查兩個系統之間的連線是否可用，以及 IBM MQ 的交互通訊元件是否已啟動。

請檢查訊息是否到達傳輸佇列，並檢查傳輸佇列及任何遠端佇列的本端佇列定義。

您是否已進行任何網路相關變更，或已變更任何可能導致問題的 IBM MQ 定義？

ULW 問題是否在一天中的特定時間發生？

如果問題發生在一天中的特定時間，則可能是取決於系統載入。通常，尖峰系統負載是在上午中和下午中，因此這些是最有可能發生負載相依問題的時間。（如果您的 IBM MQ 網路延伸到多個時區，則尖峰系統負載可能似乎在一天中的某個其他時間發生。）

ULW 問題是否間歇性？

間歇性問題可能是由處理程序彼此獨立執行方式所造成。例如，在先前的處理程序完成之前，程式可能會發出 MQGET 呼叫，但未指定等待選項。如果您的應用程式在確定放置訊息的呼叫之前嘗試從佇列取得訊息，也可能會看到間歇性問題。

IBM i 對 IBM i 進行起始檢查

在 IBM i 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

關於這項作業

您問題的原因可能是下列任何一項：

- 硬體
- 作業系統
- 相關軟體，例如語言編譯器
- 網路
- IBM MQ 產品

- 您的 IBM MQ 應用程式
- 其他應用程式
- 網站作業程序

下列程序列出您要考量的一些初步問題。如果您能夠透過執行這些初步檢查來找出問題的原因，則必要的話，您可以使用 IBM MQ 產品說明文件其他小節中的資訊，以及其他授權程式檔案庫中的資訊，來協助您解決問題。

如果您無法透過執行初步檢查來識別問題的原因，因此需要執行更詳細的調查，則在子主題中還有其他問題可供您考量。在您處理問題清單時，請記下可能與問題相關的任何事項。即使您的觀察沒有立即提出原因，但如果您必須進行有系統的問題判斷工作，則稍後可能會有用。

程序

- 請考量下列問題。

下列步驟旨在協助您隔離問題，並從 IBM MQ 應用程式的觀點來採取。在每一個階段檢查所有建議。

1. IBM MQ for IBM i 之前是否已順利執行？

是

繼續進行步驟 [第 17 頁的『2』](#)。

否

您可能未正確安裝或設定 IBM MQ。

2. 之前是否已順利執行 IBM MQ 應用程式？

是

繼續進行步驟 [第 17 頁的『3』](#)。

否

請考量下列項目：

- a. 應用程式可能無法編譯或鏈結，如果您嘗試呼叫它，則會失敗。請檢查編譯器或鏈結器的輸出。
如需如何建置應用程式的相關資訊，請參閱適當的程式設計語言參照資訊，或參閱 [開發應用程式](#)。
- b. 請考量應用程式的邏輯。例如，問題的症狀是否指出函數失敗，因此程式碼片段錯誤。
請檢查下列一般程式設計錯誤：
 - 假設佇列可以共用，但實際上它們是專用的。
 - 嘗試在沒有正確安全授權的情況下存取佇列及資料。
 - 在 MQI 呼叫中傳遞不正確的參數；如果傳遞的參數數目錯誤，則無法嘗試完成完成碼及原因碼欄位，作業會異常結束。
 - 無法檢查 MQI 要求的回覆碼。
 - 使用不正確的位址。
 - 傳遞指定了不正確長度的變數。
 - 以錯誤順序傳遞參數。
 - 無法正確起始設定 *MsgId* 和 *CorrelId*。

3. 自前次順利執行以來，IBM MQ 應用程式是否已變更？

是

可能是應用程式的新部分或已修改部分發生錯誤。請檢查所有變更，並查看您是否可以找到問題的明顯原因。

- a. 應用程式的所有功能以前是否已完全執行過？

問題是否發生在第一次使用之前從未呼叫過的應用程式部分時？如果是這樣，則可能是應用程式的該部分發生錯誤。嘗試找出應用程式失敗時執行的動作，並檢查程式該部分中的原始碼是否有錯誤。

- b. 如果之前已順利執行程式，請檢查發生錯誤時正在處理的現行佇列狀態及檔案。它們可能包含一些不尋常的資料值，會導致呼叫程式中很少使用的路徑。
- c. 應用程式收到非預期的 MQI 回覆碼。例如：
 - 您的應用程式是否假設它所存取的佇列是可共用的？如果佇列已重新定義為專用，您的應用程式是否可以處理回覆碼，指出它無法再存取該佇列？
 - 是否已變更任何佇列定義或安全設定檔？MQOPEN 呼叫可能因安全違規而失敗；您的應用程式可以從產生的回覆碼回復嗎？如需每一個回覆碼的說明，請參閱 [MQI 應用程式參照](#)，以取得您的程式設計語言。
- d. 如果您已將任何 PTF 套用至 IBM MQ for IBM i，請檢查您在安裝 PTF 時未收到任何錯誤訊息。

否

請確定您已刪除所有先前的建議，並繼續進行步驟 [第 18 頁的『4』](#)。

4. 自前次順利執行之後，伺服器系統是否維持不變？

是

繼續進行 [第 18 頁的『在 IBM i 上識別問題的性質』](#)。

否

請考量系統的所有層面，並檢閱適當的文件，以瞭解變更可能如何影響 IBM MQ 應用程式。例如：

- 與其他應用程式的介面
- 安裝新的作業系統或硬體
- PTF 的應用程式
- 作業程序變更

下一步

相關工作

[第 21 頁的『手動套用指令及程式的必要權限』](#)

部分 IBM MQ 指令依賴使用 IBM i 系統指令來建立及管理物件、檔案及檔案庫，例如 CRTMQM (建立佇列管理程式) 及 DLTMQM (刪除佇列管理程式)。同樣地，部分 IBM MQ 程式碼 (例如佇列管理程式) 依賴使用 IBM i 系統程式。

[第 9 頁的『對 UNIX, Linux, and Windows 進行起始檢查』](#)

在 UNIX, Linux, and Windows 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

[第 24 頁的『對 z/OS 進行起始檢查』](#)

在 z/OS 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

[第 36 頁的『聯絡 IBM 支援中心』](#)

如果您在使用 IBM MQ 時遇到問題需要協助，您可以透過「IBM 支援中心網站」聯絡 IBM 支援中心。您也可以訂閱 IBM MQ 修正程式、疑難排解及其他新聞的相關通知。

相關參考

[第 22 頁的『判斷應用程式、指令及訊息的問題』](#)

如果您遇到 IBM MQ 應用程式、指令及訊息的問題，您可以考量一些問題，以協助您判斷問題的原因。

相關資訊

[訊息及原因碼](#)

[PCF 原因碼](#)

[疑難排解和支援參照](#)

IBM i 在 IBM i 上識別問題的性質

如果您無法使用初步檢查來識別問題的原因，現在應該開始更詳細地查看問題的性質。

請使用下列問題作為指標，以協助您識別問題的原因：

- [第 19 頁的『您可以重新產生問題嗎?』](#)
- [第 19 頁的『問題是否間歇性?』](#)
- [第 19 頁的『指令的問題』](#)
- [第 19 頁的『問題是否會影響 IBM MQ for IBM i 應用程式的所有使用者?』](#)
- [第 20 頁的『問題會影響網路的特定部分嗎?』](#)
- [第 20 頁的『問題是否僅在 IBM MQ 上發生?』](#)
- [第 20 頁的『問題是否在一天中的特定時間發生?』](#)
- [第 20 頁的『您是否無法收到來自指令的回應?』](#)

您可以重新產生問題嗎?

如果您可以重新產生問題，請考量您執行此動作的條件：

- 是命令造成的嗎？
如果作業是由另一個方法輸入，則該作業是否正常運作？如果指令在指令行上輸入，但未在其他情況下輸入，則指令可以運作，請檢查指令伺服器是否未停止。您也必須檢查 `SYSTEM.ADMIN.COMMAND.QUEUE` 的佇列定義是否未變更。
- 是程式造成的嗎？如果是的話，是否會批次失敗？它是在所有 IBM MQ for IBM i 系統上失敗，還是僅在部分系統上失敗？
- 當發生問題時，您可以識別任何似乎一律在系統中執行的應用程式嗎？如果是這樣，請檢查應用程式，以查看它是否有錯誤。
- 是任何佇列管理程式發生問題，還是連接至某個特定佇列管理程式時發生問題？
- 問題是否發生在任何佇列管理程式上的相同類型物件，或只有一個特定物件？清除或重新定義此物件之後會發生什麼情況？
- 問題是否與任何訊息持續性設定無關？
- 只有在使用同步點時才會發生問題嗎？
- 只有在啟用一或多個佇列管理程式事件時，才會發生問題嗎？

問題是否間歇性?

間歇性問題可能是由於未考量處理程序可以彼此獨立執行的事實所導致。例如，在先前的處理程序完成之前，程式可能在未指定等待選項的情況下發出 `MQGET` 呼叫。如果您的應用程式嘗試在放置訊息的呼叫不確定時（亦即，在確定或取消之前）從佇列取得訊息，則也可能會遇到此問題。

指令的問題

使用此資訊可避免特殊字元的潛在問題。包括特殊字元時請小心，例如反斜線 (\) 及引號 (") 字元，在部分指令的說明文字中。如果您在說明文字中使用下列任一字元，請在它們之前加上反斜線 (\) 字元，例如：

- 如果文字中需要反斜線 (\) 字元，請輸入 \\。
- 如果您需要引號 (")，請輸入 \" 字元。

佇列管理程式及其相關聯的物件名稱區分大小寫。依預設，除非您以撇號 (') 字元括住名稱，否則 IBM i 會使用大寫字元。

例如，`MYQUEUE` 和 `myqueue` 轉換為 `MYQUEUE`，而 `'myqueue'` 轉換為 `myqueue`。

問題是否會影響 IBM MQ for IBM i 應用程式的所有使用者?

如果問題僅影響部分使用者，請在使用者配置其系統及佇列管理程式設定的方式中尋找差異。

請檢查檔案庫清單及使用者設定檔。是否可以透過具有 `*ALLOBJ` 權限來規避問題？

問題會影響網路的特定部分嗎？

您可能能夠識別受問題影響的網路特定部分 (例如，遠端佇列)。如果遠端訊息佇列管理程式的鏈結無法運作，則訊息無法流向遠端佇列。

請檢查下列要點：

- 兩個系統之間的連線是否可用，以及是否已啟動 IBM MQ for IBM i 的交互通訊元件？
請檢查訊息是否已到達傳輸佇列、傳輸佇列的本端佇列定義，以及任何遠端佇列。
- 您是否已進行任何可能導致問題或變更任何 IBM MQ for IBM i 定義的網路相關變更？
- 您可以區分通道定義問題與通道訊息問題嗎？
例如，重新定義通道以使用空的傳輸佇列。如果通道正確啟動，則會正確配置定義。

問題是否僅在 IBM MQ 上發生？

如果問題僅在此版本的 IBM MQ 上發生，請檢查 RETAIN 或 https://www.ibm.com/support/entry/portal/Overview/Software/WebSphere/WebSphere_MQ 上的適當資料庫，以確保您已套用所有相關 PTF。

問題是否在一天中的特定時間發生？

如果問題發生在一天中的特定時間，則可能是取決於系統載入。通常，尖峰系統負載是在上午中和下午中，因此這些時間是最可能發生負載相依問題的時間。(如果您的 IBM MQ for IBM i 網路延伸到多個時區，則尖峰系統負載可能似乎在一天中的某個其他時間發生。)

您是否無法收到來自指令的回應？

如果您已發出指令，但尚未收到回應，請考量下列問題：

- 指令伺服器是否在執行中？
使用 DSPMQMSVR 指令來檢查指令伺服器的狀態。
 - 如果此指令的回應指出指令伺服器不在執行中，請使用 STRMQMSVR 指令來啟動它。
 - 如果指令的回應指出 SYSTEM.ADMIN.COMMAND.QUEUE，請啟用 MQGET 要求的佇列。
- 回覆是否已傳送至無法傳送郵件的佇列？
無法傳送郵件的佇列標頭結構包含說明問題的原因或回饋碼。如需無法傳送郵件的佇列標頭結構 (MQDLH) 的相關資訊，請參閱 [MQDLH-無法傳送郵件的標頭](#)。
如果無法傳送郵件的佇列包含訊息，您可以使用提供的瀏覽範例應用程式 (amqsbcbg)，利用 MQGET 呼叫來瀏覽訊息。範例應用程式會逐步執行具名佇列管理程式之具名佇列上的所有訊息，同時顯示具名佇列上所有訊息的訊息描述子及訊息環境定義欄位。
- 訊息是否已傳送至錯誤日誌？
如需進一步資訊，請參閱第 41 頁的『IBM i 上的錯誤日誌』。
- 是否已針對放置及取得作業啟用佇列？
- *WaitInterval* 夠長嗎？
如果 MQGET 呼叫已逾時，則會傳回完成碼 MQCC_FAILED 及原因碼 MQRC_NO_MSG_AVAILABLE。(如需 *WaitInterval* 欄位以及 MQGET 的完成碼和原因碼的相關資訊，請參閱 [使用 MQGET 呼叫從佇列取得訊息](#)。)
- 如果您使用自己的應用程式，將指令放置到 SYSTEM.ADMIN.COMMAND.QUEUE，您需要取得同步點嗎？
除非您已從同步點排除要求訊息，否則必須先取得同步點，然後再嘗試接收回覆訊息。
- 佇列的 MAXDEPTH 及 MAXMSGL 屬性設定是否足夠高？
- 您是否正確使用 *CorrelId* 和 *MsgId* 欄位？
在應用程式中設定 *MsgId* 及 *CorrelId* 的值，以確保您從佇列接收所有訊息。

相關概念

第 7 頁的『IBM MQ 疑難排解及支援』

如果您的佇列管理程式網路或 IBM MQ 應用程式有問題，請使用說明的技術來協助您診斷及解決問題。

相關工作

第 21 頁的『手動套用指令及程式的必要權限』

部分 IBM MQ 指令依賴使用 IBM i 系統指令來建立及管理物件、檔案及檔案庫，例如 CRTMQM (建立佇列管理程式) 及 DLTMQM (刪除佇列管理程式)。同樣地，部分 IBM MQ 程式碼 (例如佇列管理程式) 依賴使用 IBM i 系統程式。

相關參考

第 22 頁的『判斷應用程式、指令及訊息的問題』

如果您遇到 IBM MQ 應用程式、指令及訊息的問題，您可以考量一些問題，以協助您判斷問題的原因。

IBM i 手動套用指令及程式的必要權限

部分 IBM MQ 指令依賴使用 IBM i 系統指令來建立及管理物件、檔案及檔案庫，例如 CRTMQM (建立佇列管理程式) 及 DLTMQM (刪除佇列管理程式)。同樣地，部分 IBM MQ 程式碼 (例如佇列管理程式) 依賴使用 IBM i 系統程式。

關於這項作業

若要啟用此依賴，指令及程式必須具有 *PUBLIC *USE 權限，或對 IBM MQ 使用者設定檔 QMQM 及 QMQMADM 的明確 *USE 權限。

在安裝程序中，會自動套用這類權限，您不需要自行套用。不過，如果您遇到問題，則可以手動設定權限，如下列步驟中所述。

程序

1. 使用具有 OBJTYPE (*CMD) 參數的 GRTOBJAUT 來設定指令的權限，例如：

```
GRTOBJAUT OBJ(QSYS/ADDLIB) OBJTYPE(*CMD) USER(QMQMADM) AUT(*USE)
```

您可以設定下列指令的權限：

- QSYS/ADDLIB
- QSYS/ADDPFM
- QSYS/CALL
- QSYS/CHGCURLIB
- QSYS/CHGJOB
- QSYS/CRTJRN
- QSYS/CRTJRNRV
- QSYS/CRTJOBQ
- QSYS/CRTJOB
- QSYS/CRTLIB
- QSYS/CRTMSGQ
- QSYS/CRTPF
- QSYS/CRTPGM
- QSYS/CRTSRCPF
- QSYS/DLTJRN
- QSYS/DLTJRNRV
- QSYS/DLTLIB

- QSYS/DLTMSGQ
- QSYS/OVRPRTF
- QSYS/RCLACTGRP
- QSYS/RTVJRNE
- QSYS/RCVJRNE
- QSYS/SBMJOB

2. 使用具有 OBJTYPE (*PGM) 參數的 GRTOBJAUT 來設定程式的權限，例如：

```
GRTOBJAUT OBJ(QSYS/QWTSETP) OBJTYPE(*PGM) USER(QMQADM) AUT(*USE)
```

您可以設定下列程式的權限：

- QSYS/QWTSETP (*PGM)
- QSYS/QSYRLSPH (*PGM)
- QSYS/QSYGETPH (*PGM)

IBM i 判斷應用程式、指令及訊息的問題

如果您遇到 IBM MQ 應用程式、指令及訊息的問題，您可以考量一些問題，以協助您判斷問題的原因。請使用下列問題作為指標，以協助您識別問題的原因：

您的部分佇列是否在運作中？

如果您懷疑只有一部分佇列發生問題，請選取您認為有問題的本端佇列名稱。

1. 使用 WRKMQSTS 或 DSPMQMQ 來顯示此佇列的相關資訊。
2. 使用顯示的資料來執行下列檢查：
 - 如果 CURDEPTH 是在 MAXDEPTH，則不會處理佇列。請檢查所有應用程式是否正常執行。
 - 如果 CURDEPTH 不是 MAXDEPTH，請檢查下列佇列屬性以確定它們是正確的：
 - 如果正在使用觸發：
 - 觸發監視器正在執行中嗎？
 - 觸發深度是否太大？
 - 程序名稱是否正確？
 - 佇列可以共用嗎？否則，另一個應用程式可能已開啟它以供輸入。
 - 是否已針對 GET 及 PUT 適當地啟用佇列？
 - 如果沒有應用程式程序從佇列取得訊息，請判斷原因 (例如，因為應用程式必須啟動、連線已中斷，或 MQOPEN 呼叫因故失敗)。

如果無法解決問題，請聯絡 IBM 支援中心以取得協助。

問題是否僅影響遠端佇列？

如果問題只影響遠端佇列，請檢查後續的點：

1. 請檢查應該將訊息放入遠端佇列的程式是否已順利執行。
2. 如果您使用觸發來啟動分散式佇列程序，請檢查傳輸佇列是否已設定觸發。此外，請檢查觸發監視器是否在執行中。
3. 必要的話，請手動啟動通道。請參閱 [分散式佇列及叢集](#)。
4. 請使用 PING 指令檢查通道。

訊息是否無法抵達佇列?

如果預期訊息未送達，請檢查下列各項：

- 您是否已選取正確的佇列管理程式，即預設佇列管理程式或具名佇列管理程式？
- 訊息是否已順利放入佇列？
 - 是否已正確定義佇列，例如 MAXMSGLEN 是否足夠大？
 - 應用程式是否可以將訊息放置在佇列上 (佇列是否已啟用放置)？
 - 如果佇列已滿，可能表示應用程式無法將必要的訊息放置在佇列上。
- 您可以從佇列中取得訊息嗎？
 - 您必須取得同步點嗎？

如果正在同步點內放置或擷取訊息，則在確定回復單元之前，其他作業無法使用這些訊息。
 - 您的逾時間隔是否足夠長？
 - 您是否在等待由訊息 ID 或相關性 ID (*MsgId* 或 *CorrelId*) 識別的特定訊息？

請確認您正在等待具有正確 *MsgId* 或 *CorrelId* 的訊息。成功的 MQGET 呼叫會將這兩個值設為所擷取訊息的值，因此您可能需要重設這些值，才能順利取得另一個訊息。

也請檢查您是否可以從佇列中取得其他訊息。
 - 其他應用程式可以從佇列取得訊息嗎？
 - 您預期的訊息定義為持續性嗎？

如果沒有，且 IBM MQ for IBM i 已重新啟動，則訊息已遺失。

如果您找不到佇列的任何錯誤，且佇列管理程式本身正在執行中，請對您預期將訊息放入佇列的處理程序進行下列檢查：

- 應用程式是否已啟動？

如果應該已觸發它，請檢查是否已指定正確的觸發選項。
- 觸發監視器是否在執行中？
- 是否已正確定義觸發程式程序？
- 它是否正確完成？

在工作日誌中尋找異常結束的證明。
- 應用程式已確定其變更，還是已取消？

如果多個交易負責處理佇列，則它們有時可能會彼此衝突。例如，一個交易可能發出緩衝區長度為零的 MQGET 呼叫，以找出訊息的長度，然後發出指定該訊息 *MsgId* 的特定 MQGET 呼叫。不過，在此期間，另一個交易可能已對該訊息發出成功 MQGET 呼叫，因此第一個應用程式會收到完成碼 MQRC_NO_MSG_AVAILABLE。預期在多台伺服器環境中執行的應用程式必須設計成處理此狀況。

請考量可能已收到訊息，但您的應用程式無法以某種方式處理它。例如，訊息預期格式的錯誤是否導致您的程式拒絕它？如果是這樣，請參閱第 24 頁的『使用分散式佇列時是否收到非預期的訊息?』。

訊息包含非預期或毀損的資訊嗎？

如果訊息中包含的資訊不是您應用程式所預期的，或已在某種程度上毀損，請考量下列要點：

- 您的應用程式或將訊息放入佇列的應用程式是否已變更？

請確定所有變更同時反映在所有需要注意變更的系統上。

例如，將訊息格式化的副本檔可能已變更，在此情況下，請重新編譯這兩個應用程式，以挑選變更。如果一個應用程式尚未重新編譯，則另一個應用程式的資料似乎已毀損。
- 應用程式是否將訊息傳送至錯誤佇列？

請檢查應用程式正在接收的訊息是否預期用於服務不同佇列的應用程式。必要的話，請變更安全定義，以防止未獲授權的應用程式將訊息放入錯誤佇列。

如果您的應用程式已使用別名佇列，請檢查別名是否指向正確的佇列。

- 是否已正確指定此佇列的觸發資訊？

請檢查您的應用程式是否應該已啟動，或是否應該已啟動不同的應用程式？

- 是否已正確設定 CCSID，或訊息格式因資料轉換而不正確。

如果這些檢查無法讓您解決問題，請檢查應用程式邏輯，包括傳送訊息的程式，以及接收訊息的程式。

使用分散式佇列時是否收到非預期的訊息？

如果您的應用程式使用分散式佇列，請考量下列要點：

- 傳送及接收系統上是否已正確安裝分散式佇列？
- 兩個系統之間是否有鏈結可用？

檢查這兩個系統是否可用，並連接至 IBM MQ for IBM i。請檢查兩個系統之間的連線是否處於作用中。

- 傳送系統中是否有觸發設定？
- 您正在等待來自遠端系統的回覆訊息嗎？

檢查是否在遠端系統中啟動觸發。

- 佇列是否已滿？

若是如此，可能表示應用程式無法將必要的訊息放到佇列中。請檢查訊息是否已放入無法遞送的訊息佇列中。

無法傳送郵件的佇列訊息標頭 (無法傳送郵件的標頭結構) 包含原因或回饋碼，說明無法將訊息放入目標佇列的原因。如需無法傳送的郵件標頭結構的相關資訊，請參閱 [MQDLH-無法傳送的郵件標頭](#)。

IBM i 若為 IBM i，另請參閱 [IBM i Application Programming Reference \(ILE/RPG\)](#)。

- 傳送端和接收端佇列管理程式之間是否不符？

例如，訊息長度可能超過接收端佇列管理程式所能處理的長度。

- 傳送及接收通道的通道定義是否相容？

例如，序號折返中的不符會停止分散式佇列元件。請參閱 [分散式佇列及叢集](#)。

z/OS 對 z/OS 進行起始檢查

在 z/OS 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

關於這項作業

您問題的原因可能是：

- IBM MQ
- 網路
- 應用程式
- 您已配置為使用 IBM MQ 的其他應用程式

程序

- 請考量下列問題清單。在您瀏覽清單時，請記下可能與問題相關的任何事項。即使您的觀察沒有立即提出原因，但如果您必須進行有系統的問題判斷工作，則稍後可能會有用。
 - [第 25 頁的『IBM MQ for z/OS 之前是否已順利執行?』](#)
 - [第 26 頁的『您是否已套用任何 APAR 或 PTF?』](#)
 - [第 26 頁的『是否有任何錯誤訊息、回覆碼或其他錯誤狀況?』](#)
 - [第 27 頁的『您的應用程式或 IBM MQ for z/OS 是否已停止處理工作?』](#)

- [第 28 頁的『IBM MQ 佇列是否有問題?』](#)
- [第 28 頁的『您的部分佇列是否在運作中?』](#)
- [第 29 頁的『是否已定義正確的佇列?』](#)
- [第 29 頁的『問題是否僅影響遠端或叢集佇列?』](#)
- [第 30 頁的『問題是否只影響共用佇列?』](#)
- [第 30 頁的『問題會影響網路的特定部分嗎?』](#)
- [第 31 頁的『在一天中特定時間發生或影響特定使用者的問題』](#)
- [第 31 頁的『問題是間歇性，還是所有 z/OS、CICS 或 IMS 系統都發生問題?』](#)
- [第 31 頁的『之前是否已順利執行應用程式?』](#)
- [第 32 頁的『自前次成功執行以來是否已進行任何變更?』](#)
- [第 33 頁的『您有程式錯誤嗎?』](#)
- [第 33 頁的『有異常終止嗎?』](#)
- [第 34 頁的『您是否取得不正確的輸出?』](#)
- [第 34 頁的『您可以重新產生問題嗎?』](#)
- [第 35 頁的『您是否無法接收來自 MQSC 指令的回應?』](#)
- [第 36 頁的『您的應用程式或 IBM MQ for z/OS 執行緩慢嗎?』](#)

相關工作

[第 9 頁的『對 UNIX, Linux, and Windows 進行起始檢查』](#)

在 UNIX, Linux, and Windows 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

[第 16 頁的『對 IBM i 進行起始檢查』](#)

在 IBM i 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

[第 36 頁的『聯絡 IBM 支援中心』](#)

如果您在使用 IBM MQ 時遇到問題需要協助，您可以透過「IBM 支援中心網站」聯絡 IBM 支援中心。您也可以訂閱 IBM MQ 修正程式、疑難排解及其他新聞的相關通知。

相關資訊

[訊息及原因碼](#)

[PCF 原因碼](#)

[疑難排解和支援參照](#)

IBM MQ for z/OS 之前是否已順利執行?

知道 IBM MQ for z/OS 是否已順利執行之前，可以協助進行問題判斷，而且您可以執行一些檢查來協助您。

如果此問題的答案是 否，請考量下列事項：

- 請檢查您的設定。
如果之前未在 z/OS 上順利執行 IBM MQ，則可能是您尚未正確設定它。如需進一步指引，請參閱 [安裝 IBM MQ for z/OS 產品](#) 中關於安裝及自訂佇列管理程式的資訊。
- 驗證安裝。
- 檢查是否已發出 CSQ9022I 訊息來回應 START QMGR 指令 (指示正常完成)。
- 確保 z/OS 將 IBM MQ 顯示為已安裝的子系統。若要判斷 IBM MQ 是否為已安裝的子系統，請使用 z/OS 指令 D OPDATA。
- 檢查安裝驗證程式 (IVP) 是否順利執行。
- 發出指令 DISPLAY DQM，以檢查通道起始程式位址空間是否在執行中，以及是否已啟動適當的接聽器。

您是否已套用任何 APAR 或 PTF?

APAR 及 PTF 有時會導致 IBM MQ 發生非預期的問題。這些修正程式可能已套用至 IBM MQ 或其他 z/OS 系統。

如果 APAR 或 PTF 已套用至 IBM MQ for z/OS，請檢查未產生任何錯誤訊息。如果安裝成功，請檢查 IBM 支援中心是否有任何 APAR 或 PTF 錯誤。

如果 APAR 或 PTF 已套用至任何其他產品，請考量它可能對 IBM MQ 與它互動的方式產生的影響。

請確定您已遵循 APAR 中任何會影響系統的指示。(例如，您可能必須重新定義資源。)

是否有任何錯誤訊息、回覆碼或其他錯誤狀況?

請利用這個主題來調查佇列管理程式或通道起始程式終止的錯誤訊息、回覆碼及狀況。

問題可能會產生下列類型的錯誤訊息或回覆碼:

CSQ 訊息及原因碼

IBM MQ for z/OS 錯誤訊息具有字首 CSQ。如果您收到任何具有此字首的訊息(例如，在主控台日誌或 CICS 日誌中)，請參閱 [IBM MQ for z/OS 訊息、完成及原因碼](#)，以取得說明。

其他訊息

如需具有不同字首的訊息，請查看適當的訊息及訊息碼主題，以取得建議的動作方針。

異常訊息

請注意與 IBM MQ for z/OS 啟動相關聯的異常訊息，或在發生錯誤之前系統執行時發出的異常訊息。任何不尋常的訊息都可能指出導致應用程式無法順利執行的某個系統問題。

應用程式 MQI 回覆碼

如果您的應用程式取得回覆碼，指出 MQI 呼叫失敗，請參閱 [回覆碼](#)，以取得該回覆碼的說明。

您是否收到非預期的錯誤訊息或回覆碼?

如果您的應用程式收到非預期的錯誤訊息，請考量錯誤訊息是來自 IBM MQ 還是來自另一個程式。

IBM MQ 錯誤訊息

IBM MQ for z/OS 錯誤訊息會以字母 CSQ 作為字首。

如果您收到非預期的 IBM MQ 錯誤訊息(例如，在主控台日誌或 CICS 日誌中)，請參閱 [IBM MQ for z/OS 訊息、完成及原因碼](#)，以取得說明。

IBM MQ for z/OS 訊息、完成及原因碼可能會提供您足夠的資訊來快速解決問題，或者它可能會將您重新導向至另一個手冊以取得進一步指引。如果無法處理訊息，您可能需要聯絡 IBM 支援中心，以取得協助。

非 IBM MQ 錯誤訊息

如果您從另一個 IBM 程式或作業系統取得錯誤訊息，請查看適當程式庫中的訊息與訊息碼手冊，以取得其意義的說明。

在佇列共用環境中，尋找下列錯誤訊息:

- XES (字首為字母 IXL)
- Db2 (字首為字母 DSN)
- RRS (字首為字母 ATR)

非預期的回覆碼

如果您的應用程式收到來自 IBM MQ 的非預期回覆碼，請參閱 [回覆碼](#)，以取得應用程式如何處理 IBM MQ 回覆碼的相關資訊。

檢查錯誤訊息

請發出 DISPLAY THREAD (*) 指令來檢查佇列管理程式是否在執行中。如需指令的相關資訊，請參閱 DISPLAY THREAD。如果佇列管理程式已停止執行，請尋找任何可能說明狀況的訊息。如果您使用作業及控制台，則會在 z/OS 主控台或終端機上顯示訊息。請使用 DISPLAY DQM 指令來查看通道起始程式是否在運作中，以及接聽器是否在作用中。z/OS 指令

```
DISPLAY R,L
```

列出具有未完成回覆的訊息。請查看其中是否有任何回覆相關。在某些情況下，例如，當它已使用所有作用中日誌時，IBM MQ for z/OS 會等待操作員介入。

未發出任何錯誤訊息

如果未發出任何錯誤訊息，請執行下列程序來判斷造成問題的原因：

1. 發出 z/OS 指令

```
DISPLAY A,xxxxMSTR  
DISPLAY A,xxxxCHIN
```

(其中 xxxx 是 IBM MQ for z/OS 子系統名稱)。如果您收到訊息，指出找不到佇列管理程式或通道起始程式，則此訊息指出子系統已終止。此狀況可能是由異常終止或系統的操作員關機所造成。

2. 如果子系統正在執行中，您會收到 IEE105I 訊息。此訊息包括 CT = nnnn 欄位，其中包含子系統所使用處理器時間的相關資訊。請記下此欄位的值，然後重新發出指令。

- 如果 CT = 值未變更，則表示子系統未使用任何處理器時間。這可能表示子系統處於等待狀態 (或沒有工作可執行)。如果您可以發出 DISPLAY DQM 之類的指令並重新取得輸出，這表示沒有工作可執行，而不是有當掉狀況。
- 如果 CT = 值已大幅變更，並在重複的顯示畫面上繼續這麼做，這可能表示子系統忙碌或可能陷入迴圈。
- 如果回覆指出現在找不到子系統，則指出在發出第一個指令時正在終止子系統。如果正在取得傾出，子系統可能需要一些時間來終止。在終止之前，會在主控台產生訊息。

如果要檢查通道起始程式是否在運作中，請發出 DISPLAY DQM 指令。如果回應未顯示通道起始程式在運作中，則可能是因為它取得的資源不足 (例如處理器)。在此情況下，請使用 z/OS 監視工具 (例如 RMF) 來判斷是否有資源問題。如果不是，請重新啟動通道起始程式。

佇列管理程式或通道起始程式是否已異常終止？

尋找指出佇列管理程式或通道起始程式位址空間已異常終止的任何訊息。如果您收到系統動作要終止 IBM MQ 的訊息，請找出是否產生系統傾出，請參閱 [IBM MQ 傾出](#)。

IBM MQ for z/OS 可能仍在執行中

另請考量 IBM MQ for z/OS 可能仍在執行中，但速度緩慢。如果它執行緩慢，您可能有效能問題。若要確認此項，請參閱 [是您的應用程式還是 IBM MQ for z/OS 慢速執行](#)。如需下一步的相關建議，請參閱 [處理效能問題](#)。

您的應用程式或 IBM MQ for z/OS 是否已停止處理工作？

系統非預期地停止處理工作的原因有數個，包括佇列管理程式、應用程式、z/OS 及資料集的問題。

系統非預期地停止處理工作有幾個原因。其中包括：

佇列管理程式問題

佇列管理程式可能正在關閉。

應用程式問題

應用程式設計錯誤可能表示程式遠離其正常處理程序，或應用程式可能進入迴圈。也可能有應用程式異常終止。

IBM MQ 問題

您的佇列可能已停用 MQPUT 或 MQGET 呼叫，無法傳送郵件的佇列可能已滿，或 IBM MQ for z/OS 可能處於等待狀態或迴圈。

z/OS 及其他系統問題

z/OS 可能處於等待狀態，或 CICS 或 IMS 可能處於等待狀態或迴圈。系統或 Sysplex 層次可能有問題影響佇列管理程式或通道起始程式。例如，分頁過多。它也可能指出 DASD 問題，或具有高處理器用量的較高優先順序作業。

Db2 和 RRS 問題

檢查 Db2 和 RRS 是否在作用中。

無論如何，請執行下列檢查來判斷問題的原因：

z/OS IBM MQ 佇列是否有問題？

請使用這個主題來調查 IBM MQ 佇列的潛在問題。

如果您懷疑有問題影響子系統上的佇列，請使用作業及控制台來顯示系統指令輸入佇列。

如果系統回應

如果系統回應，則至少有一個佇列在運作中。在此情況下，請遵循第 28 頁的『您的部分佇列是否在運作中？』中的程序。

如果系統未回應

問題可能在於整個子系統。在此實例中，請嘗試停止並重新啟動佇列管理程式，以回應所產生的任何錯誤訊息。

請檢查主控台上是否有任何需要採取動作的訊息。解決可能影響 IBM MQ 的任何問題，例如針對保存日誌裝載磁帶的要求。查看其他子系統或 CICS 區域是否受影響。

請使用 DISPLAY QMGR COMMANDQ 指令來識別系統指令輸入佇列的名稱。

如果重新啟動之後仍發生問題

請聯絡 IBM 支援中心，以取得協助 (請參閱第 36 頁的『聯絡 IBM 支援中心』)。

相關概念

第 29 頁的『是否已定義正確的佇列？』

IBM MQ 需要某些預先定義的佇列。如果未正確定義這些佇列，則可能會發生問題。

第 29 頁的『問題是否僅影響遠端或叢集佇列？』

如果問題僅發生在遠端或叢集佇列上，請使用本主題進行進一步調查。

第 30 頁的『問題是否只影響共用佇列？』

請利用這個主題來調查可能導致共用佇列問題的佇列共用群組問題。

z/OS 您的部分佇列是否在運作中？

請利用這個主題來調查佇列子集發生問題的時間。

如果您懷疑只有一部分佇列發生問題，請選取您認為有問題的本端佇列名稱，並執行下列程序：

顯示佇列資訊

使用 DISPLAY QUEUE 和 DISPLAY QSTATUS 指令來顯示佇列的相關資訊。

是否正在處理佇列？

- 如果 CURDEPTH 為 MAXDEPTH，則可能表示未處理佇列。檢查所有使用佇列的應用程式是否正常執行 (例如，檢查 CICS 系統中的交易是否在執行中，或為回應「佇列深度高」事件而啟動的應用程式是否在執行中)。
- 發出 DISPLAY QSTATUS (xx) IPPROCS 以查看佇列是否開啟以供輸入。如果沒有，請啟動應用程式。
- 如果 CURDEPTH 不是 MAXDEPTH，請檢查下列佇列屬性以確定它們是正確的：

- 如果正在使用觸發:

- 觸發監視器正在執行中嗎?
- 觸發深度是否太大?
- 程序名稱是否正確?
- 是否符合 **所有** 觸發條件?

請發出 DISPLAY QSTATUS (xx) IPPROCS , 以查看應用程式是否有相同的佇列開啟以供輸入。在部分觸發實務範例中, 如果開啟佇列以供輸入, 則不會產生觸發訊息。停止應用程式, 以導致呼叫觸發處理程序。

- 佇列可以共用嗎? 如果不是, 則為另一個應用程式 (批次、IMS 或 CICS) 可能已開啟以供輸入。
- 是否已針對 GET 及 PUT 適當地啟用佇列?

您有長時間執行的工作單元嗎?

如果 CURDEPTH 不是零, 但當您嘗試 MQGET 訊息時, 佇列管理程式會回覆沒有可用的訊息, 請發出 DIS QSTATUS (xx) TYPE (HANDLE) 以顯示佇列已開啟之應用程式的相關資訊, 或發出 DIS CONN (xx) 以提供連接至佇列之應用程式的相關資訊。

有多少作業正在存取佇列?

發出 DISPLAY QSTATUS (xx) OPPROCS IPPROCS , 以查看有多少作業將訊息放入佇列, 以及從佇列取得訊息。在佇列共用環境中, 檢查每一個佇列管理程式上的 OPPROCS 及 IPPROCS。或者, 使用 CMDSCOPE 屬性來檢查所有佇列管理程式。如果沒有應用程式程序從佇列取得訊息, 請判斷原因 (例如, 因為應用程式需要啟動、連線已中斷, 或因為 MQOPEN 呼叫因某些原因而失敗)。

此佇列是共用佇列嗎? 問題是否只影響共用佇列?

請確認支援共用佇列的 Sysplex 元素沒有問題。例如, 檢查 IBM MQ 管理的「連結機能」清單結構是否有問題。

使用 D XCF、STRUCTURE、STRNAME=ALL 來檢查「連結機能」結構是否可存取。

使用 D RRS 來檢查 RRS 是否作用中。

此佇列是叢集的一部分嗎?

請檢查佇列是否屬於叢集 (來自 CLUSTER 或 CLUSNL 屬性)。如果是, 請驗證管理佇列的佇列管理程式在叢集中仍處於作用中。

如果您無法解決問題

請聯絡 IBM 支援中心, 以取得協助 (請參閱 [第 36 頁的『聯絡 IBM 支援中心』](#))。

z/OS 是否已定義正確的佇列?

IBM MQ 需要某些預先定義的佇列。如果未正確定義這些佇列, 則可能會發生問題。

請檢查系統指令輸入佇列、系統指令回覆模型佇列及回覆目的地佇列是否正確定義, 以及 MQOPEN 呼叫是否成功。

如果您使用系統指令回覆模型佇列, 請檢查它是否已正確定義。

如果您使用叢集, 則需要定義 SYSTEM.CLUSTER.COMMAND.QUEUE 可使用與叢集處理相關的指令。

z/OS 問題是否僅影響遠端或叢集佇列?

如果問題僅發生在遠端或叢集佇列上, 請使用本主題進行進一步調查。

如果問題只影響遠端或叢集佇列, 請檢查:

是否正在存取遠端佇列?

檢查將訊息放入遠端佇列的程式是否已順利執行 (請參閱 [第 127 頁的『處理 z/OS 上不正確的輸出』](#))。

系統鏈結是否在作用中?

視情況使用 APPC 或 TCP/IP 指令, 以檢查兩個系統之間的鏈結是否在作用中。

對 TCP/IP 使用 PING 或 OPING, 或對 APPC 使用 D NET ID=xxxxx, E。

觸發是否正常運作?

如果您使用觸發來啟動分散式佇列處理程序，請檢查傳輸佇列是否已設定觸發，以及佇列是否已啟用。

通道或接聽器是否在執行中?

必要的話，請手動啟動通道或接聽器，或嘗試停止並重新啟動通道。如需相關資訊，請參閱 [配置分散式佇列作業](#)。

在啟動通道起始程式及接聽器時尋找錯誤訊息。請參閱 [IBM MQ for z/OS 訊息、完成及原因碼](#) 及 [配置分散式佇列](#)，以判斷原因。

通道狀態為何?

請使用 DISPLAY CHSTATUS (channel_name) 指令來檢查通道狀態。

您的處理程序及通道定義是否正確?

請檢查程序定義及通道定義。

如需如何使用分散式佇列的相關資訊，以及如何定義通道的相關資訊，請參閱 [配置分散式佇列](#)。

問題是否只影響共用佇列?

請利用這個主題來調查可能導致共用佇列問題的佇列共用群組問題。

如果問題只影響佇列共用群組，請使用 CSQ5PQSG 公用程式的 Verify QSG 功能。此指令會驗證在點陣圖配置欄位及 Db2 佇列管理程式、結構及共用佇列物件的物件定義方面，Db2 設定是否一致，並報告探索到任何不一致的詳細資料。

下列是具有錯誤的 Verify QSG 報告範例：

```
CSQU501I  VERIFY QSG function requested
CSQU503I  QSG=SQ02, DB2 DSG=DSN710P5, DB2 ssid=DFP5
CSQU517I  XCF group CSQGSQ02 already defined
CSQU520I  Summary information for XCF group CSQGSQ02
CSQU522I  Member=MQ04, state=QUIESCED, system=MV4A
CSQU523I  User data=D4E5F4C15AD4D8F0F4404040C4C5....
CSQU522I  Member=MQ03, state=QUIESCED, system=MV4A
CSQU523I  User data=D4E5F4C15AD4D8F0F3404040C4C6....
CSQU526I  Connected to DB2 DF4A
CSQU572E  Usage map T01_ARRAY_QMGR and DB2 table CSQ.ADMIN_B_QMGR inconsistent
CSQU573E  QMGR MQ04 in table entry 1 not set in usage map
CSQU574E  QMGR 27 in usage map has no entry in table
CSQU572E  Usage map T01_ARRAY_STRUC and DB2 table CSQ.ADMIN_B_STRUCTURE inconsistent
CSQU575E  Structure APPL2 in table entry 4 not set in usage map
CSQU576E  Structure 55 in usage map has no entry in table
CSQU572E  Usage map T03_LH_ARRAY and DB2 table CSQ.OBJ_B_QUEUE inconsistent
CSQU577E  Queue MYSQ in table entry 13 not set in usage map for structure APPL1
CSQU576E  Queue 129 in usage map for structure APPL1 has no entry in table
CSQU528I  Disconnected from DB2 DF4A
CSQU148I  CSQ5PQSG Utility completed, return code=12
```

問題會影響網路的特定部分嗎?

網路問題可能導致 MQ for z/OS 的相關問題。請利用這個主題來檢閱網路問題的可能來源。

您可能能夠識別受問題影響的網路特定部分 (例如，遠端佇列)。如果遠端佇列管理程式的鏈結無法運作，則訊息無法流向目標佇列管理程式上的目標佇列。請檢查兩個系統之間的連線是否可用，以及通道起始程式和接聽器是否已啟動。請使用 MQSC PING CHANNEL 指令來檢查連線。

請檢查訊息是否到達傳輸佇列，並檢查傳輸佇列及任何遠端佇列的本端佇列定義。使用 DISPLAY CHSTATUS 指令的 MQSC BYTSENT 關鍵字，以檢查資料是否沿通道流動。使用 DISPLAY QLOCAL (XMITQ) CURDEPTH 來檢查傳輸佇列上是否有要傳送的訊息。檢查通道兩端是否有診斷訊息通知您訊息已傳送至無法傳送郵件的佇列。

如果您使用 IBM MQ 叢集，請檢查是否已正確設定叢集定義。

您是否已進行任何可能導致問題發生的網路相關變更?

您是否已變更任何 IBM MQ 定義，或任何 CICS 或 IMS 定義? 請檢查傳輸佇列的觸發屬性。

在一天中特定時間發生或影響特定使用者的問題

請利用這個主題來檢閱在一天中特定時間或特定使用者群組發生的 IBM MQ 問題。

如果問題發生在一天中的特定時間，則可能是取決於系統載入。通常，尖峰系統負載是在上午中和下午中，因此這些期間是最有可能發生負載相依問題的時間。(如果您的網路延伸跨越多個時區，則尖峰系統負載可能似乎在一天中的某個其他時間發生。)

如果您認為 IBM MQ for z/OS 系統有效能問題，請參閱第 122 頁的『處理 z/OS 上的效能問題』。

如果問題只影響部分使用者，是否因為部分使用者沒有正確的安全授權? 如需 IBM MQ for z/OS 所檢查使用者 ID 的相關資訊，請參閱 [安全檢查的使用者 ID](#)。

問題是間歇性，還是所有 z/OS、CICS 或 IMS 系統都發生問題?

請檢閱這個主題，以考量問題是由應用程式互動所造成，還是與其他 z/OS 系統相關。

如果沒有考慮到處理程序可以彼此獨立執行的事實，可能會導致間歇性問題。例如，在先前處理程序完成之前，程式可能在未指定 WAIT 的情況下發出 MQGET 呼叫。如果您的應用程式在同步點 (亦即，在確定之前) 嘗試從佇列取得訊息，則您也可能會遇到此類型的問題。

如果只有在您存取特定 z/OS、IMS 或 CICS 系統時才發生問題，請考量此系統的不同之處。另請考量系統是否已進行任何可能影響其與 IBM MQ 互動方式的變更。

之前是否已順利執行應用程式?

通常可以透過判斷應用程式錯誤之前是否已順利執行，或是否已產生錯誤訊息及非預期的回覆碼來判斷應用程式錯誤。

如果問題似乎涉及某個特定應用程式，請考量應用程式之前是否已順利執行。

在對此問題回答是之前，請考量：

自前次順利執行應用程式之後，對應用程式進行了任何變更嗎?

如果是這樣，則錯誤可能位於應用程式的新部分或已修改部分的某個位置。請調查變更，並查看您是否可以找到問題的明顯原因。

應用程式的所有功能以前是否已完全執行過?

第一次使用應用程式之前從未啟動的部分時，是否發生問題? 如果是這樣，則可能是應用程式的該部分發生錯誤。嘗試找出應用程式失敗時執行的動作，並檢查程式該部分中的原始碼是否有錯誤。

如果已在先前多次順利執行程式，請檢查現行佇列狀態及發生錯誤時正在處理的檔案。它們可能包含一些不尋常的資料值，會導致呼叫程式中很少使用的路徑。

應用程式會檢查所有回覆碼嗎?

您的系統是否已變更，可能以較小的方式。請檢查應用程式因變更而收到的回覆碼。例如：

- 您的應用程式是否假設可以共用它所存取的佇列? 如果佇列已重新定義為專用，您的應用程式是否可以處理回覆碼，指出它無法再存取該佇列?
- 是否有任何安全設定檔已變更? MQOPEN 呼叫可能因安全違規而失敗; 您的應用程式可以從產生的回覆碼回復嗎?

應用程式是否預期特定的訊息格式?

如果已將具有非預期訊息格式的訊息放入佇列 (例如，來自不同平台上佇列管理程式的訊息)，則可能需要資料轉換或其他不同的處理形式。

應用程式是否在其他 IBM MQ for z/OS 系統上執行?

造成問題的佇列管理程式設定方式是否不同? 例如，是否已使用相同的訊息長度上限或預設優先順序來定義佇列?

應用程式是否使用 MQSET 呼叫來變更佇列屬性?

應用程式是否設計為將佇列設為沒有觸發程式，然後處理部分工作，然後將佇列設為具有觸發程式? 在佇列重設為具有觸發程式之前，應用程式可能已失敗。

應用程式是否處理導致應用程式失敗的訊息？

如果應用程式因訊息毀損而失敗，則會回復擷取的訊息。下一個應用程式可能會收到相同的訊息，並以相同的方式失敗。請確定應用程式使用取消計數；當達到取消計數臨界值時，有問題的訊息會放到取消佇列中。

如果您的應用程式之前從未順利執行過，請仔細檢查您的應用程式，以查看您是否可以找到下列任何錯誤：

轉換和編譯問題

在查看程式碼之前，請先檢查轉換器、編譯器或組譯器以及鏈結編輯器的輸出，以查看是否已報告任何錯誤。如果您的應用程式無法轉換、編譯/組合或鏈結編輯至載入程式庫，則如果您嘗試呼叫它，它也會無法執行。如需建置應用程式的相關資訊，以及所需工作控制語言 (JCL) 陳述式的範例，請參閱 [開發應用程式](#)。

批次和 TSO 程式

對於批次和 TSO 程式，請檢查是否包含正確的 Stub。有一個批次 Stub 和兩個 RRS Stub。如果您使用 RRS，請確認您沒有搭配使用 MQCMIT 和 MQBACK 呼叫與 CSQBRSTB Stub。如果您要繼續搭配使用這些呼叫與 RRS，請使用 CSQBRSI Stub。

CICS 程式集

若為 CICS 程式，請檢查程式、IBM MQ CICS Stub 及 CICS Stub 是否以正確的順序鏈結。此外，請檢查您的程式或交易是否已定義給 CICS。

IMS 程式

對於 IMS 程式，請檢查鏈結是否包含程式、IBM MQ Stub 及 IMS 語言介面模組。請確定已指定正確的進入點。如果要使用 IBM MQ，從 IMS 程式動態載入的程式也必須鏈結 Stub 和語言介面模組。

可能的程式碼問題

如果文件顯示已完成每一個步驟，且沒有錯誤，請考量撰寫應用程式的程式碼。問題的症狀是否指出失敗的功能，因而指出錯誤的程式碼片段？如需導致 IBM MQ 應用程式發生問題的一些常見錯誤範例，請參閱 [第 33 頁的『您有程式錯誤嗎？』](#)。

應用程式是否從 IBM MQ 報告錯誤？

例如，可能未針對 "gets" 啟用佇列。它會收到指定此條件的回覆碼，但不會報告它。請考量應用程式報告任何錯誤或問題的位置。

自前次成功執行以來是否已進行任何變更？

自前次順利執行以來所做的最新變更通常是非預期錯誤的來源。本主題包含在判斷問題時可以調查的部分變更的相關資訊。

當您考量最近可能已進行的變更時，請考慮 IBM MQ，以及它與之互動的其他程式、硬體及任何新的應用程式。另請考量您尚未知道的新應用程式可能已在系統上執行的可能性。

您的起始設定程序已變更嗎？

請考量這是否可能是問題的原因。您是否已變更任何資料集，或已變更程式庫定義？是否已使用不同的參數來起始設定 z/OS？此外，請檢查在起始設定期間傳送至主控台的錯誤訊息。

您已變更任何佇列定義或安全設定檔嗎？

請考量是否已變更部分佇列，使其成為叢集的成員。這項變更可能表示訊息來自不同的來源 (例如，其他佇列管理程式或應用程式)。

您是否已變更 Sysplex 中與支援及實作共用佇列相關的任何定義？

請考量變更定義的效果，例如 Sysplex 連結資料集或「連結機能」資源管理原則。這些變更可能對共用佇列的作業產生影響。此外，請考量變更對 Db2 資料共用環境的影響。

z/OS 系統上的任何軟體是否已升級至更新版本？

請考量您是否需要執行任何必要的後置安裝或移轉活動。

您的 z/OS 子系統名稱表格已變更嗎？

對並存條件軟體 (例如 z/OS 或 LE) 的層次進行變更可能需要對 IBM MQ 進行其他變更。

您的應用程式是否處理因您所做的任何變更而可能取得的回覆碼？

請確定您的應用程式會處理您所建立的任何新回覆碼。

您有程式錯誤嗎？

請利用這個主題來調查程式錯誤是否導致 IBM MQ 問題。

下列範例說明執行 IBM MQ 程式時所發生問題的最常見原因。請考量下列其中一個錯誤可能導致系統發生問題的可能性。

- 程式會發出 MQSET 來變更佇列屬性，但無法重設佇列的屬性。例如，將佇列設為 NOTRIGGER。
- 對佇列屬性做出不正確的假設。此假設包括假設佇列可以在 MQOPEN-exclusive 時使用 MQOPEN 開啟，並假設佇列不在叢集中。
- 嘗試在沒有正確安全授權的情況下存取佇列及資料。
- 鏈結沒有 Stub 或具有錯誤 Stub 的程式 (例如，具有 CICS Stub 的 TSO 程式)。這可能導致長時間執行的工作單元，或 X'0C4' 或其他異常終止。
- 在 MQI 呼叫中傳遞不正確或無效的參數；如果傳遞的參數數目錯誤，則無法嘗試完成完成碼及原因碼欄位，且作業會異常終止。(這是 X'0C4' 異常終止。)

如果您嘗試在比所撰寫的 MQSeries 版本更舊的版本上執行應用程式，其中部分 MQI 值無效，則可能會發生此問題。

- 無法正確地將 IBM MQ 模組定義至 z/OS (此錯誤會導致 CSQYASCP 中的 X'0C4' 異常終止)。
- 無法檢查 MQI 要求的回覆碼。

如果您嘗試在未檢查新回覆碼的 IBM MQ 新版本上執行應用程式，則可能會發生此問題。

- 無法以稍後 MQI 呼叫所需的正確選項開啟物件，例如使用 MQOPEN 呼叫來開啟佇列，但未指定正確選項來啟用佇列後續 MQGET 呼叫。
- 無法正確地起始設定 *MsgId* 和 *CorrelId*。

此錯誤尤其適用於 MQGET。

- 使用不正確的位址。
- 在起始設定之前使用儲存體。
- 傳遞指定了不正確長度的變數。
- 以錯誤順序傳遞參數。
- 無法對 RACF 定義正確的安全設定檔和類別。

這可能會停止佇列管理程式，或阻止您執行任何正式作業工作。

- 依賴已移轉應用程式的預設 MQI 選項。

例如，z/OS 預設為同步點的 MQGET 和 MQPUT。分散式平台預設值不同步點。

- 依賴入口網站應用程式正常或異常結束時的預設行為。

在 z/OS 上，正常結束會執行隱含 MQCMIT，異常結束會執行隱含回復。

有異常終止嗎？

請利用這個主題來調查異常終止的一般原因，以及可能造成問題的不同類型異常終止。

如果您的應用程式已停止執行，則可能是異常終止 (異常終止) 所導致。

視您使用的應用程式類型而定，您會在下列其中一個工作區收到異常終止通知：

批次

您的清單顯示異常終止。

CICS

您會看到 CICS 交易異常終止訊息。如果您的作業是終端機作業，則此訊息會顯示在您的畫面上。如果您的作業未附加至終端機，則訊息會顯示在 CICS CSMT 日誌上。

IMS

在所有情況下，您會在 IBM MQ for IMS 主要終端機及所涉及的相依區域清單中看到一則訊息。如果正在處理從終端機輸入的 IMS 交易，則也會將錯誤訊息傳送至該終端機。

TSO

您可能會在畫面上看到含有回覆碼的 TSO 訊息。(是否顯示此訊息視系統的設定方式及錯誤類型而定。)

異常終止的一般原因

異常終止可能是因為使用者在正常終止之前結束正在執行的作業；例如，如果您清除 CICS 交易。應用程式中的錯誤也可能導致異常終止。

位址空間傾出及交易傾出

對於部分異常終止，會產生位址空間傾出。對於 CICS 交易，會提供交易傾出，其中顯示交易感興趣的儲存區。

- 如果應用程式傳遞部分資料 (其位址不再有效)，則有時會在使用者的位址空間中產生傾出。

註：若為批次傾出，則會將傾出格式化並寫入 SYSUDUMP。如需 SYSUDUMP 的相關資訊，請參閱第 120 頁的『z/OS 上的 SYSUDUMP 資訊』。對於 CICS，系統傾出會寫入 SYS1.DUMP 資料集，以及所採取的交易傾出。

- 如果 IBM MQ for z/OS 本身的問題導致異常終止，則會傳回異常終止碼 X'5C6' 或 X'6C6'，以及異常終止原因碼。此原因碼唯一說明問題的原因。如需異常終止碼的相關資訊，請參閱第 98 頁的『IBM MQ for z/OS 異常終止』；如需原因碼的說明，請參閱 [回覆碼](#)。

異常程式終止

如果您的程式異常終止，請參閱第 99 頁的『處理 IBM MQ for z/OS 上的異常終止』。

如果您的系統異常終止，且您想要分析產生的傾出，請參閱第 104 頁的『IBM MQ for z/OS 傾出』。本節告訴您如何格式化傾出，以及如何解譯其中包含的資料。

z/OS 您是否取得不正確的輸出？

請利用這個主題來檢閱您收到的任何不正確輸出。

如果您已取得您認為不正確的輸出，請考量下列各項：

分類不正確的輸出

" 不正確的 outputIncorrect 可能被視為您未預期的任何輸出。不過，請在問題判斷的環境定義中小心使用這個術語，因為它可能是某種其他類型錯誤的次要效果。例如，如果您取得任何重複輸出，則可能發生迴圈，即使該輸出符合您的預期。

錯誤訊息

IBM MQ 也會透過傳送錯誤訊息來回應它偵測到的許多錯誤。您可能將這些訊息視為「不正確的 outputcorrect，但它們只是另一種問題類型的症狀。如果您從 IBM MQ 收到非預期的錯誤訊息，請參閱第 26 頁的『是否有任何錯誤訊息、回覆碼或其他錯誤狀況?』。

非預期的訊息

如果您的應用程式未收到預期的訊息、收到包含非預期或毀損資訊的訊息，或收到非預期的訊息 (例如，指定給不同應用程式的訊息)，請參閱第 127 頁的『處理 z/OS 上不正確的輸出』。

z/OS 您可以重新產生問題嗎？

重新產生問題可用來協助 IBM MQ for z/OS 的問題判斷。請利用這個主題來進一步隔離問題重現的類型。

如果您可以重新產生問題，請考量您可以重新產生問題的條件。例如：

是命令造成的嗎？

若是如此，則是從 z/OS 主控台發出的指令，從 CSQUTIL 寫入程式，以將指令放置到 SYSTEM.COMMAND.INPUT 佇列，或使用作業及控制台？

如果由另一個方法輸入指令，則該指令可以運作嗎？

如果指令在主控台中輸入時有效，但未在其他情況下有效，請檢查指令伺服器是否未停止，以及 SYSTEM.COMMAND.INPUT 佇列。

指令伺服器是否在執行中？

發出指令 DIS CMDSERV 以檢查。

是否由應用程式所造成？

若是如此，它會在 CICS、IMS、TSO 或批次中失敗嗎？

它是在所有 IBM MQ 系統上失敗，還是僅在部分系統上失敗？

應用程式是否造成問題？

當發生問題時，您可以識別任何似乎一律在系統中執行的應用程式嗎？如果是這樣，請檢查應用程式，以查看它是否有錯誤。

您是否無法接收來自 MQSC 指令的回應？

請使用本主題來調查無法從 MQSC 指令收到回應的問題。

如果您已從應用程式 (而不是從 z/OS 主控台) 發出 MQSC 指令，但尚未收到回應，請考量後續的問題：

指令伺服器是否在執行中？

檢查指令伺服器是否在執行中，如下所示：

1. 在 z/OS 主控台使用 DISPLAY CMDSERV 指令來顯示指令伺服器的狀態。
2. 如果指令伺服器不在執行中，請使用 START CMDSERV 指令來啟動它。
3. 如果指令伺服器正在執行中，請發出 DISPLAY QUEUE 指令。使用系統指令輸入佇列的名稱，以及 CURDEPTH 和 MAXDEPTH 屬性來定義顯示的資料。

如果這些值顯示佇列已滿，且指令伺服器已啟動，則不會從佇列讀取訊息。

4. 請嘗試停止指令伺服器，然後重新啟動它，以回應所產生的任何錯誤訊息。
5. 重新發出 display 指令，以查看它現在是否在運作中。

回覆是否已傳送至無法傳送郵件的佇列？

請使用 DISPLAY QMGR DEADQ 指令來找出系統無法傳送郵件的佇列名稱 (如果您不知道它是什麼)。

在 DISPLAY QUEUE 指令中搭配使用此名稱與 CURDEPTH 屬性，以查看佇列上是否有任何訊息。

無法傳送郵件的佇列訊息標頭 (無法傳送郵件的標頭結構) 包含說明問題的原因或回饋碼。(如需無法傳送郵件之標頭結構的相關資訊，請參閱 [原因 \(MQLONG\)](#)。)

是否已啟用 PUT 及 GET 的佇列？

從主控台使用 DISPLAY QUEUE 指令來檢查，例如 DISPLAY QUEUE (SYSTEM.COMMAND.INPUT) PUT GET。

WaitInterval 參數是否設為足夠長的時間？

如果 MQGET 呼叫已逾時，您的應用程式會收到完成碼 2 及原因碼 2033 (MQRC_NO_MSG_AVAILABLE)。(如需 **WaitInterval** 參數以及 MQGET 的完成碼和原因碼的相關資訊，請參閱 [WaitInterval \(MQLONG\)](#) 和 [MQGET-Get 訊息](#)。)

是否需要同步點？

如果您使用自己的應用程式將指令放入系統指令輸入佇列，請考量是否必須採用同步點。

您必須在將訊息放入佇列之後，並在嘗試接收回覆訊息之前取得同步點，或在放置訊息時使用 MQPMO_NO_SYNCPOINT。除非您已從同步點排除要求訊息，否則必須先取得同步點，然後再嘗試接收回覆訊息。

佇列的 MaxDepth 及 MaxMsgL 參數設定是否足夠高？

如需定義系統指令輸入佇列及回覆目的地佇列的相關資訊，請參閱 [CSQO016E](#)。

您是否正確使用 CorrelId 和 MsgId 參數?

您必須識別佇列，然後顯示 CURDEPTH。從主控台使用 DISPLAY QUEUE 指令 (例如，DISPLAY QUEUE (MY.REPLY.QUEUE) CURDEPTH)，以查看在回覆目的地佇列上是否有您尚未收到的訊息。

在應用程式中設定 *MsgId* 及 *CorrelId* 的值，以確保您從佇列接收所有訊息。

如果您已從 z/OS 主控台 (或其對等項目) 或應用程式發出 MQSC 指令，但尚未收到回應，則下列問題適用：

佇列管理程式是否仍在執行中，或您的指令是否導致異常終止?

尋找指出異常終止的錯誤訊息，如果發生異常終止，請參閱 [第 104 頁的『IBM MQ for z/OS 傾出』](#)。

是否發出任何錯誤訊息?

請檢查是否已發出任何可能指出錯誤本質的錯誤訊息。

如需您可以用來輸入 MQSC 指令之不同方法的相關資訊，請參閱 [發出指令](#)。

您的應用程式或 IBM MQ for z/OS 執行緩慢嗎?

應用程式本身或基礎軟體 (包括 IBM MQ) 可能導致應用程式變慢。本主題用於對慢速應用程式的起始調查。

如果您的應用程式執行緩慢，這可能表示它處於迴圈中，或正在等待無法使用的資源。

系統負載尖峰時間的問題是否更嚴重?

這也可能是效能問題所造成。可能是因為您的系統需要調整，或因為它的運作接近其容量限制。這種類型的問題可能在尖峰系統負載時間最差，通常是在上午中及下午中。(如果您的網路延伸到多個時區，您可能會覺得在其他時間發生尖峰系統負載。)

當系統負載較輕時，是否會發生問題?

如果您發現降低效能並不取決於系統負載，但有時會在系統負載較輕時發生，設計不佳的應用程式很可能是原因之一。這可能將本身視為只有在存取特定佇列時才會發生的問題。

IBM MQ for z/OS 執行緩慢嗎?

下列症狀可能指出 IBM MQ for z/OS 執行緩慢：

- 如果您的系統回應指令的速度太慢。
- 如果重複顯示佇列深度，表示針對您預期會有大量佇列活動的應用程式，正在緩慢處理佇列。



您可以在 [第 123 頁的『處理正在 z/OS 上慢速執行或已停止的應用程式』](#) 中找到處理等待及迴圈的指引，以及在 [第 122 頁的『處理 z/OS 上的效能問題』](#) 中找到處理效能問題的指引。

聯絡 IBM 支援中心

如果您在使用 IBM MQ 時遇到問題需要協助，您可以透過「IBM 支援中心網站」聯絡 IBM 支援中心。您也可以訂閱 IBM MQ 修正程式、疑難排解及其他新聞的相關通知。

關於這項作業

IBM 支援網站內的「IBM MQ 支援」頁面如下：

-  [IBM MQ for Multiplatforms 支援網頁](#)
-  [IBM MQ for z/OS 支援網頁](#)

若要接收 IBM MQ 修正程式、疑難排解及其他新聞的相關通知，您可以 [訂閱通知](#)。

如果您無法自行解決問題，且需要「IBM 支援中心」的協助，則可以開立案例 (請參閱 <https://www.ibm.com/my-support/s/createrecord/NewCase>)。



如需 IBM 支援中心 (包括如何登錄以取得支援) 的相關資訊，請參閱 [IBM 支援手冊](#)。

註：執行 `runmqras` 指令將協助您收集疑難排解資訊，然後再將它傳送至 IBM 支援中心。如需相關資訊，請參閱 [runmqras](#) (收集 IBM MQ 疑難排解資訊)。

使用錯誤日誌

您可以使用各種錯誤日誌來協助進行問題判斷及疑難排解。

在 Multiplatforms 上，使用下列鏈結以瞭解平台可用的錯誤日誌以及如何使用它們：

-  第 38 頁的『[UNIX, Linux, and Windows 上的錯誤日誌](#)』
-  第 41 頁的『[IBM i 上的錯誤日誌](#)』



 在 z/OS 上，錯誤訊息會寫入：

- z/OS 系統主控台
- 通道起始程式工作日誌

如需 IBM MQ for z/OS 上的錯誤訊息、主控台日誌及傾出的相關資訊，請參閱 [z/OS 上的問題判斷](#)。

暫停或排除錯誤日誌中的訊息

在 Multiplatforms 及 z/OS systems.:

-  如需在 Multiplatforms 上抑制部分訊息的詳細資料，請參閱 [第 44 頁的『在 Multiplatforms 上暫停來自錯誤日誌的通道錯誤訊息』](#)。
-  在 z/OS 上，如果您使用 z/OS 訊息處理機能來抑制訊息，則可以抑制主控台訊息。如需相關資訊，請參閱 [IBM MQ for z/OS 概念](#)。

AMQ_DIAGNOSTIC_MSG_SEVERITY

從 IBM MQ 9.0.3 開始，如果您設定環境變數 **AMQ_DIAGNOSTIC_MSG_SEVERITY**，對於 IBM MQ 處理程序，當該 IBM MQ 處理程序將訊息寫入錯誤日誌或主控台時，訊息嚴重性會附加至訊息號碼，以單一大寫英文字母表示，如下所示：

訊息類型	字元
參考資訊 (0)	I
警告 (10)	W
錯誤 (20 或 30)	E
嚴重 (40)	S
終止 (50)	T

例如：

```
AMQ5051I: The queue manager task 'LOGGER-IO' has started.
AMQ7075W: Unknown attribute foo at /var/mqm/qmgrs/QM1/qm.ini in
the configuration data.
AMQ9510E: Messages cannot be retrieved from a queue.
AMQ8506S: Command server MQGET failed with reason code 2009.
AMQ8301T: IBM MQ storage monitor job could not be started.
```

附註：

1. 因為佇列管理程式會寫入訊息，所以必須在啟動佇列管理程式的環境中設定環境變數。這在 Windows 上尤其重要，因為它可能是啟動佇列管理程式的 Windows 服務。
2. **AMQ_DIAGNOSTIC_MSG_SEVERITY** 也會影響程式所列印的訊息。



從 IBM MQ 9.0.4 開始，依預設會設定 **AMQ_DIAGNOSTIC_MSG_SEVERITY** 啟用的行為。您可以透過將環境變數設為 0 來關閉此行為。

請注意，新服務一律會新增嚴重性字元。



ISO 8601 時間

當 IBM MQ 處理程序將訊息寫入錯誤日誌時，會以 ISO 8601 格式 (世界標準時間 (UTC)) 併入訊息時間作為 Time() 屬性。

例如，其中 Z 時區指出 UTC:

```
11/04/2017 07:37:59 - Process(1) User(X) Program(amqzmuc0.exe)
Host(JOHNDOE) Installation(MQNI09000200)
VRMF(9.0.2.0) QMgr(QM1)
Time(2017-04-11T07:37:59.976Z)
```

在輪替上重新命名



在 IBM MQ 9.0.4 之前，當 AMQERR01.LOG 達到已配置的大小上限 AMQERR02.LOG 已重新命名為 AMQERR03.LOG。

然後，AMQERR01.LOG 會複製到 AMQERR02.LOG 和 AMQERR01.LOG 會截斷為空白。這表示在某些工具將這些訊息複製到 AMQERR02.LOG。

從 IBM MQ 9.0.4 開始，邏輯已變更，因此 AMQERR01.LOG 已重新命名為 AMQERR02.LOG。

相關概念

[第 7 頁的『IBM MQ 疑難排解及支援』](#)

如果您的佇列管理程式網路或 IBM MQ 應用程式有問題，請使用說明的技術來協助您診斷及解決問題。

[第 7 頁的『疑難排解概觀』](#)

疑難排解是尋找並消除問題原因的程序。每當您使用 IBM 軟體時發生問題，只要您問自己「發生什麼事？」，疑難排解程序即會開始。

[第 45 頁的『First Failure Support Technology \(FFST\)』](#)

First Failure Support Technology (FFST) for IBM MQ 提供事件的相關資訊，這些事件在發生錯誤時可協助 IBM 支援人員診斷問題。

相關工作

[第 56 頁的『使用追蹤』](#)

您可以使用不同類型的追蹤，協助您判斷問題及進行疑難排解。

ULW UNIX, Linux, and Windows 上的錯誤日誌

安裝 IBM MQ 時建立的 errors 子目錄最多可以包含三個錯誤日誌檔。

安裝時，會在 UNIX and Linux 系統下的 /var/mqm 檔案路徑及安裝目錄中建立 errors 子目錄，例如 Windows 系統下的 C:\Program Files\IBM\MQ\ 檔案路徑。errors 子目錄最多可以包含下列三個錯誤日誌檔:

- AMQERR01.LOG
- AMQERR02.LOG
- AMQERR03.LOG

如需儲存日誌檔之目錄的相關資訊，請參閱 [第 40 頁的『UNIX, Linux, and Windows 上的錯誤日誌目錄』](#)。

建立佇列管理程式之後，它會在需要時建立三個錯誤日誌檔。這些檔案與系統錯誤日誌目錄中的那些檔案同名。亦即，AMQERR01、AMQERR02 及 AMQERR03，且每一個都具有預設容量 **V 9.0.4** 32 MB (33554432 位元組)。容量可以在 IBM MQ Explorer 的 Extended 佇列管理程式內容頁面中變更，也可以在

qm.ini 檔的 QMErrorLog 段落中變更。這些檔案會放置在您安裝 IBM MQ 或建立佇列管理程式時所選取佇列管理程式資料目錄的 errors 子目錄中。errors 子目錄的預設位置是 UNIX and Linux 系統下的 /var/mqm/qmgrs/ qmname 檔案路徑，以及 Windows 系統下的 C:\Program Files\IBM\MQ\qmgrs\ qmname \errors 檔案路徑。

V 9.0.4 產生錯誤訊息時，會將它們放置在 AMQERR01 中。當 AMQERR01 變得大於 32 MB 時，會重新命名為 AMQERR02。

因此，最新的錯誤訊息一律會放在 AMQERR01 中，這是用來維護錯誤訊息歷程的其他檔案。

與通道相關的所有訊息也會放置在屬於佇列管理程式的適當錯誤檔中，除非佇列管理程式無法使用，或其名稱不明。在此情況下，通道相關訊息會放置在系統錯誤日誌目錄中。

若要檢查任何錯誤日誌檔的內容，請使用一般系統編輯器。

錯誤日誌範例

第 39 頁的圖 1 顯示從 IBM MQ 錯誤日誌擷取：

```
17/11/2014 10:32:29 - Process(2132.1) User(USER_1) Program(runmqchi.exe)
Host(HOST_1) Installation(Installation1)
VRMF(8.0.0.0) QMgr (A.B.C)
AMQ9542: Queue manager is ending.

EXPLANATION:
The program will end because the queue manager is quiescing.
ACTION:
None.
----- amqrimna.c : 931 -----
```

圖 1: 範例 IBM MQ 錯誤日誌

操作員訊息

操作員訊息會識別一般錯誤，通常是由使用者直接執行一些動作 (例如使用在指令上無效的參數) 所造成。操作員訊息已啟用國家語言，並將訊息型錄安裝在標準位置。

這些訊息會寫入相關聯的視窗 (如果有的話)。此外，部分操作員訊息會寫入 AMQERR01.LOG 檔，以及系統錯誤日誌目錄中對等檔案的其他檔案。

錯誤日誌存取限制

某些錯誤日誌目錄和錯誤日誌具有存取限制。

若要取得下列存取權，使用者或應用程式必須是 mqm 群組的成員：

- 對所有佇列管理程式錯誤日誌目錄的讀取及寫入權。
- 對所有佇列管理程式錯誤日誌的讀取及寫入權。
- 系統錯誤日誌的寫入權。

如果未獲授權的使用者或應用程式嘗試將訊息寫入佇列管理程式錯誤日誌目錄，則訊息會重新導向至系統錯誤日誌目錄。

忽略 UNIX and Linux 系統下的錯誤碼

在 UNIX and Linux 系統上，如果您不想要將特定錯誤訊息寫入佇列管理程式錯誤日誌，則可以使用 QMErrorLog 段落指定要忽略的錯誤碼。

如需相關資訊，請參閱 [佇列管理程式錯誤日誌](#)。

忽略 Windows 系統下的錯誤碼

在 Windows 系統上，錯誤訊息會同時寫入 IBM MQ 錯誤日誌及 Windows 應用程式事件日誌。寫入「應用程式事件日誌」的錯誤訊息包括錯誤嚴重性、警告嚴重性及資訊嚴重性的訊息。如果您不想將某些錯誤訊息寫入 Windows 應用程式事件日誌，您可以指定 Windows 登錄中要忽略的錯誤碼。

使用下列登錄機碼：

```
HKLM\Software\IBM\WebSphere MQ\Installation\MQ_INSTALLATION_NAME\IgnoredErrorCodes
```

其中 `MQ_INSTALLATION_NAME` 是與 IBM MQ 特定安裝相關聯的安裝名稱。

您設定的值是以空值字元定界的字串陣列，每一個字串值都與您要從錯誤日誌忽略的錯誤碼相關。完整清單以 NULL 字元終止，其類型為 `REG_MULTI_SZ`。

例如，如果您想要 IBM MQ 從 Windows 應用程式事件日誌中排除錯誤碼 AMQ3045、AMQ6055 及 AMQ8079，請將值設為：

```
AMQ3045\0AMQ6055\0AMQ8079\0\0
```

您要排除的訊息清單定義給機器上的所有佇列管理程式。在重新啟動每一個佇列管理程式之前，您對配置所做的任何變更都不會生效。

相關概念

第 7 頁的『[IBM MQ 疑難排解及支援](#)』

如果您的佇列管理程式網路或 IBM MQ 應用程式有問題，請使用說明的技術來協助您診斷及解決問題。

第 37 頁的『[使用錯誤日誌](#)』

您可以使用各種錯誤日誌來協助進行問題判斷及疑難排解。

第 95 頁的『[z/OS 上的問題判斷](#)』

IBM MQ for z/OS、CICS、Db2 及 IMS 會產生可用於問題判斷的診斷資訊。

相關工作

第 56 頁的『[使用追蹤](#)』

您可以使用不同類型的追蹤，協助您判斷問題及進行疑難排解。

相關參考

第 41 頁的『[IBM i 上的錯誤日誌](#)』

使用此資訊來瞭解 IBM MQ for IBM i 錯誤日誌。



UNIX, Linux, and Windows 上的錯誤日誌目錄

IBM MQ 會使用許多錯誤日誌來擷取有關其本身 IBM MQ 作業的訊息、您啟動的任何佇列管理程式，以及來自使用中通道的錯誤資料。錯誤日誌的位置取決於佇列管理程式名稱是否為已知，以及錯誤是否與用戶端相關聯。

儲存錯誤日誌的位置取決於佇列管理程式名稱是否為已知，以及錯誤是否與用戶端相關聯。

`MQ_INSTALLATION_PATH` 代表 IBM MQ 安裝所在的高階目錄。

- 如果已知佇列管理程式名稱，則錯誤日誌的位置會顯示在 [第 40 頁的表 2](#) 中。

平台	目錄
 Linux and Linux 系統 UNIX	<code>/var/mqm/qmgrs/ qmname /errors</code>
 Windows 系統	<code>MQ_INSTALLATION_PATH\QMGRS\ qmname \ERRORS\AMQERR01.LOG</code>

- 如果佇列管理程式名稱不明，則錯誤日誌的位置會顯示在 [第 41 頁的表 3](#) 中。

表 3: 系統錯誤日誌目錄	
平台	目錄
Linux 和 Linux 系統 UNIX	/var/mqm/errors
Windows 系統	MQ_INSTALLATION_PATH\QMGRS\@SYSTEM\ERRORS\AMQERR01.LOG

- 如果用戶端應用程式發生錯誤，則用戶端上錯誤日誌的位置會顯示在 [第 41 頁的表 4](#) 中。

表 4: 用戶端錯誤日誌目錄	
平台	目錄
Linux 和 Linux 系統 UNIX	/var/mqm/errors
Windows 系統	MQ_DATA_PATH\ERRORS\AMQERR01.LOG

Windows 在 IBM MQ for Windows 中，錯誤的指示也會新增至「應用程式日誌」，您可以使用 Windows 系統隨附的「事件檢視器」應用程式來檢查該日誌。

早期錯誤

在一些特殊情況下，尚未建立這些錯誤日誌，且發生錯誤。IBM MQ 會嘗試在錯誤日誌中記錄任何這類錯誤。日誌的位置視已建立多少佇列管理程式而定。

例如，如果由於配置檔毀損，無法判斷任何位置資訊，則會將錯誤記載至安裝時在根目錄 (/var/mqm 或 C:\Program Files\IBM\MQ) 上建立的錯誤目錄。

如果 IBM MQ 可以讀取其配置資訊，且可以存取「預設字首」的值，則會在「預設字首」屬性所識別目錄的 errors 子目錄中記載錯誤。例如，如果預設字首為 C:\Program Files\IBM\MQ，則會在 C:\Program Files\IBM\MQ\errors 中記載錯誤。

如需配置檔的進一步相關資訊，請參閱 [變更 IBM MQ 及佇列管理程式配置資訊](#)。

註：當佇列管理程式啟動時，訊息會通知「Windows 登錄」中的錯誤。

IBM i 上的錯誤日誌

使用此資訊來瞭解 IBM MQ for IBM i 錯誤日誌。

依預設，只有 QMQADM 群組的成員可以存取錯誤日誌。若要授與使用者對錯誤日誌的存取權 (非此群組的成員)，請將 **ValidateAuth** 設為否，並授與這些使用者 *PUBLIC 權限。如需相關資訊，請參閱 [檔案系統](#)。

IBM MQ 使用許多錯誤日誌來擷取有關 IBM MQ 本身的作業、您啟動的任何佇列管理程式，以及來自使用中通道的錯誤資料的訊息。

在安裝時，會在 IFS 中建立 /QIBM/UserData/mqm/errors 子目錄。

錯誤日誌的位置取決於佇列管理程式名稱是否已知。

在 IFS 中：

- 如果佇列管理程式名稱已知且佇列管理程式可用，則錯誤日誌位於：

```
/QIBM/UserData/mqm/qmgrs/qmname/errors
```

- 如果佇列管理程式無法使用，則錯誤日誌位於：

```
/QIBM/UserData/mqm/errors
```

您可以使用系統公用程式 EDTF 來瀏覽錯誤目錄和檔案。例如：

```
EDTF '/QIBM/UserData/mqm/errors'
```

或者，您可以從 WRKMQM 畫面對佇列管理程式使用選項 23。

錯誤子目錄最多可以包含下列三個錯誤日誌檔：

- AMQERR01.LOG
- AMQERR02.LOG
- AMQERR03.LOG

建立佇列管理程式之後，當佇列管理程式需要時，會建立三個錯誤日誌檔。這些檔案具有與 /QIBM/UserData/mqm/errors 檔案相同的名稱，即 AMQERR01、AMQERR02 及 AMQERR03，且每一個檔案都具有 2 MB (2097 152 個位元組) 的容量。這些檔案會放置在您所建立之每一個佇列管理程式的錯誤子目錄中，即 /QIBM/UserData/mqm/qmgrs/qmname/errors。

產生錯誤訊息時，會將它們放置在 AMQERR01 中。當 AMQERR01 變得大於 2 MB (2 097 152 個位元組) 時，會將它複製到 AMQERR02。在複製之前，AMQERR02 會複製到 AMQERR03.LOG。已捨棄先前的 AMQERR03 內容 (如果有的話)。

因此，最新的錯誤訊息一律會放在 AMQERR01 中，這是用來維護錯誤訊息歷程的其他檔案。

與通道相關的所有訊息也會放置在佇列管理程式的適當錯誤檔中，除非其佇列管理程式的名稱不明或佇列管理程式無法使用。當佇列管理程式名稱無法使用或無法判斷其名稱時，通道相關訊息會放在 /QIBM/UserData/mqm/errors 子目錄中。

若要檢查任何錯誤日誌檔的內容，請使用系統編輯器 EDTF 來檢視 IFS 中的串流檔。

註：

1. 請勿變更這些錯誤日誌的所有權。
2. 如果刪除任何錯誤日誌檔，則會在記載下一個錯誤訊息時自動重建。

早期錯誤

在一些特殊情況下，尚未建立錯誤日誌，且發生錯誤。IBM MQ 會嘗試在錯誤日誌中記錄任何這類錯誤。日誌的位置視已建立多少佇列管理程式而定。

例如，如果因為配置檔毀損，無法判斷任何位置資訊，則會將錯誤記載至安裝時所建立的錯誤目錄。

如果可以讀取 IBM MQ 配置檔及 AllQueueManager 段落的 DefaultPrefix 屬性，則會在 DefaultPrefix 屬性所識別目錄的 errors 子目錄中記載錯誤。

操作員訊息

操作員訊息會識別一般錯誤，通常是由使用者直接執行一些動作 (例如使用在指令上無效的參數) 所造成。操作員訊息已啟用國家語言，並將訊息型錄安裝在標準位置。

這些訊息會寫入工作日誌中 (如果有的話)。此外，部分操作員訊息會寫入佇列管理程式目錄中的 AMQERR01.LOG 檔案，其他訊息則會寫入錯誤日誌的 /QIBM/UserData/mqm/errors 目錄副本。

範例 IBM MQ 錯誤日誌

第 43 頁的圖 2 顯示從 IBM MQ 錯誤日誌中進行的一般擷取。

```

*****Beginning of data*****
07/19/02 11:15:56 AMQ9411: Repository manager ended normally.

EXPLANATION:
Cause . . . . . : The repository manager ended normally.
Recovery . . . . : None.
Technical Description . . . . . : None.
-----
07/19/02 11:15:57 AMQ9542: Queue manager is ending.

EXPLANATION:
Cause . . . . . : The program will end because the queue manager is quiescing.
Recovery . . . . : None.
Technical Description . . . . . : None.
----- amqrimna.c : 773 -----
-----
07/19/02 11:16:00 AMQ8004: IBM MQ queue manager 'mick' ended.
EXPLANATION:
Cause . . . . . : IBM MQ queue manager 'mick' ended.
Recovery . . . . : None.
Technical Description . . . . . : None.
-----
07/19/02 11:16:48 AMQ7163: IBM MQ job number 18429 started.

EXPLANATION:
Cause . . . . . : This job has started to perform work for Queue Manager
mick, The job's PID is 18429 the CCSID is 37. The job name is
582775/MQUSER/AMQZXMA0.
Recovery . . . . : None
-----
07/19/02 11:16:49 AMQ7163: IBM MQ job number 18430 started.

EXPLANATION:
Cause . . . . . : This job has started to perform work for Queue Manager
mick, The job's PID is 18430 the CCSID is 0. The job name is
582776/MQUSER/AMQZFUMA.
Recovery . . . . : None
-----
07/19/02 11:16:49 AMQ7163: IBM MQ job number 18431 started.

EXPLANATION:
Cause . . . . . : This job has started to perform work for Queue Manager
mick, The job's PID is 18431 the CCSID is 37. The job name is
582777/MQUSER/AMQZXMAX.
Recovery . . . . : None
-----
07/19/02 11:16:50 AMQ7163: IBM MQ job number 18432 started.

EXPLANATION:
Cause . . . . . : This job has started to perform work for Queue Manager
mick, The job's PID is 18432 the CCSID is 37. The job name is
582778/MQUSER/AMQALMPX.
Recovery . . . . : None
-----

```

圖 2: 從 IBM MQ 錯誤日誌擷取

相關概念

第 38 頁的『[UNIX, Linux, and Windows 上的錯誤日誌](#)』

安裝 IBM MQ 時建立的 errors 子目錄最多可以包含三個錯誤日誌檔。

第 7 頁的『[IBM MQ 疑難排解及支援](#)』

如果您的佇列管理程式網路或 IBM MQ 應用程式有問題，請使用說明的技術來協助您診斷及解決問題。

第 37 頁的『[使用錯誤日誌](#)』

您可以使用各種錯誤日誌來協助進行問題判斷及疑難排解。

第 95 頁的『[z/OS 上的問題判斷](#)』

IBM MQ for z/OS、CICS、Db2 及 IMS 會產生可用於問題判斷的診斷資訊。

相關工作

第 56 頁的『[使用追蹤](#)』

您可以使用不同類型的追蹤，協助您判斷問題及進行疑難排解。

IBM MQ classes for JMS 中的錯誤日誌

使用者可能需要更正動作的執行時期問題相關資訊會寫入 IBM MQ classes for JMS 日誌。

例如，如果應用程式嘗試設定 Connection Factory 的內容，但無法辨識該內容的名稱，則 IBM MQ classes for JMS 會將問題的相關資訊寫入其日誌。

依預設，包含日誌的檔案稱為 mqjms.log，且位於現行工作目錄中。不過，您可以在 IBM MQ classes for JMS 配置檔中設定 com.ibm.msg.client.commonservices.log.outputName 內容，來變更日誌檔的名稱和位置。如需 IBM MQ classes for JMS 配置檔的相關資訊，請參閱適用於 JMS 配置檔的 IBM MQ 類別，以及 com.ibm.msg.client.commonservices.log.outputName 內容有效值的詳細資料，請參閱第 146 頁的『IBM MQ classes for JMS 的記載錯誤』。

Multi 在 Multiplatforms 上暫停來自錯誤日誌的通道錯誤訊息

您可以防止選取的訊息在指定時間間隔內傳送至錯誤日誌，例如，如果 IBM MQ 系統產生大量參考訊息來填入錯誤日誌。

關於這項作業

有兩種方法可以抑制給定時間間隔的訊息：

- 在 qm.ini 檔的 QMErrorLog 段落中使用 SuppressMessage 和 SuppressInterval。
- 使用環境變數 MQ_CHANNEL_SUPPRESS_MSGS 及 MQ_CHANNEL_SUPPRESS_INTERVAL。

程序

- 若要使用 qm.ini 檔案中的 QMErrorLog 段落來暫停給定時間間隔的訊息，請使用 SuppressMessage 指定在給定時間間隔期間只寫入佇列管理程式錯誤日誌一次的訊息，並使用 SuppressInterval 指定要抑制訊息的時間間隔。
例如，若要暫停訊息 AMQ9999、AMQ9002、AMQ9209 達 30 秒，請在 qm.ini 檔案的 QMErrorLog 段落中包含下列資訊：

```
SuppressMessage=9001,9002,9202  
SuppressInterval=30
```

Windows **Linux** 或者，您可以使用 IBM MQ Explorer 中的「延伸佇列管理程式」內容頁來排除及抑制訊息，而不是直接編輯 qm.ini 檔案。

- 若要使用環境變數 MQ_CHANNEL_SUPPRESS_MSGS 及 MQ_CHANNEL_SUPPRESS_INTERVAL 來暫停給定時間間隔的訊息，請完成下列步驟：
 - a) 指定要使用 MQ_CHANNEL_SUPPRESS_MSGS 抑制的訊息。
在以逗點區隔的清單中，最多可以包含 20 個通道錯誤訊息碼。MQ_CHANNEL_SUPPRESS_MSGS 環境變數中沒有可包含的綜合性訊息 ID 清單。不過，訊息 ID 必須是通道訊息 (即 AMQ9xxx: messages)。
下列範例適用於訊息 AMQ9999、AMQ9002、AMQ9209。
 - **Linux** **UNIX** 在 UNIX 和 Linux 上：

```
export MQ_CHANNEL_SUPPRESS_MSGS=9999,9002,9209
```
 - **Windows** 在 Windows 上：

```
set MQ_CHANNEL_SUPPRESS_MSGS=9999,9002,9209
```
 - b) 指定要使用 MQ_CHANNEL_SUPPRESS_INTERVAL 抑制訊息的時間間隔。
預設值為 60,5，表示在 60 秒間隔中給定訊息的前 5 次出現之後，會抑制該訊息的任何進一步出現，直到該 60 秒間隔結束為止。值 0,0 表示一律抑制。值 0,n，其中 n > 0 表示永不抑制。

相關資訊

[UNIX, Linux, and Windows 上的 QMErrorLog 段落](#)

[IBM i 上的 QMErrorLog 段落](#)

[環境變數](#)

[佇列管理程式內容](#)

First Failure Support Technology (FFST)

First Failure Support Technology (FFST) for IBM MQ 提供事件的相關資訊，這些事件在發生錯誤時可協助 IBM 支援人員診斷問題。

發生內部事件時，「首次失敗資料擷取 (FFDC)」會提供系統環境的自動化 Snapshot。如果發生錯誤，IBM 支援人員會使用這個 Snapshot，讓您更瞭解系統的狀態，以及問題發生時的 IBM MQ。


事件的相關資訊包含在 FFST 檔案中。在 IBM MQ 中，FFST 檔案的檔案類型為 FDC。FFST 檔案不一定會指出錯誤。FFST 可能是參考資訊。

監視及內部管理

以下是一些提示，可協助您管理 FFST 事件：

- 監視系統的 FFST 事件，並確保在事件發生時採取適當且及時的補救動作。在某些情況下，FDC 檔可能是預期的，因此可以忽略，例如當使用者結束 IBM MQ 處理程序時所產生的 FFST 事件。透過適當的監視，您可以判斷哪些事件是預期的，哪些事件不是預期的。
- 也會針對 IBM MQ 之外的事件產生 FFST 事件。例如，如果 IO 子系統或網路有問題，則可以在 FDC 類型檔案中報告此問題。這些類型的事件不受 IBM MQ 控制，您可能需要讓協力廠商調查主要原因。
- 請確定已執行 FFST 檔案的良好內部管理。必須保存檔案，並且必須清除目錄或資料夾，以確保在支援團隊需要時，只有最新且相關的 FDC 檔案可用。

使用下列鏈結中的資訊，以瞭解不同平台中 FFST 檔案的名稱、位置及內容。

- [第 46 頁的『FFST: IBM MQ classes for JMS』](#)
- [第 50 頁的『FFST: IBM MQ for Windows』](#)
- [第 52 頁的『FFST: IBM MQ for UNIX 和 Linux 系統』](#)
-  [第 54 頁的『FFST: IBM MQ for IBM i』](#)
-

相關概念

[第 7 頁的『IBM MQ 疑難排解及支援』](#)

如果您的佇列管理程式網路或 IBM MQ 應用程式有問題，請使用說明的技術來協助您診斷及解決問題。

[第 7 頁的『疑難排解概觀』](#)

疑難排解是尋找並消除問題原因的程序。每當您使用 IBM 軟體時發生問題，只要您問自己「發生什麼事？」，疑難排解程序即會開始。

[第 37 頁的『使用錯誤日誌』](#)

您可以使用各種錯誤日誌來協助進行問題判斷及疑難排解。

[第 95 頁的『z/OS 上的問題判斷』](#)

IBM MQ for z/OS、CICS、Db2 及 IMS 會產生可用於問題判斷的診斷資訊。

相關工作

[第 56 頁的『使用追蹤』](#)

您可以使用不同類型的追蹤，協助您判斷問題及進行疑難排解。

[第 36 頁的『聯絡 IBM 支援中心』](#)

如果您在使用 IBM MQ 時遇到問題需要協助，您可以透過「IBM 支援中心網站」聯絡 IBM 支援中心。您也可以訂閱 IBM MQ 修正程式、疑難排解及其他新聞的相關通知。

FFST: IBM MQ classes for JMS

說明 First Failure Support Technology 的名稱、位置及內容 (FFST) IBM MQ classes for JMS 所產生的檔案。

當使用 IBM MQ classes for JMS 時，FFST 資訊會記錄在稱為 FFDC 的目錄中，依預設，它是產生 FFST 時所執行 IBM MQ classes for JMS 應用程式之現行工作目錄的子目錄。如果已在 IBM MQ classes for JMS 配置檔中設定內容 `com.ibm.msg.client.commonservices.trace.outputName`，則 FFDC 目錄是內容所指向之目錄的子目錄。如需 IBM MQ classes for JMS 的相關資訊，請參閱 [IBM MQ classes for JMS 配置檔](#)。

FFST 檔案包含一筆 FFST 記錄。每一筆 FFST 記錄都包含一般嚴重且可能無法回復之錯誤的相關資訊。這些記錄通常指出系統的配置問題或 IBM MQ classes for JMS 內的內部錯誤。

FFST 檔案命名為 `JMSC nnnn.FDC`，其中 `nnnn` 從 1 開始。如果完整檔名已存在，則此值會加 1，直到找到唯一的 FFST 檔名為止。

IBM MQ classes for JMS 應用程式的實例會將 FFST 資訊寫入多個 FFST 檔案。如果在單一執行應用程式期間發生多個錯誤，則每一個 FFST 記錄都會寫入至不同的 FFST 檔案。

FFST 記錄的區段

IBM MQ classes for JMS 所產生的 FFST 記錄包含下列區段：

標頭

標頭，指出建立 FFST 記錄的時間、IBM MQ classes for JMS 應用程式執行所在的平台，以及所呼叫的內部方法。標頭也包含探測 ID，可唯一識別 IBM MQ classes for JMS 內產生 FFST 記錄的位置。

資料

與 FFST 記錄相關聯的部分內部資料。

版本資訊

產生 FFST 記錄之應用程式正在使用的 IBM MQ classes for JMS 版本的相關資訊。

堆疊追蹤

產生 FFST 記錄之執行緒的 Java 堆疊追蹤。

內容儲存庫內容

在 IBM MQ classes for JMS 應用程式執行所在的「Java 執行時期環境」上設定的所有 Java 系統內容清單。

WorkQueueMananger 內容

IBM MQ classes for JMS 所使用內部執行緒儲存區的相關資訊。

執行時期內容

IBM MQ classes for JMS 應用程式執行所在系統上可用的記憶體數量及處理器數目的詳細資料。

元件管理程式內容

IBM MQ classes for JMS 所載入內部元件的部分相關資訊。

提供者特定資訊

產生 FFST 時執行的 IBM MQ classes for JMS 應用程式目前正在使用的所有作用中 JMS 連線、JMS 階段作業、MessageProducer 及 MessageConsumer 物件的相關資訊。此資訊包括 JMS 連線及 JMS 階段作業所連接的佇列管理程式名稱，以及 MessageProducers 及 MessageConsumers 所使用的 IBM MQ 佇列或主題物件名稱。

所有執行緒資訊

產生 FFST 記錄時 IBM MQ classes for JMS 應用程式執行所在之「Java 執行時期環境」中所有作用中執行緒的狀態相關詳細資料。會顯示每一個執行緒的名稱，以及每一個執行緒的 Java 堆疊追蹤。

FFST 日誌檔範例

```
-----START FFST-----
c:\JBoss-6.0.0\bin\FFDC\JMSSC0007.FDC PID:4472

JMS Common Client First Failure Symptom Report

Product      :- IBM MQ classes for JMS
```



```

Date/Time      :- Mon Feb 03 14:14:46 GMT 2014
System time    :- 1391436886081
Operating System :- Windows Server 2008
UserID         :- pault
Java Vendor    :- IBM Corporation
Java Version   :- 2.6

Source Class   :- com.ibm.msg.client.commonservices.j2se.wmqsupport.PropertyStoreImpl
Source Method  :- getBooleanProperty(String)
ProbeID        :- XS002005
Thread         :- name=pool-1-thread-3 priority=5 group=workmanager-threads
cc1=BaseClassLoader@ef1c3794{vfs:///C:/JBoss-6.0.0/server/default/deploy/basicMDB.ear}

Data
----

| name :- com.ibm.mq.connector.performJavaEEContainerChecks

Version information
-----

Java Message Service Client
7.5.0.2
p750-002-130627
Production

IBM MQ classes for Java Message Service
7.5.0.2
p750-002-130627
Production

IBM MQ JMS Provider
7.5.0.2
p750-002-130627
Production

Common Services for Java Platform, Standard Edition
7.5.0.2
p750-002-130627
Production

Stack trace
-----

Stack trace to show the location of the FFST call
| FFST Location :- java.lang.Exception
|   at com.ibm.msg.client.commonservices.trace.Trace.getCurrentPosition(Trace.java:1972)
|   at com.ibm.msg.client.commonservices.trace.Trace.createFFSTString(Trace.java:1911)
|   at com.ibm.msg.client.commonservices.trace.Trace.ffstInternal(Trace.java:1800)
|   at com.ibm.msg.client.commonservices.trace.Trace.ffst(Trace.java:1624)
|   at
com.ibm.msg.client.commonservices.j2se.propertystore.PropertyStoreImpl.getBooleanProperty(
PropertyStoreImpl.java:322)
|   at
com.ibm.msg.client.commonservices.propertystore.PropertyStore.getBooleanPropertyObject(Pr
opertyStore.java:302)
|   at
com.ibm.mq.connector.outbound.ConnectionWrapper.jcaMethodAllowed(ConnectionWrapper.java:510)
|   at
com.ibm.mq.connector.outbound.ConnectionWrapper.setExceptionHandler(ConnectionWrapper.java:244)
|   at com.ibm.basicMDB.MDB.onMessage(MDB.java:45)
...

Property Store Contents
-----

All currently set properties
|   awt.toolkit                :- sun.awt.windows.WToolkit
|   catalina.ext.dirs          :- C:\JBoss-6.0.0\server\default\lib
|   catalina.home              :- C:\JBoss-6.0.0\server\default
|   com.ibm.cpu.endian         :- little
|   com.ibm.jcl.checkClassPath :-
|   com.ibm.mq.connector.performJavaEEContainerChecks :- false
|   com.ibm.oti.configuration  :- scar
|   com.ibm.oti.jcl.build      :- 20131013_170512
|   com.ibm.oti.shared.enabled :- false
|   com.ibm.oti.vm.bootstrap.library.path :- C:\Program
Files\IBM\Java70\jre\bin\compressedrefs;C:\Program Files\IBM\Java70\jre\bin
|   com.ibm.oti.vm.library.version :- 26
|   com.ibm.system.agent.path  :- C:\Program

```

```
Files\IBM\Java70\jre\bin
| com.ibm.util.extralibs.properties      :-
| com.ibm.vm.bitmode                    :- 64
| com.ibm.zero.version                   :- 2
| console.encoding                       :- Cp850
| file.encoding                          :- Cp1252
| file.encoding.pkg                      :- sun.io
...
```

WorkQueueMananger Contents

```
| Current ThreadPool size    :- 2
| Maintain ThreadPool size   :- false
| Maximum ThreadPool size   :- -1
| ThreadPool inactive timeout :- 0
```

Runtime properties

```
| Available processors      :- 4
| Free memory in bytes (now) :- 54674936
| Max memory in bytes      :- 536870912
| Total memory in bytes (now) :- 235012096
```

Component Manager Contents

Common Services Components:

```
| CMVC      :- p750-002-130627
| Class Name :- class com.ibm.msg.client.commonservices.j2se.J2SEComponent
| Component Name :- com.ibm.msg.client.commonservices.j2se
| Component Title :- Common Services for Java Platform, Standard Edition
| Factory Class :- class com.ibm.msg.client.commonservices.j2se.CommonServicesImplementation
| Version      :- 7.5.0.2
| inPreferenceTo[0] :- com.ibm.msg.client.commonservices.j2me
```

Messaging Provider Components:

```
| CMVC      :- p750-002-130627
| Class Name :- class com.ibm.msg.client.wmq.factories.WMQComponent
| Component Name :- com.ibm.msg.client.wmq
| Component Title :- IBM MQ JMS Provider
| Factory Class :- class com.ibm.msg.client.wmq.factories.WMQFactoryFactory
| Version      :- 7.5.0.2
```

Provider Specific Information

Overview of JMS System

```
Num. Connections : 3
Num. Sessions    : 3
Num. Consumers    : 0
Num. Producers   : 0
```

Detailed JMS System Information

```
Connections :
| Instance      :- com.ibm.msg.client.wmq.internal.WMQXAConnection@bd4b665a
| connectOptions :- version:5 options:64 clientConn:
[channelName:'MY.SVRCONN' version:10 channelType:6 transportType:2 desc:'<null>'
qMgrName:'test' xmitQName:'<null>' connectionName:'9.20.124.119(1414)' mcaName:'<null>'
modeName:'<null>' tpName:'<null>' batchSize:50 discInterval:6000 shortRetryCount:10
shortRetryInterval:60 longRetryCount:999999999 longRetryInterval:1200
seqNumberWrap:99999999 maxMsgLength:104857600 putAuthority:1 dataConversion:0
userIdentifier:'<null>' password:'<null>' mcaUserIdentifier:'<null>' mcaType:1
remoteUserIdentifier:'' msgRetryExit:'<null>' msgRetryUserData:'<null>' msgRetryCount:10
heartbeatInterval:1 batchInterval:0 nonPersistentMsgSpeed:2 clustersDefined:0
networkPriority:0
mcaSecurityId:0000000000000000000000000000000000000000000000000000000000000000
000000 remoteSecurityId:0000000000000000000000000000000000000000000000000000
00000000 sslCipherSpec:'<null>' sslPeerName:'<null>' sslClientAuth:0 keepAliveInterval:-1
localAddress:'<null>' batchHeartbeat:0 hdrComplList:(0,-1)msgCompList:(0,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1)clwlChannelRank:0 clwlChannelPriority:0 clwlChannelWeight:50
channelMonitoring:0 channelStatistics:0 exitNameLength:128 exitDataLength:32
sendExitsDefined:0 sendExit:'<null>'
sendUserData:0000000000000000000000000000000000000000000000000000000000000000
sendExitPtr:<null> sendUserDataPtr:<null> receiveExitsDefined:0 receiveExit:'<null>'
receiveUserData:0000000000000000000000000000000000000000000000000000000000000000
receiveExitPtr:<null> ReceiveUserDataPtr:<null> SharingConversations:999999999
```

```
propertyControl:0 maxInstances:99999999 maxInstancesPerClient:99999999
clientChannelWeight:0 connectionAffinity:1 batchSizeLimit:5000 useDLQ:2 defReconnect:0 ]
connTag:00000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000
sslConfig:[version:1 keyRepository:'<null>' cryptoHardware:'<null>' authInfoRecCount:0
keyResetCount:0 fipsRequired:0 encryptionPolicySuiteB:(1,0,0,0)certificateValPolicy:0 ]
connectionId:414D5143746573742020202020202020208CA3E2522028FD02 securityParms:[<null>]
| exceptionListener :-
com.ibm.msg.client.jms.internal.JmsProviderExceptionListener@f17b3583
| helper :-
com.ibm.msg.client.wmq.internal.WMQConsumerOwnerShadow@adabbe93
| queueManagerName :- test
...
```

```
Sessions : 3
| Instance :- com.ibm.msg.client.wmq.internal.WMQXASession@f5c63f0a
| Parent Connection :- com.ibm.msg.client.wmq.internal.WMQXAConnection@228b45cb
| ackMode :- 0
| asfConsumer :- <null>
| asfDestination :- <null>
| asfSubName :- <null>
| asyncPutCounter :-
com.ibm.msg.client.wmq.internal.WMQSession$AsyncPutCounter@88db6ec0
| didRecovAsyncPut :- false
| helper :-
com.ibm.msg.client.wmq.internal.WMQConsumerOwnerShadow@28192ad1
| inSyncpoint :- false
| queueManagerName :- test
...
```

```
Consumers :
Producers :
```

```
All Thread Information
Name : DispatchThread:
[com.ibm.mq.jmqi.remote.impl.RemoteSession[connectionId=414D5143746573742020202020202020208
CA3E2522028FA01]]
Priority : 5
ThreadGroup : java.lang.ThreadGroup[name=JMSSCThreadPool,maxpri=10]
ID : 86
State : TIMED_WAITING
Stack : java.lang.Object.wait(Object.java:-2)
: java.lang.Object.wait(Object.java:196)
:
com.ibm.mq.jmqi.remote.impl.RemoteDispatchThread.waitOnSleepingEvent(RemoteDispatchThread
.java:151)
:
com.ibm.mq.jmqi.remote.impl.RemoteDispatchThread.sleepPhase(RemoteDispatchThread.java:636)
:
com.ibm.mq.jmqi.remote.impl.RemoteDispatchThread.run(RemoteDispatchThread.java:385)
:
com.ibm.msg.client.commonservices.workqueue.WorkQueueItem.runTask(WorkQueueItem.java:214)
:
com.ibm.msg.client.commonservices.workqueue.SimpleWorkQueueItem.runItem(SimpleWorkQueueIt
em.java:105)
:
com.ibm.msg.client.commonservices.workqueue.WorkQueueItem.run(WorkQueueItem.java:229)
:
com.ibm.msg.client.commonservices.workqueue.WorkQueueManager.runWorkQueueItem(WorkQueueMa
nager.java:303)
:
com.ibm.msg.client.commonservices.j2se.workqueue.WorkQueueManagerImplementation$ThreadPoo
lWorker.run(WorkQueueManagerImplementation.java:1219)
Name : RcvThread:
com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection@269522111[qmid=test_2014-01-
24_15.55.24,fap=10,channel=MY.SVRCONN,ccsid=850,sharecnv=10,hbint=300,peer=/9.20.124.119(
1414),localport=65243,ssl=no,hConns=0,LastDataSend=1391436871409 (0ms ago
),LastDataRecv=1391436871409 (0ms ago),]
Priority : 5
ThreadGroup : java.lang.ThreadGroup[name=JMSSCThreadPool,maxpri=10]
ID : 84
State : RUNNABLE
Stack :
java.net.SocketInputStream.socketRead0(SocketInputStream.java:-2)
:
java.net.SocketInputStream.read(SocketInputStream.java:163)
:
java.net.SocketInputStream.read(SocketInputStream.java:133)
:
com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection.receive(RemoteTCPConnection.java:1545)
:
```

```

com.ibm.mq.jmqi.remote.impl.RemoteRcvThread.receiveBuffer(RemoteRcvThread.java:794)
:
com.ibm.mq.jmqi.remote.impl.RemoteRcvThread.receiveOneTSH(RemoteRcvThread.java:757)
:
com.ibm.mq.jmqi.remote.impl.RemoteRcvThread.run(RemoteRcvThread.java:150)
:
com.ibm.msg.client.commonservices.workqueue.WorkQueueItem.runTask(WorkQueueItem.java:214)
:
com.ibm.msg.client.commonservices.workqueue.SimpleWorkQueueItem.runItem(SimpleWorkQueueItem.java:105)
:
com.ibm.msg.client.commonservices.workqueue.WorkQueueItem.run(WorkQueueItem.java:229)
:
com.ibm.msg.client.commonservices.workqueue.WorkQueueManager.runWorkQueueItem(WorkQueueManager.java:303)
:
com.ibm.msg.client.commonservices.j2se.workqueue.WorkQueueManagerImplementation$ThreadPoolWorker.run(WorkQueueManagerImplementation.java:1219)
:
:
:
First Failure Symptom Report completed at Mon Feb 03 14:14:46 GMT 2014
-----END FFST-----

```

IBM 使用 FFST 記錄之標頭、「資料」及「堆疊追蹤」區段中的資訊來協助判斷問題。在許多情況下，除了透過「IBM 支援中心」引發問題之外，系統管理者在產生 FFST 記錄時幾乎無法執行。

抑制 FFST 記錄

IBM MQ classes for JMS 所產生的 FFST 檔案包含一筆 FFST 記錄。如果在執行 IBM MQ classes for JMS 應用程式期間多次發生問題，則會產生多個具有相同探測 ID 的 FFST 檔案。這可能不是所需要的。內容 `com.ibm.msg.client.commonservices.ffst.suppress` 可用來暫停產生 FFST 檔案。此內容必須在應用程式所使用的 IBM MQ classes for JMS 配置檔 中設定，且可以採用下列值：

- 0: 輸出所有 FFDC 檔案 (預設值)。
- 1: 僅輸出探測 ID 的第一個 FFST 檔案。
- integer*: 暫停探測 ID 的所有 FFST 檔案，但那些檔案是此數目的倍數。

Windows FFST: IBM MQ for Windows

說明 First Failure Support Technology 的名稱、位置及內容 (FFST) Windows 系統的檔案。

在 IBM MQ for Windows 中，FFST 資訊記錄在 `C:\Program Files\IBM\MQ\errors` 目錄的檔案中。

FFST 檔案包含一或多筆記錄。每一筆 FFST 記錄都包含一般嚴重且可能無法回復之錯誤的相關資訊。這些記錄通常指出系統的配置問題或 IBM MQ 內部錯誤。

FFST 檔案命名為 `AMQ nnnnn.mm.FDC`，其中：

NNNNN

是報告錯誤的處理程序 ID

公釐

從 0 開始。如果完整檔名已存在，則此值會加 1，直到找到唯一的 FFST 檔名為止。如果重複使用處理程序，則 FFST 檔名可能已存在。

處理程序的實例會將所有 FFST 資訊寫入至相同的 FFST 檔案。如果在處理程序的單一執行期間發生多個錯誤，則 FFST 檔案可以包含許多記錄。

當處理程序寫入 FFST 記錄時，也會將記錄傳送至「事件日誌」。該記錄包含 FFST 檔案的名稱，以協助自動追蹤問題。在應用程式層次建立事件日誌項目。

一般 FFST 日誌顯示在 [第 51 頁的圖 3](#) 中。

```

+-----+
| WebSphere MQ First Failure Symptom Report
| =====
|
| Date/Time           :- Mon January 28 2008 21:59:06 GMT
| UTC Time/Zone       :- 1201539869.892015 0 GMT
| Host Name           :- 99VXY09 (Windows 7 Build 2600: Service Pack 1)
| PIDS                :- 5724H7200
| LVLS                :- 7.0.0.0
| Product Long Name   :- IBM MQ for Windows
| Vendor              :- IBM
| Probe Id            :- HL010004
| Application Name     :- MQM
| Component           :- hlgReserveLogSpace
| SCCS Info           :- lib/logger/amqhlge0.c, 1.26
| Line Number         :- 246
| Build Date          :- Jan 25 2008
| CMVC level          :- p000-L050202
| Build Type          :- IKAP - (Production)
| UserID              :- IBM_User
| Process Name        :- C:\Program Files\IBM\MQ\bin\amqzlaa0.exe |
| Process             :- 00003456
| Thread              :- 00000030
| QueueManager        :- qmgr2
| ConnId(1) IPCC      :- 162
| ConnId(2) QM        :- 45
| Major Errorcode     :- hrcE_LOG_FULL
| Minor Errorcode     :- OK
| Probe Type          :- MSGAMQ6709
| Probe Severity      :- 2
| Probe Description   :- AMQ6709: The log for the Queue manager is full.
| FDCSequenceNumber  :- 0
|
+-----+

MQM Function Stack
zlaMainThread
zlaProcessMessage
zlaProcessMQIRequest
zlaMOPUT
zsqMOPUT
kpiMOPUT
kqiPutIt
kqiPutMsgSegments
apiPutMessage
aqmPutMessage
aqhPutMessage
aqqWriteMsg
aqqWriteMsgData
aqlReservePutSpace
almReserveSpace
hlgReserveLogSpace
xcsFFST

MQM Trace History
-----} hlgReserveLogSpace rc=hrcW_LOG_GETTING_VERY_FULL
-----{ xllLongLockRequest
-----} xllLongLockRequest rc=OK

...

```

圖 3: 範例 IBM MQ for Windows 首次失敗症狀報告

IBM 使用「功能堆疊」和「追蹤歷程」來協助判斷問題。在許多情況下，除了透過「IBM 支援中心」引發問題之外，系統管理者在產生 FFST 記錄時幾乎無法執行。

在某些情況下，除了 FFST 檔之外，還可以產生小型傾出檔，並放在 C:\Program Files\IBM\MQ\errors 目錄中。傾出檔將具有與 FFST 檔案相同的名稱，格式為 AMQnnnnn.mm.dmp。IBM 可以使用這些檔案來協助判斷問題。

First Failure Support Technology (FFST) 檔案及 Windows 用戶端

這些檔案已格式化且位於 IBM MQ MQI client 安裝目錄的 errors 子目錄中。

這些通常是嚴重且無法復原的錯誤，並指出系統的配置問題或 IBM MQ 內部錯誤。

這些檔案命名為 AMQnnnnn.mm.FDC，其中：

- nnnnn 是報告錯誤的處理程序 ID
- mm 是序號，通常為 0

當處理程序建立 FFST 時，它也會將記錄傳送至系統日誌。該記錄包含 FFST 檔案的名稱，以協助自動追蹤問題。

系統日誌項目是在 "user.error" 層次建立。

First Failure Support Technology 在 [First Failure Support Technology \(FFST\)](#) 中有詳細說明。

Linux

UNIX

FFST: IBM MQ for UNIX 和 Linux 系統

說明 First Failure Support Technology 的名稱、位置及內容 (FFST) UNIX and Linux 系統的檔案。

若為 UNIX and Linux 系統上的 IBM MQ，FFST 資訊會記錄在 /var/mqm/errors 目錄的檔案中。

FFST 檔案包含一或多筆記錄。每一筆 FFST 記錄都包含一般嚴重且可能無法回復之錯誤的相關資訊。這些記錄指出系統的配置問題或 IBM MQ 內部錯誤。

FFST 檔案命名為 AMQ *nnnnn.mm.FDC*，其中：

NNNNN

是報告錯誤的處理程序 ID

公釐

從 0 開始。如果完整檔名已存在，則此值會加 1，直到找到唯一的 FFST 檔名為止。如果重複使用處理程序，則 FFST 檔名可能已存在。

處理程序的實例會將所有 FFST 資訊寫入至相同的 FFST 檔案。如果在處理程序的單一執行期間發生多個錯誤，則 FFST 檔案可以包含許多記錄。

若要讀取 FFST 檔案的內容，您必須是檔案的建立者或 mqm 群組的成員。

當處理程序寫入 FFST 記錄時，它也會將記錄傳送至 syslog。該記錄包含 FFST 檔案的名稱，以協助自動追蹤問題。syslog 項目是在 *user.error* 層次建立。如需配置此項的相關資訊，請參閱關於 *syslog.conf* 的作業系統說明文件。

部分一般 FFST 資料顯示在 [第 53 頁的圖 4](#) 中。


```

+-----+
| WebSphere MQ First Failure Symptom Report
| =====
|
| Date/Time           :- Mon January 28 2008 21:59:06 GMT
| UTC Time/Zone      :- 1201539869.892015 0 GMT
| Host Name          :- mqperfh2 (HP-UX B.11.23)
| PIDS               :- 5724H7202
| LVLS               :- 7.0.0.0
| Product Long Name  :- IBM MQ for HP-UX
| Vendor             :- IBM
| Probe Id           :- XC034255
| Application Name   :- MQM
| Component          :- xcsWaitEventSem
| SCCS Info          :- lib/cs/unix/amqxerrx.c, 1.204
| Line Number        :- 6262
| Build Date         :- Jan 25 2008
| CMVC level         :- p000-L050203
| Build Type         :- IKAP - (Production)
| UserID             :- 00000106 (mqperf)
| Program Name       :- amqzmuc0
| Addressing mode    :- 64-bit
| Process            :- 15497
| Thread             :- 1
| QueueManager       :- CSIM
| ConnId(2) QM       :- 4
| Major Errorcode    :- OK
| Minor Errorcode    :- OK
| Probe Type         :- INCORROUT
| Probe Severity     :- 4
| Probe Description  :- AMQ6109: An internal IBM MQ error has occurred.
| FDCSequenceNumber :- 0
|
+-----+

```

```

MQM Function Stack
amqzmuc0
xcsWaitEventSem
xcsFFST

```

```

MQM Trace History
Data: 0x00003c87
--} xcsCheckProcess rc=OK
--} xcsRequestMutexSem
--} xcsRequestMutexSem rc=OK

```

```

...

```

圖 4: IBM MQ for UNIX 系統的 FFST 報告

IBM 使用「功能堆疊」和「追蹤歷程」來協助判斷問題。在許多情況下，除了透過「IBM 支援中心」引發問題之外，系統管理者在產生 FFST 報告時幾乎無法執行。

不過，系統管理者可能可以解決一些問題。在呼叫其中一個 IPC 函數 (例如 `semop` 或 `shmget`) 時，如果 FFST 顯示資源不足或裝置說明空間不足，則可能已超出相關核心參數限制。

如果 FFST 報告顯示 `setitimer` 的問題，則可能需要變更核心計時器參數。

若要解決這些問題，請增加 IPC 限制，重建核心，然後重新啟動機器。

First Failure Support Technology (FFST) 檔案及 UNIX and Linux 用戶端

當發生嚴重 IBM MQ 錯誤時，會寫入 FFST 日誌。它們會寫入 `/var/mqm/errors` 目錄。

這些通常是嚴重且無法復原的錯誤，並指出系統的配置問題或 IBM MQ 內部錯誤。

這些檔案命名為 `AMQnnnnn.mm.FDC`，其中：

- `nnnnn` 是報告錯誤的處理程序 ID
- `mm` 是序號，通常為 0

當處理程序建立 FFST 時，它也會將記錄傳送至系統日誌。該記錄包含 FFST 檔案的名稱，以協助自動追蹤問題。

系統日誌項目是在 "user.error" 層次建立。

First Failure Support Technology 在 [First Failure Support Technology \(FFST\)](#) 中有詳細說明。

IBM i **FFST: IBM MQ for IBM i**

說明 First Failure Support Technology 的名稱、位置及內容 (FFST) IBM i 系統的檔案。

對於 IBM i，FFST 資訊記錄在 /QIBM/UserData/mqm/errors 目錄中的串流檔中。

這些錯誤通常是嚴重且無法復原的錯誤，並指出系統的配置問題或 IBM MQ 內部錯誤。

串流檔命名為 AMQ nnnnn.mm.FDC，其中：

- nnnnn 是報告錯誤之處理程序的 ID。
- mm 是序號，通常是 0。

失敗工作的工作日誌副本會寫入與 .FDC 檔同名的檔案。檔名以 .JOB 結尾。

下列範例顯示部分一般 FFST 資料。

```
-----  
| IBM MQ First Failure Symptom Report  
| =====  
|  
| Date/Time      :- Mon January 28 2008 21:59:06 GMT  
| UTC Time/Zone  :- 1201539869.892015 0 GMT  
| Host Name     :- WINAS12B.HURSLEY.IBM.COM  
| PIDS         :- 5733A38  
| LVLS         :- 520  
| Product Long Name :- IBM MQ for IBMi  
| Vendor       :- IBM  
| Probe Id     :- XY353001  
| Application Name :- MQM  
| Component    :- xehAS400ConditionHandler  
| Build Date   :- Feb 25 2008  
| UserID      :- 00000331 (MAYFCT)  
| Program Name :- STRMQM_R MAYFCT  
| Job Name     :- 020100/MAYFCT/STRMQM_R  
| Activation Group :- 101 (QMOM) (QMOM/STRMQM_R)  
| Process      :- 00001689  
| Thread       :- 00000001  
| QueueManager :- TEST.AS400.OE.P  
| Major Errorcode :- STOP  
| Minor Errorcode :- OK  
| Probe Type    :- HALT6109  
| Probe Severity :- 1  
| Probe Description :- 0  
| Arith1       :- 1 1  
| Comment1     :- 00d0  
|  
-----
```

```
MQM Function Stack  
lpiSPIMQConnect  
zstMQConnect  
ziiMQCONN  
ziiClearUpAgent  
xcsTerminate  
xlsThreadInitialization  
xcsConnectSharedMem  
xstConnSetInSPbyHandle  
xstConnSharedMemSet  
xcsFFST
```

```
MQM Trace History  
<-- xcsCheckProcess rc=xecP_E_INVALID_PID  
-->  
xcsCheckProcess  
<-- xcsCheckProcess rc=xecP_E_INVALID_PID  
-->  
xlsThreadInitialization
```

```

-->
xcsConnectSharedMem
-->
xcsRequestThreadMutexSem
<-- xcsRequestThreadMutexSem rc=OK
-->
xihGetConnSPDetailsFromList
<-- xihGetConnSPDetailsFromList rc=OK
-->
xstCreateConnExtentList
<-- xstCreateConnExtentList rc=OK
-->
xstConnSetInSPbyHandle
-->
xstSerialiseSPList
-->
xllSpinLockRequest
<-- xllSpinLockRequest rc=OK
<-- xstSerialiseSPList rc=OK
-->
xstGetSetDetailsFromSPByHandle
<-- xstGetSetDetailsFromSPByHandle rc=OK
-->
xstConnSharedMemSet
-->
xstConnectExtent
-->
xstAddConnExtentToList
<-- xstAddConnExtentToList rc=OK
<-- xstConnectExtent rc=OK
-->
xcsBuildDumpPtr
-->
xcsGetMem
<-- xcsGetMem rc=OK
<-- xcsBuildDumpPtr rc=OK
-->
xcsBuildDumpPtr
<-- xcsBuildDumpPtr rc=OK
-->
xcsBuildDumpPtr
<-- xcsBuildDumpPtr rc=OK
-->
xcsFFST

Process Control Block
SPP:0000 :1aefSTRMQM_R MAYFCT 020100 :8bba0:0:6d E7C9C8D7 000004E0 00000699 00000000 XIHP...\...r...
SPP:0000 :1aefSTRMQM_R MAYFCT 020100 :8bbb0:1:6d 00000000 00000002 00000000 00000000 .....
SPP:0000 :1aefSTRMQM_R MAYFCT 020100 :8bbc0:2:6d 80000000 00000000 EC161F7C FC002DB0 .....@...¢
SPP:0000 :1aefSTRMQM_R MAYFCT 020100 :8bbd0:3:6d 80000000 00000000 EC161F7C FC002DB0 .....@...¢
SPP:0000 :1aefSTRMQM_R MAYFCT 020100 :8bbe0:4:6d 00000000 00000000 00000000 00000000 .....

Thread Control Block
SPP:0000 :1aefSTRMQM_R MAYFCT 020100 :1db0:20:6d E7C9C8E3 00001320 00000000 00000000 XIHT.....
SPP:0000 :1aefSTRMQM_R MAYFCT 020100 :1dc0:21:6d 00000001 00000000 00000000 00000000 .....
SPP:0000 :1aefSTRMQM_R MAYFCT 020100 :1dd0:22:6d 80000000 00000000 DD13C17B 81001000 .....A#a...
SPP:0000 :1aefSTRMQM_R MAYFCT 020100 :1de0:23:6d 00000000 00000046 00000002 00000001 .....
SPP:0000 :1aefSTRMQM_R MAYFCT 020100 :1df0:24:6d 00000000 00000000 00000000 00000000 .....

RecoveryIndex
SPP:0000 :1aefSTRMQM_R MAYFCT 020100 :2064:128:6d 00000000 .....

```

註:

1. MQM Trace History 區段是 200 個最新函數追蹤陳述式的日誌，不論任何 TRCMQM 設定為何，都會記錄在 FFST 報告中。
2. 只有針對連接至佇列管理程式子儲存區的工作，才會記錄佇列管理程式詳細資料。
3. 當失敗元件為 xehAS400ConditionHandler 時，其他資料會記載在錯誤目錄中，並從與異常狀況相關的工作日誌擷取。

IBM 使用函數堆疊和追蹤歷程來協助判斷問題。在大部分情況下，除了透過「IBM 支援中心」引發問題之外，系統管理者在產生 FFST 報告時幾乎無法執行。

使用追蹤

您可以使用不同類型的追蹤，協助您判斷問題及進行疑難排解。

關於這項作業

使用此資訊可以瞭解不同類型的追蹤，以及如何針對平台執行追蹤。

- **Windows** [第 56 頁的『在 Windows 上使用追蹤』](#)
- **Linux** **UNIX** [第 58 頁的『在 UNIX and Linux 系統上使用追蹤』](#)
- **IBM i** [第 60 頁的『在 IBM i 上搭配使用追蹤與 IBM MQ 伺服器』](#)
- **IBM i** [第 63 頁的『在 IBM i 上搭配使用追蹤與 IBM MQ 用戶端』](#)
- **z/OS** [第 65 頁的『在 z/OS 上使用追蹤來判斷問題』](#)
- [第 77 頁的『追蹤 TLS: runmqakm、strmqikm 及 runmqckm 函數』](#)
- [第 78 頁的『追蹤 IBM MQ classes for JMS 應用程式』](#)
- [第 82 頁的『追蹤 IBM MQ classes for Java 應用程式』](#)
- [第 85 頁的『追蹤 IBM MQ 資源配接器』](#)
- [第 87 頁的『追蹤其他 IBM MQ Java 元件』](#)
- [第 89 頁的『使用 IBM MQ classes for Java 和 IBM MQ classes for JMS 來控制執行中處理程序中的追蹤』](#)

相關概念

[第 7 頁的『IBM MQ 疑難排解及支援』](#)

如果您的佇列管理程式網路或 IBM MQ 應用程式有問題，請使用說明的技術來協助您診斷及解決問題。

[第 7 頁的『疑難排解概觀』](#)

疑難排解是尋找並消除問題原因的程序。每當您使用 IBM 軟體時發生問題，只要您問自己「發生什麼事？」，疑難排解程序即會開始。

[第 37 頁的『使用錯誤日誌』](#)

您可以使用各種錯誤日誌來協助進行問題判斷及疑難排解。

[第 45 頁的『First Failure Support Technology \(FFST\)』](#)

First Failure Support Technology (FFST) for IBM MQ 提供事件的相關資訊，這些事件在發生錯誤時可協助 IBM 支援人員診斷問題。

相關工作

[第 36 頁的『聯絡 IBM 支援中心』](#)

如果您在使用 IBM MQ 時遇到問題需要協助，您可以透過「IBM 支援中心網站」聯絡 IBM 支援中心。您也可以訂閱 IBM MQ 修正程式、疑難排解及其他新聞的相關通知。

Windows 在 Windows 上使用追蹤

使用 **strmqtrc** 和 **endmqtrc** 指令或 IBM MQ Explorer 介面來啟動和結束追蹤。

Windows 對用戶端追蹤機能使用下列指令：

strmqtrc

開始追蹤

endmqtrc

結束追蹤

輸出檔建立在 MQ_DATA_PATH/trace 目錄中。

IBM MQ for Windows 上的追蹤檔案

追蹤檔命名為 AMQppppp.qq.TRC，其中變數如下：

ppppp

報告錯誤的處理程序 ID。

qq

序號，從 0 開始。如果完整檔名存在，則此值會加 1，直到找到唯一的追蹤檔名稱為止。如果重複使用處理程序，則追蹤檔名稱可以存在。

註：

1. 處理程序 ID 包含的數字可以比範例中顯示的數字更少或更多。
2. 作為所追蹤實體的一部分執行的每一個處理程序都有一個追蹤檔。

若要格式化或檢視追蹤檔，您必須是追蹤檔的建立者，或 mqm 群組的成員。

SSL 追蹤檔案具有名稱 AMQ.SSL.TRC 和 AMQ.SSL.TRC.1。您無法格式化 SSL 追蹤檔案；請將未變更的檔案傳送至 IBM 支援中心。

如何啟動及停止追蹤

使用 **strmqtrc** 控制指令啟用或修改追蹤 (請參閱 [strmqtrc](#))。若要停止追蹤，請使用 **endmqtrc** 控制指令 (請參閱 [endmqtrc](#))。

在 IBM MQ for Windows 系統中，您也可以使用 IBM MQ Explorer 來啟動和停止追蹤，如下所示：

1. 從 **開始** 功能表啟動 IBM MQ Explorer。
2. 在「Navigator」視圖中，用滑鼠右鍵按一下 **IBM MQ** 樹狀結構節點，然後選取 **追蹤...**。即會顯示「追蹤對話框」。
3. 適當地按一下 **啟動** 或 **停止**。

選擇性元件追蹤

使用 **-t** 及 **-x** 選項來控制要記錄的追蹤明細數量。依預設，會啟用所有追蹤點。您可以使用 **-x** 選項來指定不要追蹤的點。因此，比方說，如果您只想追蹤透過通訊網路流動的資料，請使用：

```
strmqtrc -x all -t comms
```

如需追蹤指令的詳細資訊，請參閱 [strmqtrc](#)。

選擇性處理程序追蹤

使用 **strmqtrc** 指令控制項的 **-p** 選項，將追蹤產生限制為指定的具名處理程序。例如，若要追蹤任何執行中處理程序 amqxxx.exe 所產生的所有執行緒，請使用下列指令：

```
strmqtrc -p amqxxx.exe
```

如需追蹤指令的詳細資訊，請參閱 [strmqtrc](#)。

相關概念

第 58 頁的『[在 UNIX and Linux 系統上使用追蹤](#)』

使用 **strmqtrc** 和 **endmqtrc** 指令來啟動和結束追蹤，以及使用 **dspmqtrc** 來顯示追蹤檔

第 60 頁的『[在 IBM i 上搭配使用追蹤與 IBM MQ 伺服器](#)』

使用 TRCMQM 指令來啟動和停止追蹤，並指定您需要的追蹤類型。

第 65 頁的『[在 z/OS 上使用追蹤來判斷問題](#)』

有不同的追蹤選項可用於 IBM MQ 的問題判斷。請利用這個主題來瞭解不同的選項，以及如何控制追蹤。

第 77 頁的『[追蹤 TLS: runmqakm、strmqikm 及 runmqckm 函數](#)』

如何追蹤「傳輸層安全 (TLS)」，並要求 **runmqakm** 追蹤及 **strmqikm** (iKeyman) 和 **runmqckm** (iKeycmd) 追蹤。

第 87 頁的『[追蹤其他 IBM MQ Java 元件](#)』

對於 IBM MQ 的 Java 元件 (例如 IBM MQ Explorer 及 IBM MQ Transport for SOAP 的 Java 實作)，診斷資訊是使用標準 IBM MQ 診斷機能或由 Java 診斷類別來輸出。

Linux

UNIX

在 UNIX and Linux 系統上使用追蹤

使用 **strmqtrc** 和 **endmqtrc** 指令來啟動和結束追蹤，以及使用 **dspmqtrc** 來顯示追蹤檔

UNIX and Linux 系統對 IBM MQ MQI client 追蹤機能使用下列指令：

strmqtrc

開始追蹤

endmqtrc

結束追蹤

dspmqtrc 檔名

顯示已格式化的追蹤檔

追蹤機能使用許多檔案，包括：

- 每一個要追蹤的實體各一個檔案，其中會記錄追蹤資訊
- 每一部機器上的另一個檔案，以提供用來啟動及結束追蹤之共用記憶體之參照
- 一個檔案，用於識別更新共用記憶體時使用的號誌

與追蹤相關聯的檔案會建立在檔案樹狀結構中的固定位置，即 `/var/mqm/trace`。

所有用戶端追蹤都會對這個目錄中的檔案進行。

您可以透過在此目錄上裝載暫存檔系統來處理大型追蹤檔案。

在 AIX 上，除了使用 **strmqtrc** 和 **endmqtrc** 指令之外，您還可以使用 AIX 系統追蹤。如需相關資訊，請參閱第 60 頁的『[使用 AIX 系統追蹤進行追蹤](#)』。

IBM MQ for UNIX 和 Linux 系統上的追蹤檔案

追蹤檔建立在 `/var/mqm/trace` 目錄中。

註：您可以將暫存檔系統裝載在包含追蹤檔的目錄上，以容納大型追蹤檔的正式作業。或者，重新命名追蹤目錄，並將符號鏈結 `/var/mqm/trace` 建立至不同的目錄。

追蹤檔命名為 `AMQppppp.qq.TRC`，其中變數如下：

ppppp

報告錯誤的處理程序 ID。

qq

序號，從 0 開始。如果完整檔名存在，則此值會加 1，直到找到唯一的追蹤檔名稱為止。如果重複使用處理程序，則追蹤檔名稱可以存在。

註：

1. 處理程序 ID 包含的數字可以比範例中顯示的數字更少或更多。
2. 作為所追蹤實體的一部分執行的每一個處理程序都有一個追蹤檔。

若要格式化或檢視追蹤檔，您必須是追蹤檔的建立者，或 `mqm` 群組的成員。

SSL 追蹤檔案具有名稱 `AMQ.SSL.TRC` 和 `AMQ.SSL.TRC.1`。您無法格式化 SSL 追蹤檔案；請將未變更的檔案傳送至 IBM 支援中心。

如何啟動及停止追蹤

在 IBM MQ for UNIX 和 Linux 系統中，您可以使用 **strmqtrc** 控制指令來啟用或修改追蹤 (請參閱 [strmqtrc](#))。若要停止追蹤，請使用 **endmqtrc** 控制指令 (請參閱 [endmqtrc](#))。在 IBM MQ for Linux (x86

和 x86-64 平台) 系統上, 您也可以使用 IBM MQ Explorer 來啟動和停止追蹤。不過, 您只能使用所提供的功能來追蹤所有項目, 相當於使用指令 `strmqtrc -e` 和 `endmqtrc -e`。

追蹤輸出未格式化; 在檢視之前, 請使用 `dspmqtrc` 控制指令來格式化追蹤輸出。例如, 若要格式化現行目錄中的所有追蹤檔, 請使用下列指令:

```
dspmqtrc *.TRC
```

如需控制指令 `dspmqtrc` 的詳細資訊, 請參閱 [dspmqtrc](#)。

IBM MQ for UNIX 和 Linux 系統上的選擇性元件追蹤

使用 `-t` 及 `-x` 選項來控制要記錄的追蹤明細數量。依預設, 會啟用所有追蹤點。使用 `-x` 選項指定您不想追蹤的點。例如, 如果您想要針對佇列管理程式 QM1 僅追蹤與使用「傳輸層安全 (TLS)」通道安全相關聯的輸出資料, 請使用:

```
strmqtrc -m QM1 -t ssl
```

如需追蹤指令的詳細資訊, 請參閱 [strmqtrc](#)。

IBM MQ for AIX 上的選擇性元件追蹤

請使用環境變數 `MQS_TRACE_OPTIONS` 來個別啟動高明細和參數追蹤函數。

因為 `MQS_TRACE_OPTIONS` 可讓追蹤處於作用中狀態, 而不需要高明細及參數追蹤功能, 所以當您嘗試重新產生啟用追蹤的問題時, 可以使用它來減少對效能及追蹤大小的影響。

只有在服務人員指示您這樣做時, 才設定環境變數 `MQS_TRACE_OPTIONS`。

通常必須在啟動佇列管理程式的處理程序中設定 `MQS_TRACE_OPTIONS`, 然後才能啟動佇列管理程式, 否則無法辨識該佇列管理程式。在開始追蹤之前設定 `MQS_TRACE_OPTIONS`。如果在追蹤開始之後設定, 則無法辨識。

IBM MQ for UNIX 和 Linux 系統上的選擇性處理程序追蹤

使用 `strmqtrc` 指令控制項的 `-p` 選項, 將追蹤產生限制為指定的具名處理程序。例如, 若要追蹤任何稱為 `amqxxx` 的執行中處理程序所產生的所有執行緒, 請使用下列指令:

```
strmqtrc -p amqxxx
```

如需追蹤指令的詳細資訊, 請參閱 [strmqtrc](#)。

相關概念

第 60 頁的『在 IBM i 上搭配使用追蹤與 IBM MQ 伺服器』
使用 `TRCMQM` 指令來啟動和停止追蹤, 並指定您需要的追蹤類型。

第 65 頁的『在 z/OS 上使用追蹤來判斷問題』
有不同的追蹤選項可用於 IBM MQ 的問題判斷。請利用這個主題來瞭解不同的選項, 以及如何控制追蹤。

第 77 頁的『追蹤 TLS: `runmqakm`、`strmqikm` 及 `runmqckm` 函數』
如何追蹤「傳輸層安全 (TLS)」, 並要求 `runmqakm` 追蹤及 `strmqikm` (iKeyman) 和 `runmqckm` (iKeycmd) 追蹤。

第 87 頁的『追蹤其他 IBM MQ Java 元件』
對於 IBM MQ 的 Java 元件 (例如 IBM MQ Explorer 及 IBM MQ Transport for SOAP 的 Java 實作), 診斷資訊是使用標準 IBM MQ 診斷機能或由 Java 診斷類別來輸出。

相關參考

第 56 頁的『在 Windows 上使用追蹤』
使用 `strmqtrc` 和 `endmqtrc` 指令或 IBM MQ Explorer 介面來啟動和結束追蹤。

使用 AIX 系統追蹤進行追蹤

除了 IBM MQ 追蹤之外，IBM MQ for AIX 使用者還可以使用標準 AIX 系統追蹤。

註：只有在 IBM 服務人員的指示下，您才應該使用 *aix* 選項。

AIX 系統追蹤是三個步驟的處理程序：

1. 將 `strmqtrc` 指令上的 `-o` 參數設為 *aix*。
2. 收集資料，並在完成之後執行 `endmqtrc` 指令。
3. 將結果格式化。

IBM MQ 使用兩個追蹤連結鉤 ID：

X'30D'

此事件由 IBM MQ 在進入或離開子常式時記錄。

X'30E'

IBM MQ 會記錄此事件以追蹤資料，例如透過通訊網路傳送或接收的資料。

追蹤提供詳細的執行追蹤，以協助您分析問題。IBM 服務支援人員可能會要求在啟用追蹤的情況下重建問題。追蹤所產生的檔案可能 **非常大**，因此儘可能限定追蹤是很重要的。例如，您可以選擇性地依時間和元件來限定追蹤。

有兩種方式可以執行追蹤：

1. 互動式。

下列指令序列會在程式 `myprog` 上執行互動式追蹤，並結束追蹤。

```
trace -j30D,30E -o trace.file
->!myprog
->q
```

2. 非同步。

下列一連串指令會在程式 `myprog` 上執行非同步追蹤，並結束追蹤。

```
trace -a -j30D,30E -o trace.file
myprog
trcstop
```

您可以使用下列指令來格式化追蹤檔：

```
trcrpt -t MQ_INSTALLATION_PATH/lib/amqtrc.fmt trace.file > report.file
```

`MQ_INSTALLATION_PATH` 代表 IBM MQ 安裝所在的高階目錄。

`report.file` 是您要放置格式化追蹤輸出的檔案名稱。

註：當追蹤處於作用中狀態時，會追蹤機器上的 **所有** IBM MQ 活動。

在 IBM i 上搭配使用追蹤與 IBM MQ 伺服器

使用 `TRCMQM` 指令來啟動和停止追蹤，並指定您需要的追蹤類型。

使用追蹤有兩個階段：

1. 決定您是否要提早追蹤。早期追蹤可讓您追蹤佇列管理程式的建立及啟動。不過請注意，早期追蹤很容易產生大量追蹤，因為它是透過追蹤所有佇列管理程式的所有工作來實作。若要啟用早期追蹤，請使用 `TRCEARLY` 參數設為 `*YES` 的 `TRCMQM`。
2. 使用 `TRCMQM *ON` 啟動追蹤工作。若要停止追蹤，您有兩個選項：
 - `TRCMQM *OFF`，停止收集佇列管理程式的追蹤記錄。追蹤記錄會寫入 `/QIBM/UserData/mqm/trace` 目錄中的檔案。

- TRCMQM *END, 停止收集所有佇列管理程式的追蹤記錄並停用早期追蹤。此選項會忽略 TRCEARLY 參數的值。

使用 TRCLEVEL 參數設定為下列其中一個值, 指定您想要的詳細程度:

***DFT**

適用於流程處理追蹤點的最小明細層次。

***DETAIL**

適用於流程處理追蹤點的高明細層次。

***PARMS**

對於流程處理追蹤點的預設明細層次。

使用 OUTPUT 參數設為下列其中一個值, 指定您想要的追蹤輸出類型:

***MQM**

在 TRCDIR 參數指定的目錄中收集二進位 IBM MQ 追蹤輸出。此值為預設值。

***MQMFMT**

在 TRCDIR 參數指定的目錄中收集格式化的 IBM MQ 追蹤輸出。

***PEX**

收集效能探測器 (PEX) 追蹤輸出

***ALL**

同時收集 IBM MQ 未格式化的追蹤和 PEX 追蹤輸出

選擇性追蹤

您可以使用指令 TRCMQM 搭配 F4=prompt, 然後 F9 來自訂 TRCTYPE 及 EXCLUDE 參數, 來減少要儲存的追蹤資料量, 以增進執行時期效能:

TRCTYPE

指定要儲存在追蹤檔中的追蹤資料類型。如果您省略此參數, 則會啟用 EXCLUDE 中指定的追蹤點以外的所有追蹤點。

EXCLUDE

指定要從追蹤檔中省略的追蹤資料類型。如果您省略此參數, 則會啟用 TRCTYPE 中指定的所有追蹤點。

在 TRCTYPE 及 EXCLUDE 上可用的選項如下:

***ALL (僅限 TRCTYPE)**

下列關鍵字指定的所有追蹤資料都會儲存在追蹤檔中。

trace-type-list

您可以從下列關鍵字指定多個選項, 但每一個選項只能出現一次。

***API**

輸出與 MQI 及主要佇列管理程式元件相關聯的追蹤點資料。

***CMTRY**

輸出與 IBM MQ 元件中的註解相關聯的追蹤點資料。

***COMS**

輸出與通訊網路上流動的資料相關聯的追蹤點資料。

***CSDATA**

輸出與共用服務中內部資料緩衝區相關聯的追蹤點資料。

***CSFLOW**

輸出與共用服務中處理流程相關聯的追蹤點資料。

***LQMDATA**

輸出與本端佇列管理程式中內部資料緩衝區相關聯的追蹤點資料。

***LQMFLOW**

輸出與本端佇列管理程式中處理流程相關聯的追蹤點資料。

***OTHDATA**

輸出與其他元件中內部資料緩衝區相關聯的追蹤點資料。

***OSTFlow**

輸出與其他元件中處理流程相關聯的追蹤點資料。

***RMTDATA**

輸出與通訊元件中內部資料緩衝區相關聯的追蹤點資料。

***RMTFLOW**

輸出與通訊元件中處理流程相關聯的追蹤點資料。

***SVCDATA**

輸出與服務元件中內部資料緩衝區相關聯的追蹤點資料。

***SVCFLOW**

輸出與服務元件中處理流程相關聯的追蹤點資料。

***VSNDATA**

輸出與執行的 IBM MQ 版本相關聯的追蹤點資料。

覆蓋追蹤

使用 MAXSTG 參數來折返追蹤，並指定要用於所收集追蹤記錄的儲存體大小上限。

選項有：

***DFT**

未啟用追蹤折返。對於每一個工作，追蹤資料會寫入字尾為 .TRC 的檔案，直到停止追蹤為止。

最大值-KB

已啟用追蹤折返。當追蹤檔達到其大小上限時，會以字尾 .TRS 重新命名，並開啟字尾為 .TRC 的新追蹤檔。會刪除任何現有的 .TRS 檔。請指定 1 到 16 000 範圍內的值。

格式化追蹤輸出

如果要格式化任何追蹤輸出，請執行下列動作：

- 輸入 QShell
- 輸入指令

```
/QSYS.LIB/QMQM.LIB/DSPMQTRC.PGM [-t Format] [-h] [-s]
[-o OutputFileName] InputFileName
```

其中：

InputFileName

是指定包含未格式化追蹤之檔案名稱的必要參數。例如 /QIBM/UserData/mqm/trace/AMQ12345.TRRC。

-t FormatTemplate

指定範本檔的名稱，其中包含如何顯示追蹤的詳細資料。預設值為 /QIBM/ProdData/mqm/lib/amqtrc.fmt。

-h

省略報告中的標頭資訊。

-s

擷取追蹤標頭並放入 stdout。

-o output_filename

要寫入格式化資料的檔案名稱。

您也可以指定 `dspmqttrc *` 來格式化所有追蹤。

相關概念

第 58 頁的『在 UNIX and Linux 系統上使用追蹤』

使用 `strmqtrc` 和 `endmqtrc` 指令來啟動和結束追蹤，以及使用 `dspmqttrc` 來顯示追蹤檔

第 65 頁的『在 z/OS 上使用追蹤來判斷問題』

有不同的追蹤選項可用於 IBM MQ 的問題判斷。請利用這個主題來瞭解不同的選項，以及如何控制追蹤。

第 77 頁的『[追蹤 TLS: runmqakm、strmqikm 及 runmqckm 函數](#)』

如何追蹤「傳輸層安全 (TLS)」，並要求 **runmqakm** 追蹤及 **strmqikm** (iKeyman) 和 **runmqckm** (iKeycmd) 追蹤。

第 87 頁的『[追蹤其他 IBM MQ Java 元件](#)』

對於 IBM MQ 的 Java 元件 (例如 IBM MQ Explorer 及 IBM MQ Transport for SOAP 的 Java 實作)，診斷資訊是使用標準 IBM MQ 診斷機能或由 Java 診斷類別來輸出。

相關參考

第 56 頁的『[在 Windows 上使用追蹤](#)』

使用 **strmqtrc** 和 **endmqtrc** 指令或 IBM MQ Explorer 介面來啟動和結束追蹤。

IBM i 在 IBM i 上搭配使用追蹤與 IBM MQ 用戶端

在 IBM i 上，使用獨立式 IBM MQ MQI client 時，沒有「控制語言 (CL)」指令可擷取追蹤。STRMQTRC 及 ENDMQTRC 程式可用來啟用及停用追蹤。

啟動追蹤的範例：

```
CALL PGM(QMQM/STRMQTRC) PARM('-e' '-t' 'all' '-t' 'detail')
Where -e option requests early tracing of all the process -t option for trace type
```

結束追蹤

```
CALL PGM(QMQM/ENDMQTRC) PARM('-e')
```

• 選用參數：

-t TraceType

要追蹤的點及要記錄的追蹤詳細資料量。依預設，會啟用所有追蹤點，並產生預設明細追蹤。

或者，您可以在表 1 中提供一個以上選項。對於您指定的每一個 *TraceType* 值 (包括 -t all)，請指定 -t parms 或 -t detail，以取得適當層次的追蹤明細。如果您沒有針對任何特定追蹤類型指定 -t parms 或 -t detail，則只會針對該追蹤類型產生預設明細追蹤。

如果提供多種追蹤類型，每種類型必須具有自己的 -t 旗標。您可以包含任意數目的 -t 旗標（如果每個旗標都有與其相關聯的有效追蹤類型）。

可以在多個 -t 旗標上指定相同的追蹤類型。

如需 *TraceType* 的容許值，請參閱下表。

值	說明
全部	輸出系統中每個追蹤點的資料（預設值）。使用 <i>all</i> 會在預設詳細資料層次啟動追蹤。
API	與訊息佇列介面 (MQI) 及主要佇列管理程式元件相關聯之追蹤點的輸出資料。
commentary	輸出與 IBM MQ 元件中的註解相關聯的追蹤點資料。
通訊	輸出與通訊網路上流動的資料相關聯的追蹤點資料。
csdata	輸出與共用服務中內部資料緩衝區相關聯的追蹤點資料。
csflows	輸出與共用服務中處理流程相關聯的追蹤點資料。
詳細資料	為流程處理追蹤點啟動高詳細資料層次追蹤。
lqmdata	輸出與本端佇列管理程式中內部資料緩衝區相關聯的追蹤點資料。

表 5: <i>TraceType</i> 值 (繼續)	
值	說明
lqmflows	輸出與本端佇列管理程式中處理流程相關聯的追蹤點資料。
其他資料	輸出與其他元件中內部資料緩衝區相關聯的追蹤點資料。
其他流程	輸出與其他元件中處理流程相關聯的追蹤點資料。
Parms	為流程處理追蹤點啟動預設詳細資料層次追蹤。
remote data	輸出與通訊元件中內部資料緩衝區相關聯的追蹤點資料。
remote flows	輸出與通訊元件中處理流程相關聯的追蹤點資料。
service data	輸出與服務元件中內部資料緩衝區相關聯的追蹤點資料。
服務	輸出與服務元件中處理流程相關聯的追蹤點資料。
version data	輸出與執行的 IBM MQ 版本相關聯的追蹤點資料。

-x *TraceType*

不追蹤的點。依預設，會啟用所有追蹤點，並產生預設明細追蹤。您可以指定的 *TraceType* 值與 [表 1](#) 中針對 **-t** 旗標列出的值相同。

您可以搭配使用 **-x** 旗標與 *TraceType* 值，以排除您不想記錄的那些追蹤點。排除指定的追蹤點有助於減少產生的追蹤數量。

如果您提供多個追蹤類型，則每一個都必須有自己的 **-x** 旗標。如果每一個旗標都有相關聯的有效 *TraceType*，您可以包含任意數目的 **-x** 旗標。

-s

報告目前已生效的追蹤選項。您必須單獨使用此參數，不得與其他參數一同使用。

可供儲存追蹤指令使用的插槽數目有限。當所有插槽皆在使用中時，除非取代現有插槽，否則無法再接受其他追蹤指令。插槽號碼不是固定的，因此如果移除插槽號碼 0 中的指令 (例如透過 **endmqtrc** 指令)，則所有其他插槽都會向上移動，例如插槽 1 變成插槽 0。欄位中的星號 (*) 表示未定義值，等同於星號萬用字元。

-l *MaxSize*

追蹤檔的大小上限 (AMQppppp.qq.TRC) 以 MB 為單位 (MB)。例如，如果您指定 *MaxSize* 為 1，則追蹤大小限制為 1 MB。

當追蹤檔達到指定的上限時，會將它重新命名為 AMQppppp.qq.TRS，並啟動新的 AMQppppp.qq.TRC 檔。如果存在 AMQppppp.qq.TRS 檔案的前一個副本，則會將其刪除。

MaxSize 的最高值可以是 2048 MB。

-e

要求提早追蹤所有處理程序

如需詳細資料，請參閱 [strmqtrc](#) 指令

- 若要結束追蹤:

```
/QSYS.LIB/QMQM.LIB/ENDMQTRC.PGM [-e] [-a]
```

其中:

-e

結束所有處理程序的早期追蹤。

使用不含任何參數的 **endmqtrc**，其效果與 **endmqtrc -e** 相同。您無法同時指定 **-e** 旗標與 **-m** 旗標、**-i** 旗標或 **-p** 旗標。

-a

結束所有追蹤。

如需詳細資料，請參閱 [endmqtrc endmqtrc 指令](#)

- 如果要顯示格式化的追蹤檔，請執行下列動作：

```
/QSYS.LIB/QMQM.LIB/DSPMQTRC.pgm
```

檢查 First Failure Support Technology (FFST) 檔案，請參閱 [第 54 頁的『FFST: IBM MQ for IBM i』](#)。

相關概念

[第 58 頁的『在 UNIX and Linux 系統上使用追蹤』](#)

使用 **strmqtrc** 和 **endmqtrc** 指令來啟動和結束追蹤，以及使用 **dspmqtrc** 來顯示追蹤檔

[第 65 頁的『在 z/OS 上使用追蹤來判斷問題』](#)

有不同的追蹤選項可用於 IBM MQ 的問題判斷。請利用這個主題來瞭解不同的選項，以及如何控制追蹤。

[第 77 頁的『追蹤 TLS: runmqakm、strmqikm 及 runmqckm 函數』](#)

如何追蹤「傳輸層安全 (TLS)」，並要求 **runmqakm** 追蹤及 **strmqikm** (iKeyman) 和 **runmqckm** (iKeycmd) 追蹤。

[第 87 頁的『追蹤其他 IBM MQ Java 元件』](#)

對於 IBM MQ 的 Java 元件 (例如 IBM MQ Explorer 及 IBM MQ Transport for SOAP 的 Java 實作)，診斷資訊是使用標準 IBM MQ 診斷機能或由 Java 診斷類別來輸出。

相關參考

[第 56 頁的『在 Windows 上使用追蹤』](#)

使用 **strmqtrc** 和 **endmqtrc** 指令或 IBM MQ Explorer 介面來啟動和結束追蹤。

z/OS 在 z/OS 上使用追蹤來判斷問題

有不同的追蹤選項可用於 IBM MQ 的問題判斷。請利用這個主題來瞭解不同的選項，以及如何控制追蹤。

IBM MQ for z/OS 可用的追蹤機能如下：

- 使用者參數 (或 API) 追蹤
- 支援中心使用的 IBM 內部追蹤
- 通道起始程式追蹤
- 線路追蹤

請使用下列鏈結，以瞭解如何收集及解譯使用者參數追蹤所產生的資料，並說明如何產生 IBM 內部追蹤，以供 IBM 支援中心使用。還有您可以與 IBM MQ 搭配使用之其他追蹤機能的相關資訊。

- [控制 z/OS 系統的 GTF](#)
- [控制您要收集其資料之每一個佇列管理程式子系統的 IBM MQ 追蹤](#)
- [第 68 頁的『在 z/OS 上格式化及識別控制區塊資訊』](#)
- [第 69 頁的『解譯 z/OS 上的追蹤資訊』](#)

如果未產生追蹤資料，請檢查下列項目：

- 是否已正確啟動 GTF，並在 USRP 選項上指定 EID 5E9、5EA 及 5EE？
- 是否正確地輸入 START TRACE (GLOBAL) 指令，以及是否指定相關類別？

如需 z/OS 上可用的其他追蹤選項的相關資訊，請參閱 [第 70 頁的『z/OS 上其他類型的追蹤』](#)。

相關概念

[第 58 頁的『在 UNIX and Linux 系統上使用追蹤』](#)

使用 **strmqtrc** 和 **endmqtrc** 指令來啟動和結束追蹤，以及使用 **dspmqtrc** 來顯示追蹤檔

第 60 頁的『在 IBM i 上搭配使用追蹤與 IBM MQ 伺服器』

使用 TRCMQM 指令來啟動和停止追蹤，並指定您需要的追蹤類型。

第 77 頁的『追蹤 TLS: runmqakm、strmqikm 及 runmqckm 函數』

如何追蹤「傳輸層安全 (TLS)」，並要求 **runmqakm** 追蹤及 **strmqikm** (iKeyman) 和 **runmqckm** (iKeycmd) 追蹤。

第 87 頁的『追蹤其他 IBM MQ Java 元件』

對於 IBM MQ 的 Java 元件 (例如 IBM MQ Explorer 及 IBM MQ Transport for SOAP 的 Java 實作)，診斷資訊是使用標準 IBM MQ 診斷機能或由 Java 診斷類別來輸出。

相關參考

第 56 頁的『在 Windows 上使用追蹤』

使用 **strmqtrc** 和 **endmqtrc** 指令或 IBM MQ Explorer 介面來啟動和結束追蹤。

z/OS MQI 呼叫及使用者參數，以及 z/OS 上的 GTF

請利用這個主題來瞭解如何控制 z/OS 一般性追蹤機能 (GTF) 和 IBM MQ 追蹤。

您可以在進入及結束 IBM MQ 時，取得部分 IBM MQ 呼叫所傳遞之 MQI 呼叫及使用者參數的相關資訊。若要這麼做，請將廣域追蹤與 z/OS 一般性追蹤機能 (GTF) 一起使用。

z/OS 啟動和停止 GTF

在 z/OS 上，您可以使用一般性追蹤機能 (GTF) 來記錄及診斷系統和程式問題。

關於這項作業

您可以在進入及結束 IBM MQ 時，取得部分 IBM MQ 呼叫所傳遞之 MQI 呼叫及使用者參數的相關資訊。若要這麼做，請將廣域追蹤與 z/OS 一般性追蹤機能 (GTF) 一起使用。

程序

- 輸入 **START GTF** 指令，在主控台啟動 GTF。
當您啟動 GTF 時，請指定 USRP 選項。系統會提示您輸入事件 ID (EID) 的清單。IBM MQ 使用的 EID 如下：

5E9

在進入 IBM MQ 時收集控制區塊的相關資訊

5EA

從 IBM MQ 結束時收集控制區塊的相關資訊

有時，如果發生無法自行解決的錯誤，IBM 支援中心可能會要求您提供其他內部追蹤資訊以供分析。其他追蹤類型為：

5EE

收集 IBM MQ 內部資訊

您也可以使用 **JOBNAMEP** 選項，指定批次、CICS、IMS 或 TSO 工作名稱，將追蹤輸出限制為特定工作。下列範例顯示 GTF 的啟動範例，指定四個 EID 和一個工作名稱。以 **粗體** 顯示的行是您在主控台輸入的指令；其他行是提示和回應。如需啟動 GTF 追蹤的相關資訊，請參閱 *MVS Diagnosis: Tools and Service Aids* 文件。

```
START GTFxx.yy
#HASP100 GTFxx.yy ON STCINRDR
#HASP373 GTFxx.yy STARTED
*01 AHL100A SPECIFY TRACE OPTIONS
R 01,TRACE=JOBNAMEP,USRP
TRACE=JOBNAMEP,USRP
IEE600I REPLY TO 01 IS;TRACE=JOBNAMEP,USRP
*02 ALH101A SPECIFY TRACE EVENT KEYWORDS - JOBNAME=,USR=
R 02,JOBNAME=(xxxxMSTR,xxxxCHIN,zzzzzzzz),USR=(5E9,5EA,5EE)
JOBNAME=(xxxxMSTR,xxxxCHIN,zzzzzzzz),USR=(5E9,5EA,5EE)
```

```

IEE600I REPLY TO 02 IS;JOBNAME=(xxxxMSTR,xxxxCHIN,zzzzzzzz),USR=(5E9,5EA,5EE)
*03 AHL102A CONTINUE TRACE DEFINITION OR REPLY END
R 03,END
END
IEE600I REPLY TO 03 IS;END
AHL103I TRACE OPTIONS SELECTED-USR=(5E9,5EA,5EE)
AHL103I JOBNAME=(xxxxMSTR,xxxxCHIN,zzzzzzzz)
*04 AHL125A RESPECIFY TRACE OPTIONS OR REPLY U
R 04,U
U
IEE600I REPLY TO 04 IS;U
AHL031I GTF INITIALIZATION COMPLETE

```

其中

- xx 是要使用的 GTF 程序名稱 (選用)
- yy 是這個 GTF 追蹤出現項目的 ID
- xxxx 是佇列管理程式的名稱
- zzzzzzzz 是批次工作或 CICS 區域名稱

最多可以列出 5 個工作名稱。

使用 GTF 時，除了佇列管理程式名稱 (xxxxMSTR) 之外，請指定主要工作名稱 (CHINIT、CICS 或 batch)。

- 在主控台停止 GTF。
當您輸入 GTF 的 stop 指令時，請包含您在啟動時使用的其他 ID (yy)，如下列範例所示：

```
STOP yy
```

相關資訊

[在 IBM z/OS 上產生 IBM MQ GTF 追蹤](#)

在 IBM MQ for z/OS 內控制追蹤

IBM MQ for z/OS 追蹤是使用 MQSC 指令來控制。請利用這個主題來瞭解如何控制追蹤，以及輸出的追蹤資訊類型。

使用 START TRACE 指令，指定類型 GLOBAL 以開始將 IBM MQ 記錄寫入 GTF。您也必須指定 dest (GTF)，例如在下列指令中：

```
/cpl start trace(G)class(2,3)dest(GTF)
```

若要定義您要為其產生追蹤資料的事件，請使用下列一個以上類別：

類別	已追蹤事件
2	當偵測到 MQRC_NONE 以外的完成碼時，請記錄 MQI 呼叫及 MQI 參數。
3	在進入及離開佇列管理程式時，記錄 MQI 呼叫及 MQI 參數。

啟動追蹤之後，您可以使用下列指令來顯示追蹤的相關資訊、變更追蹤的內容，以及停止追蹤：

- 顯示追蹤
- ALTER TRACE
- 停止追蹤

若要使用任何追蹤指令，您必須具有下列其中一項：

- 發出啟動及停止追蹤指令的權限 (追蹤權限)
- 發出顯示追蹤指令的權限 (顯示權限)

註：

1. 也可以透過起始設定輸入資料集來輸入追蹤指令。

2. 產生的追蹤資訊也會包含同步點流程的詳細資料-例如 PREPARE 和 COMMIT。

如需這些指令的相關資訊，請參閱 [MQSC 指令](#)。

z/OS 在 z/OS 上格式化及識別控制區塊資訊

擷取追蹤之後，必須將輸出格式化，並識別 IBM MQ 控制區塊。

- [格式化資訊](#)
- [識別與 IBM MQ 相關聯的控制區塊](#)
- [識別與控制區塊相關聯的事件 ID](#)

格式化資訊

若要格式化廣域追蹤所收集的使用者參數資料，請使用 [第 68 頁的圖 5](#) 或 IPCS GTFTRACE USR(*xxx*) 指令中顯示的批次工作，其中 *xxx* 是：

5E9

在進入 IBM MQ MQI 呼叫時，將控制區塊的相關資訊格式化。

5EA

在結束 IBM MQ MQI 呼叫時，將控制區塊的相關資訊格式化。

5EE

格式化 IBM MQ 內容的相關資訊。

您也可以指定 **JOBNAME**(*jobname*) 參數，以將格式化輸出限制為特定工作。

```
//S1 EXEC PGM=IKJEFT01,DYNAMNBR=20,REGION=4096K
//IPCSPARM DD DSN=SYS1.PARMLIB,DISP=SHR
//IPCSDDIR DD DSN=thlqual.ipcs.dataset.directory,DISP=SHR
//SYSTSPRT DD SYSOUT=*,DCB=(LRECL=137)
//IPCSTOC DD SYSOUT=*
//GTFIN DD DSN=gtf.trace,DISP=SHR
//SYSTSIN DD *
IPCS
SETDEF FILE(GTFIN) NOCONFIRM
GTFTRACE USR(5E9,5EA,5EE)
/*
//STEPLIB DD DSN=thlqual.SCSQAUTH,DISP=SHR
```

圖 5: 批次格式化 GTF 輸出

識別與 IBM MQ 相關聯的控制區塊

IBM MQ 追蹤的格式 ID 為 D9。此值出現在格式化 GTF 輸出中每一個格式化控制區塊的開頭，格式如下：

```
USR09
```

識別與控制區塊相關聯的事件 ID

追蹤格式製作程式會在每一個控制區塊的開頭插入下列其中一個訊息。這些訊息指出資料是在進入或結束 IBM MQ 時擷取：

- CSQW072I ENTRY: MQ 使用者參數追蹤
- CSQW073I EXIT: MQ 使用者參數追蹤

相關工作

[第 66 頁的『啟動和停止 GTF』](#)

在 z/OS 上，您可以使用一般性追蹤機能 (GTF) 來記錄及診斷系統和程式問題。

z/OS 解譯 z/OS 上的追蹤資訊

可以檢查 IBM MQ 所產生的 GTFTRACE，以判定位址無效、控制區塊無效及資料無效的可能錯誤。

當您查看 GTFTRACE 指令所產生的資料時，請考量下列要點：

- 如果控制區塊完全由零組成，則可能在從使用者位址空間複製資料時發生錯誤。這可能是因為所傳遞的位址無效。
- 如果控制區塊的第一部分包含非空值資料，但其餘部分由零組成，則再次可能在從使用者位址空間複製資料時發生錯誤，例如，控制區塊未完全放置在有效的儲存體內。這也可能是因為未正確起始設定控制區塊。
- 如果從 IBM MQ 結束時發生錯誤，則 IBM MQ 可能不會將資料寫入使用者的位址空間。顯示的資料是它嘗試複製到使用者位址空間的版本。

下表顯示所追蹤控制區塊的詳細資料。

第 69 頁的表 6 說明針對不同的 MQI 呼叫追蹤哪些控制區塊。

MQI 呼叫	項目	結束
MQCB	MQCBD、MQMD、MQGMO	MQCBD、MQMD、MQGMO
MQCLOSE	無	無
MQGET	MQMD, MQGMO	MQMD、MQGMO 及訊息資料的前 256 個位元組
MQINQ	選取元 (如果 <i>SelectorCount</i> 大於 0)	選取元 (如果 <i>SelectorCount</i> 大於 0) 整數屬性 (如果 <i>IntAttrCount</i> 大於 0) 字元屬性 (如果 <i>CharAttrLength</i> 大於 0)
MQOPEN	MQOD	MQOD
MQPUT	MQMD、MQPMO 及訊息資料的前 256 個位元組	MQMD、MQPMO 及訊息資料的前 256 個位元組
MQPUT1	MQMD、MQOD、MQPMO 及訊息資料的前 256 個位元組	MQMD、MQOD、MQPMO 及訊息資料的前 256 個位元組
MQSET	選取元 (如果 <i>SelectorCount</i> 大於 0) 整數屬性 (如果 <i>IntAttrCount</i> 大於 0) 字元屬性 (如果 <i>CharAttrLength</i> 大於 0)	選取元 (如果 <i>SelectorCount</i> 大於 0) 整數屬性 (如果 <i>IntAttrCount</i> 大於 0) 字元屬性 (如果 <i>CharAttrLength</i> 大於 0)
MQSTAT	MQSTS	MQSTS
MQSUB	MQSD、MQSD.ObjectString、 MQSD.SubName、 MQSD.SubUserData、 MQSD.SelectionString、 MQSD.ResObjectString	MQSD、MQSD.ObjectString、 MQSD.SubName、 MQSD.SubUserData、 MQSD.SelectionString、 MQSD.ResObjectString
MQSUBRQ	MQSRO	MQSRO

註: 在使用 WAIT 選項進行 MQGET 呼叫的特殊情況下，如果在 MQGET 要求時沒有可用的訊息，但隨後有訊息在任何指定的時間間隔到期之前變成可用，則會看到雙重項目。

這是因為，雖然應用程式已發出單一 MQGET 呼叫，但配接器會代表應用程式執行等待，當訊息變成可用時，它會重新發出呼叫。因此在追蹤中，它會顯示為第二個 MQGET 呼叫。


在某些情況下，也會產生佇列要求參數清單的特定欄位相關資訊。此清單中的欄位識別如下：

ID	說明
動作	所要求的動作
BufferL	緩衝區長度
CBD	回呼描述子的位址
CompCode	完成碼
CharAttL	字元屬性長度
DataL	資料長度
HOBJ	物件控點
HSub	訂閱控點
IntAttC	整數屬性計數
pObj 說明	物件描述子 (object descriptor)
作業	作業
選項	選項
pBuffer	緩衝區位址
pChar 屬性	字元屬性的位址
pCTLO	控制回呼選項的位址
pECB	get 中使用的 ECB 位址
pGMO	取得訊息選項的位址
pInt 屬性	整數屬性的位址
pMsg 說明	訊息描述子的位址
pPMO	放置訊息選項的位址
pSD	訂閱描述子的位址
pSelect	選取元的位址
pSRQOpt	訂閱要求選項的位址
pSTS	狀態結構的位址
原因	原因碼
RSVn	保留給 IBM
SelectC	選取器計數
執行緒	執行緒
類型	所要求的類型
UOWInfo	工作單元的相關資訊
使用者 ID	CICS 或 IMS 使用者 ID，若為批次或 TSO，則此為零

z/OS 上其他類型的追蹤

有其他追蹤機能可用來判斷問題。請利用這個主題來調查通道起始程式追蹤、線路追蹤、CICS 配接器追蹤、SSL 追蹤及 z/OS 追蹤。

將下列追蹤機能與 IBM MQ 搭配使用可能很有用。

- [通道起始程式追蹤](#)
- [線路追蹤](#)
- [CICS 配接器追蹤](#)
- [系統 SSL 追蹤](#)
-  [z/OS 追蹤](#)

通道起始程式追蹤

如需如何取得通道起始程式位址空間傾出的相關資訊，請參閱第 106 頁的圖 10。請注意，通道起始程式所產生的傾出不包含追蹤資料空間。追蹤資料空間（稱為 CSQXTRDS）包含追蹤資訊。您可以透過在 slip 設陷上或在使用 dump 指令時指定它來要求此要求。

您可以使用 START TRACE 指令來執行追蹤。您也可以使用 TRAXSTR 佇列管理程式屬性，將此追蹤設為自動啟動。如需如何執行此動作的相關資訊，請參閱 [ALTER QMGR](#)。

您可以輸入 IPCS 指令來顯示此追蹤資訊：

```
LIST 1000. DSPNAME(CSQXTRDS)
```

您可以使用下列指令來格式化它：

```
CTRACE COMP(CSQX $ssnm$ )
```

其中 $ssnm$ 是子系統名稱。

線路追蹤

每一個通道都存在繞行追蹤。此追蹤會針對通道起始程式位址空間中的每一個通道，保留在 4 KB 緩衝區中。每一個通道都會產生追蹤，因此它很適合通道似乎當掉的問題，因為在正常追蹤已覆蓋很久之後，可以收集此通道活動的相關資訊。

線路追蹤一律在作用中；您無法關閉它。它同時適用於 LU 6.2 及 TCP 通道，且應該會減少需要通訊追蹤的次數。

您可以將追蹤視為寫入 CSQSNAP 的未格式化追蹤。您可以遵循下列步驟來顯示追蹤：

1. 請確定 CHIN 程序有 SNAP DD 陳述式。
2. 指定 IFCID 202，以啟動 CHIN 追蹤，如下所示：

```
START TRACE(CHINIT) CLASS(4) IFCID(202)
```

3. 顯示需要線路追蹤之通道的通道狀態：

```
DISPLAY CHSTATUS(channel) SAVED
```

這會將所選通道的現行線路傾出至 CSQSNAP。如需進一步資訊，請參閱第 120 頁的『[z/OS 上的 SNAP 傾出](#)』。

註：

- a. 儲存體傾出的位址不正確，因為 CSQXFFST 機制在將儲存體寫入 CSQSNAP 之前會先取得儲存體副本。
- b. 只有在您第一次執行 DISPLAY CHSTATUS SAVED 指令時，才會產生傾出至 CSQSNAP。這是為了防止每次執行指令時都取得傾出。

若要取得線路追蹤資料的另一個傾出，您必須停止並重新啟動現行追蹤。

- i) 您可以使用選擇性 STOP TRACE 指令，只停止為了收集行追蹤資料而啟動的追蹤。若要執行此動作，請記下指派給追蹤的 TRACE NUMBER，如下列範例所示：

```
+ssid START TRACE(CHINIT) CLASS(4) IFCID(202)
CSQW130I +ssid 'CHINIT' TRACE STARTED, ASSIGNED TRACE NUMBER 01
```

- ii) 若要停止追蹤，請發出下列指令：

```
+ssid STOP TRACE(CHINIT) TNO(01)
```

- iii) 然後，您可以輸入另一個 START TRACE 指令及 DISPLAY CHSTATUS SAVED 指令，以收集 CSQSNAP 的更多行追蹤資料。

4. 線路追蹤緩衝區未格式化。每一個項目都以時鐘開頭，後面接著時間戳記，並指出這是 OUTBOUND 或 INBOUND 流程。使用時間戳記資訊來尋找最早的項目。

CICS 配接器追蹤

如果您的追蹤號碼設為 0 到 199 (十進位) 範圍內的值，且如果下列任一的話，則 CICS 配接器會將項目寫入 CICS 追蹤：

- 已啟用 CICS 使用者追蹤，或
- 已啟用 CICS 內部/輔助追蹤

您可以使用下列兩種方式之一來啟用 CICS 追蹤：

- 使用 CICS 提供的交易 CETR 動態
- 確定 CICS 系統起始設定表格 (SIT) 中的 USERTR 參數設為 YES

如需啟用 CICS 追蹤的相關資訊，請參閱 *CICS 問題判斷手冊*。

源自 CICS 配接卡的 CICS 追蹤項目具有值 AP0 000，其中 000 是您指定之 CICS 配接卡追蹤號碼的十進位相等值。

追蹤項目顯示在 [第 72 頁的『CICS 配接器追蹤項目』](#)中。

系統 SSL 追蹤

您可以使用「SSL 已啟動作業」來收集「系統 SSL」追蹤。如需如何設定此作業的詳細資料，請參閱 *System Secure Sockets Layer Programming* 文件 (SC24-5901)。會針對在 CHINIT 位址空間中執行的每一個 SSLTASK 產生追蹤檔。

z/OS 追蹤

z/OS

z/OS 追蹤是作為 z/OS 的正式子系統運作的所有產品所共用的，可與 IBM MQ 搭配使用。如需使用及解譯此追蹤機能的相關資訊，請參閱 *z/OS MVS Diagnosis: Tools and Service Aids* 手冊。

z/OS

CICS 配接器追蹤項目

請使用這個主題作為 CICS 配接器追蹤項目的參照。

這些值的 CICS 追蹤項目是 AP0 xxx (其中 xxx 是您在啟用 CICS 配接卡時所指定追蹤號碼的十六進位對等項目)。這些追蹤項目全部由 CSQCTRUE 發出，但 CSQCTEST 除外，CSQCRST 和 CSQCDSP 會發出。

表 7: CICS 配接器追蹤項目			
姓名	說明	追蹤順序	追蹤資料
CSQCABNT	異常終止	在對 IBM MQ 發出 END_THREAD ABNORMAL 之前。這是因為作業已結束，因此應用程式可以執行隱含的取消。在此情況下，END_THREAD 呼叫中包含 ROLLBACK 要求。	工作單元資訊。在瞭解工作狀態時，您可以使用此資訊。(例如，可以根據 DISPLAY THREAD 指令或日誌列印公用程式所產生的輸出來驗證它。)
CSQCAUID	橋接器安全	在驗證橋接器使用者密碼或 PassTicket 之前。	使用者 ID。
CSQCBACK	同步點取消	在對 IBM MQ 發出 BACKOUT 之前。這是因為來自應用程式的明確取消要求。	工作單元資訊。
CSQC CONX	MQCONN	在向 IBM MQ 發出 MQCONN 之前。	連線標籤。
CSQCCRC	完成碼和原因碼	從 API 呼叫傳回失敗之後。	完成碼和原因碼。
CSQCCOMM	同步點確定	在對 IBM MQ 發出 COMMIT 之前。這可能是由於一段式確定要求或兩段式確定要求的第二階段。要求是由於來自應用程式的明確同步點要求。	工作單元資訊。
CSQCDIFF	僅限 IBM 使用		
CSQDCIN	僅限 IBM 使用		
CSQDCOT	僅限 IBM 使用		
CSQCEXER	執行解析	對 IBM MQ 發出 EXECUTE_RESOLVE 之前。	發出 EXECUTE_RESOLVE 之工作單元的工作單元資訊。這是重新同步處理程序中最後一個不確定的工作單元。
CSQCGETW	GET 等待	在發出 CICS 等待之前。	要等待的歐洲央行位址。
CSQCGMGD	GET 訊息資料	從 MQGET 順利傳回之後。	最多 40 個位元組的訊息資料。
CSQCGMGH	GET 訊息控點	在對 IBM MQ 發出 MQGET 之前。	物件控點。
CSQCGMGI	取得訊息 ID	從 MQGET 順利傳回之後。	訊息的訊息 ID 和相關性 ID。
CSQCHCER	Hconn 錯誤	在發出任何 MQ 動詞之前。	連線控點。
CSQCINDL	不確定清單	順利從第二個 INQUIRE_INDOUBT 傳回之後。	不確定的工作單元清單。
CSQCINDO	僅限 IBM 使用		
CSQCINDS	不確定清單大小	順利從第一個 INQUIRE_INDOUBT 傳回之後，且不確定清單不是空的。	清單長度；除以 64 會提供不確定的工作單元數目。
CSQCINDW	不確定的同步點	在同步點處理期間，CICS 不確定工作單元的處置。	工作單元資訊。
CSQCINQH	INQ 控點	在將 MQINQ 發出至 IBM MQ 之前。	物件控點。
CSQCLOSH	CLOSE 控點	在向 IBM MQ 發出 MQCLOSE 之前。	物件控點。
CSQCLOST	遺失處置	在重新同步處理程序期間，CICS 會通知配接卡已冷啟動，因此沒有任何關於重新同步化工作單元的處置資訊可用。	要重新同步化之工作單元的 CICS 已知工作單元 ID。

表 7: CICS 配接器追蹤項目 (繼續)			
姓名	說明	追蹤順序	追蹤資料
CSQCNIND	處置未處於不確定狀態	在重新同步處理程序期間，CICS 會通知配接卡正在重新同步的工作單元不應處於不確定狀態 (亦即，可能它仍在執行中)。	要重新同步化之工作單元的 CICS 已知工作單元 ID。
CSQCNORT	正常終止	在對 IBM MQ 發出 END_THREAD NORMAL 之前。這是因為作業結束，因此應用程式可能會執行隱含的同步點確定。在此情況下，END_THREAD 呼叫中包含 COMMIT 要求。	工作單元資訊。
CSQCOPNH	OPEN 控點	從 MQOPEN 順利傳回之後。	物件控點。
CSQCOPNO	開啟物件	在對 IBM MQ 發出 MQOPEN 之前。	物件名稱。
CSQCPMGD	PUT 訊息資料	在對 IBM MQ 發出 MQPUT 之前。	最多 40 個位元組的訊息資料。
CSQCPMGH	PUT 訊息控點	在對 IBM MQ 發出 MQPUT 之前。	物件控點。
CSQCPMGI	PUT 訊息 ID	從 IBM MQ 順利完成 MQPUT 之後。	訊息的訊息 ID 和相關性 ID。
CSQCPREP	同步點準備	在兩段式確定處理的第一階段中對 IBM MQ 發出 PREPARE 之前。也可以從分散式佇列元件發出此呼叫作為 API 呼叫。	工作單元資訊。
CSQCP1MD	PUTONE 訊息資料	在向 IBM MQ 發出 MQPUT1 之前。	最多 40 個位元組的訊息資料。
CSQCP1MI	PUTONE 訊息 ID	從 MQPUT1 順利傳回之後。	訊息的訊息 ID 和相關性 ID。
CSQCP1ON	PUTONE 物件名稱	在向 IBM MQ 發出 MQPUT1 之前。	物件名稱。
CSQCRBAK	已解決取消	在對 IBM MQ 發出 RESOLVE_ROLLBACK 之前。	工作單元資訊。
CSQRCMT	已解決確定	在對 IBM MQ 發出 RESOLVE_COMMIT 之前。	工作單元資訊。
CSQCRMIR	RMI 回應	從特定呼叫回到 CICS RMI (資源管理程式介面) 之前。	已架構的 RMI 回應值。其意義取決於呼叫的類型。若要判斷呼叫類型，請查看 CICS RMI 元件所產生的先前追蹤項目。
CSQCRSYN	重新同步	在針對作業啟動重新同步處理程序之前。	要重新同步化之工作單元的 CICS 已知工作單元 ID。
CSQCSETH	SET 控點	在向 IBM MQ 發出 MQSET 之前。	物件控點。
CSQCTASE	僅限 IBM 使用		
CSQCTEST	追蹤測試	在 EXEC CICS ENTER TRACE 呼叫中使用，以驗證使用者提供的追蹤號碼或連線的追蹤狀態。	沒有資料。

啟用 AMSM 系統的內部追蹤

可以使用 `_AMS_MSG_LEVEL` 變數來啟用 AMSM 位址空間的追蹤，該變數會透過 ENVARS DD 卡傳遞至 AMSM 位址空間。

ENVARS DD 卡的範例資料集位於 th1qua1.SCSQPROC (CSQ40ENV) 中。

追蹤會寫入 AMSM 位址空間的 SYSOUT。

`_AMS_MSG_LEVEL` 變數指定要記載的子元件及訊息層次。星號指出要記載的所有子元件；目前只有一個子元件。

嚴重性層次如下：

- S-僅限嚴重訊息
- E-僅錯誤及嚴重訊息
- W-僅警告、錯誤及嚴重訊息
- I-參考資訊、警告、錯誤及嚴重訊息。這是預設值
- D-除錯模式，具有其他除錯診斷程式的所有訊息
- V-詳細模式，所有之前的，加上緩衝區傾出



小心：您只應該根據 IBM 客戶服務代表的建議來啟用除錯或詳細模式。

例如，若要啟用 `_AMS_MSG_LEVEL` 的預設值，請發出下列指令：

```
_AMS_MSG_LEVEL=* .i
```

若要啟用詳細模式，請發出下列指令：

```
_AMS_MSG_LEVEL=* .v
```

追蹤 Advanced Message Queuing Protocol (AMQP) 服務

提供「進階訊息佇列作業通訊協定 (AMQP) 服務」所提供的追蹤機能，可協助「IBM 支援中心」診斷與服務相關的客戶問題。

關於這項作業

有兩種方法可控制 IBM MQ AMQP 服務的追蹤：

- 使用 **strmqtrc** 和 **endmqtrc** 指令來啟動和停止追蹤。使用 **strmqtrc** 指令啟用追蹤會產生 IBM MQ AMQP 服務執行所在之整個佇列管理程式的追蹤資訊。這包括 IBM MQ AMQP 服務本身，以及服務用來與其他佇列管理程式元件進行通訊的基礎「Java 訊息佇列作業介面 (JMQUI)」。
- 執行 **controlAMQPChannel** 指令。請注意，使用 **controlAMQPChannel** 指令開啟追蹤只會追蹤 IBM MQ AMQP 服務。

如果您不確定要使用哪個選項，請聯絡「IBM 支援中心」代表，他們將能夠建議您針對您所看到的問題收集追蹤的最佳方式。

程序

1. 方法一

- a) 啟動命令提示字元，並導覽至下列目錄：

```
MQ_INSTALLATION_PATH\bin
```

- b) 執行 **strmqtrc** 指令以啟用追蹤：

```
strmqtrc -m qmgr_name
```

其中 `qmgr_name` 是 IBM MQ AMQP 服務執行所在佇列管理程式的名稱。

- c) 重新產生問題。
- d) 執行下列指令來停止追蹤：

```
endmqtrc -m qmgr_name
```


2. 方法二。

a) 啟動命令提示字元，並導覽至下列目錄：

```
MQ_INSTALLATION_PATH\bin
```

b) 執行下列指令以啟用追蹤：

• **Windows**

```
controlAMQPChannel -qmgr=qmgr_name -mode=starttrace
```

• **Linux** **UNIX**

```
./controlAMQPChannel.sh -qmgr=qmgr_name -mode=starttrace
```

其中 *qmgr_name* 是「AMQP 服務」執行所在佇列管理程式的名稱。

c) 重新產生問題。

d) 當問題發生時，請執行下列指令來停止追蹤：

• **Windows**

```
controlAMQPChannel -qmgr=qmgr_name -mode=stoptrace
```

• **Linux** **UNIX**

```
./controlAMQPChannel.sh -qmgr=qmgr_name -mode=stoptrace [clientid=ClientIdentifier]
```

其中 *qmgr_name* 是「AMQP 服務」執行所在佇列管理程式的名稱。

結果

若要檢視追蹤輸出，請跳至下列目錄：

• **Windows** MQ_DATA_PATH\trace.

• **Linux** **UNIX** /var/mqm/trace.

包含來自 AMQP 服務的資訊的追蹤檔案稱為 *amqp_N.trc*，其中 N 是數字。

JMQI 產生的追蹤資訊會寫入稱為 *amqp_PPPPP.trc* 的追蹤檔中，其中 *PPPPP* 是「AMQP 服務」的處理程序 ID。

Windows **Linux** **AIX** 使用 controlAMQPChannel 指令的其他診斷程式

使用 **controlAMQPChannel** 指令來提供 AMQP 服務的其他診斷資訊。

程序

執行下列指令，以從 MQXR 服務提供有用的診斷資訊：

```
<MQ_INSTALLATION_PATH>\amqp\bin\controlAMQPChannel -qmgr=<QMGR_NAME> -mode=diagnostics  
-diagnosticstype=<number>
```

產生的診斷資訊取決於 **-diagnosticstype=<number>** 參數的值：

-diagnosticstype= 0

寫入主控台的執行緒傾出

-diagnosticstype= 1

具有部分內部服務統計資料的 FDC

-diagnosticstype= 2

具有內部統計資料的 FDC，以及目前所連接用戶端的相關資訊

-diagnosticstype= 3

資料堆傾出

-diagnosticstype= 4

雅瓦科雷

-diagnosticstype= 5

完整系統傾出

-diagnosticstype= 6

特定用戶端的詳細資訊。請注意，您也必須為該用戶端提供 **-clientid** 參數。

追蹤 TLS: **runmqakm**、**strmqikm** 及 **runmqckm** 函數

如何追蹤「傳輸層安全 (TLS)」，並要求 **runmqakm** 追蹤及 **strmqikm** (iKeyman) 和 **runmqckm** (iKeycmd) 追蹤。

strmqikm 和 **runmqckm** 追蹤

若要要求 **strmqikm** 追蹤，請使用下列 -D 旗標針對您的平台執行 **strmqikm** 指令。

在 UNIX, Linux, and Windows 上:

```
strmqikm -Dkeyman.debug=true -Dkeyman.jnitracng=ON
```

若要要求 **runmqckm** 追蹤，請使用下列 -D 旗標針對您的平台執行 **runmqckm** 指令。

在 UNIX, Linux, and Windows 上:

```
runmqckm -Dkeyman.debug=true -Dkeyman.jnitracng=ON
```

strmqikm 及 **runmqckm** 會將三個追蹤檔寫入您從中啟動它們的目錄，因此請考量從執行時期 TLS 追蹤寫入其中的追蹤目錄啟動 iKeyman 或 **runmqckm**：/var/mqm/trace (在 UNIX and Linux 系統上) 及 MQ_INSTALLATION_PATH/trace (在 Windows 上)。MQ_INSTALLATION_PATH 代表 IBM MQ 安裝所在的高階目錄。

strmqikm 和 **runmqckm** 所產生的追蹤檔具有下列格式:

```
debugTrace. n
```

其中 *n* 是從 0 開始的遞增數字。

runmqakm trace

若要要求 **runmqakm** 追蹤，請使用下列旗標來執行 **runmqakm** 指令:

```
runmqakm -trace filename
```

其中 *filename* 是要建立的追蹤檔名稱。您無法格式化 **runmqakm** 追蹤檔。將它原封不動地傳送給 IBM 支援中心。**runmqakm** 追蹤檔是二進位檔，如果它透過 FTP 傳送至 IBM 支援中心，則必須以二進位傳送模式傳送。

執行時期 TLS 追蹤

在 UNIX, Linux, and Windows 系統上，您可以獨立要求 **strmqikm**、**runmqckm**、執行時期 TLS 功能或這些功能的組合的追蹤資訊。

執行時期 TLS 追蹤檔案具有名稱 AMQ.TLS.TRC 和 AMQ.TLS.TRC.1 及 TLS 追蹤檔案具有名稱 AMQ.SSL.TRC 和 AMQ.SSL.TRC.1。您無法格式化任何 TLS 追蹤檔案; 請將未變更的追蹤檔案傳送至 IBM 支援中心。TLS 追蹤檔案是二進位檔，如果透過 FTP 將它們傳送至 IBM 支援中心，則必須以二進位傳送模式傳送它們。

相關概念

第 58 頁的『在 UNIX and Linux 系統上使用追蹤』

使用 **strmqtrc** 和 **endmqtrc** 指令來啟動和結束追蹤，以及使用 **dspmqtrc** 來顯示追蹤檔

第 60 頁的『在 IBM i 上搭配使用追蹤與 IBM MQ 伺服器』

使用 TRCMQM 指令來啟動和停止追蹤，並指定您需要的追蹤類型。

第 65 頁的『在 z/OS 上使用追蹤來判斷問題』

有不同的追蹤選項可用於 IBM MQ 的問題判斷。請利用這個主題來瞭解不同的選項，以及如何控制追蹤。

第 87 頁的『追蹤其他 IBM MQ Java 元件』

對於 IBM MQ 的 Java 元件 (例如 IBM MQ Explorer 及 IBM MQ Transport for SOAP 的 Java 實作)，診斷資訊是使用標準 IBM MQ 診斷機能或由 Java 診斷類別來輸出。

相關參考

第 56 頁的『在 Windows 上使用追蹤』

使用 **strmqtrc** 和 **endmqtrc** 指令或 IBM MQ Explorer 介面來啟動和結束追蹤。

追蹤 IBM MQ classes for JMS 應用程式

提供 IBM MQ classes for JMS 中的追蹤機能，以協助「IBM 支援中心」診斷客戶問題。各種內容會控制此機能的行為。

如果要求您提供追蹤輸出來調查問題，請使用下列其中一個選項：

- 如果問題易於重建，請使用 Java 系統內容來收集 IBM MQ classes for JMS 追蹤。如需相關資訊，請參閱第 79 頁的『使用 Java 系統內容來收集 IBM MQ classes for JMS 追蹤』。
- 如果應用程式需要在發生問題之前執行一段時間，請使用 IBM MQ classes for JMS 配置檔來收集 IBM MQ classes for JMS 追蹤。如需相關資訊，請參閱第 80 頁的『使用 IBM MQ classes for JMS 配置檔來收集 IBM MQ classes for JMS 追蹤』。
- 如果要從目前執行中的應用程式產生追蹤，請使用 traceControl 公用程式來動態收集 IBM MQ classes for JMS 追蹤。如需相關資訊，請參閱第 81 頁的『使用 traceControl 公用程式來動態收集 IBM MQ classes for JMS 追蹤』。

如果您不確定要使用哪個選項，請聯絡「IBM 支援中心」代表，他們將能夠建議您針對您所看到的問題收集追蹤的最佳方式。

如果發生嚴重或無法復原的錯誤，First Failure Support Technology (FFST) 資訊會記錄在名稱格式為 JMSSC *xxxx*.FDC 的檔案中，其中 *xxxx* 是四位數的數字。此數目會遞增以區分 .FDC 檔案。

.FDC 檔案一律會寫入稱為 FFDC 的子目錄。子目錄位於下列兩個位置之一，視追蹤是否作用中而定：

追蹤作用中，且已設定 **traceOutputName**

FFDC 目錄會建立為要寫入追蹤檔之目錄的子目錄。

追蹤不在作用中或未設定 **traceOutputName**

FFDC 目錄會建立為現行工作目錄的子目錄。

如需 IBM MQ classes for JMS 中 FFST 的相關資訊，請參閱第 46 頁的『FFST: IBM MQ classes for JMS』。

JSE 共用服務使用 `java.util.logging` 作為其追蹤及記載基礎架構。此基礎架構的根物件是 `LogManager`。日誌管理程式有一個 `reset` 方法，可關閉所有處理程式，並將記載層次設為 `null`，這實際上會關閉所有追蹤。如果您的應用程式或應用程式伺服器呼叫

`java.util.logging.LogManager.getLogManager().reset()`，它會關閉所有追蹤，這可能會讓您無法診斷任何問題。如果要避免關閉所有追蹤，請建立 `LogManager` 類別，且所置換的 `reset()` 方法不會執行任何動作，如下列範例所示：

```
package com.ibm.javaut.tests;
import java.util.logging.LogManager;
public class JmsLogManager extends LogManager {
    // final shutdown hook to ensure that the trace is finally shutdown
    // and that the lock file is cleaned-up
    public class ShutdownHook extends Thread{
        public void run(){
```

```

        doReset();
    }
}
public JmsLogManager(){
    // add shutdown hook to ensure final cleanup
    Runtime.getRuntime().addShutdownHook(new ShutdownHook());
}
public void reset() throws SecurityException {
    // does nothing
}
public void doReset(){
    super.reset();
}
}
}

```

需要關閉連結鉤，以確保在 JVM 完成時適當地關閉追蹤。若要使用已修改的日誌管理程式而非預設日誌管理程式，請將系統內容新增至 JVM 啟動：

```
java -Djava.util.logging.manager=com. mycompany.logging.LogManager ...
```

使用 Java 系統內容來收集 IBM MQ classes for JMS 追蹤

對於可在短時間內重新產生的問題，應該在啟動應用程式時設定 Java 系統內容來收集 IBM MQ classes for JMS 追蹤。

關於這項作業

若要使用 Java 系統內容來收集追蹤，請完成下列步驟。

程序

- 使用下列指令來執行要追蹤的應用程式：

```
java -Dcom.ibm.msg.client.commonservices.trace.status=ON application_name
```

依預設，追蹤資訊會寫入應用程式現行工作目錄中的追蹤檔。追蹤檔的名稱視應用程式執行所在的環境而定：

- 若為 IBM MQ classes for JMS for IBM MQ 9.0.0 Fix Pack 1 或更早版本，追蹤會寫入稱為 mqjms_*%PID%*.trc 的檔案。
- **V 9.0.0.2** 從 IBM MQ 9.0.0 Fix Pack 2 開始，如果應用程式已從 JAR 檔 com.ibm.mqjms.jar 載入 IBM MQ classes for JMS，追蹤會寫入稱為 mqjava_*%PID%*.trc 的檔案。
- **V 9.0.0.2** 從 IBM MQ 9.0.0 Fix Pack 2，如果應用程式已從可重新定位 JAR 檔 com.ibm.mq.allclient.jar 載入 IBM MQ classes for JMS，則會將追蹤寫入稱為 mqjavaclient_*%PID%*.trc 的檔案。
- **V 9.0.0.10** 從 IBM MQ 9.0.0 Fix Pack 10 開始，如果應用程式已從 JAR 檔 com.ibm.mqjms.jar 載入 IBM MQ classes for JMS，追蹤會寫入稱為 mqjava_*%PID%.cl%u*.trc 的檔案。
- **V 9.0.0.10** 從 IBM MQ 9.0.0 Fix Pack 10，如果應用程式已從可重新定位 JAR 檔 com.ibm.mq.allclient.jar 載入 IBM MQ classes for JMS，則會將追蹤寫入稱為 mqjavaclient_*%PID%.cl%u*.trc 的檔案。

其中 *%PID%* 是所追蹤之應用程式的處理程序 ID，而 *%u* 是唯一數字，用來區分在不同 Java 類別載入器下執行追蹤之執行緒之間的檔案。

當追蹤檔停止時，應用程式會停止將資訊寫入追蹤檔。

如果在收集追蹤的問題發生之前，應用程式必須執行很長一段時間，則追蹤檔可能非常大。在此狀況下，請考量使用 IBM MQ classes for JMS 配置檔來收集追蹤 (請參閱第 80 頁的『使用 IBM MQ classes for JMS 配置檔來收集 IBM MQ classes for JMS 追蹤』)。以這種方式啟用追蹤時，可以控制 IBM MQ classes for JMS 產生的追蹤資料量。

使用 IBM MQ classes for JMS 配置檔來收集 IBM MQ classes for JMS 追蹤

如果應用程式必須在發生問題之前長時間執行，則應該使用 IBM MQ classes for JMS 配置檔來收集 IBM MQ classes for JMS 追蹤。配置檔可讓您指定各種選項來控制收集的追蹤資料量。

關於這項作業

若要使用 IBM MQ classes for JMS 配置檔來收集追蹤，請完成下列步驟。

程序

1. 建立 IBM MQ classes for JMS 配置檔。
如需此檔案的相關資訊，請參閱 [IBM MQ classes for JMS 配置檔](#)。
2. 編輯 IBM MQ classes for JMS 配置檔，讓 **com.ibm.msg.client.commonservices.trace.status** 內容設為值 ON。
3. 選擇性的: 編輯 IBM MQ classes for JMS 配置檔 Java Standard Edition 追蹤設定中列出的其他內容。
4. 使用下列指令來執行 IBM MQ classes for JMS 應用程式:

```
java -Dcom.ibm.msg.client.config.location=config_file_url  
application_name
```

其中 *config_file_url* 是統一資源定址器 (URL)，指定 IBM MQ classes for JMS 配置檔的名稱和位置。支援下列類型的 URL: http、file、ftp 和 jar。

以下是 Java 指令的範例:

```
java -Dcom.ibm.msg.client.config.location=file:/D:/mydir/myjms.config  
MyAppClass
```

這個指令會將 IBM MQ classes for JMS 配置檔識別為本端 Windows 系統上的 D:\mydir\myjms.config 檔案。

依預設，追蹤資訊會寫入應用程式現行工作目錄中的追蹤檔。追蹤檔的名稱視應用程式執行所在的環境而定:

- 若為 IBM MQ classes for JMS for IBM MQ 9.0.0 Fix Pack 1 或更早版本，追蹤會寫入稱為 mqjms_%PID%.trc 的檔案。
- **V 9.0.0.2** 從 IBM MQ 9.0.0 Fix Pack 2 開始，如果應用程式已從 JAR 檔 com.ibm.mqjms.jar 載入 IBM MQ classes for JMS，追蹤會寫入稱為 mqjava_%PID%.trc 的檔案。
- **V 9.0.0.2** 從 IBM MQ 9.0.0 Fix Pack 2，如果應用程式已從可重新定位 JAR 檔 com.ibm.mq.allclient.jar 載入 IBM MQ classes for JMS，則會將追蹤寫入稱為 mqjavaclient_%PID%.trc 的檔案。
- **V 9.0.0.10** 從 IBM MQ 9.0.0 Fix Pack 10 開始，如果應用程式已從 JAR 檔 com.ibm.mqjms.jar 載入 IBM MQ classes for JMS，追蹤會寫入稱為 mqjava_%PID%.cl%u.trc 的檔案。
- **V 9.0.0.10** 從 IBM MQ 9.0.0 Fix Pack 10，如果應用程式已從可重新定位 JAR 檔 com.ibm.mq.allclient.jar 載入 IBM MQ classes for JMS，則會將追蹤寫入稱為 mqjavaclient_%PID%.cl%u.trc 的檔案。

其中 %PID% 是所追蹤之應用程式的處理程序 ID，而 %u 是唯一數字，用來區分在不同 Java 類別載入器下執行追蹤之執行緒之間的檔案。

若要變更追蹤檔的名稱及其寫入位置，請確保應用程式使用的 IBM MQ classes for JMS 配置檔包含內容 **com.ibm.msg.client.commonservices.trace.outputName** 的項目。內容的值可以是下列其中一項:

- 在應用程式工作目錄中建立的追蹤檔名稱。
- 追蹤檔的完整名稱，包括在其中建立檔案的目錄。

例如，若要將 IBM MQ classes for JMS 配置成將應用程式的追蹤資訊寫入稱為 C:\Trace\trace.trc 的檔案，應用程式使用的 IBM MQ classes for JMS 配置檔必須包含下列項目：

```
com.ibm.msg.client.commonservices.trace.outputName=C:\Trace\trace.trc
```

使用 traceControl 公用程式來動態收集 IBM MQ classes for JMS 追蹤

IBM MQ classes for JMS 隨附的 traceControl 公用程式容許從執行中的應用程式收集追蹤。如果「IBM 支援中心」需要在發生問題時查看來自應用程式的追蹤，或需要從無法停止的重要應用程式收集追蹤，則這會非常有用。

關於這項作業

重要：只有 IBM Java runtime environments (JRE) 才支援此功能。

如需 traceControl 公用程式的相關資訊，請參閱第 89 頁的『使用 IBM MQ classes for Java 和 IBM MQ classes for JMS 來控制執行中處理程序中的追蹤』。

如果要使用 traceControl 公用程式來收集追蹤，請完成下列步驟。

程序

1. 啟動命令提示字元，並導覽至 MQ_INSTALLATION_PATH\java\lib 目錄。
2. 執行指令：

```
java -jar com.ibm.mq.traceControl.jar -list
```

此指令會啟動系統上所有 Java 處理程序的清單。

3. 識別需要追蹤之 IBM MQ classes for JMS 應用程式的處理程序 ID，並執行下列指令：

```
java -jar com.ibm.mq.traceControl.jar -i processidentifier -enable
```

現在已開啟應用程式的追蹤。

依預設，追蹤資訊會寫入應用程式現行工作目錄中的追蹤檔。追蹤檔的名稱視應用程式執行所在的環境而定：

- 若為 IBM MQ classes for JMS for IBM MQ 9.0.0 Fix Pack 1 或更早版本，追蹤會寫入稱為 mqjms_%PID%.trc 的檔案。
- **V 9.0.0.2** 從 IBM MQ 9.0.0 Fix Pack 2 開始，如果應用程式已從 JAR 檔 com.ibm.mqjms.jar 載入 IBM MQ classes for JMS，追蹤會寫入稱為 mqjava_%PID%.trc 的檔案。
- **V 9.0.0.2** 從 IBM MQ 9.0.0 Fix Pack 2，如果應用程式已從可重新定位 JAR 檔 com.ibm.mq.allclient.jar 載入 IBM MQ classes for JMS，則會將追蹤寫入稱為 mqjavaclient_%PID%.trc 的檔案。
- **V 9.0.0.10** 從 IBM MQ 9.0.0 Fix Pack 10 開始，如果應用程式已從 JAR 檔 com.ibm.mqjms.jar 載入 IBM MQ classes for JMS，追蹤會寫入稱為 mqjava_%PID%.cl%u.trc 的檔案。
- **V 9.0.0.10** 從 IBM MQ 9.0.0 Fix Pack 10，如果應用程式已從可重新定位 JAR 檔 com.ibm.mq.allclient.jar 載入 IBM MQ classes for JMS，則會將追蹤寫入稱為 mqjavaclient_%PID%.cl%u.trc 的檔案。

其中 %PID% 是所追蹤之應用程式的處理程序 ID，而 %u 是唯一數字，用來區分在不同 Java 類別載入器下執行追蹤之執行緒之間的檔案。

4. 若要關閉追蹤，請執行下列指令：

```
java -jar com.ibm.mq.traceControl.jar -i processidentifier -disable
```


追蹤 IBM MQ classes for Java 應用程式

提供 IBM MQ classes for Java 中的追蹤功能，以協助「IBM 支援中心」診斷客戶問題。各種內容會控制此機能的行為。

關於這項作業

如果您要求您提供追蹤輸出來調查問題，請使用下列其中一個選項：

- 如果問題易於重建，請使用 Java 系統內容來收集 IBM MQ classes for Java 追蹤。如需相關資訊，請參閱第 83 頁的『使用 Java 系統內容來收集 IBM MQ classes for Java 追蹤』。
- 如果應用程式需要在發生問題之前執行一段時間，請使用 IBM MQ classes for Java 配置檔來收集 IBM MQ classes for Java 追蹤。如需相關資訊，請參閱第 83 頁的『使用 IBM MQ classes for Java 配置檔來收集 IBM MQ classes for Java 追蹤』。
- 如果要從目前執行中的應用程式產生追蹤，請使用 traceControl 公用程式來動態收集 IBM MQ classes for Java 追蹤。如需相關資訊，請參閱第 84 頁的『使用 traceControl 公用程式來動態收集 IBM MQ classes for Java 追蹤』。

如果您不確定要使用哪個選項，請聯絡「IBM 支援中心」代表，他們將能夠建議您針對您所看到的問題收集追蹤的最佳方式。

如果發生嚴重或無法復原的錯誤，First Failure Support Technology (FFST) 資訊會記錄在名稱格式為 JAVACC *xxxx*.FDC 的檔案中，其中 *xxxx* 是四位數的數字。它會遞增以區分 .FDC 檔案。

.FDC 檔案一律會寫入稱為 FFDC 的子目錄。子目錄位於下列兩個位置之一，視追蹤是否作用中而定：

追蹤作用中，且已設定 *traceOutputName*

FFDC 目錄會建立為要寫入追蹤檔之目錄的子目錄。

追蹤不在作用中或未設定 *traceOutputName*

FFDC 目錄會建立為現行工作目錄的子目錄。

JSE 共用服務使用 `java.util.logging` 作為其追蹤及記載基礎架構。此基礎架構的根物件是 `LogManager`。日誌管理程式有一個 `reset` 方法，它會關閉所有處理程式，並將記載層次設為 `null`，實際上會關閉所有追蹤。如果您的應用程式或應用程式伺服器呼叫

`java.util.logging.LogManager.getLogManager().reset()`，它會關閉所有追蹤，這可能會讓您無法診斷任何問題。若要避免關閉所有追蹤，請建立 `LogManager` 類別，且其具有未執行任何動作的被置換 `reset()` 方法，如下列範例中所示：

```
package com.ibm.javaut.tests;
import java.util.logging.LogManager;
public class JmsLogManager extends LogManager {
    // final shutdown hook to ensure that the trace is finally shutdown
    // and that the lock file is cleaned-up
    public class ShutdownHook extends Thread{
        public void run(){
            doReset();
        }
    }
    public JmsLogManager(){
        // add shutdown hook to ensure final cleanup
        Runtime.getRuntime().addShutdownHook(new ShutdownHook());
    }
    public void reset() throws SecurityException {
        // does nothing
    }
    public void doReset(){
        super.reset();
    }
}
```

需要關閉連結鉤，以確保在 JVM 完成時適當地關閉追蹤。若要使用已修改的日誌管理程式而非預設日誌管理程式，請將系統內容新增至 JVM 啟動：

```
java -Djava.util.logging.manager=com.mycompany.logging.LogManager ...
```

使用 Java 系統內容來收集 IBM MQ classes for Java 追蹤

對於可在短時間內重新產生的問題，應該在啟動應用程式時設定 Java 系統內容來收集 IBM MQ classes for Java 追蹤。

關於這項作業

若要使用 Java 系統內容來收集追蹤，請完成下列步驟。

程序

- 使用下列指令來執行要追蹤的應用程式：

```
java -Dcom.ibm.msg.client.commonservices.trace.status=ON application_name
```

依預設，追蹤資訊會寫入應用程式現行工作目錄中的追蹤檔。追蹤檔的名稱視應用程式執行所在的環境而定：

- 若為 IBM MQ classes for Java for IBM MQ 9.0.0 Fix Pack 1 或更早版本，追蹤會寫入稱為 `mqjms_%PID%.trc` 的檔案。
- **V 9.0.0.2** 從 IBM MQ 9.0.0 Fix Pack 2 開始，如果應用程式已從 JAR 檔 `com.ibm.mq.jar` 載入 IBM MQ classes for Java，追蹤會寫入稱為 `mqjava_%PID%.trc` 的檔案。
- **V 9.0.0.2** 從 IBM MQ 9.0.0 Fix Pack 2，如果應用程式已從可重新定位 JAR 檔 `com.ibm.mq.allclient.jar` 載入 IBM MQ classes for Java，則會將追蹤寫入稱為 `mqjavaclient_%PID%.trc` 的檔案。
- **V 9.0.0.10** 從 IBM MQ 9.0.0 Fix Pack 10 開始，如果應用程式已從 JAR 檔 `com.ibm.mq.jar` 載入 IBM MQ classes for Java，追蹤會寫入稱為 `mqjava_%PID%.cl%u.trc` 的檔案。
- **V 9.0.0.10** 從 IBM MQ 9.0.0 Fix Pack 10，如果應用程式已從可重新定位 JAR 檔 `com.ibm.mq.allclient.jar` 載入 IBM MQ classes for Java，則會將追蹤寫入稱為 `mqjavaclient_%PID%.cl%u.trc` 的檔案。

其中 `%PID%` 是所追蹤之應用程式的處理程序 ID，而 `%u` 是唯一數字，用來區分在不同 Java 類別載入器下執行追蹤之執行緒之間的檔案。

當追蹤檔停止時，應用程式會停止將資訊寫入追蹤檔。

如果在收集追蹤的問題發生之前，應用程式必須執行很長一段時間，則追蹤檔可能非常大。在此狀況下，請考量使用 IBM MQ classes for Java 配置檔來收集追蹤 (請參閱第 83 頁的『使用 IBM MQ classes for Java 配置檔來收集 IBM MQ classes for Java 追蹤』)。以這種方式啟用追蹤時，可以控制 IBM MQ classes for Java 產生的追蹤資料量。

使用 IBM MQ classes for Java 配置檔來收集 IBM MQ classes for Java 追蹤

如果應用程式必須在發生問題之前長時間執行，則應該使用 IBM MQ classes for Java 配置檔來收集 IBM MQ classes for Java 追蹤。配置檔可讓您指定各種選項來控制收集的追蹤資料量。

關於這項作業

若要使用 IBM MQ classes for Java 配置檔來收集追蹤，請完成下列步驟。

程序

1. 建立 IBM MQ classes for Java 配置檔。
如需此檔案的相關資訊，請參閱 [IBM MQ classes for Java 配置檔](#)。
2. 編輯 IBM MQ classes for Java 配置檔，讓 **`com.ibm.msg.client.commonservices.trace.status`** 內容設為值 `ON`。
3. 選擇性的：編輯 IBM MQ classes for Java 配置檔 Java Standard Edition 追蹤設定中列出的其他內容。

4. 使用下列指令來執行 IBM MQ classes for Java 應用程式:

```
java -Dcom.ibm.msg.client.config.location=config_file_url  
application_name
```

其中 *config_file_url* 是統一資源定址器 (URL)，指定 IBM MQ classes for Java 配置檔的名稱和位置。支援下列類型的 URL: http、file、ftp 和 jar。

以下是 Java 指令的範例:

```
java -Dcom.ibm.msg.client.config.location=file:/D:/mydir/myJava.config  
MyAppClass
```

這個指令會將 IBM MQ classes for Java 配置檔識別為本端 Windows 系統上的 D:\mydir\myJava.config 檔案。

依預設，追蹤資訊會寫入應用程式現行工作目錄中的追蹤檔。追蹤檔的名稱視應用程式執行所在的環境而定:

- 若為 IBM MQ classes for Java for IBM MQ 9.0.0 Fix Pack 1 或更早版本，追蹤會寫入稱為 mqjms_*%PID%*.trc 的檔案。
- **V 9.0.0.2** 從 IBM MQ 9.0.0 Fix Pack 2 開始，如果應用程式已從 JAR 檔 com.ibm.mq.jar 載入 IBM MQ classes for Java，追蹤會寫入稱為 mqjava_*%PID%*.trc 的檔案。
- **V 9.0.0.2** 從 IBM MQ 9.0.0 Fix Pack 2，如果應用程式已從可重新定位 JAR 檔 com.ibm.mq.allclient.jar 載入 IBM MQ classes for Java，則會將追蹤寫入稱為 mqjavaclient_*%PID%*.trc 的檔案。
- **V 9.0.0.10** 從 IBM MQ 9.0.0 Fix Pack 10 開始，如果應用程式已從 JAR 檔 com.ibm.mq.jar 載入 IBM MQ classes for Java，追蹤會寫入稱為 mqjava_*%PID%.cl%u*.trc 的檔案。
- **V 9.0.0.10** 從 IBM MQ 9.0.0 Fix Pack 10，如果應用程式已從可重新定位 JAR 檔 com.ibm.mq.allclient.jar 載入 IBM MQ classes for Java，則會將追蹤寫入稱為 mqjavaclient_*%PID%.cl%u*.trc 的檔案。

其中 *%PID%* 是所追蹤之應用程式的處理程序 ID，而 *%u* 是唯一數字，用來區分在不同 Java 類別載入器下執行追蹤之執行緒之間的檔案。

若要變更追蹤檔的名稱及其寫入位置，請確保應用程式使用的 IBM MQ classes for Java 配置檔包含內容 **com.ibm.msg.client.commonservices.trace.outputName** 的項目。內容的值可以是下列其中一項:

- 在應用程式工作目錄中建立的追蹤檔名稱。
- 追蹤檔的完整名稱，包括在其中建立檔案的目錄。

例如，若要將 IBM MQ classes for Java 配置成將應用程式的追蹤資訊寫入稱為 C:\Trace\trace.trc 的檔案，應用程式使用的 IBM MQ classes for Java 配置檔必須包含下列項目:

```
com.ibm.msg.client.commonservices.trace.outputName=C:\Trace\trace.trc
```

使用 traceControl 公用程式來動態收集 IBM MQ classes for Java 追蹤

IBM MQ classes for Java 隨附的 traceControl 公用程式容許從執行中的應用程式收集追蹤。如果「IBM 支援中心」需要在發生問題時查看來自應用程式的追蹤，或需要從無法停止的重要應用程式收集追蹤，則這會非常有用。

關於這項作業

如需 traceControl 公用程式的相關資訊，請參閱第 89 頁的『[使用 IBM MQ classes for Java 和 IBM MQ classes for JMS 來控制執行中處理程序中的追蹤](#)』。

如果要使用 traceControl 公用程式來收集追蹤，請完成下列步驟。

程序

1. 啟動命令提示字元，並導覽至 `MQ_INSTALLATION_PATH\java\lib` 目錄。
2. 執行指令：

```
java -jar com.ibm.mq.traceControl.jar ...
```

此指令會啟動系統上所有 Java 處理程序的清單。

3. 識別需要追蹤之 IBM MQ classes for Java 應用程式的處理程序 ID，並執行下列指令：

```
java -jar com.ibm.mq.traceControl -i process identifier -enable
```

現在已開啟應用程式的追蹤。

依預設，追蹤資訊會寫入應用程式現行工作目錄中的追蹤檔。追蹤檔的名稱視應用程式執行所在的環境而定：

- 若為 IBM MQ classes for Java for IBM MQ 9.0.0 Fix Pack 1 或更早版本，追蹤會寫入稱為 `mqjms_%PID%.trc` 的檔案。
- **V 9.0.0.2** 從 IBM MQ 9.0.0 Fix Pack 2 開始，如果應用程式已從 JAR 檔 `com.ibm.mq.jar` 載入 IBM MQ classes for Java，追蹤會寫入稱為 `mqjava_%PID%.trc` 的檔案。
- **V 9.0.0.2** 從 IBM MQ 9.0.0 Fix Pack 2，如果應用程式已從可重新定位 JAR 檔 `com.ibm.mq.allclient.jar` 載入 IBM MQ classes for Java，則會將追蹤寫入稱為 `mqjavaclient_%PID%.trc` 的檔案。
- **V 9.0.0.10** 從 IBM MQ 9.0.0 Fix Pack 10 開始，如果應用程式已從 JAR 檔 `com.ibm.mq.jar` 載入 IBM MQ classes for Java，追蹤會寫入稱為 `mqjava_%PID%.cl%u.trc` 的檔案。
- **V 9.0.0.10** 從 IBM MQ 9.0.0 Fix Pack 10，如果應用程式已從可重新定位 JAR 檔 `com.ibm.mq.allclient.jar` 載入 IBM MQ classes for Java，則會將追蹤寫入稱為 `mqjavaclient_%PID%.cl%u.trc` 的檔案。

其中 `%PID%` 是所追蹤之應用程式的處理程序 ID，而 `%u` 是唯一數字，用來區分在不同 Java 類別載入器下執行追蹤之執行緒之間的檔案。

4. 若要關閉追蹤，請執行下列指令：

```
java -jar com.ibm.mq.traceControl -i process identifier -disable
```

追蹤 IBM MQ 資源配接器

ResourceAdapter 物件封裝 IBM MQ 資源配接器的廣域內容。如果要啟用 IBM MQ 資源配接器的追蹤，必須在 ResourceAdapter 物件中定義內容。

ResourceAdapter 物件有兩組內容：

- 與診斷追蹤相關聯的內容
- 與資源配接器所管理的連線儲存區相關聯的內容

您定義這些內容的方式取決於應用程式伺服器所提供的管理介面。

第 85 頁的表 8 列出與診斷追蹤相關聯的 ResourceAdapter 物件的內容。

內容名稱	類型	預設值	說明
traceEnabled	字串	false	啟用或停用診斷追蹤的旗標。如果值為 false，則會關閉追蹤。

表 8: 與診斷追蹤相關聯的 ResourceAdapter 物件內容 (繼續)

內容名稱	類型	預設值	說明
traceLevel	字串	3	診斷追蹤中的詳細程度。此值可以在範圍 0 (不產生追蹤) 到 10 (提供最詳細的資料) 之間。如需每一個層次的說明，請參閱 第 86 頁的表 9。如果啟用追蹤，除非 IBM 支援中心另有指定，否則 traceLevel 應該設為值 10。
logWriter 已啟用	字串	true	這個旗標用來啟用或停用將診斷追蹤傳送至應用程式伺服器所提供的 LogWriter 物件。如果值為 true，則追蹤會傳送至 LogWriter 物件。如果值為 false，則不會使用應用程式伺服器提供的任何 LogWriter 物件。

第 86 頁的表 9 說明診斷追蹤的詳細程度。

表 9: 診斷追蹤的詳細程度

層次號碼	詳細程度
0	無追蹤。
1	追蹤包含錯誤訊息。
3	追蹤包含錯誤和警告訊息。
6	追蹤包含錯誤、警告及參考訊息。
8	追蹤包含錯誤、警告和參考訊息，以及方法的進入和結束資訊。
9	追蹤包含錯誤、警告及參考訊息、方法的進入及結束資訊，以及診斷資料。
10	追蹤包含所有追蹤資訊。

註: 此表格中未包含的任何層次都相當於下一個最低層次。例如，指定追蹤層次 4 相當於指定追蹤層次 3。不過，IBM MQ 資源配接器的未來版本可能會使用未包含的層次，因此最好避免使用這些層次。

如果關閉診斷追蹤，則會將錯誤及警告訊息寫入系統錯誤串流。如果開啟診斷追蹤，則會將錯誤訊息寫入系統錯誤串流及追蹤目的地，但只會將警告訊息寫入追蹤目的地。不過，只有在追蹤層次為 3 或更高時，追蹤才會包含警告訊息。依預設，追蹤目的地是現行工作目錄，但如果設定 logWriterEnabled 內容，則會將追蹤傳送至應用程式伺服器。

一般而言，ResourceAdapter 物件不需要管理。不過，例如，若要在 UNIX and Linux 系統上啟用診斷追蹤，您可以設定下列內容：

```
traceEnabled: true
traceLevel: 10
```

如果資源配接器尚未啟動 (例如，當使用 IBM MQ 資源的應用程式只在用戶端儲存器中執行時)，則這些內容沒有作用。在此狀況下，您可以將診斷追蹤的內容設為 Java virtual machine (JVM) 系統內容。您可以在 **java** 指令上使用 **-D** 旗標來設定內容，如下列範例所示：

```
java ... -DtraceEnabled=true -DtraceLevel=10
```

提示及要訣

您不需要定義 ResourceAdapter 物件的所有內容。任何未指定的內容都會採用其預設值。

在受管理環境中，最好不要混合兩種指定內容的方式。如果您混合使用它們，JVM 系統內容的優先順序會高於 ResourceAdapter 物件的內容。

當 WebSphere Application Server traditional 9.0 與 IBM MQ 9.0 資源配接器搭配使用時，由於 Java EE 相依關係注入現在是一般 Java EE 參照範例，因此應該更新標準追蹤字串來包含 `com.ibm.ws.cdi.jms*=all`。這表示完整字串為：

```
*=info:jmsApi=all:Messaging=all:com.ibm.mq.*=all:JMSApi=all:com.ibm.ws.cdi.jms*=all
```

如需搭配使用追蹤與 WebSphere Application Server traditional 的相關資訊，請參閱 Technote [Enabling Java Message Service \(JMS\) trace for WebSphere Application Server](#)。

追蹤其他 IBM MQ Java 元件

對於 IBM MQ 的 Java 元件 (例如 IBM MQ Explorer 及 IBM MQ Transport for SOAP 的 Java 實作)，診斷資訊是使用標準 IBM MQ 診斷機能或由 Java 診斷類別來輸出。

此環境定義中的診斷資訊包含追蹤、首次失敗資料擷取 (FFDC) 及錯誤訊息。

您可以視情況選擇使用 IBM MQ 機能或 IBM MQ classes for Java 或 IBM MQ classes for JMS 的機能來產生此資訊。如果 IBM MQ 診斷機能在本端系統上可用，通常會使用這些診斷機能。

在下列情況下，您可能想要使用 Java 診斷程式：

- 在可使用佇列管理程式的系統上，如果佇列管理程式與您執行的軟體分開管理的話。
- 減少 IBM MQ 追蹤的效能影響。

若要要求及配置診斷輸出，在啟動 IBM MQ Java 程序時，會使用兩個系統內容：

- System property `com.ibm.mq.commonservices` 指定標準 Java 內容檔，其中包含用來配置診斷輸出的數行。檔案中的每一行程式碼都是任意格式，並以換行字元終止。
- System property `com.ibm.mq.commonservices.diagid` 會將追蹤和 FFDC 檔案與建立它們的處理程序相關聯。

如需使用 `com.ibm.mq.commonservices` 內容檔來配置診斷資訊的相關資訊，請參閱第 87 頁的『[使用 com.ibm.mq.commonservices](#)』。

如需尋找追蹤資訊和 FFDC 檔案的指示，請參閱第 88 頁的『[Java 追蹤和 FFDC 檔案](#)』。

相關概念

第 58 頁的『[在 UNIX and Linux 系統上使用追蹤](#)』

使用 `strmqtrc` 和 `endmqtrc` 指令來啟動和結束追蹤，以及使用 `dspmqtrc` 來顯示追蹤檔

第 60 頁的『[在 IBM i 上搭配使用追蹤與 IBM MQ 伺服器](#)』

使用 `TRCMQM` 指令來啟動和停止追蹤，並指定您需要的追蹤類型。

第 65 頁的『[在 z/OS 上使用追蹤來判斷問題](#)』

有不同的追蹤選項可用於 IBM MQ 的問題判斷。請利用這個主題來瞭解不同的選項，以及如何控制追蹤。

第 77 頁的『[追蹤 TLS: runmqakm、strmqikm 及 runmqckm 函數](#)』

如何追蹤「傳輸層安全 (TLS)」，並要求 `runmqakm` 追蹤及 `strmqikm` (iKeyman) 和 `runmqckm` (iKeycmd) 追蹤。

相關參考

第 56 頁的『[在 Windows 上使用追蹤](#)』

使用 `strmqtrc` 和 `endmqtrc` 指令或 IBM MQ Explorer 介面來啟動和結束追蹤。

使用 com.ibm.mq.commonservices

`com.ibm.mq.commonservices` 內容檔包含下列與從 IBM MQ 的 Java 元件輸出診斷程式相關的項目。

請注意，在所有這些項目中，大小寫是很重要的：

診斷程式。Java= 選項

使用 Java 追蹤來追蹤哪些元件。選項是一或多個 `explorer`、`soap` 及 `wmqjavaclasses` (以逗點區隔)，其中 "explorer" 是指來自 IBM MQ Explorer 的診斷程式，"soap" 是指來自 IBM MQ Transport for SOAP 內執行中處理程序的診斷程式，而 "wmqjavaclasses" 是指來自基礎 IBM MQ Java 類別的診斷程式。依預設，不會追蹤任何元件。

診斷。Java.Trace.Detail= high/medium/low

Java 追蹤的詳細層次。高及中詳細資料層次符合 IBM MQ 追蹤中使用的那些層次，但低是 Java 追蹤特有的。如果診斷程式，則會忽略此內容。未設定 Java。預設值為 *medium*。

診斷程式。Java.Trace.Destination.File= enabled/disabled

是否將 Java 追蹤寫入檔案。如果診斷程式，則會忽略此內容。未設定 Java。預設值為 *disabled*。

診斷。Java.Trace.Destination.Console= enabled/disabled

是否將 Java 追蹤寫入系統主控台。如果診斷程式，則會忽略此內容。未設定 Java。預設值為 *disabled*。

診斷程式。Java.Trace.Destination.Pathname= dirname

寫入 Java 追蹤的目錄。如果診斷程式，則會忽略此內容。Java 未設定或診斷。

Java.Trace.Destination.File= 已停用。在 UNIX and Linux 系統上，如果它存在，則預設值為 */var/mqm/trace*，否則為 Java 主控台 (System.err)。在 Windows 上，預設值是系統主控台。

診斷。Java.FFDC.Destination.Pathname= dirname

寫入 Java FFDC 輸出的目錄。預設值是現行工作目錄。

診斷程式。Java.Errors.Destination.Filename= filename

寫入 Java 錯誤訊息的完整檔名。預設值為 *AMQJAVA.LOG*。

第 88 頁的圖 6 中提供 *com.ibm.mq.commonservices* 內容檔的範例。以 # 記號開頭的行會被視為註解。

```
#
# Java diagnostics for IBM MQ Transport for SOAP
# and the IBM MQ Java Classes are both enabled
#
Diagnostics.Java=soap,wmqjavaclasses
#
# High detail Java trace
#
Diagnostics.Java.Trace.Detail=high
#
# Java trace is written to a file and not to the console.
#
Diagnostics.Java.Trace.Destination.File=enabled
Diagnostics.Java.Trace.Destination.Console=disabled
#
# Directory for Java trace file
#
Diagnostics.Java.Trace.Destination.Pathname=c:\\tracedir
#
# Directory for First Failure Data Capture
#
Diagnostics.Java.FFDC.Destination.Pathname=c:\\ffdcdir
#
# Directory for error logging
#
Diagnostics.Java.Errors.Destination.Filename=c:\\errorsdir\\SOAPERRORS.LOG
#
```

圖 6: 範例 *com.ibm.mq.commonservices* 內容檔

也提供範例內容檔 *WMQSoap_RAS.properties* 作為 " Java 傳訊和 SOAP 傳輸安裝選項。

Java 追蹤和 FFDC 檔案

Java 追蹤和 FFDC 檔案的檔名慣例。

當針對 IBM MQ Transport for SOAP 產生 Java 追蹤時，會將它寫入名稱格式為 *AMQ.diagid.counter*.TRC 的檔案。這裡，*diagid* 是與此 Java 處理程序相關聯的系統內容 *com.ibm.mq.commonservices.diagid* 的值 (如本節先前所述)，而 *counter* 是大於或等於 0 的整數。名稱中的所有字母都是大寫，符合一般 IBM MQ 追蹤所用的命名慣例。

如果未指定 *com.ibm.mq.commonservices.diagid*，則 *diagid* 的值是現行時間，格式為 *YYYYMMDDhhmmssmm*。

當針對 IBM MQ Explorer 產生 Java 追蹤時，會將它寫入名稱格式為 AMQYYYYMMDDHHmmssmmm.TRC.n 的檔案中。每次執行「IBM MQ Explorer」追蹤時，追蹤機能將所有先前的追蹤檔案重新命名，將檔案字尾 .n 加一。然後追蹤機能會建立新的檔案，其字尾為 .0，這一律是最新的檔案。

IBM MQ Java 類別追蹤檔具有以對等 IBM MQ SOAP 傳輸 Java 追蹤檔為基礎的名稱。名稱的不同之處在於它在 .TRC 字串之前新增了字串 .JC，其格式為 AMQ. *diagid. counter*.JC.TRC。

為 IBM MQ Explorer 或 IBM MQ Transport for SOAP 產生 Java FFDC 時，會將它寫入名稱格式為 AMQ. *diagid. counter*.FDC 的檔案，其中 *diagid* 及計數器如 Java 追蹤檔所述。

IBM MQ Explorer 及 IBM MQ Transport for SOAP 的 Java 錯誤訊息輸出會寫入適當 Java 處理程序的 *Diagnostics.Java.Errors.Destination.FileName* 所指定的檔案。這些檔案的格式與標準 IBM MQ 錯誤日誌的格式非常相符。

當處理程序將追蹤資訊寫入檔案時，它會在處理程序的生命期限內附加至單一追蹤輸出檔。同樣地，單一 FFDC 輸出檔用於處理程序的生命期限。

所有追蹤輸出都採用 UTF-8 字集。

使用 IBM MQ classes for Java 和 IBM MQ classes for JMS 來控制執行中處理程序中的追蹤

IBM MQ classes for Java 和 IBM MQ classes for JMS 登錄「標準 MBean」，可讓適當的「Java 管理延伸 (JMX)」工具控制用戶端程序追蹤行為的特定層面。

原則

作為常用通用工具 (例如 `jconsole`) 的替代方案，您可以使用可執行 JAR 檔形式的指令行工具來存取這些機能。

JAR 檔稱為 `com.ibm.mq.traceControl.jar`，儲存在 IBM MQ 安裝架構的 `java/lib` 子目錄中 (請參閱針對 IBM MQ classes for JMS 所安裝的內容及 IBM MQ classes for Java 的安裝目錄)。

註：視配置而定，JMX 工具可以在本端 (與程序相同的系統上) 或遠端使用。最初會討論當地案例。

尋找程序

若要控制程序，您必須建立 JMX 連線。若要在本端控制處理程序，您必須指定其 ID。

若要顯示執行中 Java 處理程序及其 ID 的摘要，請使用選項 `-list` 執行 JAR 執行檔。此選項會產生所找到處理程序的 ID 及說明清單。

檢查追蹤狀態

當您找到相關處理程序的 ID 時，請使用選項 `-i identifier -status` 執行 JAR 執行檔，其中 *identifier* 是您要變更之處理程序的 ID。這些選項會顯示處理程序的狀態 (`enabled` 或 `disabled`)，以及處理程序執行所在的相關資訊、追蹤檔名稱，以及代表追蹤中包含及排除套件的樹狀結構。

啟用及停用追蹤

如果要啟用處理程序的追蹤，請以 `-i identifier -enable` 選項來執行 JAR 執行檔。

如果要停用處理程序的追蹤，請使用選項 `-i identifier -disable` 來執行 JAR 執行檔。

註：您只能從 `-status`，`-enable` 及 `-disable` 集中選擇一個選項。

併入及排除套件

如果要在處理程序的追蹤中併入套件，請執行 JAR 執行檔並指定選項 `-i identifier -ip package_name`，其中 *package_name* 是套件的名稱。

如果要從程序的追蹤中排除套件，請使用 `-i identifier -ep package_name` 選項來執行 JAR 執行檔。

註: 您可以使用多個 `-ip` 和 `-ep` 選項。不會檢查這些選項的一致性。

當您指定要排除或併入的套件時，不會影響具有相符字首的套件處理。例如，從追蹤中排除套件 `com.ibm.mq.jms` 不會排除 `com.ibm.mq`、`com.ibm.msq.client.jms` 或 `com.ibm.mq.remote.api`，但會排除 `com.ibm.mq.jms.internal`。

```
C:>java -jar MQ_INSTALLATION_PATH/java/lib/com.ibm.mq.traceControl.jar -list
10008 : 'MQSample'
9004 : ' MQ_INSTALLATION_PATH/java/lib/com.ibm.mq.traceControl.jar -list'

C:>java -jar MQ_INSTALLATION_PATH/java/lib/com.ibm.mq.traceControl.jar -i 10008 -status
Tracing enabled : false
User Directory : C:\Users\IBM_ADMIN\RTCworkspace\sandpit
Trace File Name : mqjms.trc
Package Include/Exclude tree
root - Included

C:>java -jar MQ_INSTALLATION_PATH/java/lib/com.ibm.mq.traceControl.jar -i 10008 -enable
Enabling trace
Tracing enabled : true

C:>java -jar MQ_INSTALLATION_PATH/java/lib/com.ibm.mq.traceControl.jar -i 10008 -status
Tracing enabled : true
User Directory : C:\Users\IBM_ADMIN\RTCworkspace\sandpit
Trace File Name : mqjms_10008.cl0.trc
Package Include/Exclude tree
root - Included

C:>java -jar MQ_INSTALLATION_PATH/java/lib/com.ibm.mq.traceControl.jar -i 10008 -ip
com.ibm.mq.jms
Adding 'com.ibm.mq.jms' to the list of packages included in trace

C:>java -jar MQ_INSTALLATION_PATH/java/lib/com.ibm.mq.traceControl.jar -i 10008 -status
Tracing enabled : true
User Directory : C:\Users\IBM_ADMIN\RTCworkspace\sandpit
Trace File Name : mqjms_10008.cl0.trc
Package Include/Exclude tree
root - Included
com - Included
ibm - Included
mq - Included
jms - Included

C:>java -jar MQ_INSTALLATION_PATH/java/lib/com.ibm.mq.traceControl.jar -i 10008 -ip
com.acme.banana -ep com.acme.banana.split -ip com.acme.banana.shake
Adding 'com.acme.banana' to the list of packages included in trace
Adding 'com.acme.banana.shake' to the list of packages included in trace
Adding 'com.acme.banana.split' to the list of packages excluded from trace

C:>java -jar MQ_INSTALLATION_PATH/java/lib/com.ibm.mq.traceControl.jar -i 10008 -status
Tracing enabled : true User Directory : C:\Users\IBM_ADMIN\RTCworkspace\sandpit
Trace File Name : mqjms_10008.cl0.trc
Package Include/Exclude tree
root - Included
com - Included
acme - Included
banana - Included
shake - Included
split - Excluded
ibm - Included
mq - Included
jms - Included
```

套件併入-排除樹狀結構

IBM MQ classes for Java 和 IBM MQ classes for JMS 的追蹤機制會從根節點開始，透過樹狀結構來追蹤套件的併入和排除。在樹狀結構中，每一個節點代表套件名稱的一個元素，由套件名稱元素識別且包含追蹤狀態，追蹤狀態可以是已併入或已排除。例如，套件 `com.ibm.mq` 將由字串 `com`、`ibm` 和 `mq` 所識別的三個節點代表。

一開始，樹狀結構通常包含包含大部分套件的項目，但會排除標頭及 pcf 套件，因為它們會產生許多雜訊。所以初始樹會看起來像這樣

```
root - Included
com - Included
ibm - Included
mq - Included
headers - Excluded
pcf - Excluded
```

當追蹤機能決定要併入或排除套件時，它會盡可能將套件名稱的前導部分與樹狀結構中的節點比對，並取得最後相符節點的狀態。在樹狀結構的起始狀態時，會包含套件 `com.ibm.msg.client` 和 `com.ibm.mq.jms`，因為樹狀結構中符合它們的最後一個節點（分別為 `com-> ibm` 和 `com-> ibm-> mq`）會標示為已包含。相反地，會排除套件 `com.ibm.headers.internal`，因為樹狀結構（`com-> ibm-> mq->` 標頭）中的最後一個相符節點會標示為已排除。

使用 `com.ibm.mq.TraceControl.jar` 對樹狀結構進行進一步變更時，請務必記住，併入或排除只會影響套件及子套件。因此，假設先前顯示的起始狀態是指定 `-ep com.ibm.mq.jms`，則會更新樹狀結構，使其看起來如下：

```
root - Included
com - Included
ibm - Included
mq - Included
headers - Excluded
jms - Excluded
pcf - Excluded
```

這項更新會排除套件 `com.ibm.mq.jms` 和 `com.ibm.mq.jms.internal`，而不會影響 `com.ibm.mq.jms.*` 階層之外的套件。

如果接下來指定 `-ip com.ibm.mq.jms.admin`，則樹狀結構看起來如下：

```
root - Included
com - Included
ibm - Included
mq - Included
headers - Excluded
jms - Excluded
admin - Included
pcf - Excluded
```

這項更新仍會排除套件 `com.ibm.mq.jms`、`com.ibm.mq.jms.internal`，但現在會將套件 `com.ibm.mq.jms.admin` 和 `com.ibm.mq.jms.admin.internal` 併入追蹤中。

從遠端連接

只有在啟動處理程序時所使用的 JMX 代理程式已啟用遠端連線，且使用 `-Dcom.sun.management.jmxremote.port=port_number` 系統設定時，您才能從遠端連接。

開始使用此系統設定之後，您可以使用選項 `-h host_name -p port_number` 來執行 JAR 執行檔，以取代 `-i identifier` 選項，其中 `host_name` 是您要連接的主機名稱，而 `port_number` 是要使用的埠名稱。

註：您必須確保透過對連線啟用 TLS 來採取適當的步驟，以將安全風險降到最低。如需 JMX 的進一步詳細資料，請參閱 Oracle 說明文件 <https://www.oracle.com>。

限制

下列限制存在：

- 若為非 IBM JVM，工具必須以新增至其類別路徑的 `tools.jar` 來啟動。這些平台上的指令為：

```
java -cp MQ_INSTALL_DIR/java/lib/com.ibm.mq.traceControl.jar;JAVA_HOME/lib/tools.jar
com.ibm.msg.client.commonservices.trace.TraceController
```

- 本端連接由使用者 ID 控制。工具必須在與要控制的處理程序相同的 ID 下執行。

追蹤 IBM MQ .NET 應用程式

在 IBM MQ .NET 中，您可以使用 MQI 來啟動及控制追蹤機能，如同在 IBM MQ 程式中一樣。

不過，strmqtrc 指令的 -i 和 -p 參數 (可讓您指定處理程序和執行緒 ID 以及具名處理程序) 沒有作用。

通常只有在 IBM 服務要求時，您才需要使用追蹤機能。

如需追蹤指令的相關資訊，請參閱 [在 Windows 上使用追蹤](#)。

Windows V 9.0.1 z/OS Linux AIX 追蹤 IBM MQ Console 和 REST API

提供 IBM MQ Console 和 REST API 中的追蹤機能，以協助 IBM 人員診斷客戶問題。各種內容會控制這些機能的行為。

IBM MQ Console 和 REST API 由三個功能範圍組成，各有自己的追蹤機制：

- [在瀏覽器中執行的 IBM MQ Console JavaScript 程式碼](#)。
- [在 mqweb 伺服器中執行的 IBM MQ Console 及 REST API 程式碼](#)。
- [在 mqweb 伺服器中執行的 IBM MQ JMS 程式碼類別](#)。

對在瀏覽器中執行的 IBM MQ Console JavaScript 程式碼啟用追蹤

此追蹤只會從啟用它的瀏覽器輸出。在您登出 IBM MQ Console 之後，會自動停用追蹤。

如果要對在瀏覽器中執行的 IBM MQ Console JavaScript 程式碼啟用追蹤，請執行下列動作：

1. 登入 IBM MQ Console

2. 按一下儀表板功能表  圖示，然後選取 **診斷程式**。



3. 針對 IBM MQ Console 瀏覽器追蹤選取 **啟用**，然後按一下 **確定**。

4. 遵循概述的步驟，以針對在 mqweb 伺服器中執行的 IBM MQ Console 及 REST API 程式碼啟用追蹤。

然後會開始追蹤在瀏覽器中執行的動作。此追蹤會定期傳送至在 mqweb 伺服器中執行的 IBM MQ Console 程式碼，並在 mqweb 伺服器追蹤日誌中輸出。

對在 mqweb 伺服器中執行的 IBM MQ Console 及 REST API 程式碼啟用追蹤

1. 在下列其中一個目錄中找到 mqwebuser.xml 檔案：

-  `MQ_DATA_DIRECTORY/web/installations/
installationName/servers/mqweb`
-  `WLP_user_directory/servers/mqweb`

其中 `WLP_user_directory` 是執行 `crtmqweb.sh` Script 以建立 mqweb 伺服器定義時指定的目錄。

2. 在 `server` 標籤之間，將下列 XML 新增至 mqwebuser.xml 檔案：

```
<variable name="traceSpec"  
value="*=info:com.ibm.mq*=all:com.ibm.mq.rest*=all:js.mq*=all" />
```

如果 `traceSpec` 變數存在於 mqwebuser.xml 檔案中，請將值屬性新增至變數。請以冒號區隔每一個值屬性。

如果 mqweb 伺服器正在執行中，則會立即啟用追蹤。

追蹤會輸出至一組檔案。作用中檔案稱為 `trace.log`。歷程追蹤保存在稱為 `trace_timestamp.log` 的檔案中。您可以設定 `maxTraceFileSize` 和 `maxTraceFiles` 變數來配置這些追蹤檔的大小，以及保留的歷程檔數目。 **V 9.0.1** 如需相關資訊，請參閱 [配置記載](#)。

針對在 mqweb 伺服器中執行的 IBM MQ Classes for JMS 程式碼啟用追蹤

1. 在下列其中一個目錄中建立稱為 `jmstrace.config` 的檔案：

- **Windows** **Linux** **AIX** `MQ_DATA_DIRECTORY/web/installations/
installationName/servers/mqweb`
- **z/OS** `WLP_user_directory/servers/mqweb`

其中 `WLP_user_directory` 是執行 `crtmqweb.sh` Script 以建立 `mqweb` 伺服器定義時指定的目錄。

2. 將下列幾行新增至 `jmstrace.config` 檔案：

```
com.ibm.msg.client.commonservices.trace.outputName=PATH/logs/jmstrace.txt
com.ibm.msg.client.commonservices.trace.limit=104857600
com.ibm.msg.client.commonservices.trace.count=10
com.ibm.msg.client.commonservices.trace.status=0N
```

其中 `PATH` 指定您要寫入 `jmstrace.txt` 檔案之目錄的完整路徑。

這些行會將追蹤檔大小上限設為 100 MB，並將追蹤檔數目上限設為 10。請確定這些檔案有可用的磁碟空間。

3. 在與 `jmstrace.config` 檔案相同的目錄中，開啟或建立 `jvm.options` 檔案。

4. 將下列幾行新增至 `jvm.options` 檔案：

```
-Dcom.ibm.msg.client.commonservices.trace.startup=TRUE
-Dcom.ibm.msg.client.config.location=CONFIG_PATH/jmstrace.config
```

其中 `CONFIG_PATH` 將 `jmstrace.config` 檔案所在目錄的完整路徑指定為 URL。例如，`file:c:/ProgramData/IBM/MQ/web/installations/Installation2/servers/mqweb/`。

5. 在指令行上使用 `endmqweb` 及 `strmqweb` 指令，以重新啟動 `mqweb` 伺服器。

提供給 IBM 服務中心的資訊

當您收集 IBM 服務的診斷資訊時，請包括下列檔案和目錄：

- `mqweb.xml` 檔案。
- 包含 `mqweb` 伺服器定義的目錄內容：

- **Windows** **Linux** **AIX** `MQ_DATA_DIRECTORY/web/installations/
installationName`
- **z/OS**

執行 `crtmqweb.sh` Script 以建立 `mqweb` 伺服器定義時指定的目錄。依預設，此目錄為 `/var/mqm/web/installation1`。

V 9.0.2 **Linux** 追蹤 IBM MQ Bridge to Salesforce

提供 IBM MQ Bridge to Salesforce 的追蹤機能可協助 IBM 人員診斷客戶問題。當您發出 `runmqsfb` 指令來啟動橋接器時，請啟用 IBM MQ Bridge to Salesforce 的追蹤，並定義除錯層次。

程序

1. 設定環境變數 `MQSFB_EXTRA_JAVA_OPTIONS` 以指定 `-D` Java 選項並開啟 IBM MQ classes for JMS 追蹤。


```
export MQSFB_EXTRA_JAVA_OPTIONS="-Dcom.ibm.msg.client.commonservices.trace.status=ON"
```

2. 當您在執行時期發出 **runmqsfb** 指令時，請將除錯層次設為詳細模式 **-d 2**。

```
runmqsfb -f new_config.cfg -r logFile.log -d 2
```

logFile.log 包含可能有助於解決 IBM MQ Bridge to Salesforce 問題的資訊。

3. 選擇性的: 您可以透過建立 IBM MQ classes for JMS 配置檔，對確切追蹤達到更細微的控制。如需相關資訊，請參閱第 78 頁的『追蹤 IBM MQ classes for JMS 應用程式』，並遵循 IBM 服務支援代表提供的建議。

相關資訊

[runmqsfb \(執行 IBM MQ Bridge to Salesforce\)](#)

[執行 IBM MQ Bridge to Salesforce](#)

[監視 IBM MQ Bridge to Salesforce](#)

V 9.0.3

z/OS

Linux

MQ Adv. VUE

追蹤 IBM MQ Bridge to blockchain

提供 IBM MQ Bridge to blockchain 的追蹤機能可協助 IBM 人員診斷客戶問題。當您發出 **runmqbcb** 指令來啟動橋接器時，請啟用 IBM MQ Bridge to blockchain 的追蹤，並定義除錯層次。

程序

1. 設定環境變數 **MQBCB_EXTRA_JAVA_OPTIONS** 以指定 **-D** Java 選項並開啟 IBM MQ classes for JMS 追蹤。

```
export MQBCB_EXTRA_JAVA_OPTIONS="-Dcom.ibm.msg.client.commonservices.trace.status=ON"
```

2. 當您在執行時期發出 **runmqbcb** 指令時，請將除錯層次設為詳細模式 **-d 2**。

```
./runmqbcb.sh -f new_config.cfg -r logFile.log -d 2
```

logFile.log 包含可能有助於解決 IBM MQ Bridge to blockchain 問題的資訊。

3. 選擇性的: 您可以透過建立 IBM MQ classes for JMS 配置檔，對確切追蹤達到更細微的控制。如需相關資訊，請參閱第 78 頁的『追蹤 IBM MQ classes for JMS 應用程式』，並遵循 IBM 服務支援代表提供的建議。

相關資訊

[runmqbcb \(執行 IBM MQ Bridge to Blockchain\)](#)

[執行 IBM MQ Bridge to blockchain](#)

V 9.0.0.9

啟用 LDAP 用戶端程式庫程式碼的動態追蹤

從 IBM MQ 9.0.0 Fix Pack 9 開始，可以在不同時停止或啟動佇列管理程式的情況下，開啟及關閉 LDAP 用戶端追蹤。

關於這項作業

在 IBM MQ 9.0.0 Fix Pack 9 之前，如果未同時停止或啟動佇列管理程式，則無法開啟及關閉 LDAP 用戶端追蹤。

從 IBM MQ 9.0.0 Fix Pack 9 開始，您可以使用 **strmqtrc** 指令開啟 LDAP 用戶端追蹤，並使用 **endmqtrc** 指令關閉 LDAP 用戶端追蹤，而不需要停止或啟動佇列管理程式。若要啟用此行為，還需要將環境變數 **AMQ_LDAP_TRACE** 設為非空值。

當 **AMQ_LDAP_TRACE** 設為非空值，且使用 LDAP 功能時，部分佇列管理程式處理程序會在 `/var/mqm/trace` 下建立零長度檔案。然後使用 **strmqtrc** 指令開啟追蹤時，會將部分追蹤資訊寫入這些檔案。稍後，當使用 **endmqtrc** 指令關閉追蹤時，追蹤資訊會停止寫入檔案，但檔案的控點會保持開啟狀態，直到佇列管理程式結束為止。

UNIX 在 UNIX 平台上，透過使用 `rm` 指令解除鏈結這些檔案，無法完全釋放檔案系統空間。這是控點保持開啟的事實所造成的副作用。因此，每當需要釋放 `/var/mqm/trace` 中的磁碟空間時，就應該執行佇列管理程式結束。

程序

- 將環境變數 `AMQ_LDAP_TRACE` 設為非空值。
- 使用 `strmqtrc` 指令來切換追蹤：

```
strmqtrc -m QMNAME -t servicedata
```

- 使用 `endmqtrc` 指令來關閉追蹤。

z/OS 上的問題判斷

IBM MQ for z/OS、CICS、Db2 及 IMS 會產生可用於問題判斷的診斷資訊。

本節包含下列主題的相關資訊：

- 當偵測到問題時，佇列管理程式所嘗試的回復動作。
- IBM MQ for z/OS 異常終止，以及發生異常終止時所產生的資訊。
- IBM MQ for z/OS 所產生的診斷資訊，以及有用資訊的其他來源。

提供協助進行問題判斷及應用程式除錯的資訊類型，視所發現的錯誤類型及子系統的設定方式而定。

如需 IBM MQ for z/OS 上問題判斷及診斷資訊的相關資訊，請參閱下列子主題。

- [第 96 頁的『IBM MQ for z/OS 效能限制』](#)
- [第 97 頁的『IBM MQ for z/OS 回復動作』](#)
- [第 98 頁的『IBM MQ for z/OS 異常終止』](#)
- [第 101 頁的『在 IBM MQ for z/OS 上產生的診斷資訊』](#)
- [第 102 頁的『IBM MQ for z/OS 的其他問題判斷資訊來源』](#)
- [第 103 頁的『CICS 的診斷輔助工具』](#)
- [第 104 頁的『IMS 的診斷輔助工具』](#)
- [第 104 頁的『Db2 的診斷輔助工具』](#)
- [第 104 頁的『IBM MQ for z/OS 傾出』](#)
- [第 122 頁的『處理 z/OS 上的效能問題』](#)
- [第 127 頁的『處理 z/OS 上不正確的輸出』](#)

相關概念

[第 7 頁的『疑難排解概觀』](#)

疑難排解是尋找並消除問題原因的程序。每當您使用 IBM 軟體時發生問題，只要您問自己「發生什麼事？」，疑難排解程序即會開始。

[第 37 頁的『使用錯誤日誌』](#)

您可以使用各種錯誤日誌來協助進行問題判斷及疑難排解。

[第 45 頁的『First Failure Support Technology \(FFST\)』](#)

First Failure Support Technology (FFST) for IBM MQ 提供事件的相關資訊，這些事件在發生錯誤時可協助 IBM 支援人員診斷問題。

相關工作

[第 56 頁的『使用追蹤』](#)

您可以使用不同類型的追蹤，協助您判斷問題及進行疑難排解。

請利用這個主題來調查可能造成效能限制的 z/OS 資源。

在自訂 IBM MQ for z/OS 時，有一些可能會影響系統執行方式的決策。這些決定包括：

- 資料集的大小及放置
- 緩衝區配置
- 佇列在頁集之間的分佈，以及「連結機能」結構
- 您在任何時間容許存取佇列管理程式的作業數

日誌緩衝池

日誌緩衝區不足可能會導致應用程式等待日誌緩衝區可用，這可能會影響 IBM MQ 效能。RMF 報告可能會對保留日誌資料集的磁區顯示大量 I/O。

您可以使用三個參數來調整日誌緩衝區。最重要的是 OUTBUFF。如果日誌管理程式統計資料 QJSTWTB 大於 0，請增加日誌緩衝區的大小。此參數控制在寫入作用中日誌資料集之前要填入的緩衝區數目 (在 1-256 範圍內)。持續訊息的確定及非同步點外處理會導致日誌緩衝區寫出至日誌。因此，除非處理大型訊息，且確定數目或不同步點訊息數目偏低，否則此參數可能幾乎沒有作用。這些參數指定在 CSQ6LOGP 巨集中 (如需詳細資料，請參閱 [使用 CSQ6LOGP](#))，而重要的參數如下：

OUTBUFF

此參數控制輸出緩衝區的大小 (範圍從 40 KB 到 4000 KB)。

WRTHRSH

此參數控制在寫入作用中日誌資料集之前要填入的緩衝區數目 (在 1 到 256 範圍內)。

您也必須注意 CSQ6SYSP 巨集的 LOGLOAD 參數。此參數指定在檢查點記錄之間寫入的日誌記錄數。範圍是 200 到 16 000 000，但大型系統的一般值是 500 000。如果某個值太小，您會收到頻繁的檢查點，這會耗用處理器時間，並可能導致額外的磁碟 I/O。

緩衝池大小

每一個頁集都有相關聯的緩衝池。您可以使用 [DEFINE BUFFPOOL](#) 指令來指定緩衝池中的緩衝區數目。

不正確的緩衝池大小規格可能會對 IBM MQ 效能產生不利影響。緩衝池越小，需要實體 I/O 的頻率越高。RMF 可能會對保留頁集的磁區顯示大量 I/O。對於僅具有短期訊息的緩衝池，緩衝區管理程式統計資料 QPSTSLA、QPSTSOS 及 QPSTRIO 通常必須為零。對於其他緩衝池，QPSTSOS 及 QPSTSTLA 必須為零。

可用 DASD 上資料集的分佈

DASD 上的頁面資料集分佈會對 IBM MQ 的效能產生重大影響。

將日誌資料集放在低用量磁區上，並將日誌 n 和日誌 $n+1$ 放在不同磁區上。請確定雙重日誌放在 DASD 的不同控制裝置上，且磁區不在相同的實體磁碟上。

頁集上佇列的分佈

頁集上佇列的配送可能會影響效能。此效能變更可由交易使用特定佇列 (位於大量使用的頁集上) 所經歷的不良回應時間來指出。RMF 報告可能會對包含受影響頁集的磁區顯示大量 I/O。

您可以透過定義指定特定頁集的儲存類別 (STGCLASS) 物件，然後在佇列定義中定義 STGCLASS 參數，將佇列指派給特定頁集。最好以這種方式在不同頁集中定義大量使用的佇列。

在「連結機能」結構上配送佇列

在「連結機能」結構上配送佇列可能會影響效能。

佇列共用群組最多可以連接至 64 個「連結機能」結構，其中一個必須是管理結構。您可以使用 IBM MQ 資料的其餘 63 個「連結機能」結構，每一個結構最多包含 512 個佇列。如果您需要多個「連結機能」結構，請根據佇列的功能來區隔數個結構中的佇列。

您可以採取一些步驟，將效率最大化：

- 刪除您不再需要的任何「連結機能」結構。
- 將應用程式使用的所有佇列放置在相同的「連結機能」上，以提高應用程式處理效率。
- 如果工作對效能特別敏感，請選擇更快速的「連結機能」結構。

請考量如果您遺失「連結機能」結構，則會遺失其中儲存的任何非持續訊息。如果佇列分散在各種「連結機能」結構中，則遺失這些非持續訊息可能會導致一致性問題。若要使用持續訊息，您必須定義至少具有 CFLEVEL (3) 及 RECOVER (YES) 的「連結機能」結構。

並行執行緒的限制

存取佇列管理程式的作業數也會影響效能，尤其是在有其他限制 (例如儲存體) 或有許多作業存取少數佇列時。症狀可能是針對一個以上頁集的大量 I/O，或已知存取相同佇列之作業的不良回應時間。對於 TSO 和批次，IBM MQ 中的執行緒數目限制為 32767。

在 CICS 環境中，您可以使用 CICS MAXTASK 來限制並行存取。

使用 IBM MQ 追蹤進行管理

雖然您有時可能必須使用特定的追蹤，但使用追蹤機能會對系統效能產生負面影響。

請考量您要將追蹤資訊傳送至哪個目的地。使用內部追蹤表格會儲存 I/O，但對於產生大量資料的追蹤而言不夠大。

統計資料追蹤會依間隔收集資訊。間隔由 CSQ6SYSP 巨集的 STATIME 參數控制，如 [使用 CSQ6SYSP](#) 中所述。當作業或通道結束時 (可能在許多天之後)，會產生帳戶追蹤記錄。

您可以依類別、資源管理程式 ID (RMID) 及檢測機能 ID (IFCID) 來限制追蹤，以減少收集的資料量。如需相關資訊，請參閱 [START TRACE](#)。

z/OS IBM MQ for z/OS 回復動作

請利用這個主題來瞭解使用者偵測到的回復動作，以及佇列管理程式偵測到的錯誤。

IBM MQ for z/OS 可以從使用者資料不正確所造成的程式檢查中回復。會向呼叫者發出完成及原因碼。這些代碼記錄在 [IBM MQ for z/OS 訊息、完成及原因碼](#) 中。

程式錯誤

程式錯誤可能與使用者應用程式碼或 IBM MQ 程式碼相關聯，並分成兩個種類：

- 使用者偵測到錯誤
- [子系統偵測到錯誤](#)

使用者偵測到錯誤

當服務要求的結果不符合預期 (例如，非零完成碼) 時，使用者 (或使用者撰寫的應用程式) 會偵測到使用者偵測到的錯誤。無法自動收集問題判斷資料，因為偵測會在 IBM MQ 功能完成之後進行。在啟動 IBM MQ 使用者參數追蹤機能的情況下重新執行應用程式，可以提供分析問題所需的資料。此追蹤的輸出會導向一般性追蹤機能 (GTF)。

您可以使用操作員指令來開啟及關閉追蹤。如需相關資訊，請參閱第 65 頁的『[在 z/OS 上使用追蹤來判斷問題](#)』。

佇列管理程式偵測到錯誤

佇列管理程式會偵測如下的錯誤:

- 程式檢查
- 資料集填入
- 內部一致性錯誤

IBM MQ 會分析錯誤，並採取下列動作:

- 如果問題是由使用者或應用程式錯誤所造成 (例如所使用的無效位址)，則完成及原因碼會將錯誤反映回應用程式。
- 如果問題不是由使用者或應用程式錯誤所造成 (例如，已使用所有可用的 DASD，或系統偵測到內部不一致)，則可能的話，IBM MQ 會將完成及原因碼傳送至應用程式，或如果無法如此，則會停止應用程式，以回復。
- 如果 IBM MQ 無法回復，它會以特定的原因碼終止。SVC 傾出通常會在傾出的系統診斷工作區 (SDWA) 及變數記錄區 (VRA) 部分中記錄資訊，並在 SYS1.LOGREC。

z/OS IBM MQ for z/OS 異常終止

對於 z/OS 或其他 z/OS 系統，在 WebSphere 中可能會發生異常終止。請利用這個主題來瞭解 IBM MQ 系統異常終止碼，以及如何調查 CICS、IMS 和 z/OS 中發生的異常終止。

IBM MQ for z/OS 使用兩個系統異常終止完成碼: X'5C6' 和 X'6C6'。這些代碼識別:

- 作業期間發現內部錯誤
- 問題判斷的診斷資訊
- 錯誤中涉及的元件所起始的動作

X'5C6'

X'5C6' 異常終止完成碼指出 IBM MQ 偵測到內部錯誤，且已異常終止內部作業 (TCB) 或使用者連接的作業。與 X'5C6' 異常終止完成碼相關聯的錯誤之前可能有 z/OS 系統代碼或內部錯誤。

請檢查 X'5C6' 異常終止所產生的診斷資料，以判斷實際導致後續作業或子系統終止的錯誤來源。

X'6C6'

X'6C6' 異常終止完成碼指出 IBM MQ 偵測到嚴重錯誤，且已異常終止佇列管理程式。當發出 X'6C6' 時，IBM MQ 已判定繼續作業可能導致資料完整性遺失。與 X'6C6' 異常終止完成碼相關聯的錯誤之前可能有 z/OS 系統錯誤、一或多個 X'5C6' 異常終止完成碼，或有錯誤訊息 CSQV086E 指出 IBM MQ 異常終止。

第 98 頁的表 10 彙總發出這些異常終止完成碼時 IBM MQ for z/OS 可用的動作及診斷資訊。此資訊的不同部分在不同的錯誤狀況中相關。針對特定錯誤產生的資訊取決於特定問題。如需提供診斷資訊之 z/OS 服務的相關資訊，請參閱第 101 頁的『在 IBM MQ for z/OS 上產生的診斷資訊』。

	X'5C6'	X'6C6'
說明	<ul style="list-style-type: none">• IBM MQ 正常作業期間發生錯誤	<ul style="list-style-type: none">• 嚴重錯誤; 繼續作業可能會危及資料完整性
系統動作	<ul style="list-style-type: none">• 內部 IBM MQ 作業異常終止• 已連接使用者作業異常終止	<ul style="list-style-type: none">• 整個 IBM MQ 子系統異常終止• 具有作用中 IBM MQ 連線的使用者作業可能會異常終止，並產生 X'6C6' 程式碼• 所連接聯合位址空間的可能 MEMTERM (記憶體終止)

表 10: 異常終止完成碼 (繼續)

	X'5C6'	X'6C6'
診斷資訊	<ul style="list-style-type: none"> • SVC 傾出 • SYS1.LOGREC 項目 • VRA 資料項目 	<ul style="list-style-type: none"> • SYS1.LOGREC • VRA 資料項目
相關聯的原因碼	<ul style="list-style-type: none"> • IBM MQ 異常終止原因碼 (abend reason code) • 相關聯的 z/OS 系統代碼 	<ul style="list-style-type: none"> • 子系統終止原因碼 • X'6C6' 異常終止之前的 z/OS 系統完成碼及 X'5C6' 碼
隨附代碼的位置	<ul style="list-style-type: none"> • SVC 傾出標題 • 訊息 CSQW050I • Register 15 of SDWA section <i>General Purpose Registers at Time of Error</i> • SYS1.LOGREC 項目 • VRA 資料項目 	<ul style="list-style-type: none"> • SYS1.LOGREC • VRA 資料項目 • 訊息 CSQV086E, 傳送至 z/OS 系統操作員

相關概念

第 99 頁的『[處理 IBM MQ for z/OS 上的異常終止](#)』

應用程式及其他 z/OS 系統可能會發生異常終止。請利用這個主題來調查程式異常終止、批次異常終止、CICS 交易異常終止及 IMS 交易異常終止。

第 100 頁的『[CICS、IMS 及 z/OS 異常終止](#)』

請利用這個主題來調查 CICS、IMS 和 z/OS 的異常終止。

第 101 頁的『[在 IBM MQ for z/OS 上產生的診斷資訊](#)』

請利用這個主題來調查 z/OS 所產生的部分診斷資訊，這些資訊有助於判斷問題，並瞭解如何調查錯誤訊息、傾出、主控台日誌、工作輸出、症狀字串及佇列輸出。

第 104 頁的『[IBM MQ for z/OS 傾出](#)』

如需在問題判斷中使用傾出的相關資訊，請使用本主題。它說明在查看 IBM MQ for z/OS 位址空間所產生的傾出時，您應該採取的步驟。

處理 IBM MQ for z/OS 上的異常終止

應用程式及其他 z/OS 系統可能會發生異常終止。請利用這個主題來調查程式異常終止、批次異常終止、CICS 交易異常終止及 IMS 交易異常終止。

異常終止類型

程式異常終止可能是因為應用程式無法從 IBM MQ 檢查及回應原因碼。例如，如果未收到訊息，則使用訊息中已設定的欄位進行計算可能會導致 X'0C4' 或 X'0C7' 異常終止 (CICS 中的 ASRA 異常終止)。

下列資訊片段指出程式異常終止：

- 主控台日誌中來自 IBM MQ 的錯誤訊息
- CICS 錯誤訊息
- CICS 交易傾出
- IMS 區域傾出
- 使用者或主要終端機上的 IMS 訊息
- 批次或 TSO 輸出中的程式傾出資訊
- 批次工作輸出中的異常終止訊息
- TSO 畫面上的異常終止訊息

如果您有異常終止碼，請參閱下列其中一本手冊，以取得異常終止原因的說明：

- 如需 IBM MQ for z/OS 異常終止 (異常終止碼 X'5C6' 和 X'6C6')，請參閱 [IBM MQ for z/OS 訊息、完成及原因碼](#)
- 若為批次異常終止，請參閱 [z/OS MVS 系統代碼 手冊](#)
- 若為 CICS 異常終止，則為 [CICS 訊息](#)
- 若為 IMS 異常終止，[IMS 訊息與訊息碼](#)
- 若為 Db2 異常終止，[訊息](#)
- Db2
- 若為 RRS 異常終止，[z/OS MVS 系統訊息，第 3 冊](#)
- 若為 XES 異常終止，[z/OS MVS 系統訊息，第 10 冊](#)

批次異常終止

批次異常終止會導致在 syslog 中顯示包含暫存器內容相關資訊的錯誤訊息。TSO 異常終止會導致在 TSO 畫面上產生包含類似資訊的錯誤訊息。如果步驟有 SYSUDUMP DD 陳述式，則會採用 SYSUDUMP (請參閱 [第 104 頁的『IBM MQ for z/OS 傾出』](#))。

CICS 交易異常終止

CICS 交易異常終止會記錄在 CICS CSMT 日誌中，並在終端機上產生一則訊息 (如果有的話)。CICS AICA 異常終止指出可能的迴圈。如需相關資訊，請參閱 [第 126 頁的『處理 z/OS 上的迴圈』](#)。如果您有 CICS 異常終止，則使用 CEDF 及 CICS 追蹤可協助您尋找問題的原因。如需相關資訊，請參閱 [CICS 疑難排解](#) (早期稱為 [CICS 問題判斷手冊](#))。

IMS 交易異常終止

IMS 交易異常終止會記錄在 IMS 主要終端機上，並在終端機上產生錯誤訊息 (如果有的話)。如果您有 IMS 異常終止，請參閱 [IMS 疑難排解](#)。

CICS、IMS 及 z/OS 異常終止

請利用這個主題來調查 CICS、IMS 和 z/OS 的異常終止。

CICS 異常終止

如果應用程式連接至終端機或 CSMT 日誌，則會將 CICS 異常終止訊息傳送至終端機。CICS 異常終止碼在 [CICS 訊息與訊息碼 手冊](#) 中說明。

CICS 配接器會發出以字母 Q 開頭的異常終止原因碼 (例如，QDCL)。這些代碼記錄在 [IBM MQ for z/OS 訊息、完成及原因碼](#) 中

IMS 異常終止

在下列其中一種情況下，IMS 應用程式可能會異常終止：

- 正常異常終止。
- IMS 虛擬異常終止，因 ESAF 結束程式中的錯誤而產生異常終止碼，例如 U3044。
- 異常終止 3051 或 3047，當 REO (區域錯誤選項) 已指定為 "Q" 或 "A"，且 IMS 應用程式嘗試參照無法運作的外部子系統時，或當建立執行緒時無法使用資源時。

IMS 訊息會傳送至使用者終端機或工作輸出，以及 IMS 主要終端機。異常終止可能伴隨區域傾出。

z/OS 異常終止

在 IBM MQ 作業期間，z/OS 系統完成碼可能會發生異常終止。如果您收到 z/OS 異常終止，請參閱適當的 z/OS 出版品。

z/OS 在 IBM MQ for z/OS 上產生的診斷資訊

請利用這個主題來調查 z/OS 所產生的部分診斷資訊，這些資訊有助於判斷問題，並瞭解如何調查錯誤訊息、傾出、主控台日誌、工作輸出、症狀字串及佇列輸出。

IBM MQ for z/OS 功能回復常式使用 z/OS 服務來提供診斷資訊，以協助您判斷問題。

下列 z/OS 服務提供診斷資訊：

SVC 傾出

IBM MQ 異常終止完成碼 X'5C6' 使用 z/OS SDUMP 服務來建立 SVC 傾出。根據發生錯誤時的特定錯誤及佇列管理程式狀態，與這些傾出相關聯的內容及儲存區會有所不同。

SYS1.LOGREC

在使用 z/OS SETRP 服務發生錯誤時，會在 SYS1.LOGREC 資料集中要求項目。下列資訊也記錄在 SYS1.LOGREC：

- 子系統異常終止
- 回復常式中發生次要異常終止
- 來自回復終止管理程式的要求

變數記錄區域 (VRA) 資料

使用 z/OS VRA 定義的金鑰，將資料項目新增至 SDWA 的 VRA。VRA 資料包括所有 IBM MQ for z/OS 異常終止完成碼共用的一系列診斷資料項目。呼叫元件回復常式或回復終止管理程式會在起始錯誤處理期間提供其他資訊。

IBM MQ for z/OS 提供獨特的訊息，與傾出的輸出一起，旨在提供足夠的資料來診斷問題，而不必嘗試重新產生它。這稱為首次失敗資料擷取。

錯誤訊息

當偵測到問題時，IBM MQ 會產生錯誤訊息。IBM MQ 診斷訊息以字首 CSQ 開頭。IBM MQ 所產生的每一個錯誤訊息都是唯一的；也就是說，只會針對一個錯誤而產生。您可以在 [IBM MQ for z/OS 訊息、完成及原因碼](#) 中找到錯誤的相關資訊。

IBM MQ 模組名稱的前三個字元通常也是 CSQ。但 C++ (IMQ) 及標頭檔 (CMQ) 的模組例外。第四個字元可唯一識別元件。字元 5 到 8 在前四個字元所識別的群組內是唯一的。

請確定您有一些在安裝時所撰寫之程式的應用程式訊息及程式碼說明文件，以及檢視 [IBM MQ for z/OS 訊息、完成及原因碼](#)

在某些情況下，可能未產生任何訊息，或者如果產生訊息，則無法進行通訊。在這些情況下，您可能必須分析傾出，以隔離特定模組的錯誤。如需使用傾出的相關資訊，請參閱 [第 104 頁的『IBM MQ for z/OS 傾出』](#)。

傾出

傾出是問題詳細資訊的重要來源。無論它們是異常終止或使用者要求的結果，它們都可讓您查看取得傾出時所發生狀況的 Snapshot。第 104 頁的 [『IBM MQ for z/OS 傾出』](#) 包含使用傾出在 IBM MQ 系統中找出問題的指引。不過，因為它們只提供 Snapshot，所以您可能需要將它們與涵蓋較長時段的其他資訊來源 (例如日誌) 搭配使用。

在處理 MQI 呼叫時，也會針對特定類型的錯誤產生 SNAP 傾出。傾出會寫入 CSQSNAP DD。

主控台日誌及工作輸出

您可以將主控台日誌複製到永久資料集，或視需要列印它們。如果您只對特定事件感興趣，則可以選取要列印主控台日誌的哪些部分。

工作輸出包括從執行工作產生的輸出，以及從主控台產生的輸出。您可以將此輸出複製到永久資料集，或視需要列印它。您可能需要收集所有相關聯工作的輸出，例如 CICS、IMS 及 IBM MQ。

症狀字串

症狀字串以結構化格式顯示重要診斷資訊。產生症狀字串時，它可在下列一或多個位置中使用：

- 在 z/OS 系統主控台上
- 在 SYS1.LOGREC
- 在任何取得的傾出中

第 102 頁的圖 7 顯示症狀字串的範例。

```
PIDS/ 5655R3600 RIDS/CSQMAIN1 AB/S6C6 PRCS/0E30003
```

圖 7: 範例症狀字串

症狀字串提供一些關鍵字，您可以用來搜尋 IBM 軟體支援中心資料庫。如果您具有其中一個選用搜尋工具的存取權，則可以自行搜尋資料庫。如果您向 IBM 支援中心報告問題，通常會要求您以引號括住症狀字串。

雖然症狀字串設計為提供用於搜尋資料庫的關鍵字，但它也可以提供發生錯誤時所發生狀況的許多相關資訊，並可能建議開始調查的明顯原因或承諾區域。

佇列資訊

您可以使用作業及控制台來顯示佇列狀態的相關資訊。或者，您可以從 z/OS 主控台輸入 DISPLAY QUEUE 及 DISPLAY QSTATUS 指令。

註: 如果從主控台發出指令，則會將回應複製到主控台日誌，讓文件可以緊密保存在一起。

相關概念

第 65 頁的『[在 z/OS 上使用追蹤來判斷問題](#)』

有不同的追蹤選項可用於 IBM MQ 的問題判斷。請利用這個主題來瞭解不同的選項，以及如何控制追蹤。

第 102 頁的『[IBM MQ for z/OS 的其他問題判斷資訊來源](#)』

請利用這個主題來調查 IBM MQ for z/OS 問題判斷的其他資訊來源。

第 103 頁的『[CICS 的診斷輔助工具](#)』

您可以使用 CICS 診斷交易來顯示佇列管理程式作業及 MQI 呼叫的相關資訊。請利用這個主題來調查這些機能。

第 104 頁的『[IMS 的診斷輔助工具](#)』

請利用這個主題來調查 IMS 診斷機能。

第 104 頁的『[Db2 的診斷輔助工具](#)』

請利用這個主題來調查 Db2 診斷工具的參照。

z/OS

IBM MQ for z/OS 的其他問題判斷資訊來源

請利用這個主題來調查 IBM MQ for z/OS 問題判斷的其他資訊來源。

在解決 IBM MQ for z/OS 的問題時，您可能會發現下列文件項目很有用。

- [您自己的文件](#)

- [您正在使用之產品的 Documentation](#)
- [來源清單和鏈結編輯對映](#)
- [變更日誌](#)
- [系統配置圖表](#)
- [DISPLAY CONN 指令的資訊](#)

您自己的文件

您自己的文件是組織所產生的資訊集合，這些資訊是關於您的系統和應用程式應該執行的動作，以及它們應該如何執行。您需要多少此資訊取決於您對相關系統或應用程式的熟悉程度，可能包括：

- 程式說明或功能規格
- 系統中活動流程的流程圖或其他說明
- 程式的變更歷程
- 安裝的變更歷程
- 顯示平均輸入、輸出及回應時間的統計及監視設定檔

您正在使用之產品的 Documentation

您正在使用之產品的說明文件是 IBM MQ 檔案庫中的 InfoCenters，以及與應用程式搭配使用之任何其他產品的檔案庫中。

請確定您參照的任何文件層次符合您正在使用的系統層次。通常透過使用已作廢資訊或尚未安裝之產品層次的相關資訊來產生問題。

來源清單和鏈結編輯對映

使用您的文件集併入在安裝時撰寫的任何應用程式的來源清單。(它們通常是文件中最大的單一元素。)請確定您將鏈結編輯器中的相關輸出併入來源清單中，以避免浪費時間嘗試在具有過期鏈結對映的載入模組中找到您的方法。請務必在清單開頭併入 JCL，以顯示已使用的程式庫，以及載入模組所在的載入程式庫。

變更日誌

變更日誌中的資訊可以告訴您在資料處理環境中所做的變更，這些變更可能導致應用程式發生問題。如果要充分利用變更日誌，請包含硬體變更、系統軟體 (例如 z/OS 和 IBM MQ) 變更、應用程式變更，以及對作業程序所做的任何修改的相關資料。

系統配置圖表

系統配置圖表顯示哪些系統在執行中、它們在何處執行，以及系統如何彼此連接。它們也會顯示哪些 IBM MQ、CICS 或 IMS 系統是測試系統，哪些是正式作業系統。

DISPLAY CONN 指令的資訊

DISPLAY CONN 指令提供哪些應用程式連接至佇列管理程式的相關資訊，以及協助您診斷具有長時間執行工作單元之應用程式的資訊。您可以定期收集此資訊，並檢查它是否有任何長時間執行的工作單元，然後顯示該連線的詳細資訊。

CICS 的診斷輔助工具

您可以使用 CICS 診斷交易來顯示佇列管理程式作業及 MQI 呼叫的相關資訊。請利用這個主題來調查這些機能。

您可以使用 CKQC 交易 (CICS 配接卡控制台) 來顯示佇列管理程式作業的相關資訊，以及它們所處的狀態 (例如 GET WAIT)。如需 CKQC 的相關資訊，請參閱 [管理 IBM MQ for z/OS](#)。

應用程式開發環境與任何其他 CICS 應用程式相同，因此您可以使用該環境中通常使用的任何工具來開發 IBM MQ 應用程式。尤其是 CICS 執行診斷機能 (CEDF) 會針對每一個 MQI 呼叫，以及所有 CICS API 服務的設陷呼叫，設陷進入及結束 CICS 配接器。[CEDF 輸出範例](#)中提供此機能所產生的輸出範例。

CICS 配接器也會將追蹤項目寫入 CICS 追蹤。這些項目在 [第 72 頁的『CICS 配接器追蹤項目』](#)中說明。

可以從 CICS 區域取得其他追蹤及傾出資料。這些項目如 *CICS 問題判斷手冊*中所述。

z/OS IMS 的診斷輔助工具

請利用這個主題來調查 IMS 診斷機能。

應用程式開發環境與任何其他 IMS 應用程式相同，因此在該環境中通常使用的任何工具都可以用來開發 IBM MQ 應用程式。

可以從 IMS 區域取得追蹤及傾出資料。這些項目如 *IMS/ESA Diagnosis Guide and Reference* 手冊中所述。

z/OS Db2 的診斷輔助工具

請利用這個主題來調查 Db2 診斷工具的參照。

請參閱下列手冊，以取得診斷 Db2 問題的說明：

- *Db2 for z/OS Diagnosis Guide and Reference*
- Db2 訊息與訊息碼

V 9.0.3

z/OS

MQ Adv. VUE

疑難排解從 IBM MQ for z/OS 到 Product Insights 的連線

IBM Cloud Product Insights 服務不再可用。如需相關資訊，請參閱下列部落格文章：[Service Deprecation: IBM Cloud Product Insights](#)。

z/OS IBM MQ for z/OS 傾出

如需在問題判斷中使用傾出的相關資訊，請使用本主題。它說明在查看 IBM MQ for z/OS 位址空間所產生的傾出時，您應該採取的步驟。

如何使用傾出進行問題判斷

解決 IBM MQ for z/OS 系統的問題時，您可以透過兩種方式來使用傾出：

- 檢查 IBM MQ 處理來自應用程式的要求的方式。
若要這樣做，您通常需要分析整個傾出，包括控制區塊及內部追蹤。
- 若要識別 IBM MQ for z/OS 本身的問題，請在 IBM 支援中心人員的指示下。

請使用下列主題中的指示來取得及處理傾出：

- [第 105 頁的『使用 IBM MQ for z/OS 取得傾出』](#)
- [第 105 頁的『使用 z/OS DUMP 指令』](#)
- [第 107 頁的『使用 IBM MQ for z/OS 傾出顯示畫面處理傾出』](#)
- [第 111 頁的『使用線路模式 IPCS 處理 IBM MQ for z/OS 傾出』](#)
- [第 117 頁的『以批次方式使用 IPCS 處理 IBM MQ for z/OS 傾出』](#)

傾出標題可能會在異常終止及原因碼中提供足夠資訊來解決問題。您可以在主控台日誌中或使用 z/OS 指令 DISPLAY DUMP, TITLE 來查看傾出標題。[第 118 頁的『分析 z/OS 上的傾出及解譯傾出標題』](#)中說明傾出標題的格式。如需 IBM MQ for z/OS 異常終止碼的相關資訊，請參閱 [第 98 頁的『IBM MQ for z/OS 異常終止』](#)，以及 [IBM MQ for z/OS 訊息、完成及原因碼](#)中所記載的異常終止原因碼。

如果傾出標題中沒有足夠的問題相關資訊，請格式化傾出以顯示其中包含的其他資訊。

如需不同傾出類型的相關資訊，請參閱下列主題：

- [第 120 頁的『z/OS 上的 SYSUDUMP 資訊』](#)
- [第 120 頁的『z/OS 上的 SNAP 傾出』](#)
- [第 121 頁的『z/OS 上的 SYS1.LOGREC 資訊』](#)
- [第 121 頁的『z/OS 上的 SVC 傾出』](#)

相關概念

[第 65 頁的『在 z/OS 上使用追蹤來判斷問題』](#)

有不同的追蹤選項可用於 IBM MQ 的問題判斷。請利用這個主題來瞭解不同的選項，以及如何控制追蹤。

[第 98 頁的『IBM MQ for z/OS 異常終止』](#)

對於 z/OS 或其他 z/OS 系統，在 WebSphere 中可能會發生異常終止。請利用這個主題來瞭解 IBM MQ 系統異常終止碼，以及如何調查 CICS、IMS 和 z/OS 中發生的異常終止。

[第 101 頁的『在 IBM MQ for z/OS 上產生的診斷資訊』](#)

請利用這個主題來調查 z/OS 所產生的部分診斷資訊，這些資訊有助於判斷問題，並瞭解如何調查錯誤訊息、傾出、主控台日誌、工作輸出、症狀字串及佇列輸出。

z/OS 使用 IBM MQ for z/OS 取得傾出

請利用這個主題來瞭解 IBM MQ for z/OS 問題判斷的不同傾出類型。

下表顯示與 IBM MQ for z/OS 搭配使用的傾出類型及其起始方式的相關資訊。它也會顯示如何格式化傾出：

傾出類型	資料集	輸出類型	格式化方式	原因
SVC	由系統定義	機器可讀	IPCS 與 IBM MQ for z/OS 動詞結束程式一起使用	z/OS 或 IBM MQ for z/OS 功能回復常式偵測錯誤，或操作員輸入 z/OS DUMP 指令
SYSUDUMP	由 JCL 定義 (SYSOUT = A)	已格式化	正常 SYSOUT = A	異常終止狀況 (只有在步驟有 SYSUDUMP DD 陳述式時才會採取)
貼齊	由 JCL CSQSNAP 定義 (SYSOUT = A)	已格式化	正常 SYSOUT = A	向配接卡報告的非預期 MQI 呼叫錯誤，或通道起始程式的 FFST 資訊
獨立式	由安裝 (磁帶或磁碟) 定義	機器可讀	IPCS 與 IBM MQ for z/OS 動詞結束程式一起使用	獨立式傾出程式的操作員 IPL

IBM MQ for z/OS 回復常式會針對大部分 X'5C6' 異常終止要求 SVC 傾出。異常狀況列在 [第 121 頁的『z/OS 上的 SVC 傾出』](#) 中。IBM MQ for z/OS 發出的 SVC 傾出是問題診斷資訊的主要來源。

如果傾出是由 IBM MQ 子系統起始，則傾出的相關資訊會放入稱為摘要部分的區域中。這包含傾出格式化程式可用來識別主要元件的資訊。

如需 SVC 傾出的相關資訊，請參閱 *z/OS MVS Diagnosis: Tools and Service Aids* 手冊。

z/OS 使用 z/OS DUMP 指令

若要解決問題，IBM 可以要求您建立佇列管理程式位址空間、通道起始程式位址空間或連結機能結構的傾出檔。請利用這個主題來瞭解建立這些傾出檔的指令。

可能會要求您為 IBM 的下列任何或數個項目建立傾出檔，以解決問題：

- 主要 IBM MQ 位址空間
- 通道起始程式位址空間
- 連結機能應用程式結構
- 佇列共用群組的連結機能管理結構

第 106 頁的圖 8 到 第 107 頁的圖 12 顯示要執行此動作的 z/OS 指令範例，並假設子系統名稱為 CSQ1。

```
DUMP COMM=(MQ QUEUE MANAGER DUMP)
*01 IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
R 01, JOBNAME=(CSQ1MSTR, BATCH), CONT
*02 IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
IEE600I REPLY TO 01 IS;JOBNAME=CSQ1MSTR,CONT
R 02, SDATA=(CSA, RGN, PSA, SQA, LSQA, TRT, SUM), END
IEE600I REPLY TO 02 IS;SDATA=(CSA, RGN, PSA, SQA, LSQA, TRT, SUM), END
IEA794I SVC DUMP HAS CAPTURED: 869
DUMPID=001 REQUESTED BY JOB (*MASTER*)
DUMP TITLE=MQ QUEUE MANAGER MAIN DUMP
```

圖 8: 傾出 IBM MQ 佇列管理程式及應用程式位址空間

```
DUMP COMM=(MQ QUEUE MANAGER DUMP)
*01 IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
R 01, JOBNAME=(CSQ1MSTR), CONT
*02 IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
IEE600I REPLY TO 01 IS;JOBNAME=CSQ1MSTR,CONT
R 02, SDATA=(CSA, RGN, PSA, SQA, LSQA, TRT, SUM), END
IEE600I REPLY TO 02 IS;SDATA=(CSA, RGN, PSA, SQA, LSQA, TRT, SUM), END
IEA794I SVC DUMP HAS CAPTURED: 869
DUMPID=001 REQUESTED BY JOB (*MASTER*)
DUMP TITLE=MQ QUEUE MANAGER DUMP
```

圖 9: 傾出 IBM MQ 佇列管理程式位址空間

```
DUMP COMM=(MQ CHIN DUMP)
*01 IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
R 01, JOBNAME=CSQ1CHIN, CONT
*02 IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
IEE600I REPLY TO 01 IS;JOBNAME=CSQ1CHIN,CONT
R 02, SDATA=(CSA, RGN, PSA, SQA, LSQA, TRT, SUM), CONT
*03 IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
IEE600I REPLY TO 02 IS;SDATA=(CSA, RGN, PSA, SQA, LSQA, TRT, SUM), CONT
R 03, DSPNAME=('CSQ1CHIN'.CSQXTRDS), END
IEE600I REPLY TO 03 IS;DSPNAME='CSQ1CHIN'.CSQXTRDS,END
IEA794I SVC DUMP HAS CAPTURED: 869
DUMPID=001 REQUESTED BY JOB (*MASTER*)
DUMP TITLE=MQ CHIN DUMP
```

圖 10: 傾出通道起始程式位址空間

```
DUMP COMM=(MQ MSTR & CHIN DUMP)
*01 IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
R 01, JOBNAME=(CSQ1MSTR, CSQ1CHIN), CONT
*02 IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
IEE600I REPLY TO 01 IS;JOBNAME=(CSQ1MSTR, CSQ1CHIN), CONT
R 02, SDATA=(CSA, RGN, PSA, SQA, LSQA, TRT, SUM), CONT
*03 IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
IEE600I REPLY TO 02 IS;SDATA=(CSA, RGN, PSA, SQA, LSQA, TRT, SUM), CONT
R 03, DSPNAME=('CSQ1CHIN'.CSQXTRDS), END
IEE600I REPLY TO 03 IS;DSPNAME=('CSQ1CHIN'.CSQXTRDS),END
IEA794I SVC DUMP HAS CAPTURED: 869
DUMPID=001 REQUESTED BY JOB (*MASTER*)
DUMP TITLE=MQ MSTR & CHIN DUMP
```

圖 11: 傾出 IBM MQ 佇列管理程式及通道起始程式位址空間

```

DUMP COMM=('MQ APPLICATION STRUCTURE 1 DUMP')
01 IEE094D SPECIFY OPERAND(S) FOR DUMP COMMAND
R 01,STRLIST=(STRNAME=QSG1APPLICATION1,(LISTNUM=ALL,ADJUNCT=CAPTURE,ENTRYDATA=UNSER))
IEE600I REPLY TO 01 IS;STRLIST=(STRNAME=QSG1APPLICATION1,(LISTNUM=
IEA794I SVC DUMP HAS CAPTURED: 677
DUMPID=057 REQUESTED BY JOB (*MASTER*)
DUMP TITLE='MQ APPLICATION STRUCTURE 1 DUMP'

```

圖 12: 傾出連結機能結構

z/OS

使用 IBM MQ for z/OS 傾出顯示畫面處理傾出

您可以使用透過 IPCS 畫面提供的指令來處理傾出。請利用這個主題來瞭解 IPCS 選項。

IBM MQ for z/OS 提供一組畫面來協助您處理傾出。下一節說明如何使用這些畫面：

1. 從 IPCS 主要選項功能表中，選取 **ANALYSIS-分析傾出內容** (選項 2)。

即會顯示 IPCS MVS ANALYSIS OF DUMP CONTENTS 畫面。

2. 選取 **COMPONENT-MVS 元件資料** (選項 6)。

即會顯示「IPCS MVS 傾出元件資料分析」畫面。畫面的外觀取決於安裝時所安裝的產品，但將類似於 [IPCS MVS 傾出元件資料分析畫面](#) 中顯示的畫面：

```

----- IPCS MVS DUMP COMPONENT DATA ANALYSIS -----
OPTION ==>                                SCROLL ==

To display information, specify "S option name" or enter S to the
left of the option required. Enter ? to the left of an option to
display help regarding the component support.

Name      Abstract
ALCWAIT   Allocation wait summary
AOMDATA   AOM analysis
ASMCHECK  Auxiliary storage paging activity
ASMDATA   ASM control block analysis
AVMDATA   AVM control block analysis
COMCHECK  Operator communications data
CSQMAIN   WebSphere MQ dump formatter panel interface
CSQWDMP   WebSphere MQ dump formatter
CTRACE    Component trace summary
DAEDATA   DAE header data
DIVDATA   Data-in-virtual storage

```

圖 13: IPCS MVS 傾出元件資料分析畫面

3. 在行旁鍵入 s，然後按 Enter 鍵，以選取 **CSQMAIN IBM MQ 傾出格式製作程式畫面** 介面。

如果無法使用此選項，則是因為成員 CSQ7IPCS 不存在；您應該參閱 [配置 z/OS](#)，以取得安裝 IBM MQ for z/OS 傾出格式化成員的相關資訊。

註：如果您已使用傾出來執行初步分析，且想要重新檢查它，請選取 **CSQWDMP IBM MQ 傾出格式製作程式**，以使用預設選項重新顯示格式化內容。

4. 即會顯示 IBM MQ for z/OS -DUMP ANALYSIS 功能表。使用此功能表來指定您要對系統傾出執行的動作。

```

-----IBM WebSphere MQ for z/OS - DUMP ANALYSIS-----
COMMAND ==>

    1 Display all dump titles 00 through 99
    2 Manage the dump inventory
    3 Select a dump

    4 Display address spaces active at time of dump
    5 Display the symptom string
    6 Display the symptom string and other related data
    7 Display LOGREC data from the buffer in the dump
    8 Format and display the dump

    9 Issue IPCS command or CLIST

(c) Copyright IBM Corporation 1993, 2023. All rights reserved.

    F1=Help    F3=Exit    F12=Cancel

```

5. 在您可以選取特定傾出進行分析之前，您需要的傾出必須存在於傾出庫存中。若要確保如此，請執行下列步驟：
 - a. 如果您不知道包含傾出的資料集名稱，請指定選項 1- **顯示所有傾出標題 xx 到 xx**。
 這會顯示 SYS1.DUMP 資料集 (其中 xx 是 00 到 99 範圍內的數字)。您可以使用 xx 欄位來指定資料集號碼範圍，以限制選取要顯示的資料集。
 如果您想要查看所有可用傾出資料集的詳細資料，請將這些值設為 00 及 99。
 使用顯示的資訊來識別您要分析的傾出。
 - b. 如果傾出尚未複製到另一個資料集 (亦即，它位於其中一個 SYS1.DUMP 資料集)，指定選項 2- **管理傾出庫存**
 傾出庫存包含您已使用的傾出資料集。因為 SYS1.DUMP 資料集，您在步驟 第 108 頁的『5.a』中所識別的傾出名稱可能在顯示的清單中。不過，此項目會參照儲存在此資料集中的前一個傾出，因此請在其旁邊鍵入 DD 並按 Enter 鍵來刪除它。然後按 F3 以回到「傾出分析功能表」。
6. 指定選項 3- **選取傾出**，以選取您要使用的傾出。在「來源」欄位中鍵入包含傾出的資料集名稱，檢查是否在「訊息遞送」欄位中指定 NOPRINT 及 TERMINAL (這是為了確保輸出導向至終端機)，然後按 Enter 鍵。按 F3 以回到「傾出分析功能表」。
7. 選取要使用的傾出之後，您現在可以使用功能表上的其他選項來分析傾出不同部分中的資料：
 - 若要顯示在執行傾出時所有作用中位址空間的清單，請選取選項 4。
 - 若要顯示症狀字串，請選取選項 5。
 - 若要顯示症狀字串及其他服務功能資訊，包括系統診斷工作區 (SDWA) 的變數記錄區域，請選取選項 6。
 - 若要格式化並顯示儲存體內 LOGREC 緩衝區中包含的資料，請選取選項 7。

可能是造成傾出的異常終止不是錯誤的原始原因，而是先前的問題所造成。若要判斷與問題原因相關的 LOGREC 記錄，請跳至資料集結尾，鍵入 FIND ERRORID: PREV，然後按 Enter 鍵。會顯示最新 LOGREC 記錄的標頭，例如：

```

JOBNAME: NONE-FRR
ERRORID: SEQ=00081 CPU=0040 ASID=0033 TIME=14:42:47.1

SEARCH ARGUMENT ABSTRACT

PIDS/5655R3600 RIDS/CSQRLLM1#L RIDS/CSQRRHSL AB/S05C6
PRCS/00D10231 REGS/0C1F0 RIDS/CSQVEUS2#R

SYMPTOM          DESCRIPTION
-----
PIDS/5655R3600  PROGRAM ID: 5655R3600
.
.
.

```

請注意程式 ID (如果不是 5655R3600, 則問題不是由 IBM MQ for z/OS 所造成, 而且您可能看到錯誤的傾出)。另請注意 TIME 欄位的值。重複指令以尋找先前的 LOGREC 記錄, 並重新記下 TIME 欄位的值。如果兩個值彼此接近 (例如, 在大約十分之一秒或二分之一秒內), 它們可能都與相同的問題相關。

- 若要格式化並顯示傾出, 請選取選項 8。即會顯示 FORMAT AND DISPLAY THE DUMP 畫面:

```

-----IBM MQ for z/OS - FORMAT AND DISPLAY DUMP-----
COMMAND ==>

1 Display the control blocks and trace
2 Display just the control blocks
3 Display just the trace

Options:

Use the summary dump? . . . . . __ 1 Yes
2 No

Subsystem name (required if summary dump not used) ____

Address space identifier or ALL. . . . . ALL_

F1=Help F3=Exit F12=Cancel

```

- 請利用這個畫面來格式化您選取的系統傾出。您可以選擇顯示控制區塊及/或內部追蹤所產生的資料, 這是預設值。

註: 您無法對通道起始程式的傾出或連結機能結構的傾出執行此動作。

- 顯示整個傾出, 亦即:
 - 傾出標題
 - 變數記錄區域 (VRA) 診斷資訊報告
 - 儲存區域追蹤報告
 - 控制區塊摘要
 - 追蹤表格
- 選取選項 1。
- 若要顯示針對選項 1 所列出的資訊, 而不顯示追蹤表格, 請選取選項 2。
- 若要在沒有控制區塊的情況下顯示選項 1 所列出的資訊, 請選取選項 3。

您也可以使用下列選項:

- 使用摘要傾出?

請利用這個欄位來指定在格式化選取的傾出時，是否要 IBM MQ 使用摘要部分所包含的資訊。預設值為 YES。

註: 如果已取得摘要傾出，則可能包括來自多個位址空間的資料。

– 子系統名稱

使用此欄位來識別具有您要顯示之傾出資料的子系統。只有在沒有摘要資料 (例如，如果操作員要求傾出)，或您在 **使用摘要傾出?** 中指定「否」時，才需要此選項。欄位。

如果您不知道子系統名稱，請在命令提示字元中鍵入 `IPCS SELECT ALL`，然後按 Enter 鍵以顯示錯誤發生時執行的所有工作清單。如果其中一個工作在 **SELECTION CRITERIA** 直欄中具有針對它的單字 **ERROR**，請記下該工作的名稱。工作名稱的格式為 `xxxx MSTR`，其中 `xxxx` 是子系統名稱。

```
IPCS OUTPUT STREAM -----
COMMAND ==>
ASID JOBNAME ASCBADDR SELECTION CRITERIA
-----
0001 *MASTER* 00FD4D80 ALL
0002 PCAUTH 00F8AB80 ALL
0003 RASP 00F8C100 ALL
0004 TRACE 00F8BE00 ALL
0005 GRS 00F8BC00 ALL
0006 DUMPSRV 00F8DE00 ALL
0008 CONSOLE 00FA7E00 ALL
0009 ALLOCAS 00F8D780 ALL
000A SMF 00FA4A00 ALL
000B VLF 00FA4800 ALL
000C LLA 00FA4600 ALL
000D JESM 00F71E00 ALL
001F MQM1MSTR 00FA0680 ERROR ALL
```

如果沒有工作在 **SELECTION CRITERIA** 直欄中具有單字 **ERROR**，請在主要「IPCS 選項功能表」畫面上選取選項 **0-DEFAULTS**，以顯示「IPCS 預設值」畫面。請記下位址空間 ID (ASID)，然後按 **F3** 回到前一個畫面。使用 ASID 來判斷工作名稱; 格式為 `xxxx MSTR`，其中 `xxxx` 是子系統名稱。

下列指令顯示傾出資料集中的 ASID:

```
LDMP DSN('SYS1.DUMPxx') SELECT(DUMPED) NOSUMMARY
```

這會顯示針對每一個位址空間傾出的儲存體範圍。

按 **F3** 以回到「格式及顯示傾出」畫面，並在 **子系統名稱** 欄位中鍵入此名稱。

– 位址空間 ID

如果傾出中的資料來自多個位址空間，請使用此欄位。如果您只想查看來自特定位址空間的資料，請指定該位址空間的 ID (ASID)。

此欄位的預設值為 **ALL**，它會顯示傾出中與子系統相關的所有位址空間的相關資訊。請在顯示的值上鍵入 4 個字元的 ASID 來變更此欄位。

註: 因為傾出包含所有位址空間共用的儲存區，所以如果您不正確地指定位址空間 ID，則顯示的資訊可能與您的問題無關。在此情況下，請回到此畫面，並輸入正確的位址空間 ID。

相關概念

[第 111 頁的『使用線路模式 IPCS 處理 IBM MQ for z/OS 傾出』](#)
使用 IPCS 指令來格式化傾出。

[第 117 頁的『以批次方式使用 IPCS 處理 IBM MQ for z/OS 傾出』](#)
請利用這個主題來瞭解 IPCS 指令如何以批次模式來格式化 IBM MQ for z/OS 傾出。

[第 118 頁的『分析 z/OS 上的傾出及解譯傾出標題』](#)
請利用這個主題來瞭解如何格式化 IBM MQ for z/OS 傾出標題，以及如何分析傾出。

z/OS 使用線路模式 IPCS 處理 IBM MQ for z/OS 傾出

使用 IPCS 指令來格式化傾出。

若要使用線路模式 IPCS 指令來格式化傾出，請發出下列指令來選取所需的傾出：

```
SETDEF DSN('SYS1.DUMP xx')
```

(其中 SYS1.DUMP *xx* 是包含傾出的資料集名稱)。然後，您可以使用 IPCS 次指令來顯示傾出中的資料。

如需如何使用 IPCS 指令來格式化不同類型傾出的相關資訊，請參閱下列主題：

- 第 111 頁的『[格式化 IBM MQ for z/OS 傾出](#)』
- 第 116 頁的『[在 z/OS 上從通道起始程式格式化傾出](#)』

相關概念

第 107 頁的『[使用 IBM MQ for z/OS 傾出顯示畫面處理傾出](#)』

您可以使用透過 IPCS 畫面提供的指令來處理傾出。請利用這個主題來瞭解 IPCS 選項。

第 117 頁的『[以批次方式使用 IPCS 處理 IBM MQ for z/OS 傾出](#)』

請利用這個主題來瞭解 IPCS 指令如何以批次模式來格式化 IBM MQ for z/OS 傾出。

第 118 頁的『[分析 z/OS 上的傾出及解譯傾出標題](#)』

請利用這個主題來瞭解如何格式化 IBM MQ for z/OS 傾出標題，以及如何分析傾出。

z/OS 格式化 IBM MQ for z/OS 傾出

使用本主題可瞭解如何使用線路模式 IPCS 指令來格式化佇列管理程式傾出。

IPCS VERBEXIT CSQWDMP 會呼叫 IBM MQ for z/OS 傾出格式化程式 (CSQWDPD)，並可讓您格式化 SVC 傾出以顯示 IBM MQ 資料。您可以透過指定參數來限制顯示的資料量。

IBM「服務人員」可能需要佇列共用群組的連結機能管理結構及應用程式結構傾出，以及佇列共用群組中的佇列管理程式傾出，以協助診斷問題。如需格式化連結機能清單結構及 STRDATA 次指令的相關資訊，請參閱 [z/OS MVS IPCS 指令手冊](#)。

註：本節說明擷取必要資料所需的參數。以逗點區隔運算元，而不是空白。控制陳述式中任何運算元後面的空白會終止運算元清單，且會忽略任何後續的運算元。第 111 頁的表 12 說明您可以在控制陳述式中指定用來格式化傾出的每一個關鍵字。

關鍵字	說明
SUBSYS= <i>aaaa</i>	如果摘要傾出部分無法使用或無法使用，請使用此關鍵字來提供子系統名稱以格式化資訊。 <i>aaaa</i> 是一個 1 到 4 個字元的子系統名稱。
全部 (預設值)	所有控制區塊及追蹤表格。
AA	顯示所有位址空間中所有 IBM MQ for z/OS 控制區塊的資料。
DIAG=Y	列印診斷資訊。僅在 IBM 服務人員的指引下使用。DIAG=N (抑制診斷資訊的格式化) 是預設值。
EB= <i>nnnnnnnn</i>	只會顯示與此 EB 執行緒相關聯的追蹤點 (此關鍵字的格式為 EB= <i>nnnnnnnn</i> ，其中 <i>nnnnnnnn</i> 是追蹤所包含之 EB 執行緒的 8 位數位址)。這必須與 TT 關鍵字一起使用。
lg	所有控制區塊。
PTF=Y，LOAD= 載入模組名稱	報告前端的 PTF 清單 (來自 MEPL)。PTF=N (抑制這類清單的格式化) 是預設值。 選用載入子參數可讓您指定要格式化 PTF 報告的載入模組名稱 (最多 8 個字元)。

表 12: IBM MQ for z/OS dump 格式化控制陳述式的關鍵字 (繼續)

關鍵字	說明
SA= hhhh	指定位址空間的控制區塊。請使用下列其中一種格式： <ul style="list-style-type: none"> SA= hh 或 SA= hhhh 其中 h 代表十六進位數字。
SG	全系統控制區塊的子集。
TT , HANDLES=x , LOCKS=x , INSYNCS=x , URINFO=ALL/LON	格式化追蹤表格 指出具有大於 x 個控點的執行緒 指出具有大於 x 個鎖定的執行緒 指出具有大於 x 個不同步作業的執行緒 顯示所有執行緒或長時間執行執行緒的 UR 資訊

第 112 頁的表 13 詳述傾出格式化關鍵字，您可以使用這些關鍵字來格式化個別資源管理程式的相關資料。這些關鍵字不能與第 111 頁的表 12 中的任何關鍵字一起使用。

表 13: 資源管理程式傾出格式化關鍵字

關鍵字	格式化內容
BMC=1	緩衝區管理程式資料。BMC=1 會格式化所有緩衝區的控制區塊。
BMC=2(緩衝池號碼)	BMC=2 格式與 2 位數 緩衝池號碼中所識別之緩衝區相關的資料。
BMC=3(xx/yyyyyy)	如果頁面呈現在緩衝區中，則 BMC=3 及 BMC=4 會顯示頁集的頁面。 (BMC=3 與 BMC=4 之間的差異是通往頁面的路徑。)
BMC=4(xx/yyyyyy)	
BUFL= nnnnnnnnnnn	儲存體存取緩衝區配置 sz。
CALLD=Y =W	在 TT 中顯示呼叫深度的箭頭。 及縮排追蹤項目。
CALLTIME=Y	結束追蹤時列印呼叫時間。
CB = (addr/[strmodel])	將位址格式化為 IBM MQ 區塊。
CBF=1	CBF 報告第 1 級。
CCB=S	在 TT 中顯示系統 EB 的「複合功能區塊 (CCB)」。
CFS=1	CFS 報告層次 1。
CFS=2	CFS 報告層次 2。
CHLAUTH=1/2 ONAM=20 chars	CHLAUTH 報告層次。 選用 ONAM 子參數可讓您指定物件名稱 (最多 20 個字元)，以將列印資料限制在 ONAM 中以字元開頭的物件。
CLUS=1	叢集報告，包括佇列管理程式上已知的叢集儲存庫。
CLUS=2	顯示叢集登錄的叢集報告。
CLXQ=1	叢集 XMITQ 報告層次 1。

表 13: 資源管理程式傾出格式化關鍵字 (繼續)	
關鍵字	格式化內容
CLXQ=2 ONAM=20 chars	叢集 XMITQ 報告層次 2。 選用 ONAM 子參數可讓您指定物件名稱 (最多 20 個字元), 以將列印資料限制在 ONAM 中以字元開頭的物件。
CMD=0/1/2	指令追蹤表格顯示層次。
D=1/2/3	部分報告的詳細層次。
Db2=1	Db2 報告層次 1。
DMC=1, ONAM=48 chars	DMC 報告層次 1。 選用 ONAM 子參數可讓您指定物件名稱 (最多 48 個字元), 以將列印資料限制在 ONAM 中以字元開頭的物件。
DMC=2, ONAM=48 chars	DMC 報告層次 2。 選用的 ONAM 子參數可讓您將列印物件限制為名稱以 ONAM 中指定字元開頭的物件 (最多 48 個字元)。
DMC=3, ONAM=48 chars	DMC 報告層次 3。 選用的 ONAM 子參數可讓您將列印物件限制為名稱以 ONAM 中指定字元開頭的物件 (最多 48 個字元)。
GR=1	群組不確定報告層次 1。
IMS=1	IMS 報告層次 1

表 14: 資源管理程式傾出格式化關鍵字 (J -P)	
關鍵字	格式化內容
JOBNAME= xxxxxxxx	工作名稱
LKM=1	LKM 報告層次 1。
LKM=2/3, , NAME=最多 48 個字元 , NAMEX= xxxxxxxxxxxxxx , NAMESP=1/2/3/4/5/6/7/8 , TYPE=DMCP/QUALNM/TOPIC/ STGCLASS , QUAL=GET/PUT/CRE/DFXQ/ PGSYNC/CHGCNT/ 刪除/到期 LKM=3 LKM=4 , JOBNAME= xxxxxxxx , ASID= xxxx	LKM 報告層次 2/3。 名稱 (字元) 名稱 (十六進位) 名稱空間 鎖定類型 鎖定資格 LKM 報告層次 3 LKM 報告層次 4
LMC=1	LMC 報告層次 1。

表 14: 資源管理程式傾出格式化關鍵字 (J -P) (繼續)	
關鍵字	格式化內容
MAXTR= nnnnnnnnn	要格式化的追蹤項目數上限
MHASID= xxxx	內容的訊息控點 ASID
MMC=1 OBJ=MQLO/MQSH/MQRO/ MQAO/MQMO/MCHL/ MNLS/MSTC/MPRC/: ' MAUT ONAM	MMC 報告層次 1 物件類型 選用的 ONAM 子參數可讓您將列印物件限制為名稱以 ONAM 中指定字元開頭的物件 (最多 48 個字元)。
MMC=2 ONAM=48 chars	MMC 報告層次 2 選用的 ONAM 子參數可讓您將列印物件限制為名稱以 ONAM 中指定字元開頭的物件 (最多 48 個字元)。
MSG=nnnnnnnnnnnnnnnn MASID=xxxx LEN=xxxxxxxx MSGD=S/D	格式化指標處的訊息。 MASID 容許儲存在其他位址空間中。 LEN 將儲存體數量限制為格式化。 MSGD 控制詳細程度。
MSGD=S/D	DMC=3、BMC=3/4、PSID 報告中的訊息詳細資料。 參數控制詳細資料的層次，S 是摘要，D 是詳細資料。
MSGH = nnnnnnnnnnnnnnn	訊息控點 (message handle)
MT	訊息內容追蹤
MQVCX	十六進位格式的 MQCHARV
PROPS= nnnnnnnnnnnnnnn	訊息內容指標
DSID= nnnnnnnn	頁集至格式化頁面
PSTRX	十六進位格式的內容字串

表 15: 資源管理程式傾出格式化關鍵字 (R -Z)	
關鍵字	格式化內容
RPR= nnnnnnnn	要格式化的頁面或記錄
顯示	顯示 DMC=3 的已刪除記錄
SMC=1/2/3	儲存體管理程式
TC= * A E O	TT 資料字元格式，連結 以適當的字集列印全部 一律列印 ASCII 一律列印 EBCDIC 也不列印
TFMT = H/M	時間格式-人工或 STCK

表 15: 資源管理程式傾出格式化關鍵字 (R -Z) (繼續)	
關鍵字	格式化內容
THR= nnnnnnnn	執行緒位址
THR=*/2/3	設定執行緒報告層次
TOP=1	TOP 報告層次 1
TOP=2	TOP 報告層次 2
TOP= nnnnnnnnnnnnnnnn /TSTR=48 個字元 /TSTRX=hex 1208 str	Tnode 64bit 位址或 主題字串 (開頭或結尾為% 的萬用字元) ' 這會將 EBCDIC 轉換成 ASCII , 但只會有不變字元 1208 中主題字串的十六進位一律在開頭使用萬用字元。
TOP=3	TOP 報告層次 3
TOP=4	TOP 報告層次 4
TSEG=M (RU) /Q (P64) I (NTERPOLATE) F (WD) D (EBUG)	64 位元追蹤的搜尋處理程序 猜測遺漏一或多個 TSEG 位址 強制向前排序 除錯搜尋程序
TSEG = (M , Q , I , F , D)	指定多個 TSEG 選項
W=0/1/2/3	TT 寬度格式
XA=1	XA 報告層次 1
ZMH = nnnnnnnnnnnnnnnn	ZST 訊息控點

如果傾出是由操作員起始，則傾出的摘要部分中沒有資訊。第 115 頁的表 16 顯示您可以在 CSQWDMP 控制陳述式中使用的其他關鍵字。

表 16: IBM MQ for z/OS dump 格式化控制陳述式的摘要傾出關鍵字	
關鍵字	說明
SUBSYS= aaaa	如果摘要傾出部分無法使用或無法使用，請使用此關鍵字來提供子系統名稱以格式化資訊。aaaa 是一個 1 到 4 個字元的子系統名稱。
SumDUMP=NO	如果傾出有摘要部分，但您不想使用它，請使用這個關鍵字。(通常只有在 IBM 支援中心的指示下，您才會這麼做。)

下列清單顯示如何使用這些關鍵字的一些範例:

- 對於所有位址空間的預設格式，請使用傾出摘要部分中的資訊，使用:

```
VERBX CSQWDMP
```

- 若要從名為 MQMT 的子系統傾出顯示追蹤表格，該子系統由操作員起始 (因此沒有摘要部分)，請使用:

```
VERBX CSQWDMP 'TT,SUBSYS=MQMT'
```

- 若要針對 ASID (位址空間 ID) 為 1F 的位址空間顯示子系統異常終止所產生傾出的所有控制區塊及追蹤表格，請使用:

```
VERBX CSQWDMP 'TT, LG, SA=1F'
```

- 若要從與特定 EB 執行緒相關聯的傾出顯示追蹤表格的部分，請使用：

```
VERBX CSQWDMP 'TT, EB= nnnnnnnn '
```

- 如果要顯示名稱以 'ABC' 開頭的本端非共用佇列物件的訊息管理程式 1 報告，請執行下列動作：

```
VERBX CSQWDMP 'MMC=1, ONAM=ABC, Obj=MQLO'
```

第 116 頁的表 17 顯示一些經常用來分析傾出的其他指令。如需這些次指令的相關資訊，請參閱 [z/OS MVS IPCS 指令手冊](#)。

表 17: 用於傾出分析的 IPCS 次指令	
次指令	說明
STATUS	顯示通常在問題判斷程序的起始部分期間檢查的資料。
STRDATA LISTNUM (ALL) ENTRYPOS (ALL) DETAIL	格式化連結機能結構資料。
Verbexit LOGDATA	在取得傾出之前，將呈現的儲存體內 LOGREC 緩衝區記錄格式化。LOGDATA 會尋找 LOGREC 記錄緩衝區中包含的 LOGREC 項目，並呼叫 EREP 程式來格式化及列印 LOGREC 項目。這些項目會以一般明細編輯報告的樣式來格式化。
Verbexit 追蹤	格式化所有位址空間的系統追蹤登錄。
VERBEXIT SYMPTOM	格式化系統傾出 (例如獨立式傾出、SVC 傾出或 SYSUDUMP DD 陳述式所要求的異常終止傾出) 的標頭記錄中所包含的症狀字串。
Verbexit GRSTRACE	格式化來自廣域資源序列化主要控制區塊的診斷資料。
VERBEXIT SUMDUMP	尋找並顯示 SVC 傾出提供的摘要傾出資料。
VERBEXIT DAEDATA	將傾出系統的傾出分析及排除 (DAE) 資料格式化。

相關概念

第 116 頁的『在 z/OS 上從通道起始程式格式化傾出』

使用本主題可瞭解如何使用線路模式 IPCS 指令來格式化 IBM MQ for z/OS 的通道起始程式傾出。

在 z/OS 上從通道起始程式格式化傾出

使用本主題可瞭解如何使用線路模式 IPCS 指令來格式化 IBM MQ for z/OS 的通道起始程式傾出。

IPCS VERBEXIT CSQXDPRD 可讓您格式化通道起始程式傾出。您可以指定關鍵字來選取格式化的資料。

本節說明您可以指定的關鍵字。

第 116 頁的表 18 說明您可以使用 CSQXDPRD 指定的關鍵字。

表 18: IPCS VERBEXIT CSQXDPRD 的關鍵字	
關鍵字	格式化內容
SUBSYS= aaaa	與具名子系統相關聯之通道起始程式的控制區塊。所有新的格式化傾出都需要它。

表 18: IPCS VERBEXIT CSQXDPRD 的關鍵字 (繼續)

關鍵字	格式化內容
CHST=1, CNAM= 通道名稱, DUMP=S F C	所有通道資訊。 選用的 CNAM 子參數可讓您指定通道名稱 (最多 20 個字元), 以格式化其詳細資料。 選用 DUMP 子參數可讓您控制格式化的範圍, 如下所示: <ul style="list-style-type: none"> 指定 DUMP=S (代表 "short"), 以格式化通道資料十六進位傾出的第一行。 指定 DUMP=F (代表 "full") 以格式化所有資料行。 指定 DUMP=C (針對 "compressedsmaller"), 以暫停格式化只包含 X'00' 的資料中所有重複行。這是預設選項
CHST=2, CNAM= 通道名稱,	所有通道或 CNAM 關鍵字指定之通道的摘要。 如需 CNAM 子參數的詳細資料, 請參閱 CHST=1。
CHST=3, CNAM= 通道名稱,	CHST=2 所提供的資料, 以及傾出中所有通道的程式追蹤、行追蹤及格式化號誌表格列印。 如需 CNAM 子參數的詳細資料, 請參閱 CHST=1。
CLUS=1	叢集報告, 包括佇列管理程式上已知的叢集儲存庫。
CLUS=2	顯示叢集登錄的叢集報告。
CTRACE=S F, DPRO= nnnnnnnn、 TCB= nnnnnnn	選取短 (CTRACE = S) 或完整 (CTRACE = F) CTRACE。 選用性 DPRO 子參數可讓您對指定的 DPRO 指定 CTRACE。 選用 TCB 子參數可讓您為指定的工作指定 CTRACE。
DISP=1, DUMP=S F C	分派器報告 如需 DUMP 子參數的詳細資料, 請參閱 CHST=1。
BUF=1	緩衝區報告
XSMF=1	格式化傾出中可用的通道起始程式 SMF 資料。

相關概念

第 111 頁的『格式化 IBM MQ for z/OS 傾出』

使用本主題可瞭解如何使用線路模式 IPCS 指令來格式化佇列管理程式傾出。

以批次方式使用 IPCS 處理 IBM MQ for z/OS 傾出

請利用這個主題來瞭解 IPCS 指令如何以批次模式來格式化 IBM MQ for z/OS 傾出。

若要批次使用 IPCS, 請將必要的 IPCS 陳述式插入批次工作串流中 (請參閱 第 118 頁的圖 14)。

變更 DUMP00 陳述式上的資料集名稱 (DSN =), 以反映您要處理的傾出, 並插入您要使用的 IPCS 次指令。


```

//*****
//*  RUNNING IPCS IN A BATCH JOB          *
//*****
//MQMDMP EXEC PGM=IKJEFT01,REGION=5120K
//STEPLIB DD DSN=mqm.library-name,DISP=SHR
//SYSTSPRT DD SYSOUT=*
//IPCSPRNT DD SYSOUT=*
//IPCSDDIR DD DSN=dump.directory-name,DISP=OLD
//DUMP00 DD DSN=dump.name,DISP=SHR
//SYSTSIN DD *
IPCS NOPARM TASKLIB(SCSQLOAD)
SETDEF PRINT TERMINAL DDNAME(DUMP00) NOCONFIRM
*****
* INSERT YOUR IPCS COMMANDS HERE, FOR EXAMPLE: *
VERBEXIT LOGDATA
VERBEXIT SYMPTOM
VERBEXIT CSQWDMP 'TT,SUBSYS=QMGR'
*****

CLOSE ALL
END
/*

```

圖 14: 在 z/OS 環境中透過 IPCS 列印傾出的範例 JCL

相關概念

第 107 頁的『[使用 IBM MQ for z/OS 傾出顯示畫面處理傾出](#)』

您可以使用透過 IPCS 畫面提供的指令來處理傾出。請利用這個主題來瞭解 IPCS 選項。

第 111 頁的『[使用線路模式 IPCS 處理 IBM MQ for z/OS 傾出](#)』

使用 IPCS 指令來格式化傾出。

第 118 頁的『[分析 z/OS 上的傾出及解譯傾出標題](#)』

請利用這個主題來瞭解如何格式化 IBM MQ for z/OS 傾出標題，以及如何分析傾出。

z/OS 分析 z/OS 上的傾出及解譯傾出標題

請利用這個主題來瞭解如何格式化 IBM MQ for z/OS 傾出標題，以及如何分析傾出。

- [分析傾出](#)
- [使用 PSW 和 ASID 傾出標題變異](#)

分析傾出

傾出標題包括異常終止完成及原因碼、失敗載入模組及 CSECT 名稱，以及版次 ID。如需傾出標題的相關資訊，請參閱 [使用 PSW 及 ASID 傾出標題變異](#)

SVC 傾出標題的格式略有不同，視錯誤類型而定。

第 118 頁的圖 15 顯示 SVC 傾出標題的範例。圖例之後會說明標題中的每一個欄位。

```

ssnm,ABN=5C6-00D303F2,U=AUSER,C=R3600. 710.LOCK-CSQL1GET,
M=CSQGFRCV,LOC=CSQLPLM.CSQL1GET+0246

```

圖 15: SVC 傾出標題範例

ssnm,ABN=compltn-reason

- ssnm 是發出傾出的子系統名稱。
- compltn 是 3 個字元的十六進位異常終止完成碼 (在此範例中為 X'5C6')，以 U 作為使用者異常終止碼的字首。
- reason 是 4 位元組十六進位原因碼 (在此範例中為 X'00D303F2')。

註: 異常終止及原因碼可能提供足夠的資訊來解決問題。請參閱 [IBM MQ for z/OS 訊息、完成及原因碼](#)，以取得原因碼的說明。

U=userid

- `userid` 是使用者的使用者 ID (在此範例中為 `AUSER`)。通道起始程式沒有這個欄位。

C=compid.release.comp-function

- `compid` 是元件 ID 的最後 5 個字元。值 `R3600` 可唯一識別 IBM MQ for z/OS。
- `release` 是一個 3 位數代碼，指出 IBM MQ for z/OS 的版本、版次及修正層次 (在此範例中為 `710`)。
- `comp` 是異常終止時控制中元件的字首語 (在此範例中為 `LOCK`)。
- `function` 是異常終止時控制中的函數、巨集或常式名稱 (在此範例中為 `CSQL1GET`)。此欄位並非一律存在。

M=module

- `module` 是 FRR 或 ESTAE 回復常式的名稱 (在此範例中，是 `CSQGFRCV`)。此欄位並非一律存在。

註: 這不是發生異常終止的模組名稱; 由 LOC 提供。

LOC=loadmod.csect+csect_offset

- `loadmod` 是異常終止時控制中的載入模組名稱 (在此範例中為 `CSQLLPLM`)。如果不明，這可能以星號表示。
- `csect` 是異常終止時控制中的 CSECT 名稱 (在此範例中為 `CSQL1GET`)。
- `csect_offset` 是異常終止時失敗 CSECT 內的偏移 (在本例中為 `0246`)。

註: 如果服務已套用至此 CSECT，則 `csect_offset` 的值可能會有所不同，因此在建置關鍵字字串以搜尋 IBM 軟體支援中心資料庫時，請勿使用此值。

具有 PSW 和 ASID 的傾出標題變異

部分傾出標題會以 PSW (程式狀態字組) 及 ASID (位址空間 ID) 取代載入模組名稱、CSECT 名稱及 CSECT 偏移。第 119 頁的圖 16 說明此格式。

```
ssnm,ABN=compltn-reason,U=userid,C=compid.release.comp-function,  
M=module,PSW=psw_contents,ASID=address_space_id
```

圖 16: 具有 PSW 和 ASID 的傾出標題

psw_contents

- 發生錯誤時的 PSW (例如，`X'077C100000729F9C'`)。

address_space_id

- 異常終止時控制的位址空間 (例如 `X'0011 '`)。通道起始程式沒有這個欄位。

相關概念

第 107 頁的『[使用 IBM MQ for z/OS 傾出顯示畫面處理傾出](#)』

您可以使用透過 IPCS 畫面提供的指令來處理傾出。請利用這個主題來瞭解 IPCS 選項。

第 111 頁的『[使用線路模式 IPCS 處理 IBM MQ for z/OS 傾出](#)』

使用 IPCS 指令來格式化傾出。

第 117 頁的『[以批次方式使用 IPCS 處理 IBM MQ for z/OS 傾出](#)』

請利用這個主題來瞭解 IPCS 指令如何以批次模式來格式化 IBM MQ for z/OS 傾出。

z/OS 系統可以建立 SYSUDUMP，以作為問題判斷的一部分。本主題顯示範例 SYSUDUMP 輸出，並提供解譯 SYSUDUMP 之工具的參照。

SYSUDUMP 傾出提供有助於對批次和 TSO 應用程式進行除錯的資訊。如需 SYSUDUMP 傾出的相關資訊，請參閱 *z/OS MVS Diagnosis: Tools and Service Aids* 手冊。

第 120 頁的圖 17 顯示 SYSUDUMP 傾出開始的範例。

```

JOB MQMBXBA1  STEP TSUSER  TIME 102912  DATE 001019  ID = 000  CPUID = 632202333081
PAGE 00000001

COMPLETION CODE          SYSTEM = 0C1          REASON CODE = 00000001

PSW AT ENTRY TO ABEND  078D1000 000433FC          ILC 2  INTC 000D

PSW LOAD MODULE = BXBAAB01  ADDRESS = 000433FC  OFFSET = 0000A7F4

ASCB: 00F56400
+0000 ASCB..... ASCB      FWDP..... 00F60180  BWDP..... 0047800  CMSF..... 019D5A30
SVRB..... 008FE9E0
+0014 SYNC..... 00000D6F  IOSP..... 00000000  TNEW..... 00D18F0  CPUS..... 00000001
ASID..... 0066
+0026 R026..... 0000      LL5..... 00          HLHI..... 01          DPHI..... 00
DP..... 9D
+002C TRQP..... 80F5D381  LDA..... 7FF154E8  RSMF..... 00          R035..... 0000
TRQI..... 42
+0038 CSCB..... 00F4D048  TSB..... 00B61938  EJST..... 00000001  8C257E00

+0048 EWST..... 9CCDE747  76A09480          JSTL..... 00141A4  ECB..... 808FEF78
UBET..... 9CCDE740
.
.
.
ASSB: 01946600
+0000 ASSB..... ASSB      VAFN..... 00000000  EVST..... 00000000  00000000

+0010 VFAT..... 00000000  00000000          RSV..... 000      XMCC..... 0000
XMCT..... 00000000
+0020 VSC..... 00000000  NVSC..... 0000004C  ASRR..... 00000000  R02C..... 00000000
00000000 00000000
+0038          00000000  00000000

*** ADDRESS SPACE SWITCH EVENT MASK OFF (ASTESSEM = 0) ***

TCB: 008D18F0
+0000 RBP..... 008FE7D8  PIE..... 00000000  DEB..... 00B1530  TIO..... 008D4000
CMP..... 805C6000
+0014 TRN..... 40000000  MSS..... 7FFF7418  PKF..... 80          FLGS..... 01000000  00
+0022 LMP..... FF          DSP..... FE          LLS..... 00D1A88  JLB..... 00011F18
JPQ..... 00000000
+0030 GPR0-3... 00001000  008A4000  00000000  00000000
+0040 GPR4-7... 00FDC730  008A50C8  00000002  80E73F04
+0050 GPR8-11.. 81CC4360  008A6754  008A67B4  00000008

```

圖 17: SYSUDUMP 的開頭範例

Snap 傾出資料集由 z/OS JCL 指令陳述式控制。請利用這個主題來瞭解 CSQSNAP DD 陳述式。

Snap 傾出一律會傳送至 CSQSNAP DD 陳述式所定義的資料集。它們可以由配接卡或通道起始程式發出。

- 當佇列管理程式針對 MQI 呼叫傳回非預期的錯誤時，批次、CICS、IMS 或 RRS 配接器會發出 SNAP 傾出。會產生完整傾出，其中包含造成問題之程式的相關資訊。

若要產生 SNAP 傾出，CSQSNAP DD 陳述式必須位於批次應用程式 JCL、CICS JCL 或 IMS 相依區域 JCL 中。

- 在特定錯誤狀況下，通道起始程式會發出 SNAP 傾出，而不是系統傾出。傾出包含與錯誤相關的資訊。同時也會發出訊息 CSQX053E。

若要產生 SNAP 傾出，CSQSNAP DD 陳述式必須在通道起始程式啟動作業程序中。

z/OS 上的 SYS1.LOGREC 資訊

請利用這個主題來瞭解 z/OS SYS1.LOGREC 資訊可以協助判斷問題。

IBM MQ for z/OS 和 SYS1.LOGREC

SYS1.LOGREC 資料集記錄作業系統不同元件遇到的各種錯誤。如需使用 SYS1.LOGREC 記錄，請參閱 [z/OS MVS Diagnosis: Tools and Service Aids](#) 手冊。

IBM MQ for z/OS 回復常式會將系統診斷工作區 (SDWA) 中的資訊寫入 SYS1.LOGREC 資料集。多個 SYS1.LOGREC 項目，因為單一錯誤可能會發生兩次以上的重試或執行。

SYS1.LOGREC 項目可能會提供關於導致異常終止之事件的有價值歷程資訊。

尋找適用的 SYS1.LOGREC 資訊

取得 SYS1.LOGREC 清單：

- 請參閱 [z/OS MVS 診斷: 工具和服務輔助工具](#) 手冊中說明的 [EREP 選取參數](#)，以格式化 SYS1.LOGREC 資料集中的記錄。
- 在 IPCS 中指定 VERBEXIT LOGDATA 關鍵字。
- 使用 DUMP ANALYSIS MENU 上的選項 7 (請參閱 [第 107 頁的『使用 IBM MQ for z/OS 傾出顯示畫面處理傾出』](#))。

當要求傾出時，只會併入儲存體中可用的記錄。每一個格式化記錄都遵循標題 *****LOGDATA*****。

z/OS 上的 SVC 傾出

請利用這個主題來瞭解如何在 z/OS 上暫停 SVC 傾出，以及為何未產生 SVC 傾出的原因。

未產生 SVC 傾出時

在某些情況下，不會產生 SVC 傾出。通常會因為時間或空間問題或安全違規而暫停傾出。下列清單彙總可能未產生 SVC 傾出的其他原因：

- z/OS 服務功能層次指示處理 (SLIP) 指令已抑制異常終止。
z/OS MVS 起始設定及調整參照手冊中的 [IEACMD00](#) 說明列出 IPL 時所執行 SLIP 指令的預設值。
- 異常終止原因碼是不需要傾出即可判斷異常終止的原因。
- 使用 SDWACOMU 或 SDWAEAS (系統診斷工作區的一部分，SDWA) 來暫停傾出。

使用 z/OS DAE 抑制 IBM MQ for z/OS 傾出

您可以暫停複製先前傾出的 SVC 傾出。[z/OS MVS 診斷: 工具及服務輔助工具](#) 手冊提供使用 z/OS 傾出分析及排除 (DAE) 的詳細資料。

為了支援 DAE，IBM MQ for z/OS 會定義兩個變數記錄區域 (VRA) 索引鍵及最小症狀字串。這兩個 VRA 金鑰如下：

- KEY VRADAE (X'53 ')。沒有與此金鑰相關聯的資料。
- KEY VRAMINSC (X'52 ') DATA (X'08')

IBM MQ for z/OS 提供系統診斷工作區 (SDWA) 中症狀字串下限的下列資料：

- 載入模組名稱

- CSECT 名稱
- 異常終止碼
- 回復常式名稱
- 失敗指示區域
- REG/PSW 差異
- 原因碼
- 元件 ID
- 元件子函數

如果九個症狀中有八個 (VRAMINSC 鍵中的 X'08 ') 相同，則傾出會被視為重複，以抑制重複傾出。

z/OS 處理 z/OS 上的效能問題

請利用這個主題來更詳細地調查 IBM MQ for z/OS 效能問題。

效能問題的特點如下：

- 線上交易中的不良回應時間
- 批次工作需要很長時間才能完成
- 訊息傳輸緩慢

效能問題可能是由許多因素所造成，從整體 z/OS 系統中缺乏資源，到應用程式設計欠佳。

下列主題會透過特定子系統 (例如 IBM MQ 及 CICS 或 IMS) 的問題，來呈現問題及建議的解決方案，從相對容易診斷的問題開始，例如 DASD 競用。

- [第 122 頁的『IBM MQ for z/OS 系統考量』](#)
- [第 122 頁的『CICS 限制』](#)
- [第 123 頁的『處理正在 z/OS 上慢速執行或已停止的應用程式』](#)

遠端佇列作業問題可能是因為網路壅塞及其他網路問題。它們也可能是遠端佇列管理程式的問題所造成。

相關概念

[第 127 頁的『處理 z/OS 上不正確的輸出』](#)

不正確的輸出可能是遺漏、非預期或毀損的資訊。請閱讀本主題以進一步調查。

相關工作

[第 8 頁的『進行起始檢查』](#)

您可以進行一些起始檢查，以回答您可能遇到的一般問題。

z/OS IBM MQ for z/OS 系統考量

z/OS 系統是在調查效能問題時需要檢查的區域。

您可能已經知道 z/OS 系統面臨壓力，因為這些問題會影響許多子系統及應用程式。

您可以使用標準監視工具，例如「資源監視機能 (RMF)」來監視及診斷這些問題。它們可能包括：

- 儲存體上的限制 (分頁)
- 處理器週期的限制
- DASD 上的限制
- 通道路徑使用情形

請使用一般 z/OS 調整技術來解決這些問題。

z/OS CICS 限制

CICS 限制也可能會對 IBM MQ for z/OS 效能產生負面影響。如需 CICS 限制項的進一步相關資訊，請使用本主題。

IBM MQ 作業的效能可能會受到 CICS 限制項的影響。例如，您的系統可能已達到 MAXTASK，強制交易等待，或者 CICS 系統可能儲存體短缺。例如，CICS 可能不會排程交易，因為已達到並行作業數，或 CICS 偵測到資源問題。如果您懷疑 CICS 造成效能問題 (例如，因為批次和 TSO 工作順利執行，但 CICS 作業逾時，或回應時間不佳)，請參閱 *CICS 問題判斷手冊* 和 *CICS 效能手冊*。

註: CICS I/O 至暫時資料外部分割區資料集使用 z/OS RESERVE 指令。這可能會影響相同磁區上其他資料集的 I/O。

z/OS 處理正在 z/OS 上慢速執行或已停止的應用程式

等待和迴圈可能出現類似症狀。使用本主題中的鏈結，可協助區分 z/OS 上的等待與迴圈。

等待和迴圈的特點是無回應性。不過，可能很難區分等待、迴圈及效能不佳。

下列任何症狀可能是由等待或迴圈所造成，或由未正確調整或超載的系統所造成:

- 似乎已停止執行的應用程式 (如果 IBM MQ for z/OS 仍可回應，則此問題可能是由應用程式問題所造成)
- 不產生回應的 MQSC 指令
- 過多使用處理器時間

若要執行這些主題中顯示的測試，您需要存取 z/OS 主控台，並且能夠發出操作員指令。

- [第 123 頁的『在 z/OS 上區分等待與迴圈』](#)
- [第 124 頁的『處理 z/OS 上的等待』](#)
- [第 126 頁的『處理 z/OS 上的迴圈』](#)

相關工作

[第 8 頁的『進行起始檢查』](#)

您可以進行一些起始檢查，以回答您可能遇到的一般問題。

z/OS 在 z/OS 上區分等待與迴圈

IBM MQ for z/OS 上的等待及迴圈可能呈現類似症狀。請利用這個主題來協助判斷您是否經歷等待或迴圈。

由於等待和迴圈可能難以區分，在某些情況下，您需要先執行詳細調查，然後再決定適合您問題的分類。

本節為您提供選擇最佳分類的指引，並在您決定分類時提供如何執行的建議。

等待數

對於問題判斷，等待狀態被視為已暫停執行作業的狀態。亦即，作業已開始執行，但已暫停但未完成，且隨後無法回復。

系統中識別為等待的問題可能是由下列任何一項所造成:

- 等待 MQI 呼叫
- 等待 CICS 或 IMS 呼叫
- 等待另一個資源 (例如，檔案 I/O)
- 歐洲央行等待
- 等待中的 CICS 或 IMS 區域
- TSO 等待中
- IBM MQ for z/OS Waiting for work
- 明顯的等待，由迴圈所造成
- 由於優先順序較高的工作，CICS 或 MVS 未分派您的作業
- Db2 或 RRS 非作用中

迴圈

迴圈是重複執行某些程式碼。如果您尚未規劃迴圈，或如果您已將它設計成應用程式，但由於某些原因而未終止，則會出現一組症狀，視程式碼執行的動作以及任何介面元件和產品對它的反應而定。在某些情況下，一開始可能會將迴圈診斷為等待或效能問題，因為迴圈作業會與迴圈中未涉及的其他作業競爭系統資源。不過，迴圈會耗用資源，但等待則不會。

系統中的明顯迴圈問題可能是由下列任一項所造成：

- 應用程式比平常執行更多處理程序，因此需要較長時間才能完成
- 應用程式邏輯中的迴圈
- 含有 MQI 呼叫的迴圈
- 含有 CICS 或 IMS 呼叫的迴圈
- CICS 或 IMS 程式碼中的迴圈
- IBM MQ for z/OS 中的迴圈

等待及迴圈的症狀

下列任何症狀都可能是由等待、迴圈或錯誤調整或超載系統所造成：

- MQGET WAIT 上的逾時
- 暫停的批次工作
- TSO 階段作業已暫停
- CICS 作業已暫停
- 由於資源限制而未啟動交易，例如 CICS MAX 作業
- 佇列已滿且未處理
- 未接受系統指令，或未產生任何回應

相關概念

[第 124 頁的『處理 z/OS 上的等待』](#)

等待可能發生在 IBM MQ for z/OS 上的批次或 TSO 應用程式、CICS 交易及其他元件中。請利用這個主題來判斷等待可能發生的位置。

[第 126 頁的『處理 z/OS 上的迴圈』](#)

z/OS 系統的不同區域可能會出現迴圈。請利用這個主題來協助判斷發生迴圈的位置。

z/OS 處理 z/OS 上的等待

等待可能發生在 IBM MQ for z/OS 上的批次或 TSO 應用程式、CICS 交易及其他元件中。請利用這個主題來判斷等待可能發生的位置。

在調查作業或子系統等待的問題時，必須考量作業或子系統執行所在的環境。

可能是您的 z/OS 系統通常受到壓力。在此情況下，可能有許多症狀。如果沒有足夠的實際儲存體，工作體驗會在分頁岔斷或換出時等待。輸入/輸出 (I/O) 競用或高通道使用率也會導致等待。

您可以使用標準監視工具，例如資源監視機能 (RMF) 來診斷這些問題。請使用一般 z/OS 調整技術來解決這些問題。

批次或 TSO 程式是否在等待？

請考量下列幾點：

您的程式可能正在等待另一個資源

例如，另一個程式正在保留以進行更新的 VSAM 控制間隔 (CI)。

您的程式可能正在等待尚未到達的訊息

例如，如果此狀況是持續監視佇列的伺服器程式，則此狀況可能是正常行為。

或者，您的程式可能正在等待已到達但尚未確定的訊息。

發出 DIS CONN (*) TYPE (HANDLE) 指令，並檢查程式正在使用的佇列。

如果您懷疑您的程式已發出不涉及 MQGET WAIT 的 MQI 呼叫，且控制未從 IBM MQ 傳回，在取消批次或 TSO 程式之前，請先取得批次或 TSO 工作及 IBM MQ 子系統的 SVC 傾出。

另請考量等待狀態可能是另一個程式發生問題 (例如異常終止，請參閱第 128 頁的『訊息未在預期的 z/OS 上送達』) 或 IBM MQ 本身 (請參閱第 125 頁的『IBM MQ 是否正在等待 z/OS ?』)。請參閱第 104 頁的『IBM MQ for z/OS 傾出』 (特別是第 106 頁的圖 8) 以取得取得傾出的相關資訊。

如果問題持續發生，請參閱第 36 頁的『聯絡 IBM 支援中心』，以取得向 IBM 報告問題的相關資訊。

CICS 交易是否在等待?

請考量下列幾點：

CICS 可能有壓力

這可能表示已達到容許的作業數上限 (MAXTASK)，或存在儲存體短缺 (SOS) 狀況。請檢查主控台日誌，以取得可能說明此情況的訊息 (例如，SOS 訊息)，或參閱 CICS 問題判斷手冊。

交易可能正在等待另一個資源

例如，這可能是檔案 I/O。您可以使用 CEMT INQ TASK 來查看作業正在等待的內容。如果資源類型是 MQSERIES，則您的交易正在等待 IBM MQ (在 MQGET WAIT 或作業切換中)。否則，請參閱 CICS Problem Determination Guide，以判斷等待的原因。

交易可能正在等待 IBM MQ for z/OS

這可能是正常的，例如，如果您的程式是等待訊息到達佇列的伺服器程式。否則它可能是交易異常終止的結果，例如 (請參閱第 128 頁的『訊息未在預期的 z/OS 上送達』)。如果是這樣，則會在 CSMT 日誌中報告異常終止。

交易可能正在等待遠端訊息

如果您是使用分散式佇列，則程式可能正在等待尚未從遠端系統遞送的訊息 (如需進一步資訊，請參閱第 129 頁的『在 z/OS 上使用分散式佇列時遺漏訊息的問題』)。

如果您懷疑程式已發出不涉及 MQGET WAIT (亦即，它位於作業交換器中) 的 MQI 呼叫，且控制尚未從 IBM MQ 傳回，請先取得 CICS 區域及 IBM MQ 子系統的 SVC 傾出，然後再取消 CICS 交易。如需等待的相關資訊，請參閱第 126 頁的『處理 z/OS 上的迴圈』。請參閱第 104 頁的『IBM MQ for z/OS 傾出』 (特別是第 106 頁的圖 8) 以取得取得傾出的相關資訊。

如果問題持續發生，請參閱第 36 頁的『聯絡 IBM 支援中心』，以取得向 IBM 報告問題的相關資訊。

Db2 是否在等待?

如果您的調查指出 Db2 正在等待，請檢查下列項目：

1. 使用 Db2 -DISPLAY THREAD (*) 指令來判斷佇列管理程式與 Db2 子系統之間是否正在進行任何活動。
2. 請嘗試並判斷是否有任何等待是在佇列管理程式子系統的本端或跨 Db2 子系統。

RRS 是否在作用中?

- 請使用 D RRS 指令來判斷 RRS 是否為作用中。

IBM MQ 是否正在等待 z/OS ?

如果您的調查指出 IBM MQ 本身正在等待，請檢查下列項目：

1. 使用 DISPLAY THREAD (*) 指令來檢查是否有任何項目連接至 IBM MQ。
2. 使用 SDSF DA 或 z/OS 指令 DISPLAY A,xxxxMSTR 來判斷是否有任何處理器用量 (如第 27 頁的『您的應用程式或 IBM MQ for z/OS 是否已停止處理工作?』所示)。
 - 如果 IBM MQ 正在使用一些處理器時間，請重新考量 IBM MQ 可能等待的其他原因，或考量這實際上是否為效能問題。

- 如果沒有處理器活動，請檢查 IBM MQ 是否回應指令。如果您可以取得回應，請重新考量 IBM MQ 可能等待的其他原因。
- 如果無法取得回應，請檢查主控台日誌中可能說明等待的訊息 (例如，IBM MQ 可能已使用完作用中日誌資料集，且正在等待卸載處理)。

如果您滿意 IBM MQ 已停滯，請在 QUIESCE 及 FORCE 模式中使用 STOP QMGR 指令，以終止目前正在執行的任何程式。

如果 STOP QMGR 指令無法回應，請取消具有傾出的佇列管理程式，然後重新啟動。如果問題再次發生，請參閱第 36 頁的『[聯絡 IBM 支援中心](#)』以取得進一步指引。

相關概念

第 123 頁的『[在 z/OS 上區分等待與迴圈](#)』

IBM MQ for z/OS 上的等待及迴圈可能呈現類似症狀。請利用這個主題來協助判斷您是否經歷等待或迴圈。

第 126 頁的『[處理 z/OS 上的迴圈](#)』

z/OS 系統的不同區域可能會出現迴圈。請利用這個主題來協助判斷發生迴圈的位置。

處理 z/OS 上的迴圈

z/OS 系統的不同區域可能會出現迴圈。請利用這個主題來協助判斷發生迴圈的位置。

下列主題說明您可能遇到的各種類型迴圈，並建議部分回應。

是批次應用程式迴圈嗎？

如果您懷疑批次或 TSO 應用程式迴圈，請使用主控台來發出 z/OS 指令 DISPLAY JOBS,A (適用於批次應用程式) 或 DISPLAY TS,A (適用於 TSO 應用程式)。請記下所顯示資料中的 CT 值，然後重複該指令。

如果任何作業顯示 CT 值大幅增加，則可能是作業在迴圈中。您也可以使用 SDSF DA，它會顯示每一個位址空間所使用的處理器百分比。

批次工作是否產生大量輸出？

此行為的範例可能是瀏覽佇列並列印訊息的應用程式。如果已使用 BROWSE FIRST 啟動瀏覽作業，且後續呼叫未重設為 BROWSE NEXT，則應用程式會反覆地瀏覽並列印佇列上的第一個訊息。

如果您懷疑執行中工作可能導致問題，則可以使用 SDSF DA 來查看執行中工作的輸出。

CICS 區域是否顯示大量處理器活動？

可能是 CICS 應用程式在迴圈中，或 CICS 區域本身處於迴圈中。如果交易進入緊密 (不產) 迴圈，您可能看到 AICA 異常終止。

如果您懷疑 CICS 或 CICS 應用程式在迴圈中，請參閱 *CICS Problem Determination Guide*。

IMS 區域是否顯示大量處理器活動？

可能是 IMS 應用程式在迴圈中。如果您懷疑此行為，請參閱 *IMS Diagnosis Guide and Reference I*。

佇列管理程式是否顯示大量處理器活動？

請嘗試從主控台輸入 MQSC DISPLAY 指令。如果您未收到任何回應，則可能是佇列管理程式在迴圈中。遵循第 27 頁的『[您的應用程式或 IBM MQ for z/OS 是否已停止處理工作?](#)』中顯示的程序，以顯示佇列管理程式所使用處理器時間的相關資訊。如果此指令指出佇列管理程式處於迴圈中，請取得記憶體傾出，取消佇列管理程式並重新啟動。

如果問題持續發生，請參閱第 36 頁的『[聯絡 IBM 支援中心](#)』，以取得向 IBM 報告問題的相關資訊。

佇列、頁集或「連結機能」結構是否非預期地填滿？

若是如此，可能表示應用程式在迴圈中，並將訊息放到佇列中。（它可能是批次、CICS 或 TSO 應用程式。）

識別迴圈應用程式

在忙碌的系統中，可能難以識別造成問題的應用程式。如果您保留應用程式對佇列的交互參照，請終止任何可能將訊息放入佇列的程式或交易。請先調查這些程式或交易，然後再重新使用它們。（最可能的原因是新應用程式或已變更的應用程式；請檢查變更日誌以識別它們。）

請嘗試在佇列上發出 DISPLAY QSTATUS 指令。此指令會傳回佇列的相關資訊，該佇列可能有助於識別哪些應用程式在迴圈中。

觸發定義不正確

可能是因為物件定義不正確而未觸發取得應用程式，例如，佇列可能設為 NOTRIGGER。

分散式佇列 (distributed queuing)

使用分散式佇列，此問題的症狀可能是接收系統中的訊息，指出對無法傳送郵件之佇列的 MQPUT 呼叫失敗。此問題可能是因為無法傳送郵件的佇列也已填滿。無法傳送郵件的佇列訊息標頭（無法傳送郵件的標頭結構）包含原因或回饋碼，說明訊息可能無法放入目標佇列的原因。如需無法傳送的郵件標頭結構的相關資訊，請參閱 [MQDLH-無法傳送的郵件標頭](#)。

將佇列配置給頁集

如果設定的特定頁面經常填滿，則將佇列配置給頁集可能有問題。如需相關資訊，請參閱 [IBM MQ for z/OS 效能限制](#)。

共用佇列

「連結機能」結構是否已滿？z/OS 指令 DISPLAY CF 顯示「連結機能」儲存體的相關資訊，包括總量、使用中總計，以及可用控制和非控制儲存體總計。「RMF 連結機能使用情形摘要報告」提供此資訊更永久的副本。

作業及 IBM MQ for z/OS 是否顯示大量處理器活動？

在此情況下，作業可能在 MQI 呼叫上形成迴圈（例如，反覆地瀏覽相同的訊息）。

相關概念

第 123 頁的『[在 z/OS 上區分等待與迴圈](#)』

IBM MQ for z/OS 上的等待及迴圈可能呈現類似症狀。請利用這個主題來協助判斷您是否經歷等待或迴圈。

第 124 頁的『[處理 z/OS 上的等待](#)』

等待可能發生在 IBM MQ for z/OS 上的批次或 TSO 應用程式、CICS 交易及其他元件中。請利用這個主題來判斷等待可能發生的位置。

處理 z/OS 上不正確的輸出

不正確的輸出可能是遺漏、非預期或毀損的資訊。請閱讀本主題以進一步調查。

術語「不正確的 outputcorporation 可以用許多不同的方式來解譯，其在此產品說明文件中用於問題判斷的意義在 [第 34 頁的『您是否取得不正確的輸出?』](#) 中說明。

下列主題包含系統可能發生之問題的相關資訊，並分類為不正確的輸出：

- [在您預期的情況下未到達的應用程式訊息](#)
- [包含錯誤資訊或已毀損資訊的應用程式訊息](#)

如果您的應用程式使用分散式佇列，也會說明您可能遇到的其他問題。

- [第 128 頁的『訊息未在預期的 z/OS 上送達』](#)
- [第 129 頁的『在 z/OS 上使用分散式佇列時遺漏訊息的問題』](#)
- [第 130 頁的『在 z/OS 上使用訊息分組時取得訊息的問題』](#)
- [第 130 頁的『尋找傳送至 z/OS 上叢集佇列的訊息』](#)
- [第 131 頁的『尋找傳送至 IBM MQ - IMS 橋接器的訊息』](#)

- [第 131 頁的『訊息包含 z/OS 的非預期或毀損資訊』](#)

相關概念

[第 122 頁的『處理 z/OS 上的效能問題』](#)

請利用這個主題來更詳細地調查 IBM MQ for z/OS 效能問題。

相關工作

[第 8 頁的『進行起始檢查』](#)

您可以進行一些起始檢查，以回答您可能遇到的一般問題。

訊息未在預期的 z/OS 上送達

遺漏訊息可能有不同的原因。請利用這個主題來進一步調查原因。

如果訊息在您預期時未抵達佇列，請檢查下列項目：

訊息是否已順利放入佇列？

IBM MQ 是否發出 MQPUT 的回覆碼及原因碼，例如：

- 是否已正確定義佇列，例如 MAXMSGL 是否足夠大？(原因碼 2030)。
- 應用程式可以將訊息放到佇列中嗎 (佇列是否已啟用 MQPUT 呼叫)？(原因碼 2051)。
- 佇列是否已滿？這可能表示應用程式無法將必要的訊息放入佇列中 (原因碼 2053)。

佇列是共用佇列嗎？

- 是否已在 CFRM 原則資料集中順利定義「連結機能」結構？保留在共用佇列上的訊息儲存在「連結機能」內。
- 您已啟動 CFRM 原則嗎？

佇列是叢集佇列嗎？

如果是的話，在不同的佇列管理程式上可能有多個佇列實例。這表示訊息可能位於不同的佇列管理程式上。

- 您要將訊息移至叢集佇列嗎？
- 您的應用程式是否設計為使用叢集佇列？
- 是否將訊息放入與預期不同的佇列實例？

請檢查任何叢集工作量結束程式，以查看它們是否如預期般處理訊息。

你會失敗嗎？

- 應用程式是否需要採用同步點？

如果在同步點內放置或取得訊息，則在確定回復單元之前，其他作業無法使用這些訊息。

- MQGET 上的時間間隔是否足夠長？

如果您使用分散式處理，則應該容許合理的網路延遲，或遠端系統上的問題。

- 您預期的訊息定義為持續性嗎？

如果沒有，且佇列管理程式已重新啟動，則會刪除訊息。共用佇列是異常狀況，因為非持續訊息在佇列管理程式重新啟動之後仍然存在。

- 您是否在等待由訊息或相關性 ID (*MsgId* 或 *CorrelId*) 識別的特定訊息？

請確認您正在等待具有正確 *MsgId* 或 *CorrelId* 的訊息。成功的 MQGET 呼叫會將這兩個值設為所取得訊息的值，因此您可能需要重設這些值，才能順利取得另一個訊息。

也請檢查您是否可以從佇列中取得其他訊息。

- 其他應用程式可以從佇列取得訊息嗎？

如果是，是否有另一個應用程式已擷取訊息？

如果佇列是共用佇列，請檢查其他佇列管理程式上的應用程式是否未取得訊息。

如果您找不到佇列的任何錯誤，且佇列管理程式本身正在執行中，請對您預期將訊息放入佇列的處理程序進行下列檢查：

- 應用程式是否已開始？

如果應該已觸發它，請檢查是否已指定正確的觸發選項。

- 觸發監視器是否在執行中？
- 是否已正確定義觸發程式程序（針對 IBM MQ for z/OS 及 CICS 或 IMS）？
- 它是否正確完成？

例如，在 CICS 日誌中尋找異常終止的證明。

- 應用程式已確定其變更，還是已取消？

在 CICS 日誌中尋找指出此情況的訊息。

如果多個交易負責處理佇列，則它們有時可能會彼此衝突。例如，一個交易可能發出緩衝區長度為零的 MQGET 呼叫，以找出訊息的長度，然後發出指定該訊息 *MsgId* 的特定 MQGET 呼叫。不過，發生此情況時，另一個交易可能已針對該訊息發出成功 MQGET 呼叫，因此第一個應用程式會收到完成碼 MQRC_NO_MSG_AVAILABLE。預期在多台伺服器環境中執行的應用程式必須設計成處理此狀況。

您的系統是否有任何中斷？例如，如果您預期的訊息已由 CICS 應用程式放入佇列，且 CICS 系統已關閉，則該訊息可能不確定。這表示佇列管理程式不知道訊息是否應該確定或取消，因此已鎖定它，直到發生重新同步時解決為止。

註：如果 CICS 決定取消訊息，則會在重新同步之後刪除該訊息。

另請考量可能已收到訊息，但您的應用程式無法以某種方式處理該訊息。例如，訊息預期格式的錯誤是否導致您的程式拒絕它？如果是，請參閱 [第 131 頁的『訊息包含 z/OS 的非預期或毀損資訊』](#)。

z/OS 在 z/OS 上使用分散式佇列時遺漏訊息的問題

在 IBM MQ for z/OS 上使用分散式佇列作業時，請使用本主題來瞭解遺漏訊息的可能原因。

如果您的應用程式使用分散式佇列，請考量下列要點：

傳送及接收系統上是否已正確安裝分散式佇列？

請確定已正確遵循 [配置 z/OS](#) 中關於安裝分散式佇列管理機能的指示。

兩個系統之間是否有鏈結可用？

檢查這兩個系統是否可用，並連接至 IBM MQ for z/OS。請檢查兩個系統之間的 LU 6.2 或 TCP/IP 連線是否處於作用中，或檢查與之通訊的任何其他系統上的連線定義。

如需網路中追蹤路徑傳訊的相關資訊，請參閱 [監視及效能](#)。

通道是否在執行中？

- 針對傳輸佇列發出下列指令：

```
DISPLAY QUEUE (qname) IPPROCS
```

如果 IPPROCS 的值為 0，則表示負責處理此傳輸佇列的通道不在執行中。

- 針對通道發出下列指令：

```
DISPLAY CHSTATUS (channel-name) STATUS MSGS
```

請使用這個指令所產生的輸出，來檢查通道是否提供正確的傳輸佇列，以及是否連接正確的目標機器和埠。您可以從 STATUS 欄位判定通道是否在執行中。您也可以檢查 MSGS 欄位，以查看通道上是否已傳送任何訊息。

如果通道處於 RETRYING 狀態，這可能是由另一端的問題所造成。請檢查通道起始程式和接聽器是否已啟動，以及通道是否尚未停止。如果有人已停止通道，您需要手動啟動它。

傳送系統中是否有觸發設定？

請檢查通道起始程式是否在執行中。

傳輸佇列是否已設定觸發？

如果通道在特定情況下停止，則可以針對傳輸佇列設定觸發。

您正在等待來自遠端系統的回覆訊息嗎？

如先前所述，檢查遠端系統的定義，並檢查是否在遠端系統中啟動觸發。另請檢查兩個系統之間的 LU 6.2 連線是否不是單一階段作業 (如果是，則您無法接收回覆訊息)。

請檢查遠端佇列管理程式上的佇列是否存在、是否未滿，並接受訊息長度。如果未滿足其中任何準則，遠端佇列管理程式會嘗試將訊息放置在無法傳送郵件的佇列上。如果訊息長度超過通道允許的長度上限，傳送端佇列管理程式會嘗試將訊息放置在其無法傳送郵件的佇列上。

佇列是否已滿？

這可能表示應用程式無法將必要的訊息放入佇列。如果是這樣，請檢查訊息是否已放入無法傳送郵件的佇列中。

無法傳送郵件的佇列訊息標頭 (無法傳送郵件的標頭結構) 包含原因或回饋碼，說明無法將訊息放入目標佇列的原因。如需無法傳送的郵件標頭結構的相關資訊，請參閱 [MQDLH-無法傳送的郵件標頭](#)。

傳送端和接收端佇列管理程式之間是否不符？

例如，訊息長度可能超過接收端佇列管理程式所能處理的長度。請檢查主控台日誌，以取得錯誤訊息。

傳送及接收通道的通道定義是否相容？

例如，序號的折返值不符會停止通道。請參閱 [分散式佇列及叢集](#)。

是否已正確執行資料轉換？

如果訊息來自不同的佇列管理程式，則 CCSID 和編碼相同，或需要執行資料轉換。

您的通道已定義為快速遞送非持續訊息嗎？

如果已定義通道，並將 NPMSPEED 屬性設為 FAST (預設值)，且通道已因某些原因而停止，然後重新啟動，則可能會遺失非持續訊息。如需快速訊息的相關資訊，請參閱 [非持續訊息速度 \(NPMSPEED\)](#)。

通道結束程式是否導致以非預期的方式處理訊息？

例如，安全結束程式可能會阻止通道啟動，或者 MQXCC_CLOSE_CHANNEL 的 *ExitResponse* 可能會終止通道。

z/OS 在 z/OS 上使用訊息分組時取得訊息的問題

在 IBM MQ for z/OS 上使用訊息分組時，請利用這個主題來瞭解取得訊息的部分問題。

應用程式是否在等待完整的訊息群組？

請確定群組中的所有訊息都在佇列上。如果您是使用分散式佇列，請參閱第 129 頁的『在 z/OS 上使用分散式佇列時遺漏訊息的問題』。請確定群組中的最後一則訊息在訊息描述子中已設定適當的 `MsgFlags`，以指出它是最後一則訊息。請確定群組中訊息的訊息期限設為足夠長的時間，使它們在擷取之前不會到期。

如果已從群組擷取訊息，且取得要求不是依邏輯順序，請在擷取其他群組訊息時關閉等待完整群組的選項。

如果應用程式以邏輯順序對完整群組發出取得要求，且在擷取群組時中途找不到訊息：

請確定沒有其他應用程式正在針對佇列執行並取得訊息。請確定群組中訊息的訊息期限設為足夠長的時間，不會在擷取之前到期。請確定沒有人發出 `CLEAr QUEUE` 指令。您可以透過依群組 ID 取得訊息，而不指定邏輯順序選項，從佇列中擷取不完整群組。

z/OS 尋找傳送至 z/OS 上叢集佇列的訊息

請利用這個主題來瞭解在 IBM MQ for z/OS 上尋找傳送至叢集佇列的訊息所涉及的部分問題。

在您可以使用這些主題中說明的技術來尋找未到達叢集佇列的訊息之前，您需要先判斷管理訊息傳送目的地佇列的佇列管理程式。您可以下列方式來判斷：

- 您可以使用 `DISPLAY QUEUE` 指令來要求叢集佇列的相關資訊。

- 您可以使用 MQPMO 結構中傳回的佇列及佇列管理程式名稱。

如果您指定訊息的 MQOO_BIND_ON_OPEN 選項，這些欄位會提供訊息的目的地。如果訊息未連結至特定佇列及佇列管理程式，則這些欄位會提供訊息傳送至其中的第一個佇列及佇列管理程式的名稱。在此情況下，它可能不是訊息的最終目的地。

尋找傳送至 IBM MQ - IMS 橋接器的訊息

請利用這個主題來瞭解傳送至 IBM MQ - IMS 橋接器的訊息遺漏的可能原因。

如果您使用 IBM MQ - IMS 橋接器，且您的訊息未如預期般送達，請考量下列事項：

IBM MQ - IMS 橋接器是否在執行中？

針對橋接器佇列發出下列指令：

```
DISPLAY QSTATUS(qname) IPPROCS CURDEPTH
```

IPPROCS 的值應該是 1；如果它是 0，請檢查下列項目：

- 佇列是橋接器佇列嗎？
- IMS 在執行中嗎？
- OTMA 已啟動嗎？
- IBM MQ 是否已連接至 OTMA？

註：您可以使用兩則 IBM MQ 訊息來確定您是否具有 OTMA 連線。如果作業的工作日誌中出現訊息 CSQ2010I，但訊息 CSQ2011I 不存在，則 IBM MQ 會連接至 OTMA。這則訊息也會告訴您已連接哪一個 IBM MQ 系統 OTMA。如需這些訊息內容的相關資訊，請參閱 [IBM MQ for z/OS 訊息、完成及原因碼](#)。

在佇列管理程式內，有一項作業會處理每一個 IMS 橋接器佇列。此作業從佇列中取得，將要求傳送至 IMS，然後執行確定。如果使用持續訊息，則確定需要磁碟 I/O，因此處理程序需要比非持續訊息更長的時間。處理取得、傳送及確定的時間會限制作業可以處理訊息的速率。如果作業可以跟上工作量，則現行深度接近零。如果您發現現行深度通常大於零，您可以使用兩個佇列而非一個佇列來增加傳輸量。

使用 IMS 指令 /DIS OTMA 來檢查 OTMA 是否為作用中。

如果您的訊息流向 IMS，請檢查下列項目：

- 使用 IMS 指令 /DIS TMEMBER client TPIPE ALL 來顯示 IMS Tp 管道的相關資訊。從此，您可以判定每一個 Tpipe 上放入佇列及從中移出佇列的訊息數。（確定模式 1 訊息通常不會在 Tpipe 上排入佇列。）
- 使用 IMS 指令 /DIS A 來顯示是否有相依區域可供 IMS 交易在其中執行。
- 使用 IMS 指令 /DIS TRAN trancode 來顯示置入佇列以進行交易的訊息數。
- 使用 IMS 指令 /DIS PROG progname 來顯示程式是否已停止。

回覆訊息是否傳送至正確的位置？

發出下列指令：

```
DISPLAY QSTATUS(*) CURDEPTH
```

CURDEPTH 是否指出佇列上有您未預期的回覆？

訊息包含 z/OS 的非預期或毀損資訊

請利用這個主題來瞭解一些可能導致 z/OS 輸出非預期或毀損的問題。

如果訊息中包含的資訊不是您應用程式所預期的，或已在某種程度上毀損，請考量下列要點：

您的應用程式或將訊息放入佇列的應用程式是否已變更？

請確定所有變更同時反映在所有需要注意變更的系統上。

例如，將訊息格式化的記錄定義檔可能已變更，在此情況下，必須重新編譯這兩個應用程式，才能使變更生效。如果一個應用程式尚未重新編譯，另一個應用程式會出現毀損的情況。

檢查資料的外部來源 (例如 VSAM 資料集) 是否未變更。如果尚未執行任何必要的重新編譯，這也可能使您的資料失效。另請檢查您用於輸入訊息資料的任何 CICS 對映和 TSO 畫面是否未變更。

應用程式是否將訊息傳送至錯誤佇列？

請檢查應用程式正在接收的訊息是否預期用於服務不同佇列的應用程式。必要的話，請變更安全定義，以防止未獲授權的應用程式將訊息放入錯誤佇列。

如果您的應用程式已使用別名佇列，請檢查別名是否指向正確的佇列。

如果您變更佇列使其成為叢集佇列，則現在可能包含來自不同應用程式來源的訊息。

是否已正確指定此佇列的觸發資訊？

請檢查您的應用程式是否應該已啟動，或是否應該已啟動不同的應用程式？

是否已正確執行資料轉換？

如果訊息來自不同的佇列管理程式，則 CCSID 和編碼相同，或需要執行資料轉換。

檢查 MQMD 結構的 *Format* 欄位是否對應於訊息的內容。否則，資料轉換處理程序可能無法正確處理訊息。

如果這些檢查無法讓您解決問題，請檢查應用程式邏輯，包括傳送訊息的程式，以及接收訊息的程式。

處理擷取通道起始程式 (CHINIT) 的 SMF 資料時的問題

通道帳戶和 CHINIT 統計資料 SMF 資料可能因各種原因而無法擷取。

如需相關資訊，請參閱：

相關資訊

[通道起始程式的 SMF 記錄佈置](#)

通道統計資料疑難排解

檢查是否未針對通道產生通道統計 SMF 資料。

程序

1. 檢查您是否已在佇列管理程式或通道層次設定 STATCHL。
 - 通道層次的 OFF 值表示不收集此通道的資料。
 - 佇列管理程式層次的 OFF 值表示不收集具有 STATCHL (QMGR) 之通道的資料。
 - 值 NONE (僅適用於佇列管理程式層次) 表示不會收集所有通道的資料，不論其 STATCHL 設定為何。
2. 對於用戶端通道，請檢查是否在佇列管理程式層次設定 STATCHL。
3. 對於自動定義的叢集傳送端通道，請檢查是否已設定 STATACLS。
4. 發出顯示追蹤指令。您需要啟用 TRACE (A) CLASS (4)，才能收集通道帳戶資料。
5. 如果啟用追蹤，則會寫入 SMF 資料：
 - 在計時間隔上-視 STATIME 系統參數的值而定。零值表示使用 SMF 統計資料播送。使用 DIS SYSTEM 指令可顯示 STATIME 的值。
 - 如果發出 SET SYSTEM 指令來變更 STATIME 系統參數的值。
 - 當 CHINIT 關閉時。
 - 如果發出 STOP TRACE (A) CLASS (4)，則會寫出任何統計資料。

6. 在將資料寫入 SMF 資料集或 SMF 結構之前，SMF 可能會先保留記憶體中的資料。發出 MVS 指令 **D SMF,0**，並記下 MAXSOMPLE 值。SMF 可以在寫入資料之前，將資料保留在記憶體中達 MAXSONEDAL 期間。

相關資訊

[規劃通道起始程式 SMF 資料](#)

[解譯 IBM MQ 效能統計資料](#)

疑難排解 CHINIT 統計資料

檢查是否未產生 CHINIT 統計資料 SMF 資料。

程序

1. 發出顯示追蹤指令。您需要啟用 TRACE (S) CLASS (4)，以取得 CHINIT 的相關資訊。
2. 如果啟用追蹤，則會寫入 SMF 資料：
 - 在計時間隔上視 STATIME 系統參數的值而定。零值表示使用 SMF 統計資料播送。使用 DIS SYSTEM 指令可顯示 STATIME 的值。
 - 如果發出 SET SYSTEM 指令來變更 STATIME 系統參數的值。
 - 當 CHINIT 關閉時。
 - 如果發出 STOP TRACE (S) CLASS (4)，則會寫出任何統計資料。
3. SMF 可以在將資料寫入 SMF 資料集或 SMF 結構之前，先將資料保留在記憶體中。發出 MVS 指令 **D SMF,0**，並記下 MAXSOMPLE 值。SMF 可以在寫入資料之前，將資料保留在記憶體中達 MAXSONEDAL 期間。

DQM 中的問題判斷

與分散式佇列管理 (DQM) 相關的問題判斷層面，以及解決問題的建議方法。

所說明的部分問題是平台及安裝特有的問題。在此情況下，會在文字中清楚說明。

IBM MQ 提供一個公用程式來協助進行名為 **amqldmpa** 的問題判斷。在問題判斷過程中，您的 IBM 客戶服務代表可能會要求您提供公用程式的輸出。

您的 IBM 客戶服務代表將為您提供收集適當診斷資訊所需的參數，以及如何將您記錄的資料傳送至 IBM 的相關資訊。



小心：您不應該依賴此公用程式的輸出格式，因為格式可能會變更而不另行通知。

討論下列實務範例的問題判斷：

- [第 134 頁的『來自通道控制的錯誤訊息』](#)
- [第 134 頁的『連線測試』](#)
- [第 134 頁的『無法傳送郵件的佇列考量』](#)
- [第 135 頁的『驗證檢查』](#)
- [第 135 頁的『不確定的關係』](#)
- [第 135 頁的『通道啟動協議錯誤』](#)
- [第 136 頁的『當通道拒絕執行時』](#)
- [第 137 頁的『正在重試鏈結』](#)
- [第 138 頁的『資料結構』](#)
- [第 138 頁的『使用者結束程式問題』](#)
- [第 138 頁的『災難回復』](#)
- [第 138 頁的『通道切換』](#)
- [第 139 頁的『連線切換』](#)

- [第 139 頁的『用戶端問題』](#)
- [第 139 頁的『錯誤日誌』](#)
- [第 140 頁的『訊息監視』](#)

相關概念

[第 7 頁的『IBM MQ 疑難排解及支援』](#)

如果您的佇列管理程式網路或 IBM MQ 應用程式有問題，請使用說明的技術來協助您診斷及解決問題。

相關工作

[第 9 頁的『對 UNIX, Linux, and Windows 進行起始檢查』](#)

在 UNIX, Linux, and Windows 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

[第 24 頁的『對 z/OS 進行起始檢查』](#)

在 z/OS 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

[第 16 頁的『對 IBM i 進行起始檢查』](#)

在 IBM i 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

相關資訊

[配置分散式佇列](#)

[訊息及原因碼](#)

[通訊協定回覆碼](#)

來自通道控制的錯誤訊息

在通道正常作業期間發現的問題會向系統主控台及系統日誌報告。在 IBM MQ for Windows 中，它們會報告至通道日誌。問題診斷從收集日誌中的所有相關資訊開始，然後分析此資訊以識別問題。

不過，在網路中，這可能很難在正在編譯打包部分訊息的中間系統上發生問題。錯誤狀況 (例如傳輸佇列已滿，後面接著無法傳送郵件的佇列填滿) 會導致該網站的通道關閉。


在此範例中，您在錯誤日誌中收到的錯誤訊息將指出來自遠端站台的問題，但可能無法告訴您該站台錯誤的任何詳細資料。


您需要聯絡遠端網站上的對應項目，以取得問題的詳細資料，並再次收到該通道變成可用的通知。

連線測試

在判定通訊鏈結及組成訊息通道的兩個訊息通道代理程式是否跨所有介面運作時，連線測試很有用。

連線測試不使用傳輸佇列，但它會呼叫部分使用者結束程式。如果發生任何錯誤狀況，則會發出錯誤訊息。

若要使用連線測試，您可以發出 MQSC 指令 PING CHANNEL。在  z/OS  和 i5/OS 上，您也可以使用畫面介面來選取這個選項。

在 UNIX、 i5/OS、及 Windows 上，您也可以使用 MQSC 指令 [PING QMGR](#) 來測試佇列管理程式是否可回應指令。

無法傳送郵件的佇列考量

在某些 IBM MQ 實作中，無法傳送郵件的佇列稱為 未遞送訊息佇列。

如果通道因任何原因而停止執行，應用程式可能會繼續在傳輸佇列上放置訊息，造成潛在的溢位狀況。應用程式可以監視傳輸佇列，以尋找等待傳送的訊息數目，但這不是它們執行的正常功能。

當訊息原始節點中發生此情況，且本端傳輸佇列已滿時，應用程式的 PUT 會失敗。

當在暫置或目的地節點中發生此情況時，MCA 有三種方式可因應此狀況：

1. 透過呼叫 `message-retry` 結束程式 (如果已定義的話)。
2. 透過將所有溢位訊息導向至 無法傳送郵件的佇列 (DLQ) , 將異常狀況報告傳回至要求這些報告的應用程式。
註: 在分散式佇列管理中, 如果訊息對 DLQ 而言太大、DLQ 已滿或 DLQ 無法使用, 則通道會停止且訊息會保留在傳輸佇列上。請確定您的 DLQ 已定義、可用且大小符合您所處理的最大訊息。
3. 如果先前的選項都未成功, 則關閉通道。
4. 將未遞送訊息傳回至傳送端, 並將完整報告傳回至回覆目的地佇列 (MQRC_EXCEPTION_WITH_FULL_DATA 及 MQRC_DISCARD_MSG)。

如果 MCA 無法在 DLQ 上放置訊息:

- 通道停止
- 在訊息通道兩端的系統主控台會發出適當的錯誤訊息
- 工作單元已取消, 且訊息會重新出現在通道傳送通道端的傳輸佇列上
- 已停用傳輸佇列的觸發

驗證檢查

在建立、變更及刪除通道時, 會進行許多驗證檢查, 並在適當時傳回錯誤訊息。

在下列情況下可能會發生錯誤:

- 建立通道時選擇了重複的通道名稱
- 在通道參數欄位中輸入不可接受的資料
- 要變更的通道不確定, 或不存在

不確定的關係

如果通道不確定, 通常會在重新啟動時自動解決, 因此在正常情況下, 系統操作員不需要手動解決通道。如需進一步資訊, 請參閱 [不確定的通道](#)。

通道啟動協議錯誤

在通道啟動期間, 啟動端必須陳述其位置, 並同意通道執行參數與對應的通道。可能會發生兩端無法同意參數的情況, 在此情況下, 通道會關閉, 並向適當的錯誤日誌發出差錯訊息。

共用通道回復

下表顯示共用通道失敗的類型, 以及如何處理每一種類型。

失敗類型:	發生的情況:
通道起始程式通訊子系統失敗	相依於通訊子系統的通道會進入通道重試, 並由負載平衡啟動指令在適當的佇列共用群組通道起始程式上重新啟動。
通道起始程式失敗	通道起始程式失敗, 但相關聯的佇列管理程式仍保持作用中。佇列管理程式會監視失敗並起始回復處理。
佇列管理程式失敗	佇列管理程式失敗 (使相關聯通道起始程式失敗)。佇列共用群組中的其他佇列管理程式會監視事件並起始同層級回復。
共用狀態失敗	通道狀態資訊儲存在 Db2 中, 因此當通道狀態變更時, 失去與 Db2 的連線會變成失敗。執行中通道可以繼續執行, 而不需要存取這些資源。在無法存取 Db2 時, 通道會進入重試。

代表失敗系統的共用通道回復處理需要在管理回復的系統上提供與 Db2 的連線功能, 以擷取共用通道狀態。

當通道拒絕執行時

如果通道拒絕執行，有許多可能的原因。

執行下列檢查：

- 請檢查 DQM 和通道是否已正確設定。如果通道從未執行過，這可能是問題來源。原因可能是：
 - 傳送和接收通道之間的名稱不符 (請記住大寫字母和小寫字母是很重要的)
 - 指定的通道類型不正確
 - 序號佇列 (如果適用的話) 無法使用或已損壞
 - 無法使用無法傳送郵件的佇列
 - 兩個通道定義上的序號折返不同
 - 無法使用佇列管理程式或通訊鏈結
 - 接收端通道可能處於 STOPPED 狀態
 - 連線可能未正確定義
 - 通訊軟體可能有問題 (例如，TCP 是否在執行中?)
- 如果啟動時自動同步化因某些原因而失敗，則可能存在不確定的狀況。這由系統主控台上的訊息指出，狀態畫面可用來顯示不確定的通道。

對此狀況的可能回應如下：

- 以「取消」或「確定」發出「解決通道」要求。

您需要向遠端鏈結監督者確認，以建立前次確定的工作單元 ID (LUWID) 號碼。請對照鏈結尾端的最後一個數字來檢查此項目。如果遠端系統已確定一個數字，且在鏈結的末端尚未確定該數字，則發出 RESOLVE COMMIT 指令。

在所有其他情況下，請發出 RESOLVE BACKOUT 指令。

這些指令的效果是當捨棄已確定的訊息時，取消的訊息會重新出現在傳輸佇列上並重新傳送。

如果懷疑自己，也許以複製已傳送訊息的可能性來退出會是更安全的決定。

- 發出 RESET CHANNEL 指令。

此指令是在循序編號生效時使用，且應該小心使用。其目的是重設訊息的序號，您應該只在使用 RESOLVE 指令來解決任何不確定的狀況之後才使用它。

-  當使用循序編號，且傳送端通道在重設之後啟動時，傳送端通道會採取兩個動作：

- 它會告知接收端通道已重設。
- 它指定傳送端和接收端通道要使用的下一個訊息序號。

- 如果通道的接收端狀態是 STOPPED，則可以透過啟動接收端來重設它。

註：這不會啟動通道，只會重設狀態。通道仍必須從傳送端啟動。

觸發的通道

如果觸發的通道拒絕執行，請在這裡調查不確定訊息的可能性：[第 136 頁的『當通道拒絕執行時』](#)

另一個可能性是通道已將傳輸佇列上的觸發控制參數設為 NOTRIGGER。在下列情況下會發生此情況：

- 發生通道錯誤。
- 通道已停止，因為來自接收端的要求。
- 通道已停止，因為傳送端發生需要人為介入的問題。

在診斷並修正問題之後，請手動啟動通道。

觸發通道無法啟動的狀況範例如下：

1. 傳輸佇列以 FIRST 觸發類型定義。

2. 訊息到達傳輸佇列，並產生觸發訊息。
3. 通道已啟動，但由於無法使用與遠端系統的通訊而立即停止。
4. 可使用遠端系統。
5. 另一則訊息到達傳輸佇列。
6. 第二個訊息不會將佇列深度從零增加到一，因此不會產生觸發訊息 (除非通道處於 RETRY 狀態)。如果發生這種情況，請手動重新啟動通道。

在 IBM MQ for z/OS 上，如果在通道起始程式關閉期間使用 MODE (FORCE) 停止佇列管理程式，則在通道起始程式重新啟動之後，可能需要手動重新啟動部分通道。

轉換失敗

通道拒絕執行的另一個原因可能是兩端都無法執行 ASCII 與 EBCDIC 之間訊息描述子資料的必要轉換，以及整數格式。在此實例中，無法進行通訊。

網路問題

有許多事項要檢查您是否遇到網路問題。

使用 LU 6.2 時，請確定您的定義在整個網路中一致。例如，如果您已在 CICS Transaction Server for z/OS 或 Communications Manager 定義中增加 RU 大小，但在其定義中有一個 MAXDATA 值較小的控制器，則當您嘗試透過網路傳送大型訊息時，階段作業可能會失敗。此問題的症狀可能是通道協議順利進行，但發生訊息傳送時鏈結失敗。

使用 TCP 時，如果您的通道不可靠且連線中斷，您可以為系統或通道設定 KEEPALIVE 值。您可以使用 SO_KEEPALIVE 選項來設定全系統值。

 在 IBM MQ for z/OS 上，您也有下列選項：

- 使用「Keepalive 間隔通道」屬性 (KAIN) 來設定通道特定的保留作用中值。
- 請使用 RCVTIME 和 RCVTMIN 通道起始程式參數。

這些選項在 [檢查通道另一端是否仍然可用](#) 及 [保持作用中間隔 \(KAIN\)](#) 中進行討論。

採用 MCA

「採用 MCA」函數可讓 IBM MQ 取消接收端通道，並在其位置啟動新的通道。

如需此函數的相關資訊，請參閱 [採用 MCA](#)。

DDNS 的登錄時間

啟動群組 TCP/IP 接聽器時，它會向 DDNS 登錄。但在位址可供網路使用之前可能會有延遲。在此期間啟動且以新登錄通用名稱為目標的通道失敗，並出現 [通訊配置中的錯誤](#) 訊息。然後，通道會進入重試，直到名稱可供網路使用為止。延遲的長度取決於所使用的名稱伺服器配置。

撥號問題

IBM MQ 支援透過撥號線路的連線，但您應該知道使用 TCP 時，部分通訊協定提供者會在您每次撥入時指派新的 IP 位址。這可能會導致通道同步化問題，因為通道無法辨識新的 IP 位址，因此無法確保夥伴的確實性。如果您遇到此問題，則需要使用安全結束程式來置換階段作業的連線名稱。

當 IBM MQ for IBM i、UNIX 或 Windows 產品與位於相同層次的另一個產品進行通訊時，不會發生此問題，因為佇列管理程式名稱用於同步化，而非 IP 位址。

正在重試鏈結

可能發生難以辨識的錯誤情境。例如，鏈結及通道可能正常運作，但在接收端出現的某些情況會導致接收端停止。另一個非預期的狀況可能是接收端系統記憶體不足，無法完成交易。

您必須注意這類狀況可能發生，通常是系統看似忙碌，但實際上並未移動訊息。您需要在鏈結的遠端與您的對應項目一起工作，以協助偵測問題並更正問題。

重試考量

如果在正常作業期間發生鏈結失敗，則傳送端或伺服器通道程式本身會啟動另一個實例，前提是：

1. 起始資料協議和安全交換已完成
2. 通道定義中的重試次數大於零

註：對於 IBM i、UNIX 及 Windows，若要嘗試重試，通道起始程式必須在執行中。在 IBM MQ for IBM i、UNIX 及 Windows 系統以外的平台中，此通道起始程式必須監視通道正在使用之傳輸佇列中指定的起始佇列。

z/OS 上的共用通道回復

請參閱 [第 135 頁的『共用通道回復』](#)，其中包含一個表格，顯示共用通道失敗的類型以及如何處理每一種類型。

資料結構

在問題診斷期間檢查日誌和追蹤項目時，需要參考資料結構。

如需相關資訊，請參閱 [通道結束程式呼叫和資料結構](#) 及 [開發應用程式參照](#)。

使用者結束程式問題

通道程式與使用者結束程式之間的互動具有一些錯誤檢查常式，但只有在使用者結束程式遵循特定規則時，此機能才能順利運作。

這些規則在 [傳訊通道的通道結束程式](#) 中說明。當發生錯誤時，最可能的結果是通道停止，且通道程式會發出錯誤訊息，以及來自使用者結束程式的任何回覆碼。在介面的使用者結束程式端偵測到的任何錯誤，都可以透過掃描使用者結束程式本身所建立的訊息來判斷。

您可能需要使用主機系統的追蹤機能來識別問題。

災難回復

災難回復規劃是個別安裝的責任，所執行的功能可能包括供應安全儲存在離站的一般系統「Snapshot」傾出。如果發生災難，這些傾出可用於重新產生系統。如果發生這種情況，您需要知道訊息的預期內容，下列說明旨在讓您開始思考它。

請先在系統重新啟動時重新對映。如果系統因任何原因而失敗，它可能有一個系統日誌，可讓在失敗時執行的應用程式重新產生，方法是從同步點往前重播系統軟體到失敗的瞬間。如果發生此情況而沒有發生錯誤，則可能發生的最差情況是啟動時，相鄰系統的訊息通道同步點可能失敗，且將重新傳送各種通道的最後批次訊息。持續訊息將回復並重新傳送，非持續訊息可能會遺失。

如果系統沒有用於回復的系統日誌，或如果系統回復失敗，或在呼叫災難回復程序的情況下，通道和傳輸佇列可能會回復到較早的狀態，且在通道傳送和接收端保留在本端佇列上的訊息可能不一致。

在本端佇列上放置的訊息可能已遺失。發生這種情況的結果取決於特定的 IBM MQ 實作，以及通道屬性。例如，當嚴格訊息排序生效時，接收通道會偵測序號間隙，且通道會關閉以進行人為介入。然後，回復取決於應用程式設計，在最壞的情況下，傳送端應用程式可能需要從較早的訊息序號重新啟動。

通道切換

對於停止執行的通道問題，可能的解決方案是針對相同的傳輸佇列定義兩個訊息通道，但具有不同的通訊鏈結。其中一個訊息通道是偏好的，另一個是在偏好的通道無法使用時使用的替代通道。

如果這些訊息通道需要觸發，則每一個傳送端通道端都必須存在相關聯的程序定義。

如果要切換訊息通道，請執行下列動作：

- 如果觸發通道，請設定傳輸佇列屬性 NOTRIGGER。
- 請確定現行通道非作用中。
- 解決現行通道上任何不確定的訊息。
- 如果觸發通道，請變更傳輸佇列中的處理程序屬性，以命名與取代通道相關聯的處理程序。

在此環境定義中，部分實作容許通道具有空白程序物件定義，在此情況下，您可以省略此步驟，因為佇列管理程式會尋找並啟動適當的程序物件。

- 重新啟動通道，或者如果已觸發通道，請設定傳輸佇列屬性 TRIGGER。

連線切換

另一個解決方案是從傳輸佇列切換通訊連線。

若要執行此作業：

- 如果觸發傳送端通道，請設定傳輸佇列屬性 NOTRIGGER。
- 請確定通道非作用中。
- 變更連線及設定檔欄位，以連接至取代通訊鏈結。
- 請確定已定義遠端系統的對應通道。
- 重新啟動通道，或如果已觸發傳送端通道，請設定傳輸佇列屬性 TRIGGER。

用戶端問題

用戶端應用程式可能會收到非預期的錯誤回覆碼，例如：

- 佇列管理程式無法使用
- 佇列管理程式名稱錯誤
- 連線已中斷

請查看用戶端錯誤日誌，以取得說明失敗原因的訊息。視失敗的本質而定，可能也會在伺服器上記載錯誤。

終止用戶端

即使用戶端已終止，其代理程序仍有可能保持開啟其佇列。通常這只會在短時間內，直到通訊層通知夥伴已離開為止。

錯誤日誌

視平台而定，IBM MQ 錯誤訊息會放在不同的錯誤日誌中。有下列的錯誤日誌：

- Windows
- UNIX
- z/OS

Windows Windows 的錯誤日誌

IBM MQ for Windows 使用許多錯誤日誌來擷取有關 IBM MQ 本身的作業、您啟動的任何佇列管理程式，以及來自使用中通道的錯誤資料的訊息。

儲存錯誤日誌的位置取決於佇列管理程式名稱是否為已知，以及錯誤是否與用戶端相關聯。

- 如果佇列管理程式名稱已知，且佇列管理程式可供使用：

```
MQ_INSTALLATION_PATH\QMGRS\QMgrName\ERRORS\AMQERR01.LOG
```

- 如果佇列管理程式無法使用：

```
MQ_INSTALLATION_PATH\QMGRS\@SYSTEM\ERRORS\AMQERR01.LOG
```

- 如果用戶端應用程式發生錯誤:

```
MQ_INSTALLATION_PATH\ERRORS\AMQERR01.LOG
```

在 Windows 上，您也應該檢查 Windows 應用程式事件日誌，以取得相關訊息。

Linux

UNIX

UNIX and Linux 系統上的錯誤日誌

UNIX and Linux 系統上的 IBM MQ 會使用許多錯誤日誌來擷取有關 IBM MQ 本身的作業、您啟動的任何佇列管理程式，以及來自使用中通道的錯誤資料的訊息。

儲存錯誤日誌的位置取決於佇列管理程式名稱是否為已知，以及錯誤是否與用戶端相關聯。

- 如果佇列管理程式名稱已知:

```
/var/mqm/qmgrs/QMgrName/errors
```

- 如果佇列管理程式名稱不明 (例如當接聽器或 TLS 信號交換發生問題時):

```
/var/mqm/errors
```

當安裝用戶端時，如果用戶端應用程式中發生問題，則會使用下列日誌:

- 如果用戶端應用程式發生錯誤:

```
/var/mqm/errors/
```

z/OS

z/OS 上的錯誤日誌

錯誤訊息會寫入:

- z/OS 系統主控台
- 通道起始程式工作日誌

如果您使用 z/OS 訊息處理機能來抑制訊息，則可能會抑制主控台訊息。請參閱 [在 z/OS 上規劃 IBM MQ 環境](#)。

訊息監視

如果訊息未到達其預期的目的地，您可以使用 IBM MQ 透過控制指令提供的顯示路徑應用程式 `dspmqrte`，來決定訊息透過佇列管理程式網路及其最終位置所採取的路徑。

IBM MQ 顯示路徑應用程式在 [IBM MQ 顯示路徑應用程式](#) 區段中說明。

通道鑑別記錄疑難排解

如果您在使用通道鑑別記錄時發生問題，請檢查下列資訊是否說明問題。


您呈現給佇列管理程式的位址為何？

通道呈現給佇列管理程式的位址取決於所使用的網路配接卡。例如，如果您用來取得接聽器的 CONNAME 是 "localhost"，則會呈現 127.0.0.1 作為您的位址；如果它是您電腦的實際 IP 位址，則會是您提供給佇列管理程式的位址。您可以對 127.0.0.1 及實際 IP 位址呼叫不同的鑑別規則。

將 BLOCKADDR 與通道名稱搭配使用

如果您使用 SET CHLAUTH TYPE (BLOCKADR)，則它必須具有通用通道名稱 CHLAUTH (*)，而不具有其他任何名稱。您必須使用任何通道名稱來封鎖所指定位址的存取。

z/OS 系統上的 CHLAUTH (*)

 在 z/OS 上，包含星號 (*) 的通道名稱必須以引號括住。此規則也適用於使用單一星號來符合所有通道名稱。因此，在其他平台上指定 CHLAUTH (*) 時，在 z/OS 上必須指定 CHLAUTH (*')。

SET CHLAUTH 指令透過佇列管理程式重新啟動的行為

如果 SYSTEM.CHLAUTH.DATA.QUEUE 已被刪除或變更，無法再存取它，亦即 PUT (DISABLED)，**SET CHLAUTH** 指令只會部分成功。在此實例中，**SET CHLAUTH** 會更新記憶體內快取，但在強化時將會失敗。

這表示雖然 **SET CHLAUTH** 指令放置的規則最初可能可運作，但指令的效果不會在佇列管理程式重新啟動時持續保存。使用者應該調查，確定佇列可存取，然後重新發出指令 (使用 **ACTION(REPLACE)**) 在循環佇列管理程式之前。

如果 SYSTEM.CHLAUTH.DATA.QUEUE 仍無法存取，無法載入已儲存規則的快取，且會封鎖所有通道，直到佇列及規則變成可存取。

z/OS 系統上 ADDRESS 和 ADDRLIST 的大小上限



在 z/OS 上，ADDRESS 和 ADDRLIST 欄位的大小上限是 48 個字元。部分 IPv6 位址型樣可能超過此限制，例如 '0000-ffff:0000-ffff:0000-ffff:0000-ffff:0000-ffff:0000-ffff:0000-ffff'。在此情況下，您可以改用 '*'。

如果您想要使用超過 48 個字元的型樣，請嘗試以不同方式來表示需求。例如，不指定

'0001-fffe:0001-fffe:0001-fffe:0001-fffe:0001-fffe:0001-fffe:0001-fffe:0001-fffe:0001-fffe' 作為 USERSRC (MAP) 的位址型樣，您可以指定三個規則：

- 所有位址的 USERSRC (MAP) (*)
- 位址 '0000:0000:0000:0000:0000:0000:0000:0000' 的 USERSRC (NOACCESS)
- 位址 'ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff' 的 USERSRC (NOACCESS)

指令疑難排解

在說明文字中使用特殊字元時所出現錯誤的疑難排解建議。

- **實務範例:** 當您在部分指令的說明文字中使用特殊字元時，會收到錯誤。
- **說明:** 部分字元，例如反斜線 (\) 及雙引號 (") 字元與指令一起使用時具有特殊意義。
- **解決方案:** 在特殊字元之前加上 \，亦即，如果您要在文字中輸入 \ 或 "，請輸入 \\ 或 \。並非所有字元都可以與指令一起使用。如需具有特殊意義的字元及其使用方式的相關資訊，請參閱 [具有特殊意義的字元](#)。

分散式發佈/訂閱疑難排解

使用子主題中提供的建議，可協助您在使用發佈/訂閱叢集或階層時偵測及處理問題。

開始之前

如果您的問題一般與叢集作業相關，而不是使用叢集來發佈/訂閱傳訊，請參閱第 168 頁的『佇列管理程式叢集疑難排解』。

發佈/訂閱叢集中所保留發佈的設計考量中也有一些有用的疑難排解提示。

相關資訊

[配置發佈/訂閱叢集](#)

發佈/訂閱叢集的遞送: 行為注意事項

使用這裡提供的建議，可協助您在使用叢集發佈/訂閱傳訊時偵測及處理遞送問題。

如需任何佇列管理程式叢集的狀態檢查及疑難排解的相關資訊，請參閱第 168 頁的『佇列管理程式叢集疑難排解』。

- 叢集中相同具名主題物件的所有叢集定義必須具有相同的 **CLROUTE** 設定。您可以使用下列 MQSC 指令來檢查叢集中所有主機上所有主題的 **CLROUTE** 設定：

```
display tcluster(*) clroute
```

- 除非主題物件指定 **CLUSTER** 內容的值，否則 **CLROUTE** 內容沒有作用。
- 請檢查您是否已在主題上正確地拼寫叢集名稱。在定義叢集之前，您可以先定義叢集物件 (例如主題)。因此，當您定義叢集主題時，不會對叢集名稱執行任何驗證，因為它可能尚未存在。因此，產品不會警示您拼錯叢集名稱。
- 當您設定 **CLROUTE** 內容時，如果佇列管理程式知道另一個具有不同 **CLROUTE** 設定的佇列管理程式中相同物件的叢集定義，則系統會產生 MQRCCF_CLUSTER_TOPIC_CONFLICT 異常狀況。不過，透過不同佇列管理程式上近乎同時的物件定義，或與完整儲存庫的不穩定連線功能，可能會建立不同的定義。在這種情況下，完整儲存庫佇列管理程式會仲裁，接受一項定義，並報告另一項定義的錯誤。若要取得衝突的相關資訊，請使用下列 MQSC 指令來檢查叢集中所有佇列管理程式上所有主題的叢集狀態：

```
display tcluster(*) clstate
```

狀態 **invalid** 或 **pending** (如果這不會很快變成作用中) 表示有問題。如果偵測到無效的主題定義，請識別不正確的主題定義，並將其從叢集中移除。完整儲存庫具有已接受的定義及已拒絕的定義的相關資訊，而建立衝突的佇列管理程式則有一些問題本質的指示。另請參閱 **顯示主題** 中的 **CLSTATE**。

- 在主題樹狀結構中的某個點設定 **CLROUTE** 參數，會導致其下方的整個分支以該方式遞送主題。您無法變更此分支之子分支的遞送行為。因此，在主題樹狀結構中使用不同 **CLROUTE** 設定定義較低或較高節點的主題物件會因 MQRCCF_CLUSTER_TOPIC_CONFLICT 異常狀況而遭到拒絕。
- 您可以使用下列 MQSC 指令來檢查主題樹狀結構中所有主題的主題狀態：

```
display tpstatus('#')
```

如果您在主題樹狀結構中有大量分支，則前一個指令可能會顯示不方便的大量主題的狀態。如果是這種情況，您可以改為顯示樹狀結構的可管理小分支，或樹狀結構中的個別主題。顯示的資訊包括主題字串、叢集名稱及叢集路徑設定。它還包括發佈者計數及訂閱計數 (發佈者及訂閱者的數目)，以協助您判斷此主題的使用者數目是否如您所預期。

- 變更叢集中主題的叢集遞送是對發佈/訂閱拓撲的重大變更。在已叢集化主題物件 (透過設定 **CLUSTER** 內容) 之後，您無法變更 **CLROUTE** 內容的值。必須先將該物件取消叢集 (將 **CLUSTER** 設定為 ' ')，然後您才能變更此值。將主題取消叢集會將主題定義轉換成本端主題，這將導致在某個期間內，不會將發佈資訊傳遞到遠端佇列管理程式上的訂閱；在執行此變更時應將這一點考慮在內。請參閱 **定義與另一個佇列管理程式中的叢集主題具有相同名稱的非叢集主題的作用**。如果您嘗試變更叢集 **CLROUTE** 內容的值，系統會產生 MQRCCF_CLROUTE_NOT_ALTERABLE 異常狀況。
- 對於主題主機遞送，您可以在叢集佇列管理程式範圍上新增及移除相同的叢集主題定義，以探索透過叢集的替代路徑。若要停止給定的佇列管理程式作為叢集主題的主題主機，請刪除主題物件，或使用 **PUB(DISABLED)** 設定來靜止此主題的訊息資料流量，如 **PUB** 參數的特殊處理中所討論。請勿透過將 **CLUSTER** 內容設為 ' ' 來取消叢集主題，因為移除叢集名稱會將主題定義轉換為本端主題，並防止從這個佇列管理程式使用主題時的叢集行為。請參閱 **定義與另一個佇列管理程式中的叢集主題具有相同名稱的非叢集主題的作用**。
- 當主題樹狀結構的子分支已叢集化至不同的叢集，且 **CLROUTE** 設為 **TOPICHOST** 時，您無法變更該子分支的叢集。如果在定義時間偵測到這類定義，系統會產生 MQRCCF_CLUSTER_TOPIC_CONFLICT 異常狀況。

況。同樣地，在不同叢集的較高節點上插入新叢集主題定義會產生異常狀況。由於先前說明的叢集計時間題，如果稍後偵測到這類不一致，佇列管理程式會對佇列管理程式日誌發出錯誤。

相關資訊

[配置發佈/訂閱叢集](#)

[設計發佈/訂閱叢集](#)

檢查 Proxy 訂閱位置

Proxy 訂閱可讓發佈流向遠端佇列管理程式上的訂閱者。如果您的訂閱者未取得佇列管理程式網路中其他地方所發佈的訊息，請檢查您的 Proxy 訂閱是否在您預期的位置。

遺漏 Proxy 訂閱可能顯示您的應用程式未訂閱正確的主題物件或主題字串，或主題定義有問題，或通道未執行或未正確配置。

若要顯示 Proxy 訂閱，請使用下列 MQSC 指令：

```
display sub(*) subtype(proxy)
```

Proxy 訂閱用於所有分散式發佈/訂閱拓撲 (階層和叢集)。對於主題主機遞送叢集主題，該主題的每一個主題主機上都存在 Proxy 訂閱。對於直接遞送叢集主題，Proxy 訂閱存在於叢集中的每個佇列管理程式上。也可以透過在主題上設定 `proxysub(force)` 屬性，使 Proxy 訂閱存在於網路中的每個佇列管理程式上。

另請參閱 [發佈/訂閱網路中的訂閱效能](#)。

重新同步化 Proxy 訂閱

在正常情況下，佇列管理程式會自動確保系統中的 Proxy 訂閱正確反映網路中每一個佇列管理程式上的訂閱。如有需要，您可以使用 **REFRESH QMGR TYPE (PROXYSUB)** 指令，手動重新同步化佇列管理程式的本端訂閱與它在網路中延伸的 Proxy 訂閱。不過，您應該只在特殊情況下才這麼做。

何時手動重新同步化 Proxy 訂閱

當佇列管理程式正在接收不應該傳送的訂閱，或未接收應該接收的訂閱時，您應該考慮手動重新同步化 Proxy 訂閱。不過，重新同步會暫時地在網路上建立突然的額外 Proxy 訂閱負載，此負載源自發出指令的佇列管理程式。因此，除非 IBM MQ 服務、IBM MQ 文件或錯誤記載指示您這樣做，否則請勿手動重新同步化。

如果佇列管理程式即將進行自動重新驗證，您不需要手動重新同步化 Proxy 訂閱。通常，佇列管理程式會在下列時間重新驗證具有受影響直接連接佇列管理程式的 Proxy 訂閱：

- 形成階層式連線時
- 修改主題物件上的 **PUBSCOPE** 或 **SUBSCOPE** 或 **CLUSTER** 屬性時
- 重新啟動佇列管理程式時

有時配置錯誤會導致遺漏或額外的 Proxy 訂閱：

- 如果指定最符合的主題定義時，**Subscription scope** 設為佇列管理程式，或叢集名稱空白或不正確，則可能會導致遺漏 Proxy 訂閱。請注意，**Publication scope** 不會阻止傳送 Proxy 訂閱，但會阻止將發佈資訊遞送給它們。
- 如果在 **Proxy subscription behavior** 設為強制的情況下指定最符合的主題定義，則可能會導致額外的 Proxy 訂閱。

當配置錯誤導致這些問題時，手動重新同步無法解決這些問題。在這些情況下，請修正配置。

下列清單說明您應該手動重新同步化 Proxy 訂閱的異常狀況：

- 在發佈/訂閱叢集中的佇列管理程式上發出 **REFRESH CLUSTER** 指令之後。
- 當佇列管理程式錯誤日誌中的訊息告訴您執行 **REFRESH QMGR TYPE (REPOS)** 指令時。
- 當佇列管理程式無法正確傳播其 Proxy 訂閱時，可能是因為通道已停止，且無法將所有訊息排入佇列以進行傳輸，或因為操作員錯誤導致從 `SYSTEM.CLUSTER.TRANSMIT.QUEUE` 佇列中錯誤地刪除訊息。

- 當不正確地從其他系統佇列中刪除訊息時。
- 在 Proxy 訂閱上發出錯誤的 **DELETE SUB** 指令時。
- 作為災難回復的一部分。


如何手動重新同步化 Proxy 訂閱

首先更正原始問題 (例如重新啟動通道)，然後在佇列管理程式上發出下列指令：

```
REFRESH QMGR TYPE(PROXYSUB)
```

當您發出這個指令時，佇列管理程式會向其每一個直接連接的佇列管理程式傳送一份清單，列出其本身應該存在 Proxy 訂閱的主題字串。然後，直接連接的佇列管理程式會更新其保留的 Proxy 訂閱，以符合清單。接下來，直接連接的佇列管理程式會將自己應該存在 Proxy 訂閱的主題字串清單傳回給原始佇列管理程式，原始佇列管理程式會相應地更新其保留的 Proxy 訂閱。

重要使用注意事項：

- 受影響的訂閱無法回復因未備妥 Proxy 訂閱而遺失的發佈。
- 重新同步需要佇列管理程式啟動與其他佇列管理程式的通道。如果您在叢集中使用直接遞送，或使用主題主機遞送，且在主題主機佇列管理程式上發出此指令，則佇列管理程式會啟動與叢集中所有其他佇列管理程式的通道，即使未執行發佈/訂閱工作也一樣。因此，您正在重新整理的佇列管理程式必須有足夠的能力來處理與叢集中所有其他佇列管理程式的通訊。
-  如果在 z/OS 上未執行 CHINIT 時發出此指令，則會在 CHINIT 啟動時將指令排入佇列並處理。

相關資訊

[檢查分散式網路的非同步指令是否已完成發佈/訂閱叢集的 REFRESH CLUSTER 考量](#)

分散式發佈/訂閱網路中的迴圈偵測

在分散式發佈/訂閱網路中，發佈和 Proxy 訂閱不能迴圈是很重要的，因為這會導致網路癱瘓，連接的訂閱者會收到相同原始發佈的多個副本。

發佈/訂閱網路中的 Proxy 訂閱中說明的 Proxy 訂閱聚集系統不會阻止形成迴圈，但會防止 Proxy 訂閱的永久迴圈。因為發佈的傳播取決於 Proxy 訂閱的存在，所以它們可以進入永久迴圈。IBM MQ 使用下列技術來防止發佈永久迴圈：

當發佈資訊在發佈/訂閱拓撲中移動時，每一個佇列管理程式會將唯一指紋新增至訊息標頭。每當發佈/訂閱佇列管理程式從另一個發佈/訂閱佇列管理程式接收發佈時，即會檢查訊息標頭中所保留的指紋。如果它自己的指紋已存在，則發佈已在迴圈中完全循環，因此佇列管理程式會捨棄訊息，並將項目新增至錯誤日誌。

註：在迴圈中，會在迴圈周圍雙向傳播發佈資訊，且迴圈中的每一個佇列管理程式會在原始佇列管理程式捨棄迴圈發佈資訊之前接收這兩個發佈資訊。這會導致訂閱應用程式接收重複的發佈副本，直到迴圈中斷為止。

迴圈偵測指紋格式

迴圈偵測指紋會插入 RFH2 標頭或流程中，作為 8.0 通訊協定的一部分。RFH2 程式設計師需要瞭解標頭，並完整傳遞指紋資訊。舊版 IBM Integration Bus 使用不包含指紋資訊的 RFH1 標頭。

```
<ibm>
  <Rfp>uuid1</Rfp>
  <Rfp>uuid2</Rfp>
  <Rfp>uuid3</Rfp>
  .
  .
  .
</ibm>
```

<ibm> is the name of the folder that holds the list of routing fingerprints containing the unique user identifier (uuid) of each queue manager that has been visited.

Every time that a message is published by a queue manager, it adds its uuid into the <ibm> folder using the <Rfp> (routing fingerprint) tag. Whenever a publication is received, IBM MQ uses the message properties API to iterate through the <Rfp> tags to see if that particular uuid value is present. 由於 IBM MQ 的 WebSphere Platform Messaging 元件在使用排入佇列的發佈/訂閱介面時透過通道及 RFH2 訂閱連接至 IBM Integration Bus 的方式， IBM MQ 在透過該路徑接收發佈時也會建立指紋。

目標是不將任何 RFH2 遞送至應用程式 (如果它不預期)， 只是因為我們已在指紋資訊中新增。

Whenever an RFH2 is converted into message properties, it will also be necessary to convert the <ibm> folder; this removes the fingerprint information from the RFH2 that is passed on or delivered to applications that have used the IBM MQ 7.0, or later, API.

JMS 應用程式看不到指紋資訊， 因為 JMS 介面不會從 RFH2 中擷取該資訊， 因此不會將它交給其應用程式。

Rfp 訊息內容是使用 `propDesc.CopyOptions = MQCOPY_FORWARD` and `MQCOPY_PUBLISH` 建立的。這會影響應用程式接收然後重新發佈相同的訊息。這表示此類應用程式可以使用 `PutMsgOpts.Action = MQACTP_FORWARD` 來繼續遞送指紋的鏈結， 但必須適當地編碼以從鏈結中移除其自己的指紋。依預設， 應用程式會使用 `PutMsgOpts.Action = MQACTP_NEW` 並啟動新的鏈結。

Java 和 JMS 疑難排解

使用這裡提供的建議， 可協助您解決在使用 Java 或 JMS 應用程式時可能產生的一般問題。

關於這項作業

本節中的子主題提供建議， 以協助您偵測及處理在下列情況下可能遇到的問題：

- 使用 IBM MQ 資源配接器時
- 連接至具有指定提供者版本的佇列管理程式時

相關概念

[第 78 頁的『追蹤 IBM MQ classes for JMS 應用程式』](#)

提供 IBM MQ classes for JMS 中的追蹤機能， 以協助「IBM 支援中心」診斷客戶問題。 各種內容會控制此機能的行為。

[第 85 頁的『追蹤 IBM MQ 資源配接器』](#)

ResourceAdapter 物件封裝 IBM MQ 資源配接器的廣域內容。 如果要啟用 IBM MQ 資源配接器的追蹤， 必須在 ResourceAdapter 物件中定義內容。

[第 87 頁的『追蹤其他 IBM MQ Java 元件』](#)

對於 IBM MQ 的 Java 元件 (例如 IBM MQ Explorer 及 IBM MQ Transport for SOAP 的 Java 實作)， 診斷資訊是使用標準 IBM MQ 診斷機能或由 Java 診斷類別來輸出。

相關工作

[第 82 頁的『追蹤 IBM MQ classes for Java 應用程式』](#)

提供 IBM MQ classes for Java 中的追蹤機能， 以協助「IBM 支援中心」診斷客戶問題。 各種內容會控制此機能的行為。

相關資訊

[使用 IBM MQ classes for JMS](#)

[使用 IBM MQ 資源配接器](#)

[使用 IBM MQ classes for Java](#)

IBM MQ classes for JMS 問題疑難排解

您可以透過執行安裝驗證程式， 並使用追蹤及日誌機能來調查問題。

如果程式未順利完成， 請執行其中一個安裝驗證程式 (如 [適用於 IBM MQ 類別的點對點 IVT for JMS](#) 及 [適用於 IBM MQ 類別的發佈/訂閱 IVT for JMS](#) 中所述)， 並遵循診斷訊息中提供的建議。

相關概念

[第 78 頁的『追蹤 IBM MQ classes for JMS 應用程式』](#)

提供 IBM MQ classes for JMS 中的追蹤機能，以協助「IBM 支援中心」診斷客戶問題。各種內容會控制此機能的行為。

IBM MQ classes for JMS 的記載錯誤

依預設，日誌輸出會傳送至 mqjms.log 檔案。您可以將它重新導向至特定的檔案或目錄。

提供 IBM MQ classes for JMS 日誌機能以報告嚴重問題，特別是可能指出配置錯誤而非程式設計錯誤的問題。依預設，日誌輸出會傳送至 JVM 工作目錄中的 mqjms.log 檔案。

您可以設定內容 com.ibm.msg.client.commonservices.log.outputName，將日誌輸出重新導向至另一個檔案。此內容的值可以是：

- 單一路徑名稱。
- 以逗點區隔的路徑名稱清單 (所有資料都會記載至所有檔案)。

每一個路徑名稱可以是：

- 絕對或相對。
- stderr 或 System.err，代表標準誤串流。
- stdout 或 System.out，代表標準輸出串流。

如果內容的值識別目錄，則日誌輸出會寫入該目錄中的 mqjms.log。如果內容的值識別特定檔案，則會將日誌輸出寫入該檔案。

您可以在 IBM MQ classes for JMS 配置檔中設定此內容，或在 **java** 指令上設定此內容作為系統內容。在下列範例中，內容設為系統內容，並識別特定檔案：

```
java -Djava.library.path= library_path
-Dcom.ibm.msg.client.commonservices.log.outputName=/mydir/mylog.txt
MyAppClass
```

在指令中，*library_path* 是包含 IBM MQ classes for JMS 程式庫的目錄路徑 (請參閱 [配置 Java 原生介面 \(JNI\) 程式庫](#))。

您可以將 com.ibm.msg.client.commonservices.log.status 內容設為 OFF，來停用日誌輸出。此內容的預設值為 ON。

可以設定值 System.err 和 System.out，以將日誌輸出傳送至 System.err 和 System.out 串流。

JMS 提供者版本疑難排解

請使用這裡提供的建議，來協助您解決當您連接至具有指定提供者版本的佇列管理程式時可能發生的一般問題。

此連線錯誤不支援 JMS 2.0 函數

- **錯誤碼:** JMSCC5008
- **實務範例:** 您收到 此連線不支援 JMS 2.0 函數 錯誤。
- **說明:** 只有在連接至使用 IBM MQ 傳訊提供者第 8 版模式的 IBM MQ 8.0 或更新版本佇列管理程式時，才支援使用 JMS 2.0 功能。
- **解決方案:** 將應用程式變更為不使用 JMS 2.0 功能，或確保應用程式連接至使用 IBM MQ 傳訊提供者第 8 版模式的 IBM MQ 8.0 佇列管理程式。

此連線錯誤不支援 JMS 2.0 API

- **錯誤碼:** JMSCC5007
- **實務範例:** 您收到 JMS 2.0 API 不支援此連線 錯誤。
- **說明:** 只有在連接至使用 IBM MQ 傳訊提供者「一般」或「第 8 版」模式的 IBM WebSphere MQ 7 或 8 佇列管理程式時，才支援使用 JMS 2.0 API。例如，如果您嘗試連接至 IBM WebSphere MQ 6 佇列管理程

式，或您使用移轉模式進行連接，則可能會收到此錯誤。如果指定 SHARECNV (0) 或 PROVIDER_VERSION=6，通常會發生這種情況。

- **解決方案:** 將應用程式變更為不使用 JMS 2.0 API，或確保應用程式使用 IBM MQ 傳訊提供者「一般」或「第 8 版」模式連接至 IBM WebSphere MQ 7 或 8 佇列管理程式。

佇列管理程式指令層次不符合所要求的提供者版本錯誤

- **錯誤碼:** JMSFMQ0003
- **實務範例:** 您收到 佇列管理程式指令層次不符合所要求的提供者版本錯誤。
- **說明:** Connection Factory 的提供者版本內容中指定的佇列管理程式版本與所要求的佇列管理程式不相容。例如，您可能已指定 PROVIDER_VERSION=8，並嘗試連接至指令層次小於 800 的佇列管理程式，例如 750。
- **解決方案:** 修改 Connection Factory，以連接至可支援所需提供者版本的佇列管理程式。

如需提供者版本的相關資訊，請參閱 [配置 JMS PROVIDERVERSION](#) 內容。

JMS 中的 PCF 處理

IBM MQ 「可程式變更格式 (PCF)」訊息是一種彈性且功能強大的方式，可用來查詢及修改佇列管理程式的屬性，而 IBM MQ classes for Java 中所提供的 PCF 類別可讓您方便地在 Java 應用程式中存取其功能。也可以從 IBM MQ classes for JMS 存取此功能，但有潛在問題。

在 JMS 中處理 PCF 回應的一般模型

在 JMS 中處理 PCF 回應的一般方法是擷取訊息的位元組有效負載，將它包裝在 `DataInputStream` 中，並將其傳遞給 `com.ibm.mq.headers.pcf.PCFMessage` 建構子。

```
Message m = consumer.receive(10000);
//Reconstitute the PCF response.
ByteArrayInputStream bais =
    new ByteArrayInputStream(((BytesMessage)m).getBody(byte[]).class));
DataInput di = new DataInputStream(bais);
PCFMessage pcfResponseMessage = new PCFMessage(di);
```

如需一些範例，請參閱 [使用 IBM MQ 標頭套件](#)。

不幸的是，這對所有平台都不是完全可靠的方法-一般而言，此方法適用於大序排列法平台，但不適用於小序排列法平台。

有什麼問題嗎？

問題是在剖析訊息標頭時，`PCFMessage` 類別必須處理數值編碼的問題-標頭包含採用某種大序排列法或小序排列法編碼的長度欄位。

如果您將純 `DataInputStream` 傳遞給建構子，則 `PCFMessage` 類別沒有編碼的良好指示，且必須採用預設值，很可能不正確。

如果發生此狀況，您可能會在建構子中看到 "MQRCCF_STRUCTURE_TYPE_ERROR" (原因碼 3013):

```
com.ibm.mq.headers.MQDataException: MQJE001: Completion Code '2', Reason '3013'.
    at com.ibm.mq.headers.pcf.PCFParameter.nextParameter(PCFParameter.java:167)
    at com.ibm.mq.headers.pcf.PCFMessage.initialize(PCFMessage.java:854)
    at com.ibm.mq.headers.pcf.PCFMessage.<init>(PCFMessage.java:156)
```

此訊息幾乎總是表示編碼已被誤解。可能的原因是已讀取的資料是小序排列法資料，已解譯為大序排列法。

解決方案

避免此問題的方法是傳遞 `PCFMessage` 建構子內容，以告知建構子它所使用資料的數值編碼。

若要這樣做，請從收到的資料建立 `MQMessage`。

下列程式碼是您可能使用之程式碼的大綱範例。



小心: 程式碼只是一個大綱範例，不包含任何錯誤處理資訊。

```
// get a response into a JMS Message
Message receivedMessage = consumer.receive(10000);
BytesMessage bytesMessage = (BytesMessage) receivedMessage;
byte[] bytesreceived = new byte[(int) bytesMessage.getBodyLength()];
bytesMessage.readBytes(bytesreceived);

// convert to MQMessage then to PCFMessage
MQMessage mqMsg = new MQMessage();
mqMsg.write(bytesreceived);
mqMsg.encoding = receivedMessage.getIntProperty("JMS_IBM_Encoding");
mqMsg.format = receivedMessage.getStringProperty("JMS_IBM_Format");
mqMsg.seek(0);

PCFMessage pcfMsg = new PCFMessage(mqMsg);
```

JMS 連線儲存區錯誤處理

連線儲存區錯誤處理是由清除原則的各種方法來執行。

如果在應用程式使用 JMS 提供者的 JMS 連線時偵測到錯誤，則連線儲存區清除原則會開始運作。連線管理程式可以：

- 僅關閉發生問題的連線。這稱為 **FailingConnectionOnly** 清除原則，且是預設行為。

從 **Factory** 建立的任何其他連線 (亦即，其他應用程式正在使用的連線，以及在 **Factory** 的可用儲存區中的連線) 都將單獨保留。

- 關閉發生問題的連線，丟棄 **Factory** 的可用儲存區中的任何連線，並將任何使用中連線標示為過時。

下次使用連線的應用程式嘗試執行連線型作業時，應用程式會收到 **StaleConnectionException**。針對此行為，請將清除原則設為 **Entire Pool**。

清除原則-僅連線失敗

使用 **MDB** 接聽器埠如何使用連線儲存區中說明的範例。兩個 **MDB** 會部署到應用程式伺服器中，每一個都使用不同的接聽器埠。接聽器埠都使用 **jms/CF1 Connection Factory**。

在 600 秒之後，您會停止第一個接聽器，且此接聽器埠使用的連線會回到連線儲存區。

如果第二個接聽器在輪詢 **JMS** 目的地時發生網路錯誤，則接聽器埠會關閉。因為 **jms/CF1 Connection Factory** 的清除原則設為 **FailingConnectionOnly**，所以連線管理程式只會擲出第二個接聽器所使用的連線。可用儲存區中的連線會保留在它所在的位置。

如果您現在重新啟動第二個接聽器，連線管理程式會將連線從可用儲存區傳遞至接聽器。

清除原則-整個儲存區

在此狀況下，假設您在應用程式伺服器中安裝了三個 **MDB**，每一個都使用自己的接聽器埠。接聽器埠已從 **jms/CF1 Factory** 建立連線。在一段時間之後，您會停止第一個接聽器，且其連線 **c1** 會放入 **jms/CF1** 可用儲存區中。

當第二個接聽器偵測到網路錯誤時，它會自行關閉並關閉 **c2**。連線管理程式現在會關閉可用儲存區中的連線。不過，第三個接聽器所使用的連線仍然存在。

您應該將清除原則設為什麼？

如先前所述，**JMS** 連線儲存區的清除原則預設值為 **FailingConnectionOnly**。

不過，將清除原則設為 **EntirePool** 是較好的選項。在大部分情況下，如果應用程式在其與 **JMS** 提供者的連線上偵測到網路錯誤，則可能是所有從相同 **Connection Factory** 建立的開啟連線都有相同的問題。

如果清除原則設為 `FailingConnectionOnly`，連線管理程式會保留可用儲存區中的所有連線。下次應用程式嘗試建立與 JMS 提供者的連線時，連線管理程式會從可用儲存區中傳回一個 (如果有的話)。不過，當應用程式嘗試使用連線時，它會遇到與第一個應用程式相同的網路問題。

現在，請考量清除原則設為 `EntirePool` 的相同狀況。當第一個應用程式遇到網路問題時，連線管理程式會捨棄失敗的連線，並關閉該 `Factory` 的可用儲存區中的所有連線。

當新的應用程式啟動並嘗試從 `Factory` 建立連線時，連線管理程式會嘗試建立新的連線，因為可用儲存區是空的。假設網路問題已解決，則傳回至應用程式的連線是有效的。

嘗試建立 JMS 環境定義時發生連線儲存區錯誤

在您嘗試建立 JMS 環境定義時如果發生錯誤，可以從錯誤訊息判斷是頂層儲存區或底層儲存區發生問題。

儲存區如何用於環境定義

使用「連線」和「階段作業」時，每個類型的物件都有對應的儲存區，而「環境定義」也遵循類似的模型。

使用分散式交易的一般應用程式在同一個交易中同時涉及傳訊與非傳訊工作量。

假設目前沒有工作在進行，並且應用程式使它成為第一個 `createConnection` 方法呼叫，則會在連線儲存區的同等項目 (頂層儲存區) 中建立環境定義 `Facade` 或 `Proxy`。在階段作業儲存區的同等項目中建立另一個物件。此第二個物件會封裝基礎 JMS 環境定義 (底層儲存區)。

儲存區的概念是用來允許應用程式擴充。許多執行緒可存取受限的資源集。在此範例中，另一個執行緒會執行 `createContext` 方法呼叫來從儲存區取得環境定義。若其他執行緒仍在執行傳訊工作，則會將頂層儲存區擴充，為提出要求的執行緒提供額外的環境定義。

在執行緒要求環境定義，且傳訊工作已完成，但非傳訊工作未完成，因此交易未完成的情況下，即會擴充底層儲存區。頂層環境定義 `Proxy` 會維持指派給交易，直到該交易解決為止，因此無法指派給另一個交易。

在底層儲存區變得滿載的情況下，這表示非傳訊工作可能會耗費相當長的時間。

在頂層儲存區變得滿載的情況下，這表示整體傳訊工作會耗費一些時間，因此應擴充儲存區。

識別發生錯誤的來源儲存區

您可以從錯誤訊息文字判斷發生錯誤的來源儲存區：

- 針對頂層儲存區，訊息文字為**無法建立環境定義**。此訊息表示頂層儲存區充滿了環境定義 `Proxy` 物件，這些物件目前有執行傳訊的執行中交易。
- 針對底層儲存區，訊息文字為**無法設定新的 JMSContext**。此訊息表示雖然連接 `Proxy` 可用，但仍需等待非傳訊工作完成。

頂層儲存區範例

```
*****[8/19/16 10:10:48:643 UTC] 000000a2
LocalException E CNTR0020E: EJB threw an unexpected (non-declared) exception during
invocation of method "onMessage" on bean
"BeanId(SibSVTLiteMDB#SibSVTLiteMDBXA_RecoveryEJB_undeployed.jar#QueueReceiver, null)".
Exception data: javax.jms.JMSRuntimeException: Failed to create context
    at com.ibm.ejs.jms.JMSCMUtils.mapToJMSRuntimeException(JMSCMUtils.java:522)
    at
com.ibm.ejs.jms.JMSCConnectionFactoryHandle.createContextInternal(JMSCConnectionFactoryHandle.java:4
49)
    at
com.ibm.ejs.jms.JMSCConnectionFactoryHandle.createContext(JMSCConnectionFactoryHandle.java:335)
    at sib.test.svtlite.mdb.xa.SVTMDBBase.sendReplyMessage(SVTMDBBase.java:554)
    at sib.test.svtlite.mdb.xa.QueueReceiverBean.onMessage(QueueReceiverBean.java:128)
    at
sib.test.svtlite.mdb.xa.MDBProxyQueueReceiver_37ea5ce9.onMessage(MDBProxyQueueReceiver_37ea5ce9.j
ava)
    at
com.ibm.mq.connector.inbound.MessageEndpointWrapper.onMessage(MessageEndpointWrapper.java:151)
    at com.ibm.mq.jms.MQSession$FacadeMessageListener.onMessage(MQSession.java:129)
    at com.ibm.msg.client.jms.internal.JmsSessionImpl.run(JmsSessionImpl.java:3236)
    at com.ibm.mq.jms.MQSession.run(MQSession.java:937)
    at com.ibm.mq.connector.inbound.ASFWorkImpl.doDelivery(ASFWorkImpl.java:104)
    at com.ibm.mq.connector.inbound.AbstractWorkImpl.run(AbstractWorkImpl.java:233)
```

```

    at com.ibm.ejs.j2c.work.WorkProxy.run(WorkProxy.java:668)
    at com.ibm.ws.util.ThreadPool$Worker.run(ThreadPool.java:1892)
    Caused by: com.ibm.websphere.ce.j2c.ConnectionWaitTimeoutException:
CWTE_NORMAL_J2CA1009
    at com.ibm.ejs.j2c.FreePool.createOrWaitForConnection(FreePool.java:1783)
    at com.ibm.ejs.j2c.PoolManager.reserve(PoolManager.java:3896)
    at com.ibm.ejs.j2c.PoolManager.reserve(PoolManager.java:3116)
    at com.ibm.ejs.j2c.ConnectionManager.allocateMCWrapper(ConnectionManager.java:1548)
    at com.ibm.ejs.j2c.ConnectionManager.allocateConnection(ConnectionManager.java:1031)
    at
com.ibm.ejs.jms.JMSConnectionFactoryHandle.createContextInternal(JMSConnectionFactoryHandle.java:4
43)
    ... 12 more

```

底層儲存區範例

```

*****
[8/19/16 9:44:44:754 UTC] 000000ac SibMessage W   [:] CWSJY0003W: MQJCA4004: Message delivery to
an MDB
'sib.test.svt.lite.mdb.xa.MDBProxyQueueReceiver_37ea5ce9@505d4b68
(BeanId(SibSVTLiteMDB#SibSVTLiteMDBXA_RecoveryEJB_undeployed.jar#QueueReceiver, null))' failed
with exception:
'nested exception is: javax.jms.JMSRuntimeException: Failed to set up new JMSContext'.
^C[root@username-instance-2 server1]# vi SystemOut.log
    : com.ibm.ejs.j2c.work.WorkProxy.run(WorkProxy.java:668)
    : com.ibm.ws.util.ThreadPool$Worker.run(ThreadPool.java:1892)
    因 [1] --> 訊息: javax.jms.JMSRuntimeException: 無法設定新的 JMSContext
    Class : class javax.jms.JMSRuntimeException
    Stack :
com.ibm.ejs.jms.JMSCMUtils.mapToJMSRuntimeException(JMSCMUtils.java:522)
    :
com.ibm.ejs.jms.JMSContextHandle.setupInternalContext(JMSContextHandle.java:241)
    :
com.ibm.ejs.jms.JMSManagedConnection.getConnection(JMSManagedConnection.java:783)
    :
com.ibm.ejs.j2c.MCWrapper.getConnection(MCWrapper.java:2336)
    :
com.ibm.ejs.j2c.ConnectionManager.allocateConnection(ConnectionManager.java:1064)
    :
com.ibm.ejs.jms.JMSConnectionFactoryHandle.createContextInternal(JMSConnectionFactoryHandle.java:4
43)
    :
com.ibm.ejs.jms.JMSConnectionFactoryHandle.createContext(JMSConnectionFactoryHandle.java:335)
    :
sib.test.svt.lite.mdb.xa.SVTMDBBase.sendReplyMessage(SVTMDBBase.java:554)
    :
sib.test.svt.lite.mdb.xa.QueueReceiverBean.onMessage(QueueReceiverBean.java:128)
    :
sib.test.svt.lite.mdb.xa.MDBProxyQueueReceiver_37ea5ce9.onMessage(MDBProxyQueueReceiver_37ea5ce9.j
ava:-1)
    :
com.ibm.mq.connector.inbound.MessageEndpointWrapper.onMessage(MessageEndpointWrapper.java:151)
    :
com.ibm.mq.jms.MQSession$FacadeMessageListener.onMessage(MQSession.java:129)
    :
com.ibm.msg.client.jms.internal.JmsSessionImpl.run(JmsSessionImpl.java:3236)
    : com.ibm.mq.jms.MQSession.run(MQSession.java:937)
    :
com.ibm.mq.connector.inbound.ASFWorkImpl.doDelivery(ASFWorkImpl.java:104)
    :
com.ibm.mq.connector.inbound.AbstractWorkImpl.run(AbstractWorkImpl.java:233)
    : com.ibm.ejs.j2c.work.WorkProxy.run(WorkProxy.java:668)
    : com.ibm.ws.util.ThreadPool$Worker.run(ThreadPool.java:1892)
    原因: [2] --> 訊息: com.ibm.websphere.ce.j2c.ConnectionWaitTimeoutException:
CWTE_NORMAL_J2CA1009
    Class : class
com.ibm.websphere.ce.j2c.ConnectionWaitTimeoutException
    Stack : com.ibm.ejs.j2c.FreePool.createOrWaitForConnection(FreePool.java:1783)
    :
com.ibm.ejs.j2c.PoolManager.reserve(PoolManager.java:3840)
    : com.ibm.ejs.j2c.PoolManager.reserve(PoolManager.java:3116)
    :
com.ibm.ejs.j2c.ConnectionManager.allocateMCWrapper(ConnectionManager.java:1548)
    :
com.ibm.ejs.j2c.ConnectionManager.allocateConnection(ConnectionManager.java:1031)
    :
com.ibm.ejs.jms.JMSContextHandle.setupInternalContext(JMSContextHandle.java:222)
    :
com.ibm.ejs.jms.JMSManagedConnection.getConnection(JMSManagedConnection.java:783)
    :

```

```

com.ibm.ejs.j2c.MCWrapper.getConnection(MCWrapper.java:2336)
:
com.ibm.ejs.j2c.ConnectionManager.allocateConnection(ConnectionManager.java:1064)
:
com.ibm.ejs.jms.JMSConnectionFactoryHandle.createContextInternal(JMSConnectionFactoryHandle.java:443)
:
com.ibm.ejs.jms.JMSConnectionFactoryHandle.createContext(JMSConnectionFactoryHandle.java:335)
:
sib.test.svtlite.mdb.xa.SVTMDBBase.sendReplyMessage(SVTMDBBase.java:554)
:
sib.test.svtlite.mdb.xa.QueueReceiverBean.onMessage(QueueReceiverBean.java:128)
:
sib.test.svtlite.mdb.xa.MDBProxyQueueReceiver_37ea5ce9.onMessage(MDBProxyQueueReceiver_37ea5ce9.java:-1)
:
com.ibm.mq.connector.inbound.MessageEndpointWrapper.onMessage(MessageEndpointWrapper.java:151)
:
com.ibm.mq.jms.MQSession$FacadeMessageListener.onMessage(MQSession.java:129)
:
com.ibm.msg.client.jms.internal.JmsSessionImpl.run(JmsSessionImpl.java:3236)
:   com.ibm.mq.jms.MQSession.run(MQSession.java:937)
:
com.ibm.mq.connector.inbound.ASFWorkImpl.doDelivery(ASFWorkImpl.java:104)
:
com.ibm.mq.connector.inbound.AbstractWorkImpl.run(AbstractWorkImpl.java:233)
:   com.ibm.ejs.j2c.work.WorkProxy.run(WorkProxy.java:668)
:   com.ibm.ws.util.ThreadPool$Worker.run(ThreadPool.java:1892)

```

疑難排解 JMSSC0108 訊息

當您使用以「應用程式伺服器機能 (ASF)」模式執行的啟動規格及 WebSphere Application Server 接聽器埠時，可以採取一些步驟來防止 JMSSC0108 訊息發生。

當您使用以 ASF 模式 (預設作業模式) 執行的啟動規格和 WebSphere Application Server 接聽器埠時，應用程式伺服器日誌檔中可能會出現下列訊息：

JMSSC0108: 適用於 JMS 的 IBM MQ 類別偵測到一則訊息，已備妥可非同步遞送至應用程式。嘗試遞送時，訊息不再可用。

請使用本主題中的資訊來瞭解此訊息出現的原因，以及您可以採取的可能步驟來防止它發生。

啟動規格和接聽器埠如何偵測及處理訊息

啟動規格或 WebSphere Application Server 接聽器埠會在啟動時執行下列步驟：

1. 建立與已設定要使用之佇列管理程式的連線。
2. 在已配置要監視的佇列管理程式上開啟 JMS 目的地。
3. 瀏覽該目的地以取得訊息。

當偵測到訊息時，啟動規格或接聽器埠會執行下列步驟：

1. 建構代表訊息的內部訊息參照。
2. 從其內部伺服器階段作業儲存區取得伺服器階段作業。
3. 載入含有訊息參照的伺服器階段作業。
4. 排程應用程式伺服器「工作管理程式」的部分工作，以執行伺服器階段作業並處理訊息。

然後，啟動規格或接聽器埠會重新回到監視目的地，並尋找另一個要處理的訊息。

應用程式伺服器「工作管理程式」會執行啟動規格或接聽器埠在新伺服器階段作業執行緒上所提交的工作。啟動時，執行緒會完成下列動作：

- 啟動本端或廣域 (XA) 交易，視訊息驅動 Bean 是否需要 XA 交易而定，如訊息驅動 Bean 的部署描述子中所指定。
- 透過發出破壞性 MQGET API 呼叫，從目的地取得訊息。
- 執行訊息驅動 Bean 的 onMessage() 方法。
- 一旦 onMessage() 方法完成，即完成區域或廣域交易。
- 將伺服器階段作業傳回伺服器階段作業儲存區。

為何會出現 JMSCC0108 訊息，以及如何阻止它

主要啟動規格或接聽器埠執行緒會瀏覽目的地上的訊息。然後，它會要求「工作管理員」啟動新的執行緒，以破壞性方式取得訊息並處理它。這表示主要啟動規格或接聽器埠執行緒可以在目的地找到訊息，且在伺服器階段作業執行緒嘗試取得訊息時不再可用。如果發生這種情況，伺服器階段作業執行緒會將下列訊息寫入應用程式伺服器的日誌檔：

JMSCC0108: 適用於 JMS 的 IBM MQ 類別偵測到一則訊息，已備妥可非同步遞送至應用程式。嘗試遞送時，訊息不再可用。

當伺服器階段作業執行緒嘗試取得訊息時，訊息已不在目的地，有兩個原因：

- 原因 1: 另一個應用程式已耗用訊息
- 原因 2: 訊息已過期

原因 1: 另一個應用程式已耗用訊息

如果兩個以上啟動規格及/或接聽器埠正在監視相同的目的地，則它們可能會偵測到相同的訊息並嘗試處理它。當發生這種情況時：

- 由一個啟動規格或接聽器埠啟動的伺服器階段作業執行緒會取得訊息，並將它遞送至訊息驅動 Bean 進行處理。
- 由另一個啟動規格或接聽器埠啟動的伺服器階段作業執行緒會嘗試取得訊息，並發現它已不在目的地。

如果啟動規格或接聽器埠以下列任何方式連接至佇列管理程式，則會標示主要啟動規格或接聽器埠執行緒偵測到的訊息：

- 任何平台上的佇列管理程式，使用 IBM MQ 傳訊提供者標準模式。
- 任何平台上的佇列管理程式，使用 IBM MQ 傳訊提供者標準模式且有限制
- 在 z/OS 上執行的佇列管理程式，使用 IBM MQ 傳訊提供者移轉模式。

標示訊息會阻止任何其他啟動規格或接聽器埠看到該訊息，並嘗試處理它。

依預設，訊息會標示五秒。在偵測並標示訊息之後，五秒計時器即會啟動。在這五秒期間，必須執行下列步驟：

- 啟動規格或接聽器埠必須從伺服器階段作業儲存區取得伺服器階段作業。
- 伺服器階段作業必須載入要處理之訊息的詳細資料。
- 必須排定工作。
- 「工作管理員」必須處理工作要求並啟動伺服器階段作業執行緒。
- 伺服器階段作業執行緒需要啟動區域或廣域交易。
- 伺服器階段作業執行緒需要破壞性地取得訊息。

在忙碌的系統上，執行這些步驟可能需要 5 秒以上的時間。如果發生此情況，則會釋放訊息上的標記。這表示其他啟動規格或接聽器埠現在可以看到訊息，並且可能嘗試處理它，這可能會導致 JMSCC0108 訊息寫入應用程式伺服器的日誌檔。

在此情況下，您應該考量下列選項：

- 增加佇列管理程式內容「訊息標示瀏覽間隔 (MARKINT)」的值，以提供最初偵測到訊息的啟動規格或接聽器埠更多時間來取得它。理想情況下，內容應該設為大於訊息驅動 Bean 處理訊息所花時間的值。這表示如果主要啟動規格或接聽器埠執行緒因所有伺服器階段作業都忙於處理訊息而封鎖等待伺服器階段作業，則當伺服器階段作業變成可用時，仍應標示該訊息。請注意，MARKINT 內容設定在佇列管理程式上，因此適用於在該佇列管理程式上瀏覽訊息的所有應用程式。
- 增加啟動規格或接聽器埠所使用的伺服器階段作業儲存區大小。這表示有更多伺服器階段作業可用來處理訊息，這應該可確保訊息可以在指定的標示間隔內處理。在此方法中要注意的一點是，啟動規格或接聽器埠現在將能夠同時處理更多訊息，這可能會影響應用程式伺服器的整體效能。

Multi 如果啟動規格或接聽器埠使用 IBM MQ 傳訊提供者移轉模式 連接至在 IBM MQ for Multiplatforms 上執行的佇列管理程式，則標示功能無法使用。這表示無法防止兩個以上啟動規格及/或接聽器埠偵測到相同的訊息並嘗試處理它。在此狀況下，預期會有 JMSCC0108 訊息。

原因 2: 訊息已過期

產生 JMSSC0108 訊息的另一個原因是，在啟動規格或接聽器埠所偵測到的訊息與伺服器階段作業所耗用的訊息之間，該訊息已過期。如果發生這種情況，當伺服器階段作業執行緒嘗試取得訊息時，會發現它不再存在，因此會報告 JMSSC0108 訊息。

在這裡，增加啟動規格或接聽器埠所使用的伺服器階段作業儲存區大小可能有幫助。增加伺服器階段作業儲存區大小表示有更多伺服器階段作業可用來處理訊息，這可能表示訊息會在到期之前處理。請務必注意，啟動規格或接聽器埠現在能夠同時處理更多訊息，這可能會影響應用程式伺服器的整體效能。

WebSphere Application Server SystemOut.log 檔案中的 CWSJY0003W 警告訊息

當 MDB 處理來自 IBM WebSphere MQ 的 JMS 訊息時，CWSJY0003W 警告訊息會記載在 WebSphere Application Server SystemOut.log 檔中。

症狀

CWSJY0003W: IBM WebSphere MQ classes for JMS 嘗試取得先前已使用「透過標示進行瀏覽」標示的訊息以遞送至訊息接聽器，但此訊息無法使用。

原因

啟動規格及在「應用程式伺服器機能 (ASF)」模式中執行的接聽器埠用於監視 IBM WebSphere MQ 佇列管理程式上管理的佇列或主題。先瀏覽佇列或主題中的訊息。找到訊息時，會啟動新的執行緒，以破壞性地取得訊息，及將訊息傳遞至訊息驅動 Bean 應用程式的實例以進行處理。

瀏覽訊息時，佇列管理程式會標示訊息一段時間，並有效地向其他應用程式伺服器實例隱藏訊息。標示訊息的時段由佇列管理程式屬性 **MARKINT** 判斷，依預設設定為 5000 毫秒 (5 秒)。這表示，在啟動規格或接聽器埠瀏覽訊息之後，佇列管理程式會等待 5 秒鐘以破壞性地取得訊息，然後允許其他應用程式伺服器實例查看該訊息並進行處理。

下列狀況可能會發生：

- 在應用程式伺服器 1 上執行的啟動規格會瀏覽佇列中的訊息 A。
- 啟動規格會啟動新執行緒來處理訊息 A。
- 應用程式伺服器 1 中發生事件，表示訊息 A 在 5 秒後仍然在佇列中。
- 在應用程式伺服器 2 上執行的啟動規格現在會瀏覽訊息 A 並啟動新的執行緒以處理訊息 A。
- 在應用程式伺服器 2 上執行的新執行緒會破壞性地取得訊息 A，並將它傳遞至訊息驅動 Bean 實例。
- 在應用程式伺服器 1 上執行的執行緒會嘗試取得訊息 A，僅發現訊息 A 不再在佇列上。
- 此時，應用程式伺服器 1 報告 CWSJY0003W 訊息。

解析問題

有兩種您可以解決此問題的方式：

- 將佇列管理程式屬性 **MARKINT** 值增加到較高的值。**MARKINT** 的預設值是 5000 毫秒 (5 秒)。增加此值可為應用程式伺服器提供更多時間，以在偵測到訊息之後破壞性地取得訊息。變更 **MARKINT** 值會影響連接至佇列管理程式的所有應用程式，並在應用程式破壞性地取得訊息之前瀏覽訊息。
- 將 WebSphere Application Server 中 **com.ibm.msg.client.wmq.suppressBrowseMarkMessageWarning** 內容的值變更為 *true*，以抑制 CWSJY0003W 警告訊息。如果要在 WebSphere Application Server 中設定變數，請開啟管理主控台，並導覽至 伺服器-> 應用程式伺服器-> Java 和程序管理-> 程序定義-> Java 虛擬機器-> 自訂內容-> 新建

```
Name = com.ibm.msg.client.wmq.suppressBrowseMarkMessageWarning
Value = true
```


註: 如果啟動規格或接聽器埠使用 IBM WebSphere MQ 傳訊提供者移轉模式連接至 IBM WebSphere MQ, 您可以忽略這些訊息。此作業模式的設計意味著在正常作業期間可能會出現此訊息。

相關資訊

[避免重複遞送瀏覽的訊息](#)

[ALTER QMGR](#)

[啟動規格](#)

[在應用程式伺服器機能 \(ASF\) 模式中執行的接聽器埠](#)

[在非應用程式伺服器機能 \(非 ASF\) 模式中執行的接聽器埠](#)

包含方法 'xa_end' 失敗, 並顯示 errorCode '100' 錯誤的 J2CA0027E 訊息

J2CA0027E 訊息出現在 WebSphere Application Server SystemOut.log 內, 其中包含方法 'xa_end' 失敗, 並顯示 errorCode '100' 錯誤。

簡介

當使用 WebSphere Application Server IBM WebSphere MQ 傳訊提供者的應用程式嘗試確定交易時, WebSphere Application Server SystemOut.log 檔中會出現下列錯誤:

```
J2CA0027E: 在交易 ID Transaction_Identifier 內, 從  
DataSource JMS_Connection_Factory 對 XA 資源配接器呼叫結束時發生異常狀況:  
javax.transaction.xa.XAException: 方法 'xa_end' 失敗, 並顯示 errorCode '100'。
```

```
J2CA0027E: 在交易 ID Transaction_Identifier 內, 從  
DataSource JMS_Connection_Factory 對 XA 資源配接器呼叫回復時發生異常狀況:  
javax.transaction.xa.XAException: 方法 'xa_rollback' 失敗, 並顯示 errorCode '-7'。
```

原因

這些錯誤的原因可能是 WebSphere Application Server 關閉 IBM WebSphere MQ 傳訊提供者 JMS 連線的結果, 因為連線的過時逾時值已過期。

透過 JMS Connection Factory 建立的 JMS 連線。存在與每個 Connection Factory 相關聯的連線儲存區, 分為兩部分 - 作用中儲存區及可用儲存區。

應用程式中斷已在使用中的 JMS 連線時, 該連線會移至 Connection Factory 的連線儲存區的可用儲存區中 (除非該連線的存在逾時值已過期, 在此情況下連線會損毀)。如果 JMS 連線在已損毀後仍然包含在作用中交易中, 應用程式伺服器會將 xa_end() 傳送至 IBM WebSphere MQ, 指出已完成連線的所有交易式工作。

如果已在使用啟動規格或接聽器埠來監視 IBM WebSphere MQ 佇列管理程式上 JMS 目的地的交易式訊息驅動 Bean 內建立 JMS 連線, 則會導致問題。

在此狀況下, 存在使用 2 條 IBM WebSphere MQ 連線的單一交易:

- 用於從 IBM WebSphere MQ 取得訊息, 並將它遞送給訊息驅動 Bean 實例以進行處理的連線。
- 在訊息驅動 Bean 的 onMessage() 方法內建立的連線。

如果訊息驅動 Bean 關閉第二條連線, 然後因存在逾時值到期而毀損, 則 xa_end() 會傳送至 IBM WebSphere MQ, 指出已完成所有交易式工作。

訊息驅動 Bean 應用程式完成處理提供的訊息時, 應用程式伺服器需要完成交易。它透過將 xa_end() 傳送至交易中涉及的所有資源 (包括 IBM WebSphere MQ), 來執行此作業。

但是, IBM WebSphere MQ 已收到用於此特定交易的 xa_end(), 並因此將 XA_RBROLLBACK (100) 錯誤傳回至 WebSphere Application Server, 指出交易已結束, 且 IBM WebSphere MQ 已回復所有工作。這會導致應用程式伺服器報告下列錯誤:

```
J2CA0027E: 在交易 ID Transaction_Identifier 內, 從  
DataSource JMS_Connection_Factory 對 XA 資源配接器呼叫結束時發生異常狀況:  
javax.transaction.xa.XAException: 方法 'xa_end' 失敗, 並顯示 errorCode '100'。
```

然後透過將 xa_rollback() 傳送至交易中列入的所有資源, 來回復整個交易。當應用程式伺服器將 xa_rollback() 傳送至 IBM WebSphere MQ 時, 會發生下列錯誤:

J2CA0027E: 在交易 ID *Transaction_Identifier* 內, 從 *DataSource JMS_Connection_Factory* 對 XA 資源配接器呼叫回復時發生異常狀況:
javax.transaction.xa.XAException: 方法 'xa_rollback' 失敗, 並顯示 errorCode '-7'。

環境

這個問題可能會影響訊息驅動 Bean 應用程式, 這些應用程式使用啟動規格或接聽器埠來監視 IBM WebSphere MQ 佇列管理程式上所管理的 JMS Destination, 然後從其 *onMessage()* 方法內使用 JMS Connection Factory 來建立與 IBM WebSphere MQ 的新連線。

解析問題

若要解決此問題, 請確保應用程式使用的 JMS Connection Factory 已將連線儲存區內容「存在逾時值」設定為零。這會防止 JMS 連線在回到可用儲存區時關閉, 從而確保可以完成任何未執行的交易式工作。

2035 從 WebSphere Application Server 連接至 IBM MQ 時 MQRN_NOT_AUTHORIZED

當應用程式從 WebSphere Application Server 連接至 IBM WebSphere MQ 時, 可能會發生 2035 MQRN_NOT_AUTHORIZED 錯誤。

本主題涵蓋在 WebSphere Application Server 中執行的應用程式在連接至 IBM MQ 時收到 2035 MQRN_NOT_AUTHORIZED 錯誤的最常見原因。解決問題小節中提供了瞭解決開發期間發生的 2035 MQRN_NOT_AUTHORIZED 錯誤的快速步驟, 亦提供了在正式作業環境中實作安全性的考量。還為具有儲存器管理和元件管理安全的出埠實務練習行為, 及接聽器埠和啟動規格的入埠行為提供了摘要。

問題原因

下列清單說明 IBM MQ 拒絕連線的最常見原因:

- 在從應用程式伺服器至 IBM MQ 的用戶端連線間傳遞的使用者 ID: 對 IBM MQ 佇列管理程式執行所在的伺服器來說是未知的, 未授權連接至 IBM MQ, 或超過 12 個字元而被截斷。存在有關在診斷問題中如何取得及傳遞此使用者 ID 的相關資訊。
 - **Windows** 對於在 Windows 上執行的佇列管理程式, 在此實務範例的 IBM MQ 錯誤日誌中可能會看到下列錯誤: AMQ8075: Authorization failed because the SID for entity 'wasuser' cannot be obtained.
 - **UNIX** 對於 UNIX, 在 IBM MQ 錯誤日誌中看不到任何項目:
- 透過用戶端連線從應用程式伺服器傳遞至 IBM MQ 的使用者 ID, 是管理 IBM MQ 佇列管理程式之伺服器上的 *mqm* 群組成員, 且存在通道鑑別記錄 (CHLAUTH) 可封鎖對佇列管理程式的管理存取權。依預設, IBM MQ 會在 IBM WebSphere MQ 7.1 及更新版本中配置 CHLAUTH 記錄, 以封鎖所有 IBM MQ 管理者以用戶端身分連接至佇列管理程式。針對此實務範例, 會在 IBM MQ 錯誤日誌中看到下列錯誤: AMQ9777: Channel was blocked.
- 存在 Advanced Message Security 安全原則。

如需 IBM MQ 錯誤日誌的位置, 請參閱[錯誤日誌目錄](#)。

診斷問題

若要瞭解 2035 MQRN_NOT_AUTHORIZED 原因碼的原因, 您必須瞭解 IBM MQ 使用哪個使用者名稱及密碼授權應用程式伺服器。

註: 本主題中提供的知識對於開發環境非常有用, 解決正式作業環境的安全需求通常需要下列其中一種方法:

- 相互 SSL/TLS 鑑別
IBM MQ 提供了使用為 SSL/TLS 連線提供的數位憑證鑑別遠端連接的用戶端的特性。
- 自訂或第三方提供的 IBM MQ 安全結束程式

可以為 IBM MQ 撰寫安全結束程式，以針對儲存庫 (例如本端作業系統、IBM MQ 伺服器或 LDAP 儲存庫) 執行使用者名稱及密碼鑑別。當您使用安全結束程式進行鑑別時，仍然配置 SSL/TLS 傳輸安全非常重要，以確保不會以純文字傳送密碼。

在伺服器連線通道上配置 MCA 使用者 ID

如果在應用程式伺服器用於連接的伺服器連線通道上已配置 MCA 使用者 ID，且未安裝安全結束程式或對映通道鑑別記錄，則 MCA 使用者 ID 會置換應用程式伺服器提供的使用者名稱。對許多客戶而言，通常是在每個伺服器連線通道上設定 MCA 使用者 ID，及獨佔地使用相互 SSL/TLS 鑑別進行鑑別。

應用程式伺服器未提供認證時的預設行為

如果在 **createConnection** 呼叫時應用程式未提供認證，且未配置元件管理或儲存器管理安全系統，則 WebSphere Application Server 會為 IBM MQ 提供空白的使用者名稱。這會導致 IBM MQ 根據 IBM MQ 接聽器執行所依據的使用者 ID 授權用戶端。在大部分情況下，使用者 ID 在 UNIX 或 Linux 系統上為 *mqm*，在 Windows 上為 *MUSR_MQADMIN*。當這些使用者為管理 IBM MQ 使用者時，在 IBM WebSphere MQ 7.1 及更新版本中，依預設這些使用者會被封鎖，佇列管理程式的錯誤日誌中會記載 *AMQ9777* 錯誤。

出埠連線的儲存器管理安全

配置由出埠連線的應用程式伺服器傳遞至 IBM MQ 的使用者名稱及密碼的建議方法，是使用儲存器管理安全。出埠連線是使用 Connection Factory 建立的連線，而非接聽器埠或啟動規格。

應用程式伺服器會將 12 個字元或更少的使用者名稱傳遞至 v。在授權期間 (在 UNIX 上) 或在所傳送訊息的 *MQMD* 中，長度超過 12 個字元的使用者名稱會被截斷。儲存器管理安全表示應用程式的部署描述子或 EJB 3.0 註釋宣告鑑別類型設定為「儲存器」的資源參照。然後，當應用程式在 JNDI 中查閱 Connection Factory 時，它會透過資源參照間接地執行此作業。例如，EJB 2.1 應用程式將如下所示執行 JNDI 查閱，其中 *jms/MyResourceRef* 宣告為部署描述子中的資源參照。

```
ConnectionFactory myCF = (ConnectionFactory)ctx.lookup("java:comp/env/jms/MyResourceRef")
```

EJB 3.0 應用程式可能會如下所示宣告 Bean 上標註的物件內容：

```
@Resource(name = "jms/MyResourceRef"  
    authenticationType = AuthenticationType.CONTAINER)  
private javax.jms.ConnectionFactory myCF
```

管理者部署應用程式時，他們會將此鑑別別名連結至已在 JNDI 中建立的實際 Connection Factory，並在部署時為它指派 J2C 鑑別別名。它是包含在此鑑別別名中的使用者名稱及密碼，然後在應用程式連接時由應用程式伺服器傳遞至 IBM MQ 或 JMS。此方法會讓管理者控制每個應用程式使用的使用者名稱及密碼，並阻止不同的應用程式在 JNDI 中直接查閱 Connection Factory，以使用相同的名稱及密碼進行連線。在 IBM MQ Connection Factory 管理主控台的配置畫面上，您可以提供預設儲存器管理鑑別別名。僅在以下情況下使用此預設值：應用程式使用為儲存器管理安全配置的資源參照，但在部署期間管理者尚未將它連結至鑑別別名。

出埠連線的預設元件管理的鑑別別名

對於將應用程式變更為使用儲存器管理安全或將它變更為在 **createConnection** 呼叫時直接提供使用者名稱及密碼的不切實際的情況，可以提供預設值。此預設值稱為元件管理的鑑別別名，無法在管理主控台中配置 (自 WebSphere Application Server 7.0 起，從 IBM MQ Connection Factory 的畫面中移除它時)。下列 Scripting 範例顯示如何使用 *wsadmin* 配置它：

- JACL

```
wsadmin>set cell [ $AdminConfig getid "/Cell:mycell" ]  
mycell(cells/mycell|cell.xml#Cell_1)  
wsadmin>$AdminTask listWMQConnectionFactories $cell  
MyCF(cells/mycell|resources.xml#MQConnectionFactory_1247500675104)  
wsadmin>$AdminTask modifyWMQConnectionFactory MyCF(cells/mycell|  
resources.xml#MQConnectionFactory_1247500675104) { -componentAuthAlias myalias }  
MyCF(cells/mycell|resources.xml#MQConnectionFactory_1247500675104)
```

- Jython

```
wsadmin>cell = AdminConfig.getid("/Cell:mycell")  
wsadmin>AdminTask.listWMQConnectionFactories(cell)  
'MyCF(cells/mycell|resources.xml#MQConnectionFactory_1247500675104)'  
wsadmin>AdminTask.modifyWMQConnectionFactory('MyCF(cells/mycell|resos
```

```
ources.xml#MQConnectionFactory_1247500675104)', "-componentAuthAlias myalias")  
'MyCF(cells/mycell|resources.xml#MQConnectionFactory_1247500675104)'
```

使用啟動規格的入埠 MDB 連線的鑑別別名

對於使用啟動規格的入埠連線，在部署應用程式時管理者可以指定鑑別別名，或在管理主控台的啟動規格上可以指定預設鑑別別名。

使用接聽器埠的入埠 MDB 連線的鑑別別名

對於使用接聽器埠的入埠連線，則會使用 Connection Factory 的儲存器管理鑑別別名設定中指定的值。

z/OS 在 z/OS 上，先檢查並使用儲存器管理的鑑別別名（如果已設定），然後檢查並使用元件管理的鑑別別名（如果已設定）。

解析問題

解決開發環境中 2035 MQRC_NOT_AUTHORIZED 錯誤的最簡單步驟（不需要完全傳輸安全）如下所示：

- 選擇您希望鑑別 WebSphere Application Server 的使用者身分。通常，所選使用者應具有與在 WebSphere Application Server 中執行的應用程式所需作業環境定義相關的權限即可。例如，mqm 或其他超級使用者是不適當的。
- 如果此使用者是 IBM MQ 管理使用者，則會在 IBM WebSphere MQ 7.1 或更新版本中放寬通道鑑別記錄 (CHLAUTH) 安全，以便在您要使用的伺服器連線通道上不會封鎖管理連線。伺服器連線通道 WAS.CLIENTS 是 SET CHLAUTH('WAS.CLIENTS') TYPE(BLOCKUSER) USERLIST(ALLOWANY)。
- 配置伺服器連線通道以將 MCA 使用者 ID (MCAUSER) 設定為您要使用的使用者。將伺服器連線通道配置成使用 myuser 作為 MCA 使用者 ID 的範例 MQSC 指令為 ALTER CHL('WAS.CLIENTS') CHLTYPE(SVRCONN) MCAUSER('myuser')。

正式作業環境的重要額外考量

對於需要傳輸安全的所有正式作業環境，必須在應用程式伺服器與 IBM MQ 之間配置 SSL/TLS 安全。

若要配置 SSL/TLS 傳輸安全，您必須在 IBM MQ 佇列管理程式與 WebSphere Application Server 之間建立適當的信任。應用程式伺服器會起始 SSL/TLS 信號交換，且必須配置為信任 IBM MQ 佇列管理程式提供的憑證。如果應用程式伺服器配置為將憑證傳送至 IBM MQ 佇列管理程式，則佇列管理程式還必須配置為信任它。如果在兩端未正確配置信任，在連線上啟用 SSL/TLS 之後，將會發現 2393 MQRC_SSL_INITIALIZATION_ERROR 原因碼。

如果您沒有用於執行使用者名稱及密碼鑑別的安全結束程式，則應在伺服器連線通道上配置相互 SSL/TLS 鑑別，以使佇列管理程式要求應用程式伺服器提供授信憑證。若要執行此作業，請將 SSL 鑑別設定為必要（在「IBM MQ Explorer」中）或設定 SSLCAUTH(REQUIRED)（在 MQSC 中）。

如果您沒有用於執行使用者名稱及密碼鑑別的、在 IBM MQ 伺服器中安裝的安全結束程式，則將應用程式配置為提供使用者名稱及密碼由該安全結束程式進行驗證。如需如何配置應用程式伺服器傳遞至 IBM MQ 之使用者名稱及密碼的詳細資料，請參閱 診斷問題 一節。

沒有 SSL/TLS 安全的所有伺服器連線通道都應停用。停用 SYSTEM.DEF.SVRCONN 通道提供如下（假設使用者不存在於稱為 ('NOAUTH')、ALTER CHL(SYSTEM.DEF.SVRCONN) CHLTYPE(SVRCONN) MCAUSER('NOAUTH') STOP CHL(SYSTEM.DEF.SVRCONN) 的 IBM MQ 伺服器上）。

如需配置 IBM MQ 佇列管理程式的專用憑證及信任，以及在伺服器連線通道上啟用 SSL 安全的指示，請參閱在佇列管理程式上配置 SSL 及配置 SSL 通道。

如需在 WebSphere Application Server 中使用 SSL/TLS 及應用程式伺服器是否將憑證傳送至 IBM MQ 進行鑑別的相關資訊，請參閱下列資訊：

- 若要建立或修改 SSL 配置，以包含 IBM MQ 連線的適當 SSL/TLS 配置，請參閱 WebSphere Application Server 產品說明文件中的 SSL 配置。
- IBM MQ 要求您必須在連線的兩端指定相符的 CipherSpec。如需可以與 IBM MQ 搭配使用的 CipherSpec 及 CipherSuite 的相關資訊，請參閱 WebSphere® MQ 佇列管理程式連線的 CipherSuite 及 CipherSpec 名稱對映。

- 如需在用戶端連接上啟用 SSL/TLS 以及選擇要使用的 SSL 配置的相關資訊，請參閱 [WebSphere Application Server 產品說明文件中的 WebSphere MQ 傳訊提供者 Connection Factory 設定](#) 和 [WebSphere MQ 傳訊提供者啟動規格設定](#)。

相關參考

第 174 頁的『[回覆碼 = 2035 MQRC_NOT_AUTHORIZED](#)』

由於各種原因會顯示 RC2035 原因碼，包括開啟佇列或通道時發生錯誤、嘗試使用具有管理者權限的使用者 ID 時收到錯誤、使用 IBM MQ JMS 應用程式時發生錯誤，以及在叢集上開啟佇列。

MQS_REPORT_NOAUTH 和 MQSAUTHERRORS 可用來進一步診斷 RC2035。

相關資訊

[MQRC_NOT_AUTHORIZED](#)

IBM MQ 資源配接器的問題判斷

當使用 IBM MQ 資源配接器時，大部分錯誤會導致擲出異常狀況，這些異常狀況會以相依於應用程式伺服器的方式向使用者報告。資源配接器廣泛使用鏈結的異常狀況來報告問題。一般而言，鏈結中的第一個異常狀況是錯誤的高階說明，而鏈結中的後續異常狀況會提供診斷問題所需的更詳細資訊。

例如，如果 IVT 程式無法取得與 IBM MQ 佇列管理程式的連線，則可能會擲出下列異常狀況：

```
javax.jms.JMSEException: MQJCA0001: JMS 層中發生異常狀況。  
請參閱鏈結的異常狀況，以取得詳細資料。
```

鏈結至此異常狀況是第二個異常狀況：

```
javax.jms.JMSEException: MQJMS2005: 無法為建立 MQQueueManager  
'localhost:ExampleQM'
```

此異常狀況由 IBM MQ classes for JMS 擲出，且具有進一步鏈結的異常狀況：

```
com.ibm.mq.MQException: MQJE001: 發生 MQException: 完成碼 2 ,  
原因 2059
```

這個最終異常狀況指出問題的來源。原因碼 2059 是 MQRC_Q_MGR_NOT_AVAILABLE，指出 ConnectionFactory 物件定義中指定的佇列管理程式可能尚未啟動。

如果異常狀況所提供的資訊不足以診斷問題，您可能需要要求診斷追蹤。如需如何啟用診斷追蹤的相關資訊，請參閱 [IBM MQ 資源配接器的配置](#)。

配置問題通常發生在下列區域：

- 部署資源配接器
- 部署 MDB
- 建立出埠通訊的連線

相關資訊

[使用 IBM MQ 資源配接器](#)

部署資源配接器時發生問題

如果資源配接器無法部署，請檢查是否已正確配置 Java EE Connector Architecture (JCA) 資源。如果已安裝 IBM MQ，請檢查 JCA 和 IBM MQ classes for JMS 的正確版本是否在類別路徑中。

部署資源配接器失敗的原因通常是未正確配置 JCA 資源。例如，ResourceAdapter 物件的內容可能指定不正確，或應用程式伺服器所需的部署計劃可能撰寫不正確。當應用程式伺服器嘗試從 JCA 資源的定義建立物件，並將物件連結至 Java Naming Directory Interface (JNDI) 名稱空間，但未正確指定某些內容或資源定義的格式不正確時，也可能會發生失敗。

資源配接器也可能無法部署，因為它從類別路徑中的 JAR 檔載入不正確版本的 JCA 或 IBM MQ classes for JMS 類別。這種類型的故障通常發生在已安裝 IBM MQ 的系統上。在這類系統上，應用程式伺服器可能會找到 IBM MQ classes for JMS JAR 檔的現有副本，並優先從它們載入類別，以取代 IBM MQ 資源配接器 RAR 檔中所提供的類別。

相關資訊

[針對 IBM MQ for JMS 類別安裝的內容](#)

部署 MDB 時發生問題

當應用程式伺服器嘗試開始將訊息遞送至 MDB 時失敗，可能是相關聯 ActivationSpec 物件定義中的錯誤或遺漏資源所造成。

當應用程式伺服器嘗試開始將訊息遞送至 MDB 時，可能會發生失敗。這種失敗類型通常是因為相關聯 ActivationSpec 物件的定義中發生錯誤，或定義中所參照的資源無法使用。例如，佇列管理程式可能不在執行中，或指定的佇列可能不存在。

當部署 MDB 時，ActivationSpec 物件會嘗試驗證其內容。如果 ActivationSpec 物件具有任何互斥內容或沒有所有必要內容，則部署會失敗。不過，此時無法偵測到與 ActivationSpec 物件內容相關聯的所有問題。

啟動訊息遞送失敗會以取決於應用程式伺服器的方式向使用者報告。通常會在應用程式伺服器的日誌和診斷追蹤中報告這些失敗。如果啟用，IBM MQ 資源配接器的診斷追蹤也會記錄這些失敗。

建立出埠通訊的連線時發生問題

如果找不到 ConnectionFactory 物件，或找到 ConnectionFactory 物件但無法建立連線，則出埠通訊可能會失敗。這些問題有各種原因。

當應用程式嘗試在 JNDI 名稱空間中查閱並使用 ConnectionFactory 物件時，通常會發生出埠通訊失敗。如果在名稱空間中找不到 ConnectionFactory 物件，則會擲出 JNDI 異常狀況。由於下列原因，可能找不到 ConnectionFactory 物件：

- 應用程式指定了不正確的 ConnectionFactory 物件名稱。
- 應用程式伺服器無法建立 ConnectionFactory 物件並將它連結至名稱空間。在此情況下，應用程式伺服器的啟動日誌通常包含失敗的相關資訊。

如果應用程式順利從 JNDI 名稱空間擷取 ConnectionFactory 物件，當應用程式呼叫 ConnectionFactory.createConnection() 方法時，仍可能擲出異常狀況。此環境定義中的異常狀況指出無法建立與 IBM MQ 佇列管理程式的連線。以下是可能擲出異常狀況的一些常見原因：

- 無法使用或使用 ConnectionFactory 物件的內容找不到佇列管理程式。例如，佇列管理程式不在執行中，或佇列管理程式的指定主機名稱、IP 位址或埠號不正確。
- 使用者未獲授權來連接佇列管理程式。對於用戶端連線，如果 createConnection() 呼叫未指定使用者名稱，且應用程式伺服器未提供使用者身分資訊，則會將 JVM 程序 ID 作為使用者名稱傳遞給佇列管理程式。若要連線成功，在執行佇列管理程式的系統中，此程序 ID 必須是有效的使用者名稱。
- ConnectionFactory 物件具有稱為 ccdtURL 的內容及稱為通道的內容。這些內容互斥。
- 在 TLS 連線，未正確指定伺服器連線通道定義中的 TLS 相關內容或 TLS 相關屬性。
- sslFips 「必要」內容針對不同的 JCA 資源具有不同的值。如需此限制的相關資訊，請參閱 [IBM MQ 資源配接器的限制](#)。

相關資訊

指定在執行時期於 MQI 用戶端上僅使用 FIPS 認證的 CipherSpecs
[UNIX, Linux, and Windows 的聯邦資訊存取安全標準 \(FIPS\)](#)

使用 IBM MQ 連線內容置換

連線內容置換可讓您變更用戶端應用程式用來連接至佇列管理程式的詳細資料，而不需修改原始碼。

關於這項作業

有時，無法修改應用程式的原始碼，例如，如果應用程式是舊式應用程式，且原始碼不再可用。

在此情況下，如果應用程式在連接至佇列管理程式時需要指定不同的內容，或連接至不同的佇列管理程式時需要指定不同的內容，則您可以使用連線置換功能來指定新的連線詳細資料或佇列管理程式名稱。

兩個用戶端支援連線內容置換：

- [IBM MQ classes for JMS](#)

- [IBM MQ classes for Java](#)

您可以置換您要變更的內容，方法是在配置檔中定義它們，然後在啟動時由 IBM MQ classes for JMS 或 IBM MQ classes for Java 讀取。

當使用連線置換功能時，在相同 Java runtime environment 內執行的所有應用程式都會挑選並使用新的內容值。如果多個使用 IBM MQ classes for JMS 或 IBM MQ classes for Java 的應用程式在相同 Java runtime environment 內執行，則無法僅置換個別應用程式的內容。

重要: 只有在無法修改應用程式的原始碼時，才支援此功能。它不得用於原始碼可用且可更新的應用程式。

相關概念

第 78 頁的『[追蹤 IBM MQ classes for JMS 應用程式](#)』

提供 IBM MQ classes for JMS 中的追蹤機能，以協助「IBM 支援中心」診斷客戶問題。各種內容會控制此機能的行為。

相關工作

第 82 頁的『[追蹤 IBM MQ classes for Java 應用程式](#)』

提供 IBM MQ classes for Java 中的追蹤機能，以協助「IBM 支援中心」診斷客戶問題。各種內容會控制此機能的行為。

相關資訊

[使用 IBM MQ classes for JMS](#)

[使用 IBM MQ classes for Java](#)

在 IBM MQ classes for JMS 中使用連線內容置換

如果以程式化方式建立 Connection Factory，且無法修改建立它之應用程式的原始碼，則可以使用連線置換功能來變更 Connection Factory 在建立連線時所使用的內容。不過，不支援將連線置換功能與 JNDI 中定義的 Connection Factory 搭配使用。

關於這項作業

在 IBM MQ classes for JMS 中，如何連接至佇列管理程式的詳細資料儲存在 Connection Factory 中。Connection Factory 可以透過管理方式定義並儲存在 JNDI 儲存庫中，或由應用程式使用 Java API 呼叫以程式化方式建立。

如果應用程式以程式化方式建立 Connection Factory，且無法修改該應用程式的原始碼，連線置換功能可讓您在短期內置換 Connection Factory 內容。不過，長期而言，您必須將計劃放在適當的位置，以容許在不使用連線置換功能的情況下修改應用程式所使用的 Connection Factory。

如果應用程式以程式化方式建立的 Connection Factory 定義為使用「用戶端通道定義表 (CCDT)」，則會優先使用 CCDT 中的資訊，而不使用被置換的內容。如果需要變更應用程式使用的連線詳細資料，則必須建立新的 CCDT 版本，並使其可供應用程式使用。

不支援將連線置換功能與 JNDI 中定義的 Connection Factory 搭配使用。如果應用程式使用 JNDI 中所定義的 Connection Factory，且需要變更該 Connection Factory 的內容，則必須在 JNDI 中更新 Connection Factory 的定義。雖然會將連線置換功能套用至這些 Connection Factory (且被置換的內容優先於在 JNDI 中查閱的 Connection Factory 定義中的內容)，但不支援這種使用連線置換功能。

重要: 連線置換功能會影響在 Java runtime environment 內執行的所有應用程式，並套用至那些應用程式所使用的所有 Connection Factory。不能只置換個別 Connection Factory 或應用程式的內容。

當應用程式使用 Connection Factory 來建立與佇列管理程式的連線時，IBM MQ classes for JMS 會查看已置換的內容，並在建立連線時使用這些內容值，而不是 Connection Factory 中相同內容的值。

例如，假設已定義 Connection Factory，且 PORT 內容設為 1414。如果已使用連線置換功能將 PORT 內容設為 1420，則當使用 Connection Factory 來建立連線時，IBM MQ classes for JMS 會對 PORT 內容使用值 1420，而非 1414。

如果要修改從 Connection Factory 建立 JMS 連線時所使用的任何連線內容，必須執行下列步驟：

1. [將要置換的內容新增至 IBM MQ 類別。](#)
2. [啟用連線置換功能。](#)

3. 指定配置檔來啟動應用程式。

程序

1. 將要置換的內容新增至 IBM MQ classes for JMS 配置檔。

- a) 建立檔案，其中包含需要以標準 Java 內容格式置換的內容和值。

如需如何建立內容檔的詳細資料，請參閱 [IBM MQ classes for JMS 配置檔](#)。

- b) 若要置換內容，請將項目新增至內容檔。

任何 IBM MQ classes for JMS Connection Factory 內容都可以置換。以下列格式新增每一個必要項目：

```
jmscf.property name=value
```

其中 *property name* 是需要置換之內容的 JMS 管理內容名稱或 XMSC 常數。如需 Connection Factory 內容的清單，請參閱 [IBM MQ classes for JMS 物件的內容](#)。

例如，若要設定應用程式應該用來連接至佇列管理程式的通道名稱，您可以將下列項目新增至內容檔：

```
jmscf.channel=MY.NEW.SVRCONN
```

2. 啟用連線置換功能。

若要啟用連線置換，請將 **com.ibm.msg.client.jms.overrideConnectionFactory** 內容設為 true，以便使用內容檔中指定的內容來置換應用程式中指定的值。您可以將額外內容設定為配置檔本身中的另一個內容，或使用下列指令將內容傳遞為 Java 系統內容：

```
-Dcom.ibm.msg.client.jms.overrideConnectionFactory=true
```

3. 啟動應用程式，並指定配置檔。

透過設定 Java 系統內容，將您在執行時期建立的內容檔傳遞至應用程式：

```
-Dcom.ibm.msg.client.config.location
```

請注意，配置檔的位置必須指定為 URI，例如：

```
-Dcom.ibm.msg.client.config.location=file:///jms/jms.config
```

結果

當啟用連線置換功能時，每當建立連線時，IBM MQ classes for JMS 都會將項目寫入 jms 日誌。日誌中的資訊會顯示建立連線時所置換的 Connection Factory 內容，如下列範例項目所示：

```
Overriding ConnectionFactory properties:  
Overriding property channel:  
Original value = MY.OLD.SVRCONN  
New value      = MY.NEW.SVRCONN
```

相關工作

第 162 頁的『[在 IBM MQ classes for Java 中使用連線內容置換](#)』

在 IBM MQ classes for Java 中，使用不同值的組合將連線詳細資料設為內容。連線置換功能可用來置換應用程式使用的連線詳細資料 (如果無法修改應用程式的原始碼)。

第 163 頁的『[以 IBM MQ classes for JMS 置換連線內容: 範例](#)』

此範例顯示如何在使用 IBM MQ classes for JMS 時置換內容。

相關資訊

[在適用於 JMS 應用程式的 IBM MQ 類別中建立及配置 Connection Factory 和目的地](#)

在 IBM MQ classes for Java 中使用連線內容置換

在 IBM MQ classes for Java 中，使用不同值的組合將連線詳細資料設為內容。連線置換功能可用來置換應用程式使用的連線詳細資料 (如果無法修改應用程式的原始碼)。

關於這項作業

用來設定連線內容的不同值是下列項目的組合：

- 將值指派給 **MQEnvironment** 類別上的靜態欄位。
- 在 **MQEnvironment** 類別的 Hashtable 內容中設定內容值。
- 在傳遞至 **MQQueueManager** 建構子的 Hashtable 中設定內容值。

然後，當應用程式建構代表佇列管理程式連線的 **MQQueueManager** 物件時，即會使用這些內容。

如果無法修改使用 IBM MQ classes for Java 的應用程式的原始碼，以指定在建立與佇列管理程式的連線時必須使用的不同內容，則連線置換功能可讓您在短期內置換連線詳細資料。不過，長期而言，您必須將計劃放在適當的位置，以容許在不使用連線置換功能的情況下修改應用程式所使用的連線詳細資料。

當應用程式建立 **MQQueueManager** 時，IBM MQ classes for Java 會查看已置換的內容，並在建立與佇列管理程式的連線時使用這些內容值，而不是下列任何位置中的值：

- **MQEnvironment** 類別上的靜態欄位
- 儲存在 **MQEnvironment** 類別中的內容 Hashtable
- 傳遞至 **MQQueueManager** 建構子的內容 Hashtable

例如，假設應用程式建立 **MQQueueManager**，並傳入 CHANNEL 內容設為 MY.OLD.CHANNEL。如果已使用連線置換功能將 CHANNEL 內容設為 MY.NEW.CHANNEL，然後當建構 **MQQueueManager** 時，IBM MQ classes for Java 會嘗試使用通道 MY.NEW.CHANNEL 而非 MY.OLD.CHANNEL。

註：如果 **MQQueueManager** 配置為使用「用戶端通道定義表 (CCDT)」，則會優先使用 CCDT 中的資訊，而不使用被置換的內容。如果需要變更應用程式建立 **MQQueueManager** 所使用的連線詳細資料，則必須建立新的 CCDT 版本，並使其可供應用程式使用。

若要修改建立 **MQQueueManager** 時所使用的任何連線內容，必須執行下列步驟：

1. 建立稱為 `mqclassesforjava.config` 的內容檔。
2. 將 **OverrideConnectionDetails** 內容設為 `true`，以啟用連線內容置換功能。
3. 指定配置檔作為 Java 呼叫的一部分，以啟動應用程式。

程序

1. 建立稱為 `mqclassesforjava.config` 的內容檔，其中包含需要置換的內容和值。

作為 **MQQueueManager** 建構子的一部分，在連接至佇列管理程式時，可以置換 IBM MQ classes for Java 所使用的 13 個內容。下表顯示這些內容的名稱，以及在置換它們時必須指定的索引鍵：

表 19: 可置換的內容	
內容	內容索引鍵
CCSID	\$CCSID_PROPERTY
通道	\$CHANNEL_PROPERTY
連線選項	\$CONNECT_OPTIONS_PROPERTY
主機名稱	\$HOST_NAME_PROPERTY
SSL 金鑰重設	\$SSL_RESET_COUNT_PROPERTY
本端位址	\$LOCAL_ADDRESS_PROPERTY

表 19: 可置換的內容 (繼續)	
內容	內容索引鍵
佇列管理程式名稱	qmgr
密碼	\$Password_property
埠	\$PORT_PROPERTY
密碼組合	\$SSL_CIPHER_SUITE_PROPERTY
所需的 FIPS	\$SSL_FIPS_REQUIRED_PROPERTY
SSL 對等名稱	\$SSL_PEER_NAME_PROPERTY
使用者 ID	\$USER_ID_property

註: 除了佇列管理程式名稱之外, 所有內容索引鍵都以 \$ 字元開頭。這是因為佇列管理程式名稱會傳入 MQQueueManager 建構子作為引數, 而不是設為 MQEnvironment 類別上的靜態欄位, 或設為 Hashtable 中的內容, 因此在內部, 這個內容需要與其他內容稍微不同。

若要置換內容, 請將下列格式的項目新增至內容檔:

```
mqj.property key=value
```

例如, 若要設定要在建立 MQQueueManager 物件時使用的通道名稱, 您可以將下列項目新增至內容檔:

```
mqj.$CHANNEL_PROPERTY=MY.NEW.CHANNEL
```

若要變更 MQQueueManager 物件所連接的佇列管理程式名稱, 您可以將下列項目新增至內容檔:

```
mqj.qmgr=MY.OTHER.QMGR
```

- 將 **com.ibm.mq.overrideConnectionDetails** 內容設為 true, 以啟用連線置換功能。

將內容 **com.ibm.mq.overrideConnectionDetails** 設為 true 表示使用內容檔中指定的內容來置換應用程式中指定的值。您可以使用下列指令, 將額外內容設為配置檔本身中的另一個內容, 或將內容當作系統內容來傳遞:

```
-Dcom.ibm.mq.overrideConnectionDetails=true
```

- 請啟動應用程式。

透過設定 Java 系統內容, 將您在執行時期建立的內容檔傳遞至用戶端應用程式:

```
-Dcom.ibm.msg.client.config.location
```

請注意, 配置檔的位置必須指定為 URI, 例如:

```
-Dcom.ibm.msg.client.config.location=file:///classesforjava/mqclassesforjava.config
```

以 IBM MQ classes for JMS 置換連線內容: 範例

此範例顯示如何在使用 IBM MQ classes for JMS 時置換內容。

關於這項作業

下列程式碼範例顯示應用程式如何以程式化方式建立 ConnectionFactory:

```
JmsSampleApp.java
...
```

```
JmsFactoryFactory jmsff;
JmsConnectionFactory jmsConnFact;

jmsff = JmsFactoryFactory.getInstance(JmsConstants.WMQ_PROVIDER);
jmsConnFact = jmsff.createConnectionFactory();

jmsConnFact.setStringProperty(WMQConstants.WMQ_HOST_NAME, "127.0.0.1");
jmsConnFact.setIntProperty(WMQConstants.WMQ_PORT, 1414);
jmsConnFact.setStringProperty(WMQConstants.WMQ_QUEUE_MANAGER, "QM_V80");
jmsConnFact.setStringProperty(WMQConstants.WMQ_CHANNEL, "MY.CHANNEL");
jmsConnFact.setIntProperty(WMQConstants.WMQ_CONNECTION_MODE,
                           WMQConstants.WMQ_CM_CLIENT);
...

```

ConnectionFactory 配置為 QM_V80 使用 CLIENT 傳輸及通道 MY.CHANNEL 連接至佇列管理程式。

您可以使用內容檔來置換連線詳細資料，並使用下列程序來強制應用程式連接至不同的通道。

程序

1. 在 `/userHome` 目錄 (其中 `userHome` 是您的起始目錄) 中建立稱為 `jms.config` 的 IBM MQ classes for JMS 配置檔。

使用下列內容建立此檔案:

```
jmscf.CHANNEL=MY.TLS.CHANNEL
jmscf.SSLCIPHERSUITE=TLS_RSA_WITH_AES_128_CBC_SHA256

```

2. 執行應用程式，並將下列 Java 系統內容傳遞至應用程式執行所在的 Java runtime environment :

```
-Dcom.ibm.msg.client.config.location=file:///userHome/jms.config
-Dcom.ibm.msg.client.jms.overrideConnectionFactory=true

```

結果

執行此程序會置換應用程式以程式化方式建立的 ConnectionFactory，因此當應用程式建立連線時，它會嘗試使用通道 MY.TLS.CHANNEL 和密碼組合 TLS_RSA_WITH_AES_128_CBC_SHA256。

相關工作

第 159 頁的『[使用 IBM MQ 連線內容置換](#)』

連線內容置換可讓您變更用戶端應用程式用來連接至佇列管理程式的詳細資料，而不需修改原始碼。

第 160 頁的『[在 IBM MQ classes for JMS 中使用連線內容置換](#)』

如果以程式化方式建立 Connection Factory，且無法修改建立它之應用程式的原始碼，則可以使用連線置換功能來變更 Connection Factory 在建立連線時所使用的內容。不過，不支援將連線置換功能與 JNDI 中定義的 Connection Factory 搭配使用。

第 162 頁的『[在 IBM MQ classes for Java 中使用連線內容置換](#)』

在 IBM MQ classes for Java 中，使用不同值的組合將連線詳細資料設為內容。連線置換功能可用來置換應用程式使用的連線詳細資料 (如果無法修改應用程式的原始碼)。

疑難排解 IBM MQ.NET 問題

您可以使用 .NET 範例應用程式來協助疑難排解問題。

使用範例應用程式

如果程式未順利完成，請執行其中一個 .NET 範例應用程式，並遵循診斷訊息中提供的建議。這些範例應用程式在 .NET 的範例應用程式中說明。

如果問題繼續存在，且您需要聯絡 IBM 服務團隊，則可能會要求您開啟追蹤機能。如需使用追蹤機能的相關資訊，請參閱第 92 頁的『[追蹤 IBM MQ .NET 應用程式](#)』。

錯誤訊息

您可能會看到下列一般錯誤訊息：

無法處理的異常狀況，類型為 **System.IO.FileNotFoundException**

如果 amqmdnet.dll 或 amqmdxcn.dll 發生此錯誤，請確保兩者都已登錄在「廣域組件快取」中，或建立指向 amqmdnet.dll 及 amqmdxcn.dll 組件的配置檔。您可以使用 mscorcfg.msc(作為 .NET 架構的一部分提供) 來檢查及變更組件快取的內容。

如果在安裝 IBM MQ 時無法使用 .NET 架構，則可能不會在廣域組件快取中登錄類別。您可以使用指令手動重新執行登錄程序

```
amqidnet -c MQ_INSTALLATION_PATH\bin\amqidotn.txt -l logfile.txt
```

MQ_INSTALLATION_PATH 代表 IBM MQ 安裝所在的高階目錄。

此安裝的相關資訊會寫入指定的日誌檔 (此範例中的 logfile.txt)。

解決 IBM MQ MQI clients 的問題

此主題集合包含用於解決 IBM MQ MQI client 應用程式中問題之技術的相關資訊。

在 IBM MQ MQI client 環境中執行的應用程式會以與 IBM MQ 伺服器應用程式相同的方式接收 MQRC_* 原因碼。不過，與 IBM MQ MQI clients 相關聯的錯誤狀況還有其他原因碼。例如：

- 遠端機器未回應
- 通訊線路錯誤
- 機器位址無效

發生錯誤最常見的時間是應用程式發出 MQCONN 或 MQCONNX 並接收回應 MQRC_Q_MQR_NOT_AVAILABLE 時。請查看用戶端錯誤日誌，以取得說明失敗的訊息。視失敗的本質而定，可能也會在伺服器上記載錯誤。此外，請檢查 IBM MQ MQI client 上的應用程式是否與正確的程式庫檔案鏈結。

IBM MQ MQI client 無法建立連線

MQCONN 或 MQCONNX 可能會失敗，因為伺服器上或通訊協定檢查期間沒有接聽器程式在執行中。

當 IBM MQ MQI client 對伺服器發出 MQCONN 或 MQCONNX 呼叫時，會在 IBM MQ MQI client 與伺服器之間交換 Socket 及埠資訊。若要進行任何資訊交換，伺服器上必須有一個程式，其角色是在任何活動的通訊線路上「接聽」。如果沒有程式執行此動作，或有程式但未正確配置，則 MQCONN 或 MQCONNX 呼叫會失敗，且相關原因碼會傳回至 IBM MQ MQI client 應用程式。

如果連線成功，則會交換 IBM MQ 通訊協定訊息，並進行進一步檢查。在 IBM MQ 通訊協定檢查階段期間，會協議某些層面，而其他層面則會導致連線失敗。直到所有這些檢查都成功，MQCONN 或 MQCONNX 呼叫才會成功。

如需 MQRC_* 原因碼的相關資訊，請參閱 [API 完成碼和原因碼](#)。

正在停止 IBM MQ MQI clients

即使 IBM MQ MQI client 已停止，伺服器上的相關聯處理程序仍有可能保持開啟其佇列。在通訊層偵測到友機已消失之前，不會關閉佇列。

如果啟用共用交談，則伺服器通道一律處於正確狀態，讓通訊層偵測友機已消失。

IBM MQ MQI clients 的錯誤訊息

當 IBM MQ MQI client 系統發生錯誤時，會將錯誤訊息放入 IBM MQ 系統錯誤檔案中。

- 在 UNIX and Linux 系統上，這些檔案位於 /var/mqm/errors 目錄中

- 在 Windows 上，這些檔案位於 IBM MQ MQI client 安裝的錯誤子目錄中。此目錄通常是 C:\Program Files\IBM\MQ\errors。
 - 在 IBM i 上，這些檔案位於 /QIBM/UserData/mqm/errors 目錄中
- 某些用戶端錯誤也可以記錄在與用戶端所連接伺服器相關聯的 IBM MQ 錯誤檔中。

多重播送疑難排解

下列提示和要訣沒有明顯的順序，且可能在發行文件的新版本時新增至。它們是與您正在執行的工作相關的主題，可能會節省您的時間。

在非多重播送網路上測試多重播送應用程式

使用此資訊來瞭解如何在本端 (而非透過多重播送網路) 測試 IBM MQ 多重播送應用程式。

開發或測試多重播送應用程式時，您可能尚未啟用多重播送網路。若要在本端執行應用程式，您必須編輯 `mqclient.ini` 檔案，如下列範例所示：

編輯 `MQ_DATA_PATH/mqclient.ini` 的 Multicast 段落中的 **Interface** 參數：

```
Multicast:
Interface      = 127.0.0.1
```

其中 `MQ_DATA_PATH` 是 IBM MQ 資料目錄 (`/var/mqm/mqclient.ini`) 的位置。

多重播送傳輸現在只使用本端迴圈配接卡。

為多重播送資料流量設定適當的網路

開發或測試多重播送應用程式時，在本端測試之後，您可能想要透過啟用多重播送的網路來測試它們。如果應用程式只在本端傳輸，您可能必須編輯 `mqclient.ini` 檔，如本節稍後所示。如果機器設定使用多個網路配接卡或虛擬專用網路 (VPN)，則 `mqclient.ini` 檔案中的 **Interface** 參數必須設為您要使用之網路配接卡的位置。

如果 `mqclient.ini` 檔案中存在 Multicast 段落，請編輯 **Interface** 參數，如下列範例所示：

變更：

```
Multicast:
Interface      = 127.0.0.1
```

至：

```
Multicast:
Interface      = IPAddress
```

其中 `IPAddress` 是多重播送資料流量在其上流動之介面的 IP 位址。

如果 `mqclient.ini` 檔案中沒有 Multicast 段落，請新增下列範例：

```
Multicast:
Interface      = IPAddress
```

其中 `IPAddress` 是多重播送資料流量在其上流動之介面的 IP 位址。

多重播送應用程式現在透過多重播送網路執行。

多重播送主題字串太長

如果您的 IBM MQ 多重播送主題字串遭到拒絕，原因碼為 `MQRC_TOPIC_STRING_ERROR`，可能是因為字串太長。

WebSphereMQ 多重播送對主題字串有 255 個字元限制。此限制表示必須小心處理樹狀結構內節點及葉節點的名稱；如果節點及葉節點的名稱太長，則主題字串可能會超出 255 個字元，並傳回 2425 (0979) (RC2425) :MQRC_TOPIC_STRING_ERROR 原因碼。建議儘量縮短主題字串，因為較長的主題字串可能會對效能造成不利影響。

多重播送主題拓撲問題

使用這些範例來瞭解不建議使用某些 IBM MQ 多重播送主題拓撲的原因。

如 IBM MQ 多重播送主題拓撲中所述，IBM MQ 多重播送支援要求每一個子樹狀結構在整個階層中都有自己的多重播送群組和資料串流。請勿對子樹狀結構及其母項使用不同的多重播送群組位址。

具類別網路 IP 定址方法對於多重播送位址具有指定的位址空間。完整的多重播送 IP 位址範圍是 224.0.0.0 到 239.255.255.255，但其中有些位址已保留。如需保留位址清單，請聯絡您的系統管理者，或參閱 <https://www.iana.org/assignments/multicast-addresses> 以取得相關資訊。建議您使用 239.0.0.0 至 239.255.255.255 範圍內的本端範圍多重播送位址。

建議的多重播送主題拓撲

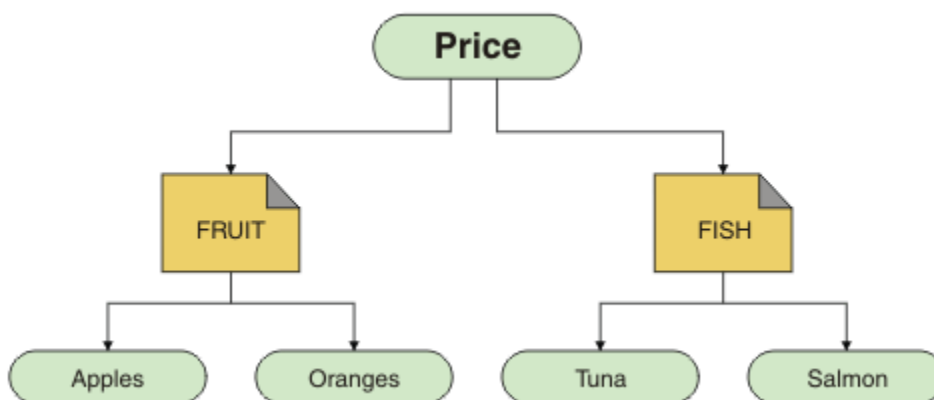
此範例與 IBM MQ 多重播送主題拓撲中的範例相同，並顯示 2 個可能的多重播送資料串流。雖然它是簡單的表示法，但它會示範 IBM MQ Multicast 設計的狀況類型，並在這裡顯示以對照 [第二個範例](#)：

```
DEF COMMINFO(MC1) GRPADDR(
227.20.133.1)
DEF COMMINFO(MC2) GRPADDR(227.20.133.2)
```

其中 227.20.133.1 和 227.20.133.2 是有效的多重播送位址。

這些主題定義用來建立主題樹狀結構，如下圖所示：

```
DEFINE TOPIC(FRUIT) TOPICSTRING('Price/FRUIT') MCAST(ENABLED) COMMINFO(MC1)
DEFINE TOPIC(FISH) TOPICSTRING('Price/FISH') MCAST(ENABLED) COMMINFO(MC2)
```



每一個多重播送通訊資訊 (COMMINFO) 物件都代表不同的資料串流，因為它們的群組位址不同。在此範例中，主題 FRUIT 定義為使用 COMMINFO 物件 MC1，而主題 FISH 定義為使用 COMMINFO 物件 MC2。

IBM MQ 多重播送的主題字串限制為 255 個字元。此限制表示必須小心處理樹狀結構中的節點及葉節點名稱；如果節點及葉節點名稱太長，則主題字串可能會超出 255 個字元，並傳回 MQRC_TOPIC_STRING_ERROR 原因碼。

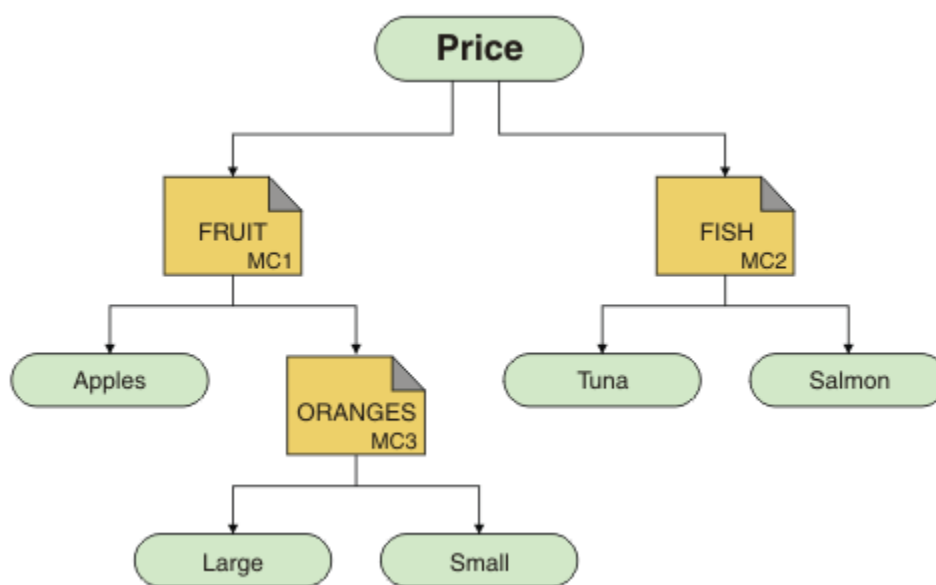
非建議的多重播送主題拓撲

此範例透過新增另一個主題物件 ORANGES 來延伸前一個範例，該主題物件定義為使用另一個 COMMINFO 物件定義 (MC3)：

```
DEF COMMINFO(MC1) GRPADDR(227.20.133.1
)
DEF COMMINFO(MC2) GRPADDR(227.20.133.2)
DEF COMMINFO(MC3) GRPADDR(227.20.133.3)
```

其中 227.20.133.1、227.20.133.2 及 227.20.133.3 是有效的多重播送位址。
這些主題定義用來建立主題樹狀結構，如下圖所示：

```
DEFINE TOPIC(FRUIT) TOPICSTRING('Price/FRUIT') MCAST(ENABLED) COMMINFO(MC1)
DEFINE TOPIC(FISH) TOPICSTRING('Price/FISH') MCAST(ENABLED) COMMINFO(MC2)
DEFINE TOPIC(ORANGES) TOPICSTRING('Price/FRUIT/ORANGES') MCAST(ENABLED) COMMINFO(MC3)
```



雖然可以建立這種多重播送拓撲，但不建議這樣做，因為應用程式可能不會收到他們預期的資料。

在 'Price/FRUIT/#' 上訂閱的應用程式會在 COMMINFO MC1 群組位址上接收多重播送傳輸。應用程式預期在主題樹狀結構中該點或以下接收所有主題的發佈。

不過，訂閱者不會接收應用程式在 'Price/FRUIT/ORANGES/Small' 上發佈所建立的訊息，因為這些訊息是在 COMMINFO MC3 的群組位址上傳送。

佇列管理程式叢集疑難排解

使用這裡提供的核對清單及子主題中提供的建議，可協助您在使用佇列管理程式叢集時偵測及處理問題。

開始之前

如果您的問題與使用叢集發佈/訂閱傳訊相關，而不是與一般叢集作業相關，請參閱 [第 142 頁的『發佈/訂閱叢集的遞送: 行為注意事項』](#)。

程序

- 檢查叢集通道是否都已配對。

每一個叢集傳送端通道都會連接至同名的叢集接收端通道。如果沒有與遠端佇列管理程式上叢集傳送端通道同名的本端叢集接收端通道，則無法運作。

- 請檢查您的通道是否在執行中。沒有任何通道應該永久處於 **RETRYING** 狀態。

使用下列指令顯示哪些通道正在執行中：

```
runmqsc display chstatus(*)
```

如果您有通道處於 **重試中** 狀態，則通道定義可能發生錯誤，或遠端佇列管理程式可能不在執行中。當通道處於此狀態時，訊息可能會建置在傳輸佇列上。如果完整儲存庫的通道處於此狀態，則叢集物件 (例如佇列和佇列管理程式) 的定義會變成過期且在整個叢集中不一致。

- 檢查沒有通道處於 **STOPPED** 狀態。

當您手動停止通道時，通道會進入 **STOPPED** 狀態。可以使用下列指令來重新啟動已停止的通道：

```
runmqsc start channel(xyz)
```

叢集佇列管理程式會根據需要自動定義叢集中其他佇列管理程式的叢集通道。除非先前已手動停止，否則這些自動定義的叢集通道會根據佇列管理程式的需要自動啟動。如果手動停止自動定義的叢集通道，佇列管理程式會記住它已手動停止，且未來不會自動啟動它。如果您需要停止通道，請記得在方便時重新啟動它，否則請發出下列指令：

```
stop channel(xyz) status(inactive)
```

status(inactive) 選項可讓佇列管理程式稍後重新啟動通道 (如果需要的話)。

- 請檢查叢集中的所有佇列管理程式是否都知道所有完整儲存庫。

您可以使用下列指令來執行此動作：

```
runmqsc display clusqmgr(*) qmtype
```

局部儲存庫可能不知道所有其他局部儲存庫。所有完整儲存庫都應該知道叢集中的所有佇列管理程式。如果遺漏叢集佇列管理程式，這可能表示某些通道未正確執行。

- 請檢查叢集中的每個佇列管理程式 (完整儲存庫和局部儲存庫) 是否都有手動定義的叢集接收端通道在執行中，且定義在正確的叢集中。

若要查看哪些其他佇列管理程式正在與叢集接收端通道交談，請使用下列指令：

```
runmqsc display chstatus(*) rqmname
```

請檢查每一個手動定義的叢集接收端是否有一個 **conname** 參數定義為 **ipaddress (port)**。如果沒有正確的連線名稱，則其他佇列管理程式不知道要在重新連接時使用的連線詳細資料。

- 請檢查每個局部儲存庫是否都有手動定義的叢集傳送端通道執行至完整儲存庫，並定義在正確的叢集中。

叢集傳送端通道名稱必須符合另一個佇列管理程式上的叢集接收端通道名稱。

- 請檢查每個完整儲存庫是否都有手動定義的叢集傳送端通道執行至所有其他完整儲存庫，並定義在正確的叢集中。

叢集傳送端通道名稱必須符合另一個佇列管理程式上的叢集接收端通道名稱。每一個完整儲存庫都不會保留叢集中其他完整儲存庫的記錄。它假設其具有手動定義之叢集傳送端通道的任何佇列管理程式都是完整儲存庫。

- 請檢查無法傳送的郵件佇列。

佇列管理程式無法遞送的訊息會傳送至無法傳送的郵件佇列。

- 請檢查對於每一個局部儲存庫佇列管理程式，您已定義單一叢集傳送端通道給其中一個完整儲存庫佇列管理程式。

此通道充當「引導」通道，局部儲存庫佇列管理程式會透過此通道初次加入叢集。

- 請檢查預期的完整儲存庫佇列管理程式是否為實際的完整儲存庫，且位於正確的叢集中。

您可以使用下列指令來執行此動作：

```
runmqsc display qmgr repos reposnl
```

- 請檢查訊息未建置在傳輸佇列或系統佇列上。

您可以使用下列指令來檢查傳輸佇列：

```
runmqsc display ql(*) curdepth where (usage eq xmitq)
```

您可以使用下列指令來檢查系統佇列：

```
display ql(system*) curdepth
```

相關工作

[第 9 頁的『對 UNIX, Linux, and Windows 進行起始檢查』](#)

在 UNIX, Linux, and Windows 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

[第 24 頁的『對 z/OS 進行起始檢查』](#)

在 z/OS 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

[第 16 頁的『對 IBM i 進行起始檢查』](#)

在 IBM i 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

相關資訊

[配置佇列管理程式叢集](#)

[訊息及原因碼](#)

執行 REFRESH CLUSTER 時看到的應用程式問題

發出 **REFRESH CLUSTER** 會干擾叢集。在 **REFRESH CLUSTER** 處理完成之前，它可能會使叢集物件暫時不可見。這可能會影響執行中的應用程式。這些注意事項說明您可能會看到的部分應用程式問題。

您可能從 MQOPEN、MQPUT 或 MQPUT1 呼叫中看到的原因碼

在 **REFRESH CLUSTER** 期間，可能會看到下列原因碼。本主題後面的一節會說明其中每一個代碼出現的原因。

- 2189 MQRC_CLUSTER_RESOLUTION_ERROR
- 2085 MQRC_UNKNOWN_OBJECT_NAME
- 2041 MQRC_OBJECT_CHANGED
- 2082 MQRC_UNKNOWN_ALIAS_BASE_Q
- 2270 MQRC_NO_DESTINATIONS_AVAILABLE

所有這些原因碼都指出 IBM MQ 程式碼中的某個層次或另一個層次的名稱查閱失敗，如果應用程式在整個 **REFRESH CLUSTER** 作業期間執行，則這是預期的。

REFRESH CLUSTER 作業可能在本端及/或遠端進行，以導致這些結果。如果完整儲存庫非常忙碌，則它們出現的可能性特別高。如果 **REFRESH CLUSTER** 活動在完整儲存庫本端執行，或在叢集或叢集中由完整儲存庫負責的其他佇列管理程式遠端執行，則會發生此情況。

對於暫時缺席且即將恢復的叢集佇列，則所有這些原因碼都是可暫時重試的條件（雖然對於 2041 MQRC_OBJECT_CHANGED，要決定條件是否可重試可能有點複雜）。如果與應用程式規則一致（例如服務時間上限），您應該重試大約一分鐘，以提供時間讓 **REFRESH CLUSTER** 活動完成。對於適度大小的叢集，完成可能要快得多。

如果從 **MQOPEN** 傳回任何這些原因碼，則不會建立任何物件控點，但稍後重試應該會成功建立一個物件控點。

如果從 **MQPUT** 傳回任何這些原因碼，則不會自動關閉物件控點，且重試最終應該會成功，而不需要先關閉物件控點。不過，如果應用程式使用 **bind-on-open** 選項開啟控點，因此需要所有訊息都移至相同的通道，則（與應用程式的預期相反）並不保證重試的 **put** 會像之前一樣移至相同的通道或佇列管理程式。因此，在這種情況下，最好關閉物件控點並開啟新的控點，以重新取得「開啟時連結」語意。

如果從 **MQPUT1** 傳回任何這些原因碼，則不知道在作業的 **open** 或 **put** 部分期間是否發生問題。不論它是什麼，都可以重試作業。在此情況下，沒有可擔心的「開啟時連結」語意，因為 **MQPUT1** 作業是在一個連續動作中執行的 **open-put-close** 序列。

多躍點實務範例

如果訊息流程納入多重中繼站（如下列範例所示），則在遠離應用程式的佇列管理程式上可能會發生 **REFRESH CLUSTER** 所導致的名稱查閱失敗。在該情況下，應用程式會收到成功（零）回覆碼，但如果名稱查閱失敗，則會阻止 **CLUSRCVR** 通道程式將訊息遞送至任何適當的目的地佇列。相反地，根據訊息的持續性，**CLUSRCVR** 通道程式會遵循一般規則，將訊息寫入無法傳送的郵件佇列。與該作業相關聯的原因碼如下：

- 2001 MQRC_ALIAS_BASE_Q_TYPE_ERROR

如果有持續訊息，且未定義任何無法傳送的郵件佇列來接收它們，則您會看到通道結束。

以下是多重中繼站實務範例：

- 佇列管理程式 **QM1** 上的 **MQOPEN** 指定 **Q2**。
- **Q2** 在遠端佇列管理程式 **QM2** 的叢集中定義為別名。
- 訊息到達 **QM2**，並發現 **Q2** 是 **Q3** 的別名。
- **Q3** 在遠端佇列管理程式 **QM3** 的叢集中定義為 **qlocal1**。
- 訊息會到達 **QM3**，並放置到 **Q3**。

當您測試多重中繼站時，可能會看到下列佇列管理程式錯誤日誌項目：

- 在傳送端和接收端，當已備妥無法傳送的郵件佇列且有持續訊息時：

AMQ9544: 訊息未放入目的地佇列

在處理通道 'CHLNAME' 期間，無法將一或多個訊息放入目的地佇列，並嘗試將它們放入無法傳送的郵件佇列。佇列的位置是 \$，其中 1 是本端無法傳送的郵件佇列，2 是遠端無法傳送的郵件佇列。

- 在接收端，當無法傳送郵件的佇列未備妥，且有持續訊息時：

AMQ9565: 未定義任何無法傳送郵件的佇列

AMQ9599: 程式無法開啟佇列管理程式物件

AMQ9999: 通道程式異常結束

- 在傳送端，當無法傳送的郵件佇列未備妥且有持續訊息時：

AMQ9506: 訊息接收確認失敗

AMQ9780: 由於發生錯誤，正在結束遠端機器 'a.b.c.d(1415)' 的通道

AMQ9999: 通道程式異常結束

執行 REFRESH CLUSTER 時可能顯示這些原因碼之原因的詳細資料

2189 (088D) (RC2189): MQRC_CLUSTER_RESOLUTION_ERROR

本端佇列管理程式向其完整儲存庫詢問佇列名稱是否存在。在寫在程式中的逾時 10 秒內，沒有來自完整儲存庫的回應。這是因為要求訊息或回應訊息在佇列中等待處理，且在適當時間將會清除此狀況。在應用程式中，條件是可重試，且在那些內部機制完成時將會成功。

MQRC_UNKNOWN_OBJECT_NAME (2085, X'825')

本端佇列管理程式已詢問（或先前已詢問）其完整儲存庫是否存在佇列名稱。完整儲存庫已回應，表示他們不知道佇列名稱。在 **REFRESH CLUSTER** 在完整和局部儲存庫上進行的環境定義中，佇列的擁有者可能尚未將佇列的相關資訊告知完整儲存庫。或者，它可能已執行此動作，但包含此資訊的內部訊息位

於佇列中等待處理，在此情況下，將在適當時間清除此狀況。在應用程式中，條件是可重試，且在那些內部機制完成時將會成功。

2041 (07F9) (RC2041) :MQRC_OBJECT_changed

最有可能從 bind-on-open **MQPUT** 中看到。本端佇列管理程式知道佇列名稱的存在，以及它所在的遠端佇列管理程式。在完整和局部儲存庫上發生的 **REFRESH CLUSTER** 環境定義中，佇列管理程式的記錄已刪除，且正在從完整儲存庫查詢。在應用程式中，決定條件是否可重試有點複雜。事實上，如果重試 **MQPUT**，則當那些內部機制完成瞭解遠端佇列管理程式的工作時，它將會成功。不過，不保證會使用相同的佇列管理程式。遵循通常在收到 **MQRC_OBJECT_CHANGED** 時建議的方法更安全，即關閉物件控點並重新開啟新的控點。

MQRC_UNKNOWN_ALIAS_BASE_Q (2082 , X'822')

與 2085 **MQRC_UNKNOWN_OBJECT_NAME** 狀況類似，當使用本端別名時，會看到此原因碼，且其 **TARGET** 是由於先前針對原因碼 2085 所說明的原因而無法存取的叢集佇列。

MQRC_ALIAS_BASE_Q_TYPE_ERROR (2001 , X'7D1')

通常在應用程式中不會看到此原因碼。只可能在佇列管理程式錯誤日誌中看到，與嘗試將訊息傳送至無法傳送的郵件佇列相關。**CLUSRCVR** 通道程式已收到來自其夥伴 **CLUSSDR** 的訊息，並正在決定放置它的位置。此實務範例只是先前針對原因碼 2082 及 2085 所說明之相同條件的變異。在此情況下，相較於在應用程式 **MQPUT** 或 **MQOPEN** 期間處理別名的位置，當在 MQ 產品中的不同點處理別名時，會看到原因碼。

2270 (08DE) (RC2270): MQRC_NO_DESTINATIONS_AVAILABLE

當應用程式使用以 **MQOO_BIND_NOT_FIXED** 開啟的佇列，且在 **REFRESH CLUSTER** 處理完成之前短時間內無法使用目的地物件時。

進一步評論

如果此環境中有任何叢集化發佈/訂閱活動，則 **REFRESH CLUSTER** 可能會有其他不想要的效果。例如，暫時遺失訂閱者的訂閱，然後發現他們遺失訊息。請參閱 [發佈/訂閱叢集的 REFRESH CLUSTER 注意事項](#)。

相關資訊

發佈/訂閱叢集的 [REFRESH CLUSTER 考量](#)

叢集作業：使用 [REFRESH CLUSTER 最佳作法](#)

[MQSC 指令參照: REFRESH CLUSTER](#)

叢集傳送端通道持續嘗試啟動

請檢查佇列管理程式和接聽器是否在執行中，以及叢集傳送端和叢集接收端通道定義是否正確。

症狀

```
1 : display chs(*)
AMQ8417: Display Channel Status details.
CHANNEL (DEMO.QM2)                XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
CONNAME (computer.ibm.com(1414))
CURRENT                            CHLTYPE (CLUSSDR)
STATUS (RETRYING)
```

原因

1. 遠端佇列管理程式無法使用。
2. 定義給本端手動叢集傳送端通道或遠端叢集接收端通道的參數不正確。

解決方案

請檢查問題是否為遠端佇列管理程式的可用性。

1. 有任何錯誤訊息嗎？
2. 佇列管理程式是否在作用中？

3. 接聽器是否在執行中?
4. 叢集傳送端通道是否能夠啟動?

如果遠端佇列管理程式可用，通道定義是否有問題? 檢查叢集佇列管理程式的定義類型，以查看通道是否持續嘗試啟動; 例如:

```
1 : dis clusqmgr(*) deftype where(channel eq DEMO.QM2)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM2) CHANNEL(DEMO.QM2) CLUSTER(DEMO)
DEFTYPE(CLUSSDRA)
```

如果定義類型是 CLUSSDR，則通道會使用本端手動叢集傳送端定義。請變更本端手動叢集傳送端定義中任何不正確的參數，然後重新啟動通道。

如果定義類型是 CLUSSDRA 或 CLUSSDRB，則通道正在使用自動定義的叢集傳送端通道。自動定義的叢集傳送端通道是以遠端叢集接收端通道的定義為基礎。變更遠端叢集接收端定義中任何不正確的參數。例如，conname 參數可能不正確:

```
1 : alter chl(demo.qm2) chltype(clusrcvr) conname('newhost(1414)')
AMQ8016: IBM MQ channel changed.
```

遠端叢集接收端定義的變更會延伸出至任何感興趣的叢集佇列管理程式。對應的自動定義通道會相應地更新。您可以檢查已變更的參數，以檢查是否已正確傳播更新項目。例如:

```
1 : dis clusqmgr(qm2) conname
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM2) CHANNEL(DEMO.QM2) CLUSTER(DEMO) CONNAME(newhost(1414))
```

如果自動定義定義現在正確，請重新啟動通道。

DISPLAY CLUSQMGR show CLUSQMGR names starting SYSTEM.TEMP.

佇列管理程式尚未從手動定義的 CLUSSDR 通道所指向的完整儲存庫佇列管理程式收到任何資訊。請檢查是否已正確定義叢集通道。

症狀

Multi

```
1 : display clusqmgr(*)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM1) CLUSTER(DEMO)
CHANNEL(DEMO.QM1)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(SYSTEM.TEMPUUID.computer.<yourdomain>(1414))
CLUSQMGR(DEMO) CHANNEL(DEMO.QM2)
```

z/OS

```
CSQM201I +CSQ2 CSQMDRTC DISPLAY CLUSQMGR DETAILS
CLUSQMGR(SYSTEM.TEMPQMGR.<HOSTNAME>(1716))
CLUSQMGR(DEMO)
CHANNEL(TO.CSQ1.DEMO)
END CLUSQMGR DETAILS
```

原因

佇列管理程式尚未從手動定義的 CLUSSDR 通道所指向的完整儲存庫佇列管理程式收到任何資訊。手動定義的 CLUSSDR 通道必須處於執行中狀態。

解決方案

請檢查 CLUSRCVR 定義是否也正確，特別是其 CONNAME 及 CLUSTER 參數。如果定義錯誤，請變更通道定義。

您也需要提供正確的權限給 SYSTEM.CLUSTER.TRANSMIT.QUEUE：

```
setmqaut -m <QMGR Name> -n SYSTEM.CLUSTER.TRANSMIT.QUEUE -t q -g mqm +all
```

遠端佇列管理程式可能需要一些時間來嘗試新的重新啟動，並以更正的定義來啟動其通道。

回覆碼 = 2035 MQRC_NOT_AUTHORIZED

由於各種原因會顯示 RC2035 原因碼，包括開啟佇列或通道時發生錯誤、嘗試使用具有管理者權限的使用者 ID 時收到錯誤、使用 IBM MQ JMS 應用程式時發生錯誤，以及在叢集上開啟佇列。MQS_REPORT_NOAUTH 和 MQSAUTHERRORS 可用來進一步診斷 RC2035。

特定問題

如需相關資訊，請參閱 [MQRC_NOT_AUTHORIZED](#)：

- JMSWMQ2013 安全鑑別無效
- 在佇列或通道上 MQRC_NOT_AUTHORIZED
- 以管理者身分 MQRC_NOT_AUTHORIZED (用戶端上的 AMQ4036)
- MQS_REPORT_NOAUTH 和 MQSAUTHERRORS 環境變數

開啟叢集中的佇列

此錯誤的解決方案取決於佇列是否在 z/OS 上。在 z/OS 上，請使用您的安全管理程式。在其他平台上，建立叢集佇列的本端別名，或授權所有使用者存取傳輸佇列。

症狀

嘗試開啟叢集中的佇列時，應用程式會收到回覆碼 2035 MQRC_NOT_AUTHORIZED。

原因

嘗試開啟叢集中的佇列時，您的應用程式會收到回覆碼 MQRC_NOT_AUTHORIZED。該佇列的授權正確。應用程式可能未獲授權放置到叢集傳輸佇列。

解決方案

解決方案取決於佇列是否在 z/OS 上。請參閱相關資訊主題。

嘗試開啟叢集中的佇列時，回覆碼 = 2085 MQRC_UNKNOWN_OBJECT_NAME

症狀

嘗試在叢集中開啟佇列時，應用程式會收到回覆碼 2085 MQRC_UNKNOWN_OBJECT_NAME。

原因

物件所在的佇列管理程式，或此佇列管理程式可能未順利進入叢集。

解決方案

請確定它們每一個都可以顯示叢集中的所有完整儲存庫。另請確定完整儲存庫的 CLUSSDR 通道正在嘗試啟動。

如果佇列位於叢集中，請確認您已使用適當的開啟選項。您無法從遠端叢集佇列取得訊息，因此請確定開啟選項僅用於輸出。

```
1 : display clusqmgr(*) qmtype status
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM1)          CLUSTER(DEMO)
CHANNEL(DEMO.QM1)      QMTYPE(NORMAL)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM2)          CLUSTER(DEMO)
CHANNEL(DEMO.QM2)      QMTYPE(REPOS)
STATUS(RUNNING)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM3)          CLUSTER(DEMO)
CHANNEL(DEMO.QM3)      QMTYPE(REPOS)
STATUS(RUNNING)
```

註: 當 IBM MQ 與 WebSphere Application Server 搭配使用時，如果您有 JMS 應用程式連接至屬於 IBM MQ 叢集的 IBM MQ 佇列管理程式，且您的 JMS 應用程式嘗試存取叢集中其他位置的叢集佇列，則也可能會看到此問題。如果您的應用程式想要開啟位於叢集中的叢集佇列，或指定管理叢集佇列之叢集中的佇列管理程式名稱，則必須將佇列管理程式保留空白。

相關資訊

[MQRC_UNKNOWN_OBJECT_NAME \(2085, X'825'\)](#)

嘗試開啟叢集中的佇列時，回覆碼 = 2189 MQRC_CLUSTER_RESOLUTION_ERROR

請確定連至完整儲存庫的 CLUSSDR 通道不會持續嘗試啟動。

症狀

嘗試開啟叢集中的佇列時，應用程式會收到回覆碼 2189 MQRC_CLUSTER_RESOLUTION_ERROR。

原因

第一次開啟佇列時，佇列管理程式無法聯絡任何完整儲存庫。

解決方案

請確定連至完整儲存庫的 CLUSSDR 通道不會持續嘗試啟動。

```
1 : display clusqmgr(*) qmtype status
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM1)          CLUSTER(DEMO)
CHANNEL(DEMO.QM1)      QMTYPE(NORMAL)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM2)          CLUSTER(DEMO)
CHANNEL(DEMO.QM2)      QMTYPE(REPOS)
STATUS(RUNNING)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM3)          CLUSTER(DEMO)
CHANNEL(DEMO.QM3)      QMTYPE(REPOS)
STATUS(RUNNING)
```

相關資訊

[2189 \(088D\) \(RC2189\): MQRC_CLUSTER_RESOLUTION_ERROR](#)

傳回 code=2082 MQRC_UNKNOWN_ALIAS_BASE_Q 開啟叢集中的佇列

嘗試開啟叢集中的佇列時，應用程式會取得 rc=2082 MQRC_UNKNOWN_ALIAS_BASE_Q。

問題

已發出 MQOPEN 或 MQPUT1 呼叫，並將別名佇列指定為目標，但別名佇列屬性中的 *BaseQName* 無法辨識為佇列名稱。

當 *BaseQName* 是無法順利解析的叢集佇列名稱時，也會發生此原因碼。

MQRC_UNKNOWN_ALIAS_BASE_Q 可能指出應用程式正在指定其連接之佇列管理程式的 **ObjectQmgrName**，以及管理別名佇列的佇列管理程式。這表示佇列管理程式會在指定的佇列管理程式上尋找別名目標佇列，但因為別名目標佇列不在本端佇列管理程式上而失敗。

解決方案

將 **ObjectQmgrName** 參數保留空白，以便叢集作業決定要遞送至哪個佇列管理程式。

如果佇列位於叢集中，請確認您已使用適當的開啟選項。您無法從遠端叢集佇列取得訊息，因此請確定開啟選項僅用於輸出。

相關資訊

[MQRC_UNKNOWN_ALIAS_BASE_Q \(2082, X'822'\)](#)

訊息未到達目的地佇列

請確定對應的叢集傳輸佇列是空的，且目的地佇列管理程式的通道正在執行中。

症狀

訊息未到達目的地佇列。

原因

訊息可能停留在其原始佇列管理程式。

解決方案

1. 識別正在將訊息傳送至目的地的傳輸佇列，以及通道的狀態。

```
1 : dis clusqmgr(QM1) CHANNEL(*) STATUS DEFTYPE QMTYPE XMITQ
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM1) CLUSTER(DEMO)
CHANNEL(DEMO.QM1) DEFTYPE(CLUSSDRA)
QMTYPE(NORMAL) STATUS(RUNNING)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.DEMO.QM1)
```

2. 請確定叢集傳輸佇列是空的。

```
1 : display ql(SYSTEM.CLUSTER.TRANSMIT.DEMO.QM1) curdepth
AMQ8409: Display Queue details.
QUEUE(SYSTEM.CLUSTER.TRANSMIT.DEMO.QM1) CURDEPTH(0)
```

放入叢集別名佇列的訊息會移至 **SYSTEM.DEAD.LETTER.QUEUE**

叢集別名佇列解析為不存在的本端佇列。

症狀

放入別名佇列的訊息會移至 **SYSTEM.DEAD.LETTER.QUEUE**，原因為 **MQRC_UNKNOWN_ALIAS_BASE_Q**。

原因

訊息會遞送至已定義叢集別名佇列的佇列管理程式。未在該佇列管理程式上定義本端目標佇列。因為已使用 **MQOO_BIND_ON_OPEN** 開啟選項放置訊息，所以佇列管理程式無法將訊息重新排入佇列。

使用 **MQOO_BIND_ON_OPEN** 時，會牢牢連結叢集佇列別名。解析的名稱是目標佇列的名稱，以及在其中定義叢集佇列別名的任何佇列管理程式。佇列管理程式名稱會放在傳輸佇列標頭中。如果目標佇列不存在於訊息傳送至的佇列管理程式上，則會將訊息放置在無法傳送的郵件佇列上。未重新計算目的地，因為傳輸標

頭包含 MQ00_BIND_ON_OPEN 所解析的目標佇列管理程式名稱。如果已使用 MQ00_BIND_NOT_FIXED 開啟別名佇列，則傳輸佇列標頭會包含空白佇列管理程式名稱，且會重新計算目的地。在此情況下，如果在叢集中的其他位置定義本端佇列，則會在該處傳送訊息。

解決方案

1. 變更所有別名佇列定義以指定 DEFBIND (NOTFIXED)。
2. 開啟佇列時，請使用 MQ00_BIND_NOT_FIXED 作為開啟選項。
3. 如果您指定 MQ00_BIND_ON_OPEN，請確保將叢集別名解析為與別名定義在相同佇列管理程式上的本端佇列。

佇列管理程式叢集中佇列及通道的相關資訊已過期

症狀

DISPLAY QCLUSTER 和 DISPLAY CLUSQMGR 會顯示過期的物件。

原因

叢集的更新項目只會透過手動定義的 CLUSSDR 通道在完整儲存庫之間流動。在叢集形成 CLUSSDR 通道之後，會顯示為 DEFTYPE (CLUSSDRB) 通道，因為它們同時是手動及自動通道。必須有足夠的 CLUSSDR 通道，才能在所有完整儲存庫之間形成完整網路。

解決方案

- 請檢查物件所在的佇列管理程式及本端佇列管理程式是否仍連接至叢集。
- 檢查每一個佇列管理程式是否可以顯示叢集中的所有完整儲存庫。
- 請檢查連至完整儲存庫的 CLUSSDR 通道是否持續嘗試重新啟動。
- 請檢查完整儲存庫是否已定義足夠的 CLUSSDR 通道，以正確地將它們連接在一起。

```
1 : dis clusqmgr(QM1) CHANNEL(*) STATUS DEFTYPE QMTYPE
XMITQ
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM1) CLUSTER(DEMO)
CHANNEL (DEMO.QM1) DEFTYPE(CLUSSDRA)
QMTYPE(NORMAL) STATUS(RUNNING)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.DEMO.QM1)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM2) CLUSTER(DEMO)
CHANNEL (DEMO.QM2) DEFTYPE(CLUSRCVR)
QMTYPE(REPOS)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.DEMO.QM2)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM3) CLUSTER(DEMO)
CHANNEL (DEMO.QM3) DEFTYPE(CLUSSDRB)
QMTYPE(REPOS) STATUS(RUNNING)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.DEMO.QM3)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM4) CLUSTER(DEMO)
CHANNEL (DEMO.QM4) DEFTYPE(CLUSSDRA)
QMTYPE(NORMAL) STATUS(RUNNING)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.DEMO.QM4)
```

叢集中沒有任何變更反映在本端佇列管理程式中

儲存庫管理程式處理程序未處理儲存庫指令，可能是因為接收或處理指令佇列中的訊息時發生問題。

症狀

叢集中沒有任何變更反映在本端佇列管理程式中。


原因

儲存庫管理程式處理程序未處理儲存庫指令。

解決方案

1. 確認 SYSTEM.CLUSTER.COMMAND.QUEUE 是空的。

```
1 : display ql(SYSTEM.CLUSTER.COMMAND.QUEUE) curdepth
AMQ8409: Display Queue details.
QUEUE(SYSTEM.CLUSTER.COMMAND.QUEUE) CURDEPTH(0)
```

2.  檢查通道起始程式是否在 z/OS 上執行。
3. 請檢查錯誤日誌中沒有指出佇列管理程式暫時資源不足的錯誤訊息。

「DISPLAY CLUSQMGR」會顯示佇列管理程式兩次

使用 RESET CLUSTER 指令來移除佇列管理程式舊實例的所有追蹤資料。

```
1 : display clusqmgr(QM1) qmid
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM1) CLUSTER(DEMO)
CHANNEL(DEMO.QM1) QMID(QM1_2002-03-04_11.07.01)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(QM1) CLUSTER(DEMO)
CHANNEL(DEMO.QM1) QMID(QM1_2002-03-04_11.04.19)
```

在忽略舊版佇列管理程式的情況下，叢集正常運作。大約 90 天之後，叢集對舊版佇列管理程式的知識會到期，並自動刪除。不過，您可能偏好手動刪除此資訊。

原因

1. 佇列管理程式可能已刪除，然後重建並重新定義。
2. 它可能已在 z/OS 上冷啟動，而未先遵循從叢集中移除佇列管理程式的程序。

解決方案

若要立即移除佇列管理程式的所有追蹤，請從完整儲存庫佇列管理程式中使用 RESET CLUSTER 指令。這個指令會從叢集中移除較舊的不想要佇列管理程式及其佇列。

```
2 : reset cluster(DEMO) qmid('QM1_2002-03-04_11.04.19') action(FORCEREMOVE) queues(yes)
AMQ8559: RESET CLUSTER accepted.
```

使用 RESET CLUSTER 指令會停止受影響佇列管理程式的自動定義叢集傳送端通道。在完成 RESET CLUSTER 指令之後，您必須手動重新啟動任何已停止的叢集傳送端通道。

佇列管理程式未重新結合叢集

發出 RESET 或 REFRESH 叢集指令之後，可能會停止從佇列管理程式到叢集的通道。請檢查叢集通道狀態，然後重新啟動通道。

症狀

在發出 RESET CLUSTER 和 REFRESH CLUSTER 指令之後，佇列管理程式不會重新結合叢集。

原因

RESET 和 REFRESH 指令的負面影響可能是通道已停止。通道會停止，以便在 RESET 或 REFRESH 指令完成時執行正確版本的通道。

解決方案

請檢查問題佇列管理程式與完整儲存庫之間的通道是否在執行中，必要的話，請使用 `START CHANNEL` 指令。

相關資訊

[叢集作業：使用 REFRESH CLUSTER 最佳作法](#)

叢集傳送端通道上設定的工作量平衡無法運作

您在叢集傳送端通道上指定的任何工作量平衡都可能被忽略。請改為在目標佇列管理程式的叢集接收端通道上指定叢集工作量通道屬性。

症狀

您已在叢集傳送端通道上指定一或多個叢集工作量通道屬性。產生的工作量平衡不是您所預期的。

原因

您在叢集傳送端通道上指定的任何工作量平衡都可能被忽略。如需此說明，請參閱 [叢集通道](#)。請注意，您仍會根據叢集預設值或目標佇列管理程式中相符叢集接收端通道上所設定的內容，取得某種形式的工作量平衡。

解決方案

在目標佇列管理程式的叢集接收端通道上指定叢集工作量通道屬性。

相關資訊

[CLWLPRTY 通道屬性](#)

[CLWLRANK 通道屬性](#)

[CLWLWGHT 通道屬性](#)

[NETPRTY 通道屬性](#)

已還原叢集中的過期資訊

還原佇列管理程式之後，其叢集資訊已過期。使用 `REFRESH CLUSTER` 指令重新整理叢集資訊。

問題

在 QM1 的映像檔備份之後，已還原叢集 DEMO 中的局部儲存庫，且它包含的叢集資訊已過期。

解決方案

在 QM1 上，發出指令 `REFRESH CLUSTER(DEMO)`。

註：對於大型叢集，使用 `REFRESH CLUSTER` 指令會干擾進行中的叢集，而此後每隔 27 天，當叢集物件自動將狀態更新傳送給所有相關的佇列管理程式時，會再次造成干擾。請參閱 [在大型叢集中重新整理可能影響叢集的效能及可用性](#)。

當您在 QM1 上執行 `REFRESH CLUSTER(DEMO)` 時，您會移除 QM1 關於叢集 DEMO 的所有資訊，但 QM1 本身及其本身佇列的知識以及如何存取叢集中完整儲存庫的資訊除外。然後，QM1 會聯絡完整儲存庫，並告知其本身及其佇列。QM1 是局部儲存庫，因此完整儲存庫不會立即告知 QM1 叢集中所有其他局部儲存庫的相關資訊。相反地，QM1 會透過它接收的資訊，慢慢地建立它對其他局部儲存庫的知識，當其他佇列和佇列管理程式中的每一個都是下一個作用中的叢集時。

錯誤地強制從完整儲存庫中移除叢集佇列管理程式

在已從儲存庫移除的佇列管理程式上發出指令 `REFRESH CLUSTER`，將佇列管理程式還原至完整儲存庫。

問題

在叢集 DEMO 中的完整儲存庫上錯誤地發出指令 `RESET CLUSTER(DEMO) QMNAME(QM1) ACTION(FORCEREMOVE)`。

解決方案

在 QM1 上，發出指令 `REFRESH CLUSTER(DEMO)`。

註: 對於大型叢集，使用 **REFRESH CLUSTER** 指令會干擾進行中的叢集，而此後每隔 27 天，當叢集物件自動將狀態更新傳送給所有相關的佇列管理程式時，會再次造成干擾。請參閱在大型叢集中重新整理可能影響叢集的效能及可用性。

已刪除可能的儲存庫訊息

送往佇列管理程式的訊息已從其他佇列管理程式中的 `SYSTEM.CLUSTER.TRANSMIT.QUEUE` 移除。在受影響的佇列管理程式上發出 `REFRESH CLUSTER` 指令，以還原資訊。

問題

以 QM1 為目的地的訊息已從其他佇列管理程式中的 `SYSTEM.CLUSTER.TRANSMIT.QUEUE` 移除，它們可能是儲存庫訊息。

解決方案

在 QM1 上，發出指令 `REFRESH CLUSTER(DEMO)`。

註: 對於大型叢集，使用 **REFRESH CLUSTER** 指令會干擾進行中的叢集，而此後每隔 27 天，當叢集物件自動將狀態更新傳送給所有相關的佇列管理程式時，會再次造成干擾。請參閱在大型叢集中重新整理可能影響叢集的效能及可用性。

QM1 會移除它所擁有的所有叢集相關資訊 DEMO，但與叢集中完整儲存庫的叢集佇列管理程式相關的資訊除外。假設此資訊仍然正確，QM1 會聯絡完整儲存庫。QM1 會通知完整儲存庫本身及其佇列。當佇列及佇列管理程式開啟時，它會回復存在於叢集中其他位置的佇列及佇列管理程式資訊。

同時移動兩個完整儲存庫

如果您同時將兩個完整儲存庫移至新的網址，則不會自動以新位址來更新叢集。請遵循程序來傳送新的網址。請一次移動一個儲存庫，以避免發生問題。

問題

叢集 DEMO 包含兩個完整儲存庫: QM1 和 QM2。他們都同時移至網路上的新位置。

解決方案

1. 變更 `CLUSRCVR` 及 `CLUSDR` 通道中的 `CONNAME`，以指定新的網址。
2. 變更其中一個佇列管理程式 (QM1 或 QM2)，使其不再是任何叢集的完整儲存庫。
3. 在已變更的佇列管理程式上，發出指令 `REFRESH CLUSTER(*) REPOS(YES)`。

註: 對於大型叢集，使用 **REFRESH CLUSTER** 指令會干擾進行中的叢集，而此後每隔 27 天，當叢集物件自動將狀態更新傳送給所有相關的佇列管理程式時，會再次造成干擾。請參閱在大型叢集中重新整理可能影響叢集的效能及可用性。

4. 請變更佇列管理程式，使它成為完整儲存庫。

建議

您可以避免此問題，如下所示:

1. 將其中一個佇列管理程式 (例如 QM2) 移至其新的網址。

2. 變更 QM2 CLUSRCVR 通道中的網址。
3. 啟動 QM2 CLUSRCVR 通道。
4. 等待其他完整儲存庫佇列管理程式 QM1 瞭解 QM2 的新位址。
5. 將另一個完整儲存庫佇列管理程式 QM1 移至其新的網址。
6. 變更 QM1 CLUSRCVR 通道中的網址。
7. 啟動 QM1 CLUSRCVR 通道。
8. 為了清楚起見，請變更手動定義的 CLUSSDR 通道，但在此階段中不需要這些通道即可正確執行叢集作業。

此程序會強制 QM2 重複使用來自正確 CLUSSDR 通道的資訊，以重新建立與 QM1 的聯絡，然後重建其叢集知識。此外，再次聯絡 QM1 之後，會根據 QM2 CLUSRCVR 定義中的 CONNAME，為它提供自己的正確網址。

叢集的不明狀態

透過從叢集中的所有局部儲存庫重建完整儲存庫，將所有完整儲存庫中的叢集資訊還原至已知狀態。

問題

在正常狀況下，完整儲存庫會交換叢集中佇列及佇列管理程式的相關資訊。如果重新整理一個完整儲存庫，則會從另一個完整儲存庫回復叢集資訊。

問題是如何完全重設叢集中的所有系統，以還原叢集的已知狀態。

解決方案

為了停止從完整儲存庫的不明狀態更新叢集資訊，所有到完整儲存庫的 CLUSRCVR 通道都會停止。CLUSSDR 通道會變更為非作用中。

當您重新整理完整儲存庫系統時，它們都無法通訊，因此它們會從相同的已清除狀態開始。

當您重新整理局部儲存庫系統時，它們會重新結合叢集，並將它重建成完整的佇列管理程式和佇列集。完整重建中的叢集資訊會還原至已知狀態。

註：對於大型叢集，使用 **REFRESH CLUSTER** 指令會干擾進行中的叢集，而此後每隔 27 天，當叢集物件自動將狀態更新傳送給所有相關的佇列管理程式時，會再次造成干擾。請參閱 [在大型叢集中重新整理可能影響叢集的效能及可用性](#)。

1. 在所有完整儲存庫佇列管理程式上，遵循下列步驟：
 - a. 變更完整儲存庫的佇列管理程式，使它們不再是完整儲存庫。
 - b. 解決任何不確定的 CLUSSDR 通道。
 - c. 等待 CLUSSDR 通道變成非作用中。
 - d. 停止 CLUSRCVR 通道。
 - e. 當所有完整儲存庫系統上的所有 CLUSRCVR 通道都已停止時，請發出指令 REFRESH CLUSTER(DEMO) REPOS(YES)。
 - f. 請變更佇列管理程式，使它們是完整儲存庫。
 - g. 啟動 CLUSRCVR 通道，以重新啟用它們進行通訊。
2. 在所有局部儲存庫佇列管理程式上，遵循下列步驟：
 - a. 解決任何不確定的 CLUSSDR 通道。
 - b. 請確定佇列管理程式上的所有 CLUSSDR 通道都已停止或非作用中。
 - c. 發出指令 REFRESH CLUSTER(DEMO) REPOS(YES)。

叢集佇列管理程式失敗時發生的情況

當叢集佇列管理程式失敗時，部分未遞送的訊息會傳送至叢集中的其他佇列管理程式。進行中的訊息會等待佇列管理程式重新啟動。使用高可用性機制來自動重新啟動佇列管理程式。

問題

如果訊息批次傳送至特定佇列管理程式，且該佇列管理程式變成無法使用，則傳送端佇列管理程式會發生什麼情況？

說明

除了 NPMSPEED (FAST) 通道上的非持續訊息之外，未遞送的訊息批次會取消至傳送端佇列管理程式上的叢集傳輸佇列。在 NPMSPEED (FAST) 通道上，未批次處理非持續訊息，且可能會遺失非持續訊息。

- 不確定的訊息及連結至無法使用的佇列管理程式的訊息，請等待直到佇列管理程式重新變成可用為止。
- 其他訊息會遞送至工作量管理常式所選取的替代佇列管理程式。

解決方案

可以透過將叢集佇列管理程式配置為多重實例佇列管理程式，或透過平台專用高可用性機制，自動重新啟動無法使用的叢集佇列管理程式。




儲存庫失敗時發生的情況

您如何知道儲存庫已失敗，以及要如何修正它？

問題

1. 叢集資訊會傳送至稱為 `SYSTEM.CLUSTER.COMMAND.QUEUE` 的本端佇列上的儲存庫 (不論完整或局部)。如果這個佇列已滿，可能是因為佇列管理程式已停止運作，叢集參考訊息會遞送至無法傳送郵件的佇列。
2. 儲存庫用完儲存體。

解決方案

1. 監視佇列管理程式日誌  或 z/OS 系統主控台上的訊息，以偵測 `SYSTEM.CLUSTER.COMMAND.QUEUE` 是否填滿。如果是的話，您需要執行應用程式，以從無法傳送郵件的佇列擷取訊息，並將它們重新遞送至正確的目的地。
2. 如果儲存庫佇列管理程式上發生錯誤，則訊息會告訴您發生了什麼錯誤，以及在嘗試重新啟動之前佇列管理程式等待的時間。
 -  在 IBM MQ for z/OS 上，`MQGET` 會停用 `SYSTEM.CLUSTER.COMMAND.QUEUE`。
 - 當您已識別並解決錯誤時，請啟用 `SYSTEM.CLUSTER.COMMAND.QUEUE`，以便佇列管理程式可以順利重新啟動。
3. 在儲存庫耗盡儲存體的不太可能事件中，儲存體配置錯誤會傳送至佇列管理程式日誌  或 z/OS 系統主控台。若要修正儲存體問題，請停止然後重新啟動佇列管理程式。當佇列管理程式重新啟動時，會自動配置更多儲存體來保留所有儲存庫資訊。

針對 MQPUT 停用叢集佇列會發生什麼情況

針對 MQPUT，可能會停用在於工作量平衡之叢集佇列的所有實例。將訊息放入佇列的應用程式會收到 `MQRC_CLUSTER_PUT_INHIBITED` 或 `MQRC_PUT_INHIBITED` 回覆碼。您可能想要修改此行為。

問題

當針對 MQPUT 停用叢集佇列時，其狀態會反映在對該佇列感興趣的每一個佇列管理程式的儲存庫中。工作量管理演算法會嘗試將訊息傳送至針對 MQPUT 啟用的目的地。如果未啟用 MQPUT 的目的地，且沒有佇列的本端實例，則指定 `MQOO_BIND_ON_OPEN` 的 `MQOPEN` 呼叫會將回覆碼 `MQRC_CLUSTER_PUT_INHIBITED` 傳回給應用程式。如果指定 `MQOO_BIND_NOT_FIXED`，或有佇列的本端實例，則 `MQOPEN` 呼叫成功，但後續的 MQPUT 呼叫失敗，回覆碼為 `MQRC_PUT_INHIBITED`。

解決方案

您可以撰寫使用者結束程式來修改工作量管理常式，以便將訊息遞送至針對 MQPUT 停用的目的地。

訊息可以到達已針對 MQPUT 停用的目的地。在佇列變成已停用時，訊息可能已在進行中，或者工作量結束程式可能已明確選擇目的地。目的地佇列管理程式中的工作量管理常式有許多處理訊息的方法：

- 如果有的話，請選擇另一個適當的目的地。
- 將訊息放在無法傳送郵件的佇列上。
- 如果沒有無法傳送郵件的佇列，則將訊息傳回給發送端

切換傳輸佇列時的潛在問題

切換傳輸佇列時可能遇到的部分問題、其原因及最可能的解決方案清單。

對 z/OS 上傳輸佇列的存取權不足

症狀

z/OS 上的叢集傳送端通道可能報告未獲授權開啟其傳輸佇列。

原因

通道正在切換或已切換傳輸佇列，且尚未授與通道起始程式存取新佇列的權限。

解決方案

授與通道起始程式與針對傳輸佇列 SYSTEM.CLUSTER.TRANSMIT.QUEUE。使用 DEFCLXQ 時，是 SYSTEM.CLUSTER.TRANSMIT. * * 每當新的佇列管理程式加入叢集時，即會避免發生此問題。

移動訊息失敗

症狀

通道會停止傳送訊息，且訊息會在通道的舊傳輸佇列中保持佇列狀態。

原因

佇列管理程式已停止將訊息從舊傳輸佇列移至新傳輸佇列，因為發生無法復原的錯誤。例如，新的傳輸佇列可能已滿，或其後端儲存體已耗盡。

解決方案

請檢閱寫入佇列管理程式錯誤日誌 (z/OS 上的工作日誌) 的錯誤訊息，以判斷問題並解決其主要原因。一旦解決，請重新啟動通道以回復切換處理程序，或停止通道，然後改用 **runswch1** (z/OS 上的 CSQUTIL)。

交換器未完成

症狀

佇列管理程式會反覆地發出訊息，指出它正在移動訊息。交換器永未完成，因為舊傳輸佇列上一律會保留訊息。

原因 1

通道的訊息放入舊傳輸佇列的速度比佇列管理程式將它們移至新傳輸佇列的速度更快。這可能是尖峰工作期間的暫時性問題，因為如果是常見的，則通道不太可能能夠足夠快地透過網路傳輸訊息。

原因 2

舊傳輸佇列上有通道的未確定的訊息。

解決方案

解決任何未確定的訊息的工作單元，及/或減少或暫停應用程式工作量，以容許移動訊息階段完成。

意外刪除傳輸佇列

症狀 1

由於移除相符的 CLCHNAME 值，通道非預期地切換。

症狀 2

使用 MQRC_UNKNOWN_XMIT_Q 放置到叢集佇列失敗。

症狀 3

通道異常結束，因為其傳輸佇列不存在。

症狀 4

佇列管理程式無法移動訊息來完成切換作業，因為它無法開啟舊或新的傳輸佇列。

原因

通道目前使用的傳輸佇列或其前一個傳輸佇列 (如果交換器尚未完成) 已被刪除。

解決方案

重新定義傳輸佇列。如果它是已刪除的舊傳輸佇列，則管理者也可以使用 **runswchl** 搭配 **-n** 參數 (或 CSQUTIL 搭配 MOVEMSGS (NO) on z/OS) 來完成切換作業。

請謹慎使用 **-n** 參數，因為如果使用不適當，通道的訊息可以完成並完成處理，但不會在舊傳輸佇列上更新。在此實務範例中，它是安全的，因為佇列不存在，因此無法完成任何訊息並完成處理。

佇列管理程式疑難排解

請使用這裡提供的建議來協助您解決使用佇列管理程式時可能發生的一般問題。

佇列管理程式無法使用錯誤

- **實務範例:** 您收到 佇列管理程式無法使用 錯誤。
- **說明:** 配置檔錯誤通常會阻止找到佇列管理程式，並導致 佇列管理程式無法使用 錯誤。在 Windows 上，當佇列管理程式啟動時，qm.ini 檔中的問題可能會導致 佇列管理程式無法使用 錯誤。
- **解決方案:** 確定配置檔存在，且 IBM MQ 配置檔參照正確的佇列管理程式和日誌目錄。在 Windows 上，檢查 qm.ini 檔案中是否有問題。

IBM MQ 協調 Db2 作為資源管理程式錯誤

- **實務範例:** 您從 IBM MQ Explorer 啟動佇列管理程式，且在協調 Db2 時發生問題。當您檢查佇列管理程式錯誤日誌時，會看到類似下列範例所示的錯誤：

```
23/09/2008 15:43:54-處理程序 (5508.1) 使用者 (MUSR_MQADMIN) 程式 (amqzma0.exe)
主機 (HOST_1) 安裝 (Installation1)
VMRF (7.1.0.0) 佇列管理程式 (A.B.C)
AMQ7604: XA 資源管理程式 'DB2 MQBankDB database' 在呼叫時無法使用
for xa_open. 在沒有這個資源管理程式的情況下，佇列管理程式會繼續進行。
```

- **說明:** 執行 IBM MQ 服務處理程序 amqsvc.exe 的使用者 ID (預設名稱為 MUSR_MQADMIN) 仍在執行中，其存取記號不包含群組 DB2USERS 的群組成員資格資訊。
- **解決方案:** 確定 IBM MQ 服務使用者 ID 是 DB2USERS 的成員之後，請使用下列指令序列：
 1. 停止該服務。
 2. 停止在相同使用者 ID 下執行的任何其他處理程序。
 3. 重新啟動這些處理程序。

將機器重新開機可確保先前的步驟，但並非必要。

未遞送訊息疑難排解

使用這裡提供的建議，可協助您在未順利遞送訊息時解決問題。

- **實務範例:** 當您預期訊息到達佇列時，訊息不會到達佇列。
- **說明:** 由於某些原因而無法遞送的訊息會置於無法傳送郵件的佇列中。
- **解決方案:** 您可以發出 MQSC DISPLAY QUEUE 指令來檢查佇列是否包含任何訊息。

如果佇列包含訊息，您可以使用提供的瀏覽範例應用程式 (amqsbcg)，利用 MQGET 呼叫來瀏覽佇列上的訊息。範例應用程式會逐步執行具名佇列管理程式之具名佇列上的所有訊息，同時顯示具名佇列上所有訊息的訊息描述子及訊息環境定義欄位。

您必須根據將訊息放入佇列的原因，決定如何處置在無法傳送郵件的佇列中找到的任何訊息。如果您未將無法傳送郵件的佇列與每一個佇列管理程式相關聯，則可能會發生問題。

如需無法傳送郵件的佇列及處理未遞送訊息的相關資訊，請參閱 [使用無法傳送郵件的佇列](#)。

TLS 疑難排解資訊

請使用這裡列出的資訊來協助您解決 TLS 系統的問題。

概觀

對於在用戶端上使用已啟用 FIPS 的非 FIPS 密碼所導致的錯誤，您會收到下列錯誤訊息：

JMSCMQ001

IBM MQ 呼叫失敗，完成碼為 2 ('MQCC_FAILED') 原因 2397 ('MQRC_JSSE_ERROR')

對於本主題內記載的所有其他問題，您會收到先前的錯誤訊息及/或下列錯誤訊息：

JMSWMQ0018

無法以連線模式 'connection_mode' 及主機名稱 'host_name' 連接至佇列管理程式 'queue_manager_name'

對於本主題內所記載的每一個問題，會提供下列資訊：

- 來自範例 SystemOut.log 或 Console 的輸出，詳細說明異常狀況的原因。
- 佇列管理程式錯誤日誌資訊。
- 問題的解決方案。

註：

- 您應該一律列出第一個異常狀況的堆疊及原因。

- 是否將錯誤資訊寫入 `stdout` 日誌檔，取決於應用程式的撰寫方式，以及您使用的架構。
- 範例程式碼包括堆疊及行號。此資訊是有用的指引，但堆疊及行號可能會從一個修正套件變更為另一個修正套件。您應該使用堆疊和行號作為尋找正確區段的指引，而不是使用專門用於診斷目的的資訊。

用戶端上未設定密碼組合

輸出

原因：

```
com.ibm.mq.jmqi.JmqiException: CC=2;RC=2397;AMQ9641: Remote CipherSpec error for channel
'SYSTEM.DEF.SVRCONN' to host ''. [3=SYSTEM.DEF.SVRCONN]
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.analyseErrorSegment(RemoteConnection.java:4176)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.receiveTSH(RemoteConnection.java:2969)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.initSess(RemoteConnection.java:1180)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.connect(RemoteConnection.java:838)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSessionFromNewConnection
(RemoteConnectionSpecification.java:409)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSession
(RemoteConnectionSpecification.java:305)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionPool.getSession(RemoteConnectionPool.java:146)
at com.ibm.mq.jmqi.remote.api.RemoteFAP.jmqiConnect(RemoteFAP.java:1868)
```

佇列管理程式錯誤日誌

AMQ9639: 遠端通道 'SYSTEM.DEF.SVRCONN' 未指定 CipherSpec。

解決方案

在用戶端上設定 CipherSuite，以便通道兩端具有相符的 CipherSuite 或 CipherSpec 配對。

伺服器上未設定密碼組合

輸出

原因：

```
com.ibm.mq.jmqi.JmqiException: CC=2;RC=2397;AMQ9641: Remote CipherSpec error
for channel 'SYSTEM.DEF.SVRCONN' to host ''. [3=SYSTEM.DEF.SVRCONN]
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.analyseErrorSegment(RemoteConnection.java:4176)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.receiveTSH(RemoteConnection.java:2969)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.initSess(RemoteConnection.java:1180)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.connect(RemoteConnection.java:838)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSessionFromNewConnection
(RemoteConnectionSpecification.java:409)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSession
(RemoteConnectionSpecification.java:305)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionPool.getSession(RemoteConnectionPool.java:146)
at com.ibm.mq.jmqi.remote.api.RemoteFAP.jmqiConnect(RemoteFAP.java:1868)
```

佇列管理程式錯誤日誌

AMQ9639: 遠端通道 'SYSTEM.DEF.SVRCONN' 未指定 CipherSpec。

解決方案

變更通道 `SYSTEM.DEF.SVRCONN` 可指定有效的 CipherSpec。

密碼不符

輸出

原因：

```
com.ibm.mq.jmqi.JmqiException: CC=2;RC=2397;AMQ9641: Remote CipherSpec error
for channel 'SYSTEM.DEF.SVRCONN' to host ''. [3=SYSTEM.DEF.SVRCONN]
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.analyseErrorSegment(RemoteConnection.java:4176)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.receiveTSH(RemoteConnection.java:2969)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.initSess(RemoteConnection.java:1180)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.connect(RemoteConnection.java:838)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSessionFromNewConnection
(RemoteConnectionSpecification.java:409)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSession
(RemoteConnectionSpecification.java:305)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionPool.getSession(RemoteConnectionPool.java:146)
at com.ibm.mq.jmqi.remote.api.RemoteFAP.jmqiConnect(RemoteFAP.java:1868)
```

佇列管理程式錯誤日誌

AMQ9631: 在 TLS 信號交換期間協議的 CipherSpec 不符合通道 'SYSTEM.DEF.SVRCONN' 所需的 CipherSpec。

解決方案

請變更伺服器連線通道或用戶端的「密碼組合」的 SSLCIPH 定義，使兩端具有相符的 CipherSuite 或 CipherSpec 配對。

遺漏用戶端個人憑證

輸出

原因：

```
com.ibm.mq.jmqi.JmqiException: CC=2;RC=2059;AMQ9503: Channel negotiation failed. [3=SYSTEM.DEF.SVRCONN]
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.analyseErrorSegment(RemoteConnection.java:4176)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.receiveTSH(RemoteConnection.java:2969)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.initSess(RemoteConnection.java:1180)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.connect(RemoteConnection.java:838)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSessionFromNewConnection
(RemoteConnectionSpecification.java:409)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSession
(RemoteConnectionSpecification.java:305)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionPool.getSession(RemoteConnectionPool.java:146)
at com.ibm.mq.jmqi.remote.api.RemoteFAP.jmqiConnect(RemoteFAP.java:1868)
```

佇列管理程式錯誤日誌

AMQ9637: 通道缺少憑證。

解決方案

請確定佇列管理程式的金鑰資料庫包含來自用戶端信任儲存庫的已簽署個人憑證。

遺漏伺服器個人憑證

輸出

原因：

```
com.ibm.mq.jmqi.JmqiException: CC=2;RC=2397;AMQ9771: SSL handshake failed.
[1=javax.net.ssl.SSLHandshakeException[Remote host closed connection during handshake],
3=localhost/127.0.0.1:1418 (localhost),4=SSLSocket.startHandshake,5=default]
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection.protocolConnect(RemoteTCPConnection.java:1173)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.connect(RemoteConnection.java:835)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSessionFromNewConnection
(RemoteConnectionSpecification.java:409)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSession
(RemoteConnectionSpecification.java:305)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionPool.getSession(RemoteConnectionPool.java:146)
at com.ibm.mq.jmqi.remote.api.RemoteFAP.jmqiConnect(RemoteFAP.java:1868)
... 12 more
```

原因：

```
javax.net.ssl.SSLHandshakeException: Remote host closed connection during handshake
at com.ibm.jsse2.qc.a(qc.java:158)
at com.ibm.jsse2.qc.h(qc.java:185)
at com.ibm.jsse2.qc.a(qc.java:566)
at com.ibm.jsse2.qc.startHandshake(qc.java:120)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection$6.run(RemoteTCPConnection.java:1142)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection$6.run(RemoteTCPConnection.java:1134)
at java.security.AccessController.doPrivileged(AccessController.java:229)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection.protocolConnect(RemoteTCPConnection.java:1134)
... 17 more
```

原因：

```
java.io.EOFException: SSL peer shut down incorrectly
at com.ibm.jsse2.a.a(a.java:19)
at com.ibm.jsse2.qc.a(qc.java:207)
```

佇列管理程式錯誤日誌

AMQ9637: 通道缺少憑證。

解決方案

請確定佇列管理程式的金鑰資料庫包含來自用戶端信任儲存庫的已簽署個人憑證。

用戶端上遺漏伺服器簽章者

輸出

原因：

```
com.ibm.mq.jmqi.JmqiException: CC=2;RC=2397;AMQ9771: SSL handshake failed.
[1=javax.net.ssl.SSLHandshakeException[com.ibm.jsse2.util.j:
PKIX path validation failed: java.security.cert.CertPathValidatorException:
The certificate issued by CN=JohnDoe, O=COMPANY, L=YOURSITE, C=XX is not trusted; internal cause is:
java.security.cert.CertPathValidatorException: Signature does not match.],3=localhost/127.0.0.1:1418
(localhost),4=SSLSocket.startHandshake,5=default]
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection.protocolConnect(RemoteTCPConnection.java:1173)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.connect(RemoteConnection.java:835)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSessionFromNewConnection
(RemoteConnectionSpecification.java:409)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSession
(RemoteConnectionSpecification.java:305)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionPool.getSession(RemoteConnectionPool.java:146)
at com.ibm.mq.jmqi.remote.api.RemoteFAP.jmqiConnect(RemoteFAP.java:1868)
...
```

原因:

```
javax.net.ssl.SSLHandshakeException: com.ibm.jsse2.util.j: PKIX path validation failed:
java.security.cert.CertPathValidatorException:
The certificate issued by CN=JohnDoe, O=COMPANY, L=YOURSITE, C=XX is not trusted;
internal cause is: java.security.cert.CertPathValidatorException: Signature does not match.
...
```

原因:

```
com.ibm.jsse2.util.j: PKIX path validation failed: java.security.cert.CertPathValidatorException:
The certificate issued by CN=JohnDoe, O=COMPANY, L=YOURSITE, C=XX is not trusted;
internal cause is: java.security.cert.CertPathValidatorException: Signature does not match.
at com.ibm.jsse2.util.h.a(h.java:99)
at com.ibm.jsse2.util.h.b(h.java:27)
at com.ibm.jsse2.util.g.a(g.java:14)
at com.ibm.jsse2.yc.a(yc.java:68)
at com.ibm.jsse2.yc.a(yc.java:17)
at com.ibm.jsse2.yc.checkServerTrusted(yc.java:154)
at com.ibm.jsse2.bb.a(bb.java:246)
... 28 more
```

原因:

```
java.security.cert.CertPathValidatorException:
The certificate issued by CN=JohnDoe, O=COMPANY, L=YOURSITE, C=XX is not trusted;
internal cause is: java.security.cert.CertPathValidatorException: Signature does not match.
at com.ibm.security.cert.BasicChecker.(BasicChecker.java:111)
at com.ibm.security.cert.PKIXCertPathValidatorImpl.engineValidate(PKIXCertPathValidatorImpl.java:174)
at java.security.cert.CertPathValidator.validate(CertPathValidator.java:265)
at com.ibm.jsse2.util.h.a(h.java:13)
... 34 more
```

原因:

```
java.security.cert.CertPathValidatorException: Signature does not match.
at com.ibm.security.cert.CertPathUtil.findIssuer(CertPathUtil.java:297)
at com.ibm.security.cert.BasicChecker.(BasicChecker.java:108)
```

佇列管理程式錯誤日誌

AMQ9665: 通道 '????' 的遠端系統已關閉 SSL 連線。

解決方案

將用來簽署佇列管理程式個人憑證的憑證新增至用戶端的信任儲存庫。

伺服器上遺漏用戶端簽章者

輸出

原因:

```
com.ibm.mq.jmqi.JmqiException: CC=2;RC=2397;AMQ9771: SSL handshake failed.
[1=java.net.SocketException[Software caused connection abort: socket write error],
3=localhost/127.0.0.1:1418 (localhost),4=SSLSocket.startHandshake,5=default]
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection.protocolConnect(RemoteTCPConnection.java:1173)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.connect(RemoteConnection.java:835)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSessionFromNewConnection
(RemoteConnectionSpecification.java:409)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSession
(RemoteConnectionSpecification.java:305)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionPool.getSession(RemoteConnectionPool.java:146)
at com.ibm.mq.jmqi.remote.api.RemoteFAP.jmqiConnect(RemoteFAP.java:1868)
... 12 more
```

原因:

```
java.net.SocketException: Software caused connection abort: socket write error
at java.net.SocketOutputStream.socketWrite(SocketOutputStream.java:120)
at java.net.SocketOutputStream.write(SocketOutputStream.java:164)
at com.ibm.jsse2.c.a(c.java:57)
at com.ibm.jsse2.c.a(c.java:34)
at com.ibm.jsse2.qc.b(qc.java:527)
at com.ibm.jsse2.qc.a(qc.java:635)
at com.ibm.jsse2.qc.a(qc.java:743)
at com.ibm.jsse2.ab.a(ab.java:550)
at com.ibm.jsse2.bb.b(bb.java:194)
at com.ibm.jsse2.bb.a(bb.java:162)
at com.ibm.jsse2.bb.a(bb.java:7)
at com.ibm.jsse2.ab.r(ab.java:529)
at com.ibm.jsse2.ab.a(ab.java:332)
at com.ibm.jsse2.qc.a(qc.java:435)
at com.ibm.jsse2.qc.h(qc.java:185)
at com.ibm.jsse2.qc.a(qc.java:566)
at com.ibm.jsse2.qc.startHandshake(qc.java:120)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection$6.run(RemoteTCPConnection.java:1142)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection$6.run(RemoteTCPConnection.java:1134)
at java.security.AccessController.doPrivileged(AccessController.java:229)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection.protocolConnect(RemoteTCPConnection.java:1134)
```

佇列管理程式錯誤日誌

AMQ9633: 通道 '????' 的 SSL 憑證不正確。

解決方案

將用來簽署用戶端個人憑證的憑證新增至佇列管理程式的金鑰資料庫。

伺服器上設定的 SSLPEER 不符合憑證

輸出

原因:

```
com.ibm.mq.jmqi.JmqiException: CC=2;RC=2397;AMQ9643: Remote SSL peer name error for channel
'SYSTEM.DEF.SVRCONN' on host ''. [3=SYSTEM.DEF.SVRCONN]
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.analyseErrorSegment(RemoteConnection.java:4176)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.receiveTSH(RemoteConnection.java:2969)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.initSess(RemoteConnection.java:1180)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.connect(RemoteConnection.java:838)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSessionFromNewConnection
(RemoteConnectionSpecification.java:409)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSession
(RemoteConnectionSpecification.java:305)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionPool.getSession(RemoteConnectionPool.java:146)
at com.ibm.mq.jmqi.remote.api.RemoteFAP.jmqiConnect(RemoteFAP.java:1868)
```

佇列管理程式錯誤日誌

AMQ9636: SSL 識別名稱不合同層級名稱，通道 'SYSTEM.DEF.SVRCONN'。

解決方案

請確定伺服器連線通道上所設定的 SSLPEER 值符合憑證的識別名稱。

用戶端上設定的 SSLPEER 不符合憑證

輸出

原因:

```
com.ibm.mq.jmqi.JmqiException: CC=2;RC=2398;AMQ9636: SSL distinguished name does not match peer name,
channel '?'. [CN=JohnDoe, O=COMPANY, L=YOURSITE, C=XX]
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection.protocolConnect(RemoteTCPConnection.java:1215)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.connect(RemoteConnection.java:835)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSessionFromNewConnection
(RemoteConnectionSpecification.java:409)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSession
(RemoteConnectionSpecification.java:305)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionPool.getSession(RemoteConnectionPool.java:146)
at com.ibm.mq.jmqi.remote.api.RemoteFAP.jmqiConnect(RemoteFAP.java:1868)
```

佇列管理程式錯誤日誌

AMQ9208: 從主機 *host-name (address)* 接收時發生錯誤。

解決方案

請確定用戶端中設定的 SSLPEER 值符合憑證的識別名稱。

在用戶端上使用已啟用 FIPS 的非 FIPS 密碼

輸出

```
Check the queue manager is started and if running in client mode, check there is a listener running.
Please see the linked exception for more information.
at com.ibm.msg.client.wmq.common.internal.Reason.reasonToException(Reason.java:578)
at com.ibm.msg.client.wmq.common.internal.Reason.createException(Reason.java:214)
at com.ibm.msg.client.wmq.internal.WMQConnection.getConnectOptions(WMQConnection.java:1423)
at com.ibm.msg.client.wmq.internal.WMQConnection.(WMQConnection.java:339)
at com.ibm.msg.client.wmq.factories.WMQConnectionFactory.createV7ProviderConnection
(WMQConnectionFactory.java:6865)
at com.ibm.msg.client.wmq.factories.WMQConnectionFactory.createProviderConnection
(WMQConnectionFactory.java:6221)
at com.ibm.msg.client.jms.admin.JmsConnectionFactoryImpl._createConnection
(JmsConnectionFactoryImpl.java:285)
at com.ibm.msg.client.jms.admin.JmsConnectionFactoryImpl.createConnection
(JmsConnectionFactoryImpl.java:233)
at com.ibm.mq.jms.MQConnectionFactory.createCommonConnection(MQConnectionFactory.java:6016)
at com.ibm.mq.jms.MQConnectionFactory.createConnection(MQConnectionFactory.java:6041)
at tests.SimpleSSLConn.runTest(SimpleSSLConn.java:46)
at tests.SimpleSSLConn.main(SimpleSSLConn.java:26)
```

原因:

```
com.ibm.mq.MQException: JMSCMQ0001: IBM MQ call failed with compcode '2' ('MQCC_FAILED')
reason '2400' ('MQRC_UNSUPPORTED_CIPHER_SUITE').
at com.ibm.msg.client.wmq.common.internal.Reason.createException(Reason.java:202)
```

佇列管理程式錯誤日誌

不適用。

解決方案

使用已啟用 FIPS 的密碼，或在用戶端上停用 FIPS。

在佇列管理程式上使用已啟用 FIPS 的非 FIPS 密碼

輸出

原因:

```
com.ibm.mq.jmqi.JmqiException: CC=2;RC=2397;AMQ9771: SSL handshake failed.
[1=javax.net.ssl.SSLHandshakeException[Received fatal alert: handshake_failure],
3=localhost/127.0.0.1:1418 (localhost),4=SSLSocket.startHandshake,5=default]
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection.protocolConnect(RemoteTCPConnection.java:1173)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.connect(RemoteConnection.java:835)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSessionFromNewConnection
(RemoteConnectionSpecification.java:409)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSession
(RemoteConnectionSpecification.java:305)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionPool.getSession(RemoteConnectionPool.java:146)
at com.ibm.mq.jmqi.remote.api.RemoteFAP.jmqiConnect(RemoteFAP.java:1868)
... 12 more
```

原因:

```
javax.net.ssl.SSLHandshakeException: Received fatal alert: handshake_failure
at com.ibm.jsse2.j.a(j.java:13)
at com.ibm.jsse2.j.a(j.java:18)
at com.ibm.jsse2.qc.b(qc.java:601)
at com.ibm.jsse2.qc.a(qc.java:100)
at com.ibm.jsse2.qc.h(qc.java:185)
at com.ibm.jsse2.qc.a(qc.java:566)
at com.ibm.jsse2.qc.startHandshake(qc.java:120)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection$6.run(RemoteTCPConnection.java:1142)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection$6.run(RemoteTCPConnection.java:1134)
at java.security.AccessController.doPrivileged(AccessController.java:229)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection.protocolConnect(RemoteTCPConnection.java:1134)
```

佇列管理程式錯誤日誌

AMQ9616: 伺服器上未啟用提議的 CipherSpec。

解決方案

使用已啟用 FIPS 的密碼，或在佇列管理程式上停用 FIPS。

使用 IBM JRE 找不到用戶端金鑰儲存庫

輸出

原因:

```
com.ibm.mq.jmqi.JmqiException: CC=2;RC=2059;AMQ9204: Connection to host 'localhost(1418)' rejected.
[1=com.ibm.mq.jmqi.JmqiException[CC=2;RC=2059;AMQ9503: Channel negotiation failed.
[3=SYSTEM.DEF.SVRCONN]], 3=localhost(1418), 5=RemoteConnection.analyseErrorSegment]
at com.ibm.mq.jmqi.remote.api.RemoteFAP.jmqiConnect(RemoteFAP.java:2450)
at com.ibm.mq.jmqi.remote.api.RemoteFAP.jmqiConnect(RemoteFAP.java:1396)
at com.ibm.mq.esj.jmqi.InterceptedJmqiImpl.jmqiConnect(InterceptedJmqiImpl.java:376)
at com.ibm.mq.esj.jmqi.ESEJMQI.jmqiConnect(ESEJMQI.java:561)
at com.ibm.mq.msg.client.wmq.internal.WMQConnection.(WMQConnection.java:342)
... 8 more
```

原因:

```
com.ibm.mq.jmqi.JmqiException: CC=2;RC=2059;AMQ9503: Channel negotiation failed. [3=SYSTEM.DEF.SVRCONN]
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.analyseErrorSegment(RemoteConnection.java:4176)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.receiveTSH(RemoteConnection.java:2969)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.initSess(RemoteConnection.java:1180)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.connect(RemoteConnection.java:838)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSessionFromNewConnection
(RemoteConnectionSpecification.java:409)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSession
(RemoteConnectionSpecification.java:305)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionPool.getSession(RemoteConnectionPool.java:146)
at com.ibm.mq.jmqi.remote.api.RemoteFAP.jmqiConnect(RemoteFAP.java:1868)
```

佇列管理程式錯誤日誌

AMQ9637: 通道缺少憑證。

解決方案

請確定 JVM 內容 `javax.net.ssl.keyStore` 指定有效金鑰儲存庫的位置。

使用 Oracle JRE 找不到用戶端金鑰儲存庫

輸出

原因:

```
java.security.PrivilegedActionException: java.io.FileNotFoundException:
C:\filepath\wrongkey.jks (The system cannot find the file specified)
at java.security.AccessController.doPrivileged(Native Method)
at sun.security.ssl.SSLContextImpl$DefaultSSLContext.getDefaultKeyManager(Unknown Source)
at sun.security.ssl.SSLContextImpl$DefaultSSLContext.(Unknown Source)
at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
at sun.reflect.NativeConstructorAccessorImpl.newInstance(Unknown Source)
at sun.reflect.DelegatingConstructorAccessorImpl.newInstance(Unknown Source)
at java.lang.reflect.Constructor.newInstance(Unknown Source)
at java.lang.Class.newInstance0(Unknown Source)
at java.lang.Class.newInstance(Unknown Source)
... 28 more
```

原因:

```
java.io.FileNotFoundException: C:\filepath\wrongkey.jks (The system cannot find the file specified)
at java.io.FileInputStream.open(Native Method)
at java.io.FileInputStream.(Unknown Source)
at java.io.FileInputStream.(Unknown Source)
at sun.security.ssl.SSLContextImpl$DefaultSSLContext$2.run(Unknown Source)
at sun.security.ssl.SSLContextImpl$DefaultSSLContext$2.run(Unknown Source)
```

佇列管理程式錯誤日誌

AMQ9637: 通道缺少憑證。

解決方案

請確定 JVM 內容 `javax.net.ssl.keyStore` 指定有效金鑰儲存庫的位置。

金鑰儲存庫密碼錯誤- IBM JRE

輸出

原因:

```
com.ibm.mq.jmqi.JmqiException: CC=2;RC=2059;AMQ9503: Channel negotiation failed. [3=SYSTEM.DEF.SVRCONN]
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.analyseErrorSegment(RemoteConnection.java:4176)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.receiveTSH(RemoteConnection.java:2969)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.initSess(RemoteConnection.java:1180)
at com.ibm.mq.jmqi.remote.impl.RemoteConnection.connect(RemoteConnection.java:838)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSessionFromNewConnection
(RemoteConnectionSpecification.java:409)
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionSpecification.getSession
(RemoteConnectionSpecification.java:305)
```

```
at com.ibm.mq.jmqi.remote.impl.RemoteConnectionPool.getSession(RemoteConnectionPool.java:146)
at com.ibm.mq.jmqi.remote.api.RemoteFAP.jmqiConnect(RemoteFAP.java:1868)
```

佇列管理程式錯誤日誌

AMQ9637: 通道缺少憑證。

解決方案

請確定 JVM 內容 `javax.net.ssl.keyStorePassword` 的值指定 `javax.net.ssl.keyStore` 所指定金鑰儲存庫的密碼。

信任儲存庫密碼錯誤- IBM JRE

輸出

原因:

```
javax.net.ssl.SSLHandshakeException: java.security.cert.CertificateException:
No X509TrustManager implementation available
at com.ibm.jsse2.j.a(j.java:13)
at com.ibm.jsse2.qc.a(qc.java:204)
at com.ibm.jsse2.ab.a(ab.java:342)
at com.ibm.jsse2.ab.a(ab.java:222)
at com.ibm.jsse2.bb.a(bb.java:157)
at com.ibm.jsse2.bb.a(bb.java:492)
at com.ibm.jsse2.ab.r(ab.java:529)
at com.ibm.jsse2.ab.a(ab.java:332)
at com.ibm.jsse2.qc.a(qc.java:435)
at com.ibm.jsse2.qc.h(qc.java:185)
at com.ibm.jsse2.qc.a(qc.java:566)
at com.ibm.jsse2.qc.startHandshake(qc.java:120)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection$6.run(RemoteTCPConnection.java:1142)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection$6.run(RemoteTCPConnection.java:1134)
at java.security.AccessController.doPrivileged(AccessController.java:229)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection.protocolConnect(RemoteTCPConnection.java:1134)
... 17 more
```

原因:

```
java.security.cert.CertificateException: No X509TrustManager implementation available
at com.ibm.jsse2.xc.checkServerTrusted(xc.java:2)
at com.ibm.jsse2.bb.a(bb.java:246)
```

佇列管理程式錯誤日誌

AMQ9665: 通道 '????' 遠端已關閉 SSL 連線。

解決方案

請確定 JVM 內容 `javax.net.ssl.trustStorePassword` 的值指定 `javax.net.ssl.trustStore` 所指定金鑰儲存庫的密碼。

找不到或無法開啟佇列管理程式金鑰資料庫

輸出

原因:

```
javax.net.ssl.SSLHandshakeException: Remote host closed connection during handshake
at com.ibm.jsse2.qc.a(qc.java:158)
at com.ibm.jsse2.qc.h(qc.java:185)
at com.ibm.jsse2.qc.a(qc.java:566)
at com.ibm.jsse2.qc.startHandshake(qc.java:120)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection$6.run(RemoteTCPConnection.java:1142)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection$6.run(RemoteTCPConnection.java:1134)
at java.security.AccessController.doPrivileged(AccessController.java:229)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection.protocolConnect(RemoteTCPConnection.java:1134)
... 17 more
```

原因:

```
java.io.EOFException: SSL peer shut down incorrectly
at com.ibm.jsse2.a.a(a.java:19)
at com.ibm.jsse2.qc.a(qc.java:207)
```

佇列管理程式錯誤日誌

AMQ9657: 無法開啟金鑰儲存庫 (通道 '????')。

解決方案

請確定您指定的金鑰儲存庫存在，且其許可權可讓所涉及的 IBM MQ 程序從中讀取。

找不到或無法使用佇列管理程式金鑰資料庫密碼隱藏檔

輸出

原因:

```
javax.net.ssl.SSLHandshakeException: Remote host closed connection during handshake
at com.ibm.jsse2.qc.a(qc.java:158)
at com.ibm.jsse2.qc.h(qc.java:185)
at com.ibm.jsse2.qc.a(qc.java:566)
at com.ibm.jsse2.qc.startHandshake(qc.java:120)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection$6.run(RemoteTCPConnection.java:1142)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection$6.run(RemoteTCPConnection.java:1134)
at java.security.AccessController.doPrivileged(AccessController.java:229)
at com.ibm.mq.jmqi.remote.impl.RemoteTCPConnection.protocolConnect(RemoteTCPConnection.java:1134)
... 17 more
```

原因:

```
ava.io.EOFException: SSL peer shut down incorrectly
at com.ibm.jsse2.a.a(a.java:19)
at com.ibm.jsse2.qc.a(qc.java:207)
```

佇列管理程式錯誤日誌

AMQ9660: SSL 金鑰儲存庫: 密碼隱藏檔缺少或無法使用。

解決方案

請確定密碼隱藏檔已與相同目錄中的金鑰資料庫檔相關聯，且執行 IBM MQ 的使用者 ID 具有這兩個檔案的讀取權。

RDQM 配置疑難排解

這些主題提供有助於對 RDQM 高可用性 (HA) 及災難回復 (DR) 配置進行疑難排解的資訊。

RDQM HA 架構

說明抄寫的資料佇列管理程式高可用性 (RDQM HA) 配置的基本架構，以協助疑難排解。

資源名稱

會針對每一個 RDQM 佇列管理程式建立各種資源，且這些資源具有以佇列管理程式的目錄名稱為基礎的名稱。此名稱可在檔案 `/var/mqm/mqs.ini` 中找到，在這裡稱為 *qm*。例如，對於名為 TMPQM1 的 RDQM HA 佇列管理程式，*qm* 會是 tmpqm1。

架構

RDQM 高可用性 (HA) 的架構涉及 DRBD (用於資料抄寫) 和 Pacemaker(用於管理 HA RDQM 佇列管理程式執行的位置)。

當您建立 RDQM HA 佇列管理程式時，會完成下列步驟：

1. 會建立 DRBD 資源來抄寫佇列管理程式的資料。
2. 建立並配置佇列管理程式，以使用 DRBD 資源作為其儲存體。
3. 會建立一組 Pacemaker 資源，以監視及管理佇列管理程式。

DRBD

每一個 RDQM HA 佇列管理程式都會為其產生名為 `/etc/drbd.d/qm.res` 的 DRBD 資源檔。例如，當建立名為 HAQM1 的 RDQM HA 佇列管理程式時，DRBD 資源檔為 `/etc/drbd.d/haqm1.res`。

`.res` 檔案中用於疑難排解的最重要資訊是此特定 DRBD 資源的裝置次要號碼。DRBD 記載的許多訊息都使用這個次要號碼。對於範例佇列管理程式 HAQM1，`.res` 檔案包含下列資訊：

```
device minor 100;
```

對於此佇列管理程式，您應該尋找類似下列範例的訊息：

```
Jul 31 00:17:24 mqhavm13 kernel: drbd haqm1/0 drbd100 mqhavm15.gamsworthwilliam.com:
drbd_sync_handshake:
```

字串 drbd100 的存在表示訊息與 HAQM1 相關。並非所有由 DRBD 記載的訊息都使用裝置次要號碼，有些會使用 DRBD 資源名稱，這與 RDQM HA 佇列管理程式的「目錄」名稱相同。例如：

```
Jul 31 00:17:22 mqhavm13 kernel: drbd haqm1 mqhavm15.gamsworthwilliam.com: Connection closed
```

Pacemaker

針對 RDQM HA 佇列管理程式產生一些 Pacemaker 資源：

qm

是代表 RDQM HA 佇列管理程式的主要資源。

p_rdqmx_qm

是內部資源。

p_fs_qm

是標準檔案系統資源，在 `/var/mqm/vols/qm` 上裝載佇列管理程式的磁區

ms_drbd_qm

是 RDQM 之 DRBD 資源的主要/從屬資源。

p_drbd_qm

是 RDQM 之 DRBD 資源的基本資源。

如果為 HA RDQM 配置浮動 IP 位址，則會配置其他資源：

p_ip_qm

RDQM HA 配置和錯誤範例

RDQM HA 配置範例，其中包含範例錯誤以及如何解決這些錯誤的相關資訊。

範例 RDQM HA 群組由三個節點組成：

- mqhavm13.gamsworthwilliam.com (稱為 vm13)。
- mqhavm14.gamsworthwilliam.com (稱為 vm14)。
- mqhavm15.gamsworthwilliam.com (稱為 vm15)。

已建立三個 RDQM HA 佇列管理程式：

- HAQM1 (建立於 vm13)
- HAQM2 (在 vm14 上建立)
- HAQM3 (建立於 vm15)

起始條件

下列清單中提供每一個節點上的起始條件：

vm13

```
[midtownjojo@mqhavm13 ~]$ rdqmstatus -m HAQM1
Node:                               mqhavm13.gamsworthwilliam.com
Queue manager status:              Running
CPU:                                0.00%
Memory:                             135MB
Queue manager file system:         51MB used, 1.0GB allocated [5%]
HA role:                             Primary
HA status:                           Normal
HA control:                           Enabled
HA current location:                 This node
HA preferred location:                 This node
HA floating IP interface:             None
HA floating IP address:               None
Node:                               mqhavm14.gamsworthwilliam.com
```

```

HA status: Normal
Node: mqhavam15.gamsworthwilliam.com
HA status: Normal
Command '/opt/mqm/bin/rdqmstatus' run with sudo.

[midtownjojo@mqhavam13 ~]$ rdqmstatus -m HAQM2
Node: mqhavam13.gamsworthwilliam.com
Queue manager status: Running elsewhere
HA role: Secondary
HA status: Normal
HA control: Enabled
HA current location: mqhavam14.gamsworthwilliam.com
HA preferred location: mqhavam14.gamsworthwilliam.com
HA floating IP interface: None
HA floating IP address: None

Node: mqhavam14.gamsworthwilliam.com
HA status: Normal

Node: mqhavam15.gamsworthwilliam.com
HA status: Normal
Command '/opt/mqm/bin/rdqmstatus' run with sudo.

[midtownjojo@mqhavam13 ~]$ rdqmstatus -m HAQM3
Node: mqhavam13.gamsworthwilliam.com
Queue manager status: Running elsewhere
HA role: Secondary
HA status: Normal
HA control: Enabled
HA current location: mqhavam15.gamsworthwilliam.com
HA preferred location: mqhavam15.gamsworthwilliam.com
HA floating IP interface: None
HA floating IP address: None

Node: mqhavam14.gamsworthwilliam.com
HA status: Normal

Node: mqhavam15.gamsworthwilliam.com
HA status: Normal
Command '/opt/mqm/bin/rdqmstatus' run with sudo.

```

vm14

```

[midtownjojo@mqhavam14 ~]$ rdqmstatus -m HAQM1
Node: mqhavam14.gamsworthwilliam.com
Queue manager status: Running elsewhere
HA role: Secondary
HA status: Normal
HA control: Enabled
HA current location: mqhavam13.gamsworthwilliam.com
HA preferred location: mqhavam13.gamsworthwilliam.com
HA floating IP interface: None
HA floating IP address: None

Node: mqhavam13.gamsworthwilliam.com
HA status: Normal

Node: mqhavam15.gamsworthwilliam.com
HA status: Normal
Command '/opt/mqm/bin/rdqmstatus' run with sudo.

[midtownjojo@mqhavam14 ~]$ rdqmstatus -m HAQM2
Node: mqhavam14.gamsworthwilliam.com
Queue manager status: Running
CPU: 0.00%
Memory: 135MB
Queue manager file system: 51MB used, 1.0GB allocated [5%]
HA role: Primary
HA status: Normal
HA control: Enabled
HA current location: This node
HA preferred location: This node
HA floating IP interface: None
HA floating IP address: None

Node: mqhavam13.gamsworthwilliam.com
HA status: Normal

Node: mqhavam15.gamsworthwilliam.com
HA status: Normal

```


Command '/opt/mqm/bin/rdqmstatus' run with sudo.

```
[midtwnjojo@mqhavm14 ~]$ rdqmstatus -m HAQM3
Node: mqhavam14.gamsworthwilliam.com
Queue manager status: Running elsewhere
HA role: Secondary
HA status: Normal
HA control: Enabled
HA current location: mqhavam15.gamsworthwilliam.com
HA preferred location: mqhavam15.gamsworthwilliam.com
HA floating IP interface: None
HA floating IP address: None

Node: mqhavam13.gamsworthwilliam.com
HA status: Normal

Node: mqhavam15.gamsworthwilliam.com
HA status: Normal
Command '/opt/mqm/bin/rdqmstatus' run with sudo.
```

vm15

```
[midtwnjojo@mqhavam15 ~]$ rdqmstatus -m HAQM1
Node: mqhavam15.gamsworthwilliam.com
Queue manager status: Running elsewhere
HA role: Secondary
HA status: Normal
HA control: Enabled
HA current location: mqhavam13.gamsworthwilliam.com
HA preferred location: mqhavam13.gamsworthwilliam.com
HA floating IP interface: None
HA floating IP address: None

Node: mqhavam13.gamsworthwilliam.com
HA status: Normal

Node: mqhavam14.gamsworthwilliam.com
HA status: Normal
Command '/opt/mqm/bin/rdqmstatus' run with sudo.

[midtwnjojo@mqhavam15 ~]$ rdqmstatus -m HAQM2
Node: mqhavam15.gamsworthwilliam.com
Queue manager status: Running elsewhere
HA role: Secondary
HA status: Normal
HA control: Enabled
HA current location: mqhavam14.gamsworthwilliam.com
HA preferred location: mqhavam14.gamsworthwilliam.com
HA floating IP interface: None
HA floating IP address: None

Node: mqhavam13.gamsworthwilliam.com
HA status: Normal

Node: mqhavam14.gamsworthwilliam.com
HA status: Normal
Command '/opt/mqm/bin/rdqmstatus' run with sudo.

[midtwnjojo@mqhavam15 ~]$ rdqmstatus -m HAQM3
Node: mqhavam15.gamsworthwilliam.com
Queue manager status: Running
CPU: 0.02%
Memory: 135MB
Queue manager file system: 51MB used, 1.0GB allocated [5%]
HA role: Primary
HA status: Normal
HA control: Enabled
HA current location: This node
HA preferred location: This node
HA floating IP interface: None
HA floating IP address: None

Node: mqhavam13.gamsworthwilliam.com
HA status: Normal

Node: mqhavam14.gamsworthwilliam.com
HA status: Normal
Command '/opt/mqm/bin/rdqmstatus' run with sudo.
```

DRBD 實務範例

RDQM HA 配置使用 DRBD 進行資料抄寫。下列實務範例說明 DRBD 的下列可能問題：

- 失去 DRBD 額定
- 失去單一 DRBD 連線
- 同步化停滯

DRBD 實務範例 1: 失去 DRBD 額定

如果執行 RDQM HA 佇列管理程式的節點失去對應於佇列管理程式之 DRBD 資源的 DRBD 額定，則 DRBD 會立即開始從 I/O 作業傳回錯誤，這會導致佇列管理程式開始產生 FDC 並最終停止。

如果其餘兩個節點具有 DRBD 資源的 DRBD 額定，則 Pacemaker 會選擇兩個節點之一來啟動佇列管理程式。因為從失去仲裁的時間開始，原始節點上沒有任何更新項目，所以在其他位置啟動佇列管理程式是安全的。

您可以監視 DRBD 額定遺失的兩種主要方式：

- 使用 **rdqmstatus** 指令。
- 監視 RDQM HA 佇列管理程式起始執行所在節點的 syslog。

rdqmstatus

如果您使用 **rdqmstatus** 指令，則當節點 vm13 失去 HAQM1 之 DRBD 資源的 DRBD 額定時，您可能看到類似下列範例的狀態：

```
[midtownjojo@mqhavm13 ~]$ rdqmstatus -m HAQM1
Node:                               mqhavm13.gamsworthwilliam.com
Queue manager status:               Running elsewhere
HA role:                             Secondary
HA status:                           Remote unavailable
HA control:                           Enabled
HA current location:                 mqhavm14.gamsworthwilliam.com
HA preferred location:                This node
HA floating IP interface:             None
HA floating IP address:               None

Node:                               mqhavm14.gamsworthwilliam.com
HA status:                           Remote unavailable
HA out of sync data:                 0KB

Node:                               mqhavm15.gamsworthwilliam.com
HA status:                           Remote unavailable
HA out of sync data:                 0KB
Command '/opt/mqm/bin/rdqmstatus' run with sudo.
```

請注意，HA status 已變更為 Remote unavailable，這表示與其他節點的兩個 DRBD 連線都已遺失。

在此情況下，其他兩個節點具有 DRBD 資源的 DRBD 額定，因此 RDQM 正在 mqhavm14.gamsworthwilliam.com 上的其他位置執行，如 HA current location 的值所示。

監視 syslog

如果您監視 syslog，當它失去資源的額定時，您會看到 DRBD 記載一則訊息：

```
Jul 30 09:38:36 mqhavm13 kernel: drbd haqm1/0 drbd100: quorum( yes -> no )
```

還原仲裁時，會記載類似訊息：

```
Jul 30 10:27:32 mqhavm13 kernel: drbd haqm1/0 drbd100: quorum( no -> yes )
```

DRBD 實務範例 2: 失去單一 DRBD 連線

如果執行 RDQM HA 佇列管理程式的節點中只有兩個 DRBD 連線之一遺失，則佇列管理程式不會移動。

從與第一個實務範例相同的起始條件開始，在僅封鎖其中一個 DRBD 抄寫鏈結之後，vm13 上 **rdqmstatus** 所報告的狀態類似於下列範例：

```
Node: mqhavam13.gamsworthwilliam.com
Queue manager status: Running
CPU: 0.01%
Memory: 133MB
Queue manager file system: 52MB used, 1.0GB allocated [5%]
HA role: Primary
HA status: Mixed
HA control: Enabled
HA current location: This node
HA preferred location: This node
HA floating IP interface: None
HA floating IP address: None

Node: mqhavam14.gamsworthwilliam.com

HA status: Remote unavailable
HA out of sync data: OKB

Node: mqhavam15.gamsworthwilliam.com
HA status: Normal
Command '/opt/mqm/bin/rdqmstatus' run with sudo.
```

DRBD 實務 3: 同步化停滯

部分 DRBD 版本發生同步化似乎停滯的問題，這使得 RDQM HA 佇列管理程式無法在與節點同步仍在進行時失效接手至節點。

其中一種方法是使用 `drbdadm status` 指令。正常運作時，會輸出類似下列範例的回應：

```
[midtownjojo@mqhavam13 ~]$ drbdadm status
haqm1 role:Primary
  disk:UpToDate
  mqhavam14.gamsworthwilliam.com role:Secondary
  peer-disk:UpToDate
  mqhavam15.gamsworthwilliam.com role:Secondary
  peer-disk:UpToDate

haqm2 role:Secondary
  disk:UpToDate
  mqhavam14.gamsworthwilliam.com role:Primary
  peer-disk:UpToDate
  mqhavam15.gamsworthwilliam.com role:Secondary
  peer-disk:UpToDate

haqm3 role:Secondary
  disk:UpToDate
  mqhavam14.gamsworthwilliam.com role:Secondary
  peer-disk:UpToDate
  mqhavam15.gamsworthwilliam.com role:Primary
  peer-disk:UpToDate
```

如果同步化停滯，回應類似於下列範例：

```
[midtownjojo@mqhavam13 ~]$ drbdadm status
haqm1 role:Primary
  disk:UpToDate
  mqhavam14.gamsworthwilliam.com role:Secondary
  peer-disk:UpToDate
  mqhavam15.gamsworthwilliam.com role:Secondary
  replication:SyncSource peer-disk:Inconsistent done:90.91

haqm2 role:Secondary
  disk:UpToDate
  mqhavam14.gamsworthwilliam.com role:Primary
  peer-disk:UpToDate
  mqhavam15.gamsworthwilliam.com role:Secondary
  peer-disk:UpToDate

haqm3 role:Secondary
  disk:UpToDate
  mqhavam14.gamsworthwilliam.com role:Secondary
  peer-disk:UpToDate
```

```
mqhavm15.gamsworthwilliam.com role:Primary
peer-disk:UpToDate
```

在此情況下，RDQM HA 佇列管理程式 HAQM1 無法移至 vm15，因為 vm15 上的磁碟是 Inconsistent。done 值是完成百分比。如果該值未增加，您可以嘗試中斷該抄本的連線，然後在 vm13: 上使用下列指令重新連接它 (以 root 身分執行)

```
drbdadm disconnect haqm1:mqhavm15.gamsworthwilliam.com
drbdadm connect haqm1:mqhavm15.gamsworthwilliam.com
```

如果兩個「次要」節點的抄寫停滯，您可以執行 **disconnect** 和 **connect** 指令而不指定節點，這會切斷這兩個連線：

```
drbdadm disconnect haqm1
drbdadm connect haqm1
```

Pacemaker 實務範例

RDQM HA 配置使用 Pacemaker 來判斷 RDQM HA 佇列管理程式的執行位置。下列實務範例說明涉及 Pacemaker 的下列可能問題：

- 未排定 Corosync 主要處理程序
- RDQM HA 佇列管理程式未在它應該執行的位置執行

Pacemaker 實務範例 1: Corosync 未排定主要處理程序

如果您在 syslog 中看到類似下列範例的訊息，則指出系統太忙，無法將 CPU 時間排定到主要 Corosync 處理程序，或者更常見的情況是系統是「虛擬機器」，而 Hypervisor 尚未對整個 VM 排定任何 CPU 時間。

```
corosync[10800]: [MAIN ] Corosync main process was not scheduled for 2787.0891 ms (threshold is 1320.0000 ms). Consider token timeout increase.
```

Pacemaker (和 Corosync) 和 DRBD 都有用來偵測仲裁流失的計時器，因此範例之類的訊息指出節點未執行太久，已從仲裁中捨棄。Corosync 逾時為 1.65 秒，而 1.32 秒的臨界值為 80%，因此當主要 Corosync 處理程序排程中的延遲達到逾時的 80% 時，會列印範例中顯示的訊息。在此範例中，處理程序未排程近三秒。任何導致這類問題的問題都必須解決。在類似狀況中可能有幫助的一件事是減少 VM 的需求，例如減少所需的 vCPUs 數目，因為這可讓 Hypervisor 更容易排定 VM。

Pacemaker 實務範例 2 :RDQM HA 佇列管理程式未在應該執行的位置執行

在此實務範例中協助疑難排解的主要工具是 **crm status** 指令。下列範例顯示當一切如預期運作時，配置的回應：

```
Stack: corosync
Current DC: mqhavm13.gamsworthwilliam.com (version 1.1.20.linbit-1+20190404+eab6a2092b71.e17.2-eab6a2092b) - partition with quorum
Last updated: Tue Jul 30 09:11:29 2019
Last change: Tue Jul 30 09:10:34 2019 by root via crm_attribute on mqhavm14.gamsworthwilliam.com
```

```
3 nodes configured
18 resources configured
```

```
Online: [ mqhavm13.gamsworthwilliam.com mqhavm14.gamsworthwilliam.com
mqhavm15.gamsworthwilliam.com ]
```

```
Full list of resources:
```

```
Master/Slave Set: ms_drbd_haqm1 [p_drbd_haqm1]
Masters: [ mqhavm13.gamsworthwilliam.com ]
Slaves: [ mqhavm14.gamsworthwilliam.com mqhavm15.gamsworthwilliam.com ]
p_fs_haqm1 (ocf::heartbeat:Filesystem): Started mqhavm13.gamsworthwilliam.com
p_rdqmx_haqm1 (ocf::ibm:rdqmx): Started mqhavm13.gamsworthwilliam.com
haqm1 (ocf::ibm:rdqm): Started mqhavm13.gamsworthwilliam.com
Master/Slave Set: ms_drbd_haqm2 [p_drbd_haqm2]
Masters: [ mqhavm14.gamsworthwilliam.com ]
Slaves: [ mqhavm13.gamsworthwilliam.com mqhavm15.gamsworthwilliam.com ]
p_fs_haqm2 (ocf::heartbeat:Filesystem): Started mqhavm14.gamsworthwilliam.com
```

```

p_rdqmx_haqm2 (ocf::ibm:rdqmx): Started mqhavm14.gamsworthwilliam.com
haqm2 (ocf::ibm:rdqm): Started mqhavm14.gamsworthwilliam.com
Master/Slave Set: ms_drbd_haqm3 [p_drbd_haqm3]
Masters: [ mqhavm15.gamsworthwilliam.com ]
Slaves: [ mqhavm13.gamsworthwilliam.com mqhavm14.gamsworthwilliam.com ]
p_fs_haqm3 (ocf::heartbeat:Filesystem): Started mqhavm15.gamsworthwilliam.com
p_rdqmx_haqm3 (ocf::ibm:rdqmx): Started mqhavm15.gamsworthwilliam.com
haqm3 (ocf::ibm:rdqm): Started mqhavm15.gamsworthwilliam.com

```

請注意下列幾點：

- 所有三個節點都會顯示為 Online。
- 每一個 RDQM HA 佇列管理程式都在其建立所在的節點上執行，例如， HAQM1 在 vm13 上執行，依此類推。

此實務範例的建構方式是防止 HAQM1 在 vm14 上執行，然後嘗試將 HAQM1 移至 vm14。HAQM1 無法在 vm14 上執行，因為 vm14 上的檔案 /var/mqm/mqs.ini 對於佇列管理程式 HAQM1 的「目錄」具有無效值。

在 vm13: 上執行下列指令，將 HAQM1 的偏好位置變更為 vm14

```
rdqmadm -m HAQM1 -n mqhavm14.gamsworthwilliam.com -p
```

此指令通常會導致 HAQM1 移至 vm14，但在此情況下檢查 vm13 上的狀態會傳回下列資訊：

```

[midtonjojo@mqhavm13 ~]$ rdqmstatus -m HAQM1
Node: mqhavm13.gamsworthwilliam.com
Queue manager status: Running
CPU: 0.15%
Memory: 133MB
Queue manager file system: 52MB used, 1.0GB allocated [5%]
HA role: Primary
HA status: Normal
HA control: Enabled
HA current location: This node
HA preferred location: mqhavm14.gamsworthwilliam.com
HA floating IP interface: None
HA floating IP address: None

Node: mqhavm14.gamsworthwilliam.com
HA status: Normal

Node: mqhavm15.gamsworthwilliam.com
HA status: Normal
Command '/opt/mqm/bin/rdqmstatus' run with sudo.

```

HAQM1 仍在 vm13 上執行，尚未依要求移至 vm14，且需要調查原因。檢查 Pacemaker 狀態會提供下列回應：

```

[midtownjojo@mqhavm13 ~]$ crm status
Stack: corosync
Current DC: mqhavm13.gamsworthwilliam.com (version 1.1.20.linbit-1+20190404+eab6a2092b71.e17.2-eab6a2092b) - partition with quorum
Last updated: Thu Aug 1 14:16:40 2019
Last change: Thu Aug 1 14:16:35 2019 by hacluster via crmd on mqhavm14.gamsworthwilliam.com

3 nodes configured
18 resources configured

Online: [ mqhavm13.gamsworthwilliam.com mqhavm14.gamsworthwilliam.com
mqhavm15.gamsworthwilliam.com ]

Full list of resources:

Master/Slave Set: ms_drbd_haqm1 [p_drbd_haqm1]
Masters: [ mqhavm13.gamsworthwilliam.com ]
Slaves: [ mqhavm14.gamsworthwilliam.com mqhavm15.gamsworthwilliam.com ]
p_fs_haqm1 (ocf::heartbeat:Filesystem): Started mqhavm13.gamsworthwilliam.com
p_rdqmx_haqm1 (ocf::ibm:rdqmx): Started mqhavm13.gamsworthwilliam.com
haqm1 (ocf::ibm:rdqm): Started mqhavm13.gamsworthwilliam.com
Master/Slave Set: ms_drbd_haqm2 [p_drbd_haqm2]
Masters: [ mqhavm14.gamsworthwilliam.com ]
Slaves: [ mqhavm13.gamsworthwilliam.com mqhavm15.gamsworthwilliam.com ]
p_fs_haqm2 (ocf::heartbeat:Filesystem): Started mqhavm14.gamsworthwilliam.com
p_rdqmx_haqm2 (ocf::ibm:rdqmx): Started mqhavm14.gamsworthwilliam.com

```

```
haqm2 (ocf::ibm:rdqm): Started mqhavam14.gamsworthwilliam.com
Master/Slave Set: ms_drbd_haqm3 [p_drbd_haqm3]
Masters: [ mqhavam15.gamsworthwilliam.com ]
Slaves: [ mqhavam13.gamsworthwilliam.com mqhavam14.gamsworthwilliam.com ]
p_fs_haqm3 (ocf::heartbeat:Filesystem): Started mqhavam15.gamsworthwilliam.com
p_rdqm_haqm3 (ocf::ibm:rdqm): Started mqhavam15.gamsworthwilliam.com
haqm3 (ocf::ibm:rdqm): Started mqhavam15.gamsworthwilliam.com
```

Failed Resource Actions:

```
* haqm1_monitor_0 on mqhavam14.gamsworthwilliam.com 'not installed' (5): call=372,
status=complete, exitreason='',
last-rc-change='Thu Aug 1 14:16:37 2019', queued=0ms, exec=17ms
```

記下出現的 Failed Resource Actions 區段。

動作名稱 `haqm1_monitor_0` 指出它是 RDQM HAQM1 的監視動作失敗，且在 `mqhavam14.gamsworthwilliam.com` 上失敗，因此它看起來像是 Pacemaker 嘗試執行我們預期的動作，並在 `vm14` 上啟動 HAQM1，但由於某些原因它無法執行。

您可以從 `last-rc-change` 的值查看 Pacemaker 嘗試執行此動作的時間

瞭解失敗

若要瞭解失敗，我們需要在失敗時查看 `vm14` 的 `syslog`：

```
Aug 1 14:16:37 mqhavam14 cmd[26377]: notice: Result of probe operation for haqm1 on
mqhavam14.gamsworthwilliam.com: 5 (not installed)
```

該項目顯示當 Pacemaker 嘗試檢查 `vm14` 上 `haqm1` 的狀態時發生錯誤，因為未配置 `haqm1`，這是因為在 `/var/mqm/mqs.ini` 中故意配置錯誤。

更正失敗

若要更正失敗，您必須更正基礎問題 (在此情況下，會還原 `vm14` 上 `/var/mqm/mqs.ini` 中 `haqm1` 的正確目錄值)。然後，您必須在適當的資源上使用指令 `crm resource cleanup` 來清除失敗的動作，在此情況下是資源 `haqm1`，因為該資源是失敗動作中提及的資源。例如：

```
[midtownjojo@mqhavam13 ~]$ crm resource cleanup haqm1
Cleaned up haqm1 on mqhavam15.gamsworthwilliam.com
Cleaned up haqm1 on mqhavam14.gamsworthwilliam.com
Cleaned up haqm1 on mqhavam13.gamsworthwilliam.com
```

然後再次檢查 Pacemaker 狀態：

```
[midtownjojo@mqhavam13 ~]$ crm status
Stack: corosync
Current DC: mqhavam13.gamsworthwilliam.com (version 1.1.20.linbit-1+20190404+eab6a2092b71.e17.2-
eab6a2092b) - partition with quorum
Last updated: Thu Aug 1 14:23:17 2019
Last change: Thu Aug 1 14:23:03 2019 by hacluster via crmd on mqhavam13.gamsworthwilliam.com
```

```
3 nodes configured
18 resources configured
```

```
Online: [ mqhavam13.gamsworthwilliam.com mqhavam14.gamsworthwilliam.com
mqhavam15.gamsworthwilliam.com ]
```

Full list of resources:

```
Master/Slave Set: ms_drbd_haqm1 [p_drbd_haqm1]
Masters: [ mqhavam14.gamsworthwilliam.com ]
Slaves: [ mqhavam13.gamsworthwilliam.com mqhavam15.gamsworthwilliam.com ]
p_fs_haqm1 (ocf::heartbeat:Filesystem): Started mqhavam14.gamsworthwilliam.com
p_rdqm_haqm1 (ocf::ibm:rdqm): Started mqhavam14.gamsworthwilliam.com
haqm1 (ocf::ibm:rdqm): Started mqhavam14.gamsworthwilliam.com
Master/Slave Set: ms_drbd_haqm2 [p_drbd_haqm2]
Masters: [ mqhavam14.gamsworthwilliam.com ]
Slaves: [ mqhavam13.gamsworthwilliam.com mqhavam15.gamsworthwilliam.com ]
p_fs_haqm2 (ocf::heartbeat:Filesystem): Started mqhavam14.gamsworthwilliam.com
p_rdqm_haqm2 (ocf::ibm:rdqm): Started mqhavam14.gamsworthwilliam.com
haqm2 (ocf::ibm:rdqm): Started mqhavam14.gamsworthwilliam.com
Master/Slave Set: ms_drbd_haqm3 [p_drbd_haqm3]
Masters: [ mqhavam15.gamsworthwilliam.com ]
```



```
Slaves: [ mqhavam13.gamsworthwilliam.com mqhavam14.gamsworthwilliam.com ]
p_fs_haqm3 (ocf::heartbeat:Filesystem): Started mqhavam15.gamsworthwilliam.com
p_rdqmx_haqm3 (ocf::ibm:rdqmx): Started mqhavam15.gamsworthwilliam.com
haqm3 (ocf::ibm:rdqm): Started mqhavam15.gamsworthwilliam.com
```

失敗的動作已消失， HAQM1 現在如預期在 vm14 上執行。 下列範例顯示 RDQM 狀態：

```
[midtownjojo@mqhavam13 ~]$ rdqmstatus -m HAQM1
Node: mqhavam13.gamsworthwilliam.com
Queue manager status: Running elsewhere
HA role: Secondary
HA status: Normal
HA control: Enabled
HA current location: mqhavam14.gamsworthwilliam.com
HA preferred location: mqhavam14.gamsworthwilliam.com
HA floating IP interface: None
HA floating IP address: None

Node: mqhavam14.gamsworthwilliam.com
HA status: Normal

Node: mqhavam15.gamsworthwilliam.com
HA status: Normal
Command '/opt/mqm/bin/rdqmstatus' run with sudo.
```

Windows

Linux

AIX

MQ Telemetry 疑難排解

尋找疑難排解作業，以協助您解決執行 MQ Telemetry 應用程式的問題。

相關資訊

[MQ Telemetry](#)

Windows

Linux

AIX

Telemetry 日誌、錯誤日誌和配置檔的位置

尋找 MQ Telemetry 使用的日誌、錯誤日誌及配置檔。

註：這些範例是針對 Windows 系統進行編碼。變更語法以在 AIX 或 Linux 系統上執行範例。

伺服器端日誌

遙測 (MQXR) 服務會將 FDC 檔案寫入 IBM MQ 錯誤目錄：

```
WMQ data directory\errors\AMQ nnn.n.FDC
```

FDC 檔案的格式為 MQXRn.FDC。

它還會寫入遙測 (MQXR) 服務的日誌。日誌路徑是：

```
WMQ data directory\Qmgrs\qMgrName\errors\mqxr.log
```

日誌檔的格式為 mqxr_n.log。

IBM MQ Explorer 所建立的 IBM MQ 遙測範例配置會使用指令 **runMQXRService** 來啟動遙測 (MQXR) 服務，該指令位於 *WMQ Telemetry installation directory\bin* 中。此指令會寫入：

```
WMQ data directory\Qmgrs\qMgrName\mqxr.stdout
WMQ data directory\Qmgrs\qMgrName\mqxr.stderr
```

伺服器端配置檔

遙測通道及遙測 (MQXR) 服務

限制：遙測通道配置檔的格式、位置、內容和解譯在未來的版本中可能會發生變化。您必須使用「IBM MQ 探險家」或 MQSC 指令，來配置遙測通道。

IBM MQ Explorer 會將遙測配置儲存在 Windows 系統上的 `mqxr_win.properties` 檔案中，以及 AIX 或 Linux 系統上的 `mqxr_unix.properties` 檔案中。這些內容檔會儲存在遙測配置目錄中：

```
WMQ data directory\Qmgrs\qMgrName\mqxr
```

圖 18: Windows 上的遙測配置目錄

```
/var/mqm/qmgrs/qMgrName/mqxr
```

圖 19: AIX 或 Linux 上的遙測配置目錄

JVM

在檔案 `java.properties` 中設定作為引數傳遞至遙測 (MQXR) 服務的 Java 內容。該檔案中的內容會直接傳遞至執行遙測 (MQXR) 服務的 JVM。它們在 Java 指令行上作為其他 JVM 內容傳遞。指令行上所設定的內容，優先於從 `java.properties` 檔案新增至指令行的內容。

在遙測配置的相同資料夾中尋找 `java.properties` 檔案。請參閱第 203 頁的圖 18 和第 203 頁的圖 19。

以分隔線形式指定每個內容，修改 `java.properties`。將每一個內容完全按照您將內容傳遞至 JVM 作為引數的方式來格式化。例如：

```
-Xmx1024m  
-Xms1024m
```

JAAS

JAAS 配置檔在 [遙測通道 JAAS 配置](#) 中說明，其中包含 MQ Telemetry 隨附的範例 JAAS 配置檔 `JAAS.config`。

如果您配置 JAAS，則您必定會要撰寫類別來鑑別使用者，以取代標準的 JAAS 鑑別程序。

若要將 Login 類別併入遙測 (MQXR) 服務類別路徑所使用的類別路徑中，請提供 IBM MQ `service.env` 配置檔。

在 `service.env` 中設定您 JAAS LoginModule 的類別路徑。不能在 `service.env` 中使用變數 `%classpath%`。`service.env` 中的類別路徑會新增至遙測 (MQXR) 服務定義中已設定的類別路徑。

透過將 `echo set classpath` 新增至 `runMQXRService.bat`，可顯示遙測 (MQXR) 服務所使用的類別路徑。會將輸出傳送至 `mqxr.stdout`。

`service.env` 檔的預設位置是：

```
WMQ data directory\service.env
```

利用下列位置中每個佇列管理程式的 `service.env` 檔置換這些設定：

```
WMQ data directory\Qmgrs\qMgrName\service.env
```

```
CLASSPATH= WMQ Installation Directory\mqxr\samples\samples
```

註： `service.env` 不得包含任何變數。請替換 *WMQ Installation Directory* 的實際值。

圖 20: `service.env` for Windows 範例

追蹤

請參閱第 204 頁的『[追蹤遙測 \(MQXR\) 服務](#)』。用於配置追蹤的參數儲存在兩個檔案中：

```
WMQ data directory\Qmgrs\qMgrName\mqxr\trace.config  
WMQ data directory\Qmgrs\qMgrName\mqxr\mqxrtraceOn.properties
```

且有一個對應的檔案：

```
WMQ data directory\Qmgrs\qMgrName\mqxr\mqxrtrace0ff.properties
```

用戶端日誌檔和用戶端配置檔

如需最新資訊及下載項目，請參閱下列資源：

- Eclipse Paho 專案和 MQTT.org 具有最新遙測用戶端的免費下載，以及一系列程式設計語言的範例。使用這些網站來協助您開發程式範例以發佈和訂閱 IBM MQ Telemetry Transport 及新增安全特性。
- IBM Messaging Telemetry Clients SupportPac 已無法再下載。如果有先前下載的副本，則它具有下列內容：

- IBM Messaging Telemetry Clients SupportPac 的 MA9B 版本包括完整的範例應用程式 (mqttv3app.jar) 及關聯用戶端程式庫 (mqttv3.jar)。在一下目錄提供：

- ma9b/SDK/clients/java/org.eclipse.paho.sample.mqttv3app.jar
- ma9b/SDK/clients/java/org.eclipse.paho.client.mqttv3.jar

- 在此 SupportPac 的 MA9C 版本中，已移除 /SDK/ 目錄及內容：

- 只提供了範例應用程式 (mqttv3app.jar) 的來源。它位於下列目錄中：

```
ma9c/clients/java/samples/org/eclipse/paho/sample/mqttv3app/*.java
```

- 仍提供已編譯的用戶端程式庫。它位於下列目錄中：

```
ma9c/clients/java/org.eclipse.paho.client.mqttv3-1.0.2.jar
```

Windows

Linux

AIX

追蹤遙測 (MQXR) 服務

提供 IBM MQ 遙測 (MQXR) 服務所提供的追蹤機能，可協助「IBM 支援中心」診斷與服務相關的客戶問題。

關於這項作業

有兩種方法可以控制 IBM MQ 遙測服務的追蹤：

- 使用 **strmqtrc** 和 **endmqtrc** 指令來啟動和停止追蹤。使用 **strmqtrc** 指令來啟用追蹤，會針對執行 IBM MQ 遙測服務的整個佇列管理程式產生追蹤資訊。這包括 IBM MQ 遙測服務本身，以及服務用來與其他佇列管理程式元件通訊的基礎「Java 訊息佇列作業介面 (JMQUI)」。
- 執行 **controlMQXRChannel** 指令。請注意，使用 **controlMQXRChannel** 指令開啟追蹤只會追蹤 IBM MQ 遙測服務。

如果您不確定要使用哪個選項，請聯絡「IBM 支援中心」代表，他們將能夠建議您針對您所看到的問題收集追蹤的最佳方式。

程序

1. 方法一

- a) 啟動命令提示字元，並導覽至下列目錄：

```
MQ_INSTALLATION_PATH\bin
```

- b) 執行 **strmqtrc** 指令以啟用追蹤。

```
strmqtrc -m qmgr_name
```

其中 *qmgr_name* 是 IBM MQ MQXR 服務執行所在的佇列管理程式名稱。

- c) 重新產生問題。

- d) 執行下列指令來停止追蹤：

```
endmqtrc -m qmgr_name
```

2. 方法二。

- a) 啟動命令提示字元，並導覽至下列目錄：

`MQ_INSTALLATION_PATH\mqxr\bin`

- b) 執行下列指令以啟用追蹤：

• **Windows**

```
controlMQXRChannel -qmgr=qmgr_name -mode=starttrace [clientid=ClientIdentifier]
```

• **Linux** **UNIX**

```
./controlMQXRChannel.sh -qmgr=qmgr_name -mode=starttrace [clientid=ClientIdentifier]
```

其中 `qmgr_name` 是執行 MQXR 服務的佇列管理程式名稱。

將 `ClientIdentifier` 設為 MQTT 用戶端的用戶端 ID。如果您指定 **clientid** 參數，IBM MQ 遙測服務追蹤只會擷取具有該用戶端 ID 的 MQTT 用戶端的活動。

如果您想要追蹤多個特定 MQTT 用戶端的 IBM MQ 遙測服務活動，您可以多次執行指令，每次都指定不同的用戶端 ID。

- c) 重新產生問題。
d) 當問題發生時，請執行下列指令來停止追蹤：

• **Windows**

```
controlMQXRChannel -qmgr=qmgr_name -mode=stoptrace
```

• **Linux** **UNIX**

```
./controlMQXRChannel.sh -qmgr=qmgr_name -mode=stoptrace [clientid=ClientIdentifier]
```

其中 `qmgr_name` 是執行 MQXR 服務的佇列管理程式名稱。

結果

若要檢視追蹤輸出，請跳至下列目錄：

- **Windows** `MQ_DATA_PATH\trace.`
- **Linux** **UNIX** `/var/mqm/trace.`

包含來自 MQXR 服務之資訊的追蹤檔稱為 `mqxr_N.trc`，其中 `N` 是數字。

JMQI 產生的追蹤資訊會寫入稱為 `mqxr_PPPPP.trc` 的追蹤檔，其中 `PPPPP` 是「MQXR 服務」的處理程序 ID。

相關資訊

[strmqtrc](#)

Windows **Linux** **AIX** 使用 `controlMQXRChannel` 指令的其他診斷程式

使用 `controlMQXRChannel` 指令來提供 MQXR 服務的其他診斷資訊。

程序

執行下列指令，以從 MQXR 服務提供有用的診斷資訊：

```
<MQ_INSTALLATION_PATH>\mqxr\bin\controlMQXRChannel -qmgr=<QMGR_NAME> -mode=diagnostics  
-diagnosticstype=<number>
```

產生的診斷資訊取決於 **-diagnosticstype=<number>** 參數的值：

-diagnosticstype= 0

寫入主控台的執行緒傾出

-diagnosticstype= 1

具有部分內部服務統計資料的 FDC

-diagnosticstype= 2

具有內部統計資料的 FDC，以及目前所連接用戶端的相關資訊

-diagnosticstype= 3

資料堆傾出

-diagnosticstype= 4

雅瓦科雷

-diagnosticstype= 5

完整系統傾出

-diagnosticstype= 6

特定用戶端的詳細資訊。請注意，您也必須為該用戶端提供 **-clientid** 參數。

Windows Linux AIX **解決問題: MQTT 用戶端未連接**

解決 MQTT 用戶端程式無法連接至遙測 (MQXR) 服務的問題。

開始之前

問題是出在伺服器上、用戶端還是連線上？您是否已使用 C 或 Java IBM MQTT 用戶端撰寫自己的 MQTT v3 通訊協定處理用戶端或 MQTT 用戶端應用程式？

如需進一步資訊，請參閱 [驗證 MQ Telemetry](#) 的安裝，並檢查遙測通道及遙測 (MQXR) 服務是否正確執行。

關於這項作業

MQTT 用戶端可能未連接至遙測伺服器，或您可能認為它未連接至遙測伺服器，有許多原因。

程序

- 1. 考量可從遙測 (MQXR) 服務傳回至 `MqttClient.Connect` 的原因碼推出的結論。連線失敗的類型為何？

選項	敘述
REASON_CODE_INVALID_PROTOCOL_VERSION	請確保 Socket 位址對應於遙測通道，而且您未將相同的 Socket 位址用於另一個分配管理系統。
REASON_CODE_INVALID_CLIENT_ID	請檢查用戶端 ID 不超過 23 個位元組，且只包含下列範圍內的字元: A-Z, a-z, 0-9, '._/%
REASON_CODE_SERVER_CONNECT_ERROR	檢查遙測 (MQXR) 服務及佇列管理程式是否正確執行。使用 netstat 來檢查 Socket 位址是否未配置給另一個應用程式。

如果您已撰寫 MQTT 用戶端程式庫，而不是使用 MQ Telemetry 所提供的其中一個程式庫，請查看 CONNACK 回覆碼。

從這三個錯誤中，您可以推斷用戶端已連接至遙測 (MQXR) 服務，但該服務已發現錯誤。

- 2. 考量可從遙測 (MQXR) 服務未回應時用戶端所產生的原因碼推出的結論：

選項	敘述
REASON_CODE_CLIENT_EXCEPTION REASON_CODE_CLIENT_TIMEOUT	在伺服器上尋找 FDC 檔; 請參閱 第 202 頁的『 伺服器端日誌 』。當遙測 (MQXR) 服務偵測到用戶端逾時時，它會寫入首

選項	敘述
	次失敗資料擷取 (FDC) 檔案。無論連線何時非預期地岔斷，它都會寫入 FDC 檔。

遙測 (MQXR) 服務可能未回應用戶端，並且用戶端逾時到期。只有在應用程式設定無限期逾時時，MQ Telemetry Java 用戶端才會當掉。在為 `MqttClient.Connect` 設定的逾時到期之後，如果發生無法診斷的問題，則用戶端會擲出這三個異常狀況之一。

除非您找到與連線失敗關聯的 FDC 檔，否則您無法推斷出該用戶端嘗試連接至伺服器：

- a) 確認用戶端已傳送連線要求。
請使用 `tcpmon` 之類的工具來檢查 TCPIP 要求，該工具可從 (例如) <https://code.google.com/p/tcpmon/> 取得
 - b) 用戶端使用的遠端 Socket 位址與定義給遙測通道的 Socket 位址是否相符？
IBM MQ Telemetry 隨附的 Java SE MQTT 用戶端中的預設檔案持續性類別會在用戶端工作目錄中建立名為 `clientIdentifier-tcpHostName` 埠或 `clientIdentifier-sslHostName` 埠的資料夾。資料夾名稱告知您在連線嘗試中使用的 `hostName` 和埠；請參閱第 204 頁的『用戶端日誌檔和用戶端配置檔』。
 - c) 您可以 ping 遠端伺服器位址嗎？
 - d) 在伺服器上執行 `netstat`，是否顯示遙測通道正在用戶端所連接的埠上執行？
3. 檢查遙測 (MQXR) 服務是否在用戶端要求中發現問題。
遙測 (MQXR) 服務會將它偵測到的錯誤寫入 `mqxr_n.log`，而佇列管理程式會將錯誤寫入 `AMQERR01.LOG`；見
 4. 嘗試透過執行另一個用戶端隔離問題。
如需進一步資訊，請參閱 [驗證 MQ Telemetry 安裝](#)
在伺服器平台上執行程式範例以排除網路連線相關的不確定問題，然後在用戶端平台上執行範例。
 5. 要檢查的其他內容：
 - a) 數萬個 MQTT 用戶端是否同時嘗試連接？
遙測通道具有佇列，可以緩存送入連線待辦事項。每秒處理的連線數超過 10,000。待辦事項緩衝區的大小，可以使用「IBM MQ 探險家」中的遙測通道精靈進行配置。其預設大小為 4096。檢查待辦事項是否配置為較低的值。
 - b) 遙測 (MQXR) 服務及佇列管理程式仍在執行中嗎？
 - c) 用戶端是否已連接至已切換其 TCP/IP 位址的高可用性佇列管理程式？
 - d) 防火牆是否選擇性地過濾掉埠或傳回資料封包？

Windows

Linux

AIX

解決問題: 已捨棄 MQTT 用戶端連線

找出導致用戶端在順利連接並執行或長或短的一段時間之後，擲出非預期的 `ConnectionLost` 異常狀況的原因。

開始之前

MQTT 用戶端已順利連接。用戶端可能會執行一長段時間。如果用戶端啟動間隔很短，則順利連接與連接中斷之間的時間可能很短。

區分中斷的連線與成功建立、隨後中斷的連線並不困難。已捨棄連線由呼叫 `MqttCallback.ConnectionLost` 方法的 MQTT 用戶端定義。該方法僅在順利建立連線之後呼叫。該症狀與 `MqttClient.Connect` 在收到負面確認通知或逾時之後擲出異常狀況不同。

如果 MQTT 用戶端應用程式未使用 IBM MQ 所提供的 MQTT 用戶端程式庫，則症狀取決於用戶端。在 MQTT v3 通訊協定中，症狀是對伺服器要求沒有及時回應，或 TCP/IP 連線失敗。

關於這項作業

MQTT 用戶端會以可擲出的異常狀況來呼叫 `MqttCallback.ConnectionLost`，以回應在收到正面連線確認通知之後所遇到的任何伺服器端問題。當 MQTT 用戶端從 `MqttTopic.publish` 返回且 `MqttClient.subscribe` 時，會將要求傳送至負責傳送及接收訊息的 MQTT 用戶端執行緒。會透過將可擲出異常狀況傳遞至 `ConnectionLost` 回呼方法，來以非同步方式報告伺服器端錯誤。

程序

1. 是否已啟動另一個使用相同 `ClientIdentifier` 的用戶端？

如果已啟動第二個用戶端或已重新啟動相同的用戶端（使用相同的 `ClientIdentifier`），則會中斷與第一個用戶端的第一個連線。

2. 用戶端是否存取了未獲授權來發佈至或訂閱的主題？

Telemetry 服務代表用戶端執行的所有動作，它們傳回 `MQCC_FAIL` 導致服務中斷用戶端連線連線。

不會將原因碼傳回至用戶端。

- 在用戶端連接至的佇列管理程式之 `mqxr.log` 和 `AMQERR01.LOG` 檔中尋找日誌訊息；請參閱第 202 頁的『伺服器端日誌』。

3. TCP/IP 連線是否已中斷？

防火牆可能設定較低的逾時以將 TCP/IP 連線標示為非作用中，從而中斷連線。

- 使用 `MqttConnectOptions.setKeepAliveInterval` 縮短非作用中的 TCP/IP 連線時間。

Windows

Linux

AIX

解決問題: MQTT 應用程式中遺失訊息

解決遺失訊息問題。訊息是非持續性的、傳送至錯誤的位置還是從未傳送？編寫不正確的用戶端程式可能會遺失訊息。

開始之前

您是否確定所傳送訊息已遺失？您可以因未收到訊息而推斷它已遺失嗎？如果訊息是個發佈，遺失哪一種訊息：發佈者傳送的訊息還是傳送至訂閱者的訊息？或者是訂閱遺失，還是分配管理系統未將該訂閱的發佈傳送至訂閱者？

如果解決方案涉及使用叢集或發佈/訂閱階層的分散式發佈/訂閱，則有許多配置問題可能會導致訊息遺失的現象。

如果您傳送含有至少一次或最多一次服務品質的訊息，則可能未以您預期的方式遞送您認為遺失的訊息。不太可能已從系統誤刪該訊息。系統可能無法建立您預期的發佈或訂閱。

在進行遺失訊息問題判斷過程中，您執行的最重要步驟是確認訊息已遺失。重建該實務範例，遺失更多訊息。請使用至少一次或最多一次服務品質來刪除系統捨棄訊息的所有情況。

關於這項作業

診斷遺失的訊息有四個要素。

1. 發動並忘記 依設計運作的訊息。系統有時會捨棄 隨發動即忘 訊息。
2. 配置：並非以直接明確的方式，在分散式環境中使用正確的權限來設定發佈/訂閱。
3. 用戶端程式設計錯誤：訊息遞送的責任不只是由 IBM 撰寫的程式碼。
4. 如果您已用盡所有這些可能性，則可能決定涉及「IBM 支援中心」。

程序

1. 如果遺失訊息具有 發動並忘記 服務品質，請設定至少一次或最多一次服務品質。重試遺失該訊息。

- 在許多情況下，IBM MQ 會捨棄使用 Fire and 忘記 服務品質所傳送的訊息：
 - 通訊中斷和通道已停止。

- 佇列管理程式關閉。
 - 訊息數過多。
 - **發動並忘記** 訊息的遞送取決於 TCP/IP 的可靠性。TCP/IP 會不斷重新傳送資料封包，直到其遞送得到確認為止。如果 TCP/IP 階段作業岔斷，則會遺失具有 **Fire and 忘記** 服務品質的訊息。階段作業被岔斷的可能原因有：用戶端或伺服器關閉、發生通訊問題或防火牆使階段作業中斷連線。
2. 檢查用戶端是否正在重新啟動前一個階段作業，以便再次傳送具有 **至少一次** 或 **最多一次** 服務品質的未遞送訊息。
 - a) 如果用戶端應用程式正在使用 Java SE MQTT 用戶端，請檢查它是否將 `MqttClient.CleanSession` 設為 `false`
 - b) 如果您使用的是不同的用戶端程式庫，請檢查是否正確地重新啟動階段作業。
 3. 請檢查用戶端應用程式是否正在重新啟動相同的階段作業，且未錯誤地啟動不同的階段作業。
若要再次啟動相同的階段作業，請設定 `cleanSession = false`，而且 `Mqttclient.clientIdentifier` 和 `MqttClient.serverURI` 必須與前一個階段作業相同。
 4. 如果階段作業已永久關閉，請檢查用戶端持續性儲存庫中可用的訊息，以再次傳送。
 - a) 如果用戶端應用程式使用 Java SE MQTT 用戶端，請檢查訊息是否儲存在持續性資料夾中；請參閱 [第 204 頁的『用戶端日誌檔和用戶端配置檔』](#)
 - b) 如果您使用的是不同的用戶端程式庫，或者您已經實作自己的持續性機制，請檢查它是否正確地運作。
 5. 檢查訊息在遞送之前沒有被刪除。
等待遞送至 MQTT 用戶端的未遞送訊息儲存在 `SYSTEM.MQTT.TRANSMIT.QUEUE` 中。等待遞送至遙測伺服器的訊息採用用戶端持續性機制進行儲存；請參閱 [MQTT 用戶端中的訊息持續性](#)。
 6. 檢查用戶端是否有訂閱預期要接收的發佈。
使用 IBM MQ Explorer 或使用 `runmqsc` 或 PCF 指令來列出訂閱。所有 MQTT 用戶端訂閱皆已命名。他們會取得下列格式的名稱：`ClientIdentifier:Topic name`
 7. 檢查發佈者是否具有發佈的權限，以及訂閱者是否具有訂閱發佈主題的權限。

```
dspmqaut -m qMgr -n topicName -t topic -p user ID
```

在叢集化發佈/訂閱系統中，必須針對訂閱者所連接佇列管理程式上的主題，為訂閱者授與權限。不必針對訂閱發佈所在佇列管理程式上的主題，為訂閱者授與權限。必須正確授與佇列管理程式之間通道的權限，才能傳遞 Proxy 訂閱和轉遞發佈。

建立相同的訂閱，並使用 IBM MQ Explorer 發佈至該訂閱。使用用戶端公用程式，模擬應用程式用戶端的發佈和訂閱功能。從 IBM MQ Explorer 啟動公用程式，並變更其使用者 ID，以符合用戶端應用程式所採用的使用者 ID。

8. 檢查訂閱者是否具有可將發佈放置在 `SYSTEM.MQTT.TRANSMIT.QUEUE` 上的權限。

```
dspmqaut -m qMgr -n queueName -t queue -p user ID
```

9. 檢查 IBM MQ 點對點應用程式是否有權將其訊息放置在 `SYSTEM.MQTT.TRANSMIT.QUEUE` 上。

```
dspmqaut -m qMgr -n queueName -t queue -p user ID
```

請參閱 [直接傳送訊息至用戶端](#)。

Windows

Linux

AIX

解決問題：遙測 (MQXR) 服務未啟動

解決遙測 (MQXR) 服務無法啟動的問題。請檢查 MQ Telemetry 安裝，且沒有任何檔案遺漏、移動或具有錯誤的許可權。檢查遙測 (MQXR) 服務所使用的路徑，找出遙測 (MQXR) 服務程式。

開始之前

已安裝 MQ Telemetry 特性。IBM MQ Explorer 在 **IBM MQ > 佇列管理程式 > qMgr 名稱 > 遙測**中具有遙測資料夾。如果該資料夾不存在，則安裝失敗。

遙測 (MQXR) 服務必須已建立，其才能啟動。如果尚未建立遙測 (MQXR) 服務，請執行 **定義配置範例 ... Telemetry** 資料夾中的精靈。

如果先前曾啟動過遙測 (MQXR) 服務，則會在 Telemetry 資料夾下另外建立**通道及通道狀態**資料夾。Telemetry 服務 SYSTEM.MQXR.SERVICE 位於 **Services** 資料夾。如果按一下顯示「系統物件」的 IBM MQ Explorer 圓鈕，則可以看到它。

用滑鼠右鍵按一下 SYSTEM.MQXR.SERVICE，以啟動和停止服務，顯示其狀態，並顯示您的使用者 ID 是否有權啟動服務。

關於這項作業

SYSTEM.MQXR.SERVICE 遙測 (MQXR) 服務無法啟動。無法以兩種不同的方法啟動資訊清單本身：

1. start 指令立即失敗。
2. start 指令成功，但是服務隨即停止。

程序

1. 啟動服務。

結果

服務立即停止。視窗顯示錯誤訊息；例如：

```
IBM MQ cannot process the request because the executable specified cannot be started. (AMQ4160)
```

原因

安裝中遺漏檔案，或者已安裝檔案的權限設定錯誤。

MQ Telemetry 特性僅安裝在一對高可用性佇列管理程式中的其中一個上。如果佇列管理程式實例切換至待用伺服器，則它會嘗試啟動 SYSTEM.MQXR.SERVICE。啟動服務的指令會因待用伺服器上未安裝遙測 (MQXR) 服務而失敗。

調查

查看錯誤日誌；請參閱第 202 頁的『[伺服器端日誌](#)』。

動作

安裝 MQ Telemetry 特性，或將它解除安裝後再重新安裝。

2. 啟動服務；等待 30 秒；重新整理 IBM MQ Explorer 並檢查服務狀態。

結果

服務啟動，然後停止。

原因

SYSTEM.MQXR.SERVICE 已啟動 **runMQXRService** 指令，但指令失敗。

調查

查看錯誤日誌；請參閱第 202 頁的『[伺服器端日誌](#)』。

瞭解是否僅定義的範例通道發生問題。備份並清除 *WMQ data*

directory\Qmgrs\qMgrName\mqxr 目錄的內容。執行配置範例精靈，嘗試啟動該服務。

動作

尋找權限和路徑問題。

Windows

Linux

AIX

解決問題：Telemetry 服務未呼叫 JAAS 登入模組

瞭解遙測 (MQXR) 服務是否未呼叫您的 JAAS 登入模組，並配置 JAAS 以更正問題。

開始之前

您已修改 `WMQ installation directory\mqxr\samples\samples>LoginModule.java`，以建立您自己的鑑別類別 `WMQ installation directory\mqxr\samples\samples>LoginModule.class`。此外，您已撰寫自己適用的 JAAS 鑑別類別，並將它們放置在您選擇的目錄中。對遙測 (MQXR) 服務進行一些初步測試後，您懷疑遙測 (MQXR) 服務未呼叫您的鑑別類別。

註：防止您的鑑別類別可能被套用至 IBM MQ 的維護所改寫。針對鑑別類別使用您自己的路徑，而不是 IBM MQ 目錄樹狀結構內的路徑。

關於這項作業

作業使用實務範例來說明如何解決問題。在該實務範例中，名為 `security.jaas` 的套件包含名為 `JAASLogin.class` 的鑑別類別。該類別儲存在路徑 `C:\WMQTelemetryApps\security\jaas` 中。請參閱 [遙測通道 JAAS 配置](#) 及 [AuthCallback MQXR 類別](#)，以取得為 MQ Telemetry 配置 JAAS 的說明。範例第 211 頁的『範例 JAAS 配置』是一個配置範例。

程序

1. 在 `mqxr.log` 中尋找 `javax.security.auth.login.LoginException` 擲出的異常狀況。
請參閱第 202 頁的『伺服器端日誌』以取得 `mqxr.log` 的路徑，並參閱第 213 頁的圖 26 取得日誌中列出的異常狀況範例。
2. 透過將 JAAS 配置與第 211 頁的『範例 JAAS 配置』中已運作的範例進行比較，更正 JAAS 配置。
3. 將範例 `JAASLoginModule` 重構到鑑別套件並使用相同的路徑進行部署之後，用其取代登入類別。將 `loggedIn` 的值在 `true` 和 `false` 之間切換。
如果 `loggedIn` 是 `true` 時問題消失，而 `loggedIn` 是 `false` 時問題出現，則登入類別有問題。
4. 檢查問題是否與授權相關，而不是與鑑別相關。
 - a) 變更遙測通道定義，以使用固定的使用者 ID 執行授權檢查。選取屬於 `mqm` 群組的使用者 ID。
 - b) 重新執行用戶端應用程式。

如果問題消失，則解決方案在於要傳遞以授權的使用者 ID。要傳遞的使用者名稱為何？請將其列印到登入模組中的檔案。使用「IBM MQ 探險家」或 `dspmqaauth` 來檢查其存取權。

範例 JAAS 配置

在「IBM MQ 探險家」中使用 [新建遙測通道](#) 精靈，以配置遙測通道。

JAAS 配置檔有一個名為 `JAASConfig` 的段落，它會將 Java 類別命名為 `security.jaas.JAASLogin`，JAAS 用來鑑別用戶端。

```
JAASConfig {
    security.jaas.JAASLogin required debug=true;
};
```

圖 21: `WMQ Installation directory\data\qmgrs\qMgrName\mqxr\jaas.config`

當 `SYSTEM.MQTT.SERVICE` 啟動時，它會將第 211 頁的圖 22 中的路徑新增至其類別路徑。

```
CLASSPATH=C:\WMQTelemetryApps;
```

圖 22: `WMQ Installation directory\data\qmgrs\qMgrName\service.env`

第 212 頁的圖 23 顯示新增至為遙測 (MQXR) 服務所設定類別路徑的第 211 頁的圖 22 中的附加路徑。

```
CLASSPATH=;C:\IBM\MQ\Program\mqxr\bin\..\lib\MQXRListener.jar;  
C:\IBM\MQ\Program\mqxr\bin\..\lib\WMQCommonServices.jar;  
C:\IBM\MQ\Program\mqxr\bin\..\lib\objectManager.utils.jar;  
C:\IBM\MQ\Program\mqxr\bin\..\lib\com.ibm.micro.xr.jar;  
C:\IBM\MQ\Program\mqxr\bin\..\..\java\lib\com.ibm.mq.jmqi.jar;  
C:\IBM\MQ\Program\mqxr\bin\..\..\java\lib\com.ibm.mqjms.jar;  
C:\IBM\MQ\Program\mqxr\bin\..\..\java\lib\com.ibm.mq.jar;  
C:\WMQTelemetryApps;
```

圖 23: *runMQXRService.bat* 的類別路徑輸出

第 212 頁的圖 24 中的輸出顯示遙測 (MQXR) 服務已啟動。

```
21/05/2010 15:32:12 [main] com.ibm.mq.MQXRService.MQXRPropertiesFile  
AMQXR2011I: Property com.ibm.mq.MQXR.channel/JAASMCUser value  
com.ibm.mq.MQXR.Port=1884;  
com.ibm.mq.MQXR.JAASConfig=JAASConfig;  
com.ibm.mq.MQXR.UserName=Admin;  
com.ibm.mq.MQXR.StartWithMQXRService=true
```

圖 24: *WMQ Installation directory\data\qmgrs\qMgrName\errors*

當用戶端應用程式連接至 JAAS 通道時，如果 `com.ibm.mq.MQXR.JAASConfig=JAASWrongConfig` 不符合 `jaas.config` 檔中 JAAS 段落的名稱，連線會失敗，且用戶端會擲出異常狀況，回覆碼為 0；請參閱第 212 頁的圖 25。擲出第二個異常狀況 `Client is not connected (32104)` 的原因，是用戶端在未連接的情況下，嘗試中斷連線。

```
Connecting to tcp://localhost:1883 with client ID SampleJavaV3_publish  
reason 5  
msg Not authorized to connect  
loc Not authorized to connect  
cause null  
except Not authorized to connect (5)  
Not authorized to connect (5)  
at  
org.eclipse.paho.client.mqttv3.internal.ExceptionHelper.createMqttException(ExceptionHelper.java  
:28)  
at  
org.eclipse.paho.client.mqttv3.internal.ClientState.notifyReceivedAck(ClientState.java:885)  
at org.eclipse.paho.client.mqttv3.internal.CommsReceiver.run(CommsReceiver.java:118)  
at java.lang.Thread.run(Thread.java:809)
```

圖 25: 連接至 *Eclipse Paho* 範例時擲出異常狀況

`mqxr.log` 包含第 212 頁的圖 25 中所示的其他輸出。

JAAS 會偵測到錯誤，會擲出 `javax.security.auth.login.LoginException`，原因為 `No LoginModules configured for JAAS`。原因可能是配置名稱不正確，如第 213 頁的圖 26 所示。它也可能是在載入 JAAS 配置期間，JAAS 發生的其他問題引起的。

如果 JAAS 未報告異常狀況，則 JAAS 已順利載入在 `JAASConfig` 段落中指名的 `security.jaas.JAASLogin` 類別。

```
15/06/15 13:49:28.337
AMQXR2050E: Unable to load JAAS config:MQXRWrongConfig.
The following exception occurred javax.security.auth.login.LoginException:
No LoginModules configured for MQXRWrongConfig
```

圖 26: 載入 JAAS 配置時發生錯誤

失敗後回復

請遵循一組程序，在發生嚴重問題之後進行回復。

關於這項作業

如果您無法使用「疑難排解及支援」一節中所說明的診斷技術來解決基礎問題，請使用這裡說明的回復方法。如果無法使用這些回復技術來解決您的問題，請聯絡「IBM 支援中心」。

程序

如需如何從不同類型的失敗回復的指示，請參閱下列鏈結：

- [第 214 頁的『磁碟機故障』](#)
- [第 215 頁的『佇列管理程式物件已損壞』](#)
- [第 215 頁的『損壞的單一物件』](#)
- [第 215 頁的『自動媒體回復失敗』](#)

z/OS

如需如何在 IBM MQ for z/OS 上從不同類型的失敗回復的指示，請參閱下列鏈結：

- [z/OS 第 216 頁的『共用佇列問題』](#)
- [z/OS 第 216 頁的『作用中日誌問題』](#)
- [z/OS 第 221 頁的『保存日誌問題』](#)
- [z/OS 第 224 頁的『BSDS 問題』](#)
- [z/OS 第 230 頁的『頁集問題』](#)
- [z/OS 第 231 頁的『連結機能及 Db2 問題』](#)
- [z/OS 第 233 頁的『長時間執行工作單元的問題』](#)
- [z/OS 第 234 頁的『IMS 相關問題』](#)
- [z/OS 第 235 頁的『硬體問題』](#)

相關概念

[第 7 頁的『IBM MQ 疑難排解及支援』](#)

如果您的佇列管理程式網路或 IBM MQ 應用程式有問題，請使用說明的技術來協助您診斷及解決問題。

第 7 頁的『疑難排解概觀』

疑難排解是尋找並消除問題原因的程序。每當您使用 IBM 軟體時發生問題，只要您問自己「發生什麼事？」，疑難排解程序即會開始。

相關工作

第 36 頁的『聯絡 IBM 支援中心』

如果您在使用 IBM MQ 時遇到問題需要協助，您可以透過「IBM 支援中心網站」聯絡 IBM 支援中心。您也可以訂閱 IBM MQ 修正程式、疑難排解及其他新聞的相關通知。

第 9 頁的『對 UNIX, Linux, and Windows 進行起始檢查』

在 UNIX, Linux, and Windows 上詳細開始問題判斷之前，請考量是否有明顯的問題原因，或可能提供有用結果的調查區域。這種診斷方法通常可以藉由強調簡單錯誤或縮小可能性範圍來節省大量工作。

相關資訊

備份及還原 IBM MQ

 在 z/OS 上規劃備份及回復

磁碟機故障

包含佇列管理程式資料及/或日誌的磁碟機可能有問題。問題可能包括資料流失或毀損。這三個觀察值僅在存活的資料部分 (如果有的話) 有差異。

在 **所有** 案例中，請先檢查目錄結構是否有任何損壞，必要的話，請修復這類損壞。如果您遺失佇列管理程式資料，佇列管理程式目錄結構可能已損壞。如果是，請先手動重建目錄樹狀結構，然後再重新啟動佇列管理程式。

如果佇列管理程式資料檔已發生損壞，但佇列管理程式日誌檔未發生損壞，則佇列管理程式通常可以重新啟動。如果佇列管理程式日誌檔發生任何損壞，則佇列管理程式可能無法重新啟動。

在檢查結構損壞之後，您可以執行一些動作，視您使用的記載類型而定。

- **如果目錄結構有重大損壞或日誌有任何損壞**，請將所有舊檔案 (包括配置檔、日誌及佇列管理程式目錄) 移除回到 QMgrName 層次，並還原前次備份，然後重新啟動佇列管理程式。
- **對於具有媒體回復的線性記載**，請確保目錄結構完整，並重新啟動佇列管理程式。如果佇列管理程式重新啟動，請使用 MQSC 指令 (例如 DISPLAY QUEUE) 來檢查是否有任何其他物件已損壞。使用 rcrmobj 指令來回復您找到的那些資料。例如：

```
rcrmobj -m QMgrName -t all *
```

其中 QMgrName 是要回復的佇列管理程式。-t all * 指出要回復任何類型的所有損壞物件。如果只有一或兩個物件報告為已損壞，則您可以在這裡依名稱及類型指定那些物件。

- **對於具有媒體回復及未損壞日誌的線性記載**，您可以還原佇列管理程式資料的備份，讓現有的日誌檔及日誌控制檔保持不變。當發生失敗時，啟動佇列管理程式會套用日誌中的變更，使佇列管理程式回到其狀態。

此方法依賴兩件事：

1. 您必須將檢查點檔案還原為佇列管理程式資料的一部分。此檔案包含決定必須套用日誌中多少資料以提供一致佇列管理程式的資訊。
2. 您必須具有在備份時啟動佇列管理程式所需的最舊日誌檔，以及日誌檔目錄中可用的所有後續日誌檔。

如果無法這樣做，請同時還原佇列管理程式資料和日誌兩者的備份，這兩者都是同時取得的。這會導致訊息完整性遺失。

- **對於循環式記載**，如果佇列管理程式日誌檔已損壞，請從您擁有的最新備份還原佇列管理程式。還原備份之後，請重新啟動佇列管理程式，並檢查是否有損壞的物件。不過，因為您沒有媒體回復，所以您必須尋找其他重建損壞物件的方法。

如果佇列管理程式日誌檔未損壞，佇列管理程式通常可以重新啟動。重新啟動之後，您必須識別所有損壞的物件，然後刪除並重新定義它們。

佇列管理程式物件已損壞

如果佇列管理程式在正常作業期間報告損壞的物件，該如何處理。

在這些情況下，有兩種回復方式，視您使用的記載類型而定：

- 對於線性記載，請手動刪除包含損壞物件的檔案，然後重新啟動佇列管理程式。(您可以使用 `dspmqls` 指令來判定損壞物件的實際檔案系統名稱。)損壞物件的媒體回復是自動的。
- 若為循環式記載，請還原佇列管理程式資料及日誌的前次備份，然後重新啟動佇列管理程式。

如果您使用循環式記載，則還有另一個選項。對於損壞的佇列或其他物件，請刪除物件並重新定義物件。如果是佇列，這個選項不容許您回復佇列上的任何資料。

註：從備份還原可能已過期，因為您必須關閉佇列管理程式，才能取得佇列檔的全新備份。

損壞的單一物件

如果單一物件在正常作業期間報告為已損壞，則對於線性記載，您可以從其媒體映像檔重建物件。不過，對於循環式記載，您無法重建單一物件。

自動媒體回復失敗

如果具有線性日誌的佇列管理程式啟動所需的本端佇列已損壞，且自動媒體回復失敗，請還原佇列管理程式資料及日誌的前次備份，然後重新啟動佇列管理程式。

z/OS z/OS 上的回復程序範例

請使用本主題作為各種回復程序的參考。

本主題說明在各種錯誤狀況之後回復 IBM MQ 的程序。這些錯誤狀況分為下列種類：

問題種類	問題	下一個要尋找的位置
共用佇列問題	專用及共用佇列的定義衝突。	第 216 頁的『共用佇列問題』
作用中日誌問題	<ul style="list-style-type: none">• 遺失雙重記載。• 作用中日誌已停止。• 作用中日誌資料集的一個或兩個副本已損壞。• 在作用中日誌資料集上寫入錯誤。• 作用中日誌已滿或已滿。• 作用中日誌資料集上的讀取錯誤。	第 216 頁的『作用中日誌問題』
保存日誌問題	<ul style="list-style-type: none">• DASD 空間不足，無法完成卸載作用中日誌資料集。• 卸載作業已異常終止。• 保存資料集配置問題。 1• 重新啟動期間，保存資料集上的讀取 I/O 錯誤。	第 221 頁的『保存日誌問題』
BSDS 問題	<ul style="list-style-type: none">• 開啟 BSDS 時發生錯誤。• 日誌內容與 BSDS 資訊不對應。• BSDS 的兩個副本都已損壞。• 時間戳記不相等。• 雙重 BSDS 資料集不同步。• BSDS 上的 I/O 錯誤。	第 224 頁的『BSDS 問題』

表 20: 回復程序範例 (繼續)		
問題種類	問題	下一個要尋找的位置
頁集問題	<ul style="list-style-type: none"> • 頁集已滿。 • 頁集有 I/O 錯誤。 	第 230 頁的『 頁集問題 』
連結機能及 Db2 問題	<ul style="list-style-type: none"> • 儲存體媒體已滿。 • Db2 系統失敗。 • Db2 資料共用群組失敗。 • Db2 及連結機能失敗。 	第 231 頁的『 連結機能及 Db2 問題 』
工作單元問題	發現長時間執行的工作單元。	第 233 頁的『 長時間執行工作單元的問題 』
IMS 問題	<ul style="list-style-type: none"> • IMS 應用程式異常終止。 • IMS 配接器無法連接至 IBM MQ。 • IMS 無法運作。 	第 234 頁的『 IMS 相關問題 』
硬體問題	媒體回復程序	第 235 頁的『 硬體問題 』

z/OS 共用佇列問題

如果 IBM MQ 探索到已定義頁集型佇列及相同名稱的共用佇列，則會發生問題。

症狀

IBM MQ 會發出下列訊息：

```
CSQI063E +CSQ1 QUEUE queue-name IS BOTH PRIVATE AND SHARED
```

在佇列管理程式重新啟動期間，IBM MQ 發現以頁集為基礎的佇列與同名的共用佇列同時存在。

系統動作

重新啟動處理完成之後，對該佇列名稱的任何 MQOPEN 要求都會失敗，指出共存性問題。

系統程式設計師動作

無。

操作員動作

請刪除佇列的一個版本，以容許處理該佇列名稱。如果佇列中有必須保留的訊息，您可以使用 MOVE QLOCAL 指令將訊息移至另一個佇列。

z/OS 作用中日誌問題

請利用這個主題來解決作用中日誌的不同問題。

本主題涵蓋下列作用中日誌問題：

- [第 217 頁的『遺失雙重記載』](#)
- [第 217 頁的『作用中日誌已停止』](#)
- [第 218 頁的『作用中日誌資料集的一個或兩個副本已損壞』](#)
- [第 218 頁的『在作用中日誌資料集上寫入 I/O 錯誤』](#)
- [第 219 頁的『讀取作用中日誌時發生 I/O 錯誤』](#)
- [第 220 頁的『作用中日誌已滿』](#)

- 作用中日誌已滿

遺失雙重記載

症狀

IBM MQ 會發出下列訊息：

```
CSQJ004I +CSQ1 ACTIVE LOG COPY n INACTIVE, LOG IN SINGLE MODE,  
ENDRBA=...
```

完成一個作用中日誌資料集之後，IBM MQ 發現後續的 (COPY n) 資料集未卸載或標示為已停止。

系統動作

IBM MQ 會以單一模式繼續執行，直到卸載完成，然後回到雙重模式。

系統程式設計師動作

無。

操作員動作

請檢查卸載處理程序是否正在繼續，且未等待磁帶裝載。您可能需要執行列印日誌對映公用程式，以判定所有資料集的狀態。您可能需要定義其他資料集。

作用中日誌已停止

症狀

IBM MQ 會發出下列訊息：

```
CSQJ030E +CSQ1 RBA RANGE startdba TO enddba NOT AVAILABLE IN ACTIVE  
LOG DATA SETS
```

系統動作

IBM MQ 無法使用包含訊息 CSQJ030E 中所報告 RBA 範圍的作用中日誌資料集。這些日誌在 BSDS 中的狀態為 STOPPED。佇列管理程式會以傾出終止。

系統程式設計師動作

您必須先解決此問題，然後再重新啟動佇列管理程式。日誌 RBA 範圍必須可用，IBM MQ 才能回復。在 BSDS 中標示為 STOPPED 的作用中日誌將永不重複使用或保存，這會在日誌中產生一個孔。

尋找指出日誌資料集停止原因的訊息，並遵循這些訊息的指示。

修改 BSDS 作用中日誌庫存，以重設 STOPPED 狀態。如果要這麼做，請在佇列管理程式終止之後遵循下列程序：

1. 使用列印日誌公用程式 (CSQJU004) 來取得 BSDS 日誌庫存的副本。這會顯示日誌資料集的狀態。
2. 使用變更日誌庫存公用程式 (CSQJU003) 的 DELETE 函數來刪除標示為 STOPPED 的作用中日誌資料集。
3. 使用 CSQJU003 的 NEWLOG 功能，將作用中日誌新增回 BSDS 庫存。必須在 NEWLOG 陳述式上指定每一個作用中日誌資料集的開始及結束 RBA。(您可以從步驟 1 中取得的列印日誌公用程式報告中找到要使用的正確值。)
4. 重新執行 CSQJU004。標示為 STOPPED 的作用中日誌資料集現在會顯示為 NEW 且無法重複使用。這些作用中日誌將在適當時間保存。
5. 重新啟動佇列管理程式。

註：如果您的佇列管理程式以雙重 BSDS 模式執行，則必須更新兩個 BSDS 庫存。

作用中日誌資料集的一個或兩個副本已損壞

症狀

IBM MQ 會發出下列訊息:

```
CSQJ102E +CSQ1 LOG RBA CONTENT OF LOG DATA SET DSNAME=...,  
          STARTRBA=..., ENDRBA=...,  
          DOES NOT AGREE WITH BSDS INFORMATION  
CSQJ232E +CSQ1 OUTPUT DATA SET CONTROL INITIALIZATION PROCESS FAILED
```

系統動作

佇列管理程式啟動處理程序已終止。

系統程式設計師動作

如果資料集的一個副本已損壞，請執行下列步驟:

1. 重新命名損壞的作用中日誌資料集，並定義取代資料集。
2. 將未損壞的資料集複製到取代資料集。
3. 使用變更日誌庫存公用程式來執行下列動作:
 - 從 BSDS 移除與損壞資料集相關的資訊。
 - 將與取代資料集相關的資訊新增至 BSDS。
4. 重新啟動佇列管理程式。

如果兩個作用中日誌資料集副本都已損壞，現行頁集可供使用，**且佇列管理程式已完全關閉**，請執行下列步驟:

1. 重新命名損壞的作用中日誌資料集，並定義取代資料集。
2. 使用變更日誌記錄公用程式來執行下列動作:
 - 從 BSDS 移除與損壞資料集相關的資訊。
 - 將與取代資料集相關的資訊新增至 BSDS。
3. 重新命名現行頁面集並定義取代頁面集。
4. 使用 CSQUTIL (FORMAT 和 RESETPAGE) 來格式化取代頁集，並將重新命名的頁集複製到它們。
RESETPAGE 函數也會重設取代頁集中的日誌資訊。

如果佇列管理程式未完全關閉，您必須從先前已知的一致性點還原系統，或執行冷啟動(如 [重新起始設定佇列管理程式](#) 中所述)。

操作員動作

無。

在作用中日誌資料集上寫入 I/O 錯誤

症狀

IBM MQ 會發出下列訊息:

```
CSQJ105E +CSQ1 csect-name LOG WRITE ERROR DSNAME=...,  
          LOGRBA=..., ERROR STATUS=ccccffss
```

系統動作

IBM MQ 會執行下列步驟:

1. 在 BSDS 中將錯誤為 TRUNCATED 的日誌資料集標示為 TRUNCATED。
2. 移至下一個可用的資料集。

3. 如果使用雙重作用中記載，則會在相同點截斷另一個副本。

截斷資料集中的資料稍後會如常卸載。

資料集將在下一個週期重複使用。

系統程式設計師動作

無。

操作員動作

如果此資料集上仍有錯誤，請在下一次卸載處理程序之後關閉佇列管理程式。然後使用「存取方法服務 (AMS)」及變更日誌庫存公用程式來新增取代項目。(如需指示，請參閱 [變更 BSDS](#)。)

讀取作用中日誌時發生 I/O 錯誤

症狀

IBM MQ 會發出下列訊息：

```
CSQJ106E +CSQ1 LOG READ ERROR DSNAME=..., LOGRBA=...,  
ERROR STATUS=ccccffss
```

系統動作

這取決於發生錯誤的時間：

- 如果在卸載處理程序期間發生錯誤，則處理程序會嘗試從第二個副本讀取 RBA 範圍。
 - 如果沒有第二個副本存在，則會停止作用中日誌資料集。
 - 如果第二個副本也有錯誤，則只會停止觸發卸載程序的原始資料集。然後會終止保存日誌資料集，並在保存日誌 RBA 範圍中留下間隙。
 - 發出此訊息：

```
CSQJ124E +CSQ1 OFFLOAD OF ACTIVE LOG SUSPENDED FROM  
RBA xxxxxx TO RBA xxxxxx DUE TO I/O ERROR
```

- 如果第二個副本令人滿意，則第一個副本不會停止。
- 如果在回復期間發生錯誤，則 IBM MQ 會提供從另一個副本或保存所要求的特定日誌 RBA 中的資料。如果不成功，則回復不會成功，且佇列管理程式會異常終止。
- 如果在重新啟動期間發生錯誤，如果使用雙重記載，則 IBM MQ 會繼續替代日誌資料集，否則佇列管理程式會異常結束。

系統程式設計師動作

尋找系統訊息 (例如 IEC 字首訊息)，並嘗試使用這些訊息的建議動作來解決問題。

如果作用中日誌資料集已停止，則不會用於記載。資料集未取消配置；它仍用於讀取。即使資料集未停止，也應該取代提供持續錯誤的作用中日誌資料集。

操作員動作

無。

取代資料集

如何取代資料集取決於您是使用單一或雙重作用中記載。

如果您使用雙重作用中記載：

1. 請確定已儲存資料。

資料會儲存在另一個作用中日誌上，且可以複製到替代作用中日誌。

2. 使用「存取方法服務」時發生錯誤，請停止佇列管理程式並刪除資料集。

3. 請使用 Access Method Services DEFINE 來重新定義新的日誌資料集，以便您可以寫入其中。使用 DFDSS 或 Access Method Services REPRO 將良好登入複製到重新定義的資料集，以便您有兩個一致且正確的日誌。
4. 使用變更日誌庫存公用程式 CSQJU003 來更新 BSDS 中關於毀損資料集的資訊，如下所示：
 - a. 使用 DELETE 函數來移除毀損資料集的相關資訊。
 - b. 使用 NEWLOG 函數將新的資料集命名為新的作用中日誌資料集，並為它提供已順利複製的 RBA 範圍。

您可以在相同的工作步驟中執行 DELETE 及 NEWLOG 功能。在 SYSIN 輸入資料集中，將 DELETE 陳述式放在 NEWLOG 陳述式之前。
5. 重新啟動佇列管理程式。

如果您使用單一作用中記載：

1. 請確定已儲存資料。
2. 停止佇列管理程式。
3. 判定是否已卸載具有錯誤的資料集：
 - a. 使用 CSQJU003 公用程式來列出 BSDS 中保存日誌資料集的相關資訊。
 - b. 搜尋清單中具有 RBA 範圍 (包括毀損資料集的 RBA) 的資料集。
4. 如果已卸載毀損的資料集，請將保存日誌中的備份複製到新的資料集。然後，跳至步驟 6。
5. 如果作用中日誌資料集已停止，則不會卸載 RBA。使用 DFDSS 或 Access Method Services REPRO，將資料從毀損的資料集複製到新的資料集。

如果進一步 I/O 錯誤導致您無法複製整個資料集，則日誌中會出現間隙。

註：如果在日誌中偵測到間隙，則佇列管理程式重新啟動將不會成功。

6. 使用變更日誌庫存公用程式 CSQJU003 來更新 BSDS 中關於毀損資料集的資訊，如下所示：
 - a. 使用 DELETE 函數來移除毀損資料集的相關資訊。
 - b. 使用 NEWLOG 函數將新資料集命名為新的作用中日誌資料集，並為它提供已順利複製的 RBA 範圍。

DELETE 及 NEWLOG 函數可以在相同的工作步驟中執行。在 SYSIN 輸入資料集中，將 DELETE 陳述式放在 NEWLOG 陳述式之前。
7. 重新啟動佇列管理程式。

作用中日誌已滿

作用中日誌可能因數個原因而填滿，例如，卸載延遲及記載過多。如果作用中日誌空間不足，則會產生嚴重後果。當作用中日誌已滿時，佇列管理程式會中止處理，直到卸載處理程序完成為止。如果卸載處理在作用中日誌已滿時停止，佇列管理程式可能會異常結束。需要先採取更正動作，才能重新啟動佇列管理程式。

症狀

由於作用中日誌已滿的嚴重含意，當最後一個可用的作用中日誌資料集已滿 5% 時，佇列管理程式會發出下列警告訊息：

```
CSQJ110E +CSQ1 LAST COPYn ACTIVE LOG DATA SET IS nnn PERCENT FULL
```

並在填入每額外 5% 的資料集空間之後重新發出訊息。每次發出訊息時，都會啟動卸載處理程序。

系統動作

已發出訊息並已啟動卸載處理。如果作用中日誌已滿，則會採取進一步動作。請參閱 [第 221 頁的『作用中日誌已滿』](#)

系統程式設計師動作

使用 DEFINE LOG 指令來動態新增進一步的作用中日誌資料集。這允許 IBM MQ 在更正導致卸載問題的錯誤時繼續其正常作業。如需 DEFINE LOG 指令的相關資訊，請參閱 [DEFINE LOG](#)。

作用中日誌已滿

症狀

當作用中日誌已滿時，佇列管理程式會中止處理，直到卸載處理程序完成為止。如果卸載處理在作用中日誌已滿時停止，佇列管理程式可能會異常結束。需要先採取更正動作，才能重新啟動佇列管理程式。

IBM MQ 會發出下列 CSQJ111A 訊息：

```
CSQJ111A +CSQ1 OUT OF SPACE IN ACTIVE LOG DATA SETS
```

且卸載處理程序已啟動。然後，佇列管理程式會中止處理，直到卸載處理程序完成為止。

系統動作

在回復正常 IBM MQ 處理之前，IBM MQ 會等待可用的作用中日誌資料集。無法使用 QUIESCE 或 FORCE 正常關機，因為關機順序需要日誌空間來記錄與關機相關的系統事件 (例如，檢查點記錄)。如果卸載處理在作用中日誌已滿時停止，則佇列管理程式會因 X'6C6' 異常終止而停止；在此情況下，重新啟動需要特別注意。如需詳細資料，請參閱第 95 頁的『[z/OS 上的問題判斷](#)』。

系統程式設計師動作

在重新啟動佇列管理程式之前，您可以提供其他作用中日誌資料集。這允許 IBM MQ 在更正導致卸載處理程序問題的錯誤時繼續其正常作業。若要新增作用中日誌資料集，請在佇列管理程式非作用中時使用變更日誌庫存公用程式 (CSQJU003)。如需新增作用中日誌資料集的詳細資料，請參閱 [變更 BSDS](#)。

請考量將日誌數目增加下列項目：

1. 請確定佇列管理程式已停止，然後使用 Access Method Services DEFINE 指令來定義新的作用中日誌資料集。
2. 使用變更日誌庫存公用程式 (CSQJU003) 在 BSDS 中定義新的作用中日誌資料集。
3. 使用 [DEFINE LOG](#) 指令動態新增其他日誌資料集。

當您重新啟動佇列管理程式時，卸載會在啟動期間自動啟動，且在強制停止 IBM MQ 時進行中的任何工作都會回復。

操作員動作

檢查卸載處理程序是否正在等待磁帶機。如果是的話，請裝載磁帶。如果您無法裝載磁帶，請使用 z/OS CANCEL 指令強制 IBM MQ 停止。

保存日誌問題

請利用這個主題來調查及解決保存日誌的問題。

這個主題涵蓋下列保存日誌問題：

- [第 221 頁的『配置問題』](#)
- [第 222 頁的『卸載作業異常終止』](#)
- [第 223 頁的『DASD 空間不足，無法完成卸載處理』](#)
- [第 223 頁的『重新啟動 IBM MQ 時，保存資料集上的讀取 I/O 錯誤』](#)

配置問題

症狀

IBM MQ 問題訊息: [CSQJ103E](#)

```
CSQJ103E +CSQ1 LOG ALLOCATION ERROR DSNAME=dsname,  
ERROR STATUS=eeeeiiii, SMS REASON CODE=sss
```

z/OS 動態配置提供 ERROR STATUS。如果配置是用於卸載處理，則也會顯示下列訊息: [CSQJ115E](#):

```
CSQJ115E +CSQ1 OFFLOAD FAILED, COULD NOT ALLOCATE AN ARCHIVE  
DATA SET
```

系統動作

會執行下列動作:

- 如果回復需要輸入，且回復未順利完成，則佇列管理程式會異常結束。
- 如果作用中日誌已滿，且卸載作業已排程但未完成，則卸載作業會在下一次觸發時重試。作用中日誌不會重複使用尚未保存的資料集。

系統程式設計師動作

無。

操作員動作

請檢查配置錯誤碼，以找出問題的原因，並更正它。請確保磁碟機可用，並重新啟動或等待重試卸載作業。如果已針對保存日誌資料集寫入 DFP/DFSMS ACS 使用者結束程式過濾器，請小心，因為當佇列管理程式嘗試讀取保存日誌資料集時，這可能會導致裝置配置錯誤。

卸載作業異常終止

症狀

寫入 I/O 錯誤不會發出特定的 IBM MQ 訊息。

只會出現 z/OS 錯誤回復程式訊息。如果您收到 IBM MQ message [CSQJ128E](#)，則卸載作業已異常結束。

系統動作

會執行下列動作:

- 卸載作業會放棄輸出資料集; 在 BSDS 中不會產生任何項目。
- 卸載作業會動態配置新的保存檔，並從先前觸發它的點重新啟動卸載。
- 如果新資料集發生錯誤:
 - 在雙重保存模式中，會產生訊息 [CSQJ114I](#)，且卸載處理程序會變更為單一模式:

```
CSQJ114I +CSQ1 ERROR ON ARCHIVE DATA SET, OFFLOAD  
CONTINUING WITH ONLY ONE ARCHIVE DATA SET BEING  
GENERATED
```

- 在單一保存模式中，會放棄輸出資料集。下次觸發卸載處理程序時，會再次嘗試處理此 RBA 範圍。
- 作用中日誌不會折返; 如果沒有其他作用中日誌，資料不會遺失。

系統程式設計師動作

無。

操作員動作

請確定已在可靠的磁碟機和控制裝置上配置卸載作業。

DASD 空間不足，無法完成卸載處理

症狀

將作用中日誌資料集卸載至 DASD 時，程序非預期地終止。IBM MQ issues message CSQJ128E:

```
CSQJ128E +CSQ1 LOG OFF-LOAD TASK FAILED FOR ACTIVE LOG nnnnn
```

錯誤之前有 z/OS 訊息 IEC030I、IEC031I 或 IEC032I。

系統動作

IBM MQ 會取消配置發生錯誤的資料集。如果 IBM MQ 以雙重保存模式執行，則 IBM MQ 會變更為單一保存模式，並繼續卸載作業。如果無法在單一保存模式下完成卸載作業，則無法卸載作用中日誌資料集，且作用中日誌資料集的狀態維持「不可重複使用」。下次觸發卸載作業時，會再次嘗試處理已放棄之作用中日誌資料集的 RBA 範圍。

系統程式設計師動作

這些症狀最可能的原因如下：

- 保存日誌資料集的大小太小，無法在卸載處理期間包含作用中日誌資料集的資料。已使用所有次要空間配置。此條件通常隨附 z/OS message IEC030I。此訊息中的回覆碼可能會提供這些症狀原因的進一步說明。

解決問題

1. 發出指令 `CANCEL queue_manager name` 以取消佇列管理程式工作
2. 增加保存日誌資料集 (在 CSQ6ARVP 系統參數中) 的主要或次要配置 (或兩者)，或減少作用中日誌資料集的大小。

如果要卸載的資料很大，您可以裝載另一個線上儲存磁區，或使其可供 IBM MQ 使用。

3. 重新啟動佇列管理程式。

- 正在寫入保存資料集的 DASD 磁區上所有可用空間已耗盡。此條件通常隨附 z/OS message IEC032I。

若要解決此問題，請在 DASD 磁區上提供更多可用空間，或讓另一個線上儲存磁區可供 IBM MQ 使用。

- 保存日誌資料集的主要空間配置 (如 CSQ6ARVP 系統參數中所指定) 太大，無法配置給任何可用的線上 DASD 裝置。此條件通常隨附 z/OS message IEC032I。

若要解決此問題，請在 DASD 磁區上提供更多可用空間，或讓另一個線上儲存磁區可供 IBM MQ 使用。如果無法這樣做，您必須調整 CSQ6ARVP 系統參數中 PRIQTY 的值，以減少主要配置。(如需詳細資料，請參閱 [使用 CSQ6ARVP](#)。)

註: 如果您減少主要配置，則可能必須增加次要空間配置的大小，以避免未來異常終止。

操作員動作

無。

重新啟動 IBM MQ 時，保存資料集上的讀取 I/O 錯誤

症狀

不會發出特定的 IBM MQ 訊息; 只會出現 z/OS 錯誤回復程式訊息。

系統動作

這取決於第二個副本是否存在:

- 如果第二個副本存在，則會配置並使用它。
- 如果第二個副本不存在，則重新啟動失敗。

系統程式設計師動作

無。

操作員動作

請嘗試使用不同的磁碟機來重新啟動。

z/OS **BSDS 問題**

請利用這個主題來調查及解決 BSDS 的問題。

如需引導資料集 (BSDS) 的背景資訊，請參閱 [在 z/OS 上規劃 IBM MQ 環境](#)。

本主題說明下列 BSDS 問題：

- [第 224 頁的『開啟 BSDS 時發生錯誤』](#)
- [第 225 頁的『日誌內容不同意 BSDS 資訊』](#)
- [第 225 頁的『BSDS 的兩個副本都已損壞』](#)
- [第 225 頁的『不相等時間戳記』](#)
- [第 226 頁的『不同步』](#)
- [第 227 頁的『I/O 錯誤』](#)
- [第 227 頁的『日誌範圍問題』](#)

一般而言，BSDS 有兩個副本，但如果其中一個已損壞，IBM MQ 會立即變更為單一 BSDS 模式。不過，在重新啟動之前，必須先回復 BSDS 的損壞副本。如果您處於單一模式並損壞 BSDS 的唯一副本，或者如果您處於雙重模式並損壞這兩個副本，請使用 [回復 BSDS](#) 中說明的程序。

本節涵蓋啟動時可能發生的部分 BSDS 問題。這裡未涵蓋的問題包括：

- RECOVER BSDS 指令錯誤 (訊息 CSQJ301E - CSQJ307I)
- 變更日誌庫存公用程式錯誤 (訊息 CSQJ123E)
- 卸載處理程序傾出 BSDS 備份時發生錯誤 (訊息 CSQJ125E)

開啟 BSDS 時發生錯誤

症狀

IBM MQ 會發出下列訊息：

```
CSQJ100E +CSQ1 ERROR OPENING BSDSn DSNAME=..., ERROR STATUS=eeii
```

其中 *eeii* 是 VSAM 回覆碼。如需 VSAM 程式碼的相關資訊，請參閱 *DFSMS/MVS* 資料集的巨集指示文件。

系統動作

在系統起始設定期間，會終止啟動。

在 RECOVER BSDS 指令期間，系統會以單一 BSDS 模式繼續執行。

系統程式設計師動作

無。

操作員動作

請執行下列步驟：

1. 在 BSDS 的兩個副本上執行列印日誌對映公用程式，並比較清單以判定哪個副本正確或最新。
2. 重新命名有問題的資料集，並定義其取代項目。
3. 使用「存取方法服務」，將精確資料集複製到取代資料集。

4. 重新啟動佇列管理程式。

日誌內容不同意 BSDS 資訊

症狀

IBM MQ 會發出下列訊息：

```
CSQJ102E +CSQ1 LOG RBA CONTENT OF LOG DATA SET DSNAME=... ,  
          STARTRBA=... , ENDRBA=... ,  
          DOES NOT AGREE WITH BSDS INFORMATION
```

此訊息指出未正確使用變更日誌庫存公用程式，或正在使用舊版資料集。

系統動作

佇列管理程式啟動處理程序已終止。

系統程式設計師動作

無。

操作員動作

執行列印日誌對映公用程式及變更日誌庫存公用程式，以列印並更正 BSDS 的內容。

BSDS 的兩個副本都已損壞

症狀

IBM MQ 會發出下列訊息：

```
CSQJ107E +CSQ1 READ ERROR ON BSDS  
          DSNAME=... ERROR STATUS=0874  
CSQJ117E +CSQ1 REG8 INITIALIZATION ERROR READING BSDS  
          DSNAME=... ERROR STATUS=0874  
CSQJ119E +CSQ1 BOOTSTRAP ACCESS INITIALIZATION PROCESSING FAILED
```

系統動作

佇列管理程式啟動處理程序已終止。

系統程式設計師動作

請執行下列步驟：

1. 重新命名資料集，並定義其取代項目。
2. 找出與最新保存日誌資料集相關聯的 BSDS，並將它複製到取代資料集。
3. 使用列印日誌對映公用程式來列印取代 BSDS 的內容。
4. 使用列印日誌記錄公用程式來列印取代 BSDS 中遺漏的作用中日誌資料集摘要報告，以及建立 RBA 範圍。
5. 使用變更日誌庫存公用程式來更新取代 BSDS 中遺漏的作用中日誌資料集庫存。
6. 如果已使用雙重 BSDS 資料集，請將更新的 BSDS 複製到 BSDS 的第二個副本。
7. 重新啟動佇列管理程式。

操作員動作

無。

不平等時間戳記

症狀

IBM MQ 會發出下列訊息：


```
CSQJ120E +CSQ1 DUAL BSDS DATA SETS HAVE UNEQUAL TIME STAMPS,  
SYSTEM BSDS1=...,BSDS2=...,  
UTILITY BSDS1=...,BSDS2=...
```

可能的原因如下:

- 已還原 BSDS 的一個副本。所還原 BSDS 的所有相關資訊都是舊版。舊版 BSDS 具有較早的時間戳記。
- 其中一個包含 BSDS 的磁區已還原。還原磁區的所有相關資訊都是舊版。如果磁區包含任何作用中日誌資料集或 IBM MQ 資料，則它們也是舊版。舊版磁區具有較早的時間戳記。
- 雙重記載已降級為單一記載，且您正在嘗試在未回復損壞日誌的情況下啟動。
- 在更新 BSDS 的一個副本之後，但在更新第二個副本之前，佇列管理程式已異常終止。

系統動作

IBM MQ 會嘗試使用較新的副本來重新同步化 BSDS 資料集。如果失敗，則會終止佇列管理程式啟動。

系統程式設計師動作

無。

操作員動作

如果自動重新同步失敗，請執行下列步驟:

1. 在 BSDS 的兩個副本上執行列印日誌對映公用程式，並比較清單以判斷哪個副本正確或最新。
2. 重新命名舊版資料集，並定義其取代項目。
3. 使用「存取方法服務」，將良好資料集複製到取代資料集。
4. 如果適用的話，請判斷包含舊版 BSDS 的磁區是否已還原。如果已還原，則該磁區上的所有資料(例如作用中日誌資料)也會是舊版。

如果還原的磁區包含作用中日誌資料，且您在個別磁區上使用雙重作用中日誌，則需要將現行版本的作用中日誌複製到舊版日誌資料集。如需如何執行此動作的詳細資料，請參閱 [回復日誌](#)。

不同步

症狀

在佇列管理程式起始設定期間，IBM MQ 會發出下列訊息:

```
CSQJ122E +CSQ1 DUAL BSDS DATA SETS ARE OUT OF SYNCHRONIZATION
```

兩個資料集的系統時間戳記相同。如果在使用變更日誌庫存公用程式時發生操作員錯誤，則可能存在差異。(例如，變更日誌庫存公用程式只在一個副本上執行。)變更日誌庫存公用程式會在 BSDS 控制記錄啟動時設定專用時間戳記，並在結束時設定關閉旗標。IBM MQ 會檢查變更日誌庫存公用程式時間戳記，如果它們不同，或它們相同但未設定一個關閉旗標，則 IBM MQ 會比較 BSDS 的副本。如果副本不同，則會發出 CSQJ122E。

如果指定兩個輸入 BSDS，且發現兩個 BSDS 副本之間的記錄不同，則 BSDS 轉換公用程式也會發出此訊息。如果佇列管理程式在執行 BSDS 轉換公用程式之前異常終止，則會發生此狀況。

系統動作

佇列管理程式啟動或公用程式已終止。

系統程式設計師動作

無。

操作員動作

如果在佇列管理程式起始設定期間發生錯誤，請執行下列步驟:

1. 在 BSDS 的兩個副本上執行列印日誌對映公用程式，並比較清單以判定哪個副本正確或最新。
2. 重新命名有問題的資料集，並定義其取代項目。
3. 使用存取方法服務，將精確資料集複製到取代資料集。
4. 重新啟動佇列管理程式。

如果執行 BSDS 轉換公用程式時發生錯誤，請執行下列步驟：

1. 嘗試重新啟動佇列管理程式並完全關閉它，然後再次嘗試執行 BSDS 轉換公用程式。
2. 如果這無法解決問題，請在 BSDS 的兩個副本上執行列印日誌對映公用程式，並比較清單以判定哪個副本正確或最新。
3. 在重新提交工作之前，請變更用來呼叫 BSDS 轉換公用程式的 JCL，以在 SYSUT1 DD 陳述式中指定現行 BSDS，並移除 SYSUT2 DD 陳述式。

I/O 錯誤

症狀

IBM MQ 變更為單一 BSDS 模式並發出使用者訊息：

```
CSQJ126E +CSQ1 BSDS ERROR FORCED SINGLE BSDS MODE
```

後面接著下列其中一則訊息：

```
CSQJ107E +CSQ1 READ ERROR ON BSDS
          DSNAME=... ERROR STATUS=...

CSQJ108E +CSQ1 WRITE ERROR ON BSDS
          DSNAME=... ERROR STATUS=...
```

系統動作

BSDS 模式從雙重變更為單一。

系統程式設計師動作

無。

操作員動作

請執行下列步驟：

1. 使用「存取方法服務」來重新命名或刪除損壞的 BSDS，以及定義與發生錯誤的 BSDS 同名的新 BSDS。在 thlqual.SCSQPROC 的工作 CSQ4BREC 中可以找到控制陳述式範例。
2. 發出 IBM MQ 指令 RECOVER BSDS，以在新配置的資料集中建立良好 BSDS 的副本，並恢復雙重 BSDS 模式。另請參閱 [回復 BSDS](#)。

日誌範圍問題

症狀

IBM MQ 已在讀取自己的日誌時發出訊息 [CSQJ113E](#)，或在讀取佇列共用群組中佇列管理程式的日誌時發出訊息 [CSQJ133E](#) 或 [CSQJ134E](#)。當您沒有重新啟動佇列管理程式或回復 CF 結構所需的保存日誌時，可能會發生這種情況。

系統動作

視正在讀取的日誌記錄及原因而定，要求者可能會異常結束，原因碼為 X'00D1032A'。

系統程式設計師動作

執行列印日誌對映公用程式 (CSQJU004)，以判斷錯誤的原因。當發出訊息 CSQJ133E 或 CSQJ134E 時，請針對訊息中指出的佇列管理程式 BSDS 執行公用程式。

如果您具有：

- 從 BSDS 中刪除具有日誌範圍 (包含訊息中指出的日誌 RBA 或 LRSN) 的項目，以及
- 未刪除或重複使用資料集

您可以使用下列程序，將項目新增回 BSDS：

1. 透過查看 BSDS 內容的舊副本，或針對 BSDS 備份執行 CSQJU004，來識別包含所需 RBA 或 LRSN 的資料集。
2. 使用變更日誌庫存公用程式 (CSQJU003) 將資料集新增回 BSDS。
3. 重新啟動佇列管理程式。

如果已刪除保存日誌資料集，您將無法回復需要保存日誌的頁集或 CF 結構。識別佇列管理程式需要讀取日誌記錄的原因，然後根據受影響的頁面集或 CF 結構採取下列其中一個動作。

頁面集

在佇列管理程式重新啟動的回復階段期間，訊息 CSQJ113E 指出需要日誌才能執行媒體回復，以讓頁面設定保持最新。

在佇列管理程式重新啟動期間，查看針對每一個頁集發出的 CSQI1049I 訊息中的媒體回復 RBA，以識別需要已刪除日誌資料集才能進行媒體回復的頁集，然後執行下列動作。

• 頁集零

您可以使用下列程序來回復頁集零上的物件。



小心：當您執执行程序時，所有其他頁集中的所有資料都會遺失。

1. 使用 CSQUTIL 公用程式的函數 SDEFS 來產生 IBM MQ DEFINE 指令的檔案。
2. 使用 CSQUTIL 將頁集格式化為零，然後依照下一節的說明重新定義其他頁集。
3. 重新啟動佇列管理程式。
4. 使用 CSQUTIL，利用公用程式在步驟 1 中產生的 DEFINE 指令來重新定義物件。

• 頁集 1-99

請使用下列程序來重新定義頁集。



小心：當您執行此作業時，頁集上的任何資料都會遺失。

1. 如果您可以存取頁集而沒有任何 I/O 錯誤，請使用 CSQUTIL 公用程式搭配指令 FORMAT TYPE (NEW) 來重新格式化頁集。
2. 如果存取頁集時發生 I/O 錯誤，請刪除頁集並重建它。

如果您想要頁集與之前的大小相同，請使用指令 LISTCAT ENT (*dsname*) ALLOC 來取得現有的空間配置，並在 z/OS DEFINE CLUSTER 指令中使用這些配置。

使用 CSQUTIL 公用程式搭配指令 FORMAT TYPE (NEW) 來格式化新頁集。

3. 重新啟動佇列管理程式。您可能必須採取某些動作，例如重設通道或解決不確定的通道。

CF 結構

在 CF 結構回復期間，訊息 CSQJ113E、CSQJ133E 或 CSQJ134E 指出回復結構所需的日誌在佇列共用群組的至少一個成員上無法使用。

視受影響的結構而定，採取下列其中一個動作：

應用程式 CF 結構

發出指令 RECOVER CFSTRUCT (結構名稱) TYPE (PURGE)。

此處理程序會清空結構，因此結構上的任何訊息都會遺失。

CSQSYSAPPL 結構

請聯絡 IBM 支援中心。

管理結構

此結構是使用自每一個佇列管理程式上的最後一個檢查點 (應該位於作用中日誌中) 之後的日誌資料來重建。

如果您在管理結構回復期間收到此錯誤, 請聯絡 IBM 支援中心, 因為這指出作用中日誌無法使用。

回復頁面集或 CF 結構之後, 請執行日誌、BSDS、頁面集及 CF 結構的備份。

若要防止此問題再次發生, 請增加:

- 要較長的保存日誌保留 (ARCRETN) 值, 以及
- 增加 CF 結構備份的頻率。

回復 CF 結構

在概念上, 會從 IBM MQ 日誌中讀取先前所備份 CF 結構中的資料; 從備份中讀取日誌, 並將任何變更重新套用到還原的結構。

關於這項作業

要使用的日誌範圍是從要回復之每一個結構的最新備份到現行時間。日誌範圍由日誌範圍序號 (LRSN) 值識別。

LRSN 使用「儲存庫時鐘值」的六個最高有效位數。

請注意, 如果您尚未執行結構的備份, 則會讀取整個日誌 (回到建立結構的時間)。

程序

1. 請檢查是否已讀取佇列共用群組 (QSG) 中每一個佇列管理程式的日誌, 以取得此 LRSN 範圍內的記錄。請注意, 日誌是向後讀取。
2. 檢查是否已建置每一個要回復之結構的變更清單。
3. 會讀取連結機能 (CF) 結構備份中的資料, 並還原資料。

例如, 如果已在佇列管理程式 A 上完成備份, 且正在佇列管理程式 B 上執行回復, 則佇列管理程式 B 會從佇列管理程式讀取日誌 A 以還原結構。

當讀取 CF 結構的備份開始時, 會啟動內部作業, 以取得該結構的已還原資料, 並將它與從日誌讀取的變更合併。

4. 請檢查正在還原的每一個結構是否繼續執行處理程序。

範例

在下列範例中, 已發出指令 RECOVER CFSTRUCT (APP3), 並產生下列訊息:

```
04:00:00 CSQE132I CDL2 CSQERRPB Structure recovery started, using log range from
LRSN=CC56D01026CC
to LRSN=CC56DC368924
This is the start of reading the logs backwards from each qmgr in the queue sharing group from
the time
of failure to the to the structure backup. The LRSN values give the ranges being used.
Log records for all structures (just one structure in this example) being recovered are
processed at the same time.

04:02:00 CSQE133I CDL2 CSQERPLS Structure recovery reading log backwards, LRSN=CC56D0414372
This message is produced periodically to show the process

04:02:22 CSQE134I CDL2 CSQERRPB Structure recovery reading log completed
The above process of replaying the logs backwards has finished,

04:02:22 CSQE130I CDL2 CSQERCF2 Recovery of structure APP3 started, using CDL1 log range
from RBA=000EE86D902E to RBA=000EF5E8E4DC
The task to process the data for APP3 has been started. The last backup of CF structure
APP3 was done on CDL1 within the given RBA range, so this log range has to be read.
```

04:02:29 CSQE131I CDL2 CSQERCF2 Recovery of structure APP3 completed
The data merge has completed. The structure is recovered.

z/OS 頁集問題

請利用這個主題來調查及解決頁集的問題。

本主題涵蓋您可能會遇到的頁面集問題:

- [第 230 頁的『頁集 I/O 錯誤』](#) 說明頁集損壞時所發生的情況。
- [第 230 頁的『頁集已滿』](#) 說明頁面集上沒有足夠空間可供任何其他 MQI 作業使用時所發生的情況。

頁集 I/O 錯誤

問題

頁集有 I/O 錯誤。

症狀

發出此訊息:

```
CSQP004E +CSQ1 csect-name I/O ERROR STATUS ret-code  
PSID psid RBA rba
```

系統動作

佇列管理程式異常終止。

系統程式設計師動作

無。

操作員動作

修復 I/O 錯誤原因。

如果沒有任何頁集損壞，請重新啟動佇列管理程式。IBM MQ 會自動從日誌將頁面集還原至一致狀態。

如果一個以上頁集已損壞:

1. 重新命名損壞的頁集並定義取代頁集。
2. 將最新備份頁面集複製到取代頁面集。
3. 重新啟動佇列管理程式。IBM MQ 會自動套用日誌中的任何必要更新項目。

如果頁集零無法使用，則無法重新啟動佇列管理程式。如果其中一個其他頁集無法使用，您可以在佇列管理程式啟動 JCL 程序中註銷頁集 DD 陳述式。這可讓您延遲回復毀損的頁集，讓其他使用者能夠繼續存取 IBM MQ。

當您將頁面集新增回 JCL 程序時，系統重新啟動會從從 JCL 移除頁面集的點到日誌結尾讀取日誌。如果已記載大量資料，則此程序可能需要很長時間。

MQRC_PAGESET_ERROR 的原因碼會傳回給嘗試存取定義在無法使用之頁集上的佇列的任何應用程式。

當您還原毀損的頁集時，請還原其相關聯的 DD 陳述式，並重新啟動佇列管理程式。

只有在所有日誌資料集都可用時，才能執行這裡說明的操作員動作。如果日誌資料集遺失或損壞，請參閱[如果日誌資料集遺失則重新啟動](#)。

頁集已滿

問題

頁集上沒有足夠的空間可用於下列其中一項:

- 要完成的 MQPUT 或 MQPUT1 呼叫
- 要完成的物件操作指令 (例如, DEFINE QLOCAL)
- 要完成之動態佇列的 MQOPEN 呼叫數

症狀

要求失敗, 原因碼為 MQRC_STORAGE_MEDIUM_FULL。佇列管理程式無法完成要求, 因為頁集上剩餘的空間不足。

即使頁面集 expand 屬性設為 EXPAND (USER), 也可能發生原因碼 MQRC_STORAGE_MEDIUM_FULL。在將原因碼 MQRC_STORAGE_MEDIUM_FULL 傳回至應用程式碼之前, 佇列管理程式會嘗試展開頁集, 然後重試 API 要求。在大量載入的系統上, 在重試 API 之前, 其他 IO 作業可能可以使用擴充儲存體。請參閱 [管理頁集](#)。

此問題的原因可能是訊息在傳輸佇列上累計, 因為它們無法傳送至另一個系統。

系統動作

會封鎖使用此頁集的進一步要求, 直到移除足夠的訊息或刪除物件以騰出空間給新的送入要求為止。

操作員動作

使用 IBM MQ 指令 DISPLAY USAGE PSID (*) 來識別哪個頁集已滿。

系統程式設計師動作

您可以放大所涉及的頁集, 或透過將佇列移至另一個頁集來減少該頁集上的負載。如需這些作業的相關資訊, 請參閱 [管理頁面集](#)。如果問題的原因是訊息累積在傳輸佇列上, 請考慮啟動分散式佇列來傳輸訊息。

連結機能及 Db2 問題

請利用這個主題來調查及解決連結機能及 Db2 的問題。

本節涵蓋您使用連結機能及 Db2 時可能遇到的問題:

- [第 231 頁的『儲存中等滿』](#)
- [第 232 頁的『Db2 系統失敗』](#)
- [第 232 頁的『Db2 資料共用群組失敗』](#)
- [第 233 頁的『Db2 且連結機能失敗』](#)

儲存中等滿

問題

連結機能結構已滿。

症狀

如果佇列結構已滿, 則會將回覆碼 MQRC_STORAGE_MEDIUM_FULL 傳回給應用程式。

如果管理結構已滿, 則確切症狀取決於發生錯誤的處理程序, 其範圍可能從沒有回應到 CMDSCOPE (GROUP) 指令, 到在確定處理期間發生問題導致佇列管理程式失敗。

系統程式設計師動作

您可以使用 IBM MQ 來禁止對結構中部分佇列執行 MQPUT 作業, 以防止應用程式寫入更多訊息、啟動更多應用程式以從佇列取得訊息, 或靜止將訊息放入佇列的部分應用程式。

或者, 您可以使用 XES 機能來就地變更結構大小。下列 z/OS 指令會變更結構的大小:

```
SETXCF START,ALTER,STRNAME= structure-name,SIZE= newsize
```

其中 *newsiz*e 是小於在 CFRM 原則上為結構指定的 MAXSIZE 值, 但大於現行連結機能大小的值。

您可以使用 DISPLAY CFSTATUS 指令來監視連結機能結構的使用率。

Db2 系統失敗

如果 IBM MQ 所連接的 Db2 子系統失敗，IBM MQ 會嘗試重新連接至子系統，並繼續工作。如果您在 CSQ6SYSP 系統參數模組的 QSGDATA 參數中指定了 Db2 群組連接名稱，則 IBM MQ 會重新連接至另一個作用中 Db2，它是與失敗 Db2 相同的資料共用群組的成員 (如果在相同的 z/OS 映像檔上可用的話)。

當 IBM MQ 未連接至 Db2 時，有些佇列管理程式作業無法運作。它們是：

- 刪除共用佇列或群組物件定義。
- 在共用佇列或群組物件定義上變更或發出 MQSET。共用佇列上 MQSET 的限制表示觸發或產生效能事件之類的作業無法正確運作。
- 定義新的共用佇列或群組物件。
- 顯示共用佇列或群組物件。
- 針對共用通道啟動、停止或其他動作。
- 第一次透過發出 MQOPEN 來開啟共用佇列時，從 Db2 讀取共用佇列定義。

對於共用佇列，其他 IBM MQ API 作業會繼續正常運作，並且可以針對從 GROUP 物件建置的佇列管理程式專用版本 (COPY 物件) 執行所有 IBM MQ 作業。同樣地，任何正在正常執行的共用通道都會繼續執行，直到它們結束或發生錯誤 (當它們進入重試狀態時)。

當 IBM MQ 重新連接至 Db2 時，會在佇列管理程式與 Db2 之間執行重新同步。這包括通知佇列管理程式在其斷線時已在 Db2 中定義的新物件 (其他佇列管理程式可能已能夠透過其他 Db2 子系統在其他 z/OS 映像檔上繼續正常運作)，以及更新 Db2 中已變更之共用佇列的物件屬性。任何處於重試狀態的共用通道都會回復。

如果 Db2 失敗，則在失敗時它可能擁有 Db2 資源的鎖定。在某些情況下，這可能會讓其他未受其他影響的佇列管理程式無法使用某些 IBM MQ 物件。若要解決此問題，請重新啟動失敗的 Db2，以便它可以執行回復處理並釋放鎖定。

Db2 資料共用群組失敗

如果整個 Db2 資料共用群組失敗，則回復可能是失敗的時間或前一個復原點。

在回復至失敗點的情況下，IBM MQ 會在已回復 Db2 時重新連接，重新同步處理程序會取得位置，並回復一般佇列管理程式功能。

不過，如果 Db2 回復至前一個復原點，則連結機能結構中的實際佇列與那些佇列的 Db2 視圖之間可能會不一致。例如，在復原點，Db2 會回復至已存在且已刪除的佇列，且其在連結機能結構中的位置已由現在包含訊息的新佇列定義所重複使用。

如果您發現自己處於此狀況，則必須停止佇列共用群組中的所有佇列管理程式，清除連結機能結構，然後重新啟動佇列管理程式。然後，您必須使用 IBM MQ 指令來定義任何遺漏的物件。如果要執行這個動作，請使用下列程序：

1. 在公用程式模式下啟動 Db2，或變更安全設定檔，以防止 IBM MQ 重新連接至 Db2。
2. 如果您在共用佇列上有任何重要訊息，則可以使用 CSQUTIL 公用程式的 COPY 函數來卸載它們，但這可能無法運作。
3. 終止所有佇列管理程式。
4. 使用下列 z/OS 指令來清除所有結構：

```
SETXCF FORCE,STRUCTURE,STRNAME=
```

5. 將 Db2 還原至歷程復原點。
6. 重新建立對 Db2 的佇列管理程式存取權。
7. 重新啟動佇列管理程式。

8. 從備份副本回復 IBM MQ 定義。
9. 將任何卸載的訊息重新載入至共用佇列。

當佇列管理程式重新啟動時，它們會嘗試重新同步化本端 COPY 物件與 Db2 GROUP 物件。這可能會導致 IBM MQ 嘗試執行下列動作：

- 為存在於 Db2 已回復至的時間點的舊 GROUP 物件建立 COPY 物件。
- 刪除自復原點 Db2 已回復至且資料庫中不存在之 GROUP 物件的 COPY 物件。

使用 NOPURGE 選項嘗試 DELETE of COPY 物件，因此對於仍在這些 COPY 佇列中有訊息的佇列管理程式，它會失敗。

Db2 且連結機能失敗

如果連結機能失敗，佇列管理程式可能會失敗，且如果 Db2 使用此連結機能，它也會失敗。

使用 Db2 回復程序來回復 Db2。重新啟動 Db2 之後，您可以重新啟動佇列管理程式。CF 管理結構也會失敗，但會透過重新啟動佇列共用群組內的所有佇列管理程式來重建此結構。

如果連結機能內的單一應用程式結構失敗，則對佇列管理程式的影響取決於佇列管理程式的層次，以及失敗 CF 結構的 CFLEVEL：

- 如果 CF 應用程式結構是 CFLEVEL (3) 或更高版本，且 RECOVER 設為 YES，則除非您對將執行回復的佇列管理程式發出 MQSC RECOVER CFSTRUCT 指令來回復 CF 結構，否則它將無法使用。您可以指定要回復的單一 CF 結構，也可以同時回復數個 CF 結構。執行回復的佇列管理程式會使用 Db2 中的資料及引導資料集，在所有其他佇列管理程式日誌上尋找相關備份。佇列管理程式會在佇列共用群組中以正確的時間順序重播這些備份，從最後一個備份之前到失敗點。如果可回復的應用程式結構失敗，則在回復結構之前，會阻止任何進一步的應用程式活動。如果管理結構也失敗，則必須先啟動佇列共用群組中的所有佇列管理程式，才能發出 RECOVER CFSTRUCT 指令。在回復失敗的 CF 結構期間，所有佇列管理程式都可以繼續使用其他 CF 結構中的本端佇列及佇列。
- 如果 CF 應用程式結構是 CFLEVEL (3) 或更高版本，且 RECOVER 設為 NO，則在結構中定義的佇列上執行的下一個 MQOPEN 要求會自動重新配置結構。所有訊息都會遺失，因為結構只能包含非持續訊息。
- 如果 CF 應用程式結構的 CFLEVEL 小於 3，則佇列管理程式會失敗。在佇列管理程式重新啟動時，同層級回復會嘗試連接至結構，偵測到結構失敗，並配置新版本的結構。在受連結機能失敗影響的 CF 結構中，共用佇列上的所有訊息都會遺失。

從 IBM WebSphere MQ 7.1 開始，佇列共用群組中的佇列管理程式已能夠容忍失去與連結機能結構的連線功能，而不會失敗。如果結構經歷連線失敗，則會嘗試在另一個連結機能中重建結構，並具有更好的連線功能，以儘快重新取得共用佇列的存取權。

長時間執行工作單元的問題

請利用這個主題來調查及解決長時間執行工作單元的問題。

本主題說明在重新啟動期間遇到長時間執行的工作單元時要執行的動作。在此環境定義中，這表示工作單元已長時間處於作用中(可能是幾天，甚至是幾週)，因此工作單元的原始 RBA 不在現行作用中日誌的範圍內。這表示重新啟動可能需要很長時間，因為必須讀取與工作單元相關的所有日誌記錄，這可能涉及讀取保存日誌。

在重新啟動期間找到舊工作單元

問題

在重新啟動期間偵測到其原始 RBA 早於最舊作用中日誌的工作單元。

症狀

IBM MQ 會發出下列訊息：

```
CSQR020I +CSQ1 OLD UOW FOUND
```

系統動作

會顯示工作單元的相關資訊，並發出 CSQR021D 訊息，要求操作員回應。

系統程式設計師動作

無。

操作員動作

決定是否確定工作單元。如果您選擇不確定工作單元，則由正常重新啟動回復處理程序處理。因為工作單元是舊的，這可能涉及使用保存日誌，因此需要較長時間才能完成。

IMS 相關問題

請利用這個主題來調查及解決 IMS 和 IBM MQ 的問題。

本主題包括您在 IMS 環境中可能遇到的下列問題的計劃：

- [第 234 頁的『IMS 無法連接至 IBM MQ』](#)
- [第 234 頁的『IMS 應用程式問題』](#)
- [第 235 頁的『IMS 無法運作』](#)

IMS 無法連接至 IBM MQ

問題

IMS 配接器無法連接至 IBM MQ。

症狀

IMS 仍然有效。IMS 配接器會針對控制區域連接發出下列訊息：

- CSQQ001I
- CSQQ002E
- CSQQ003E
- CSQQ004E
- CSQQ005E
- CSQQ007E

如需詳細資料，請參閱 [IBM MQ for z/OS 訊息](#)，[完成](#)，及[原因碼](#) 文件。

當 IMS 配接器無法連接時，如果 IMS 應用程式嘗試存取 IBM MQ，它可能會收到完成碼和原因碼，或異常終止。這取決於 IMS PROCLIB 的 SSM 成員中 REO 選項的值。

系統動作

IMS 訊息中也會報告所有連線錯誤 DFS3611.

系統程式設計師動作

無。

操作員動作

分析並更正問題，然後使用 IMS 指令重新啟動連線：

```
/START SUBSYS subsysname
```

IMS 要求配接器解決不確定的回復單元。

IMS 應用程式問題

問題

IMS 應用程式異常終止。

症狀

下列訊息會傳送至使用者的終端機：

```
DFS555I TRANSACTION tran-id ABEND abcode  
MSG IN PROCESS: message data:
```

其中 *tran-id* 代表任何異常終止的 IMS 交易，*abcode* 是異常終止碼。

系統動作

IMS 要求配接器解決回復單元。IMS 仍然連接至 IBM MQ。

系統程式設計師動作

無。

操作員動作

如 IMS 主要終端機上訊息 DFS554A 所示。

IMS 無法運作

問題

IMS 無法運作。

症狀

可能有多個症狀：

- IMS 等待或迴圈

IBM MQ 無法在 IMS 中偵測到等待或迴圈，因此您必須尋找等待或迴圈的原點。這可以是 IMS、IMS 應用程式或 IMS 配接器。

- IMS 異常終止。

– 如需相關資訊，請參閱手冊 *IMS/ESA* 訊息與訊息碼 及 *IMS/ESA* 失敗分析結構表格。

– 如果執行緒在 IMS 終止時連接至 IBM MQ，則 IBM MQ 會發出訊息 CSQ3201E。這則訊息指出已執行 IBM MQ 作業結束 (EOT) 常式來清除及切斷任何連接的執行緒。

系統動作

IBM MQ 會偵測 IMS 錯誤，並：

- 取消進行中的工作。
- 重新連接 IMS 時，儲存要解決的不確定回復單元。

系統程式設計師動作

無。

操作員動作

解決並更正導致 IMS 異常終止的問題，然後執行 IMS 緊急重新啟動。緊急重新啟動：

- 取消變更 IMS 資源的進行中交易。
- 記住具有 IBM MQ 存取權且可能不確定的交易。

您可能需要使用 IMS 指令重新啟動與 IBM MQ 的連線：

```
/START SUBSYS subsystemname
```

在啟動期間，IMS 會要求配接器解決不確定的回復單元。

z/OS 硬體問題

請利用這個主題作為起點來調查硬體問題。

如果硬體錯誤導致無法讀取資料，則仍然可以使用 媒體回復 技術來回復 IBM MQ：

1. 若要回復資料，您需要資料的備份副本。定期使用 DFDSS 或 Access Method Services REPRO 來製作資料副本。

2. 恢復最新備份副本。
3. 重新啟動佇列管理程式。

您的備份副本越新，您的子系統就可以更快地重新變成可用。

當佇列管理程式重新啟動時，它會使用保存日誌來恢復自取得備份副本以來所做的變更。您必須保留足夠的保存日誌，以啟用 IBM MQ 來完整恢復變更。在有備份副本包含日誌中的所有變更之前，請勿刪除保存日誌。

注意事項

本資訊係針對 IBM 在美國所提供之產品與服務所開發。

在其他國家中，IBM 可能不會提供本書中所提的各項產品、服務或功能。請洽當地 IBM 業務代表，以取得當地目前提供的產品和服務之相關資訊。這份文件在提及 IBM 的產品、程式或服務時，不表示或暗示只能使用 IBM 的產品、程式或服務。只要未侵犯 IBM 的智慧財產權，任何功能相當的產品、程式或服務都可以取代 IBM 的產品、程式或服務。不過，任何非 IBM 的產品、程式或服務，使用者必須自行負責作業的評估和驗證責任。

本文件所說明之主題內容，IBM 可能擁有其專利或專利申請案。提供本文件不代表提供這些專利的授權。您可以書面提出授權查詢，來函請寄到：

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

如果是有關雙位元組 (DBCS) 資訊的授權查詢，請洽詢所在國的 IBM 智慧財產部門，或書面提出授權查詢，來函請寄到：

智慧財產權授權
法務部與智慧財產權法律
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

下列段落不適用於英國，若與任何其他國家之法律條款抵觸，亦不適用於該國： International Business Machines Corporation 只依 "現況" 提供本出版品，不提供任何明示或默示之保證，其中包括且不限於不侵權、可商用性或特定目的之適用性的隱含保證。有些地區在特定交易上，不允許排除明示或暗示的保證，因此，這項聲明不一定適合您。

這項資訊中可能有技術上或排版印刷上的訛誤。因此，IBM 會定期修訂；並將修訂後的內容納入新版中。IBM 隨時會改進及/或變更本出版品所提及的產品及/或程式，不另行通知。

本資訊中任何對非 IBM 網站的敘述僅供參考，IBM 對該網站並不提供任何保證。這些網站所提供的資料不是 IBM 本產品的資料內容，如果要使用這些網站的資料，您必須自行承擔風險。

IBM 得以各種適當的方式使用或散布由您提供的任何資訊，無需對您負責。

如果本程式的獲授權人為了 (i) 在個別建立的程式和其他程式（包括本程式）之間交換資訊，以及 (ii) 相互使用所交換的資訊，因而需要相關的資訊，請洽詢：

IBM Corporation
軟體交互作業能力協調程式，部門 49XA
3605 公路 52 N
Rochester, MN 55901
U.S.A.

在適當條款與條件之下，包括某些情況下（支付費用），或可使用此類資訊。

IBM 基於雙方之 IBM 客戶合約、IBM 國際程式授權合約或任何同等合約之條款，提供本資訊所提及的授權程式與其所有適用的授權資料。

本文件中所含的任何效能資料都是在受管制的環境下判定。因此不同作業環境之下所得的結果，可能會有很大的差異。有些測定已在開發階段系統上做過，不過這並不保證在一般系統上會出現相同結果。甚至有部分的測量，是利用插補法而得的估計值，實際結果可能有所不同。本書的使用者應依自己的特定環境，查證適用的資料。

本文件所提及之非 IBM 產品資訊，取自產品的供應商，或其發佈的聲明或其他公開管道。IBM 並未測試過這些產品，也無法確認這些非 IBM 產品的執行效能、相容性或任何對產品的其他主張是否完全無誤。有關非 IBM 產品的性能問題應直接洽詢該產品供應商。

有關 IBM 未來方針或目的之所有聲明，僅代表 IBM 的目標與主旨，隨時可能變更或撤銷，不必另行通知。

這份資訊含有日常商業運作所用的資料和報告範例。為了要使它們儘可能完整，範例包括個人、公司、品牌和產品名稱。這些名稱全屬虛構，如與實際公司的名稱和住址雷同，純屬巧合。

著作權授權：

本資訊含有原始語言之範例應用程式，用以說明各作業平台中之程式設計技術。您可以基於研發、使用、銷售或散布符合作業平台（撰寫範例程式的作業平台）之應用程式介面的應用程式等目的，以任何形式複製、修改及散布這些範例程式，而不必向 IBM 付費。這些範例並未在所有情況下完整測試。因此，IBM 不保證或暗示這些程式的可靠性、有用性或功能。

若貴客戶正在閱讀本項資訊的電子檔，可能不會有照片和彩色說明。

程式設計介面資訊

程式設計介面資訊 (如果有提供的話) 旨在協助您建立與此程式搭配使用的應用軟體。

本書包含預期程式設計介面的相關資訊，可讓客戶撰寫程式以取得 WebSphere MQ 的服務。

不過，本資訊也可能包含診斷、修正和調整資訊。提供診斷、修正和調整資訊，是要協助您進行應用軟體的除錯。

重要：請勿使用此診斷、修改及調整資訊作為程式設計介面，因為它可能會變更。

商標

IBM、IBM 標誌 [ibm.com](http://www.ibm.com) 是 IBM Corporation 在全球許多適用範圍的商標。IBM 商標的最新清單可在 Web 的 "Copyright and trademark information" www.ibm.com/legal/copytrade.shtml 中找到。其他產品和服務名稱，可能是 IBM 或其他公司的商標。

Microsoft 及 Windows 是 Microsoft Corporation 在美國及/或其他國家或地區的商標。

UNIX 是 The Open Group 在美國及/或其他國家/地區的註冊商標。

Linux 是 Linus Torvalds 在美國及/或其他國家或地區的註冊商標。

本產品包含 Eclipse Project (<http://www.eclipse.org/>) 所開發的軟體。

Java 和所有以 Java 為基礎的商標及標誌是 Oracle 及/或其子公司的商標或註冊商標。



產品編號:

(1P) P/N: