



WebSphere MQ Everyplace 2.0.2 版

目錄

歡迎使用 MQe	1	使用什麼機器	21
MQe 概觀	1	使用哪一個程式碼庫	22
程式碼庫	3	MQe 開發周期	22
2.0.2 版的新功能	4	體驗 MQe	23
使用本資訊中心	4	使用 MQ 與 MQe	23
什麼是 MQe	6	進一步資訊	26
MQe 簡介	6	注意事項 & 商標	27
MQ 系列的 MQe	6	注意事項	27
MQe 可能的用途	7	商標	28
MQe 如何運作	8	名詞解釋	28
MQe SupportPacs	20		
規劃實作	21	索引	33
授權	21		

歡迎使用 MQe

這個產品的完整名稱是 WebSphere® MQ Everyplace® Version 2 Release 0 Modification 1，但是比較常表示為 WebSphere MQ Everyplace 2.0.1 版。

在這份文件中，產品一般簡稱為 *MQe*。

MQe 用於感應器、電話、個人數位助理 (PDA) 和膝上型電腦等輕型裝置以及桌上型電腦的安全傳訊。

程式設計介面

MQe 的應用程式設計介面在本文件中，稱為 *MQe API*。MQe API 支援的 Java™ 及 C 程式設計語言如下：

Java 提供所有 MQe 在第 2 版中功能的存取權。

C 支援有三個版本：

原生 C 程式碼庫

提供 MQe 功能子集的存取權，主要的限制是僅供裝置佇列管理程式使用，所以只能傳送訊息，而非接收。

C Bindings

提供使用一直到「原生 C」程式碼庫提供完整的功能。您的應用程式在 Bindings 中呼叫 C API，並且這個呼叫會遞送到 MQe 的 Java 類別，來完成功能。這些 Bindings 是為 MQe 第 1 版而撰寫，但可存取幾乎所有 MQe 第 2 版的 Java 功能。

如需程式碼庫的詳細資訊，請參閱 第 3 頁的『程式碼庫』。

另請參閱：第 28 頁的『商標』。

MQe 概觀

MQe 的用途

- 用於感應器、電話、個人數位助理 (PDA) 和膝上型電腦等輕型裝置以及桌上型電腦的安全傳訊。

什麼是傳訊？

- 一種 (MQ 及 MQe 內建的) 軟體，可供您傳送及接收應用程式之間及網路上的資料。應用程式設計師毋需高深的通訊程式設計知識，即可從應用程式遞送可靠、低耦合的訊息。
- 應用程式將資料傳送給其他應用程式時，會先將資料放進訊息，再將訊息放上佇列。
- 佇列管理程式會擁有並執行該佇列。
- 您可配置其他應用程式 (或同一個應用程式的其他部分) 進行下列步驟之一：
 - 其他應用程式可自同一佇列擷取這些訊息。
 - 佇列管理程式透過網路上的連線，將佇列上的訊息傳送到另一台電腦遠端佇列管理程式的佇列上，其他應用程式可從另一台電腦擷取訊息。
 - 目的地應用程式可在需要時，從網路上拉出訊息。
- 單一佇列管理程式上可以有多個佇列。
- MQe 上，每個 JVM 或程序只能有一個佇列管理程式。

什麼是 MQe？

- 一種工具集，可在多種平台上使用：
- 有 Java 及 C 兩種 API。
- 產品功能有 Java 類別及 C .DLL 兩種形式。

如何運作？

- 佇列管理程式是一個資訊集，說明佇列管理程式的原始基本組態。
- 這些資訊保留在 MQe 登錄裡（附註：若使用 Windows® 系統，這些資訊指的並不是 Windows 登錄）。
- 裝置上的應用程式可以啟動佇列管理程式，只要執行應用程式，佇列管理程式就會跟著執行。一旦應用程式停止，佇列管理程式也會跟著停止。
- 佇列管理程式執行時，會一面傳送 MQe 管理訊息給自己，一面不斷進行配置及重新配置，並同時更新登錄內的資訊。通常，這些訊息都是用管理工具 MQe_Script 及 MQe_Explorer 產生的。
- 您可以在伺服器上撰寫應用程式來執行佇列管理程式。之後持續地執行此應用程式，用戶端佇列管理程式便會在必要時傳送訊息。Windows 系統可將佇列管理程式當成服務來執行，每次電腦啟動時，該服務就會執行。

如何使用？

- 使用 MQe 前，您必須先撰寫您自己的應用程式。
- 您可在 PC 上建立自己的 MQe 裝置佇列管理程式配置並開發自己的 MQe 應用程式。
- 您只須將（您的應用程式及 MQe）所需的檔案及元件下載到裝置上，即可在裝置上執行。
- 您是在電腦上（而非裝置上）執行 MQe 伺服器佇列管理程式，裝置會連線到該電腦並將訊息傳送到此。
- 聯絡 MQ 時，請執行電腦上（而非裝置上）的 MQe 閘道佇列管理程式，此媒介程式可接收您裝置（或伺服器）上的 MQe 訊息，將之轉換成可與 MQ 相容的訊息，然後與 MQ 交換。
- 您可下載 MQe_Script（可撰寫的命令）或 MQe_Explorer (GUI) 這兩個 SupportPacs，來建立佇列管理程式，並在遠端管理您的 MQe 網路。
- 透過這些工具，您毋需撰寫應用程式，即可使用 MQe。

是否有其他特殊功能？

- 您可利用配接器將 MQe 對映到裝置介面。比方說：
 - 通道（用於連線的兩端）利用通訊協定配接器在 HTTP、原生 TCP/IP、UDP 及其他通訊協定上執行。
 - 佇列可利用欄位儲存體配接器來連接儲存體子系統（如記憶體或檔案系統）。
- 您有三重安全保障：

本端安全

保護本端層次的訊息相關資料。

訊息安全

保護起始與接收 MQe 應用程式之間的訊息。

佇列安全

保護起始佇列管理程式與目標佇列之間的訊息。

- 您可使用規則來自訂某些主要 MQe 元件的行為。
- 您可使用追蹤及事件日誌來進行應用程式除錯。

另請參閱：

『程式碼庫』

第 20 頁的『MQe SupportPacs』

程式碼庫

概觀

MQe 應用程式設計介面 (API) 是 MQe 的程式設計介面。MQe API 支援的 Java 及 C 程式設計語言如下：

Java 版本

可存取所有 MQe 第 2 版的功能。詳細類別、方法及程序請參閱 *Java Programming Reference*。本資訊中心將提供多個 MQe 程式設計範例。

C 支援有三個版本：

原生 C 程式碼庫

可存取 MQe 功能子集，主要的限制是只能使用裝置佇列管理程式。其他限制如下（另請參閱下列表格）：

- 不支援儲存及轉遞佇列或橋接佇列
- 僅支援 HTTP 配接器
- 僅支援 RLE 壓縮程式
- 僅支援 RC4 加密程式
- 僅支援 *MAttribute* 及本端安全功能

詳細的方法和程序請參閱 *C Programming Reference*。本資訊中心將提供多個 C API 的程式設計 MQe 範例。

C Bindings

「原生 C」程式碼庫的功能完備之前，都將使用 C Bindings。您的應用程式在 Bindings 中呼叫 C API，並且這個呼叫會遞送到 MQe 的 Java 類別，來完成功能。這些 Bindings 乃為 MQe 第 1 版而撰寫，但可存取幾乎所有 MQe 第 2 版的 Java 功能。詳細的方法和程序請參閱 *C Programming Reference*。C Bindings 的程式設計 MQe 範例請參閱 *C Bindings Programming Reference*。

佇列管理程式類型

本文件及下表皆使用下列佇列管理程式說明，請務必清楚區分這些佇列管理程式：

裝置佇列管理程式

此佇列管理程式無接聽器元件及橋接器元件。因此只能傳送訊息，無法接收訊息。

伺服器佇列管理程式

此佇列管理程式可連接一個接聽器。透過接聽器即可接收訊息，亦可傳送訊息。

閘道佇列管理程式

此佇列管理程式可連接一個接聽器及一個橋接器。透過接聽器來接收及傳送訊息，透過橋接器來與 MQ 通訊。

選項表格

選項	Java	原生 C	C Bindings
作業系統	任何有 Java 2 的作業系統（指 Java Version 1.2 以上）	PocketPC2002、PocketPC2003、Windows (MQe 2.0.1.5 版以上)	Windows 32 位元
佇列管理程式	全部	僅裝置	全部
連接 MQ 的閘道（有橋接器及接聽器的佇列管理程式）	是	否	是
儲存及轉遞佇列、橋接佇列	是	否	是
配接器	全部	僅 HTTP	全部
壓縮程式	全部	僅 RLE	全部
加密程式	全部	RC4	全部
安全特性	全部	僅 MAttribute 及本端	全部
新增訊息到追蹤	是	否	是
事件日誌	是	否	是
專用登錄和認證	是	否	是
內容規則	是	否	是
橋接規則	是	否	是
自訂類別	全部	否	部份
應用程式載入	是	否	是

2.0.2 版的新功能

WebSphere MQ Everyplace 2.0.2 版的新功能

- 已移除所有工具集安裝內的文件，改放在 Web 上。
- 「C 原生」程式碼庫新增可支援代理字元的 UTF-16。
- 「C 原生」程式碼庫支援 WinCE .NET 4.2。請注意：我們使用 .NET 環境所提供的 Platform Invocation Services 來支援 WinCE .NET 4.2 上的 WebSphere MQ Everyplace C DLL。
- 應要求新增一移轉章節。本節說明如何使用 MQe 隨附的新類別，讓使用者透過閘道佇列管理程式，向 WebSphere Message Broker 傳送、發佈及訂閱訊息。

使用本資訊中心

協助工具

本頁後續幾節僅說明與 MQe 文件相關的 Eclipse 使用要訣。

其他 Eclipse 協助工具的一般資訊請參閱 Eclipse 在 *Accessibility in the help system* 中的說明。

鍵盤要訣

- 切換窗格時請使用功能鍵 F6。Mozilla 1.2 或以上請使用 Ctrl-F6。
- 切換窗格或工具列上的圖示或物件時，請使用 Tab 鍵。
- 要回到上一個檢視主題時，請使用倒退鍵。前進及後退時，請使用 Alt-向右鍵及 Alt-向左鍵。

- 由鏈結或搜尋結果進入一主題後，按一下主題工具列上的「顯示於目錄」圖示（請使用功能鍵 F6 來切換窗格，Mozilla 1.2 或以上的瀏覽器請使用 Ctrl-F6，之後請使用 Tab 鍵來選取工具列上的圖示）即可找出該主題在哪一個目錄下。
- 開啓快速功能表時，請使用 Shift-F10。您也可使用鍵盤上的特殊快速功能表鍵（如果有的話）。

窗格

- 左邊是導覽窗格，內含下列標籤：
 - 內容
 - 搜尋結果
- 而您現在正在閱讀主題窗格，這裡會顯示目前的說明主題。按一下反白的鏈結即可直接跳至其他主題（請使用鍵盤上的 Tab 鍵來移動反白，然後按 Enter 進入該鏈結。按一下主題並按下 **Ctrl-F**，即可搜尋該頁面上的文字（請使用鍵盤上的功能鍵 F6 來切換窗格，Mozilla 1.2 或以上的瀏覽器請使用 Ctrl-F6）。請使用鍵盤上的上移鍵及下移鍵來捲動頁面。
- 「主題窗格」上方的工具列包含下列圖示：「回上頁」、「回下頁」、「顯示於目錄」（供搜尋後使用）以及「列印」（分別是鍵盤上的 Alt-向左鍵、Alt-向右鍵、工具列上的圖示以及 Ctrl-P；請使用功能鍵 F6 來切換窗格，Mozilla 1.2 或以上的瀏覽器請使用 Ctrl-F6，之後請使用 Tab 鍵來選取工具列上的圖示）。
- 「導覽窗格」上方的工具列可用來搜尋。

實用的功能

- 使用「搜尋」工具列找到一主題後，按一下主題工具列上的顯示於目錄圖示（請使用鍵盤上的功能鍵 F6 來切換窗格，Mozilla 1.2 或以上的瀏覽器請使用 Ctrl-F6，之後請使用 Tab 鍵來選取工具列上的圖示）即可找出該主題在哪一個目錄下。
- 按一下「主題窗格」（請使用鍵盤上的功能鍵 F6 來切換窗格，Mozilla 1.2 或以上的瀏覽器請使用 Ctrl-F6）並按下 **Ctrl-F**，即可搜尋頁面上的文字。
- 用滑鼠右鍵按一下「目錄」裡的項目，並選取內容，即可開啓對話框。您可滑過該主題的 URL 並將其複製起來，以供日後參考。例如，您可將其加到瀏覽器的書籤清單或複製到電子郵件裡。

列印

- 按一下工具列上的列印圖示，或按一下主題並按下 Ctrl-P（請使用鍵盤上的功能鍵 F6 來切換窗格，Mozilla 1.2 或以上的瀏覽器請使用 Ctrl-F6）。

字型大小

- 請使用瀏覽器本身的功能來調整字型大小。

搜尋要訣

請使用「導覽窗格」上方的「搜尋」工具列來尋找整個檔案庫中的主題。請注意：

- 請使用「進階搜尋」來縮小搜尋範圍。
- 有些 Eclipse 說明瀏覽器版本在搜尋數字時，必須將數字用雙引號括起來，例如："2.0.1"。
- 請以 * 來表示不完整的字，例如：搜尋 **admin*** 將可找到 *administration*、*administer* 等字。
- 搜尋多個字時請用空格將每個字隔開，例如，**queue manager** 將可找到包含 *queue*，以及 *manager* 的主題。
- 搜尋詞組時請將詞組用雙引號括起來，例如：搜尋 "**queue manager**" 將可找到包含完整詞組 *queue manager* 的主題。
- 檢視主題時，您所搜尋的字會反白。
- 若要尋找顯示主題內的字，請按一下主題窗格（請使用鍵盤上的功能鍵 F6 來切換窗格，Mozilla 1.2 或以上的瀏覽器請使用 Ctrl-F6）並按下 **Ctrl-F**。

什麼是 MQe

MQe 簡介

MQe 是 WebSphere MQ 商業傳訊系列產品的成員之一，也是 WebSphere Everyplace 系列產品的成員之一。您可使用 MQe 來撰寫您自己的應用程式，以 *once and once-only* 確定遞送方式，來交換包含資料的訊息。MQe 可與 WebSphere MQ 系列的其他成員整合。

MQe 旨在滿足感應器、電話、個人數位助理 (PDA) 等輕型裝置以及膝上型/桌上型電腦的傳訊需求。它支援行動式環境，而且適合於透過公用網路來使用，同時也支援使用脆弱的通訊網路而產生之需求。由於許多 MQe 應用程式在執行時，都沒有受到網際網路防火牆的保護，因此 MQe 本身即具備安全功能。

瞭解安全傳訊的概念有助於您瞭解本產品及文件。

如果您不具備這個基礎概念，請參閱 WebSphere MQ 書籍 *An Introduction to Messaging and Queuing*，GC33-0805。您可以在線上 WebSphere MQ 書庫的 Book 一節下，找到本書的電子檔。請造訪 WebSphere MQ 網頁：<http://www.ibm.com/software/integration/websphere/library/>。

MQ 系列的 MQe

基本傳訊

傳訊的基礎在於佇列及佇列管理程式，而與特定產品或產品群組無關。佇列管理程式負責管理可儲存訊息的佇列。應用程式會與本端佇列管理程式通訊，並會取得佇列中的訊息，或將訊息放置在佇列中。如果訊息要放置到遠端佇列（另一個佇列管理程式所擁有的佇列），則訊息將透過連線傳輸至遠端佇列管理程式。在這個方式下，可以經由一或多個中間佇列管理程式，將訊息傳送到目的地。傳訊的本質在於將傳送的應用程式和接收的應用程式分開，並在必要時，從中將訊息放入佇列。

MQ 及 MQe 提供 MQ 系列傳訊功能。這兩者的設計都能夠支援一或多個硬體伺服器平台，以及大多數相關的作業系統。在平台功能具有高度多樣性的情況下，這些個別產品分成幾個產品群，來反映共通的功能和設計：

分散式傳訊

WebSphere MQ 支援 Windows NT[®]、Windows 2000、AIX[®]、iSeries[™]、HP-UX、Solaris 等平台

主機傳訊

WebSphere MQ 支援 z/OS[®]

通用傳訊

MQe 支援 Windows、AIX、Solaris、Linux[®] 及 HP-UX

詳細資訊請參閱第 8 頁的『MQe 如何運作』。

MQe

MQe 支援多種網路配置。它不是用 MQ 主機或分散式產品中的用戶端或伺服器概念。而是將 MQe 佇列管理程式配置成用戶端或伺服器，來執行應用程式定義的作業。

例如，您可以自訂 MQe 的配置，使之能夠與 MQ 主機佇列管理程式交換訊息。但您必須先幫 MQe 佇列管理程式配置上橋接器功能。如果沒有橋接器，MQe 佇列管理程式僅能與其他 MQe 佇列管理程式作直接通訊。然而，它可以透過網路中具有橋接器功能的其他佇列管理程式進行間接通訊。

詳細資訊請參閱第 8 頁的『MQe 如何運作』。

MQe 如何延伸 MQ 系列

MQe 以下列方式延伸 MQ 系列的傳訊範圍：

- 支援低階裝置，如 PDA、電話和感應器。MQe 也支援進階級裝置，如膝上型電腦、工作站、分散式和主機平台。MQe 提供訊息的 once and once-only 確定遞送，且容許與其他系列成員交換訊息。
- 提供小型傳訊機能。
- 提供範圍更廣的安全特性來保護儲存體和傳輸中的訊息、佇列和相關資料。
- 在網路不穩定或頻寬非常有限的不良通訊環境中，使作業更有效。MQe 具備高效的有線通訊協定，可在通訊鏈結失敗時自動回復。
- 支援行動式使用者，讓網路連線功能的指向能隨著裝置的漫遊而變更。電池資源和網路不佳時，MQe 仍可控制行爲。
- 透過適當配置的防火牆來作業。
- 將使用者的管理作業減至最少。因此 MQe 很適合拿來建置公用程式型應用程式。
- 您可透過應用程式提供的規則，輕鬆地自訂及延伸。

MQe 並不支援 MQ 的所有功能。在環境、作業系統和通訊考量之外，一些更重要的差異如下：

- 沒有叢集支援
- 沒有分送清單支援
- 沒有群組化或區段化的訊息
- 沒有平衡資料流量或暖機待用功能
- 沒有參照訊息
- 沒有報告選項
- 沒有共用佇列支援
- 沒有觸發
- 沒有工作單元支援，沒有 XA 對等關係
- 不同的可調整性及效能特性

不過您可以利用 MQe 的功能，或利用子類別，或取代提供的類別，或利用規則、介面及其他產品內建的自訂功能，在 MQe 內完成許多應用程式作業。

MQe 可能的用途

MQe 支援機動性及脆弱的通訊網路。因為 MQe 的目標在於輕型裝置，所以它在使用系統資源時很節省。它會提供訂做的功能及介面，但它不是旨在提供與 MQ 系列的其他成員相同的功能。它也會併入唯一的功能以支援使用者的特定類別，例如：廣泛安全供應、訊息、同步和非同步傳訊、遠端佇列存取，以及訊息推入和拉出。

實務練習及應用

MQe 應用程式可以有多種類型，可針對特定使用者群組來開發自訂應用程式。下列清單提供一些範例：

零售應用程式

- 收銀機交易對於主機系統的零星回饋，如訊息分配管理系統

消費者應用程式

- 在家中使用 PDA 進行超級市場購物
- 收集航空公司提供的旅客優惠
- 從行動電話進行財務交易

控制應用程式

- 收集和整合透過衛星傳輸的油管感應器資料
- 安全地遠端操作設備（如活門）以保證操作員的效力

行動式人力

- 拜訪專家，例如，保險代理人
- 迅速發佈包裹遞送公司的客戶收據證明
- 廚房與伺者之間的資訊交換
- 高爾夫球比賽分數的保存
- 警察的安全行動式系統傳訊
- 在經常失去通訊的情況下，公用事業工人的工作資訊
- 公制讀數

個人生產力

- 郵件及行事曆抄寫
- 資料庫抄寫
- 下載至膝上型電腦

MQe 如何運作

MQe 程式設計模型的基本元素有訊息、佇列及佇列管理程式。

- MQe 訊息含有應用程式定義的內容。訊息儲存在佇列，而且可以透過 MQe 網路來移動。您可以指定目標佇列管理程式和佇列名稱，將訊息傳送到目標佇列中。
- 應用程式會利用放置作業，將訊息放在佇列中，且通常會利用取得作業來擷取它們。
- 佇列可以是本端或遠端，而且是由佇列管理程式加以管理。
- 登錄儲存配置資料。

請閱讀此章節中其他的主題，來學習更多資訊。

訊息

訊息是指由一個應用程式發出給另一個應用程式的資料集合。MQe 訊息與 MQ 傳訊所支援的訊息不同。

- 在 MQ 中，訊息是位元組陣列，可分成訊息標頭和訊息主體。MQ 會建立訊息標頭，內含最重要的資訊，例如辨識回覆佇列、回覆佇列管理程式、訊息 ID 以及相關 ID。訊息主體含有只對應用程式有用的資料。
- MQe 訊息則沒有標頭或訊息主體的概念。它們以 *MQeFields* 的形式呈現，內含名稱、資料類型及資料本身。訊息名稱是沒有長度限制的 ASCII 字串，但不包括這些字元的任一個：

```
{ } [ ] # ( ) : ; , ' " =
```

表 1 說明不同的資料類型：

表 1. 資料類型

類型	說明
ASCII	字串或固定 ASCII 字串的動態陣列，其中不含 { } [] # () : ; , ' " =
Boolean	True 或 false 值
位元組	固定陣列，或位元組值的動態陣列
倍精準浮點	值、固定陣列，或倍精準浮點值的動態陣列

表 1. 資料類型 (繼續)

類型	說明
欄位	物件或欄位物件的動態陣列 (因此, 支援欄位物件的巢狀結構)
浮點數	值、固定陣列, 或倍精準浮點值的動態陣列
整數	整數的 4 位元組值、固定陣列或動態陣列
長整數	長整數的 8 位元組值、固定陣列或動態陣列
短整數	短整數的 2 位元組值、固定陣列或動態陣列
Unicode	Unicode 字串的字串或動態陣列

另外, 訊息還包括 MQe 所產生的 UID (唯一 ID)。這個 UID 可識別整個 MQe 網路中的每一個個別訊息物件, 它由下列項目構成:

原始佇列管理程式

這是原始佇列管理程式的名稱, 它必須是唯一的。它是佇列管理程式在接收訊息時所加入的。當它是 ASCII 時, 每一個字元的長度都是一個位元組。

建立時間

這是訊息的*時間戳記*。因此在 Java 中, 這是訊息建立時, 本端系統的時間; 在 C 中, 這是欄位項目加入佇列時, 本端系統的時間。欄位項目隨後將變成訊息。

即將傳給其他 MQe 佇列管理程式的訊息並不需要其他資訊, 但其他內容幾乎都一定會顯示。其他的內容可以:

- 反映現行狀態
- 與特定訊息子類別建立關聯性
- 允許您自訂訊息

註: 在 C 程式碼庫中, 欄位項目要在抵達佇列後才會變成訊息。

MQe 會在訊息中加入與內容相關的資訊 (之後再將之移除) 以便實作傳訊和佇列作業。在佇列管理程式之間傳送訊息時, 您可以新增重送資訊, 以指出正在重新傳輸資料。

訊息也可以具有屬性。屬性是 MQe 安全模型的基礎, 容許選擇性存取內容和保護內容。它們含有下列內容:

表 2. 屬性物件內容

內容	說明
鑑別	控制存取
加密	在傾出物件時保護內容 (並容許還原)
壓縮	縮減儲存體需求 (用於傳輸和儲存)
規則 (不適用於 C 程式碼庫)	控制容許的作業

如需表 2 中內容的詳細資訊, 請參閱第 16 頁的『安全』。

佇列

佇列用以保留訊息。應用程式不會直接存取佇列, 而是透過佇列管理程式來使用佇列。

佇列是依名稱識別的, 名稱可以是無限字串長度的 ASCII 字元, 但不包含下列字元:

{ } [] # () : ; , ' " =

然而，在特定佇列管理程式內，佇列名稱必須是唯一的。若要與 MQ 交互運作，建議您也遵守 MQ 的命名限制，例如名稱長度不超過 48 個字元。名稱長度也有可能受您所使用的檔案系統所限制。MQe 支援許多不同的佇列類型：

本端佇列

應用程式會利用本端佇列，以一種安全的方法（不包括硬體故障或失去裝置）來儲存訊息。本端佇列屬於特定佇列管理程式。這可以是獨立式佇列管理程式，或是連接至網路的佇列管理程式。

遠端佇列

遠端佇列是 MQe 網路中，位於另一個佇列管理程式的佇列之本端參照。本端參照的名稱和目標佇列相同，但遠端佇列定義會指出擁有真正佇列的佇列管理程式。遠端佇列也有與存取、安全特性，以及傳輸選項相關的內容。它們的存取模式可以是同步或非同步。同步遠端佇列沒有儲存體，訊息會立即送到遠端佇列管理程式。非同步佇列則會將訊息儲存在本端儲存體，然後再用背景緒將訊息傳送到遠端佇列管理程式。

儲存及轉遞佇列

儲存及轉遞佇列會先替一或多個佇列管理程式儲存訊息，直到該佇列管理程式可以接收訊息。C 程式碼庫中沒有此類佇列。儲存及轉遞佇列具有兩個主要用途：

1. 在網路中啟用中間的訊息儲存，使訊息能夠前往它們的目的地（轉遞角色）。
2. 透過起始伺服器佇列保留訊息等待集合

此類佇列通常（但不必然）會定義在伺服器或閘道上。儲存及轉遞佇列可以保存許多目標佇列管理程式的訊息，或每一個目標佇列管理程式都可以有一個儲存及轉遞佇列。

起始伺服器佇列

當遠端佇列和儲存及轉遞佇列跨越網路來推送訊息時，起始伺服器佇列會利用起始傳輸的傳送佇列從遠端佇列拉出訊息。訊息決不會傳至起始伺服器佇列。

起始伺服器佇列定義負責識別出遠端佇列管理程式中的儲存及轉遞佇列。之後，起始伺服器佇列會從儲存及轉遞佇列中拉出以它的本端佇列管理程式為目的地的訊息。在單一佇列管理程式中可以定義多個起始伺服器佇列定義，其中的每個定義都會關聯於不同的遠端儲存及轉遞佇列。

管理佇列

管理佇列是一種接受管理訊息的本端佇列。管理訊息內含經過應用程式內部處理、與 MQe 特定元素相關的指令。每一個管理動作都可以選擇性地導致管理回覆訊息傳回給原始應用程式。這些回覆訊息將通知您管理動作成功或失敗。在這種方式下，使用管理佇列容許某個佇列管理程式上的元素，以同步或非同步方式控制第二個佇列管理程式的配置。管理訊息會按抵達管理佇列的次序來加以處理。如需進一步資訊，請參閱有關第 12 頁的『管理』的一節。

MQ 橋接佇列

橋接佇列是一種特殊格式的遠端佇列，用來說明 MQ 遠端佇列管理程式上的佇列。橋接佇列會從它們參照的 MQ 佇列進行放置或取得作業。此佇列僅出現在 Java 中，訊息在 MQe 與 MW 系統之間交換時，它會使用轉換程式，在必要時將資料或訊息重新格式化。您僅能在閘道佇列管理程式上建立橋接佇列。

MQe 會將資料安全地儲存在佇列中，確保訊息有實際寫入媒體，而不只是儲存在作業系統中。不過 MQe 不會獨立記載訊息和佇列的變更。因此，若要從媒體失效中復原，您需要部署硬體解決方案，如 RAID 磁碟系統。另外，將佇列對映至可回復的儲存體，比方說，資料庫子系統。MQe 用戶端位於輕型系統上，但伺服器佇列管理程式常必須全天候 (24/7) 執行，以便隨時進行失效接手。

MQe 具有四個常用的系統佇列：

管理佇列

接收管理訊息

無法傳送的郵件佇列

儲存其他方式所無法遞送的訊息

管理回覆佇列

接收給管理訊息的回覆

SYSTEM.DEFAULT.LOCAL.QUEUE

與 MQ 伺服器上必要的系統佇列共用名稱。

佇列管理程式

MQue 佇列管理程式使 MQue 得以支援各種網路配置。它提供：

- 存取中心點，用來集中存取 MQue 應用程式的傳訊及佇列網路
- 選用的用戶端佇列作業
- 連線控制
- 選用的管理功能
- 訊息的 once and once-only 確定遞送
- 失效狀況的自動復原
- 可自訂的規則型行為

在 MQue 中，一台 Java 虛擬機器 (JVM) 上或一個原生應用程式程序裡，一次僅能動用一個佇列管理程式。若要在機器上具有多個佇列管理程式，您需要多個 JVM 或多個原生應用程式程序。

佇列管理程式是以廣域唯一的名稱及沒有長度限制的 ASCII 字串（不包括下列任一個字元）來加以識別：

{ } [] # () : ; , ' " =

MQue 或 MQ 並不強制執行這項限制，但重複的佇列管理程式名稱可能會使訊息送到錯誤的佇列管理程式。如需交互運作，我們建議您應該將名稱長度限制為最多只能有 48 個字元。您正在使用的檔案系統也可能會限制名稱長度。

您可以配置有或沒有佇列作業的佇列管理程式。所有佇列管理程式都支援同步的傳訊作業。含本端佇列作業的佇列管理程式也支援進行非同步的訊息遞送。非同步訊息遞送和同步訊息遞送有非常不同的特性和結果：

同步訊息遞送

使用同步訊息遞送時，應用程式會將訊息放置在 MQue 中，以便遞送到遠端佇列。MQue 會聯絡目標佇列，同時遞送訊息。遞送之後，MQue 會立即回傳給應用程式。如果訊息無法遞送，傳送端的應用程式會立即收到通知。MQue 不負責同步訊息遞送（不確保訊息遞送）。

非同步訊息遞送

使用非同步訊息遞送時，應用程式會將訊息放置在 MQue 中，以便遞送到遠端佇列。MQue 會立即回傳給應用程式。如果可以立即遞送訊息，或移到適當的暫置點的話，便會傳送它。如果不可以，會將它儲存在本端。非同步遞送提供 once and once-only 確定遞送，因為訊息已先傳給 MQue，而 MQue 會負責遞送（確保訊息遞送）。

佇列管理程式配置： MQue 佇列管理程式載入前，需要有一應用程式來建立所需的環境。意即，應用程式如果要啟動佇列管理程式，該應用程式必須能在載入佇列管理程式之前，先存取特定資訊。例如，如果您要設定 Java 中通訊配接卡的封包大小，則在載入佇列管理程式之前，您必須先設定好必要的 Java 內容。應用程式會需要兩項資訊：「MQue 登錄」的位置及佇列儲存檔的位置。登錄（請勿與「Windows 登錄」混為一談）會保存所有與

佇列管理程式相關的物件的定義，如實例佇列及連線定義。如此，佇列管理程式便可在載入時，建立正確的物件。佇列儲存檔是佇列的所在位置，可供佇列管理程式保存本端佇列的訊息，佇列管理程式停止後到下一次啟動前，這些訊息都會在那裡。

您可以用 API、MQe 隨附的公用程式或 MQe_Explorer 等管理工具來配置 MQe 環境。這些方法可以擷取起始設定檔中的環境參數，不過，這是選用性的。

您可以用 MQ 橋接功能來配置佇列管理程式。我們稱之為閘道，它可在 Java 中與 MQ 主機及分散式產品交換訊息。C 程式碼庫僅能使用裝置佇列管理程式。

佇列管理程式作業： 佇列管理程式支援傳訊作業並管理佇列。應用程式會利用如下的方法，透過佇列管理程式的服務來存取訊息：

取得 這個作業會從佇列中移除訊息。

放置 這個作業會將訊息放到佇列上。

刪除 只要指定 UID，即可從佇列中刪除訊息，毋需進行「取得」。

瀏覽 您可以使用過濾器（請參閱後續說明）來瀏覽佇列以取得訊息。瀏覽會擷取符合過濾器的所有訊息，但會將它們留在佇列中。MQe 並支援鎖定瀏覽。可讓您鎖定符合的訊息。

等待 在 Java 中，訊息到達佇列之前，應用程式可等待一段指定時間。這不適用於 C 程式碼庫。

接聽 在 Java 中，應用程式也可以用選用的過濾器來接聽 MQe 訊息事件。然而，為了達成這個目的，您必須新增一個接聽器至佇列。當訊息到達佇列時，接聽器會獲得通知。這不適用於 C 程式碼庫。

其中的許多作業都會使用 **filter** 參數。過濾器會比對元素以判定是否相等，而且訊息的任何部份都可以用於選擇性擷取。大部份方法呼叫都會併入訊息的編碼和解碼所用的屬性。

管理

MQe 介面可產生及接收管理訊息，以利管理。應用程式負責與訊息相關的功能，管理則協助配置和管理 MQe 資源，如佇列與連線。

要求會傳送到目標佇列管理程式的管理佇列中，必要時，可能會收到回答。任何本端或遠端 MQe 應用程式都可以直接建立和處理管理訊息，也可以間接透過 **helper** 方法來建立和處理管理訊息。

C 程式碼庫內建管理者，可供您執行部份管理動作。這些動作僅會在本端佇列管理程式管理的資源上執行。

管理佇列本身無法對個別資源執行管理。每一個資源及其對應的管理訊息中都含有相關資訊。

管理訊息： 佇列管理程式一旦建立好，您便可將管理訊息傳送到目標佇列管理程式的管理佇列，以配置佇列管理程式。佇列管理程式若沒有管理佇列，將無法管理它。使用管理訊息旨在同時以相同方式執行本端及遠端管理。

將建立管理訊息並傳送至要管理之佇列管理程式的管理佇列。您可以套用佇列型安全屬性來控制存取。管理訊息包括要求的詳細資料、指出是否需要回應，以及含有用以識別目標佇列管理程式及佇列的位址。因此，MQe 有下列幾種管理訊息：

- 指出不需要回覆之管理動作的指令
- 需要回覆的要求
- 從原始訊息的複本建構的回覆訊息

傳送端可以新增其他的欄位，供接收端使用。管理佇列本身將對訊息起作用。管理訊息可以查詢、建立、刪除或更新物件。對於物件子集，它們可以執行其他的功能，如停止及啟動。

您也可以間接透過 MQe_Explorer 來產生管理訊息，這個管理工具可以用圖形使用者介面來管理系統。MQe 不會隨附 MQe_Explorer，但您可到 SupportPac™ 免費下載。

選擇性管理： 管理佇列上的鑑別程式可以控制管理的存取。提供的鑑別程式會考慮本端應用程式來代表相同的本端使用者，因此，可啟用或阻止所有應用程式的管理。

在任何管理訊息流動之前，若在連線線上啟動鑑別程式，將控制遠端管理應用程式。這將區分不同的遠端應用程式，然後啟用或阻止每一個遠端應用程式的管理。在所有情況下，管理不是完全地啟用，就是完全地阻止。

鑑別程式可以追蹤使用者身份所關聯的許可權，之後，便能在這些許可權的基礎上，處理管理訊息。請參閱第 16 頁的『安全』提供有關鑑別的詳細資訊。您也可以使用與佇列相關的規則，以類似方式啟用或阻止動作。請參閱第 18 頁的『自訂規則』提供有關規則的詳細資訊。

監視和相關動作： 管理不只包括建立及修改元素，還可以包括監視系統狀態，並在佇列已滿時通知操作員，或是處理錯誤狀況，比方說，在訊息到達目標佇列而訊息太大時，採取適當動作。元素狀態有顯著變更時，或特定類型的錯誤狀況出現時，MQe 會使用規則來處理上述問題。MQe 有預設的規則實作可供使用者自訂。相關詳細資訊請參閱第 18 頁的『自訂規則』。

連線

連線提供佇列管理程式建立與遠端佇列管理程式通訊鏈結的資訊。然後，佇列管理程式將使用這些連線來交換資訊。連線定義會儲存在每個佇列管理程式的本端環境內。

註： C 程式碼庫僅能使用裝置佇列管理程式。

連線的部份主要功能如下：

同時支援同步及非同步傳訊

同步傳訊提供一個直接從來源應用程式至目標佇列的傳輸服務，不需佇列在來源佇列管理程式。非同步傳訊是從來源佇列管理程式到目標佇列的傳輸服務，可能會經過來源佇列管理程式的佇列。

端對端服務規定

從來源佇列管理程式到目的地佇列管理程式的連線，可能會經過中間佇列管理程式。所用的基礎傳輸通訊協定可能會隨著連線所經過的這些中介而改變。數個連線可以鏈結一起，以構成端對端連線。

支援壓縮、加密、及鑑別

擁有這些安全特性的連線，可在傳輸時保護資料。

支援主從架構作業

主從架構連線是要求/回應。用戶端向伺服器端提出要求，伺服器端則回應這個要求。請注意，這不會限制訊息流動。訊息可往返於用戶端與伺服器之間。

下列圖解顯示部分典型的 MQe 配置。基於清晰之目的，圖解只顯示已定義的直接連線。您也可以定義使用直接連線的間接連線。圖解中，箭頭指向伺服器的線條表示主從架構連線。用戶端可以利用主從架構連線，將訊息傳送給伺服器，或從該伺服器中拉出要傳給他們的訊息。沒有箭頭的線條表示 MQe 和 MQ 之間的通訊通道。



圖 1. 獨立式 MQe 佇列管理程式

圖 1 為獨立式佇列管理程式，支援一或多個以佇列交換資料的應用程式。

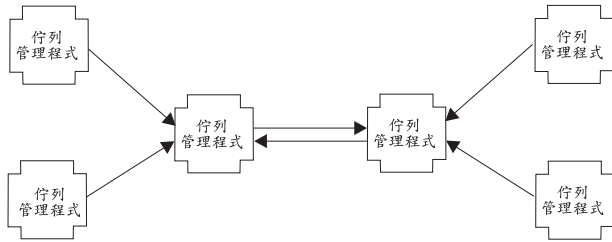


圖 2. 小型網路配置

圖 2 為小型網路配置，中央伺服器佇列管理程式用一對直接主從架構連線來交換資訊。每一個用戶端佇列管理程式都用直接主從架構連線，連結到其中一個伺服器佇列管理程式。

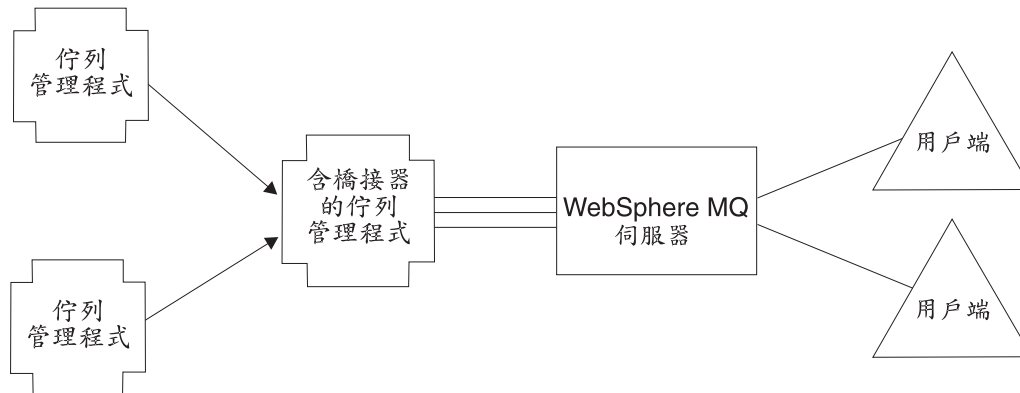


圖 3. 整合式 MQ 系列網路

圖 3 (圖 3) 為一個 MQe 配置，一個佇列管理程式已配置橋接器選項，用戶端通道儲存區也已導向單一個目標 MQ 主機/分散式伺服器。

當連線通常定義遠端佇列管理程式的存取時，它們有時被稱為遠端佇列管理程式定義。

您也可以指定間接連線。在這個情況下，MQe 會透過其他（可鏈接的）佇列管理程式來遞送連線，遞送途中通訊協定可以改變。如果希望裝置能由單一點進入 MQ 網路，建議您使用間接連線。

如同大部份 MQ 元素一般，您可以定義連線的別名。使用本端連線（定義為名稱符合本端佇列管理程式名稱的連線）來定義本端佇列管理程式本身的別名。

連線支援雙向流動，而且佇列管理程式可以視需要來建立它們。非同步和同步傳訊用的是同一個連線，所用的通訊協定僅 MQe 可以使用。

連線定義會判斷特定連線所用的鏈結和通訊協定。在每一個中間節點，任何流過的訊息都會在該點傳給佇列管理程式。佇列管理程式將根據它具有的資源來處理訊息。所以訊息可以放置在可能是本端佇列、遠端佇列或儲存及轉遞佇列的佇列。根據遠端佇列的類型，放置在遠端佇列的訊息將繼續它們的行程。同步遠端佇列將立即將訊息移至前面。非同步遠端佇列將在移動它們的訊息之前先儲存它們。

應用程式設計師或管理者無法直接見到連線，它們是佇列管理程式在需要時所建立的。連線會將佇列管理程式鏈結起來，MQe 會根據要流動的資訊來變更其性質。傳輸程式乃 MQe 元件之一，可利用連線來提供佇列層次的通訊。再說明一次，這些都是應用程式設計師或管理者無法見到的。

MQe 一定會將訊息遞送到應用程式一次，且只遞送這麼一次，以此實現確定傳訊。它是在傳輸結束時，刪除副本之前，先確保訊息已順利從一個佇列管理程式傳到另一佇列管理程式，並經過認可，來做到這一點。萬一通訊失敗，如果沒有收到認可，可以重新傳輸訊息，因為單次遞送並沒有暗示只能一次傳輸。然而，不會重複遞送。

連線樣式: MQe 支援主從架構作業。用戶端可以起始與伺服器的通訊。伺服器僅能回應由用戶端起始的要求。涉及的元件包括：

接聽器 接聽送入的連線要求。

佇列管理程式

提供傳訊和佇列功能來支援應用程式。

表 3. 連線樣式

	佇列管理程式	接聽器
用戶端	是	否
主從架構	是	是
伺服器	是	是
Servlet	是	否

『表 3.』顯示這些元件和連線樣式之間的關係。主從架構連線樣式意指，MQe 能否在用戶端或伺服器模式下作業。Servlet 選項意指，能否將 MQe 配置成 HTTP Servlet，而由 HTTP 伺服器本身來負責接聽進入的連線要求。

MQe 應用程式不會直接知道佇列管理程式是使用何種連線樣式。不過樣式非常重要，它會影響到各方能夠使用哪些資源、哪些佇列管理程式可連接其他佇列管理程式、MQe 使用多少記憶體，以及哪些連線可同時存在。

配接器

配接器可將 MQe 對映到裝置介面。比方說：

- 通道利用通訊協定配接器，在 HTTP、原生 TCP/IP、UDP 及其他通訊協定上執行。
- 佇列利用欄位儲存體配接器，來連接儲存體子系統（如記憶體或檔案系統）。

配接器賦予 MQe 一種機制，來延伸裝置支援並進行版本控制。

註: C 程式碼庫與 MQe Java 程式碼庫不同，前者僅使用 HTTP 配接器。

撥號連線管理

裝置的撥號網路支援由裝置的作業系統來處理。

如果 MQe 試圖使用網路（例如，要傳送訊息），但 MQe 所在裝置並未連線，且網路堆疊不在作用中，則作業系統本身會啟動遠端存取服務 (RAS)。通常會顯示畫面給使用者來提供撥號連線設定檔。

在建立好連線之前，由作業系統來控制。所以，裝置使用者必須確定作業系統有適當的撥號連線設定檔可用。MQe 並不絕對支援撥號網路。

追蹤

您所執行的獨立程式如果可執行追蹤動作，追蹤就會啟動。

MQe 內建追蹤呼叫，可追蹤資訊、警告及錯誤狀況。

應用程式也可以直接呼叫追蹤並新增訊息，但只能使用 Java 程式碼庫。

由於系統會發佈追蹤處理常式必須實作哪個介面，因此解決方案可以實作這個介面，以便收集 MQe 及應用程式追蹤，進行交錯，並直接輸出到追蹤的收集點。產品程式中已包含數個追蹤處理常式。

此外，由於大部份 MQe 異常狀況都會送交應用程式來處理，因此應用程式異常狀況處理程式也可以將異常狀況交付追蹤。

事件日誌

這不適用於 C 程式碼庫。

MQe 提供可用來記載狀態的事件日誌機制和介面。例如，如果無法遞送非同步遠端佇列上的訊息，那麼就會記載警告訊息到日誌中。

依預設，日誌記載是寫入至 `system.out`，但是您可以截取這個記載，將它導向至其他地方。

MQe 事件日誌不會記載資訊資料，也不能用來復原訊息或佇列。

安全

MQe 提供全套整合式安全特性，可保護本端及轉送中的訊息資料。

MQe 的安全特性有三種：

本端安全

保護本端層次的訊息相關資料

訊息安全

保護起始 MQe 應用程式與接收 MQe 應用程式之間的訊息

佇列安全

保護起始佇列管理程式與目標佇列之間的訊息

本端和訊息安全可供 MQe 內部及 MQe 應用程式使用。MQe 佇列安全則僅為內部服務。

這三種 MQe 安全特性都是利用屬性（如 `MQeAttribute`）在保護訊息資料。屬性有可能是外部屬性或內部屬性，視安全特性種類而異。

每個屬性都可以包含：

驗證程式

提供其他的控制，以阻止未經授權的使用者存取本端資料

加密程式

控制所需的保護的強度

壓縮程式

將受保護的資料的大小最佳化

金鑰

藉由要求密碼來控制存取

目標實體名稱

要求目標佇列名稱

上述元素的用法因 MQe 安全特性種類而異，不過只要隨時呼叫訊息上附加的屬性，MQe 安全特性就會保護該訊息。

登錄：登錄是佇列管理程式相關資訊的主要儲存區。每一個佇列管理程式都有至少一個登錄。每個佇列管理程式都會利用登錄來保留它的：

- 佇列管理程式配置資料
- 佇列定義
- 遠端佇列定義
- 連線定義
- 使用者資料（包括與配置相依的安全資訊）

登錄資訊是使用配接器來儲存，通常是 MQeDiskFields 配接器。

專用登錄和認證： 本節不適用於 C 程式碼庫。

當每一個實體需要它自己的認證進行鑑別時，我們需要知道：

1. 如何執行登錄來取得認證
2. 在何處以安全方式管理認證

專用登錄會啓用一實體之專用認證的安全管理，公用登錄則管理整組公用認證。

專用登錄會提供一個基本登錄，內含安全或加密記號。它就像公用元素（如迷你認證）及專用元素（如私密金鑰）的安全儲存庫。

專用登錄僅容許已獲授權使用者存取專用元素。通常，僅合法的佇列管理程式使用者可以使用 PIN 存取登錄。然而，如果您完全不在乎安全問題，配置選項可讓您略過此情況。

專用登錄利用專用物件絕對不會離開專用儲存庫的方式來支援各服務程式，比方說，數位簽章、RSA 解密。它提供一個通用介面來隱藏基礎的裝置支援，目前這個基礎裝置支援只以本端檔案系統爲限。

自動登錄： MQe 有預設服務程式可支援自動登錄。當配置可鑑別實體時，會自動觸發這些服務程式，比方說，當啓動佇列管理程式之時，或定義新佇列時。這兩種情況都會觸發登錄，並建立新認證，然後儲存在別實體專用登錄中。因此，自動登錄會提供簡單機制，來建立訊息層次保護的認證。

自動登錄步驟包括：

1. 產生新的 RSA 金鑰組
2. 保護私密金鑰並將它儲存在專用登錄中
3. 將新認證要求中的公開金鑰包裝進預設的迷你認證伺服器裡

迷你認證伺服器配置完成可供使用後，該伺服器會把實體的迷你認證連同自己的迷你認證一起傳回。這些伺服器及受保護的私密金鑰都會儲存在實體的專用登錄中，作為它的新認證。

公用登錄和認證抄寫： MQe 有預設服務程式可啓動 MQe 元件來共用迷你認證。MQe 公用登錄將迷你認證存放在可公開存取的儲存庫中。就像行動電話的個人電話簿，只是這裡存放的是常常登錄之可鑑別實體的迷你認證，而不是電話號碼。

公用登錄並不絕對是被動的。假設公用登錄中並沒有您輸入的迷你認證，則如果公用登錄已配置有效的起始伺服器元件，它就會自動試著從起始伺服器的公用登錄中提取所要求的迷你認證。這些服務程式可提供智慧型自動化迷你認證抄寫服務，以便適時取得正確的迷你認證。

登錄服務的應用： MQe 佇列管理程式雖會利用使用專用和公用登錄服務的好處，但不會限制這些服務的存取。MQe 解決方案可以定義及管理它們自己的實體，如使用者。然後，您可以利用專用登錄服務來自動登錄和管理新實體的認證，以及利用公用登錄服務，將公用認證提供到任何有需要之處。

預設迷你認證簽發服務： SupportPac MQe 伺服器支援內含一「WTLS 迷你認證伺服器」，供您單獨免費下載。

這個軟體套件提供 WTLS 認證的認證簽發服務。您可以在這個認證簽發伺服器上配置佇列管理程式及佇列實體，來提供預設迷你認證簽發服務，以簽發 WTLS 認證來滿足專用登錄的自動登錄要求。您可以使用 MQe 認證簽發服務程式來設定及管理迷你認證簽發服務，謹慎控制哪些實體名稱可收到迷你認證。這個簽發服務的特性如下：

- 管理已登錄的可鑑別實體組
- 迷你認證簽發
- WAP WTLS 迷你認證儲存庫管理

安全介面： 有一個提供的選用性介面，可由自訂安全管理程式來實作。安全管理程式可以利用各個方法來授權或拒絕關聯於下列各項的要求：

- 新增及移除類別別名
- 定義配接器
- 對映檔案描述子
- 處理連線指令

自訂規則

使用者可利用規則來自訂某些主要 MQe 元件的行為。

必要時，MQe 會提供預設規則，但您可以將這些規則換成應用程式或安裝作業特有的規則，以符合客戶需求。

所支援的各種規則類型，其不同之處在於觸發它們的方式，以及它們能夠執行的動作。

規則含有邏輯，因此可以執行多種功能。

屬性規則： 屬性規則僅適用於 Java 程式碼庫。

每當試圖要變更狀態時，都會將控制權交給這個規則類別，比方說，變更：

- 鑑別程式
- 壓縮程式
- 加密程式

規則通常會接受或拒絕變更。

MQ 橋接規則： MQ 橋接規則僅適用於 Java 程式碼庫。

MQe 與 MQ 之間橋接程式碼的狀態如果有變化，就會動用此類規則。有一個獨立的橋接規則類別會決定該如何處理下列狀況：

- 接聽器無法將來自 MQ 的訊息遞送到 MQe 時（例如訊息太大或佇列不存在），要如何處理該訊息
- 伺服器實例建立好了之後，橋接器管理的佇列應從哪一狀態啟動
- 橋接器發現 MQ 同步佇列（此佇列乃供故障復原之用的持續儲存庫）有錯誤時，要如何處理（預設規則僅會顯示問題）
- 如何使用轉換程式將 MQe 訊息轉換成 MQ 訊息或轉換回來

佇列規則： 佇列遇到生命週期中下列關鍵時刻時，會呼叫此類規則：

- 有訊息新增至佇列時，此類規則會查看是否超出臨界值（例如，訊息數量及大小是否超出限制）。
- 開啓或關閉佇列時。
- 從佇列管理程式移除佇列時。原生 C 程式碼庫不適用於此。

- 佇列上的訊息已超出佇列或訊息本身的期限間隔。

佇列管理程式規則： 佇列管理程式遇到生命週期中下列關鍵時刻時，會呼叫此類規則：

- 佇列管理程式開啓時，此類規則會啓動背景計時器執行緒，以進行計時動作。
- 佇列管理程式關閉時，此類規則會終止背景計時器執行緒。
- 佇列管理程式的擱置訊息開始傳輸時。

類別

本節不適用於 MQe 原生 C 程式碼庫。

MQe 為某些功能提供了幾個類別選項，您可以自訂 MQe 行爲，來符合特定應用程式的需求。在某些情況下，會產生類別介面的文件，以便於發展其他替代方案。下表簡單地說明各種可能性。您可以明確地識別類別，也可以利用別名來識別類別。

註： 有些類別 C Bindings API 沒有提供。正確的支援類別清單請參閱 Java Programming Reference 及 C Programming Reference。

MQe 已經自動幫許多類別定好別名。請參閱 Java Programming Reference，網址 com.ibm.mqe.MQe.alias。

表 4. 類別選項

類別	提供的替代方案	產生介面文件	MQe 套件	如何實作
管理	否	是		
鑑別程式	是	否	com.ibm.mqe.attributes	延伸 com.ibm.mqe.MQeAuthenticator
通訊配接器	是	是	com.ibm.mqe.adapters	延伸 com.ibm.mqe.adapters.MQeCommunicationsAdapter
通訊樣式	是	否		
壓縮程式	是	否	com.ibm.mqe.attributes	延伸 com.ibm.mqe.MQeCompressor
加密程式	是	否	com.ibm.mqe.attributes	延伸 com.ibm.mqe.MQeCryptor
事件日誌	提供範例	是		實作 com.ibm.mqe.MQeEventLogInterface
訊息	否	是	com.ibm.mqe	延伸 com.ibm.mqe.MQeMsgObject
佇列儲存體	是	否		通常會使用別名 MsgLog: 定義的預設值。
規則	提供預設類別	是		延伸 com.ibm.mqe.MQeRule
儲存體配接器	是	是	com.ibm.mqe.adapters	延伸 com.ibm.mqe.adapters.MQeAdapter
追蹤	提供範例	是	com.ibm.mqe.trace	

應用程式載入：

註： 本節不適用於 C 程式碼庫。

載入 MQe 佇列管理程式時，起始應用程式必須將其他應用程式載入至 JVM。

標準的 Java 設備及 MQe 提供的類別載入器皆可進行此類載入。

載入之後：

- 多個應用程式就可以同時利用同一台 JVM 中的同一個佇列管理程式。

- 您也可以使用多台 JVM，但這樣的話，每台 JVM 就都要有自己的佇列管理程式，且每個佇列管理程式都要有自己唯一的名稱。

MQe SupportPacs

MQe 產品系列提供各式工具，來開發、部署和管理 MQe 傳訊及佇列解決方案。這個系列包括：

1. **MQe 授權產品**，可取自 IBM® 實體媒體或由 Web 下載：

<http://www.ibm.com/software/integration/wmqe/>

授權產品包括：

- MQe Java 類別
- Helper 類別
- MQe C Binding 檔案及原生 C 程式碼庫
- 應用程式原始碼範例
- 公用程式
- 參考手冊
- 授權資訊

實際「程式產品」也包括在特定平台上非開發性地使用產品的權利。如果要在更大型的機器上使用，或要搭配 MQ 橋接器來使用，請額外購買產能單位。

2. **MQe SupportPacs**，可自 Web 下載：

<http://www.ibm.com/software/integration/support/supportpacs/>

或

<http://www.ibm.com/software/integration/wmqe/>

MQe SupportPacs 內的管理工具，在應用程式各個開發和首展階段中，都扮演著極重要的角色。它們比授權產品所提供的公用程式還複雜，主要協助入門、配置、檢查試用網路，以及管理正式運作系統。

EA01: WebSphere MQ Everyplace - XML conversion utility

此軟體可將 MQeFields 物件轉換為 XML 表示法或轉換回來。

EP02: WebSphere MQ Everyplace - DB2® Adapter User Guide

將持久性儲存體延伸至 DB2 資料庫內。

ES06: WebSphere MQ Everyplace Server Support

集合 MQe_Explorer、MQe_Script、MQe_MinicertServer 及 MQe_Service，提供 MQe 的管理及安全功能。

MS0B: MQSeries® Java classes for PCF

本 Java 程式碼可支援 PCF 訊息。其用法請參閱『MS0B - MQSeries Java classes for PCF』

MS0B - MQSeries Java classes for PCF

PCF 訊息乃 MQ 佇列管理程式所使用的管理訊息。本 SupportPac 中的 Java 程式碼可支援 PCF 訊息。

下載及安裝後，請將 `com.ibm.mq.pcf.jar` 檔放到類別路徑環境變數下，您就可以存取 Java 類別，來動態操作 MQ 資源了。將 PCF 訊息與 MQe 管理訊息結合之後，您就可以將橋接器資源與 MQe 佇列管理程式上相對應的資源進行完整的程式化配置。`examples.mqbridge.administration.programming.AdminHelperMQ` 類別中的程式碼範例會示範如何操作，請將程式碼範例與 `examples.mqbridge.administration.programming.MQAgent` 搭配使用。程式碼範例已新增至 `examples.awt.AwtMQeServer` program，您只需選取 **View** → **Connect local MQ default queue manager**，即可：

- 確保已有橋接器物件，必要時可自行建立。

- 查詢預設 MQ 佇列管理程式中的內容。
- 試圖將該佇列管理程式連接到執行中的 MQe 佇列管理程式。
- 確保已有代表預設 MQ 佇列管理程式的 proxy 物件，必要時可自行建立。
- 確保已有 MQe 用戶端連線，並確保已有相對應的 MQ 伺服器連線通道，必要時可自行建立。
- 確保 MQ 佇列管理程式上已有同步佇列。
- 確保 MQ 上已有傳輸佇列，必要時可自行建立。
- 確保目前的 MQe 佇列管理程式已配置相對應的 MQ 傳輸佇列，必要時可自行建立。
- 確保所有橋接器資源皆已啟動。
- 確保 MQ 佇列管理程式上已有測試佇列，必要時可自行建立。
- 確保已有相對應的 MQe 橋接佇列，可與該測試佇列相互參照。
- 傳送測試 MQeMQMsgObject 到測試佇列，確保配置順利運作。
- 自測試佇列取得測試 MQeMQMsgObject，確保配置順利運作。

規劃實作

授權

您要有授權才能部署 MQe 應用程式

MQe 是一個工具集，供使用者撰寫 MQe 應用程式，並建立執行這些應用程式的環境。在部署本產品或部署使用本產品的應用程式之前，請確定您有足夠的授權。

1. 在**伺服器**上使用「本程式」時，我們以**處理器授權單位**來計價。在伺服器上使用「本程式」時，每一台要執行「本程式」的伺服器上的每一個處理器或對稱多重處理器，都要有一個**處理器授權單位**。您需要的是哪一種**處理器授權單位**及**裝置使用授權**，取決於「本程式」是在銷售點（即零售）設備上執行，還是在其他類型的電腦上執行。在零售設備上執行會需要**零售**伺服器授權，在其他（非零售）設備上執行則需要**網路**伺服器授權。
2. 在獨立的**用戶端**裝置上使用「本程式」則需要其他的**裝置使用授權**，除非該裝置已包含在底下 3. 所說明的**網路**伺服器授權裡。
3. 每一個**網路**伺服器授權都有授權使用限制，即在同一家商業企業或組織中，「本程式」的同一個伺服器拷貝不得用於一百 (100) 台以上的用戶端裝置。

詳細限制資訊請參閱 <http://www.ibm.com/software/integration/wmqe/>。

您必須有**裝置平台使用授權**（記錄在「授權證明書」文件中且可有效支援您使用 MQe）才能在指定的用戶端平台上使用產品（不得用於開發和測試程式碼）。這些授權不表示使用者有權使用「MQe 橋接器」，或有權在特殊伺服器平台上執行，特殊伺服器平台意指 IBM 公佈之 MQe 計價群組清單上所列的伺服器平台，請見下列 URL：

詳細限制資訊請參閱 <http://www.ibm.com/software/integration/mqfamily/>。

使用什麼機器

應使用什麼機器來開發及部署應用程式

您需要：

- PC，用以撰寫及編譯應用程式。

建議使用 Windows 系統，以便您執行各種 MQe SupportPacs，特別是 MQe Explorer，對您的開發工作極有助益。

該電腦必須可使用網際網路，以便您下載 MQe、SupportPacs、文件等。

- 至少一台測試用的電腦或裝置，請使用您日後要部署該應用程式的電腦或裝置。
- 將裝置連接至開發用 PC 所需之介面及接線。

使用哪一個程式碼庫

「MQe 應用程式設計介面 (API)」是 MQe 的程式設計介面。它支援 Java 及 C 兩種語言。

Java 版本可使用 MQe 的所有功能。詳細類別、方法及程序請參閱 Java Programming Reference。本資訊中心將提供多個 MQe 程式設計範例。

C 支援有三個版本：

原生 C 程式碼庫可存取 MQe 功能的主要子集。C 程式碼庫是一種裝置佇列管理程式，但它：

- 不支援儲存及轉遞佇列或橋接佇列
- 僅支援 HTTP 配接器
- 僅支援 RLE 壓縮程式
- 僅支援 RC4 加密程式
- 僅支援 *MAttribute* 及本端安全功能

詳細的方法和程序請參閱 C Programming Reference。本資訊中心將提供多個 C Binding 的程式設計 MQe 範例。

「原生 C」程式碼庫的功能完備之前，都將使用 **C Binding**。C Binding 可存取 MQe 功能的主要子集。詳細的方法和程序請參閱 C Programming Reference。C Bindings 的程式設計 MQe 範例請參閱 C Bindings Programming Reference。

MQe 開發周期

MQe 的使用範圍極廣，其安裝、配置和部署都跟 MQ 系列的其他成員不同。MQe 的導入可分為三個階段：

1. 開發和原型設計階段

有些 MQe 開發授權條款可免費安裝和使用 MQe。MQe 應用程式是利用 JMS API 或 MQe Java 及 C API 的函數開發而來。建立及使用 MQe 佇列管理程式時請考慮下列事項。這會影響到您應用程式的實作方式。

- 一台 JVM 或一個 C 程序中，一次僅能動用一個 MQe 佇列管理程式。如果您試圖啟動第二個佇列管理程式，MQe 內會進行檢查並擲出錯誤。如果有多個應用程式要使用同一個佇列管理程式，這些應用程式必須位於同一台 JVM 上。在同一台 JVM 上執行多個應用程式時，應用程式可使用 `MQeQueueManager.getDefaultQueueManager()` 方法，來檢查佇列管理程式是否已在執行中。單一機器上可以有無限多個應用程式及佇列管理程式，但請考慮預期效能及系統資源。
- 單一訊息儲存檔上只能啟動一個 MQe 佇列管理程式。如要啟動多個應用程式實例，使用圖形式使用者介面會比較容易。如此會比較容易啟動單一訊息儲存檔上執行之單一佇列管理程式內的多個實例。但這可能導致訊息儲存檔含混不清，特別是在使用非同步佇列時。

C C API 可運用與用戶端佇列管理程式的角色相當的功能。您可以用 Platform Invocation Services 來呼叫 MQe C DLL，這樣就能使用 .NET 環境。您必須將所需的 DLL 下載到裝置上。

非常建議您在原型設計階段中，直接在正式作業的網路上，用正式作業的資料來進行測試。如此將有助於您設定正確的效能預期以及評估 MQe 的預設通訊設定。

IBM 的支援不包含在開發授權中。不過，部署授權提供應用程式開發期間及後續的支援（請參閱下文）。

2. 部署階段

部署階段是要使用您開發出來的應用程式，根據 MQe 授權條款，您必須有產能單位才能使用產品。您必須獲得 IBM 同意，或者使用者必須已獲得授權，才能分送 Java 類別及 C API。否則，Java 使用者將必須自行訂必要的類別，而 C 使用者將必須將 MQe 複製至裝置。

3. 管理階段

接下來，您開始在網路中使用 MQe 佇列管理程式時，您必須使用工具來檢查和管理佇列管理程式。「國際程式授權合約」中有 MQe 的支援條款。

支援層次

整個導入生命週期中，IBM 有不同的平台支援層次。就 MQe 的產能單位以及 Category 3 SupportPacs 而言，IBM 的支援層次可分為下列三種：

- 支援安裝和應用程式開發的平台：
 - 接受有關安裝、應用程式開發及使用方面的問題報告
- 允許應用程式部署但沒有直接支援的平台：
 - 如果要在支援的平台上重新產生，可能必須報告問題
- 支援應用程式部署的平台：
 - 接受因應用程式部署而產生的問題報告

體驗 MQe

MQe 的入門途徑有許多種。

- 要熟悉本產品及其概念最有效率的方法，就是先啟動和執行佇列管理程式，並設定簡單的 MQe 網路。
- 撰寫一個簡單應用程式深入研究產品細節則是非常好的準備工作。
- 剛開始時，您對 MQ 系列其他成員的瞭解對您並沒有很大幫助。但如果您之後要使用橋接功能，這點就很重要了。

因此，我們建議新使用者好好瞭解本文件此一簡介部份的重要概念。

如果您的機器是用 Windows 作業系統，請下載「MQe 伺服器支援」SupportPac ES06（內含 MQe_Explorer）並遵循 MQe 的啟動說明。您並不需要事先安裝授權產品，但如果沒有事先安裝的話，您將受到授權條款的限制。

使用 MQ 與 MQe

簡介

即使沒有 MQ 伺服器或網路，MQe 網路也可以獨立存在。但通常在實務上，MQe 還是會與現有的 MQ 安裝搭配使用。

因為這樣不僅能將 MQ 延伸到新平台和裝置，還能獲得更高階的功能（如佇列或訊息型的安全及同步傳訊功能）。

對 MQe 應用程式而言，MQ 佇列和佇列管理程式就像其他的遠端佇列及佇列管理程式。不過，由於這些佇列無法直接透過 MQe 連線和 MQe 佇列管理程式來存取，而必須藉助 MQe 闢道之手，許多功能也因而受限。

閘道可以直接或間接地透過 MQ 用戶端通道，傳送訊息給多個 MQ 佇列管理程式。間接連線時，訊息會透過 MQ 用戶端通道傳送到中間 MQ 佇列管理程式，再繼續透過 MQ 訊息通道傳送給目標佇列管理程式。

閘道（橋接器）至 MQ

本節不適用於 C 程式碼庫。

MQe 支援 MQ 橋接器，後者就像 MQe 與 MQ 網路之間的連接介面。

橋接器用 MQ Java 用戶端來連接一或多個 MQ 佇列管理程式，使訊息能夠往返於 MQe 和 MQ 之間。

MQe 現行版本中：

- 建議在每台伺服器上安裝一個這樣的橋接器
- 每個橋接器聯結多個 MQ 佇列管理程式 *proxy*（MQ 佇列管理程式的定義）
- 每個要跟 MQe 通訊的 MQ 佇列管理程式，都要有一個佇列管理程式 *proxy* 定義
- 每個定義可聯結一或多個用戶端連線服務，而每一個服務代表一個連線，連接到單一個 MQ 佇列管理程式。
- 每個服務會透過不同的 MQ 伺服器連線連接到佇列管理程式，您可以選擇性地為每個服務設定不同的內容，如使用者跳出程式或使用者埠等。

訊息轉換

要送到 MQ 的 MQe 訊息會在通過橋接器時，被預設的轉換程式或目標佇列指定的轉換程式，轉換成 MQ 格式。自訂轉換程式彈性較大，例如，您可以利用 MQe 訊息類別的子類別，來代表 MQe 網路上的特定訊息類型。閘道上的轉換程式將訊息轉換成 MQ 格式時，可以比對欄位與 MQ 值，把具有特定子類別特徵的資料放到正確的欄位裡。

從 MQe 到 MQ 的預設轉換程式無法利用子類別資訊，但它有許多其他用途。它有下列特性：

- 從 MQe 流向 MQ 的訊息：

從 MQe 到 MQ 的預設轉換程式可搭配 *MQeMQMsgObject* 類別來使用。這個類別囊括了所有 MQ 訊息的標頭欄位。

在 *MQeMQMsgObject* 的輔助下，應用程式可以用 *set()* 方法來設定值。*MQeMQMsgObject* 本身或衍生自 *MQeMQMsgObject* 的物件通過預設的 MQe 轉換程式（即 *MQeBaseTransformer*）時，*MQeBaseTransformer* 會取得 *MQeMQMsgObject* 內部的值，並在 MQ 訊息內設定相對應的值，例如，將優先值直接複製到 MQ 訊息。

如果所傳送的訊息不是 *MQeMQMsgObject*，也不是衍生自 *MQeMQMsgObject* 類別，轉換程式就會將整個 MQe 訊息複製到 MQ 訊息的主體欄位中。這就是所謂的通過。MQ 訊息標頭中的訊息格式欄位會指出，該 MQ 訊息裡有一個 MQe 通過格式的訊息。

- 從 MQ 流向 MQe 的訊息：

訊息從 MQ 流向 MQe 時，其實就是把上述流程反向操作。預設轉換程式會檢查 MQ 標頭的訊息類型欄位，據此採取行動。

如果 MQ 標頭指出這是個通過格式的 MQe 訊息，MQ 訊息主體就會被重新建構成原始的 MQe 訊息，並放回 MQe 網路。

如果不是通過 MQe 訊息，轉換程式會擷取 MQ 訊息標頭內容，將其組合成 *MQeMQMsgObject*。MQ 訊息主體也會被當成簡單的位元組欄位，放進 *MQeMQMsgObject* 中。完成之後再將 *MQeMQMsgObject* 放上 MQe 網路。

此 MQeMQMsgObject 類別和預設的轉換程式行為表示：

- MQe 訊息從 MQ 網路流到 MQe 網路時，內容不會改變。
- MQ 訊息從 MQe 網路流到 MQ 網路時，內容不會改變。
- MQe 應用程式可搭配現有的 MQ 應用程式來運作，且不會改變這些 MQ 應用程式。

功能

MQ 遠端佇列可以同步放置多個 MQe 佇列管理程式中的 MQe 訊息。

所有其他傳訊作業都必須是非同步。

您不能將 MQe 管理訊息傳送給 MQ 佇列管理程式。因為 MQ 沒有管理佇列，MQ 與 MQe 的管理訊息格式也不一樣。

相容性

即使沒有 MQ，MQe 網路也可以獨立於存在。但此二者常需要互相搭配，才能滿足應用程式的需求。MQe 可以整合到現存的 MQ 網路中，其相容的部份大致如下：

定址和命名：

- 利用佇列管理程式或佇列位址的相同定址語意
- 通用 ASCII 名稱空間

應用程式：

- MQe 可支援現有的 MQ 應用程式，毋需改變應用程式。

連線：

- MQe 開道可使用 MQ 用戶端通道。

訊息交換和內容：

- MQe 與 MQ 之間可進行訊息交換
- 隱藏式訊息網路（MQe 或 MQ 訊息毋需變更即可進入彼此的網路）
- 支援彼此 MQ 訊息標頭的識別欄位
- Once and once-only 確定訊息遞送

MQe 並不支援 MQ 的所有功能。在環境、作業系統和通訊考量之外，一些更重要的差異如下：

- 沒有叢集支援。
- 沒有分送清單支援
- 沒有群組化或區段化的訊息
- 沒有平衡資料流量或暖機待用功能
- 沒有參照訊息
- 沒有報告選項
- 沒有共用佇列支援。
- 沒有觸發
- 沒有工作單元支援，沒有 XA 對等關係。
- 不同的可調整性及效能特性

不過您可以利用 MQe 的功能，或利用子類別，或取代提供的類別，或利用規則、介面及其他產品內建的自訂功能，在 MQe 內完成許多應用程式作業。

確定遞送

雖然 MQe 和 MQ 都提供確定遞送，但它們提供的確定分屬不同的層次。

- 當訊息從 MQe 流向 MQ，僅在使用 *putMessage* 及 *confirmPutMessage* 的組合時，才能確定訊息轉送。
- 當訊息從 MQ 流向 MQe 時，必須將 MQ 訊息定義為具有持續性，才能確定這項轉送作業。

進一步資訊

MQ 的相關資訊

您可參考下列 MQ 相關書籍：

WebSphere MQ: An Introduction to Messaging and Queuing (GC33-0805)

本書簡要地向您介紹 MQ、其運作方式，以及其如何解決一些典型的互運性問題。

WebSphere MQ: Quick Beginnings series

MQ 支援的每個平台都有相關的「MQ 快速入門」書籍，指導您如何在特定平台上規劃及安裝 MQ。

網站

MQe 首頁位於：

<http://www.ibm.com/software/integration/wmq/>

首頁中的鏈結可協助您：

- 進一步瞭解 MQe 的功能及好處
- 取得訓練及認證的相關資訊
- 取得 PDF 及 HTML 格式的 MQe 手冊
- 下載最新的升級版及試用碼

到下列網頁選擇 *WebSphere MQ Everyplace* 即可下載 MQe SupportPacs：

<http://www.ibm.com/software/integration/support/supportpacs/>

如果您想進一步瞭解 MQ，請至 MQ 首頁：

<http://www.ibm.com/software/integration/wmq/>

或 MQ 系列首頁：

<http://www.ibm.com/software/integration/mqfamily/>

MQ 產品系列書庫請至：

<http://www.ibm.com/software/integration/websphere/library/books/>

翻譯文件

MQe 簡介書籍已翻譯成英文以外的語言。請至 MQ 書庫網站下載翻譯文件：

<http://www.ibm.com/software/integration/websphere/library/>.

Newsgroups

這些 Newsgroups 都在 news.software.ibm.com，也將可以在許多其他公用 Newsgroups 中找到。

MQe:

- ibm.software.websphere.mqeveryplace

MQ:

- ibm.software.websphere.mq
- ibm.software.websphere.mq.administration
- ibm.software.websphere.mq.programming

其他相關：

- ibm.software.websphere.mqintegrator
- ibm.software.websphere.studio
- ibm.software.websphere.studio.various

MQe 認證

現在已經有訓練及認證可用。相關的詳細資訊，請造訪下列網址：

<http://www.ibm.com/software/integration/websphere/education/>

注意事項 & 商標

注意事項

本資訊是針對 IBM 在美國所提供之產品與服務開發出來的。IBM 不一定會在其他國家提供本書中所提的各項產品、服務或功能。欲知您所在地區是否有這些產品於服務，請聯繫當地的 IBM 業務代表。本文提及 IBM 產品、程式或服務時，並不試圖表示或暗示您僅能使用該項 IBM 產品、程式或服務。只要不侵犯 IBM 的智慧財產權，您可以用任何功能相當的產品、程式或服務，來代替 IBM 的產品、程式或服務。不過，其他非 IBM 產品、程式、或服務在運作上的評價與驗證，其責任屬於使用者。

本文題材可能涉及 IBM 的專利或專利申請案。本書使用者並不享有前述專利之任何授權。您可以用書面方式來查詢授權，來函請寄到：

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

下列段落若與該國之法律條款抵觸，即視為不適用：國際商業機器公司僅以『現狀』提供本書，而不提供任何明示或默示之保證（包括但不限於可售性或符合特定效用的保證）。若有些地區在某些交易上並不允許排除上述保證，則該排除無效。

本資訊中可能會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 得隨時改進並（或）變動本書中所提及的產品及（或）程式，毋需另行通知。

本資訊中任何對非 IBM 網站的敘述僅供參考，IBM 對該網站並不提供保證。這些網站上的資料並非 IBM 產品的相關資料，請自行負責使用該網站資料的後果。

IBM 得以以其認為適當的方式，使用或散布您所提供的任何資訊，而無需對您負責。

本程式之獲授權者若希望取得相關資料，以便使用下列資訊者可洽詢 IBM。其下列資訊指的是：(1) 獨立建立的程式與其他程式（包括此程式）之間更換資訊的方式 (2) 相互使用已交換之資訊方法若有任何問題請聯絡：

IBM United Kingdom Laboratories,
Mail Point 151,
Hursley Park,
Winchester,
Hampshire
England
SO21 2JN

上述資料之取得有其特殊要件，在某些情況下必須付費方得使用。

IBM 乃根據與您之間的「IBM客戶合約」、「IBM 國際程式授權合約」或任何同等合約之條款，來提供本書所說的授權程式及其所有適用的授權資料。

商標

下列術語是國際商業機器公司在美國和（或）其他國家或地區的商標。

AIX Everyplace IBM IBMLink iSeries MQSeries SupportPac WebSphere z/OS zSeries

Microsoft、Windows、Windows NT 和 Windows 標誌是 Microsoft Corporation 在美國和（或）其他國家或地區的商標。

Java 和所有標有 Java 商標和標誌是 Sun Microsystems, Inc. 在美國和（或）其他國家或地區的商標或註冊商標。

Linux 是 Linus Torvalds 在美國和（或）其他國家或地區的商標。

其他公司、產品和服務名稱，可能是第三者的商標或服務標記。

名詞解釋

此名詞解釋說明本書使用之專用詞彙，以及具有特殊意義的日常詞彙。在某些情況下，一個詞彙可能有不只一個適用的定義，不過，它會提供這個詞彙在本書內所採用的特定意義。

如果您找不到所要的詞彙，請搜尋電子檔或查看印刷本索引，或查閱 *IBM Dictionary of Computing*, New York: McGraw-Hill, 1994。

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

A

應用程式設計介面 (application programming interface, API)

應用程式設計介面由若干函數和變數組成，程式設計師可以在他們的應用程式中使用這些函數和變數。

非同步傳訊 (asynchronous messaging)

這是一種各程式將訊息放到訊息佇列中來相互通訊的方法。在非同步的傳訊中，傳送端的程式不需要等待訊息的回答，可以繼續它本身的處理程序。請對照同步傳訊 (synchronous messaging)。

驗證程式 (authenticator)

這是一種驗證訊息之傳送者和接收者的程式。

B

橋接器 (bridge)

MQe 佇列管理程式利用此附加元件來與 MQ 通訊。請參閱 MQe 佇列管理程式。

C

通道 (channel)

請參閱動態通道 (*dynamic channel*) 和 MQI 通道 (*MQI channel*)。

通道管理程式 (channel manager)

此 MQe 物件支援端點之間的邏輯多重並行通訊管線。

類別 (class)

資料和操作資料之方法的封裝集合。類別可以案例化，以產生類別案例物件。

用戶端 (client)

用戶端是 MQ 中的執行時期元件，協助本端使用者應用程式將訊息傳送到伺服器。

壓縮程式 (compressor)

這是壓縮訊息以縮小要傳輸之資料量的程式。

連線 (connection)

鏈結 MQe 裝置，雙向傳輸同步和非同步訊息/回應。

加密程式 (cryptor)

這是為訊息加密以提供傳輸安全的程式。

D

裝置平台 (device platform)

此一小型電腦可以只將 MQe 當成用戶端來執行，意即，只與裝置佇列管理程式共同執行。

裝置佇列管理程式 (device queue manager)

請參閱 MQe 佇列管理程式。

E

封裝 (encapsulation)

此一物件導向的程式設計技術可使物件資料成為專用或受保護的資料，讓程式設計師只能透過方法呼叫來存取和操作資料。

G

閘道 (gateway)

具備 MQ 橋接功能、可執行 MQe 閘道佇列管理程式的各種大小型電腦。請參閱 MQe 佇列管理程式。

閘道佇列管理程式 (gateway queue manager)

具備接聽器及橋接器的佇列管理程式。請參閱 MQe 佇列管理程式。

H

超文字標記語言 (Hypertext Markup Language, HTML)

這是一種用來定義要呈現於全球資訊網之資訊的語言。

I

實例 (instance)

物件。當類別案例化以產生物件時，這個物件是類別的一個案例。

介面 (interface)

只含有抽象方法，不含案例變數的類別。介面提供一組若干不同類別之子類別所能實作的共用方法。

網際網路 (internet)

共用資訊的協力公用網路。在實體上，Internet 會利用目前現存的所有公用電信網路之總資源的子集。在技術上，將網際網路區分為協力公用網路的因素，在於它使用一組稱為 TCP/IP（傳輸控制通訊協定/網際網路通訊協定）的通訊協定。

J

Java Development Kit (JDK)

Sun Microsystems 分送給 Java 開發人員的一套軟體套件。內含 Java 直譯器、Java 類別及 Java 開發工具：編譯器、除錯器、反組譯器、appletviewer、Stub 檔產生器以及文件產生器。

Java 名稱及目錄服務 (Java Naming and Directory Service, JNDI)

Java 程式設計語言指定的 API。提供名稱及目錄功能，供以 Java 程式設計語言寫成的應用程式使用。

L

輕量型目錄存取通訊協定 (Lightweight Directory Access Protocol, LDAP)

用來存取目錄服務的主從式通訊協定。

M

訊息 (message)

在訊息佇列作業應用程式中，意指程式之間傳送的通訊。

訊息佇列 (message queue)

請參閱佇列 (queue)。

訊息佇列作業 (message queuing)

這是一種程式設計技術，在這裡，應用程式內的每個程式都是將訊息放到佇列中，來與其他程式通信。

方法 (method)

此一物件導向程式設計術語專指函數或程序。

MQ 橋接器 (MQ bridge)

有關佇列管理程式的電腦即可與 MQ 通訊。請參閱 MQe 佇列管理程式。

MQ 及 MQ 系列 (MQ and MQ family)

即 WebSphere MQ，包含下列產品：

- **WebSphereMQ Workflow** 將需要人力及應用程式的商業程序自動化，簡化整個企業的整合。
- **WebSphere MQ Integrator** 是一種訊息代理軟體，可根據規則提供即時、智慧型的訊息遞送及內容轉換/格式化。
- **WebSphere MQ Messaging** 可在各種桌面與各種大型電腦之間建立連線，提供高品質商業傳訊，其所支援的平台逾35 種。

MQ 傳訊 (MQ Messaging)

請參閱下列 **WebSphere MQ** 傳訊產品群組：

- **分散式傳訊**：支援 Windows NT 及 Windows 2000、AIX、iSeries、HP-UX、Solaris 等平台的 MQ
- **主機傳訊**：支援 z/OS 的 MQ
- **通用傳訊**：MQe

MQe 即 **WebSphere MQ Everyplace**，MQ 通用傳訊產品群組。

MQI 通道 (MQI channel)

可將 MQ 用戶端連接到伺服器系統中的佇列管理程式，並採用雙向方式來轉送 MQI 呼叫和回應。

O

物件 (object)

(1) 在 Java 中指類別中的實例。類別是事物群組的模型；物件是該群組之特定成員的模型。(2) 在 MQ 中指佇列管理程式、佇列或通道。

P

套件 (package)

Java 程式碼可透過 Java 套件來存取特定類別。一套件內的 Java 程式碼可以存取該套件內的所有類別，以及這些類別內的所有非專用方法或欄位。

個人數位助理 (personal digital assistant, PDA)

這是一種口袋大小的個人電腦。

專用 (private)

在類別之外，無法見到類別內的專用欄位。

受保護的 (protected)

只有在類別本身之內，或該類別的子類別內，或該類別所屬套件內，才能見到該類別內受保護的欄位。

公用 (public)

在任何位置中，都能見到公用類別或介面。只要是可見到類別之處，也都能見到該類別的公用方法或變數。

Q

佇列 (queue)

一種 MQ 物件。訊息佇列作業應用程式可以將訊息放到佇列中，或取得佇列中的訊息。

佇列管理程式 (queue manager)

佇列管理程式是一個系統程式，負責提供訊息佇列作業服務給應用程式。

遠端佇列管理程式 (remote queue manager)

遠端佇列定義中會用到此詞彙。遠端佇列定義會有一個目標本端佇列，而此詞彙意指擁有該目標本端佇列的遠端佇列管理程式。

裝置佇列管理程式 (device queue manager)

MQe 中的佇列管理程式，沒有接聽及橋接元件，因此只能傳送訊息，無法接收訊息。

伺服器佇列管理程式 (server queue manager)

MQe 中的佇列管理程式，可連接接聽器，透過接聽器即可接收訊息，亦可傳送訊息。

閘道佇列管理程式 (gateway queue manager)

MQe 中的佇列管理程式，可連接接聽器及橋接器，透過接聽器來接收及傳送訊息，透過橋接器來與 MQ 通訊。

R

登錄 (registry)

儲存佇列管理程式配置資訊。

S

伺服器 (server)

1. MQe 伺服器是一個已配置 MQe 接聽器的裝置，可回應主從架構設定中的資訊要求。
2. MQ 伺服器是一個佇列管理程式，負責提供訊息佇列作業服務給遠端工作站上執行的用戶端應用程式。
3. 廣義上，伺服器是一種程式，可回應主從架構中某兩個程式的資訊流模型中的資訊要求。
4. 用來執行伺服器程式的電腦。

伺服器佇列管理程式 (server queue manager)

此佇列管理程式具備接聽器，可接收及傳送訊息。請參閱 MQe 佇列管理程式。

伺服器平台 (server platform)

可將 MQe 當成伺服器或用戶端來執行的各種大小電腦。

servlet

一種僅能在 Web 伺服器上執行的 Java 程式。

子類別 (subclass)

子類別是繼承其他類別的類別。子類別會繼承其超類別的公用和受保護的方法和變數。

超類別 (superclass)

超類別是其他類別所繼承的類別。子類別可以使用超類別的公用和受保護的方法與變數。

同步傳訊 (synchronous messaging)

這是一種各程式將訊息放到訊息佇列中來相互通信的通信方法。在同步傳訊中，傳送端的程式先等待訊息的回答，再繼續它本身的處理程序。請對照非同步傳訊 (*asynchronous messaging*)。

T

傳輸控制通訊協定/網際網路通訊協定 (Transmission Control Protocol/Internet Protocol, TCP/IP)

這是一組支援本端和廣域網路之對等式連線功能的通訊協定。

轉換程式 (transformer)

用於執行資料或訊息重新格式化的程式碼片段。

W

Web 請參考「全球資訊網 (World Wide Web)」。

Web 瀏覽器 (Web browser)

這是一種會建立全球資訊網所分送之資訊的格式並加以顯示的程式。

全球資訊網 (World Wide Web, WWW)

「全球資訊網」是以一組通訊協定為基礎的 Internet 服務，可讓特定配置的伺服器電腦採用標準方式，透過 Internet 來分送文件。

索引

索引順序以中文字，英文字，及特殊符號之次序排列。

〔四劃〕

介面, 安全 18
介面, 程式設計 1, 3
公用登錄 17

〔五劃〕

加密 16
本端佇列 10
用 MQe 管理 12
用戶端通道 8

〔六劃〕

名詞解釋 28
安全介面 18
安全, MQe 16
自動登錄 17

〔七劃〕

佇列規則 18
佇列管理程式 11, 15
佇列管理程式作業 12
佇列管理程式規則 19
佇列管理程式, MQe 11
佇列, MQ 橋接器 10
佇列, MQe 9
佇列, 本端 10
佇列, 遠端 10
佇列, 儲存及轉遞 10
作業, 佇列管理程式 12

〔八劃〕

事件日誌 16
注意事項, 法務 27
法務注意事項 27

〔十劃〕

訊息 8
訊息遞送, 確定 26
訊息轉換 24
訊息, 管理 12

起始伺服器佇列 10
迷你認證簽發服務 17
追蹤 MQe 15
配接器, MQe 15
配置 18
配置範例 13
配置, 範例 13

〔十一劃〕

商標 28
專用登錄 17
接聽器 15
規則, MQe 18
規則, 自訂 18
通訊 15
通道 13
通道, 用戶端 8
連接 MQ 24
連線 8, 13
連線管理程式 15
連線樣式 15
連線, 動態 8

〔十二劃〕

登錄 16
登錄, MQe 8
登錄, 公用 17
登錄, 專用 17
程式設計介面 1, 3
詞彙 28

〔十三劃〕

裝置, MQe 8
載入應用程式 19

〔十四劃〕

監視 13
管理訊息 12
與 MQ 的相容性 25
認證抄寫 17
遠端佇列 10

〔十五劃〕

撥號連線管理 15
確定訊息遞送 26

〔十六劃〕

橋接器, MQ 24

〔十七劃〕

儲存及轉遞佇列 10
壓縮 16
應用程式, MQe 7
應用程式, 載入 19
總覽 6

〔十八劃〕

轉換程式 24

〔十九劃〕

類別, MQe 19

〔二十一劃〕

屬性規則 18

〔二十二劃〕

歡迎使用 1

A

API 1, 3

M

MQ PCF 訊息 20
MQ 系列 6
MQ 傳訊 6
MQ 橋接佇列 10
MQ 橋接規則 18
MQ 橋接器 24
MQe 8
MQe 安全 16
MQe 佇列 9
MQe 佇列管理程式 11
MQe 配接器 15
MQe 規則 18
MQe 登錄 8, 16
MQe 裝置 8
MQe 管理 12

MQe 應用程式 7
MQe 簡介 6

MQe 類別 19
MQ, 相容 25

MQ, 連接 24