

9.4

IBM MQ Szenarios

IBM

Hinweis

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen unter „Bemerkungen“ auf Seite 233 gelesen werden.

Diese Ausgabe bezieht sich auf Version 9 Release 4 von IBM® MQ und alle nachfolgenden Releases und Modifikationen, bis dieser Hinweis in einer Neuauflage geändert wird.

Wenn Sie Informationen an IBMsenden, erteilen Sie IBM ein nicht ausschließliches Recht, die Informationen in beliebiger Weise zu verwenden oder zu verteilen, ohne dass eine Verpflichtung für Sie entsteht.

© **Copyright International Business Machines Corporation 2007, 2024.**

Inhaltsverzeichnis

Szenarios.....	5
Erste Schritte mit IBM MQ.....	5
Lösung planen.....	5
Lösung implementieren.....	7
Nächste Schritte.....	19
Punkt-zu-Punkt-Szenario.....	20
Lösung planen.....	20
Lösung implementieren.....	22
Punkt-zu-Punkt-Topologie schützen.....	29
Streaming-Warteschlangen.....	32
Konfiguration der Streaming-Warteschlangen.....	33
Streaming in ferne und Aliaswarteschlangen.....	35
Einschränkungen für Streaming-Warteschlangen.....	36
Datenstromwarteschlangen und Transaktionen.....	36
Streaming in und aus Clusterwarteschlangen.....	37
Streaming-Warteschlangen zum Speichern eines Nachrichtenverlaufs verwenden.....	37
Publish/Subscribe-Szenarios.....	39
Szenario: Publish/Subscribe-Cluster erstellen.....	39
Publish/Subscribe-Hierarchieszenarios.....	45
Transaktionsunterstützungsszenarios.....	55
Arbeitseinheiten - Einführung.....	56
Szenario 1: Der Warteschlangenmanager nimmt die Koordinierung vor.....	57
Szenario 2: Für die Koordination ist andere Software zuständig.....	84
Globale Arbeitseinheiten beenden.....	92
Unit of recovery disposition.....	93
Sicherheitsszenarios.....	94
Security scenario: two queue managers on z/OS.....	94
Security scenario: queue sharing group on z/OS.....	101
Server-to-server message channel interception example configurations.....	106
Zwei Warteschlangenmanager über SSL/TLS verbinden.....	107
Client sicher mit einem WS-Manager verbinden.....	114
Migration unter Windows.....	121
Lösung planen.....	121
Lösung über die grafische Benutzerschnittstelle implementieren.....	126
Höhere Version von IBM MQ installieren, die mit einer früheren Version unter Windows koexistieren soll.....	153
Übersicht über die Verwendung mehrerer Installationen.....	153
Installation einer neueren Version von IBM MQ parallel zu einer früheren Version.....	154
Befehl setmqenv zur Ausführung mit beiden Versionen von IBM MQ verwenden.....	156
Einen Warteschlangenmanager erstellen.....	158
Warteschlangenmanager zu einer höheren Version von IBM MQ migrieren.....	159
Fixpack unter IBM MQ 9.4 installieren.....	161
Managed File Transfer-Szenario.....	163
Gängige MFT-Topologien.....	163
Basisserver konfigurieren.....	167
Erste Schritte mit IBM MQ Internet Pass-Thru.....	175
Ordnungsgemäße Funktion von MQIPT überprüfen.....	177
Testzertifikate erstellen.....	178
Fordern Sie ein von einer Zertifizierungsstelle signiertes Zertifikat für MQIPT an.....	180
TLS-Server authentifizieren.....	182
TLS-Client authentifizieren.....	184
TLS-Client und -Server authentifizieren.....	187

HTTP-Tunnelung konfigurieren.....	191
Zugriffssteuerung konfigurieren.....	192
SOCKS-Proxy konfigurieren.....	195
SOCKS-Client konfigurieren.....	197
Clustering-Unterstützung für MQIPT konfigurieren.....	199
Portnummern zuordnen.....	202
CRLs mithilfe eines LDAP-Servers abrufen.....	203
MQIPT im TLS-Proxy-Modus ausführen.....	206
MQIPT im TLS-Proxy-Modus mit Sicherheitsmanager ausführen.....	208
Sicherheitsexit verwenden.....	211
Clientverbindungsanforderungen mithilfe von Sicherheitsexits an IBM MQ-Server für Warteschlangenmanager weiterleiten.....	213
Clientverbindungsanforderungen dynamisch weiterleiten.....	216
Zertifikatsexit für die Authentifizierung eines TLS-Servers verwenden.....	219
Kafka Connect-Szenarios.....	222
Kafka Connect-Allgemeine Topologien.....	222
Unterstützung genau einmal.....	231
Bemerkungen.....	233
Informationen zu Programmierschnittstellen.....	234
Marken.....	235

IBM MQ-Szenarios

Jedes Szenario führt Sie durch eine Reihe von Aufgaben und hilft Ihnen bei der Konfiguration einer wichtigen Produktfunktion. Die Szenarios enthalten hilfreiche Links zu anderen Inhalten, die Ihnen ein besseres Verständnis zu dem Bereich vermitteln, für den Sie sich interessieren.

Die verfügbaren IBM MQ -Szenarien werden in den folgenden Unterabschnitten beschrieben.

Windows Erste Schritte mit IBM MQ

Dieses Szenario veranschaulicht die ersten Schritte in IBM MQ auf einer Windows-Plattform. Verwenden Sie dieses Szenario, wenn IBM MQ für Sie neu ist und Sie möglichst schnell damit arbeiten möchten.

In diesem Szenario werden die grundlegenden Schritte für die Installation, Konfiguration und Überprüfung von IBM MQ unter Windows beschrieben, wenn das Produkt noch nicht auf dem System installiert ist. Sie können die Schritte des Szenarios entweder über die grafische Benutzerschnittstelle oder die Befehlszeilenschnittstelle durchführen.

Lösung planen

Wählen Sie eine Methode für die Installation von IBM MQ unter Windows aus. Verwenden Sie die grafische Benutzerschnittstelle und Assistenten, die Sie durch den Installations- und Konfigurationsprozess führen, oder führen Sie über die Befehlszeile eine unbeaufsichtigte Installation durch.

Übersicht: bereitgestellte logische Topologie

Hier finden Sie eine Beschreibung der bereitgestellten logischen Topologie nach Ausführung des Szenarios.

Die installierte IBM MQ-Serverinstanz ermöglicht die Erstellung von IBM MQ-Objekten: Warteschlangen und Warteschlangenmanager. Sie können IBM MQ Explorer verwenden, um Nachrichten aus der lokalen Warteschlange über den Warteschlangenmanager zu einzureihen und abzurufen. Nach Abschluss dieses Szenarios sieht die bereitgestellte Topologie wie in [Abbildung 1](#) dargestellt aus.

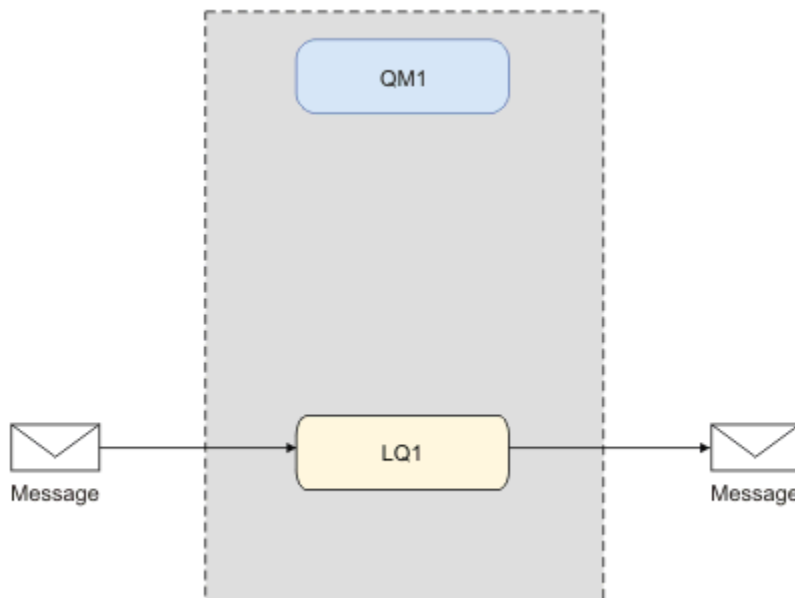


Abbildung 1. Nachricht in LQ1 einreihen, Nachricht aus LQ1 abrufen

Basiskonzepte und wichtige Begriffe

In diesem Abschnitt werden die Basiskonzepte und Schlüsselbegriffe beschrieben, die Sie kennen sollten, bevor Sie das Szenario 'Erste Schritte in IBM MQ' verwenden.

Basiskonzepte

IBM MQ ermöglicht es Anwendungen, Nachrichten aus einer Warteschlange zu lesen und in eine Warteschlange zu schreiben. Die Anwendung, die die Nachricht liest, ist unabhängig von der Anwendung, die die Nachricht schreibt. Es wird auch nicht vorausgesetzt, dass die beiden Anwendungen gleichzeitig aktiv sind. Wenn keine Anwendung zum Lesen der Nachricht verfügbar ist, wird die Nachricht in die IBM MQ-Warteschlange eingereiht, bis sie von einer Anwendung gelesen wird.

In diesem Szenario haben Sie folgende Möglichkeiten zur Installation und Konfiguration von IBM MQ:

„Installation und Konfiguration über die grafische Benutzerschnittstelle“ auf Seite 7

Bei Verwendung der grafischen Benutzerschnittstelle werden Sie bei der Installation durch verschiedene Anzeigen und Assistenten geleitet, um Sie bei Auswahl und Angabe der entsprechenden Optionen bzw. Einstellungen zu unterstützen:

Launchpad

Über das Launchpad können Sie Softwarevoraussetzungen überprüfen, Netzinformationen angeben und den Installationsassistenten für IBM MQ starten.

IBM MQ-Installationsassistent

Mit diesem Assistent wird die Software installiert und der Prepare IBM MQ Wizard gestartet.

Prepare IBM MQ Wizard

Mithilfe dieses Assistenten können Sie den IBM MQ-Service und IBM MQ Explorer starten.

IBM MQ Explorer

Hier können Sie Warteschlangen und Warteschlangenmanager verwalten.

„Installation und Konfiguration über die Befehlszeilenschnittstelle“ auf Seite 12

Die Installation über die Befehlszeilenschnittstelle kann unbeaufsichtigt oder interaktiv erfolgen. Die unbeaufsichtigte Installation ist uneingeschränkt für behindertengerechte Bedienung geeignet und die Installation, die in diesem Szenario verwendet wird. Während der Installation über die Befehlszeile werden Sie durch mehrere Schritte zur Festlegung der erforderlichen Optionen und Einstellungen geführt:

- IBM MQ installieren
- Erstellen und konfigurieren Sie IBM MQ-Objekte (Warteschlangenmanager und Warteschlangen).
- Überprüfen Sie die Installation, indem Sie mit `amqsput` eine Nachricht in die Warteschlange einreißen und mit `amqsget` eine Nachricht aus der Warteschlange abrufen.

Statt mit IBM MQ Explorer oder über die Befehlszeile können Sie IBM MQ-Objekte auch über die programmierbare Schnittstelle erstellen. Dies ist jedoch nicht Bestandteil des aktuellen Szenarios.

Schlüsselbegriffe

Es folgt eine Liste mit Schlüsselbegriffen zur Steuerung von Nachrichtenwarteschlangen (Message-Queuing).

Schlüsselbegriffe zum Message-Queuing.

Laufzeit	Beschreibung
<u>Warteschlangenmanager</u>	Der Warteschlangenmanager ist zuständig für die Verwaltung aller ihm zugeordneten Warteschlangen; er muss außerdem alle Nachrichten, die er empfängt, in den entsprechenden Warteschlangen speichern.
<u>Nachrichten</u>	Eine Nachricht ist eine Bytezeichenfolge, die Informationen für die Anwendungen bereitstellt, die diese Nachricht verwenden. Mithilfe von Nachrichten werden Informationen von einem Anwendungsprogramm an ein anderes übertragen. Die Anwendungen können auf demselben oder auf verschiedenen Computern aktiv sein.
<u>Lokale Warteschlangen</u>	Eine lokale Warteschlange ist eine Datenstruktur, in der Nachrichten gespeichert werden. Dabei kann es sich um eine normale Warteschlange oder eine Übertragungswarteschlange handeln. Eine normale Warteschlange enthält Nachrichten, die von einer Anwendung gelesen werden sollen, die die Nachricht direkt vom Warteschlangenmanager liest. Eine Übertragungswarteschlange enthält Nachrichten, die an einen anderen Warteschlangenmanager übertragen werden sollen.

Lösung implementieren

Implementieren Sie die Lösung im Szenario. Installieren Sie IBM MQ unter Windows mit dem Installationslaunchpad und überprüfen Sie dann die Installation mit IBM MQ Explorer.

Installation und Konfiguration über die grafische Benutzerschnittstelle

Installieren Sie IBM MQ unter Windows mit dem Installationslaunchpad und überprüfen Sie die Installation mit IBM MQ Explorer. Erstellen Sie nach erfolgreicher Überprüfung der Installation einen Warteschlangenmanager und eine Warteschlange und versuchen Sie dann, eine Nachricht in die Warteschlange einzureihen und eine Nachricht aus der Warteschlange abzurufen.

Installation über das Launchpad

Installieren Sie IBM MQ unter Windows mithilfe des Installationslaunchpads.

Vorbereitende Schritte

Führen Sie vor Beginn dieser Aufgabe folgende Überprüfungen durch:

- Beim Installieren müssen Sie über eine lokale Administratorberechtigung verfügen. Definieren Sie diese Berechtigung über die Windows-Funktionen.
- Achten Sie darauf, dass der Maschinename keine Leerzeichen enthält.
- Stellen Sie sicher, dass genügend Plattenspeicherplatz für die vollständige Installation von IBM MQ for Windows vorhanden ist. Weitere Informationen finden Sie unter [Plattenspeicherbedarf auf Multiplatforms](#).
- Bestimmen Sie, ob Sie Windows -Domänenbenutzer-IDs für IBM MQ -Benutzer definieren müssen.

Überprüfen Sie vor der Installation von IBM MQ, ob Ihr System die Hardware- und Softwarevoraussetzungen erfüllt. Weitere Informationen zu Hardware- und Softwarevoraussetzungen auf allen unterstützten Plattformen finden Sie unter [Systemvoraussetzungen für IBM MQ](#).

Informationen zu diesem Vorgang

Das Launchpad und nachfolgende Assistenten führen Sie durch den Installationsprozess und unterstützen Sie bei der Überprüfung der Softwarevoraussetzungen und IBM MQ-Einstellungen.

Bei dieser Task wird vorausgesetzt, dass Sie IBM MQ zum ersten Mal auf Ihrem System installieren. Außerdem wird davon ausgegangen, dass Sie die Standardpositionen verwenden. Die Position der IBM MQ -Programmdateien ist standardmäßig C:\Programme\IBM\MQ, und die Daten- und Protokolldateiposition ist C:\ProgramData \IBM \MQ.

Anmerkung: Falls Sie IBM MQ 9.0 installieren und auf Ihrem System bereits frühere Installationen von IBM MQ vorhanden sind, weicht die Position der Programm- und Datendateien von der Standardposition ab. Weitere Informationen finden Sie im Abschnitt Positionen von Programm- und Datenverzeichnis. Wenn Sie dieses Szenario schon einmal ausgeführt haben und es mit einer einzelnen, neuen Installation an den Standardpositionen wiederholen möchten, entfernen Sie die vorherige Installation, bevor Sie das Szenario erneut starten. Informationen zur Deinstallation einer vorhandenen Instanz von IBM MQ von einem System finden Sie im Abschnitt „IBM MQ Deinstallieren“ auf Seite 18.

Die Installationsprogramme enthalten Links zu weiteren Informationen, die während des Installationsprozesses hilfreich sein können.

Vorgehensweise

1. Starten Sie das Launchpad und prüfen Sie die Softwarevoraussetzungen und die Netzkonfiguration, damit Sie eventuell erforderliche Änderungen vornehmen können.

- a) Navigieren Sie zum IBM MQ -Software-Verzeichnis, und klicken Sie doppelt auf die Datei Set-up .exe , um das Launchpad zu starten.
- b) Wählen Sie die Registerkarte **Softwarevoraussetzungen** aus, um die Einstellungen **Softwarevoraussetzungen** anzuzeigen.
- c) Prüfen Sie, ob die Softwarevoraussetzungen erfüllt werden. Der jeweilige Eintrag für die Voraussetzung muss mit einem grünen Häkchen und dem Wort OK angezeigt werden. Nehmen Sie alle angegebenen Korrekturen vor.

Anmerkung:

Falls Sie Details zu einer Voraussetzung erfahren möchten, klicken Sie auf das Kontrollkästchen, um eine Registerkarte mit Informationen einzublenden.

- d) Wählen Sie die Registerkarte **Netzkonfiguration** aus, um die Einstellungen **Netzkonfiguration** anzuzeigen.
- e) Wählen Sie **Nein** aus.

Anmerkung: Bei diesem Szenario wird vorausgesetzt, dass Sie keine Domänenbenutzer-ID für IBM MQ konfigurieren müssen. Wenn Sie weitere Informationen zur Konfiguration von IBM MQ für Windows-Domänenbenutzern benötigen, klicken Sie auf **Weitere Informationen**.

- f) Wählen Sie im Launchpad auf der Registerkarte **IBM MQ-Installation** die Installationssprache aus und klicken Sie dann auf **Installationsprogramm von IBM MQ starten**, um den Installationsassistenten von IBM MQ zu starten.

Sie haben die Einrichtung von IBM MQ abgeschlossen, indem Sie Ihre Installationsvoraussetzungen erfüllt oder angegeben haben, und Sie haben den Installationsassistenten von IBM MQ gestartet.

2. Verwenden Sie den IBM MQ-Installationsassistenten zur Installation der Software und starten Sie den Prepare IBM MQ Wizarden.

- a) Lesen Sie im IBM MQ-Installationsassistenten die Lizenzvereinbarung und klicken Sie auf das Kontrollkästchen **Lizenzvereinbarung akzeptieren** und anschließend auf **Weiter**.
- b) Klicken Sie auf **Standard** und anschließend auf **Weiter**.
- c) Prüfen Sie auf der Seite **Bereit zum Installieren von IBM MQ** die Installationsinformationen und klicken Sie auf **Installieren**.

Anmerkung: Beachten Sie die folgenden Details:

- Installationsname
- Ordner der höchsten Ebene für Programmdateien
- Ordner der höchsten Ebene für Datendateien

Die folgenden Komponenten werden installiert:

- IBM MQ-Server

- IBM MQ: eine grafische Schnittstelle für die Verwaltung und Überwachung der IBM MQ-Ressourcen
- Java und .NET Messaging- und Web-Services
- IBM MQ-Entwicklungstoolkit

Der Installationsprozess wird gestartet. Je nach Ihrem System kann der Installationsprozess mehrere Minuten dauern.

Am Ende des Installationsprozesses wird im Fenster IBM MQ Setup die Nachricht **Installation Wizard Completed Successfully** angezeigt.

d) Klicken Sie auf **Fertigstellen**.

Sie haben IBM MQ erfolgreich installiert. Der Prepare IBM MQ Wizard startet automatisch und zeigt dann die Seite des **Prepare IBM MQ Wizarden** an.

3. Verwenden Sie den Prepare IBM MQ Wizarden zum Starten des IBM MQ-Service.

a) Wählen Sie auf der Seite 'Willkommen beim Prepare IBM MQ Wizarden' die Option **Weiter** aus.

Die Prepare IBM MQ Wizard zeigt die Nachricht **Status: Checking IBM MQ Configuration** und einen Fortschrittsanzeiger an. Sobald der Prozess abgeschlossen ist, wird die Seite 'IBM MQ-Netzkonfiguration' angezeigt.

b) Wählen Sie auf der Seite 'IBM MQ -Netzkonfiguration' der Prepare IBM MQ Wizard **Neinaus**.

c) Klicken Sie auf **Weiter**.

Prepare IBM MQ Wizard zeigt eine Nachricht **Status: starting the IBM MQ Service** und einen Fortschrittsanzeiger an. Wenn der Prozess abgeschlossen ist, zeigt der Assistent die Seite 'Prepare IBM MQ Wizard wird beendet' an.

d) Wählen Sie **IBM MQ Explorer starten** aus und geben Sie an, ob die Releaseinformationen angezeigt werden sollen. Klicken Sie dann auf die Schaltfläche **Fertigstellen**.

IBM MQ Explorer wird gestartet.

Sie haben IBM MQ installiert. Außerdem haben Sie IBM MQ Explorer gestartet.

Ergebnisse

IBM MQ wurde installiert und überprüft und Sie können jetzt Objekte wie Warteschlangenmanager und Warteschlangen konfigurieren.

Nächste Schritte

Befolgen Sie die Anweisungen in [„Erstellen eines Warteschlangenmanagers mit dem Namen QM1“](#) auf Seite 9.

Zugehörige Konzepte

[Erforderlicher Plattenspeicherplatz](#)

[Hardware- und Softwarevoraussetzungen auf Windows-Systemen](#)

[Einführung in IBM MQ](#)

Zugehörige Tasks

[IBM MQ-Server unter Windows installieren](#)

[IBM MQ-Server konfigurieren](#)

Erstellen eines Warteschlangenmanagers mit dem Namen QM1

Erstellen Sie einen Warteschlangenmanager mit dem Namen QM1, indem Sie IBM MQ Explorer verwenden. Warteschlangenmanager sind die Hauptkomponenten in einem IBM MQ-Nachrichtenübertragungsnetz.

Vorbereitende Schritte

IBM MQ muss bereits installiert sein. Andernfalls finden Sie Informationen hierzu im Abschnitt [„Installation über das Launchpad“](#) auf Seite 7.

Informationen zu diesem Vorgang

In diesem Beispiel sind alle Namen in Großbuchstaben geschrieben. Da IBM MQ die Groß-/Kleinschreibung beachtet, müssen Sie die Namen ebenfalls groß eingeben.

Führen Sie die folgenden Schritte aus, um einen Warteschlangenmanager mithilfe IBM MQ Explorer von zu erstellen und zu starten.

Vorgehensweise

1. Starten Sie IBM MQ Explorer als Administrator.
2. Klicken Sie in der **Navigatoransicht** mit der rechten Maustaste auf den Ordner **Queue Managers** und klicken Sie anschließend auf **Neu > Warteschlangenmanager**. Der Assistent **Warteschlangenmanager erstellen** wird gestartet.
3. Geben Sie im Feld **Warteschlangenmanager-Name** den Namen QM1 ein.
4. Wählen Sie das Kontrollkästchen `Make this the default queue manager` aus.
5. Geben Sie im Feld **Warteschlange für nicht zustellbare Nachrichten** den Namen `SYS-TEM.DEAD.LETTER.QUEUE` ein.
Dies ist der Name der Warteschlange für nicht zustellbare Nachrichten, die beim Erstellen des Warteschlangenmanagers automatisch erstellt wird.
6. Lassen Sie die übrigen Felder leer und klicken Sie auf **Fertigstellen** oder, falls diese Schaltfläche inaktiviert ist, auf **Weiter**.
Die Schaltfläche **Fertigstellen** ist inaktiviert, wenn es bei der Portnummer einen Konflikt mit einem vorhandenen Warteschlangenmanager gibt, z. B. mit dem Warteschlangenmanager, der im Rahmen der Standardkonfiguration erstellt wird. Sie müssen mit dem Assistenten fortfahren, um die Standardportnummer zu ändern.
7. Wenn Sie auf **Weiter** geklickt haben, akzeptieren Sie die Standardwerte und klicken Sie auf jeder Seite auf **Weiter**, bis Sie zur letzten Seite des Assistenten gelangen, auf der die Schaltfläche **Fertigstellen** dann verfügbar ist. Ändern Sie die angegebene Portnummer, z. B. in 1415, und klicken Sie auf **Fertigstellen**.

IBM MQ zeigt das Dialogfenster **Warteschlangenmanager wird erstellt** an, während der Warteschlangenmanager erstellt und gestartet wird.

Nächste Schritte

Informationen zum Erstellen einer Warteschlange finden Sie im Abschnitt [„Warteschlange LQ1 erstellen“](#) auf Seite 10.

Zugehörige Tasks

[Warteschlangenmanager auf Multiplatforms erstellen und verwalten](#)

Warteschlange LQ1 erstellen

Erstellen Sie eine Warteschlange mithilfe von IBM MQ Explorer. Warteschlangen sind Strukturen, die zum Speichern von Nachrichten verwendet werden und bei denen es sich um Warteschlangenmanager-Objekte von IBM MQ handelt.

Informationen zu diesem Vorgang

In dieser Task können Sie IBM MQ-Objekte mit IBM MQ Explorer erstellen.

Führen Sie die folgenden Schritte aus, um eine Warteschlange mit IBM MQ Explorer zu erstellen und zu starten.

Vorgehensweise

1. Erweitern Sie in der Ansicht **Navigator** den Ordner **Warteschlangenmanager**.
2. Erweitern Sie den Warteschlangenmanager **QM1**.

3. Klicken Sie auf den Ordner **Warteschlangen** , und klicken Sie dann auf **Neu > Lokale Warteschlange ...** . Der Assistent **Neue lokale Warteschlange** wird gestartet.
4. Geben Sie im Feld **Name** den Wert LQ1ein.
5. Klicken Sie auf **Fertigstellen**.

Die neue Warteschlange LQ1 wird in der Ansicht **Inhalt** angezeigt. Falls die Warteschlange nicht in der Ansicht **Inhalt** angezeigt wird, klicken Sie auf die Schaltfläche **Aktualisieren** am oberen Rand der Ansicht **Inhalt**.

Nächste Schritte

Sie können jetzt eine Nachricht in die Warteschlange einreihen. Informationen zum Einreihen einer Nachricht in eine Warteschlange finden Sie im Abschnitt [„Nachricht in die Warteschlange LQ1 einreihen“](#) auf Seite 11.

Nachricht in die Warteschlange LQ1 einreihen

Reihen Sie eine Nachricht in die Warteschlange LQ1 ein, indem Sie IBM MQ Explorer verwenden.

Informationen zu diesem Vorgang

Diese Aufgabe setzt voraus, dass Sie bereits einen Warteschlangenmanager namens QM1 (wie in [„Erstellen eines Warteschlangenmanagers mit dem Namen QM1“](#) auf Seite 15 beschrieben) und eine Warteschlange namens LQ1 (wie in [„Warteschlange LQ1 erstellen“](#) auf Seite 10 beschrieben) erstellt haben.

Führen Sie die folgenden Schritte aus, um eine Nachricht mithilfe von IBM MQ Explorer in die Warteschlange einzureihen.

Vorgehensweise

1. Erweitern Sie in der Ansicht **Navigator** den Ordner **Warteschlangenmanager**.
2. Erweitern Sie den von Ihnen erstellten Warteschlangenmanager QM1.
3. Klicken Sie auf den Ordner **Warteschlangen**. Die Warteschlangen des Warteschlangenmanagers werden in der Inhaltsansicht aufgelistet.
4. Klicken Sie in der Inhaltsansicht mit der rechten Maustaste auf die lokale Warteschlange LQ1 und klicken Sie dann auf **Testnachricht einreihen**.
Der Dialog **Testnachricht einreihen** wird geöffnet.
5. Geben Sie im Feld **Nachrichtendaten** Text ein, z. B. Hello World, und klicken Sie dann auf **Nachricht einreihen**.
Der Inhalt des Felds **Nachrichtendaten** wird gelöscht und die Nachricht in die Warteschlange eingereiht.
6. Klicken Sie auf **Schließen** .
Beachten Sie in der Inhaltsansicht, dass der LQ1 **Aktuelle Warteschlangenlänge** -Wert jetzt 1 ist. Wenn die Spalte **Aktuelle Warteschlangenlänge** nicht sichtbar ist, müssen Sie möglicherweise nach rechts von **Inhaltsansicht**blättern.

Nächste Schritte

Informationen zum Abrufen einer Nachricht aus einer Warteschlange finden Sie im Abschnitt [„Nachricht aus der Warteschlange LQ1 abrufen“](#) auf Seite 11.

Nachricht aus der Warteschlange LQ1 abrufen

Sie können eine Nachricht aus der Warteschlange LQ1 mithilfe von IBM MQ Explorer abrufen.

Informationen zu diesem Vorgang

Diese Aufgabe setzt voraus, dass Sie bereits eine Nachricht in LQ1 eingereiht haben (wie in [„Nachricht in die Warteschlange LQ1 einreihen“](#) auf Seite 11 beschrieben).

Führen Sie die folgenden Schritte aus, um eine Nachricht aus der Warteschlange abzurufen, indem Sie IBM MQ Explorer verwenden.

Vorgehensweise

1. Erweitern Sie in der Ansicht **Navigator** den Ordner **Warteschlangenmanager** und dann den Warteschlangenmanager QM1.
2. Klicken Sie auf den Ordner **Warteschlangen**.
3. Klicken Sie in der Ansicht **Inhalt** mit der rechten Maustaste auf die lokale Warteschlange LQ1 und klicken Sie dann auf **Nachrichten anzeigen**. Im **Nachrichten-Browser** wird eine Liste mit den Nachrichten angezeigt, die aktuell von QM1 verwaltet werden.
4. Doppelklicken Sie auf die letzte Nachricht, um den Eigenschaftendialog zu öffnen.

Im Feld **Nachrichtendaten** auf der Seite **Daten** des Eigenschaftendialogs wird der Inhalt der Nachricht in lesbarer Form angezeigt.

Nächste Schritte

Folgen Sie den Anweisungen in nachfolgenden Szenarios, um weitere IBM MQ-Funktionen zu erkunden.

Informationen zum Schreiben von Anwendungen zur Warteschlangensteuerung, zum Herstellen und Trennen einer Verbindung mit einem Warteschlangenmanager, zum Publish/Subscribe sowie zum Öffnen und Schließen von Objekten finden Sie im Abschnitt [Prozedurale Anwendung für die Warteschlangensteuerung schreiben](#).

Installation und Konfiguration über die Befehlszeilenschnittstelle

Installieren Sie IBM MQ unter Windows über die Befehlszeile, um eine unbeaufsichtigte Installation durchzuführen, und setzen Sie die Umgebungsvariable. Erstellen Sie nach erfolgreicher Überprüfung der Installation einen Warteschlangenmanager und eine Warteschlange und versuchen Sie dann, eine Nachricht in die Warteschlange einzureihen und eine Nachricht aus der Warteschlange abzurufen.

Installation mithilfe der unbeaufsichtigten Installation

Installieren Sie IBM MQ unter Windows über die Befehlszeile, um eine unbeaufsichtigte Installation durchzuführen, und überprüfen Sie, ob die Umgebung für die Installation ordnungsgemäß eingerichtet ist.

Vorbereitende Schritte

Prüfen Sie vor Beginn dieser Task zunächst folgende Punkte:

- Beim Installieren müssen Sie über eine lokale Administratorberechtigung verfügen. Definieren Sie diese Berechtigung über die Windows-Funktionen.
- Achten Sie darauf, dass der Maschinenname keine Leerzeichen enthält.
- Stellen Sie sicher, dass ausreichender Plattenspeicherplatz vorhanden ist. Weitere Informationen finden Sie unter [Plattenspeicherbedarf auf Multiplattformen](#).
- Bestimmen Sie, ob Sie Windows -Domänenbenutzer-IDs für IBM MQ -Benutzer definieren müssen.

Überprüfen Sie vor der Installation von IBM MQ, ob Ihr System die Hardware- und Softwarevoraussetzungen erfüllt. Aktuelle Informationen über Hardware- und Softwarevoraussetzungen auf allen unterstützten Plattformen finden Sie unter [Systemvoraussetzungen für IBM MQ](#).

Informationen zu diesem Vorgang

Bei diesem Szenario wird vorausgesetzt, dass Sie IBM MQ zum ersten Mal auf Ihrem System installieren. Außerdem wird davon ausgegangen, dass Sie die Standardpositionen verwenden. Die Position der IBM MQ 9.0 -Programmdateien ist standardmäßig C:\Programme\IBM\MQ, und die Daten- und Protokolldateiposition ist C:\ProgramData \IBM \MQ.

Anmerkung: Falls auf Ihrem System bereits frühere Installationen von IBM MQ vorhanden sind, weichen die Standardpositionen der Programm- und Datendateien möglicherweise ab. Weitere Informationen finden Sie im Abschnitt [Positionen von Programm- und Datenverzeichnis](#). Wenn Sie dieses Szenario schon einmal ausgeführt haben und es mit einer einzelnen, neuen Installation an den Standardpositionen wiederholen möchten, entfernen Sie die vorherige Installation, bevor Sie das Szenario erneut starten. Informationen zur Deinstallation einer vorhandenen Instanz von IBM MQ von einem System finden Sie im Abschnitt [„IBM MQ Deinstallieren“](#) auf Seite 18.

IBM MQ unter Windows verwendet die MSI-Technologie zur Installation der Software. Weitere Informationen zur Installation mithilfe der MSI-Technologie finden Sie im Abschnitt [Erweiterte Installation mit msiexec](#).

Für die Installation von IBM MQ über die Befehlszeile müssen Sie folgende Parameter angeben:

- `/i "MQ_INSTALLATION_MEDIA\MSI\IBM MQ.msi"`, wobei `MQ_INSTALLATION_MEDIA` die Position der Datei `IBM MQ.msi` ist. Dieses Argument gibt die Position der Datei mit der Erweiterung `.msi` an.
- `/l*v USER_LOGFILE_LOCATION\install.log` Dabei ist `USER_LOGFILE_LOCATION` das Verzeichnis, in das die Installationsprotokolle geschrieben werden sollen.

Anmerkung: Der Ordner, in dem die `install.log` erstellt werden soll, muss vorhanden sein, bevor Sie den Befehl ausführen.

- `/q[n|b|r|f]` /q muss mit `n`, `b`, `r` oder `f` paarweise verbunden sein. Wenn Sie den Befehl `msiexec` an einer Eingabeaufforderung ausführen, wird die Hilfedatei geöffnet, die die korrekte Syntax anzeigt.
- `USEINI="RESPONSE_FILE"`, wobei `ANTWORTDATEI` der Name und die Position der Antwortdatei ist, die von der unbeaufsichtigten Installation verwendet werden soll. In diesem Szenario wird die Beispieldatei `Response.ini` verwendet, die sich auf den Installationsmedien von IBM MQ befindet.
- `TRANSFORMS="TRANSFORM_FILE"`, wobei `TRANSFORM_FILE` der Name der Transformationsdatei ist, die auf die Installation angewendet wird. In diesem Szenario wird die Transformationsdatei `1033.mst` für amerikanisches Englisch verwendet.
- `AGREETOLICENSE="YES"`. Dieser Parameter muss eingeschlossen werden, da die Installation andernfalls nicht ausgeführt werden kann.
- `ADDLOCAL="Server"`. Dieser Parameter listet die zu installierenden Komponenten auf.

Vorgehensweise

1. Verwenden Sie die Befehlszeile für die Durchführung einer unbeaufsichtigten Installation.

- a) Rufen Sie die unbeaufsichtigte Installation aus einer erweiterten Eingabeaufforderung auf, indem Sie auf die **Starttaste** in der **Windows-Taskleiste** klicken und `cmd` im Feld **Programme/Dateien durchsuchen** eingeben. Klicken Sie mit der rechten Maustaste auf das Programm `cmd.exe` und wählen Sie **Als Administrator ausführen** aus.
- b) Geben Sie folgenden Befehl in der Windows-Eingabeaufforderung ein:

Anmerkung: Im vorliegenden Beispiel wird der Befehl in mehreren Zeilen dargestellt, er muss jedoch in einer Zeile eingegeben werden.

```
msiexec /i "MQ_INSTALLATION_MEDIA\MSI\IBM MQ.msi"  
/l*v c:\wmqinslogs\install.log  
/q USEINI="MQ_INSTALLATION_MEDIA\Response.ini"  
TRANSFORMS="1033.mst"  
AGREETOLICENSE="yes"  
ADDLOCAL="Server"
```

Dabei muss für `MQ-INSTALLATIONSMEDIEN` der Pfad Ihrer IBM MQ-Installationsmedien angegeben werden.

Anmerkung: Der Ordner, in dem die `install.log` erstellt werden soll, muss vorhanden sein, bevor Sie den Befehl ausführen.

Nachdem Sie den Befehl eingegeben haben, wird in der Befehlszeile wieder die Eingabeaufforderung angezeigt.

- c) Um den Installationsfortschritt anzuzeigen, öffnen Sie die Protokolldatei, die Sie angegeben haben. Wenn die Installation erfolgreich abgeschlossen wurde, wird die Nachricht `Product: IBM MQ (Installation1) -- Installation operation completed successfully`. zwei Absätze nach oben vom Ende der Protokolldatei angezeigt.
- d) Sobald die Installation abgeschlossen ist, wird der Service gestartet und in der Taskleiste das Symbol von IBM MQ angezeigt.

Sie haben IBM MQ installiert und den IBM MQ-Service gestartet.

2. Richten Sie mit dem Befehl **setmqenv** Umgebungsvariablen für Ihre Installation ein.

- a) Geben Sie den folgenden Befehl in der Befehlszeile ein:

```
"MQ_INSTALLATION_PATH/bin/setmqenv" -s
```

Dabei ist `MQ_INSTALLATION_PATH` das Verzeichnis, in dem IBM MQ installiert ist. Sie müssen den Pfad zu **setmqenv** im Ordner `bin` in Anführungszeichen setzen, um zu verhindern, dass die Eingabeaufforderung einen Fehler zurückgibt.

Anmerkung: Wenn Sie die Standardposition verwendet haben, lautet der Pfad zu Ihrer Installation `C:\Programme\IBM\MQ`.

- b) Überprüfen Sie mit dem folgenden Befehl, dass die Umgebung korrekt eingerichtet ist:

```
dspmqr
```

Wenn der Befehl erfolgreich ausgeführt wird und Versionsnummer und Installationsname wie erwartet zurückgegeben werden, ist die Umgebung ordnungsgemäß eingerichtet. Die Nachricht sollte die folgende Zeile enthalten:

```
Version: n.n.n.n
```

Dabei steht `n.n.n.n` für die Versionsnummer und, wenn Sie keinen nicht standardmäßigen Installationsnamen angegeben haben, die Zeile:

```
InstName: Installation1
```

Sie haben IBM MQ erfolgreich mithilfe einer unbeaufsichtigten Installation installiert.

Ergebnisse

Sie haben eine unbeaufsichtigte Installation von IBM MQ durchgeführt und überprüft, ob die Umgebung ordnungsgemäß eingerichtet ist.

Nächste Schritte

- Sie können den [Prepare IBM MQ Wizard](#) ausführen.
- Befolgen Sie die Anweisungen in [„Erstellen eines Warteschlangenmanagers mit dem Namen QM1“](#) auf [Seite 15](#).

Falls während der Installation Probleme auftreten, prüfen Sie das Installationsprotokoll an der Position, die Sie im Befehl **msiexec** angegeben haben. In diesem Szenario lautet die Position der Protokolldatei wie folgt: `c:\wmqinslogs\install.log`. Ergreifen Sie die im Protokoll angegebenen Maßnahmen und führen Sie die Installation erneut aus. Sie können auch die mit dem Befehl übergebenen Parameter überprüfen, um sicherzustellen, dass Sie alle erforderlichen Parameter eingeschlossen haben.

Zugehörige Tasks

[Server mithilfe von 'msiexec' installieren](#)

[Umsetzungsdateien mit 'msiexec' verwenden](#)

[IBM MQ installieren - Übersicht](#)

Erstellen eines Warteschlangenmanagers mit dem Namen QM1

Erstellen Sie über die Befehlszeilenschnittstelle einen Warteschlangenmanager mit dem Namen QM1.

Warteschlangenmanager sind die Hauptkomponenten in einem IBM MQ-Nachrichtenübertragungsnetz.

Vorbereitende Schritte

IBM MQ muss bereits installiert sein. Andernfalls finden Sie Informationen hierzu im Abschnitt [„Installation mithilfe der unbeaufsichtigten Installation“](#) auf Seite 12.

Informationen zu diesem Vorgang

In diesem Beispiel sind alle Namen in Großbuchstaben geschrieben. Da IBM MQ die Groß-/Kleinschreibung beachtet, müssen Sie die Namen ebenfalls groß eingeben.

Vorgehensweise

1. Öffnen Sie als Administrator eine Eingabeaufforderung.
2. Erstellen Sie einen Warteschlangenmanager mit dem Namen QM1, indem Sie folgenden Befehl eingeben:

```
crtmqm QM1
```

Während das System den Warteschlangenmanager erstellt, wird folgende Ausgabe angezeigt:

```
C:\>crtmqm QM1
IBM MQ queue manager created.
Creating or replacing default objects for QM1.
Default objects statistics : 61 created. 0 replaced. 0 failed.
Completing setup.
Setup completed.
```

Der Warteschlangenmanager wird erstellt und ist gestoppt. Sie müssen den Warteschlangenmanager starten, um ihn verwalten und Nachrichten aus seinen Warteschlangen lesen und in seine Warteschlangen schreiben zu können.

3. Starten Sie den Warteschlangenmanager, indem Sie den folgenden Befehl eingeben:

```
strmqm QM1
```

Bei einem erfolgreichen Start des Warteschlangenmanagers wird folgende Ausgabe angezeigt:

```
C:\>strmqm QM1
IBM MQ queue manager 'QM1' starting.
5 log records accessed on queue manager 'QM1' during the log replay phase.
Log replay for queue manager 'QM1' complete.
Transaction manager state recovered for queue manager 'QM1'.
IBM MQ queue manager 'QM1' started.
```

Der Warteschlangenmanager ist gestartet.

Nächste Schritte

Informationen zum Erstellen einer Warteschlange finden Sie im Abschnitt [„Warteschlange LQ1 erstellen“](#) auf Seite 16.

Zugehörige Tasks

[Warteschlangenmanager auf Multiplatforms erstellen und verwalten](#)

Warteschlange LQ1 erstellen

Erstellen Sie über die Befehlszeilenschnittstelle eine Warteschlange. Warteschlangen sind Strukturen, die zum Speichern von Nachrichten verwendet werden und bei denen es sich um Warteschlangenmanager-Objekte von IBM MQ handelt.

Informationen zu diesem Vorgang

Es gibt drei Möglichkeiten, IBM MQ-Objekte zu erstellen:

- über die Befehlszeile
- IBM MQ Explorer.
- über eine programmierbare Schnittstelle

In dieser Aufgabe können Sie IBM MQ-Objekte über die Befehlszeile erstellen.

Für die Befehlszeilenschnittstelle gibt es die Scriptsprache IBM MQ Script Commands (MQSC). Zur Ausführung des Scripts für einen Warteschlangenmanager wird das Scripting-Tool **runmqsc** verwendet. Gehen Sie wie folgt vor, um eine Warteschlange über die Befehlszeilenschnittstelle zu erstellen und zu starten.

Vorgehensweise

1. Starten Sie das Scripting-Tool mit folgendem Befehl:

```
runmqsc QM1
```

Beim Start des Scripting-Tools wird folgende Ausgabe angezeigt:

```
C:\>runmqsc QM1
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM1.
```

Das Tool ist nun für WebSphere MQ-Scriptbefehle bereit.

2. Erstellen Sie eine lokale Warteschlange namens LQ1, indem Sie den folgenden MQSC-Befehl eingeben:

```
define qlocal(LQ1)
```

Während die Warteschlange erstellt wird, wird folgende Ausgabe angezeigt:

```
define qlocal(LQ1)
2 : define qlocal(LQ1)
AMQ8006: IBM MQ queue created.
```

3. Mit folgendem WebSphere MQ-Scriptbefehl stoppen Sie das Scripting-Tool wieder:

```
end
```

Bei Beendigung des Scripting-Tools wird folgende Ausgabe angezeigt:

```
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.

C:\>
```


Nächste Schritte

Sie können jetzt eine Nachricht in die Warteschlange einreihen. Informationen zum Einreihen einer Nachricht in eine Warteschlange finden Sie im Abschnitt [„Nachricht in die Warteschlange LQ1 einreihen“](#) auf Seite 17.

Nachricht in die Warteschlange LQ1 einreihen

Reihen Sie über die Befehlszeilenschnittstelle eine Nachricht in die Warteschlange LQ1 ein.

Informationen zu diesem Vorgang

IBM MQ stellt standardmäßig die Beispieldanwendung **amqsput** bereit. Diese Anwendung reiht eine Nachricht in eine vordefinierte Warteschlange ein.

Gehen Sie wie folgt vor, um über die Befehlszeilenschnittstelle eine Nachricht in die Warteschlange einzureihen:

Vorgehensweise

1. Reihen Sie mithilfe der Beispieldanwendung **amqsput** eine Nachricht in die Warteschlange LQ1 ein, indem Sie folgenden Befehl eingeben:

```
amqsput LQ1 QM1
```

Beim Start der Beispieldanwendung wird folgende Ausgabe angezeigt:

```
C:\>amqsput LQ1 QM1
Sample AMQSPUT0 start
target queue is LQ1
```

2. Geben Sie `Hello World` ein und drücken Sie die Eingabetaste. Sie haben eine Nachricht mit dem Text "Hello World" in die Warteschlange LQ1, die vom Warteschlangenmanager QM1 verwaltet wird, eingereicht.
3. Drücken Sie zum Beenden von **amqsput** die **Eingabetaste**. Die folgende Ausgabe wird angezeigt:

```
C:\>amqsput LQ1 QM1
Sample AMQSPUT0 start
target queue is LQ1
Hello World

Sample AMQSPUT0 end
```

Nächste Schritte

Informationen zum Abrufen einer Nachricht aus einer Warteschlange finden Sie im Abschnitt [„Nachricht aus der Warteschlange LQ1 abrufen“](#) auf Seite 17.

Nachricht aus der Warteschlange LQ1 abrufen

Rufen Sie über die Befehlszeilenschnittstelle eine Nachricht aus der Warteschlange LQ1 ab.

Informationen zu diesem Vorgang

IBM MQ stellt standardmäßig die Beispieldanwendung **amqsget** bereit. Diese Anwendung liest Nachrichten aus einer Warteschlange.

Gehen Sie wie folgt vor, um über die Befehlszeilenschnittstelle eine Nachricht aus der Warteschlange abzurufen:

Vorgehensweise

Lesen Sie mithilfe der Beispielanwendung **amqsget** eine Nachricht aus der Warteschlange LQ1, indem Sie folgenden Befehl eingeben:

```
amqsget LQ1 QM1
```

Beim Start der Beispielanwendung wird folgende Ausgabe angezeigt:

```
C:\>amqsget LQ1 QM1
Sample AMQSGETO start
message <Hello World>
no more messages
Sample AMQSGETO end
```

Die Anwendung **amqsget** wird 30 Sekunden nach dem Lesen der Nachricht beendet.

Nächste Schritte

Folgen Sie den Anweisungen in nachfolgenden Szenarios, um weitere IBM MQ-Funktionen zu erkunden.

Informationen zum Schreiben von Anwendungen zur Warteschlangensteuerung, zum Herstellen und Trennen einer Verbindung mit einem Warteschlangenmanager, zum Publish/Subscribe sowie zum Öffnen und Schließen von Objekten finden Sie im Abschnitt [Prozedurale Anwendung für die Warteschlangensteuerung schreiben](#).

IBM MQ Deinstallieren

Stoppen und deinstallieren Sie dann IBM MQ und entfernen Sie dabei alle Warteschlangenmanager und deren Objekte. Nach Abschluss dieser Aufgabe können Sie IBM MQ erneut installieren.

Informationen zu diesem Vorgang

Diese Task beschreibt die Schritte zum Deinstallieren von IBM MQ unter Windows mithilfe des heruntergeladenen Installationsimage.

Das Szenario 'Erste Schritte' führt Sie durch die Optionen für die Installation von IBM MQ über das Launchpad oder die Befehlszeile. Obwohl mehrere Installationen von IBM MQ vorhanden sein können, basiert dieses Szenario auf einer neuen Installation auf einem einzelnen Server. Falls Sie das Szenario wiederholen oder eine andere Installationsmethode ausprobieren möchten, müssen Sie deshalb zuerst die vorhandenen IBM MQ-Komponenten deinstallieren (einschließlich aller vorhandenen Warteschlangenmanager und ihrer Objekte), damit Sie wieder mit einer neuen Installation starten können.

Möglicherweise müssen Sie die Deinstallation so ausführen, dass Sie eine neue Installation für einige andere Szenarios in diesem Abschnitt durchführen können.

Vorgehensweise

1. Stoppen Sie den Service IBM MQ.

- a) Klicken Sie in der Taskleiste mit der rechten Maustaste auf das **IBM MQ**-Symbol und wählen Sie **IBM MQ stoppen** aus, um den IBM MQ-Service zu stoppen.

Es wird ein Dialogfenster mit folgender Nachricht angezeigt:

```
Shutting down IBM MQ installation "Installation1" terminates all running queue managers and
IBM MQ processes for that installation, except those under Microsoft Failover Cluster con-
trol.
```

Soll der Vorgang fortgesetzt werden?

- b) Klicken Sie auf **Ja** und warten Sie, bis IBM MQ gestoppt wird.
- c) Wenn IBM MQ gestoppt wird, klicken Sie mit der rechten Maustaste auf das Symbol **IBM MQ** in der Taskleiste und klicken Sie dann auf **Beenden**.

2. Starten Sie den Deinstallationsprozess.

Laden Sie die komprimierte Datei mit dem Installationsimage herunter und dekomprimieren Sie sie anschließend in einem temporären Verzeichnis. Navigieren Sie zu diesem Verzeichnis und doppelklicken Sie anschließend auf `setup.exe`.

Das Fenster IBM MQ **Installation Launchpad** wird angezeigt.

3. Entfernen Sie IBM MQ.

a) Klicken Sie auf **IBM MQ-Installation**.

b) Klicken Sie auf **IBM MQ-Installationsprogramm starten** und klicken Sie auf **Weiter**, bis das IBM MQ-Fenster **Programmwartung** mit einer Willkommensnachricht angezeigt wird.

Falls dieses Fenster nicht angezeigt wird, ist IBM MQ for Windows aktuell nicht installiert.

c) Klicken Sie auf **Bestehende Instanz verwalten oder aktualisieren**. Wählen Sie **Installation1** aus, um sie zu entfernen. Klicken Sie auf **Weiter** und im **Programmwartungsfenster** auf **Entfernen** und dann auf **Weiter**.

Das Fenster zum Entfernen des Server-Features wird angezeigt.

d) Wählen Sie **Entfernen** aus: Vorhandene Warteschlangenmanager und deren Objekte werden entfernt.

Klicken Sie auf **Weiter**.

Es wird das Fenster 'IBM MQ entfernen' mit einer Zusammenfassung der zu entfernenden Installation angezeigt.

e) Klicken Sie auf **Entfernen**, um fortzufahren.

Wenn in einer Nachricht angezeigt wird, dass gesperrte Dateien gefunden wurden, stellen Sie sicher, dass keine IBM MQ-Programme aktiv sind. Weitere Informationen finden Sie unter [IBM MQ auf Windows-Systemen deinstallieren](#).

Nachdem IBM MQ deinstalliert wurde, wird eine entsprechende Nachricht angezeigt.

f) Klicken Sie auf **Fertigstellen**.


IBM MQ wurde erfolgreich deinstalliert.

Zugehörige Tasks

[IBM MQ auf Windows-Systemen deinstallieren](#)

Nächste Schritte

In diesem Abschnitt werden die weiteren Schritte zur Beendigung des Szenarios 'Erste Schritte in IBM MQ' beschrieben.

 Lernprogramme, die Sie bei der Installation und beim Upgrade unterstützen, finden Sie unter [Eine Sammlung von Lernprogrammen zur Installation und zum Upgrade von IBM MQ unter AIX, Linux® und Windows](#). Die Lernprogramme decken Folgendes ab:

- Vorbereiten eines Hosts für IBM MQ.
- IBM MQ -Code herunterladen.
- IBM MQ -Code installieren und deinstallieren und Fixpacks anwenden
- Upgrade von einer Version von IBM MQ auf eine andere und Verschieben eines Warteschlangenmanagers von einem Host auf einen anderen.

Die IBM MQ-Produktdokumentation enthält weitere Abschnitte, die Sie anzeigen können. Möglicherweise interessieren Sie sich für folgende Abschnitte:

- [Verwaltung IBM MQ](#)

IBM MQ stellt Steuerbefehle bereit, die Sie verwenden können. Zwei dieser Befehle werden in diesem Szenario verwendet: **crtmqm** und **strmqm**. Dieser Abschnitt gibt auch eine gute Übersicht über das Message-Queuing (Steuerung von Nachrichtenwarteschlangen).

- [IBM MQ mit MQSC-Befehlen verwalten](#)

In diesem Szenario verwenden Sie den Befehl `define qlocal('LQ1')`, um eine lokale Warteschlange mit dem Namen LQ1 zu definieren. Bei diesem Befehl handelt es sich um einen MQSC-Befehl. IBM MQ-Systemadministratoren verwalten mit diesen Befehlen ihre Warteschlangenmanager. Dieser Abschnitt gibt eine Einführung zu den Befehlen und zeigt ihre Verwendung. Eine detaillierte Beschreibung der Befehle in alphabetischer Reihenfolge finden Sie im Referenzabschnitt [MQSC-Befehle](#).

- [Warteschlangenmanager-Cluster konfigurieren](#)

Dieser Abschnitt beschreibt, wie Warteschlangenmanager in virtuellen Gruppen, sogenannten Clustern, organisiert, verwendet und verwaltet werden. Das Clustering stellt sicher, dass jeder Warteschlangenmanager innerhalb eines Clusters alle anderen Warteschlangenmanager in seinem Cluster kennt. Es vereinfacht außerdem die Verwaltung von komplexen Warteschlangenmanagernetzen.

Punkt-zu-Punkt-Szenario

Verbinden Sie zwei IBM MQ-Warteschlangenmanager in einer Punkt-zu-Punkt-Topologie, um die verteilte Steuerung von Warteschlangen zu ermöglichen.

Informationen zu diesem Vorgang

Erstellen Sie zwei Warteschlangenmanager und die entsprechenden Warteschlangen und Kanäle, um eine unidirektionale Punkt-zu-Punkt-Messaging-Infrastruktur einzurichten. Erstellen Sie die Warteschlangenmanager auf separaten Hosts, um die Kommunikation über ein Netz zu ermöglichen. Fügen Sie zur Erweiterung des Szenarios dem Kanal eine Transport Layer Security hinzu, damit eine sichere Übertragung der Daten möglich ist.

Lösung planen

Die einfachste Form der Nachrichtenübermittlung in IBM MQ ist das Punkt-zu-Punkt-Messaging. Beim Punkt-zu-Punkt-Messaging muss die sendende Anwendung über bestimmte Informationen zu der empfangenden Anwendung verfügen, damit sie Nachrichten senden kann. Die sendende Anwendung benötigt zum Beispiel eine Methode zur Adressierung der fernen Warteschlange. Nutzen Sie das Punkt-zu-Punkt-Messaging zum Senden einer Nachricht an einen fernen Warteschlangenmanager mit einer Beispielanwendung.

Übersicht: bereitgestellte logische Topologie

Hier finden Sie eine Beschreibung der bereitgestellten logischen Topologie nach Ausführung des Szenarios.

Die Punkt-zu-Punkt-Infrastruktur ermöglicht ein unidirektionales Messaging zwischen Warteschlangenmanagern auf anderen Hostmaschinen. Der Warteschlangenmanager 1 (QM1) auf Host 1 sendet Nachrichten an den Warteschlangenmanager 2 (QM2) auf Host 2. Nach Abschluss dieses Szenarios sieht die bereitgestellte Topologie wie in [Abbildung 1](#) dargestellt aus.

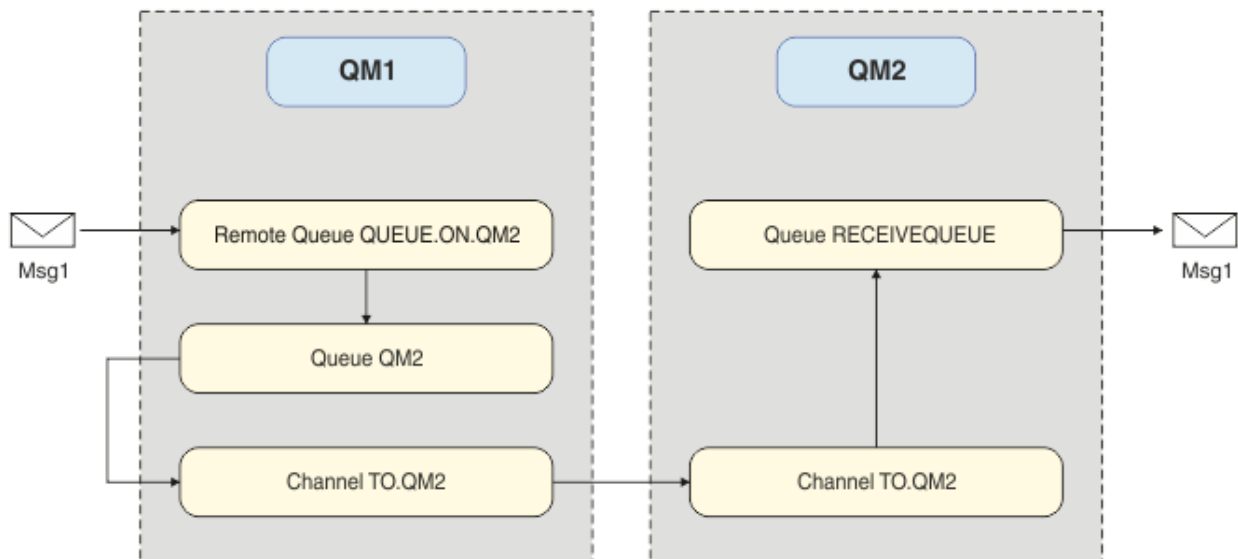


Abbildung 2. QM1 sendet eine Nachricht an QM2

Basiskonzepte und wichtige Begriffe

In diesem Abschnitt werden die Basiskonzepte und Schlüsselbegriffe beschrieben, die Sie kennen sollten, bevor Sie das Punkt-zu-Punkt-Szenario verwenden.

Basiskonzepte

IBM MQ ermöglicht es Anwendungen, Nachrichten aus einer Warteschlange zu lesen und in eine Warteschlange zu schreiben. Die Anwendung, die die Nachricht liest, ist unabhängig von der Anwendung, die die Nachricht schreibt. Es wird auch nicht vorausgesetzt, dass die beiden Anwendungen gleichzeitig aktiv sind. Wenn keine Anwendung zum Lesen der Nachricht verfügbar ist, wird die Nachricht in die IBM MQ-Warteschlange eingereiht, bis sie von einer Anwendung gelesen wird.

Schlüsselbegriffe

Es folgt eine Liste mit Schlüsselbegriffen zur Steuerung von Nachrichtenwarteschlangen (Message-Queuing).

Schlüsselbegriffe zum Message-Queuing.

Laufzeit	Beschreibung
Warteschlangenmanager	Der Warteschlangenmanager ist zuständig für die Verwaltung aller ihm zugeordneten Warteschlangen; er muss außerdem alle Nachrichten, die er empfängt, in den entsprechenden Warteschlangen speichern.
Nachrichten	Eine Nachricht ist eine Bytezeichenfolge, die Informationen für die Anwendungen bereitstellt, die diese Nachricht verwenden. Mithilfe von Nachrichten werden Informationen von einem Anwendungsprogramm an ein anderes übertragen. Die Anwendungen können auf demselben oder auf verschiedenen Computern aktiv sein.
Lokale Warteschlangen	Eine lokale Warteschlange ist eine Datenstruktur, in der Nachrichten gespeichert werden. Dabei kann es sich um eine normale Warteschlange oder eine Übertragungswarteschlange handeln. Eine normale Warteschlange enthält Nachrichten, die von einer Anwendung gelesen werden sollen, die die Nachricht direkt vom Warteschlangenmanager liest. Eine Übertragungswarteschlange enthält Nachrichten, die an einen anderen Warteschlangenmanager übertragen werden sollen.
Ferne Warteschlangen	Über eine ferne Warteschlange wird eine Nachricht an einen anderen Warteschlangenmanager adressiert.
Kanäle	Über Kanäle werden Nachrichten zwischen Warteschlangenmanagern gesendet und empfangen.
Empfangsprogramme	Empfangsprogramme sind Prozesse, die Netzanforderungen von anderen Warteschlangenmanagern oder Clientanwendungen annehmen und die zugeordneten Kanäle starten.

Lösung implementieren

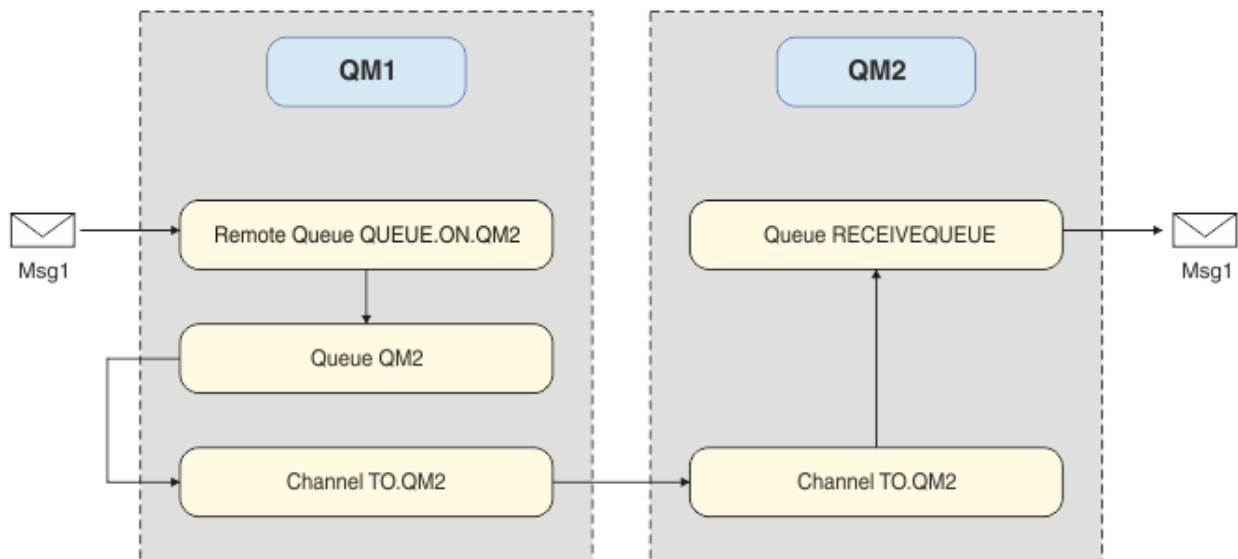
Implementieren Sie die Lösung im Szenario. Erstellen Sie zwei IBM MQ-Warteschlangenmanager auf zwei separaten Hosts, und zwar den Quellen-Warteschlangenmanager für das Senden von Nachrichten und den Ziel-Warteschlangenmanager für den Empfang von Nachrichten.

Vorbereitende Schritte

Der Ausgangspunkt für dieses Szenario ist eine bereits vorhandene und geprüfte IBM MQ-Installation. Zur Installation von IBM MQ folgen Sie den Schritten unter [IBM MQ-Server unter Windows installieren](#).

Informationen zu diesem Vorgang

Erstellen Sie zwei Warteschlangenmanager mithilfe der Befehlszeilenschnittstelle und definieren Sie die erforderlichen Empfangsprogramme, Warteschlangen und Kanäle. Die bereitgestellte logische Topologie zeigt die Funktionen, die durch die Implementierung der Lösung hinzugefügt wurden.



Warteschlangenmanager erstellen

Erstellen Sie einen IBM MQ-Warteschlangenmanager, um Nachrichten an den Ziel-Warteschlangenmanager zu senden.

Vorbereitende Schritte

- IBM MQ muss bereits installiert sein. Weitere Informationen zur Installation von IBM MQ finden Sie im Abschnitt [Installation und Deinstallation](#).

Informationen zu diesem Vorgang

Erstellen Sie den IBM MQ-Warteschlangenmanager mithilfe der Befehlszeilenschnittstelle.

Vorgehensweise

1. Erstellen Sie einen Warteschlangenmanager namens QM1. Geben Sie in der Befehlszeile Folgendes ein:

```
crtmqm QM1
```

Die folgenden Nachrichten werden angezeigt, um zu bestätigen, dass der Warteschlangenmanager erstellt wird:

```
IBM MQ queue manager created.
Creating or replacing default objects for QM1.
Default objects statistics : 61 created. 0 replaced. 0 failed.
Completing setup.
Setup completed.
```

2. Starten Sie den Warteschlangenmanager. Geben Sie in der Befehlszeile Folgendes ein:

```
strmqm QM1
```

Die folgenden Nachrichten werden angezeigt, um zu bestätigen, dass der Warteschlangenmanager gestartet wird:

```
IBM MQ queue manager 'QM1' starting.
5 log records accessed on queue manager 'QM1' during the log replay phase.
Log replay for queue manager 'QM1' complete.
```

```
Transaction manager state recovered for queue manager 'QM1'.  
IBM MQ queue manager 'QM1' started.
```

Ergebnisse

Der IBM MQ-Warteschlangenmanager QM1 wurde erstellt und gestartet.

Nächste Schritte

Wenn Sie die Warteschlangen erstellen möchten, die in Verbindung mit QM1 verwendet werden sollen, befolgen Sie die Anweisungen unter [„Warteschlangen erstellen“](#) auf Seite 24.

Warteschlangen erstellen

Erstellen Sie IBM MQ-Warteschlangen, die vom IBM MQ-Warteschlangenmanager verwaltet werden.

Vorbereitende Schritte

Sie müssen über einen IBM MQ-Warteschlangenmanager verfügen, der wie im Abschnitt [„Warteschlangenmanager erstellen“](#) auf Seite 23 beschrieben eingerichtet wurde.

Informationen zu diesem Vorgang

Starten Sie die **MQSC**-Schnittstelle, um Objekte zu verwalten, die mit dem Warteschlangenmanager verbunden sind. Erstellen Sie eine Übertragungswarteschlange und eine Definition einer fernen Warteschlange. Verlassen Sie die **MQSC**-Schnittstelle.

Vorgehensweise

1. Geben Sie in der Befehlszeile Folgendes ein:

```
runmqsc QM1
```

Nach einer Bestätigungsnachricht ist das Tool bereit für Befehle.

2. Erstellen Sie eine Übertragungswarteschlange namens QM2. Es hat sich bewährt, der Übertragungswarteschlange denselben Namen wie dem fernen Warteschlangenmanager zu geben. Geben Sie in der Schnittstelle für WebSphere MQ-Scriptbefehle Folgendes ein:

```
DEFINE QLOCAL(QM2) DESCR('Transmission queue to QM2') USAGE(XMITQ)
```

Die Übertragungswarteschlange wird erstellt.

3. Erstellen Sie eine Definition einer fernen Warteschlange. Geben Sie dieser den Namen `QUEUE.ON.QM2`. Die Definition der fernen Warteschlange muss auf den Namen der lokale Warteschlange auf dem fernen Host verweisen. Geben Sie in der Schnittstelle für WebSphere MQ-Scriptbefehle Folgendes ein:

```
DEFINE QREMOTE(QUEUE.ON.QM2) DESCR('Remote queue for QM2') XMITQ(QM2) RNAME(RECEIVEQUEUE)  
RQMNAME(QM2)
```

Die Definition der fernen Warteschlange wird erstellt.

4. Geben Sie `end` ein, um die MQSC-Schnittstelle zu beenden.

Nächste Schritte

Befolgen Sie zur Erstellung des Senderkanals, der für die Verbindung mit dem Ziel-Warteschlangenmanager verwendet wird, die Anweisungen im Abschnitt [„Senderkanal erstellen“](#) auf Seite 25.

Senderkanal erstellen

Erstellen Sie den Senderkanal auf dem Quellen-Warteschlangenmanager. Der Kanal wird für die Herstellung einer Verbindung zum Ziel-Warteschlangenmanager verwendet.

Vorbereitende Schritte

Befolgen Sie die Anweisungen im Abschnitt [„Kanäle für die Verwendung von TLS \(Transport Layer Security\) erstellen“](#) auf Seite 31, um einen Kanal zu erstellen, der TLS verwendet. Falls Sie die Lösung zunächst ohne TLS-Sicherheit testen möchten, können Sie diesen Schritt zu einem späteren Zeitpunkt ausführen.

Informationen zu diesem Vorgang

Starten Sie die **MQSC**-Schnittstelle, um Objekte zu verwalten, die mit dem Warteschlangenmanager verbunden sind, und um den Senderkanal zu erstellen. Dieser Kanal wird für die Verbindung mit dem Ziel-Warteschlangenmanager QM2 verwendet.

Vorgehensweise

1. Geben Sie in der Befehlszeile Folgendes ein:

```
runmqsc QM1
```

Nach einer Bestätigungsnachricht ist das Tool bereit für Befehle.

2. Erstellen Sie den Senderkanal TO.QM2. Geben Sie in der Schnittstelle für WebSphere MQ-Scriptbefehle Folgendes ein:

```
DEFINE CHANNEL(TO.QM2) CHLTYPE(SDR) CONNAME(' remoteHost ') TRPTYPE(TCP) XMITQ(QM2)
```

Anmerkung: Die Variable *remoteHost* ist der Hostname oder die IP-Adresse des Zielwarteschlangenmanagers.

Der Senderkanal wird erstellt.

Nächste Schritte

Befolgen Sie zur Erstellung der Topologie mit verteilten Warteschlangenmanagern die Anweisungen im Abschnitt [„Topologie mit verteilten Warteschlangenmanagern erstellen“](#) auf Seite 25.

Topologie mit verteilten Warteschlangenmanagern erstellen

Die einfachste Form der Nachrichtenübermittlung in IBM MQ ist das Punkt-zu-Punkt-Messaging. Beim Punkt-zu-Punkt-Messaging muss die sendende Anwendung über bestimmte Informationen zu der empfangenden Anwendung verfügen, damit sie Nachrichten senden kann. Die sendende Anwendung benötigt zum Beispiel eine Methode zur Adressierung der fernen Warteschlange. Nutzen Sie das Punkt-zu-Punkt-Messaging zum Senden einer Nachricht an einen zweiten Warteschlangenmanager mit einer Beispielanwendung.

Vorbereitende Schritte

Sie müssen den Quellen-Warteschlangenmanager gemäß der Beschreibung im Abschnitt [„Warteschlangenmanager erstellen“](#) auf Seite 23 eingerichtet haben.

Informationen zu diesem Vorgang

Erstellen Sie den Ziel-Warteschlangenmanager auf einem fernen Host. Überprüfen Sie die Kommunikation zwischen den Quellen- und Ziel-Warteschlangenmanagern mithilfe der Beispielanwendungen.

Warteschlangenmanager erstellen

Erstellen Sie einen IBM MQ-Warteschlangenmanager, um Nachrichten von einem fernen Warteschlangenmanager zu empfangen.

Vorbereitende Schritte

IBM MQ muss bereits installiert sein. Weitere Informationen zur Installation von IBM MQ finden Sie im Abschnitt [IBM MQ-Server unter Windows installieren](#).

Informationen zu diesem Vorgang

Erstellen Sie den IBM MQ-Warteschlangenmanager mithilfe der Befehlszeilenschnittstelle.

Vorgehensweise

1. Erstellen Sie einen Warteschlangenmanager namens QM2. Geben Sie in der Befehlszeile Folgendes ein:

```
crtmqm QM2
```

Folgende Nachrichten werden angezeigt:

```
IBM MQ queue manager created.  
Creating or replacing default objects for QM2.  
Default objects statistics : 61 created. 0 replaced. 0 failed.  
Completing setup.  
Setup completed.
```

2. Starten Sie den Warteschlangenmanager. Geben Sie in der Befehlszeile Folgendes ein:

```
strmqm QM2
```

Die folgenden Nachrichten werden angezeigt, um zu bestätigen, dass der Warteschlangenmanager gestartet wird:

```
IBM MQ queue manager 'QM2' starting.  
5 log records accessed on queue manager 'QM2' during the log replay phase.  
Log replay for queue manager 'QM2' complete.  
Transaction manager state recovered for queue manager 'QM2'.  
IBM MQ queue manager 'QM2' started.
```

Ergebnisse

Der IBM MQ-Warteschlangenmanager QM2 wurde erstellt und gestartet.

Nächste Schritte

Wenn Sie die Warteschlange erstellen möchten, die in Verbindung mit QM2 verwendet werden soll, befolgen Sie die Anweisungen unter [„Warteschlange erstellen“](#) auf Seite 26.

Warteschlange erstellen

Erstellen Sie die lokale Warteschlange für den Empfang von Nachrichten auf dem Ziel-Warteschlangenmanager sowie das Empfangsprogramm für die Verbindung am eingehenden Kanal.

Informationen zu diesem Vorgang

Nach dem Start des Scripting-Tools **runmqsc** können Sie WebSphere MQ-Scriptbefehle zum Erstellen einer lokalen Warteschlange und eines Empfangsprogramms verwenden.

Vorgehensweise

1. Starten Sie das Scripting-Tool mit folgendem Befehl:

```
runmqsc QM2
```

In einer Nachricht wird bestätigt, dass das Tool gestartet wurde.

- Erstellen Sie eine lokale Warteschlange namens RECEIVEQUEUE. Die Warteschlange muss denselben Namen haben, auf den in der Definition der fernen Warteschlange auf dem Quellenwarteschlangenmanager verwiesen wird. Geben Sie in der Schnittstelle für WebSphere MQ-Scriptbefehle Folgendes ein:

```
DEFINE QLOCAL(RECEIVEQUEUE) DESCR('Receiving queue')
```

Die lokale Warteschlange wird erstellt.

- Erstellen Sie ein Empfangsprogramm namens LISTENER1. Geben Sie in der Schnittstelle für WebSphere MQ-Scriptbefehle Folgendes ein:

```
DEFINE LISTENER(LISTENER1) TRPTYPE(TCP) PORT(1414) CONTROL(QMGR)
```

Anmerkung: Port 1414 ist der Standardport für IBM MQ. Wenn Sie eine andere Portnummer auswählen, müssen Sie diese dem CONNAME des Senderkanals auf dem sendenden Warteschlangenmanager hinzufügen.

- Starten Sie das Empfangsprogramm, damit es bereit für eingehende Verbindungen ist. Geben Sie in der Schnittstelle für WebSphere MQ-Scriptbefehle Folgendes ein:

```
START LISTENER(LISTENER1)
```

Anmerkung: Da das Empfangsprogramm mit der Option CONTROL (QMGR) erstellt wurde, wird es beim nächsten Start des Warteschlangenmanagers automatisch gestartet.

- Geben Sie end ein, um die **MQSC**-Schnittstelle zu beenden.

Nächste Schritte

Wenn Sie den Empfängerkanal erstellen möchten, um die Verbindung zwischen den Quellen- und Ziel-Warteschlangenmanagern einzurichten, befolgen Sie die Anweisungen im Abschnitt [„Empfängerkanal erstellen“](#) auf Seite 27.

Empfängerkanal erstellen

Erstellen Sie den Empfängerkanal für den Ziel-Warteschlangenmanager, um die Kommunikation zwischen den Quellen- und Ziel-Warteschlangenmanagern zu ermöglichen.

Vorbereitende Schritte

Befolgen Sie die Anweisungen im Abschnitt [„Kanäle für die Verwendung von TLS \(Transport Layer Security\) erstellen“](#) auf Seite 31, um einen Kanal zu erstellen, der TLS verwendet. Falls Sie die Lösung zunächst ohne TLS-Sicherheit testen möchten, können Sie diesen Schritt zu einem späteren Zeitpunkt ausführen.

Informationen zu diesem Vorgang

Verwenden Sie die Schnittstelle **MQSC** (WebSphere MQ-Scriptbefehle) für die Erstellung eines Empfängerkanals, der von QM2 verwaltet wird.

Vorgehensweise

- Geben Sie in der Befehlszeile Folgendes ein:

```
runmqsc QM2
```

Nach einer Bestätigungsnachricht ist das Tool bereit für Befehle.

- Erstellen Sie einen Empfängerkanal namens TO.QM2. Der Kanal muss denselben Namen wie der Senderkanal beim Quellenwarteschlangenmanager haben. Geben Sie in der Schnittstelle für WebSphere MQ-Scriptbefehle Folgendes ein:

```
DEFINE CHANNEL(TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP)
```

Der Empfängerkanal wird erstellt.

Nächste Schritte

Wenn Sie den Senderkanal auf dem Quellen-Warteschlangenmanager starten möchten, der wiederum den Empfängerkanal auf dem Ziel-Warteschlangenmanager initialisiert, folgen Sie den Anweisungen im Abschnitt „[Senderkanal starten](#)“ auf Seite 28.

Senderkanal starten

Starten Sie den Senderkanal auf dem Quellen-Warteschlangenmanager. Der Empfängerkanal auf dem Ziel-Warteschlangenmanager wird ebenfalls gestartet. Nachrichten können vom Quellen-Warteschlangenmanager an den Ziel-Warteschlangenmanager gesendet werden.

Informationen zu diesem Vorgang

Starten Sie die **MQSC**-Schnittstelle, um Objekte zu verwalten, die mit dem Warteschlangenmanager verbunden sind. Starten Sie den Senderkanal, um eine Verbindung zum Ziel-Warteschlangenmanager herzustellen und dadurch die Kommunikation zu ermöglichen. Sobald der Quellenkanal gestartet wird, wird der Empfängerkanal automatisch ebenfalls gestartet.

Vorgehensweise

1. Geben Sie in der Befehlszeile Folgendes ein:

```
runmqsc QM1
```

Nach einer Bestätigungsnachricht ist das Tool bereit für Befehle.

2. Starten Sie den Senderkanal auf dem Quellenwarteschlangenmanager. Geben Sie in der Schnittstelle für WebSphere MQ-Scriptbefehle Folgendes ein:

```
START CHANNEL(TO.QM2)
```

Der Senderkanal wird gestartet und der Empfängerkanal auf dem Ziel-Warteschlangenmanager wird ebenfalls gestartet.

3. Vergewissern Sie sich, dass der Kanal aktiv ist. Geben Sie in der Schnittstelle für WebSphere MQ-Scriptbefehle Folgendes ein:

```
DISPLAY CHSTATUS(TO.QM2)
```

Wenn der Kanal aktiv ist, gibt er **STATUS (RUNNING)** zurück. Gibt er einen anderen Wert als **STATUS** zurück, überprüfen Sie das [Fehlerprotokoll](#).

Nächste Schritte

Prüfen Sie mithilfe der Anweisungen im Abschnitt „[Lösung prüfen](#)“ auf Seite 28, ob der Quellen-Warteschlangenmanager Nachrichten an den Ziel-Warteschlangenmanager senden kann.

Lösung prüfen

Prüfen Sie, ob der Quellen-Warteschlangenmanager eine Nachricht in die ferne Warteschlange einreihen kann. Vergewissern Sie sich außerdem, dass der Ziel-Warteschlangenmanager die Nachricht aus der Warteschlange abrufen kann.

Informationen zu diesem Vorgang

Überprüfen Sie die Lösung mithilfe der Beispielanwendungen **amqspvt** und **amqsgt**.

Vorgehensweise

1. Senden Sie vom Quellen-Warteschlangenmanager aus eine Nachricht an den Ziel-Warteschlangenmanager QM2.
 - a) Geben Sie in der Befehlszeilenschnittstelle Folgendes ein:

```
amqsput QUEUE.ON.QM2 QM1
```

Sie müssen für das Senden der Nachricht an den Ziel-Warteschlangenmanager den Namen der Definition einer fernen Warteschlange verwenden.

Folgende Nachricht wird angezeigt:

```
Sample AMQSPUT0 start  
target queue is QUEUE.ON.QM2
```

- b) Geben Sie `Hello world.` ein und drücken Sie zweimal die Eingabetaste.
2. Rufen Sie die Nachricht auf dem Zielwarteschlangenmanager ab.
 - a) Geben Sie in der Befehlszeilenschnittstelle Folgendes ein:

```
amqsget RECEIVEQUEUE QM2
```

Folgende Nachricht wird angezeigt:

```
Sample AMQSGET0 start  
message <Hello world.>  
no more messages  
Sample AMQSGET0 end
```

Ergebnisse

Der Ziel-Warteschlangenmanager hat die Nachricht vom Quellen-Warteschlangenmanager erhalten. Dadurch wurde verifiziert, dass eine Punkt-zu-Punkt-Kommunikation möglich ist.

Nächste Schritte

Wenn Sie der Lösung Sicherheitsfunktionen hinzufügen möchten, befolgen Sie die Anweisungen im Abschnitt [„Punkt-zu-Punkt-Topologie schützen“](#) auf Seite 29.

Punkt-zu-Punkt-Topologie schützen

Schützen Sie die Punkt-zu-Punkt-Topologie, damit Nachrichten in einer Produktionsumgebung übertragen werden können.

Informationen zu diesem Vorgang

Schützen Sie die Objekte der Quellen- und Ziel-Warteschlangenmanager durch die Erteilung der jeweils richtigen Zugriffsebene. Definieren Sie, welche Benutzergruppen Zugriff auf die Warteschlangen und Warteschlangenmanager erhalten. Schützen Sie die Netzverbindung mithilfe von digital signierten Zertifikaten, die für die Verbindung mit Transport Layer Security (TLS) verwendet werden.

Objekte des Quellen-Warteschlangenmanagers schützen

Legen Sie die Berechtigungswerte für die Objekte auf dem Quellen-Warteschlangenmanager fest.

Informationen zu diesem Vorgang

Mit dem Befehl `setmqaut` können Sie der Benutzergruppe, die die Anwendung ausführt, Berechtigungen erteilen.

Vorgehensweise

1. Wenn Sie der angegebenen Benutzergruppe die Berechtigung *connect* für den Warteschlangenmanager erteilen möchten, geben Sie in der Befehlszeilenschnittstelle Folgendes ein:

```
setmqaut -m QM1 -t qmgr -g userGroup +connect
```

2. Wenn Sie der angegebenen Benutzergruppe die Berechtigung *put* on der Definition einer fernen Warteschlange erteilen möchten, geben Sie in der Befehlszeilenschnittstelle Folgendes ein:

```
setmqaut -m QM1 -t q -n "QUEUE.ON.QM2" -g userGroup +put
```

Objekte des Ziel-Warteschlangenmanagers schützen

Legen Sie die Berechtigungswerte für die Objekte auf dem Ziel-Warteschlangenmanager fest.

Informationen zu diesem Vorgang

Mit dem Befehl **setmqaut** können Sie der Benutzergruppe, die die Anwendung ausführt, Berechtigungen erteilen.

Vorgehensweise

1. Wenn Sie der angegebenen Benutzergruppe die Berechtigung *connect* für den Warteschlangenmanager erteilen möchten, geben Sie in der Befehlszeilenschnittstelle Folgendes ein:

```
setmqaut -m QM2 -t qmgr -g userGroup +connect
```

2. Wenn Sie der angegebenen Benutzergruppe die Berechtigung *get* in der Definition einer fernen Warteschlange erteilen möchten, geben Sie in der Befehlszeilenschnittstelle Folgendes ein:

```
setmqaut -m QM2 -t q -n "RECEIVEQUEUE" -g userGroup +get
```

Netz schützen

Schützen Sie die Netzverbindungen zwischen der Quelle und fernen Warteschlangenmanagern.

Informationen zu diesem Vorgang

Überprüfen Sie die Authentizität der Quelle und der fernen Warteschlangenmanager mithilfe von signierten Zertifikaten. Übertragen Sie Nachrichten unter Verwendung eines TLS-Netzes, um die Nachrichten zu verschlüsseln.

Warteschlangenmanager für die Verwendung von TLS (Transport Layer Security) vorbereiten

Im Schlüsselrepositorium des IBM MQ-Warteschlangenmanagers werden das persönliche Zertifikat des Warteschlangenmanagers und das öffentliche Zertifikat einer Zertifizierungsstelle gespeichert. Die Anforderung des persönlichen Zertifikats des IBM MQ-Warteschlangenmanagers muss von der Zertifizierungsstelle signiert werden, während das öffentliche Zertifikat von den anderen Entitäten für die Authentifizierung des IBM MQ-Warteschlangenmanagers verwendet wird.

Vorbereitende Schritte

Das öffentliche Zertifikat der Zertifizierungsstelle muss Ihnen in einer Datei vorliegen.

Informationen zu diesem Vorgang

Erstellen Sie das Schlüsselrepository des IBM MQ-Warteschlangenmanagers, importieren Sie das Unterzeichnerzertifikat der Zertifizierungsstelle und erstellen Sie die Anforderung des persönlichen Zertifikats vom Warteschlangenmanager.

Vorgehensweise

1. Erstellen Sie eine CMS-Schlüsselrepository-Datei für den Warteschlangenmanager mit dem Namen `key.kdb`. Navigieren Sie zum Verzeichnis `Qmgrs\QM1\ssl` und geben Sie in der Befehlszeile Folgendes ein:

```
runmqakm -keydb -create -db key.kdb -pw passw0rd -type cms -stash
```

Anmerkung: In diesem einfachen Beispiel wurde das Kennwort `passw0rd` verwendet. Sie können auch ein anderes Kennwort verwenden, müssen dann aber in den folgenden Befehlen das Kennwort entsprechend ändern.

2. Fügen Sie die Datei mit dem Zertifikat der Zertifizierungsstelle dem Schlüsselrepository hinzu. Geben Sie dazu in der Befehlszeile Folgendes ein:

```
runmqakm -cert -add -file CA-certificate-file -db key.kdb -pw passw0rd -label TrustedCA
```

3. Fordern Sie ein persönliches Zertifikat an, das in eine Anforderungsdatei mit dem Namen `QM1req.req` beschrieben wird.

Geben Sie in der Befehlszeile Folgendes ein:

```
runmqakm -certreq -create -db key.kdb -pw passw0rd -label ibmwebspheremqmqm1  
-dn CN="QM1" -size 1024 -file QM1req.req -sig_alg SHA1WithRSA
```

In diesem Beispiel wird die Standardzertifikatsbezeichnung verwendet. Sie können auch einen anderen Namen eingeben. Details finden Sie unter [Bezeichnungen für digitale Zertifikate](#).

4. Senden Sie die Zertifikatanforderungsdatei an Ihre Zertifizierungsstelle, die ein digital signiertes Zertifikat ausstellen wird. Kopieren Sie die empfangene signierte Zertifikatsdatei in ein geeignetes Verzeichnis, aus dem sie in das Schlüsselrepository des Warteschlangenmanagers übertragen werden kann.
5. Laden Sie das signierte persönliche Zertifikat in das Schlüsselrepository des Warteschlangenmanagers.

```
runmqakm -cert -receive -file Signed-certificate-file -db key.kdb -pw passw0rd -format ascii
```

6. Führen Sie diese Schritte für jeden Warteschlangenmanager aus und ändern Sie dabei den Warteschlangenmanager-Namen entsprechend.

Nächste Schritte

Um eine sichere Kommunikation über die Sender- und Empfängerkanäle zu ermöglichen, befolgen Sie die Anweisungen im Abschnitt [„Kanäle für die Verwendung von TLS \(Transport Layer Security\) erstellen“](#) auf Seite 31.

Kanäle für die Verwendung von TLS (Transport Layer Security) erstellen

Erstellen Sie einen neuen Kanal, der TLS verwendet, um eine Verbindung zu erstellen.

Vorbereitende Schritte

Um über einen Kanal zu kommunizieren, der TLS verwendet, müssen Sie für jedes Ende der Verbindung zuerst die erforderlichen Zertifikate haben. Befolgen Sie zur Erstellung der erforderlichen Zertifikate die Anweisungen im Abschnitt [„Warteschlangenmanager für die Verwendung von TLS \(Transport Layer Security\) vorbereiten“](#) auf Seite 30.

Informationen zu diesem Vorgang

Definieren Sie in der MQSC-Schnittstelle Kanäle mit TLS-Attributen. Mit dem Schlüsselwort REPLACE kann diese Task sogar ausgeführt werden, wenn Sie Ihre Kanäle in einem früheren Schritt ohne TLS definiert haben.

Vorgehensweise

1. Geben Sie in der Befehlszeile Folgendes ein:

```
runmqsc QM1
```

2. Erstellen Sie auf QM1 den Senderkanal TO.QM2. Geben Sie dazu in der MQSC-Schnittstelle Folgendes ein:

```
DEFINE CHANNEL(TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP)
CONNNAME('remoteHost') XMITQ(QM2)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256)
DESCR('Sender channel using TLS from QM1 to QM2')
REPLACE
```

Anmerkung: Die Variable *fernerHost* ist der Hostname oder die IP-Adresse des Zielwarteschlangenmanagers.

Sie können für den Kanal das Attribut CERTLABL festlegen. In diesem Fall muss er mit dem Wert im Parameter **-label** des Befehls **runmqakm** übereinstimmen, den Sie zuvor in Schritt „3“ auf Seite 31 von „Warteschlangenmanager für die Verwendung von TLS (Transport Layer Security) vorbereiten“ auf Seite 30 ausgeführt haben. Weitere Informationen zu Zertifikatsbezeichnungen finden Sie im Abschnitt [Digitale Zertifikatsbezeichnungen - Anforderungen](#).

3. Geben Sie end ein, um die MQSC-Schnittstelle zu beenden.
4. Geben Sie in der Befehlszeile Folgendes ein:

```
runmqsc QM2
```

5. Erstellen Sie auf QM2 den Empfängerkanal TO.QM2. Geben Sie dazu in der MQSC-Schnittstelle Folgendes ein:

```
DEFINE CHANNEL(TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256) SSLCAUTH(REQUIRED)
DESCR('Receiver channel using TLS from QM1 to QM2')
REPLACE
```

6. Geben Sie end ein, um die MQSC-Schnittstelle zu beenden.

Nächste Schritte

Prüfen Sie mithilfe der Anweisungen im Abschnitt „Lösung prüfen“ auf Seite 28, ob der Quellen-Warteschlangenmanager unter Verwendung von Transport Layer Security Nachrichten an den Ziel-Warteschlangenmanager senden kann.

Streaming-Warteschlangen

Mit dem Feature "Streaming-Warteschlangen" von IBM MQ können Sie eine Warteschlange so konfigurieren, dass eine nahezu identische Kopie jeder Nachricht in eine zweite Warteschlange eingereicht wird.

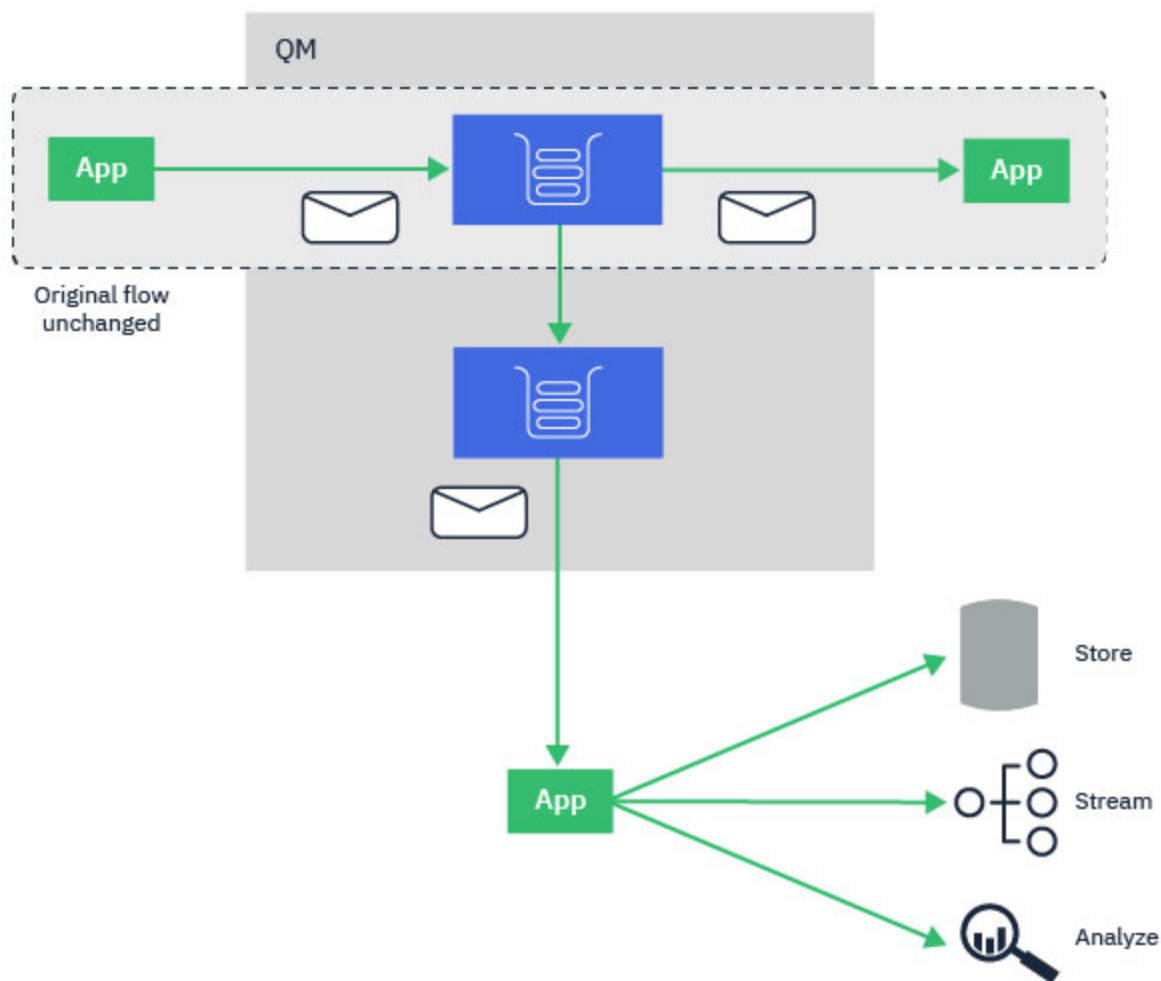
Streaming-Warteschlangen können in bestimmten Szenarien nützlich sein, in denen Sie eine Kopie Ihrer Nachrichten erstellen müssen. For example:

- Nachrichten an Apache Kafka über den Kafka-Connect-Quellconnector für IBM MQ streamen. Weitere Informationen finden Sie unter [kafka_connect_mq_source](#).
- Analyse der Daten, die das System durchlaufen.

- Speichern von Nachrichten für die Wiederherstellung zu einem späteren Zeitpunkt.
- Erfassen einer Gruppe von Nachrichten, die in Entwicklungs- und Testsystemen verwendet werden sollen.
- IBM MQ-Ereignisnachrichten aus den Systemereigniswarteschlangen konsumieren und zusätzliche Kopien an andere Warteschlangen oder Topics senden.

In allen diesen Szenarien können Sie Streaming-Warteschlangen konfigurieren, um sicherzustellen, dass die ursprünglichen Nachrichten vom Streaming-Prozess nicht beeinflusst werden. Auf diese Weise wird sichergestellt, dass Kerngeschäftsanwendungen keine Auswirkungen auf das Streaming haben.

In der folgenden Veranschaulichung wird Folgendes dargestellt:



Zugehörige Konzepte

[Sicherheit für Streaming-Warteschlangen](#)
[Streaming-Warteschlangen und AMS](#)

Konfiguration der Streaming-Warteschlangen

Die Streaming-Warteschlangen-Feature von IBM MQ wird vom Administrator in einzelnen Warteschlangen konfiguriert, und die Nachrichten werden vom Warteschlangenmanager gestreamt, nicht von der Anwendung selbst.

Das bedeutet, dass die Anwendung, die Nachrichten in die ursprüngliche Warteschlange stellt, in fast allen Fällen gar nicht weiß, dass ein Streaming stattfindet. Auch die Anwendung, die Nachrichten aus der ursprünglichen Warteschlange abrufen, weiß nicht, dass ein Nachrichten-Streaming stattgefunden hat.

Anmerkung: Die Version der IBM MQ-Clientbibliothek muss nicht aktualisiert werden, um die Streaming-Warteschlangen zu nutzen, und die ursprünglichen Nachrichten werden durch den Streaming-Prozess nicht verändert.

Sie können Streaming-Warteschlangen in einem von zwei Modi konfigurieren:

Bestmöglich

In diesem Modus hält der Warteschlangenmanager es für wichtiger, dass die Zustellung der ursprünglichen Nachricht nicht durch die Zustellung der gestreamten Nachricht beeinflusst wird.

Wenn die ursprüngliche Nachricht zugestellt werden kann, die gestreamte Nachricht jedoch nicht, wird die ursprüngliche Nachricht immer noch an die zugehörige Warteschlange zugestellt. Dieser Modus eignet sich am besten für diese Anwendungen, bei denen es wichtig ist, dass die ursprüngliche Geschäftsanwendung durch den Streaming-Prozess nicht beeinflusst wird.

Muss duplizieren

In diesem Modus stellt der Warteschlangenmanager sicher, dass sowohl die ursprüngliche Nachricht als auch die gestreamte Nachricht erfolgreich an ihre Warteschlangen zugestellt werden.

Wenn die gestreamte Nachricht aus irgendeinem Grund nicht in ihre Warteschlange zugestellt werden kann, z. B. weil die zweite Warteschlange voll ist, wird die ursprüngliche Nachricht auch nicht an die zugehörige Warteschlange zugestellt. Die Einreihungsanwendung erhält einen Fehlerursachencode und muss versuchen, die Nachricht erneut zu einzureihen.

Informationen zu den zusätzlichen Attributen, die den lokalen Warteschlangen und den Modellwarteschlangen hinzugefügt werden, die das Nachrichtenstreaming aktivieren, finden Sie unter [Vorgehensweise zur Konfiguration von Streaming-Warteschlangen](#).

Gestreamte Nachrichten

In den meisten Fällen handelt es sich bei der Kopie der Nachricht, die an die zweite Warteschlange zugestellt wurde, um ein Duplikat der ursprünglichen Nachricht. Dazu gehören alle Nachrichtendeskriptorfelder, einschließlich der Nachrichten-ID und der Korrelations-ID. Bei den gestreamten Nachrichten handelt es sich um sehr genaue Kopien der ursprünglichen Nachrichten, sodass sie leichter zu finden sind und bei Bedarf wieder in ein anderes IBM MQ-System zurückgespielt werden können.

Es gibt einige Nachrichtendeskriptorfelder, die in der gestreamten Nachricht nicht beibehalten werden. Die folgenden Änderungen werden an der gestreamten Nachricht vorgenommen, bevor sie in die zweite Warteschlange angeordnet werden:

- Der Ablauf der gestreamten Nachricht wird unabhängig vom Ablauf der ursprünglichen Nachricht auf "MQEI_UNLIMITED" gesetzt. Wenn **CAPEXPRT** für die sekundäre Warteschlange festgelegt wurde, wird ihr Wert verwendet, um die Ablaufzeit der gestreamten Nachricht festzulegen.
- Wenn eine der folgenden Berichtsoptionen in der ursprünglichen Nachricht festgelegt ist, sind sie in der gestreamten Nachricht nicht aktiviert. Damit stellen Sie sicher, dass keine unerwarteten Berichtsnachrichten an Anwendungen zugestellt werden, die nicht für den Empfang von Anwendungen konzipiert sind:
 - Aktivitätsberichte
 - Ablaufberichte
 - Ausnahmeberichte
 - Bestätigung bei Ankunft (COA)
 - Bestätigung bei Lieferung (COD)

Aufgrund des nahezu identischen Charakters der gestreamten Nachrichten wirken sich die meisten Attribute der sekundären Warteschlange nicht auf die Nachrichtendeskriptorfelder der gestreamten Nachricht aus. Beispielsweise haben die Attribute **DEFPSIST** und **DEFPTY** der sekundären Warteschlange keine Auswirkungen auf die gestreamte Nachricht.

Die folgenden Ausnahmen gelten für die gestreamte Nachricht:

- **CAPEXPRT** Attribut

Wenn die sekundäre Warteschlange mit einem Attribut **CAPEXPY** konfiguriert wurde, wird diese Ablaufbegrenzung auf den Ablauf der Datenstromnachricht angewendet.

- **DEFBIND** für Clusterwarteschlangen

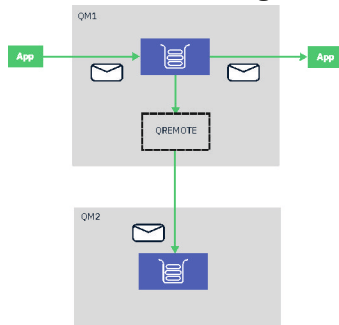
Wenn es sich bei der sekundären Warteschlange um eine Clusterwarteschlange handelt, wird die Datenstromnachricht unter Verwendung der Bindeoption eingereicht, die im Attribut **DEFBIND** der sekundären Warteschlange festgelegt ist.

Streaming in ferne und Aliaswarteschlangen

Es ist möglich, Nachrichten an ferne Warteschlangen und Aliaswarteschlangen zu übertragen. Q1 kann beispielsweise mit **STREAMQ (MY.REMOTE.Q)** konfiguriert werden, wobei **MY.REMOTE.Q** für eine ferne Warteschlangendefinition steht.

Streaming in ferne Warteschlangen

Durch das Streaming von Nachrichten aus einer lokalen Warteschlange in eine ferne Warteschlange können die doppelten Nachrichten an eine Warteschlange in einem anderen Warteschlangenmanager in Ihrem IBM MQ-Netz gesendet werden, wie in der folgenden Abbildung dargestellt:



Streaming in Aliaswarteschlangen

Durch das Streaming von Nachrichten in eine Aliaswarteschlange ist es möglich, die doppelten Nachrichten an das Ziel der Aliaswarteschlange zu senden. Da das Ziel einer Aliaswarteschlange auch ein Thema sein kann, ist es daher möglich, die doppelten Nachrichten an ein Publish/Subscribe-Thema zu senden. Alle Abonnenten des Aliasthemas erhalten eine Kopie der doppelten Nachricht. Auf diese Weise können Sie mehrere Kopien der ursprünglichen Nachricht erstellen. Die vorhandenen Regeln für Publish/Subscribe-Nachrichten werden jedoch auf die doppelte Nachricht angewendet. Dies bedeutet, dass die Nachrichten, die an Abonnenten gesendet werden, nicht identisch mit der ursprünglichen Nachricht sein werden, einschließlich:

- Es wird eine neue Nachrichten-ID angegeben.
- Eine generierte Korrelations-ID wird abhängig von der Konfiguration des Abonnements generiert.
- Das Feld "UserIdentifier", das auf den Benutzer gesetzt wird, der vom Warteschlangenmanager ausgeführt wird, ist nicht der Benutzer, der die Nachricht eingereicht hat.
- Der Name "PutApplName", der den Namen des Warteschlangenmanagers und nicht den Namen der ankommenden Anwendung anzeigt.

Anmerkungen:

1. Es ist nicht möglich, das Attribut **STREAMQ** in fernen Warteschlangen oder Aliaswarteschlangen selbst zu konfigurieren. Sie können nur Nachrichten an sie streamen, nicht von ihnen.
2. Wenn Nachrichten an einen Warteschlangenaliasnamen übertragen werden, kann das Ziel des Aliasnamens des Warteschlangenmanagers nicht sein **STREAMQ**-Attribut festlegen.

Einschränkungen für Streaming-Warteschlangen

Bestimmte Konfigurationen werden bei der Verwendung von Streaming-Warteschlangen in IBM MQ nicht unterstützt; diese werden hier dokumentiert.

In der folgenden Liste werden die Konfigurationen angegeben, die nicht unterstützt werden:

- Definition einer Kette von Warteschlangen, die zueinander streamen, zum Beispiel Q1->Q2, Q2->Q3, Q3->Q4
- Definieren einer Schleife von Streaming-Warteschlangen, z. B. Q1-> Q2, Q2-> Q1
- Definieren einer Subskription mit einem angegebenen Ziel, wobei für dieses Ziel eine STREAMQ-Definition definiert ist
- Definieren von STREAMQ in einer Warteschlange, die mit "USAGE(XMITQ)" konfiguriert ist

Anmerkung: STREAMQ kann eine ferne Warteschlange sein, aber das Attribut STREAMQ kann nicht in einer Definition einer fernen Warteschlange konfiguriert werden.

- Attribut "STREAMQ" einer dynamischen Warteschlange ändern
- STREAMQ wird auf einen beliebigen Wert gesetzt, der mit SYSTEM. * , mit Ausnahme von SYSTEM.DEFAULT.LOCAL.QUEUE
- STREAMQ in einer beliebigen Warteschlange mit dem Namen SYSTEM. * , mit den folgenden Ausnahmen:
 - SYSTEM.DEFAULT.LOCAL.QUEUE
 - SYSTEM.ADMIN.ACCOUNTING.QUEUE
 - SYSTEM.ADMIN.ACTIVITY.QUEUE
 - SYSTEM.ADMIN.CHANNEL.EVENT
 - SYSTEM.ADMIN.COMMAND.EVENT
 - SYSTEM.ADMIN.CONFIG.EVENT
 - SYSTEM.ADMIN.LOGGER.EVENT
 - SYSTEM.ADMIN.PERFM.EVENT
 - SYSTEM.ADMIN.PUBSUB.EVENT
 - SYSTEM.ADMIN.QMGR.EVENT
 - SYSTEM.ADMIN.STATISTICS.QUEUE
 - SYSTEM.DEFAULT.MODEL.QUEUE
 - SYSTEM.JMS.TEMPQ.MODEL
- "STREAMQ" auf den Namen einer Modellwarteschlange setzen

Datenstromwarteschlangen und Transaktionen

Mit dem Feature der Streaming-Warteschlangen kann eine Nachricht, die in eine Warteschlange eingereicht wurde, in eine zweite Warteschlange dupliziert werden. In den meisten Fällen werden die beiden Nachrichten in ihre jeweiligen Warteschlangen unter einer Arbeitseinheit eingereicht.

Wenn die ursprüngliche Nachricht mit dem Befehl "MQPMO_SYNCPOINT" eingereicht wurde, wird die doppelte Nachricht in die Datenstromwarteschlange unter derselben Arbeitseinheit eingereicht, die für das ursprüngliche Einreihen gestartet wurde.

Wenn das Original mit "MQPMO_NO_SYNCPOINT" eingereicht wurde, wird eine Arbeitseinheit gestartet, obwohl das ursprüngliche Einreihen keine Anforderung gestellt hat. Dies geschieht aus zwei Gründen:

1. Es stellt sicher, dass die doppelte Nachricht nicht zugestellt wird, wenn die ursprüngliche Nachricht nicht angegeben werden konnte. Das Feature Streaming-Warteschlangen liefert nur dann Nachrichten an Datenstromwarteschlangen, wenn die ursprüngliche Nachricht ebenfalls zugestellt wurde.
2. Es kann eine Leistungsverbesserung geben, wenn beide Einreichungen innerhalb einer Arbeitseinheit durchgeführt werden

Die Nachrichten werden nur dann nicht innerhalb einer Arbeitseinheit zugestellt, wenn die ursprüngliche "MQPUT" mit "MQPMO_NO_SYNCPOINT" nicht permanent ist und das **STRMQOS**-Attribut der Warteschlange auf "BESTEF" (bester Aufwand) gesetzt ist.

Anmerkungen:

1. Der zusätzliche Einreihungswert für die Datenstromwarteschlange gilt nicht für den Grenzwert "MAXUMSGS".
2. Bei einer Warteschlange, die mit dem Befehl "STRMQOS(BESTEF)" konfiguriert wurde, führt das Fehlschlagen der Übergabe der doppelten Nachricht nicht dazu, dass die Arbeitseinheit rückgängig gemacht wird.

Streaming in und aus Clusterwarteschlangen

Es ist möglich, Nachrichten aus einer lokalen Warteschlange in eine Clusterwarteschlange zu streamen und Nachrichten aus Clusterwarteschlangeninstanzen in eine lokale Warteschlange zu streamen.

Streaming in eine Clusterwarteschlange

Dies kann nützlich sein, wenn Sie über eine lokale Warteschlange verfügen, in der ursprüngliche Nachrichten zugestellt werden, und eine Kopie jeder Nachricht an eine oder mehrere Instanzen einer Clusterwarteschlange übertragen möchten. Dies kann der Fall sein, um die Arbeitslast bei der Verarbeitung der doppelten Nachrichten auszugleichen oder einfach, um doppelte Nachrichten an eine andere Warteschlange an anderer Stelle im Cluster zu streamen.

Wenn Nachrichten in eine Clusterwarteschlange gestreamt werden, werden Nachrichten mithilfe des Algorithmus für den Clusterlastausgleich verteilt. Eine Clusterwarteschlangeninstanz wird auf der Basis des Attributs "DEFBIND" der Clusterwarteschlange ausgewählt.

Wenn die Clusterwarteschlange beispielsweise mit dem Befehl "DEFBIND(OPEN)" konfiguriert ist, wird eine Instanz der Clusterwarteschlange ausgewählt, wenn die ursprüngliche Warteschlange geöffnet wird. Alle doppelten Nachrichten werden an dieselbe Clusterwarteschlangeninstanz gesendet, bis die ursprüngliche Warteschlange von der Anwendung erneut geöffnet wird.

Wenn die Clusterwarteschlange mit "DEFBIND(NOTFIXED)" konfiguriert ist, wird eine Instanz der Clusterwarteschlange für jede MQPUT-Operation ausgewählt.

Anmerkung: Sie sollten alle Clusterwarteschlangeninstanzen mit dem gleichen Wert für das Attribut "DEFBIND" konfigurieren.

Streaming aus einer Clusterwarteschlange

Dies kann hilfreich sein, wenn Sie bereits Nachrichten an mehrere Instanzen einer Clusterwarteschlange senden und eine Kopie jeder Nachricht an eine Datenstromwarteschlange auf demselben Warteschlangenmanager wie die Clusterwarteschlangeninstanz übergeben möchten.

Wenn die ursprüngliche Nachricht an eine der Clusterwarteschlangeninstanzen zugestellt wird, wird eine doppelte Nachricht vom Clusterempfängerkanal an die Datenstromwarteschlange zugestellt.

Multi V 9.4.0 Streaming-Warteschlangen zum Speichern eines Nachrichtenverlaufs verwenden

Sie können Streaming-Warteschlangen verwenden, um ein Nachrichtenprotokoll für einen begrenzten Zeitraum aufzubewahren. Dies erreichen Sie, indem Sie das Attribut CAPEXPY in der Warteschlange konfigurieren, in die Nachrichten gestreamt werden.

Einführung

Wenn Nachrichten aus einer Warteschlange in eine andere gestreamt werden, wird jeder in der Nachricht festgelegte Ablaufwert auf den Wert MQEI_UNLIMITED für die Duplikatkopie zurückgesetzt. Dies führt

standardmäßig zu einem stetigen Aufbau von Nachrichten in der Warteschlange, in die Sie streamen, wenn keine Anwendung sie konsumiert.

In diesem Szenario möchten Sie eine Kopie der Nachrichten für einen begrenzten Zeitraum aufbewahren, damit Sie darauf zugreifen können. Beispiel: Die ursprüngliche Nachricht wurde versehentlich von einer konsumierenden Anwendung entfernt.

Es ist nicht möglich, eine Kopie jeder Nachricht unbegrenzt aufzubewahren und zu verhindern, dass die Warteschlange, in die Sie streamen, voll wird, gibt es zwei Möglichkeiten:

- Führen Sie eine Anwendung aus, um die Nachrichten so häufig zu entfernen
- Konfigurieren Sie die Nachrichten mit einem Ablauf, sodass sie von IBM MQ nach einem bestimmten Zeitraum entfernt werden.

Die zweite Option kann viel bequemer sein, da es nicht erforderlich ist, eine Anwendung auszuführen und zu verwalten, nur um zu verhindern, dass die Warteschlange voll wird.

CAPEXPY konfigurieren

Im Abschnitt Kürzere Ablaufzeiten erzwingen wird beschrieben, wie CAPEXPY für eine Warteschlange konfiguriert wird. Für dieses Szenario müssen Sie das Attribut CAPEXPY für die Warteschlange festlegen, an die Sie Nachrichten streamen.

Anmerkung: Sie müssen den Wert des Attributs CAPEXPY in der Warteschlange, aus der Nachrichten gestreamt werden, nicht ändern.

Wählen Sie unter Berücksichtigung der folgenden Aspekte eine geeignete Ablaufzeit für die doppelten Nachrichten aus:

1. Wie lange Sie möglicherweise Zugriff auf die Nachrichten für benötigen
2. Was das Attribut MAXDEPTH der Warteschlange sein muss, basierend auf der Rate der Nachrichten, die in die ursprüngliche Warteschlange eingereicht werden
3. Wie viel Speicher Sie benötigen, um die doppelten Nachrichten zu speichern.

Dies erfordert möglicherweise die Berücksichtigung der Dateisystemgröße des Warteschlangenmanagers und des Attributs MAXFSIZE der Warteschlange.

Bei Bedarf auf doppelte Nachrichten zugreifen

Wenn ein Problem auftreten sollte, bei dem Sie auf einige oder alle doppelten Nachrichten zugreifen und diese wiederherstellen müssen, verwenden Sie den Befehl **dmpmqmsg**, um den Zugriff auf diese Nachrichten zu erleichtern.

dmpmqmsg bietet Optionen für:

- Nachrichten aus der Warteschlange lesen und eine Kopie in eine Datei schreiben, um später darauf zuzugreifen
- Nachrichten aus einer Datei lesen und in eine Warteschlange zurückschreiben, damit Anwendungen sie verarbeiten können

Die Nachrichtenheader können bearbeitet werden, sobald sie von **dmpmqmsg** in eine Datei geschrieben wurden. Sie können beispielsweise den Ablauf der Nachrichten auf MQEI_UNLIMITED zurücksetzen, bevor **dmqmqmsg** sie zur Verarbeitung in eine Warteschlange zurückstellt, indem Sie den EXP-Wert jeder Nachricht in der Datei in -1 ändern.

Leistungsaspekte

Das Streaming von doppelten Nachrichten in eine andere Warteschlange und deren Ablauf, wenn sie nicht mehr benötigt werden, verursacht geringe Kosten. Die Kosten sind jedoch viel geringer als das manuelle Einreihen einer Kopie in eine zweite Warteschlange und das Löschen der Kopie durch eine Anwendung nach einem bestimmten Zeitraum. Der Streaming-Warteschlangenleistungsbericht enthält weitere Informationen zur Leistung dieses Szenarios.

Zugehörige Konzepte

[Sicherheit für Streaming-Warteschlangen](#)

[Streaming-Warteschlangen und AMS](#)

Publish/Subscribe-Szenarios

Zwei Gruppen von Szenarios, die die Verwendung von Publish/Subscribe-Clustern und Publish/Subscribe-Hierarchien demonstrieren.

Die verfügbaren Publish/Subscribe-Szenarios werden in den folgenden Unterabschnitten beschrieben:

Szenario: Publish/Subscribe-Cluster erstellen

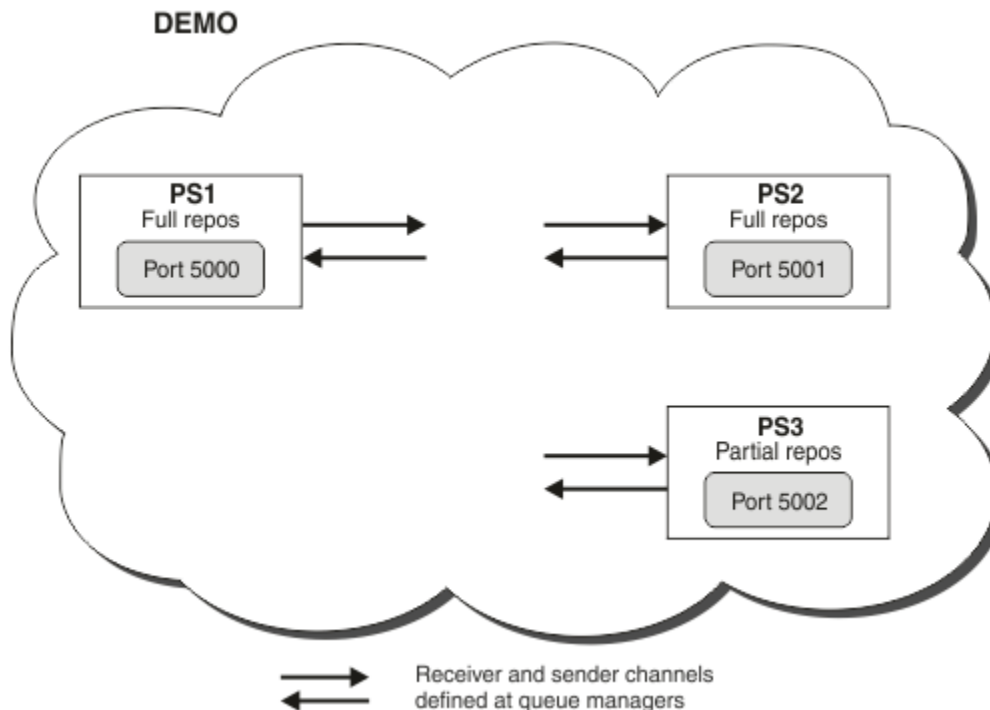
In diesem Szenario erstellen Sie ein einfaches Cluster mit drei Warteschlangenmanagern und konfigurieren es so, dass die auf einem Warteschlangenmanager erstellten Subskriptionen Nachrichten empfangen können, die von einer mit einem anderen Warteschlangenmanager verbundenen Anwendung veröffentlicht wurden.

Vorbereitende Schritte

Der Ausgangspunkt für dieses Szenario ist eine bereits vorhandene IBM MQ-Installation. Zur Installation von IBM MQ folgen Sie den Schritten unter [IBM MQ-Server unter Windows installieren](#).

Informationen zu diesem Vorgang

Wenn Sie die Schritte in diesem Szenario ausführen, erstellen Sie zuerst den folgenden Cluster:



Dieser Cluster besteht aus drei Warteschlangenmanagern, von denen zwei als Warteschlangenmanager mit vollständigen Repositories definiert sind.

Anschließend definieren Sie ein Cluster-Topic auf Warteschlangenmanager PS3. Durch Erstellen des Cluster-Topics haben Sie aus dem Cluster einen Publish/Subscribe-Cluster gemacht. Zum Testen des Publish/Subscribe-Clusters abonnieren Sie das Thema auf einem beliebigen Warteschlangenmanager und veröffentlichen dann eine Nachricht zu dem Thema auf einem anderen Warteschlangenmanager. Anschließend prüfen Sie, ob Ihre Subskription die Nachricht erhält.

Zugehörige Tasks

[Publish/Subscribe-Cluster entwerfen](#)

[WS-Manager-Cluster konfigurieren](#)

Warteschlangenmanager erstellen und starten

Erstellen und starten Sie die drei Warteschlangenmanager PS1, PS2 und PS3.

Vorgehensweise

1. Erstellen und starten Sie Warteschlangenmanager PS1.

a) Erstellen Sie den Warteschlangenmanager.

Geben Sie in der Befehlszeile den folgenden Befehl ein:

```
crtmqm PS1
```

b) Starten Sie den Warteschlangenmanager.

Geben Sie in der Befehlszeile den folgenden Befehl ein:

```
strmqm PS1
```

2. Wiederholen Sie Schritt 1 zur Erstellung von Warteschlangenmanager PS2.

3. Wiederholen Sie Schritt 1 zur Erstellung von Warteschlangenmanager PS3.

Nächste Schritte

Sie können nun den [ersten Warteschlangenmanager konfigurieren](#).

Ersten Warteschlangenmanager konfigurieren

Definieren Sie in der MQSC-Schnittstelle ein Empfangsprogramm und einen Empfängerkanal für PS1, richten Sie den Warteschlangenmanager als vollständiges Repository für den Cluster ein und definieren Sie einen Senderkanal von PS1 zu PS2, über den die beiden vollständigen Repositories Informationen austauschen können.

Vorbereitende Schritte

Diese Task setzt voraus, dass die im Abschnitt [„Warteschlangenmanager erstellen und starten“](#) auf Seite 40 beschriebenen Schritte ausgeführt wurden.

Vorgehensweise

1. Definieren und starten Sie ein Empfangsprogramm für PS1.

a) Starten Sie die MQSC-Schnittstelle.

Geben Sie in der Befehlszeile den folgenden Befehl ein:

```
runmqsc PS1
```

b) Definieren Sie ein Empfangsprogramm.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
DEFINE LISTENER(PS1_LS) TRPTYPE(TCP) CONTROL(QMGR) PORT(5000)
```

c) Starten Sie den Listener.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:


```
START LISTENER(PS1_LS)
```

2. Legen Sie den Warteschlangenmanager als vollständiges Repository für den Cluster fest.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
ALTER QMGR REPOS(DEMO)
```

3. Definieren Sie einen Empfängerkanal für PS1, damit andere Warteschlangenmanager des Clusters mit ihm kommunizieren können.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
DEFINE CHANNEL(DEMO.PS1) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CONNAME('$HOSTNAME(5000)') CLUS  
TER(DEMO)  
DESCR('TCP Cluster-receiver channel for queue manager PS1')
```

4. Definieren Sie einen Senderkanal von PS1 zu PS2, damit die beiden vollständigen Repositories Informationen austauschen können.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
DEFINE CHANNEL(DEMO.PS2) CHLTYPE(CLUSSDR) TRPTYPE(TCP) CONNAME('$HOSTNAME(5001)') CLUSTER(DE  
MO)  
DESCR('TCP Cluster-sender channel from PS1 to queue manager PS2')
```

Nächste Schritte

Sie können nun den [zweiten Warteschlangenmanager konfigurieren](#).

Zweiten Warteschlangenmanager konfigurieren

Definieren Sie in der MQSC-Schnittstelle ein Empfangsprogramm und einen Empfängerkanal für PS2, richten Sie den Warteschlangenmanager als vollständiges Repository für den Cluster ein und definieren Sie einen Senderkanal von PS2 zu PS1, über den die beiden vollständigen Repositories Informationen austauschen können.

Vorbereitende Schritte

Diese Task setzt voraus, dass die im Abschnitt [„Ersten Warteschlangenmanager konfigurieren“](#) auf Seite 40 beschriebenen Schritte ausgeführt wurden.

Vorgehensweise

1. Definieren und starten Sie ein Empfangsprogramm für PS2.

- a) Starten Sie die MQSC-Schnittstelle.

Geben Sie in der Befehlszeile den folgenden Befehl ein:

```
runmqsc PS2
```

- b) Definieren Sie ein Empfangsprogramm.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
DEFINE LISTENER(PS2_LS) TRPTYPE(TCP) CONTROL(QMGR) PORT(5001)
```

- c) Starten Sie den Listener.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
START LISTENER(PS2_LS)
```

2. Legen Sie den Warteschlangenmanager als vollständiges Repository für den Cluster fest.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
ALTER QMGR REPOS(DEMO)
```

3. Definieren Sie einen Empfängerkanal für PS2, damit andere Warteschlangenmanager des Clusters mit ihm kommunizieren können.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
DEFINE CHANNEL(DEMO.PS2) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CONNAME('$HOSTNAME(5001)') CLUSTER(DEMO)
DESCR('TCP Cluster-receiver channel for queue manager PS2')
```

4. Definieren Sie einen Senderkanal von PS2 zu PS1, damit die beiden vollständigen Repositorys Informationen austauschen können.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
DEFINE CHANNEL(DEMO.PS1) CHLTYPE(CLUSSDR) TRPTYPE(TCP) CONNAME('$HOSTNAME(5000)') CLUSTER(DEMO)
DESCR('TCP Cluster-sender channel from PS2 to PS1')
```

Nächste Schritte

Sie können nun den [dritten Warteschlangenmanager konfigurieren](#).

Dritten Warteschlangenmanager konfigurieren

Definieren Sie in der MQSC-Schnittstelle ein Empfangsprogramm und einen Empfängerkanal für PS3. Fügen Sie PS3 dem Cluster hinzu, indem Sie von PS3 einen Senderkanal zu einem der Warteschlangenmanager mit vollständigem Repository definieren.

Vorbereitende Schritte

Diese Task setzt voraus, dass die im Abschnitt [„Zweiten Warteschlangenmanager konfigurieren“](#) auf Seite 41 beschriebenen Schritte ausgeführt wurden.

Vorgehensweise

1. Definieren und starten Sie ein Empfangsprogramm für PS3.

- a) Starten Sie die MQSC-Schnittstelle.

Geben Sie in der Befehlszeile den folgenden Befehl ein:

```
runmqsc PS3
```

- b) Definieren Sie ein Empfangsprogramm.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
DEFINE LISTENER(PS3_LS) TRPTYPE(TCP) CONTROL(QMGR) PORT(5002)
```

- c) Starten Sie den Listener.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
START LISTENER(PS3_LS)
```

2. Definieren Sie einen Empfängerkanal für PS3, damit andere Warteschlangenmanager des Clusters mit ihm kommunizieren können.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
DEFINE CHANNEL(DEMO.PS3) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CONNAME('$HOSTNAME(5002)') CLUS  
TER(DEMO)  
DESCR('TCP Cluster-receiver channel for queue manager PS3')
```

3. Definieren Sie einen Senderkanal von PS3 zu einem der Warteschlangenmanager mit vollständigem Repository (z. B. PS1). Dadurch wird PS3 in den Cluster aufgenommen.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
DEFINE CHANNEL(DEMO.PS1) CHLTYPE(CLUSSDR) TRPTYPE(TCP) CONNAME('$HOSTNAME(5000)') CLUSTER(DE  
MO)  
DESCR('TCP Cluster-sender channel from PS3 to PS1')
```

4. Prüfen Sie, ob PS3 erfolgreich dem Cluster hinzugefügt wurde.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
DISPLAY CLUSQMGR(*) QMTYPE
```

Dieser Befehl gibt drei Einträge zurück, jeweils einen für QM1, QM2 und QM3. QM1 und QM2 sollten den **QMTYPE** REPOS haben, QM3 hingegen den **QMTYPE** NORMAL.

Nächste Schritte

Sie können nun ein [Cluster-Topic definieren](#).

Clusterthemen definieren

Publish/Subscribe-Anwendungen können für Veröffentlichungen jede Topic-Zeichenfolge verwenden, ohne dass ein entsprechendes verwaltetes Themenobjekt definiert sein muss. Sind die veröffentlichenden Anwendungen jedoch nicht mit dem Cluster-Warteschlangenmanager verbunden, auf denen die Subskriptionen erstellt wurden, muss ein verwaltetes Themenobjekt definiert und dem Cluster hinzugefügt werden. Um aus einem Thema ein Cluster-Topic zu machen, geben Sie in seiner Definition den Namen des Clusters ein.

Vorbereitende Schritte

Diese Task setzt voraus, dass die im Abschnitt [„Dritten Warteschlangenmanager konfigurieren“](#) auf Seite 42 beschriebenen Schritte ausgeführt wurden.

Informationen zu diesem Vorgang

Das verwaltete Themenobjekt gibt den Punkt in der Themenstruktur an, der durch seine Topic-Zeichenfolge geclustert ist. Publish/Subscribe-Anwendungen können an oder unterhalb dieses Punkts jede Topic-Zeichenfolge verwenden, damit ihre Nachrichten automatisch zwischen den Warteschlangenmanagern übertragen werden.

Bei der Definition eines Cluster-Topics wählen Sie auch sein Routing-Modell. Weitere Informationen zum Routing von Veröffentlichungen in Clustern finden Sie im Abschnitt [Publish/Subscribe-Cluster entwerfen](#).

In diesem Szenario wird die Standard-Routing-Methode *DIRECT* verwendet. Das bedeutet, dass eine Nachricht direkt vom veröffentlichenden Warteschlangenmanager an den abonnierenden Warteschlangenmanager gesendet wird.

Vorgehensweise

1. Definieren Sie das Cluster-Topic SCORES auf PS3.

Um ein Topic zu einem Cluster-Topic zu machen, geben Sie den Namen des Clusters an und legen Sie das Cluster-Routing (**CLRROUTE**) fest, das für Veröffentlichungen und Subskriptionen für dieses Topic verwendet werden soll.

- a) Starten Sie die MQSC-Schnittstelle.

Geben Sie in der Befehlszeile den folgenden Befehl ein:

```
runmqsc PS3
```

- b) Definieren Sie das Cluster-Topic SCORES.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
DEFINE TOPIC(SCORES) TOPICSTR('/Sport/Scores') CLUSTER(DEMO) CLRROUTE(DIRECT)
```

- c) Geben Sie end ein, um die MQSC-Schnittstelle auf PS3 zu beenden.

2. Prüfen Sie die Themendefinition auf PS1.

- a) Starten Sie die MQSC-Schnittstelle auf PS1.

Geben Sie in der Befehlszeile den folgenden Befehl ein:

```
runmqsc PS1
```

- b) Zeigen Sie den Clusterstatus für das Cluster-Topic SCORES an.

Geben Sie folgenden WebSphere MQ-Scriptbefehl ein:

```
DISPLAY TCLUSTER(SCORES) CLSTATE
```

Der **CLSTATE**-Status des Cluster-Topics SCORES zeigt ACTIVE an.

Nächste Schritte

Nähere Ausführungen zu dieser Task finden Sie im Abschnitt [Publish/Subscribe-Cluster konfigurieren](#).

Sie können die Lösung nun prüfen. Weitere Informationen finden Sie unter [„Publish/Subscribe-Cluster testen“](#) auf Seite 44.

Publish/Subscribe-Cluster testen

Testen Sie den Publish/Subscribe-Cluster durch die Veröffentlichung und Subskription einer Themenzeichenfolge von verschiedenen Warteschlangenmanagern des Clusters.

Vorbereitende Schritte

Diese Task setzt voraus, dass die im Abschnitt [„Clusterthemen definieren“](#) auf Seite 43 beschriebenen Schritte ausgeführt wurden.

Informationen zu diesem Vorgang

Mithilfe der Befehlszeile und der Beispielanwendungen amqspub und amqssub, die in IBM MQ enthalten sind, können Sie ein Thema aus einem Warteschlangenmanager veröffentlichen und das Thema mit den anderen Warteschlangenmanagern subskribieren. Sobald eine Nachricht für das Thema veröffentlicht wird, wird sie von den abonnierenden Warteschlangenmanagern empfangen.

Vorgehensweise

1. Geben Sie in der Befehlszeile den folgenden Befehl ein:

```
amqspub /Sport/Scores/Football PS1
```

2. Geben Sie gleichzeitig in jeweils eigenen Befehlszeilen die folgenden Befehle ein:

```
amqssub /Sport/Scores/Football PS2
```

```
amqssub /Sport/Scores/Football PS3
```

3. Geben Sie in der ersten Befehlszeile eine Nachricht ein.

Die Nachricht wird in beiden abonnierenden Befehlszeilen angezeigt.

Anmerkung: Wenn eine Veröffentlichung nicht innerhalb von zehn Sekunden eingeht, wird die Anwendung amqssub aufgrund der Überschreitung des Zeitlimits geschlossen.

Ergebnisse

Der Publish/Subscribe-Cluster ist eingerichtet.

Nächste Schritte

Versuchen Sie auch, andere Themenobjekte für andere Zweige der Themenstruktur mit anderen Routing-Modellen zu definieren.

Publish/Subscribe-Hierarchieszenarios

Dies sind drei Szenarios, die die Verwendung von Publish/Subscribe-Hierarchien veranschaulichen. In allen drei Szenarios wird die gleiche einfache Publish/Subscribe-Topologie eingerichtet. In jedem Szenario verwenden die Warteschlangenmanager eine andere Methode für die Verbindung mit den benachbarten Warteschlangenmanagern der Hierarchie.

Die verfügbaren Publish/Subscribe-Hierarchieszenarios werden in folgenden Unterabschnitten beschrieben:

Zugehörige Konzepte

[Publish/Subscribe-Hierarchien](#)

Szenario 1 zu Publish/Subscribe-Hierarchie: Punkt-zu-Punkt-Kanäle mit einem Warteschlangenmanager-Aliasnamen verwenden

Dies ist das erste von drei Szenarios, in denen auf unterschiedliche Arten eine Publish/Subscribe-Hierarchie zur Herstellung der Verbindung zwischen Warteschlangenmanagern aufgebaut wird. In diesem Szenario wird eine Publish/Subscribe-Hierarchie eingerichtet, die Punkt-zu-Punkt-Kanäle mit einem Warteschlangenmanager-Aliasnamen verwendet.

Informationen zu diesem Vorgang

Diese Gruppe von Szenarios verwendet alle einen übergeordneten Warteschlangenmanager namens QM1 und zwei untergeordnete Warteschlangenmanager namens QM2 und QM3.

Szenario 1 ist in kleinere Abschnitte aufgeteilt, um dem Prozess einfacher folgen zu können.

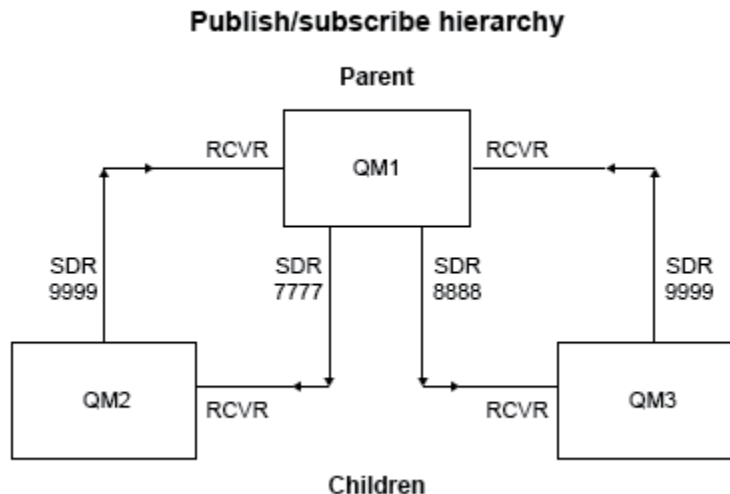


Abbildung 3. Das Topologiediagramm zeigt die Beziehung zwischen Warteschlangenmanagern in einer typischen Publish/Subscribe-Hierarchie.

Vorgehensweise

1. Erstellen Sie die Warteschlangenmanager.

- a) Erstellen und starten Sie mit folgenden Befehlen die drei Warteschlangenmanager QM1, QM2 und QM3:

```

crtmqm -u SYSTEM.DEAD.LETTER.QUEUE QM1
stimqm QM1

crtmqm -u SYSTEM.DEAD.LETTER.QUEUE QM2
stimqm QM2

crtmqm -u SYSTEM.DEAD.LETTER.QUEUE QM3
stimqm QM3
  
```

- b) Aktivieren Sie den Warteschlangenmanager-Publish/Subscribe-Modus, indem Sie folgenden Befehl auf alle drei Warteschlangenmanager anwenden:

```
ALTER QMGR PSMODE(ENABLED)
```

2. Stellen Sie Punkt-zu-Punkt-Kanalverbindungen zwischen Warteschlangenmanagern her, indem Sie einen Warteschlangenmanager-Alias mit demselben Namen wie der übergeordnete Warteschlangenmanager verwenden.

- a) Definieren Sie eine Übertragungswarteschlange und einen Warteschlangenmanager-Aliasnamen auf QM2 zu QM1. Definieren Sie einen Senderkanal zu QM1 und einen Empfängerkanal für den auf QM1 für QM2 erstellten Senderkanal:

```

DEFINE QLOCAL(QM1.XMITQ) USAGE(XMITQ)

DEFINE QREMOTE (QM1) RNAME('') RQMNAME(QM1) XMITQ(QM1.XMITQ)

DEFINE CHANNEL('QM2.TO.QM1') CHLTYPE(SDR) CONNAME('localhost(9999)') XMITQ(QM1.XMITQ)
TRPTYPE(TCP)

DEFINE CHANNEL('QM1.TO.QM2') CHLTYPE(RCVR) TRPTYPE(TCP)
  
```

- b) Definieren Sie eine Übertragungswarteschlange und einen Warteschlangenmanager-Aliasnamen auf QM3 zu QM1. Definieren Sie einen Senderkanal zu QM1 und einen Empfängerkanal für den auf QM1 für QM3 erstellten Senderkanal:

```
DEFINE QLOCAL(QM1.XMITQ) USAGE(XMITQ)
DEFINE QREMOTE (QM1) RNAME('') RQMNAME(QM1) XMITQ(QM1.XMITQ)
DEFINE CHANNEL('QM3.TO.QM1') CHLTYPE(SDR) CONNAME('localhost(9999)') XMITQ(QM1.XMITQ)
TRPTYPE(TCP)
DEFINE CHANNEL('QM1.TO.QM3') CHLTYPE(RCVR) TRPTYPE(TCP)
```

- c) Definieren Sie eine Übertragungswarteschlange und einen Warteschlangenmanager-Aliasnamen auf QM1 zu QM2 und QM3. Definieren Sie einen Senderkanal zu QM2 und QM3 sowie einen Empfängerkanal für die auf QM2 und QM3 für QM1 erstellten Senderkanäle:

```
DEFINE QLOCAL(QM2.XMITQ) USAGE(XMITQ)
DEFINE QREMOTE (QM2) RNAME('') RQMNAME(QM2) XMITQ(QM2.XMITQ)
DEFINE CHANNEL('QM1.TO.QM2') CHLTYPE(SDR) CONNAME('localhost(7777)') XMITQ(QM2.XMITQ)
TRPTYPE(TCP)
DEFINE CHANNEL('QM2.TO.QM1') CHLTYPE(RCVR) TRPTYPE(TCP)
DEFINE QLOCAL(QM3.XMITQ) USAGE(XMITQ)
DEFINE QREMOTE (QM3) RNAME('') RQMNAME(QM3) XMITQ(QM3.XMITQ)
DEFINE CHANNEL('QM1.TO.QM3') CHLTYPE(SDR) CONNAME('localhost(8888)') XMITQ(QM3.XMITQ)
TRPTYPE(TCP)
DEFINE CHANNEL('QM3.TO.QM1') CHLTYPE(RCVR) TRPTYPE(TCP)
```

- d) Starten Sie die entsprechenden Empfangsprogramme auf den Warteschlangenmanagern:

```
runmqclsr -m QM1 -t TCP -p 9999 &
runmqclsr -m QM2 -t TCP -p 7777 &
runmqclsr -m QM3 -t TCP -p 8888 &
```

- e) Starten Sie folgende Kanäle:

- i) Auf QM1:

```
START CHANNEL('QM1.TO.QM2')
START CHANNEL('QM1.TO.QM3')
```

- ii) Auf QM2:

```
START CHANNEL('QM2.TO.QM1')
```

- iii) Auf QM3:

```
START CHANNEL('QM3.TO.QM1')
```

- f) Überprüfen Sie, ob alle Kanäle gestartet wurden:

```
DISPLAY CHSTATUS('QM1.TO.QM2')
DISPLAY CHSTATUS('QM1.TO.QM3')
DISPLAY CHSTATUS('QM2.TO.QM1')
DISPLAY CHSTATUS('QM3.TO.QM1')
```

g)

3. Verbinden Sie die Warteschlangenmanager und definieren Sie ein Thema.

Verbinden Sie die untergeordneten Warteschlangenmanager QM2 und QM3 mit dem übergeordneten Warteschlangenmanager QM1.

a) Setzen Sie auf QM2 und QM3 den übergeordneten Warteschlangenmanager auf QM1:

```
ALTER QMGR PARENT (QM1)
```

b) Führen Sie folgenden Befehl für alle Warteschlangenmanager aus, um zu überprüfen, ob die untergeordneten Warteschlangenmanager mit dem übergeordneten Warteschlangenmanager verbunden sind:

```
DISPLAY PUBSUB TYPE(ALL)
```

Die Befehlsausgabe wird angezeigt. Es folgt ein Beispiel für die Ausgabe von QM1, in der die wichtigsten Details hervorgehoben sind:

```
DISPLAY PUBSUB ALL
1 : DISPLAY PUBSUB ALL
AMQ8723: Display pub/sub status details.
QMNAME(QM1)           TYPE(LOCAL)
STATUS(ACTIVE)        SUBCOUNT(6)
TPCOUNT(9)
AMQ8723: Display pub/sub status details.
QMNAME(QM2) TYPE(CHILD)
STATUS(ACTIVE) SUBCOUNT(NONE)
TPCOUNT(NONE)
AMQ8723: Display pub/sub status details.
QMNAME(QM3) TYPE(CHILD)
STATUS(ACTIVE) SUBCOUNT(NONE)
TPCOUNT(NONE)
```

4. Verwenden Sie die Anwendungen amqspub.exe und amqssub.exe, um das Thema zu veröffentlichen und zu subscribieren.

a) Führen Sie im ersten Befehlsfenster folgenden Befehl aus:

```
amqspub Sport/Soccer QM2
```

b) Führen Sie im zweiten Befehlsfenster folgenden Befehl aus:

```
amqssub Sport/Soccer QM1
```

c) Führen Sie im dritten Befehlsfenster folgenden Befehl aus:

```
amqssub Sport/Soccer QM3
```

Ergebnisse

Die amqssub.exe -Anwendungen im zweiten und dritten Befehlsfenster empfangen die im ersten Befehlsfenster veröffentlichten Nachrichten

Szenario 2 zu Publish/Subscribe-Hierarchie: Punkt-zu-Punkt-Kanäle verwenden, wobei der Name der Übertragungswarteschlange mit dem des fernen Warteschlangenmanagers identisch ist

Dies ist das zweite von drei Szenarios, in denen auf unterschiedliche Arten eine Publish/Subscribe-Hierarchie zur Herstellung der Verbindung zwischen Warteschlangenmanagern aufgebaut wird. In diesem Szenario wird eine Publish/Subscribe-Hierarchie eingerichtet, die Punkt-zu-Punkt-Kanäle verwendet, wo-

bei der Name der Übertragungswarteschlange mit dem des fernen Warteschlangenmanagers identisch ist.

Informationen zu diesem Vorgang

Diese Gruppe von Szenarios verwendet alle einen übergeordneten Warteschlangenmanager namens QM1 und zwei untergeordnete Warteschlangenmanager namens QM2 und QM3.

In diesem Szenario werden die Schritte 1, 3 und 4 aus „Szenario 1 zu Publish/Subscribe-Hierarchie: Punkt-zu-Punkt-Kanäle mit einem Warteschlangenmanager-Aliasnamen verwenden“ auf Seite 45 wieder verwendet.

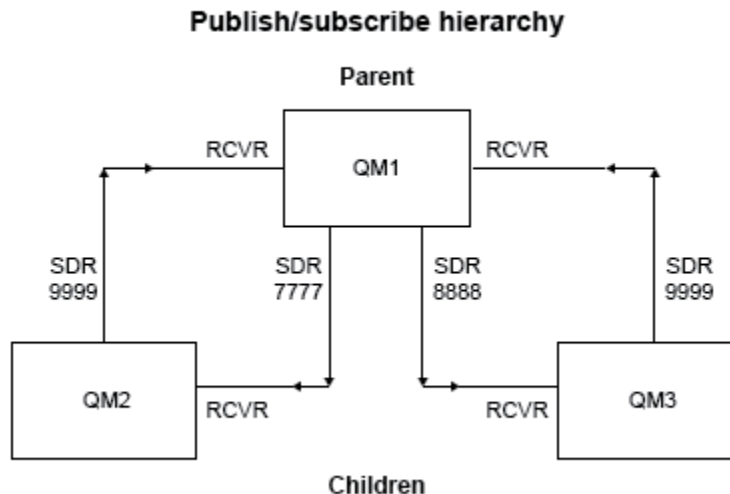


Abbildung 4. Das Topologiediagramm zeigt die Beziehung zwischen Warteschlangenmanagern in einer typischen Publish/Subscribe-Hierarchie.

Vorgehensweise

1. Erstellen Sie die Warteschlangenmanager.

- a) Erstellen und starten Sie mit folgenden Befehlen die drei Warteschlangenmanager QM1, QM2 und QM3:

```
crtmqm -u SYSTEM.DEAD.LETTER.QUEUE QM1
strmqm QM1

crtmqm -u SYSTEM.DEAD.LETTER.QUEUE QM2
strmqm QM2

crtmqm -u SYSTEM.DEAD.LETTER.QUEUE QM3
strmqm QM3
```

- b) Aktivieren Sie den Warteschlangenmanager-Publish/Subscribe-Modus, indem Sie folgenden Befehl auf alle drei Warteschlangenmanager anwenden:

```
ALTER QMGR PSMODE(ENABLED)
```

2. Stellen Sie Punkt-zu-Punkt-Kanalverbindungen zwischen Warteschlangenmanagern über eine Übertragungswarteschlange mit demselben Namen wie der übergeordnete Warteschlangenmanager her.

- a) Definieren Sie eine Übertragungswarteschlange auf QM2 zu QM1. Definieren Sie einen Senderkanal zu QM1 und einen Empfängerkanal für den Senderkanal für QM2, der auf QM1 erstellt wurde:

```

DEFINE QLOCAL(QM1) USAGE(XMITQ)

DEFINE CHANNEL('QM2.TO.QM1') CHLTYPE(SDR) CONNAME('localhost(9999)') XMITQ(QM1)
TRPTYPE(TCP)

DEFINE CHANNEL('QM1.TO.QM2') CHLTYPE(RCVR) TRPTYPE(TCP)

```

- b) Definieren Sie eine Übertragungswarteschlange auf QM3 zu QM1. Definieren Sie einen Senderkanal zu QM1 und einen Empfängerkanal für den auf QM1 für QM3 erstellten Senderkanal:

```

DEFINE QLOCAL(QM1) USAGE(XMITQ)

DEFINE CHANNEL('QM3.TO.QM1') CHLTYPE(SDR) CONNAME('localhost(9999)') XMITQ(QM1)
TRPTYPE(TCP)

DEFINE CHANNEL('QM1.TO.QM3') CHLTYPE(RCVR) TRPTYPE(TCP)

```

- c) Definieren Sie Übertragungswarteschlangen auf QM1 zu QM2 und QM3. Definieren Sie Senderkanäle zu QM2 und QM3 sowie einen Empfängerkanal für die auf QM2 und QM3 für QM1 erstellten Senderkanäle:

```

DEFINE QLOCAL(QM2) USAGE(XMITQ)

DEFINE CHANNEL('QM1.TO.QM2') CHLTYPE(SDR) CONNAME('localhost(7777)') XMITQ(QM2)
TRPTYPE(TCP)

DEFINE CHANNEL('QM2.TO.QM1') CHLTYPE(RCVR) TRPTYPE(TCP)

DEFINE QLOCAL(QM3) USAGE(XMITQ)

DEFINE CHANNEL('QM1.TO.QM3') CHLTYPE(SDR) CONNAME('localhost(8888)') XMITQ(QM3)
TRPTYPE(TCP)

DEFINE CHANNEL('QM3.TO.QM1') CHLTYPE(RCVR) TRPTYPE(TCP)

```

- d) Starten Sie die entsprechenden Empfangsprogramme auf den Warteschlangenmanagern:

```

runmqclsr -m QM1 -t TCP -p 9999 &
runmqclsr -m QM2 -t TCP -p 7777 &
runmqclsr -m QM3 -t TCP -p 8888 &

```

- e) Starten Sie folgende Kanäle:

- i) Auf QM1:

```

START CHANNEL('QM1.TO.QM2')

START CHANNEL('QM1.TO.QM3')

```

- ii) Auf QM2:

```

START CHANNEL('QM2.TO.QM1')

```

- iii) Auf QM3:

```

START CHANNEL('QM3.TO.QM1')

```

- f) Überprüfen Sie, ob alle Kanäle gestartet wurden:

```

DISPLAY CHSTATUS('QM1.TO.QM2')

DISPLAY CHSTATUS('QM1.TO.QM3')

DISPLAY CHSTATUS('QM2.TO.QM1')

```

```
DISPLAY CHSTATUS('QM3.TO.QM1')
```

3. Verbinden Sie die Warteschlangenmanager und definieren Sie ein Thema.

Verbinden Sie die untergeordneten Warteschlangenmanager QM2 und QM3 mit dem übergeordneten Warteschlangenmanager QM1.

- a) Setzen Sie auf QM2 und QM3 den übergeordneten Warteschlangenmanager auf QM1:

```
ALTER QMGR PARENT (QM1)
```

- b) Führen Sie folgenden Befehl für alle Warteschlangenmanager aus, um zu überprüfen, ob die untergeordneten Warteschlangenmanager mit dem übergeordneten Warteschlangenmanager verbunden sind:

```
DISPLAY PUBSUB TYPE(ALL)
```

Die Befehlsausgabe wird angezeigt. Es folgt ein Beispiel für die Ausgabe von QM1, in der die wichtigsten Details hervorgehoben sind:

```
DISPLAY PUBSUB ALL
1 : DISPLAY PUBSUB ALL
AMQ8723: Display pub/sub status details.
QMNAME(QM1) TYPE(LOCAL)
STATUS(ACTIVE) SUBCOUNT(6)
TPCOUNT(9)
AMQ8723: Display pub/sub status details.
QMNAME(QM2) TYPE(CHILD)
STATUS(ACTIVE) SUBCOUNT(NONE)
TPCOUNT(NONE)
AMQ8723: Display pub/sub status details.
QMNAME(QM3) TYPE(CHILD)
STATUS(ACTIVE) SUBCOUNT(NONE)
TPCOUNT(NONE)
```

4. Verwenden Sie die Anwendungen amqspub.exe und amqssub.exe, um das Thema zu veröffentlichen und zu abonnieren.

- a) Führen Sie im ersten Befehlsfenster folgenden Befehl aus:

```
amqspub Sport/Soccer QM2
```

- b) Führen Sie im zweiten Befehlsfenster folgenden Befehl aus:

```
amqssub Sport/Soccer QM1
```

- c) Führen Sie im dritten Befehlsfenster folgenden Befehl aus:

```
amqssub Sport/Soccer QM3
```

Ergebnisse

Die amqssub.exe -Anwendungen im zweiten und dritten Befehlsfenster empfangen die im ersten Befehlsfenster veröffentlichten Nachrichten

Zugehörige Tasks

„Szenario 1 zu Publish/Subscribe-Hierarchie: Punkt-zu-Punkt-Kanäle mit einem Warteschlangenmanager-Aliasnamen verwenden“ auf Seite 45

Dies ist das erste von drei Szenarios, in denen auf unterschiedliche Arten eine Publish/Subscribe-Hierarchie zur Herstellung der Verbindung zwischen Warteschlangenmanagern aufgebaut wird. In diesem Szenario wird eine Publish/Subscribe-Hierarchie eingerichtet, die Punkt-zu-Punkt-Kanäle mit einem Warteschlangenmanager-Aliasnamen verwendet.

„Szenario 3 zu Publish/Subscribe-Hierarchie: Clusterkanal zum Hinzufügen eines Warteschlangenmanagers verwenden“ auf Seite 52

Dies ist das dritte von drei Szenarios, in denen auf unterschiedliche Arten eine Publish/Subscribe-Hierarchie zur Herstellung der Verbindung zwischen Warteschlangenmanagern aufgebaut wird. In diesem Szenario wird ein Clusterkanal zum Hinzufügen eines Warteschlangenmanagers zu einer Hierarchie verwendet.

WS-Manager mit einer Publish/Subscribe-Hierarchie verbinden

Szenario 3 zu Publish/Subscribe-Hierarchie: Clusterkanal zum Hinzufügen eines Warteschlangenmanagers verwenden

Dies ist das dritte von drei Szenarios, in denen auf unterschiedliche Arten eine Publish/Subscribe-Hierarchie zur Herstellung der Verbindung zwischen Warteschlangenmanagern aufgebaut wird. In diesem Szenario wird ein Clusterkanal zum Hinzufügen eines Warteschlangenmanagers zu einer Hierarchie verwendet.

Informationen zu diesem Vorgang

Diese Gruppe von Szenarios verwendet alle einen übergeordneten Warteschlangenmanager namens QM1 und zwei untergeordnete Warteschlangenmanager namens QM2 und QM3.

Anmerkung: In diesem Szenario wird die Clusterkonfiguration nur zur Verbindung der Warteschlangenmanager verwendet, nicht zur Übertragung von Publish/Subscribe-Daten durch das Clustering von Themen. Wenn Sie innerhalb desselben Clusters Hierarchiebeziehungen zwischen Warteschlangenmanagern einrichten, erfolgt die Weiterleitung von Veröffentlichungen zwischen den Warteschlangenmanagern auf Basis der Veröffentlichungs- und Subskriptionsbereichseinstellungen der Themen in der Themenstruktur. Zum Hinzufügen von Themen in den Cluster darf daher keinesfalls die Clusternamenseinstellung des Themas verwendet werden. Sobald Sie den Clusternamen verwenden, ändert sich die Topologie in einen Publish/Subscribe-Cluster, für den keine Hierarchiebeziehungen definiert werden müssen. Weitere Informationen hierzu finden Sie in den Abschnitten „Szenario: Publish/Subscribe-Cluster erstellen“ auf Seite 39 und Netz mit verteiltem Publish/Subscribe planen.

In diesem Szenario werden die Schritte 1, 3 und 4 aus „Szenario 1 zu Publish/Subscribe-Hierarchie: Punkt-zu-Punkt-Kanäle mit einem Warteschlangenmanager-Aliasnamen verwenden“ auf Seite 45 wieder verwendet.

In diesem Szenario wird ein Cluster mit dem Namen DEMO erstellt, in dem QM1 und QM2 vollständige Repositories sind und QM3 ein Teilrepository ist. Warteschlangenmanager QM1 ist der übergeordnete Warteschlangenmanager der Warteschlangenmanager QM2 und QM3.

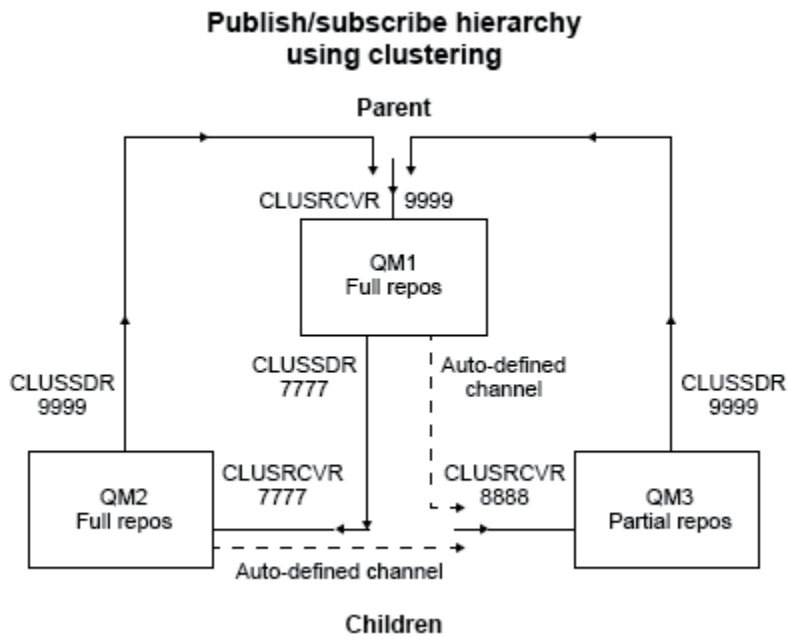


Abbildung 5. Das Topologiediagramm zeigt die Beziehung zwischen Warteschlangenmanagern, die einen Clusterkanal verwenden.

Vorgehensweise

1. Erstellen Sie die Warteschlangenmanager.

a) Erstellen und starten Sie mit folgenden Befehlen die drei Warteschlangenmanager QM1, QM2 und QM3:

```

ctmqm -u SYSTEM.DEAD.LETTER.QUEUE QM1
strmqm QM1

ctmqm -u SYSTEM.DEAD.LETTER.QUEUE QM2
strmqm QM2

ctmqm -u SYSTEM.DEAD.LETTER.QUEUE QM3
strmqm QM3

```

b) Aktivieren Sie den Warteschlangenmanager-Publish/Subscribe-Modus, indem Sie folgenden Befehl auf alle drei Warteschlangenmanager anwenden:

```
ALTER QMGR PSMODE(ENABLED)
```

2. Stellen Sie die Punkt-zu-Punkt-Kanalverbindungen zwischen den Warteschlangenmanagern in einem Cluster her.

a) Setzen Sie auf QM1 und QM2 den Parameter **REPOS** auf den Namen des Clusters DEMO:

```
ALTER QMGR REPOS(DEMO)
```

b) Starten Sie die entsprechenden Empfangsprogramme auf den Warteschlangenmanagern:

```

iunmq1sr -m QM1 -t TCP -p 9999 &
iunmq1sr -m QM2 -t TCP -p 7777 &
iunmq1sr -m QM3 -t TCP -p 8888 &

```

c) Definieren Sie auf jedem Warteschlangenmanager den Clusterempfängerkanal:

i) Auf QM1:

```
DEFINE CHANNEL(TO.QM1) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CONNAME('localhost(9999)') CLUS□  
TER(DEMO)
```

ii) Auf QM2:

```
DEFINE CHANNEL(TO.QM2) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CONNAME('localhost(7777)') CLUS□  
TER(DEMO)
```

iii) Auf QM3:

```
DEFINE CHANNEL(TO.QM3) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CONNAME('localhost(8888)') CLUS□  
TER(DEMO)
```

d) Definieren Sie auf jedem Warteschlangenmanager im Cluster einen Clustersenderkanal zu einem vollständigen Repository:

i) Auf QM1:

```
DEFINE CHANNEL(TO.QM2) CHLTYPE(CLUSSDR) TRPTYPE(TCP) CONNAME('localhost(7777)') CLUS□  
TER(DEMO)
```

ii) Auf QM2:

```
DEFINE CHANNEL(TO.QM1) CHLTYPE(CLUSSDR) TRPTYPE(TCP) CONNAME('localhost(9999)') CLUS□  
TER(DEMO)
```

iii) QM3 kann über einen Clustersenderkanal entweder zum vollständigen Repository auf QM1 oder auf QM2 verfügen. In diesem Beispiel wird der Kanal zu QM1 definiert:

```
DEFINE CHANNEL(TO.QM1) CHLTYPE(CLUSSDR) TRPTYPE(TCP) CONNAME('localhost(9999)') CLUS□  
TER(DEMO)
```

3. Verbinden Sie die Warteschlangenmanager und definieren Sie ein Thema.

Verbinden Sie die untergeordneten Warteschlangenmanager QM2 und QM3 mit dem übergeordneten Warteschlangenmanager QM1.

a) Setzen Sie auf QM2 und QM3 den übergeordneten Warteschlangenmanager auf QM1:

```
ALTER QMGR PARENT (QM1)
```

b) Führen Sie folgenden Befehl für alle Warteschlangenmanager aus, um zu überprüfen, ob die untergeordneten Warteschlangenmanager mit dem übergeordneten Warteschlangenmanager verbunden sind:

```
DISPLAY PUBSUB TYPE(ALL)
```

Die Befehlsausgabe wird angezeigt. Es folgt ein Beispiel für die Ausgabe von QM1, in der die wichtigsten Details hervorgehoben sind:

```
DISPLAY PUBSUB ALL  
1 : DISPLAY PUBSUB ALL  
AMQ8723: Display pub/sub status details.  
QMNAME(QM1) TYPE(LOCAL)  
STATUS(ACTIVE) SUBCOUNT(6)  
TPCOUNT(9)  
AMQ8723: Display pub/sub status details.  
QMNAME(QM2) TYPE(CHILD)  
STATUS(ACTIVE) SUBCOUNT(NONE)  
TPCOUNT(NONE)  
AMQ8723: Display pub/sub status details.
```

QMNAME(QM3) TYPE(CHILD)
STATUS(ACTIVE) SUBCOUNT(NONE)
TPCOUNT(NONE)

4. Verwenden Sie die Anwendungen amqspub.exe und amqssub.exe, um das Thema zu veröffentlichen und zu abonnieren.

a) Führen Sie im ersten Befehlsfenster folgenden Befehl aus:

```
amqspub Sport/Soccer QM2
```

b) Führen Sie im zweiten Befehlsfenster folgenden Befehl aus:

```
amqssub Sport/Soccer QM1
```

c) Führen Sie im dritten Befehlsfenster folgenden Befehl aus:

```
amqssub Sport/Soccer QM3
```

Ergebnisse

Die amqssub.exe -Anwendungen im zweiten und dritten Befehlsfenster empfangen die im ersten Befehlsfenster veröffentlichten Nachrichten

Zugehörige Tasks

„Szenario 1 zu Publish/Subscribe-Hierarchie: Punkt-zu-Punkt-Kanäle mit einem Warteschlangenmanager-Aliasnamen verwenden“ auf Seite 45

Dies ist das erste von drei Szenarios, in denen auf unterschiedliche Arten eine Publish/Subscribe-Hierarchie zur Herstellung der Verbindung zwischen Warteschlangenmanagern aufgebaut wird. In diesem Szenario wird eine Publish/Subscribe-Hierarchie eingerichtet, die Punkt-zu-Punkt-Kanäle mit einem Warteschlangenmanager-Aliasnamen verwendet.

„Szenario 2 zu Publish/Subscribe-Hierarchie: Punkt-zu-Punkt-Kanäle verwenden, wobei der Name der Übertragungswarteschlange mit dem des fernen Warteschlangenmanagers identisch ist“ auf Seite 48

Dies ist das zweite von drei Szenarios, in denen auf unterschiedliche Arten eine Publish/Subscribe-Hierarchie zur Herstellung der Verbindung zwischen Warteschlangenmanagern aufgebaut wird. In diesem Szenario wird eine Publish/Subscribe-Hierarchie eingerichtet, die Punkt-zu-Punkt-Kanäle verwendet, wobei der Name der Übertragungswarteschlange mit dem des fernen Warteschlangenmanagers identisch ist.

WS-Manager mit einer Publish/Subscribe-Hierarchie verbinden

Transaktionsunterstützungsszenarios

Mittels Transaktionsunterstützung konfigurieren Sie für Ihre Anwendungen eine zuverlässige Interaktion mit Datenbanken.

In diesem Abschnitt wird die Transaktionsunterstützung beschrieben. Damit Ihre Anwendungen IBM MQ mit einem Datenbankprodukt verwenden können, sind u. a. Maßnahmen im Bereich der Anwendungsprogrammierung und Systemverwaltung erforderlich. Verwenden Sie die hier aufgeführten Informationen zusammen mit den Angaben im Abschnitt Arbeitseinheiten festschreiben und zurücksetzen.

Zunächst sollen die einzelnen Arbeitseinheiten der Transaktionen vorgestellt werden, anschließend wird beschrieben, wie IBM MQ für die Koordination von Transaktionen mit Datenbanken vorbereitet werden kann.

Zugehörige Konzepte

„Arbeitseinheiten - Einführung“ auf Seite 56

In diesem Abschnitt werden die allgemeinen Begriffe 'Arbeitseinheit', 'Festschreibung', 'Backout' und 'Synchronisationspunkt' vorgestellt und definiert. Außerdem finden Sie hier zwei Szenarios zur Veranschaulichung globaler Arbeitseinheiten.

Arbeitseinheiten - Einführung

In diesem Abschnitt werden die allgemeinen Begriffe 'Arbeitseinheit', 'Festschreibung', 'Backout' und 'Synchronisationspunkt' vorgestellt und definiert. Außerdem finden Sie hier zwei Szenarios zur Veranschaulichung globaler Arbeitseinheiten.

Wenn ein Programm Nachrichten in Warteschlangen innerhalb einer Arbeitseinheit einreicht, werden diese Nachrichten für andere Programme erst sichtbar, wenn das Programm die Arbeitseinheit *fest schreibt*. Zur Gewährleistung der Datenintegrität müssen für das Commit einer Arbeitseinheit alle Aktualisierungen erfolgreich durchgeführt worden sein.

Falls das Programm einen Fehler erkennt und beschließt, die Put-Operation nicht permanent zu machen, kann es die Arbeitseinheit zurücksetzen (*Backout*). Beim Backout eines Programms stellt IBM MQ die Warteschlangen wieder her und entfernt zu diesem Zweck die Nachrichten, die von der betreffenden Arbeitseinheit in die Warteschlangen eingereicht wurden.

Ähnlich gilt: Wenn ein Programm Nachrichten aus einer oder mehreren Warteschlangen innerhalb einer Arbeitseinheit abrufen, bleiben diese Nachrichten in den Warteschlangen, bis das Programm die Arbeitseinheit fest schreibt, können jedoch nicht von anderen Programmen abgerufen werden. Wenn das Programm die Arbeitseinheit fest schreibt, werden die Nachrichten permanent aus den Warteschlangen gelöscht. Bei einem Backout der Arbeitseinheit durch das Programm stellt IBM MQ die Warteschlangen wieder her, indem es sie wieder für den Abruf durch andere Programme zur Verfügung stellt.

Die Entscheidung über Festschreibung oder Backout der Änderungen fällt im einfachsten Fall am Ende einer Task. Es kann jedoch für eine Anwendung sinnvoller sein, Datenänderungen an anderen logischen Punkten innerhalb einer Task zu synchronisieren. Diese logischen Punkte werden Synchronisationspunkte genannt, der Zeitraum der Verarbeitung einer Reihe von Aktualisierungen zwischen zwei Synchronisationspunkten wird als *Arbeitseinheit* bezeichnet. Eine einzelne Arbeitseinheit kann mehrere MQGET- und MQPUT-Aufrufe umfassen.

Bei IBM MQ muss zwischen *lokalen* und *globalen* Arbeitseinheiten unterschieden werden:

Lokale Arbeitseinheiten

Arbeitseinheiten, bei denen lediglich Nachrichten in IBM MQ-Warteschlangen eingereicht und daraus abgerufen werden und die Koordination der einzelnen Arbeitseinheiten innerhalb des Warteschlangenmanagers mithilfe einer *einphasigen Festschreibung* erfolgt.

Verwenden Sie lokale Arbeitseinheiten, wenn als einzige Ressourcen Warteschlangen aktualisiert werden müssen, die von einem einzelnen IBM MQ-Warteschlangenmanager verwaltet werden. Aktualisierungen werden mit dem Verb MQCMIT festgeschrieben oder mit MQBACK zurückgesetzt.

Im Zusammenhang mit lokalen Arbeitseinheiten fallen abgesehen von der Protokollverwaltung keine Systemverwaltungstasks an. Versuchen Sie bei Anwendungen, bei denen Sie die Aufrufe MQPUT und MQGET mit MQCMIT und MQBACK verwenden, die Optionen MQPMO_SYNCPOINT und MQGMO_SYNCPOINT zu verwenden. (Informationen zur Protokollverwaltung finden Sie im Abschnitt [Managing log files](#) (Protokolldateien verwalten).)

Globale Arbeitseinheiten

Arbeitseinheiten, in denen auch andere Ressourcen wie z. B. Tabellen in einer relationalen Datenbank aktualisiert werden. Ist mehr als ein *Ressourcenmanager* beteiligt, wird *Transaktionsmanager*-Software benötigt, die eine *zweiphasige Festschreibung* zur Koordination der globalen Arbeitseinheit verwendet.

Verwenden Sie globale Arbeitseinheiten, wenn auch Aktualisierungen der Software für den Manager für relationale Datenbanken wie z. B. Db2, Oracle, Sybase und Informix einbezogen werden müssen.

Es gibt mehrere mögliche Szenarios für die Verwendung globaler Arbeitseinheiten. Hier sind zwei Szenarios dokumentiert:

1. Im ersten Szenario agiert der Warteschlangenmanager selbst als Transaktionsmanager. In diesem Szenario werden die globalen Arbeitseinheiten über MQI-Verben gesteuert; mit dem Verb MQBEGIN werden sie in Anwendungen gestartet und anschließend mit MQCMIT festgeschrieben oder mit MQBACK zurückgesetzt.

2. Im zweiten Szenario ist für die Rolle des Transaktionsmanagers andere Software zuständig, z. B. TXSeries, Encina oder Tuxedo. In diesem Szenario wird die Arbeitseinheit über eine von der Transaktionsmanagersoftware bereitgestellte API gesteuert (z. B. CICS SYNCPOINT für TXSeries).

In den folgenden Abschnitten werden - für das jeweilige Szenario - alle zur Verwendung globaler Arbeitseinheiten erforderlichen Schritte beschrieben:

- „Szenario 1: Der Warteschlangenmanager nimmt die Koordinierung vor“ auf Seite 57
- „Szenario 2: Für die Koordination ist andere Software zuständig“ auf Seite 84

Multi Szenario 1: Der Warteschlangenmanager nimmt die Koordinierung vor

In Szenario 1 agiert der Warteschlangenmanager als Transaktionsmanager. In diesem Szenario werden die globalen Arbeitseinheiten über MQI-Verben gesteuert; mit dem Verb MQBEGIN werden sie in Anwendungen gestartet und anschließend mit MQCMIT festgeschrieben oder mit MQBACK zurückgesetzt.

Multi Isolationsstufe

In IBM MQ kann es je nach in der Datenbank implementierter Transaktionsisolationsgestaltung vorkommen, dass vor einem Datenbankupdate eine Nachricht in einer Warteschlange zu sehen ist.

Wenn ein IBM MQ-Warteschlangenmanager als XA-Transaktionsmanager fungiert, wird zur Koordination von Aktualisierungen der XA-Ressourcenmanager das folgende Festschreibungsprotokoll befolgt:

1. Vorbereitung aller XA-Ressourcenmanager.
2. Festschreibung des IBM MQ-Warteschlangenmanager-Ressourcenmanagers.
3. Festschreibung anderer Ressourcenmanager.

Zwischen Schritt 2 und Schritt 3 kann es vorkommen, dass eine Anwendung eine in der Warteschlange festgeschriebene Nachricht sieht, diese jedoch in der entsprechenden Zeile in der Datenbank nicht wiedergegeben ist.

Dies ist kein Problem, wenn die Datenbank so konfiguriert ist, dass die Datenbank-API-Aufrufe der Anwendung auf den Abschluss ausstehender Aktualisierungen warten.

Sie können diese Situation beheben, indem Sie die Datenbank anders konfigurieren. Der benötigte Konfigurationstyp wird als "Isolationsstufe" bezeichnet. Weitere Informationen zu Isolationsstufen finden Sie in der Datenbankdokumentation. Alternativ dazu können Sie den Warteschlangenmanager so konfigurieren, dass die Ressourcenmanager in der folgenden umgekehrten Reihenfolge festgeschrieben werden.

1. Vorbereitung aller XA-Ressourcenmanager.
2. Festschreibung anderer Ressourcenmanager.
3. Festschreibung des IBM MQ-Warteschlangenmanager-Ressourcenmanagers.

Bei einer Protokolländerung wird der IBM MQ-Warteschlangenmanager zuletzt festgeschrieben, daher erhalten Anwendungen, die Nachrichten aus den Warteschlangen lesen, erst nach erfolgter Aktualisierung der entsprechenden Datenbank eine Nachricht.

Definieren Sie zur Konfiguration des Warteschlangenmanagers für die Verwendung dieses geänderten Protokolls die Umgebungsvariable **AMQ_REVERSE_COMMIT_ORDER**.

Definieren Sie diese Umgebungsvariable in der Umgebung, in welcher der Befehl **strmqm** zum Starten des Warteschlangenmanagers ausgeführt wird. Führen Sie beispielsweise unmittelbar vor dem Start des Warteschlangenmanagers den folgenden Befehl in der Shell aus:

```
export AMQ_REVERSE_COMMIT_ORDER=1
```

Anmerkung: Wenn Sie diese Umgebungsvariable definieren, wird möglicherweise für jede Transaktion ein zusätzlicher Protokolleintrag erstellt, dies hat also einen leichten Einfluss auf die Leistung der einzelnen Transaktionen.

Wenn der Warteschlangenmanager die globalen Arbeitseinheiten selbst koordiniert, ist es möglich, Datenbankaktualisierungen in die Arbeitseinheiten zu integrieren. Somit kann eine heterogene MQI- und SQL-Anwendung geschrieben werden und mithilfe der Verben MQCMIT und MQBACK können die Änderungen an den Warteschlangen und Datenbanken zusammen festgeschrieben oder rückgängig gemacht werden.

Der Warteschlangenmanager verwendet hierfür das in *X/Open Distributed Transaction Processing: The XA Specification* beschriebene Protokoll für zweiphasige Festschreibung. Wenn eine Arbeitseinheit festgeschrieben werden muss, fragt der Warteschlangenmanager zunächst alle beteiligten Datenbankmanager, ob sie zur Festschreibung ihrer Aktualisierungen bereit sind. Nur wenn alle Beteiligten einschließlich des Warteschlangenmanagers selbst zur Festschreibung bereit sind, werden alle Aktualisierungen der Warteschlangen und Datenbanken festgeschrieben. Falls ein Beteiligter seine Aktualisierungen nicht vorbereiten kann, wird die Arbeitseinheit stattdessen zurückgesetzt.

Im Allgemeinen wird eine globale Arbeitseinheit mit der folgenden Methode in einer Anwendung implementiert (in Pseudocode):

```
MQBEGIN
MQGET (nehmen Sie das Flag MQGMO_SYNCPOINT in die Nachrichtenoptionen auf)
MQPUT (nehmen Sie das Flag MQPMO_SYNCPOINT in die Nachrichtenoptionen auf)
SQL INSERT
MQCMIT
```

Mit MQBEGIN wird der Anfang einer globalen Arbeitseinheit bezeichnet. Mit MQCMIT wird das Ende der globalen Arbeitseinheit bezeichnet und die Arbeitseinheit mit allen beteiligten Ressourcenmanagern unter Verwendung des Protokolls für zweiphasige Festschreibung abgeschlossen.

Nach erfolgreichem Abschluss der Arbeitseinheit (auch als *Transaktion* bezeichnet) mit MQCMIT sind alle Aktionen innerhalb der Arbeitseinheit permanent bzw. irreversibel. Falls die Arbeitseinheit aus irgendeinem Grund fehlschlägt, werden alle Aktionen stattdessen zurückgesetzt. Es ist nicht möglich, eine Aktion in einer Arbeitseinheit permanent zu machen, während eine andere zurückgesetzt wird. Darin besteht das Prinzip einer Arbeitseinheit: entweder werden alle Aktionen innerhalb der Arbeitseinheit permanent gemacht oder keine.

Anmerkung:

1. Durch Aufruf von MQBACK kann der Anwendungsprogrammierer die Zurücksetzung (Backout) einer Arbeitseinheit erzwingen. Außerdem wird die Arbeitseinheit vom Warteschlangenmanager zurückgesetzt, wenn die Anwendung oder Datenbank vor Aufruf von MQCMIT *fehlschlägt*.
2. Wenn eine Anwendung MQDISC, nicht jedoch MQCMIT aufruft, verfährt der Warteschlangenmanager, als ob MQCMIT aufgerufen worden wäre, und schreibt die Arbeitseinheit fest.

Zwischen MQBEGIN und MQCMIT richtet der Warteschlangenmanager keine Aufrufe an die Datenbank zur Ressourcenaktualisierung. Somit werden die Datenbanktabellen also nur durch Ihren Code geändert (beispielsweise durch SQL INSERT im Pseudocode).

Wenn der Warteschlangenmanager während des Festschreibungsprotokolls den Kontakt zu einem der Datenbankmanager verliert, wird Gesamtwiederherstellungsunterstützung geboten. Wenn ein unbestätigter Datenbankmanager nicht mehr verfügbar ist, also ein Datenbankmanager, der erfolgreich für die Festschreibung vorbereitet wurde, jedoch noch auf eine Festschreibungs- oder Backoutentscheidung wartet, merkt sich der Warteschlangenmanager das Ergebnis der Arbeitseinheit, bis das betreffende Ergebnis erfolgreich an die Datenbank übergeben wurde. Ähnlich gilt: wenn der Warteschlangenmanager mit ausstehenden unvollständigen Festschreibungsoperationen beendet wird, bleiben diese über einen Neustart des Warteschlangenmanagers hinweg registriert. Falls eine Anwendung unerwartet beendet wird, ist die Integrität der Arbeitseinheit nicht beeinträchtigt, das Ergebnis hängt allerdings davon ab, an welcher Stelle des Prozesses die Anwendung beendet wurde (siehe [Tabelle 2 auf Seite 59](#)).

In den folgenden Tabellen ist zusammengefasst dargestellt, was geschieht, wenn die Datenbank oder das Anwendungsprogramm fehlschlägt:

Tabelle 1. Was geschieht, wenn ein Datenbankserver fehlschlägt?	
Auftreten des Fehlers	Ergebnis
Vor Anwendungsaufruf von MQCMIT.	Die Arbeitseinheit wird zurückgesetzt.
Während des Anwendungsaufrufs von MQCMIT, bevor alle Datenbanken angegeben haben, dass sie erfolgreich Vorbereitungen getroffen haben.	Die Arbeitseinheit wird mit dem Ursachencode MQRC_BACKED_OUT zurückgesetzt.
Während des Anwendungsaufrufs von MQCMIT, nachdem alle Datenbanken angegeben haben, dass sie erfolgreich Vorbereitungen getroffen haben, jedoch bevor alle die erfolgreiche Festschreibung gemeldet haben.	Die Arbeitseinheit wird vom Warteschlangenmanager in wiederherstellbarem Status gehalten, mit dem Ursachencode MQRC_OUTCOME_PENDING.
Während des Anwendungsaufrufs von MQCMIT, nachdem alle Datenbanken die erfolgreiche Festschreibung gemeldet haben.	Die Arbeitseinheit wird mit dem Ursachencode MQRC_NONE festgeschrieben.
Nach dem Anwendungsaufruf von MQCMIT.	Die Arbeitseinheit wird mit dem Ursachencode MQRC_NONE festgeschrieben.

Tabelle 2. Was geschieht, wenn ein Anwendungsprogramm fehlschlägt?	
Auftreten des Fehlers	Ergebnis
Vor Anwendungsaufruf von MQCMIT.	Die Arbeitseinheit wird zurückgesetzt.
Während des Anwendungsaufrufs von MQCMIT, bevor der Warteschlangenmanager die Anforderung MQCMIT der Anwendung erhalten hat.	Die Arbeitseinheit wird zurückgesetzt.
Während des Anwendungsaufrufs von MQCMIT, nachdem der Warteschlangenmanager die Anforderung MQCMIT der Anwendung erhalten hat.	Der Warteschlangenmanager versucht eine zweiphasige Festschreibung (vorausgesetzt, die Datenbankprodukte führen ihre Teile der Arbeitseinheit erfolgreich aus und schreiben diese fest).

Falls von MQCMIT der Ursachencode MQRC_OUTCOME_PENDING zurückgegeben wird, merkt sich der Warteschlangenmanager die Arbeitseinheit, bis er wieder Kontakt zum Datenbankserver herstellen und diesen auffordern kann, seinen Teil der Arbeitseinheit festzuschreiben. Informationen zur Vorgehensweise sowie zum Zeitpunkt der Wiederherstellung finden Sie im Abschnitt „Überlegungen zur Vorgehensweise bei Verlust des Kontakts zum XA-Ressourcenmanager“ auf Seite 77.

Der Warteschlangenmanager kommuniziert über die XA-Schnittstelle mit den Datenbankmanagern, wie in *X/Open Distributed Transaction Processing: The XA Specification* beschrieben. Beispiele für diese Funktionsaufrufe sind 'xa_open', 'xa_start', 'xa_end', 'xa_prepare' und 'xa_commit'. Wir verwenden die Begriffe *Transaktionsmanager* und *Ressourcenmanager* im Sinne der XA-Spezifikation.

Multi **Einschränkungen**

Bezüglich der Unterstützung der Datenbankkoordination gibt es gewisse Einschränkungen.

Es gelten folgende Einschränkungen:

- Die Möglichkeit der Koordination von Datenbankaktualisierungen innerhalb von IBM MQ-Arbeitseinheiten wird in einer MQI-Clientanwendung **nicht** unterstützt. Bei Verwendung von MQBEGIN in einer Clientanwendung kommt es zu einem Fehler. Ein Programm, das MQBEGIN aufruft, muss als *Serveranwendung* auf demselben System wie der Warteschlangenmanager ausgeführt werden.

Anmerkung: Bei einer *Serveranwendung* handelt es sich um ein Programm, das mit den erforderlichen IBM MQ-Serverbibliotheken verknüpft wurde; eine *Clientanwendung* ist ein Programm, das mit den erforderlichen IBM MQ-Clientbibliotheken verknüpft wurde. Details zum Kompilieren und Verlinken von

Programmen, die Sie in einer prozeduralen Sprache schreiben, finden Sie unter [Anwendungen für IBM MQ MQI clients erstellen](#) und [Erstellung einer prozeduralen Anwendung](#).

- Der Datenbankserver kann sich auf einem anderen System als der Warteschlangenmanagerserver befinden, wenn der Datenbankclient auf demselben System wie der Warteschlangenmanager installiert ist und diese Funktion unterstützt. Prüfen Sie in der Dokumentation zu dem Datenbankprodukt, ob die Client-Software für Systeme mit zweiphasiger Festschreibung verwendet werden kann.
- Zwar fungiert der Warteschlangenmanager als Ressourcenmanager (um in die globalen Arbeitseinheiten aus Szenario 2 einbezogen zu werden), es ist jedoch nicht möglich, einen Warteschlangenmanager einen anderen Warteschlangenmanager innerhalb der globalen Arbeitseinheiten von Szenario 1 koordinieren zu lassen.

Multi **Switchloaddateien**

Bei der Switchloaddatei handelt es sich um eine freigegebene Bibliothek (eine DLL in Windows-Systemen), die vom Code in Ihrer IBM MQ-Anwendung und dem Warteschlangenmanager geladen wird. Mit dieser Datei soll das Laden der freigegebenen Clientbibliothek der Datenbank vereinfacht werden, außerdem sollen die Verweise auf die XA-Funktionen zurückgegeben werden.

Die Details der Switchloaddatei müssen vor dem Start des Warteschlangenmanagers angegeben werden. Die Details werden in der Datei "qm.ini" auf AIX, Linux, and Windows-Systemen gespeichert.

- Aktualisieren Sie unter Windows und Linux (x86- und x86-64-Plattformen) die Datei 'qm.ini' in IBM MQ Explorer.
- Bearbeiten Sie die Datei 'qm.ini' auf allen anderen Systemen direkt.

Die C-Quelle für die Switchloaddatei wird mit der IBM MQ-Installation bereitgestellt, falls die globalen Arbeitseinheiten aus Szenario 1 unterstützt werden. Die Quelle enthält eine Funktion mit der Bezeichnung 'MQStart'. Beim Laden der Switchloaddatei ruft der Warteschlangenmanager diese Funktion auf, die die Adresse einer Struktur, eines sogenannten *XA-Switches*, zurückgibt.

Die XA-Switchstruktur ist in der freigegebenen Datenbankclientbibliothek enthalten und umfasst eine Reihe von Funktionszeigern, wie in [Tabelle 3 auf Seite 60](#) beschrieben:

Funktionszeigername	XA-Funktion	Zweck
xa_open_entry	xa_open	Verbindung zur Datenbank
xa_close_entry	xa_close	Trennung der Verbindung zur Datenbank
xa_start_entry	xa_start	Starten einer Verzweigung einer globalen Arbeitseinheit
xa_end_entry	xa_end	Aussetzen einer Verzweigung einer globalen Arbeitseinheit
xa_rollback_entry	xa_rollback	Rollback einer Verzweigung einer globalen Arbeitseinheit
xa_prepare_entry	xa_prepare	Vorbereitung der Festschreibung einer Verzweigung einer globalen Arbeitseinheit
xa_commit_entry	xa_commit	Festschreibung einer Verzweigung einer globalen Arbeitseinheit
xa_recover_entry	xa_recover	Erkennen unbestätigter Arbeitseinheiten in der Datenbank

Tabelle 3. XA-Switch-Funktionszeiger (Forts.)		
Funktionszeigername	XA-Funktion	Zweck
xa_forget_entry	xa_forget	Übergehen einer Verzweigung einer globalen Arbeitseinheit durch die Datenbank zulassen
xa_complete_entry	xa_complete	Beendigung einer Verzweigung einer globalen Arbeitseinheit

Während des ersten Aufrufs MQBEGIN in Ihrer Anwendung lädt der im Rahmen von MQBEGIN ausgeführte IBM MQ-Code die Switchloaddatei und ruft die Funktion 'xa_open' in der freigegebenen Datenbankbibliothek auf. In ähnlicher Weise laden manche Warteschlangenmanagerprozesse beim Start des Warteschlangenmanagers und späteren Gelegenheiten die Switchloaddatei und rufen die Funktion 'xa_open' auf.

Über die *dynamische Registrierung* kann die Zahl der Aufrufe 'xa_*' reduziert werden. Eine umfassende Beschreibung dieses Optimierungsverfahrens finden Sie im Abschnitt „Dynamische XA-Registrierung“ auf Seite 81.

Multi System für die Datenbankkoordination konfigurieren

Bevor ein Datenbankmanager an den vom Warteschlangenmanager koordinierten globalen Arbeitseinheiten teilnehmen kann, müssen mehrere Tasks ausgeführt werden. Diese werden in den folgenden Abschnitten beschrieben:

- „Installation und Konfiguration des Datenbankprodukts“ auf Seite 61
- „Switchloaddateien erstellen“ auf Seite 62
- „Konfigurationsinformationen zum Warteschlangenmanager hinzufügen“ auf Seite 63
- „Anwendungen schreiben und ändern“ auf Seite 64
- „System testen“ auf Seite 65

Multi Installation und Konfiguration des Datenbankprodukts

Beachten Sie für die Installation und Konfiguration Ihres Datenbankprodukts die produkteigene Dokumentation. In den Abschnitten in diesem Kapitel werden allgemeine Themen im Zusammenhang mit der Konfiguration beschrieben und es wird erläutert, wie sich diese auf den gemeinsamen Einsatz von IBM MQ und der Datenbank auswirken.

Datenbankverbindungen

Anwendungen, die eine Standardverbindung zum Warteschlangenmanager aufbauen, werden einem Thread in einem separaten Agentenprozess des lokalen Warteschlangenmanagers zugeordnet. (In diesem Zusammenhang gelten Verbindungen, bei denen es sich nicht um *Fastpath*-Verbindungen handelt, als *Standardverbindungen*. Beachten Sie die Informationen im Abschnitt Verbindung zu einem Warteschlangenmanager über MQCONN-Aufruf herstellen.)

Wenn die Anwendung **MQBEGIN** ausgibt, wird die Funktion **xa_open** in der Datenbankclientbibliothek sowohl von der Anwendung selbst als auch vom Agentenprozess aufgerufen. Als Antwort darauf stellt der Code der Datenbankclientbibliothek *sowohl von den Anwendungs- als auch von den Warteschlangenmanagerprozessen* aus eine *Verbindung* zu der Datenbank her, die in die Arbeitseinheit einbezogen werden soll. Diese Datenbankverbindungen bleiben erhalten, solange die Anwendung mit dem Warteschlangenmanager verbunden bleibt.

Dies ist ein wichtiger Gesichtspunkt, wenn die Datenbank nur eine begrenzte Anzahl an Benutzern oder Verbindungen unterstützt, da zur Unterstützung des einen Anwendungsprogramms zwei Verbindungen zur Datenbank hergestellt werden.

Client-/Serverkonfiguration

Die in die IBM MQ-Warteschlangenmanager- und -Anwendungsprozesse geladene Datenbankclientbibliothek **mus** in der Lage sein, Daten an den Server zu senden und von dort zu empfangen. Vergewissern Sie sich, dass folgende Voraussetzungen erfüllt sind:

- Die Angaben in den Client-/Server-Konfigurationsdateien der Datenbank sind korrekt.
- In der Umgebung der Warteschlangenmanager- **und** Anwendungsprozesse die relevanten Umgebungsvariablen definiert.

Multi **Switchloaddateien erstellen**

IBM MQ wird mit einer Beispielmakedatei geliefert, die zum Erstellen von Switchloaddateien für die unterstützten Datenbankmanager verwendet wird.

`MQ_INSTALLATION_PATH` steht für das übergeordnete Verzeichnis, in dem IBM MQ installiert ist.

Die Beispielmakedatei wird zusammen mit allen zum Erstellen der Switchloaddateien erforderlichen zugehörigen C-Quellendateien in den folgenden Verzeichnissen installiert:

- Für IBM MQ for Windowsim Verzeichnis `MQ_INSTALLATION_PATH\tools\c\samples\xatm\`
- Für IBM MQ for UNIX -und Linux -Systeme im Verzeichnis `MQ_INSTALLATION_PATH/samp/xatm/`

Für die Erstellung der Switchloaddateien werden folgende Beispielizele verwendet:

- Für Db2 `db2swit`.
- Für Oracle, `oraswit`
- Für Informix `infswit`
- Für Sybase: `sybswit`

Windows

Die generierte Switchdatei wird in das Verzeichnis `C:\Programme\IBM\MQ\exits` gestellt.

Linux

UNIX

Bei Verwendung von 32-Bit-Warteschlangenmanagern installiert die Beispielmakedatei 'xaswit.mak' eine 32-Bit-Switchloaddatei im Verzeichnis `/var/mqm/exits`.

Linux

UNIX

Bei Verwendung von 64-Bit-Warteschlangenmanagern installiert die Beispielmakedatei 'xaswit.mak' eine 32-Bit-Switchloaddatei im Verzeichnis `/var/mqm/exits` und eine 64-Bit-Switchloaddatei im Verzeichnis `/var/mqm/exits64`.

Linux

UNIX

Wenn Ihr System die 32-Bit-Kompilierung nicht unterstützt, verwenden Sie das nur 64-Bit-Ziel für Ihre Datenbank:

- Für Db2, `db2swit64`
- Für Oracle: `oraswit64`
- Für Informix, `infswit64`
- Für Sybase, `sybswit64`

Dateisicherheit

Möglicherweise kann die Switchloaddatei von IBM MQ unter Ihrem Betriebssystem nicht geladen werden, wobei die Gründe hierfür außerhalb der Kontrolle von IBM MQ liegen. In diesem Fall werden Fehlernachrichten in die IBM MQ-Fehlerprotokolle geschrieben und unter Umständen schlägt der Aufruf `MQBEGIN` fehl. Um sicherzugehen, dass die Switchloaddatei unter Ihrem Betriebssystem geladen werden kann, müssen folgende Voraussetzungen erfüllt sein:

1. Die Switchloaddatei muss sich in dem in der Datei 'qm.ini' angegebenen Verzeichnis befinden.
2. Die Switchloaddatei muss für alle Prozesse zugänglich sein, die diese laden müssen, einschließlich der Warteschlangenmanager- und Anwendungsprozesse.

3. Alle Bibliotheken, von denen die Switchloaddatei abhängt, einschließlich der vom Datenbankprodukt bereitgestellten Bibliotheken, müssen vorhanden und zugänglich sein.

Multi Konfigurationsinformationen zum Warteschlangenmanager hinzufügen

Wenn Sie eine Switchloaddatei für Ihren Datenbankmanager erstellt und in ein sicheres Verzeichnis gestellt haben, müssen Sie dieses Verzeichnis Ihrem Warteschlangenmanager angeben.

Gehen Sie zur Angabe des Verzeichnisses wie folgt vor:

- Verwenden Sie unter Windows und Linux (x86- und x86-64-Plattformen) den IBM MQ Explorer. Geben Sie die Details zu der Switchloaddatei in der Eigenschaftsanzeige für den Warteschlangenmanager unter dem XA-Ressourcenmanager an.
- Geben Sie bei allen anderen Systemen die Details zu der Switchloaddatei in der XAResourceManager-Zeilengruppe in der Datei 'qm.ini' des Warteschlangenmanagers an.

Fügen Sie für die Datenbank, die Ihr Warteschlangenmanager koordinieren soll, eine XAResourceManager-Zeilengruppe hinzu. In den meisten Fällen gibt es nur eine Datenbank und daher auch nur eine XAResourceManager-Zeilengruppe. Ausführliche Informationen zu komplizierteren Konfigurationen mit mehreren Datenbanken finden Sie im Abschnitt „Mehrere Datenbankkonfigurationen“ auf Seite 75. Die Attribute der XAResourceManager-Zeilengruppe lauten wie folgt:

Name=name

Vom Benutzer ausgewählte Zeichenfolge zur Angabe des Ressourcenmanagers. Damit wird der XAResourceManager-Zeilengruppe praktisch ein Name gegeben. Der Name ist obligatorisch und kann bis zu 31 Zeichen umfassen.

Sie müssen einen eindeutigen Namen auswählen; es darf nur eine XAResourceManager-Zeilengruppe mit diesem Namen in der Datei 'qm.ini' geben. Der Name sollte auch aussagekräftig sein, da der Warteschlangenmanager diesen Namen sowohl in den Fehlerprotokollnachrichten des Warteschlangenmanagers als auch in der Ausgabe des Befehls `dspmqrtrn` zur Angabe dieses Ressourcenmanagers verwendet. (Weitere Informationen hierzu finden Sie im Abschnitt „Ausstehende Arbeitseinheiten mit dem Befehl 'dspmqrtrn' anzeigen“ auf Seite 78.)

Ändern Sie das Attribut 'Name' nicht mehr, nachdem Sie einen Namen ausgewählt und den Warteschlangenmanager gestartet haben. Weitere Informationen zum Ändern der Konfigurationsinformationen finden Sie im Abschnitt „Konfigurationsinformationen ändern“ auf Seite 80.

SwitchFile=name

Dies ist der Name der zuvor erstellten XA-Switchloaddatei. Hierbei handelt es sich um ein obligatorisches Attribut. Der Code in den Warteschlangenmanager- und IBM MQ-Anwendungsprozessen versucht bei zwei Gelegenheiten, die Switchloaddatei zu laden:

1. Beim Start des Warteschlangenmanagers
2. Beim ersten Aufruf von MQBEGIN in Ihrem IBM MQ-Anwendungsprozess

Die Prozesse müssen über die Sicherheits- und Berechtigungsattribute der Switchloaddatei zu dieser Aktion berechtigt sein.

XAOpenString=string

Eine Datenzeichenfolge, die der IBM MQ-Code in seinen Aufrufen an die Funktion 'xa_open' des Datenbankmanagers übergibt. Hierbei handelt es sich um ein optionales Attribut. Wenn es nicht angegeben ist, wird eine Zeichenfolge der Länge null vorausgesetzt.

Der Code in den Warteschlangenmanager- und IBM MQ-Anwendungsprozessen ruft die Funktion 'xa_open' bei zwei Gelegenheiten auf:

1. Beim Start des Warteschlangenmanagers
2. Beim ersten Aufruf von MQBEGIN in Ihrem IBM MQ-Anwendungsprozess

Für diese Zeichenfolge gilt bei jedem Datenbankprodukt ein besonderes Format, das in der zugehörigen Produktdokumentation beschrieben wird. Im Allgemeinen enthält die Zeichenfolge 'xa_open' Authentifizierungsinformationen (Benutzername und Kennwort), mit denen in Warteschlangenmanager- und Anwendungsprozessen eine Verbindung zu der Datenbank hergestellt werden kann.

Ab IBM MQ 8.0.0 Fix Pack 4 können Sie, wenn XAOpenString ein Kennwort enthält, dafür sorgen, dass IBM MQ diese Information schützt. Das Kennwort ist dann nicht mehr als Klartext in der Datei `qm.ini` sichtbar. IBM MQ speichert den Benutzernamen und das Kennwort (in verschlüsselter Form) in einer anderen Datei und verwendet diese Berechtigungsnachweise für Verbindungen mit der Datenbank. Weitere Informationen finden Sie im Abschnitt [Schutz von Datenbankauthentifizierungsdetails](#).

XACloseString=string

Eine Datenzeichenfolge, die der IBM MQ-Code in seinen Aufrufen an die Funktion 'xa_close' des Datenbankmanagers übergibt. Hierbei handelt es sich um ein optionales Attribut. Wenn es nicht angegeben ist, wird eine Zeichenfolge der Länge null vorausgesetzt.

Der Code in den Warteschlangenmanager- und IBM MQ-Anwendungsprozessen ruft die Funktion 'xa_close' bei zwei Gelegenheiten auf:

1. Beim Start des Warteschlangenmanagers
2. Beim Aufruf von MQDISC in Ihrem IBM MQ-Anwendungsprozess nach einem vorherigen Aufruf von MQBEGIN

Für diese Zeichenfolge gilt bei jedem Datenbankprodukt ein besonderes Format, das in der zugehörigen Produktdokumentation beschrieben wird. Im Allgemeinen ist die Zeichenfolge leer, das Attribut 'XACloseString' wird häufig aus der XAResourceManager-Zeilengruppe ausgeschlossen.

ThreadOfControl=THREAD |PROCESS (Standardwert)

Der Wert für 'ThreadOfControl' kann THREAD oder PROCESS lauten. Der Warteschlangenmanager verwendet das Attribut für Serialisierungszwecke. Hierbei handelt es sich um ein optionales Attribut, wenn es nicht angegeben ist, wird der Wert PROCESS vorausgesetzt.

Wenn der Code für den Datenbankclient den Threads den Aufruf der XA-Funktionen ohne Serialisierung ermöglicht, kann der Wert für 'ThreadOfControl' THREAD lauten. Der Warteschlangenmanager geht davon aus, dass er die XA-Funktionen in der freigegebenen Bibliothek des Datenbankclients gegebenenfalls von mehreren Threads aus gleichzeitig aufrufen kann.

Lässt der Code für den Datenbankclient nicht zu, dass Threads die XA-Funktionen auf diese Weise aufrufen, muss der Wert für 'ThreadOfControl' PROCESS lauten. Der Warteschlangenmanager serialisiert dann alle Aufrufe der freigegebenen Datenbankclientbibliothek, sodass immer nur jeweils ein Aufruf aus einem bestimmten Prozess erfolgt. Vermutlich müssen Sie auch dafür sorgen, dass Ihre Anwendung eine ähnliche Serialisierung vornimmt, wenn sie mit mehreren Threads ausgeführt wird.

Beachten Sie, dass die Fähigkeit des Datenbankprodukts, mit Multithread-Prozessen auf diese Weise zu verfahren, Sache des betreffenden Produkthanbieters ist. Lesen Sie in der Dokumentation zu Ihrem Datenbankprodukt nach, ob Sie das Attribut 'ThreadOfControl' auf THREAD oder PROCESS setzen können. Wir empfehlen, falls möglich die Einstellung THREAD für 'ThreadOfControl' zu verwenden. Im Zweifelsfall ist es *sicherer*, die Einstellung PROCESS zu verwenden, Sie verlieren dann allerdings die potenziellen Leistungsvorteile der Verwendung von THREAD.

Multi *Anwendungen schreiben und ändern*

Hier wird die Vorgehensweise zum Implementieren einer globalen Arbeitseinheit erläutert.

Die mit IBM MQ-Installationen bereitgestellten Beispielanwendungsprogramme für die globalen Arbeitseinheiten aus Szenario 1 werden im Abschnitt [„Arbeitseinheiten - Einführung“](#) auf Seite 56 beschrieben.

Im Allgemeinen wird eine globale Arbeitseinheit mit der folgenden Methode in einer Anwendung implementiert (in Pseudocode):

```
MQBEGIN
MQGET
MQPUT
SQL INSERT
MQCMIT
```

Mit MQBEGIN wird der Anfang einer globalen Arbeitseinheit bezeichnet. Mit MQCMIT wird das Ende der globalen Arbeitseinheit bezeichnet und die Arbeitseinheit mit allen beteiligten Ressourcenmanagern unter Verwendung des Protokolls für zweiphasige Festschreibung abgeschlossen.

Zwischen MQBEGIN und MQCMIT richtet der Warteschlangenmanager keine Aufrufe an die Datenbank zur Ressourcenaktualisierung. Somit werden die Datenbanktabellen also nur durch Ihren Code geändert (beispielsweise durch SQL INSERT im Pseudocode).

Die Rolle des Warteschlangenmanagers in Hinblick auf die Datenbank besteht darin, dieser mitzuteilen, wenn eine globale Arbeitseinheit gestartet oder beendet wurde und ob die globale Arbeitseinheit festgeschrieben oder rückgängig gemacht werden sollte.

Was Ihre Anwendung betrifft, übt der Warteschlangenmanager zwei Rollen aus: er ist Ressourcenmanager (wenn die Ressourcen Nachrichten in Warteschlangen sind) und Transaktionsmanager für die globale Arbeitseinheit.

Beginnen Sie mit den bereitgestellten Beispielprogrammen und arbeiten Sie die verschiedenen IBM MQ- und Datenbank-API-Aufrufe durch, die in diesen Programmen erfolgen. Die betreffenden API-Aufrufe werden in den Abschnitten Prozedurale IBM MQ-Programme, Im MQI verwendete Datentypen und (im Falle der datenbankeigenen API) in der Datenbankdokumentation beschrieben.

Multi System testen

Nur durch Testläufe können Sie feststellen, ob Ihre Anwendung und Ihr System korrekt konfiguriert sind. Die Systemkonfiguration (die erfolgreiche Kommunikation zwischen Warteschlangenmanager und Datenbank) kann durch Build und Ausführung eines der bereitgestellten Beispielprogramme getestet werden.

Multi Db2 konfigurieren

Db2-Support- und -Konfigurationsinformationen

Die unterstützten Db2-Versionen sind auf der Seite Systemvoraussetzungen für IBM MQ definiert.

Anmerkung: 32-Bit-Instanzen von Db2 werden auf Plattformen mit 64-Bit-Warteschlangenmanager nicht unterstützt.

Gehen Sie wie folgt vor:

1. Prüfen Sie die Einstellungen der Umgebungsvariablen.
2. Erstellen Sie die Db2-Switchloaddatei.
3. Fügen Sie Konfigurationsinformationen für den Ressourcenmanager hinzu.
4. Ändern Sie bei Bedarf die Db2-Konfigurationsparameter.

Lesen Sie diese Informationen zusammen mit den allgemeinen Angaben im Abschnitt „System für die Datenbankkoordination konfigurieren“ auf Seite 61.

Warnung: Wenn Sie `db2profile` auf den Plattformen AIX and Linux ausführen, werden die Umgebungsvariablen `LIBPATH` und `LD_LIBRARY_PATH` definiert. Es wird empfohlen, diese Umgebungsvariablen `unset` zu verwenden. Weitere Informationen finden Sie unter `crtmqenv` oder `setmqenv`.

Einstellungen der Db2-Umgebungsvariablen prüfen

Stellen Sie sicher, dass Ihre Db2-Umgebungsvariablen sowohl für Warteschlangenmanagerprozesse **als auch in** Ihren Anwendungsprozessen definiert sind. Insbesondere muss die Umgebungsvariable `Db2INSTANCE` immer **vor** dem Start des Warteschlangenmanagers definiert werden. Über die Umgebungsvariable `DB2INSTANCE` wird die Db2-Instanz angegeben, in der die zu aktualisierenden Db2-Datenbanken enthalten sind. For example:

- Verwenden Sie auf Systemen mit AIX and Linux folgende Angabe:

```
export DB2INSTANCE=db2inst1
```

- Verwenden Sie auf Systemen mit Windows folgende Angabe:

```
set DB2INSTANCE=Db2
```

Unter Windows mit einer Db2 -Datenbank müssen Sie den Benutzer MUSR_MQADMIN zur Gruppe DB2USERS hinzufügen, damit der Warteschlangenmanager gestartet werden kann.

Db2-Switchloaddatei erstellen

Am einfachsten kann die Db2-Switchloaddatei unter Verwendung der Beispieldatei 'xaswit.mak' erstellt werden, die von IBM MQ zum Erstellen von Switchloaddateien für eine Vielzahl von Datenbankprodukten bereitgestellt wird.

Windows Auf Windows -Systemen finden Sie `xaswit.mak` im Verzeichnis `MQ_INSTALLATION_PATH\tools\c\samples\xatm`. `MQ_INSTALLATION_PATH` steht für das übergeordnete Verzeichnis, in dem IBM MQ installiert ist. Verwenden Sie zum Erstellen der Db2-Switchloaddatei mit Microsoft Visual C++ folgenden Befehl:

```
nmake /f xaswit.mak db2swit.dll
```

Die generierte Switchdatei wird in das Verzeichnis `C:\Programme\IBM\MQ\exits` gestellt.

Linux **UNIX** Sie finden `xaswit.mak` im Verzeichnis `MQ_INSTALLATION_PATH/samp/xatm`. `MQ_INSTALLATION_PATH` steht für das übergeordnete Verzeichnis, in dem IBM MQ installiert ist.

Bearbeiten Sie die Datei `xaswit.mak` und *entfernen Sie die Kommentarzeichen* für die entsprechenden Zeilen der von Ihnen verwendeten Db2-Version. Führen Sie anschließend die Makedatei mit dem folgenden Befehl aus:

```
make -f xaswit.mak db2swit
```

Die generierte 32-Bit-Switchloaddatei wird in `/var/mqm/exits` gespeichert.

Die generierte 64-Bit-Switchloaddatei wird in `/var/mqm/exits64` gespeichert.

Wenn Ihr System die 32-Bit-Kompilierung nicht unterstützt, verwenden Sie das 64-Bit-Ziel:

```
make -f xaswit.mak db2swit64
```

Ressourcenmanager-Konfigurationsinformationen für Db2 hinzufügen

Sie müssen die Konfigurationsinformationen für den Warteschlangenmanager ändern und die Beteiligung von Db2 an globalen Arbeitseinheiten deklarieren. Die entsprechende Änderung der Konfigurationsinformationen wird im Abschnitt „[Konfigurationsinformationen zum Warteschlangenmanager hinzufügen](#)“ auf [Seite 63](#) ausführlicher beschrieben.

- Verwenden Sie unter Windows und Linux (x86- und x86-64-Plattformen) IBM MQ Explorer. Geben Sie die Details zu der Switchloaddatei in der Eigenschaftsanzeige für den Warteschlangenmanager unter dem XA-Ressourcenmanager an.
- Geben Sie bei allen anderen Systemen die Details zu der Switchloaddatei in der XAResourceManager-Zeilengruppe in der Datei 'qm.ini' des Warteschlangenmanagers an.

Abbildung 6 auf [Seite 67](#) ist ein UNIX-Beispiel mit einem XAResourceManager-Eintrag, bei dem die zu koordinierende Datenbank die Bezeichnung `mydbname` trägt, wobei dieser Namen in `XAOpenString` angegeben wird:

```
XAResourceManager:  
  Name=mydb2  
  SwitchFile=db2swit  
  XAOpenString=mydbname,myuser,mypasswd,toc=t  
  ThreadOfControl=THREAD
```

Abbildung 6. XAResourceManager-Beispielintrag für Db2 auf UNIXen

Anmerkung:

1. ThreadOfControl=THREAD kann nicht mit Db2 -Versionen vor der 8. verwendet werden. Legen Sie ThreadOfControl und den XAOpenString-Parameter toc auf eine der folgenden Kombinationen fest:

- ThreadOfControl=THREAD und toc=t
- ThreadOfControl=PROCESS und toc=p

Wenn Sie die XA-Switchloaddatei 'jdbcd2' verwenden, um die JDBC/JTA-Koordination zu ermöglichen, müssen Sie ThreadOfControl=PROCESS und toc=p verwenden.

Db2-Konfigurationsparameter ändern

Für jede vom Warteschlangenmanager koordinierte Db2-Datenbank müssen Datenbankberechtigungen festgelegt, der Parameter 'tp_mon_name' geändert und der Parameter 'maxappls' zurückgesetzt werden. Gehen Sie hierfür wie folgt vor:

Legen Sie Datenbankberechtigungen fest

Die Warteschlangenmanagerprozesse werden auf Systemen mit AIX and Linux mit dem ausführenden Benutzer und der Gruppe 'mqm' ausgeführt. Auf Windows-Systemen werden sie unter dem Benutzer ausgeführt, der den Warteschlangenmanager gestartet hat. Hierbei kann es sich um einen der folgenden Benutzer handeln:

1. Benutzer, der den Befehl stirmqm ausgegeben hat, oder
2. Benutzer, unter dem der IBM MQ Service COM-Server ausgeführt wird

Standardmäßig heißt dieser Benutzer MUSR_MQADMIN.

Wenn Sie in der Zeichenfolge 'xa_open' keinen Benutzernamen und kein Kennwort angegeben haben, wird **der Benutzer, unter dem der Warteschlangenmanager ausgeführt wird** von Db2 zur Authentifizierung des Aufrufs 'xa_open' verwendet. Verfügt dieser Benutzer (z. B. der Benutzer 'mqm' auf Systemen mit AIX and Linux) nicht über minimale Berechtigungen in der Datenbank, verweigert die Datenbank die Authentifizierung des Aufrufs 'xa_open'.

Dieselben Überlegungen gelten auch für Ihren Anwendungsprozess. Wenn Sie in der Zeichenfolge 'xa_open' keinen Benutzernamen und kein Kennwort angegeben haben, wird der Benutzer, unter dem die Anwendung ausgeführt wird, von Db2 zur Authentifizierung des Aufrufs 'xa_open' beim ersten MQBEGIN verwendet. Auch hier funktioniert dies nur, wenn der betreffende Benutzer über minimale Berechtigungen in der Datenbank verfügt.

Erteilen Sie beispielsweise dem Benutzer 'mqm' Verbindungsberechtigung in der Datenbank 'mydbname' und geben Sie dazu folgende Db2-Befehle aus:

```
db2 connect to mydbname  
db2 grant connect on database to user mqm
```

Weitere Informationen zur Sicherheit finden Sie im Abschnitt „Sicherheitsaspekte“ auf Seite 76.

Windows Ändern Sie den Parameter TP_MON_NAME.

Ändern Sie nur für Db2 auf Windows-Systemen den Konfigurationsparameter TP_MON_NAME und geben Sie hierfür die DLL an, die von Db2 zum Aufruf des Warteschlangenmanagers zur dynamischen Registrierung verwendet wird.

Verwenden Sie den Befehl `db2 update dbm cfg using TP_MON_NAME mqmax`, um MQMAX.DLL als Bibliothek, die Db2 zum Aufrufen des Warteschlangenmanagers verwendet. Diese muss sich in einem Verzeichnis innerhalb von PATH befinden.

Setzen Sie den Parameter 'maxappls' zurück

Unter Umständen müssen Sie die Einstellung für den Parameter *maxappls* prüfen, mit dem die maximale Anzahl an Anwendungen begrenzt wird, die mit einer Datenbank verbunden werden können. Weitere Informationen finden Sie im Thema „[Installation und Konfiguration des Datenbankprodukts](#)“ auf Seite 61.

Multi Oracle konfigurieren

Oracle-Support- und -Konfigurationsinformationen.

Führen Sie die folgenden Schritte aus:

1. Prüfen Sie die Einstellungen der Umgebungsvariablen.
2. Erstellen Sie die Oracle-Switchloaddatei.
3. Fügen Sie Konfigurationsinformationen für den Ressourcenmanager hinzu.
4. Ändern Sie gegebenenfalls die Oracle-Konfigurationsparameter.

Eine aktuelle Liste der von IBM MQ unterstützten Oracle-Versionen finden Sie unter [Systemvoraussetzungen für IBM MQ](#).

Einstellungen der Oracle-Umgebungsvariablen prüfen

Vergewissern Sie sich, dass Ihre Oracle-Umgebungsvariablen sowohl für Warteschlangenmanagerprozesse als auch in Ihren Anwendungsprozessen definiert sind. Legen Sie insbesondere immer die folgenden Umgebungsvariablen fest, bevor Sie den Warteschlangenmanager starten:

ORACLE_HOME

Das Oracle-Ausgangsverzeichnis. Verwenden Sie zum Beispiel auf Systemen mit AIX and Linux:

```
export ORACLE_HOME=/opt/oracle/product/8.1.6
```

Verwenden Sie auf Systemen mit Windows folgende Angabe:

```
set ORACLE_HOME=c:\oracle\ora81
```

ORACLE_SID

Die verwendete Oracle-SID. Wenn Sie für die Client/Server-Konnektivität Net8 verwenden, müssen Sie diese Umgebungsvariable möglicherweise nicht festlegen. Lesen Sie dies in Ihrer Oracle-Dokumentation nach.

Nachfolgend finden Sie ein Beispiel für die Definition dieser Umgebungsvariablen auf Systemen mit AIX and Linux:

```
export ORACLE_SID=sid1
```

Die äquivalente Angabe auf Windows-Systemen lautet wie folgt:

```
set ORACLE_SID=sid1
```

Anmerkung: Die Umgebungsvariable PATH muss so eingestellt sein, dass sie das Verzeichnis "binaries" (z. B. ORACLE_INSTALL_DIR/VERSION/32BIT_NAME/bin oder ORACLE_INSTALL_DIR/VERSION/64BIT_NAME/bin) enthält. Andernfalls wird möglicherweise eine Nachricht angezeigt, die besagt, dass Oracle-Bibliotheken von der Maschine fehlen.

Bei Ausführung von Warteschlangenmanagern auf Windows-64-Bit-Systemen dürfen nur 64-Bit-Oracle-Clients installiert werden. Die von 64-Bit-Warteschlangenmanagern geladene Switchloaddatei muss auf die Oracle-64-Bit-Clientbibliotheken zugreifen.

Oracle-Switchloaddatei erstellen

Verwenden Sie zum Erstellen der Oracle-Switchloaddatei die Beispieldatei `xaswit.mak`, die IBM MQ bereitstellt, um die Switchloaddateien für verschiedene Datenbankprodukte zu erstellen.

Windows Auf Windows-Systemen finden Sie `xaswit.mak` im Verzeichnis `C:\Programme\IBM\MQ\tools\c\samples\xatm`. Um die Oracle-Switchloaddatei mit Microsoft Visual C++ zu erstellen, verwenden Sie Folgendes:

```
nmake /f xaswit.mak oraswit.dll
```

Anmerkung: Diese Switchloaddateien können nur mit in der Programmiersprache C geschriebenen Anwendungen verwendet werden. Für Java-Anwendungen lesen Sie hierzu bitte den Abschnitt [JTA/JDBC-Koordination mit IBM MQ classes for Java](#).

Die generierte Switchdatei wird in das Verzeichnis `MQ_INSTALLATION_PATH\exits` gestellt. `MQ_INSTALLATION_PATH` steht für das übergeordnete Verzeichnis, in dem IBM MQ installiert ist.

Linux **UNIX** Die Datei `xaswit.mak` ist im Verzeichnis `MQ_INSTALLATION_PATH/samp/xatm` zu finden. `MQ_INSTALLATION_PATH` steht für das übergeordnete Verzeichnis, in dem IBM MQ installiert ist.

Bearbeiten Sie `xaswit.mak`, um die Kommentarzeichen für die Zeilen zu entfernen, die für die von Ihnen verwendete Version von Oracle geeignet sind. Führen Sie anschließend die Makedatei mit dem folgenden Befehl aus:

```
make -f xaswit.mak oraswit
```

Die Inhalte von `MQ_INSTALLATION_PATH/samp/xatm` sind bei der Installation von IBM MQ schreibgeschützt; um `xaswit.mak` editieren zu können, müssen Sie daher alle Dateien aus dem Verzeichnis `samp/xatm` in ein anderes Verzeichnis kopieren, `xaswit.mak` entsprechend modifizieren und dann den Befehl `make -f xaswit.mak oraswit` aus diesem Verzeichnis heraus ausführen.

Die generierte 32-Bit-Switchloaddatei wird in das Verzeichnis `/var/mqm/exits` gestellt.

Die generierte 64-Bit-Switchloaddatei wird in das Verzeichnis `/var/mqm/exits64` gestellt.

Wenn Ihr System die 32-Bit-Kompilierung nicht unterstützt, verwenden Sie das 64-Bit-Ziel:

```
make -f xaswit.mak oraswit64
```

Ressourcenmanager-Konfigurationsinformationen für Oracle hinzufügen

Sie müssen die Konfigurationsinformationen für den Warteschlangenmanager ändern und die Beteiligung von Oracle an globalen Arbeitseinheiten deklarieren. Die entsprechende Änderung der Konfigurationsinformationen für den Warteschlangenmanager wird im Abschnitt [„Konfigurationsinformationen zum Warteschlangenmanager hinzufügen“](#) auf Seite 63 ausführlicher beschrieben.

- Verwenden Sie unter Windows und Linux (x86- und x86-64-Plattformen) IBM MQ Explorer. Geben Sie die Details zu der Switchloaddatei in der Eigenschaftsanzeige für den Warteschlangenmanager unter dem XA-Ressourcenmanager an.

- Geben Sie auf allen anderen Systemen die Details der Switchloaddatei in der Zeilengruppe "XAResourceManager" in der `qm.ini`-Datei des Warteschlangenmanagers an.

Abbildung 7 auf Seite 70 ist ein Beispiel für Systeme mit AIX and Linux mit einem XAResourceManager-Eintrag. Der XA-Zeichenfolge zum Öffnen muss `LogDir` hinzugefügt werden, damit alle Fehler- und Traceinformationen an derselben Stelle protokolliert werden.

```
XAResourceManager:
  Name=myoracle
  SwitchFile=oraswit
  XAOpenString=oracle_XA+Acc=P/myuser/mypasswd+SesTm=35+LogDir=/tmp+threads=true
  ThreadOfControl=THREAD
```

Abbildung 7. XAResourceManager-Beispieleintrag für Oracle auf den Plattformen AIX and Linux

Anmerkung:

1. In Abbildung 7 auf Seite 70 wurde die Zeichenfolge 'xa_open' mit vier Parametern verwendet. Weitere Parameter können wie in der Oracle-Dokumentation beschrieben hinzugefügt werden.
2. Bei Verwendung des IBM MQ-Parameters `ThreadOfControl=THREAD` müssen Sie den Oracle-Parameter `+threads=true` in der XAResourceManager-Zeilengruppe verwenden.

Weitere Informationen zur Zeichenfolge 'xa_open' finden Sie im *Oracle8 Server Application Developer's Guide*.

Oracle-Konfigurationsparameter ändern

Für jede vom Warteschlangenmanager koordinierte Oracle-Datenbank muss die maximale Sitzungsanzahl geprüft werden und es müssen Datenbankberechtigungen festgelegt werden. Gehen Sie hierfür wie folgt vor:

Prüfen Sie die maximale Sitzungsanzahl

Möglicherweise müssen Sie die Einstellungen `LICENSE_MAX_SESSIONS` und `PROCESSES` prüfen und dabei die von den zum Warteschlangenmanager gehörigen Prozessen benötigten zusätzlichen Verbindungen berücksichtigen. Weitere Informationen finden Sie unter „Installation und Konfiguration des Datenbankprodukts“ auf Seite 61.

Legen Sie Datenbankberechtigungen fest

Der in der Zeichenfolge 'xa_open' angegebene Oracle-Benutzername muss über Zugriffsberechtigung für die Ansicht `DBA_PENDING_TRANSACTIONS` verfügen, wie in der Oracle-Dokumentation beschrieben.

Die erforderliche Berechtigung kann mit dem folgenden Beispielbefehl erteilt werden:

```
grant select on DBA_PENDING_TRANSACTIONS to myuser;
```

Multi Informix konfigurieren

Informix-Support- und -Konfigurationsinformationen

Führen Sie die folgenden Schritte aus:

1. Vergewissern Sie sich, dass Sie das entsprechende Informix Client SDK installiert haben:
 - Für 32-Bit-Warteschlangenmanager und -Anwendungen wird ein 32-Bit-Informix Client SDK benötigt.
 - Für 64-Bit-Warteschlangenmanager und -Anwendungen wird ein 64-Bit-Informix Client SDK benötigt.
2. Stellen Sie sicher, dass die Informix-Datenbanken ordnungsgemäß erstellt wurden.
3. Prüfen Sie die Einstellungen der Umgebungsvariablen.

4. Erstellen Sie die Informix-Switchloaddatei.

5. Fügen Sie Konfigurationsinformationen für den Ressourcenmanager hinzu.

Eine aktuelle Liste der Informix -Versionen, die von IBM MQunterstützt werden, finden Sie unter [Systemvoraussetzungen für IBM MQ](#).

Ordnungsgemäße Erstellung der Informix-Datenbanken sicherstellen

Jede Informix-Datenbank, die von einem IBM MQ-Warteschlangenmanager koordiniert werden soll, muss mit dem Parameter **log** erstellt werden. For example:

```
create database mydbname with log;
```

IBM MQ -Warteschlangenmanager können Informix -Datenbanken, für die der Parameter **log** bei der Erstellung nicht angegeben wurde, nicht koordinieren. Wenn ein Warteschlangenmanager versucht, eine Informix-Datenbank zu koordinieren, bei deren Erstellung der Parameter **log** nicht angegeben wurde, schlägt der Aufruf 'xa_open' für Informix fehl und es wird eine Reihe von FFST-Fehlern generiert.

Einstellungen der Informix-Umgebungsvariablen prüfen

Stellen Sie sicher, dass Ihre Informix-Umgebungsvariablen sowohl für Warteschlangenmanagerprozesse **als auch in** Ihren Anwendungsprozessen definiert sind. Legen Sie insbesondere immer die folgenden Umgebungsvariablen fest, **bevor** Sie den Warteschlangenmanager starten:

INFORMIXDIR

Verzeichnis der Informix-Produktinstallation

- Verwenden Sie bei 32-Bit-Anwendungen unter AIX and Linux den folgenden Befehl:

```
export INFORMIXDIR=/opt/informix/32-bit
```

- Verwenden Sie bei 64-Bit-Anwendungen unter AIX and Linux den folgenden Befehl:

```
export INFORMIXDIR=/opt/informix/64-bit
```

- Verwenden Sie bei Windows-Anwendungen folgenden Befehl:

```
set INFORMIXDIR=c:\informix
```

Bei Systemen mit 64-Bit-Warteschlangenmanagern, die sowohl 32- als auch 64-Bit-Anwendungen unterstützen müssen, müssen sowohl das Informix Client SDK mit 32 Bit als auch das Informix Client SDK mit 64 Bit installiert sein. Mit der zum Erstellen einer Switchloaddatei verwendeten Beispielmakrofile `xaswit.mak` werden ebenfalls beide Produktinstallationsverzeichnisse festgelegt.

INFORMIXSERVER

Der Name des Informix-Servers. Verwenden Sie zum Beispiel auf Systemen mit AIX and Linux:

```
export INFORMIXSERVER=hostname_1
```

Verwenden Sie auf Systemen mit Windows folgende Angabe:

```
set INFORMIXSERVER=hostname_1
```

ONCONFIG

Der Name der Informix-Serverkonfigurationsdatei. Verwenden Sie zum Beispiel auf Systemen mit AIX and Linux:

```
export ONCONFIG=onconfig.hostname_1
```

Verwenden Sie auf Systemen mit Windows folgende Angabe:

```
set ONCONFIG=onconfig.hostname_1
```

Informix-Switchloaddatei erstellen

Verwenden Sie zum Erstellen der Informix-Switchloaddatei die Beispieldatei xaswit.mak, die von IBM MQ zum Erstellen von Switchloaddateien für verschiedene Datenbankprodukte bereitgestellt wird.

Windows Auf Windows -Systemen finden Sie xaswit.mak im Verzeichnis `MQ_INSTALLATION_PATH\tools\c\samples\xa\m`. `MQ_INSTALLATION_PATH` steht für das übergeordnete Verzeichnis, in dem IBM MQ installiert ist. Verwenden Sie zum Erstellen der Informix-Switchloaddatei mit Microsoft Visual C++ folgenden Befehl:

```
nmake /f xaswit.mak infswit.dll
```

Die generierte Switchdatei wird in das Verzeichnis `C:\Programme\IBM\MQ\exits` gestellt.

Linux **UNIX** Sie finden xaswit.mak im Verzeichnis `MQ_INSTALLATION_PATH/samp/xatm`. `MQ_INSTALLATION_PATH` steht für das übergeordnete Verzeichnis, in dem IBM MQ installiert ist.

Bearbeiten Sie die Datei 'xaswit.mak' und *entfernen Sie die Kommentarzeichen* für die entsprechenden Zeilen der von Ihnen verwendeten Informix-Version. Führen Sie anschließend die Makedatei mit dem folgenden Befehl aus:

```
make -f xaswit.mak infswit
```

Die generierte 32-Bit-Switchloaddatei wird in das Verzeichnis `/var/mqm/exits` gestellt.

Die generierte 64-Bit-Switchloaddatei wird in das Verzeichnis `/var/mqm/exits64` gestellt.

Wenn Ihr System die 32-Bit-Kompilierung nicht unterstützt, verwenden Sie das 64-Bit-Ziel:

```
make -f xaswit.mak infswit64
```

Ressourcenmanager-Konfigurationsinformationen für Informix hinzufügen

Sie müssen die Konfigurationsinformationen für den Warteschlangenmanager ändern und die Beteiligung von Informix an globalen Arbeitseinheiten deklarieren. Die entsprechende Änderung der Konfigurationsinformationen für den Warteschlangenmanager wird im Abschnitt [„Konfigurationsinformationen zum Warteschlangenmanager hinzufügen“](#) auf Seite 63 ausführlicher beschrieben.

- Verwenden Sie unter Windows und Linux (x86- und x86-64-Plattformen) den IBM MQ Explorer. Geben Sie die Details zu der Switchloaddatei in der Eigenschaftenanzeige für den Warteschlangenmanager unter dem XA-Ressourcenmanager an.
- Geben Sie bei allen anderen Systemen die Details zu der Switchloaddatei in der XAResourceManager-Zeilengruppe in der Datei 'qm.ini' des Warteschlangenmanagers an.

[Abbildung 8](#) auf Seite 73 ist ein UNIX-Beispiel mit einem XAResourceManager-Eintrag, bei dem die zu koordinierende Datenbank die Bezeichnung mydbname trägt, wobei dieser Namen in XAOpenString angegeben wird:


```
XAResourceManager:  
Name=myinformix  
SwitchFile=infswit  
XAOpenString=DB=mydbname@myinformixserver\;USER=myuser\;PASSWD=mypasswd  
ThreadOfControl=THREAD
```

Abbildung 8. XAResourceManager-Beispielintrag für Informix auf UNIXen

Anmerkung: Standardmäßig wird mit der Beispieldatei 'xaswit.mak' auf UNIXen eine Switchloaddatei erstellt, die Informix-Threadbibliotheken verwendet. Bei Verwendung dieser Informix-Bibliotheken muss sichergestellt sein, dass 'ThreadOfControl' auf THREAD gesetzt ist. In Abbildung 8 auf Seite 73 ist in der XAResourceManager-Zeilengruppe in der Datei 'qm.ini' das Attribut 'ThreadOfControl' auf THREAD gesetzt. Bei Angabe von THREAD müssen Anwendungen unter Verwendung der Informix-Threadbibliotheken sowie der API-Threadbibliotheken von IBM MQ erstellt werden.

Im Attribut 'XAOpenString' muss der Datenbankname gefolgt vom Symbol @ und anschließend dem Informix-Servernamen angegeben sein.

Für die Verwendung der Informix-Bibliotheken ohne Threads muss sichergestellt sein, dass das Attribut 'ThreadOfControl' in der XAResourceManager-Zeilengruppe der Datei 'qm.ini' auf PROCESS gesetzt ist. Außerdem müssen in der Beispieldatei 'xaswit.mak' folgende Änderungen vorgenommen werden:

1. Die Kommentarzeichen für das Erstellen einer Switchloaddatei ohne Threads muss entfernt werden.
2. Das Erstellen der Switchloaddatei mit Threads muss auf Kommentar gesetzt werden.

Multi Sybase-Konfiguration

Sybase-Support- und -Konfigurationsinformationen

Führen Sie die folgenden Schritte aus:

1. Vergewissern Sie sich, dass Sie die Sybase-XA-Bibliotheken installiert haben, und installieren Sie dazu beispielsweise die XA-DTM-Option.
2. Prüfen Sie die Einstellungen der Umgebungsvariablen.
3. Aktivieren Sie die Sybase-XA-Unterstützung.
4. Erstellen Sie die Sybase-Switchloaddatei.
5. Fügen Sie Konfigurationsinformationen für den Ressourcenmanager hinzu.

Eine aktuelle Liste der von IBM MQ unterstützten Sybase-Versionen finden Sie unter [Systemvoraussetzungen für IBM MQ](#).

Einstellungen der Sybase-Umgebungsvariablen prüfen

Vergewissern Sie sich, dass Ihre Sybase-Umgebungsvariablen sowohl für Warteschlangenmanagerprozesse als auch in Ihren Anwendungsprozessen definiert sind. Legen Sie insbesondere immer die folgenden Umgebungsvariablen fest, bevor Sie den Warteschlangenmanager starten:

SYBASE

Linux **AIX** Die Speicherposition der Sybase-Produktinstallation. Verwenden Sie zum Beispiel auf Systemen mit AIX and Linux:

```
export SYBASE=/sybase
```

Windows Verwenden Sie auf Systemen mit Windows folgende Angabe:

```
set SYBASE=c:\sybase
```

SYBASE_OCS

Das Verzeichnis unter SYBASE, in dem Sie die Sybase-Clientdateien installiert haben.

Linux **AIX** Verwenden Sie zum Beispiel auf Systemen mit AIX and Linux:

```
export SYBASE_OCS=OCS-12_0
```

Windows Verwenden Sie auf Systemen mit Windows folgende Angabe:

```
set SYBASE_OCS=OCS-12_0
```

Sybase-XA-Unterstützung aktivieren

Definieren Sie in der Sybase -XA-Konfigurationsdatei `$$SYBASE/$SYBASE_OCS/xa_config` für jede Verbindung zu dem Sybase -Server, der aktualisiert wird, einen logischen Ressourcenmanager (LRM). Im Folgenden sehen Sie ein Beispiel für den Inhalt von `$$SYBASE/$SYBASE_OCS/xa_config`:

```
# The first line must always be a comment
[xa]
LRM=lrmname
server=servername
```

Sybase-Switchloaddatei erstellen

Verwenden Sie zum Erstellen der Sybase-Switchloaddatei die mit IBM MQ bereitgestellten Beispieldateien.

Windows Auf Windows-Systemen finden Sie `xaswit.mak` im Verzeichnis `C:\Programme\IBM\MQ\tools\c\samples\xatm`. Verwenden Sie zum Erstellen der Sybase-Switchloaddatei mit Microsoft Visual C++ folgenden Befehl:

```
nmake /f xaswit.mak sybswit.dll
```

Die generierte Switchdatei wird in das Verzeichnis `C:\Programme\IBM\MQ\exits` gestellt.

Linux **UNIX** Die Datei `xaswit.mak` ist im Verzeichnis `MQ_INSTALLATION_PATH/samp/xatm` zu finden. `MQ_INSTALLATION_PATH` steht für das übergeordnete Verzeichnis, in dem IBM MQ installiert ist.

Bearbeiten Sie `xaswit.mak` für *Kommentarzeichen entfernen*, die Linien, die angemessen sind für die Version von Sybase das Sie verwenden Führen Sie anschließend die Makedatei mit dem folgenden Befehl aus:

```
make -f xaswit.mak sybswit
```

Die generierte 32-Bit-Switchloaddatei wird in `/var/mqm/exits` gespeichert.

Die generierte 64-Bit-Switchloaddatei wird in `/var/mqm/exits64` gespeichert.

Wenn Ihr System die 32-Bit-Kompilierung nicht unterstützt, verwenden Sie das 64-Bit-Ziel:

```
make -f xaswit.mak sybswit64
```

Anmerkung: **AIX** Unter AIX wurde die Beispiel-Makefile wie im folgenden Beispiel geändert, sodass Sie je nachdem, ob Sie Sybase 15 ESD#5 oder höher verwenden oder eine frühere Version von Sybase, einen anderen Wert für SYBLINKFLAG64 auswählen können.

```
SYBLINKFLAGS32=-brt1
# The following line is for Sybase 15
#SYBLINKFLAGS64=-brt1
# The following line is for Sybase 16
SYBLINKFLAGS64=-bstatic -bdynamic
```

Die einzige Änderung, die Sie an der Makefile vornehmen müssen, besteht darin, sicherzustellen, dass nur einer der Werte für SYBLINKFLAGS64 nicht auf Kommentar gesetzt ist. Der Standardwert ist Sybase 16; dies ist der Wert, der für 15 #ESD5 und höher verwendet wird.

Jede XA-Switchdatei, die erstellt wird, ist mit dem spezifischen Release von Sybase verknüpft und darf nicht auf andere Plattformen verschoben werden.

Wenn die Sybase-Version geändert wird, sollte die XA-Switchdatei neu erstellt werden.

Ressourcenmanager-Konfigurationsinformationen für Sybase hinzufügen

Sie müssen die Konfigurationsinformationen für den Warteschlangenmanager ändern und die Beteiligung von Sybase an globalen Arbeitseinheiten deklarieren. Die Änderung der Konfigurationsinformationen wird im Abschnitt „[Konfigurationsinformationen zum Warteschlangenmanager hinzufügen](#)“ auf Seite 63 ausführlicher beschrieben.

- **Windows** **Linux** Verwenden Sie unter Windows und Linux (x86- und x86-64-Plattformen) IBM MQ Explorer. Geben Sie die Details zu der Switchloaddatei in der Eigenschaftsanzeige für den Warteschlangenmanager unter dem XA-Ressourcenmanager an.
- Geben Sie auf allen anderen Systemen die Details der Switchloaddatei in der XAResourceManager-Zeilengruppe in der `qm.ini`-Datei des Warteschlangenmanagers an.

Linux **AIX** Das folgende Beispiel zeigt einen Beispiel-XAResourceManager-Eintrag für Sybase unter AIX and Linux, der die Datenbank verwendet, die der LRM-Definition "lrmname" in der Sybase -XA-Konfigurationsdatei `$SYBASE/$SYBASE_OCS/xa_configzugeordnet` ist. Geben Sie auch einen Protokolldateinamen an, wenn die XA-Funktionsaufrufe protokolliert werden sollen:

```
XAResourceManager:
Name=mysybase
SwitchFile=sybswit
XAOpenString=-User -Ppassword -Nlrmname -L/tmp/sybase.log -Txa
ThreadOfControl=THREAD
```

Multithread-Programme mit Sybase verwenden

Wenn Sie Multithread-Programme mit globalen IBM MQ-Arbeitseinheiten verwenden, die Sybase-Aktualisierungen beinhalten, müssen Sie den Wert `THREAD` für den Parameter 'ThreadOfControl' verwenden. Achten Sie außerdem darauf, dass Sie Ihr Programm (und die Switchloaddatei) mit den threadsicheren Sybase-Bibliotheken (`_r`-Versionen) verknüpfen. Die Verwendung des Werts `THREAD` für den Parameter 'ThreadOfControl' ist im vorherigen Beispiel dargestellt.

Multi Mehrere Datenbankkonfigurationen

Wenn Sie den Warteschlangenmanager so konfigurieren möchten, dass Aktualisierungen mehrerer Datenbanken in globale Arbeitseinheiten einbezogen werden können, fügen Sie für jede Datenbank eine XAResourceManager-Zeilengruppe hinzu.

Falls die Datenbanken alle von demselben Datenbankmanager verwaltet werden, wird in jeder Zeilengruppe eine andere Datenbank definiert. Zwar ist in jeder Zeilengruppe dieselbe *Switchdatei* angegeben, die Inhalte von *XAOpenString* sind jedoch unterschiedlich, da hier der Name der zu aktualisierenden Da-

tenbank angegeben ist. Mit den in *Abbildung 9* auf Seite 76 gezeigten Zeilengruppen wird beispielsweise der Warteschlangenmanager mit den Db2-Datenbanken *MQBankDB* und *MQFeeDB* auf Systemen mit AIX and Linux konfiguriert.

Wichtig: Es können nicht mehrere Zeilengruppen vorhanden sein, die auf die gleiche Datenbank verweisen. Diese Konfiguration kann nicht funktionieren und wird bei einem Konfigurationsversuch fehlschlagen.

Sie erhalten Fehler im Format `when the MQ code makes its second xa_open call in any process in this environment, the database software fails the second xa_open with a -5 error, XAER_INVAL.`

```
XAResourceManager:  
Name=DB2 MQBankDB  
SwitchFile=db2swit  
XAOpenString=MQBankDB
```

```
XAResourceManager:  
Name=DB2 MQFeeDB  
SwitchFile=db2swit  
XAOpenString=MQFeeDB
```

Abbildung 9. XAResourceManager-Beispieleinträge für mehrere Db2-Datenbanken

Falls die zu aktualisierenden Datenbanken von verschiedenen Datenbankmanagern aktualisiert werden, fügen Sie jeweils eine XAResourceManager-Zeilengruppe hinzu. In diesem Fall ist in jeder Zeilengruppe eine andere *Switchdatei* angegeben. Wenn *MQFeeDB* beispielsweise von Oracle anstelle von Db2 verwaltet wird, verwenden Sie die folgenden Zeilengruppen auf Systemen mit AIX and Linux:

```
XAResourceManager:  
Name=DB2 MQBankDB  
SwitchFile=db2swit  
XAOpenString=MQBankDB
```

```
XAResourceManager:  
Name=Oracle MQFeeDB  
SwitchFile=oraswit  
XAOpenString=Oracle_XA+Acc=P/myuser/mypassword+SesTm=35+LogDir=/tmp/ora.log+DB=MQFeeDB
```

Abbildung 10. XAResourceManager-Beispieleinträge für eine Db2- und Oracle-Datenbank

Im Prinzip können beliebig viele Datenbankinstanzen mit einem einzelnen Warteschlangenmanager konfiguriert werden.

Anmerkung: Informationen zur Unterstützung für die Einbeziehung von Informix-Datenbanken in mehrere Datenbankaktualisierungen innerhalb globaler Arbeitseinheiten finden Sie in der Readme-Datei zu dem Produkt.

Multi Sicherheitsaspekte

Aspekte der Datenbankausführung unter dem XA-Modell.

Die folgenden Informationen sollen nur zur Orientierung dienen. Stellen Sie in allen Fällen anhand der mit dem Datenbankmanager bereitgestellten Dokumentation fest, wie sich die Ausführung Ihrer Datenbank unter dem XA-Modell auf die Sicherheit auswirkt.

In Anwendungsprozessen wird der Beginn einer globalen Arbeitseinheit mit dem Verb `MQBEGIN` bezeichnet. Beim ersten von einer Anwendung ausgegebenen Aufruf `MQBEGIN` wird durch Aufruf des entsprechenden Clientbibliothekcodes am `xa_open`-Einstiegspunkt eine Verbindung zu allen beteiligten Datenbanken hergestellt. Alle Datenbankmanager bieten einen Mechanismus zur Angabe einer Benutzer-ID und eines Kennworts in ihrer XA-Zeichenfolge zum Öffnen (`XAOpenString`). Dies ist der einzige Zeitpunkt, an dem Authentifizierungsdaten fließen.

Beachten Sie, dass Fastpath-Anwendungen auf den Plattformen AIX and Linux mit der effektiven Benutzer-ID 'mqm' ausgeführt werden müssen, wenn MQI-Aufrufe erfolgen.

Multi Überlegungen zur Vorgehensweise bei Verlust des Kontakts zum XA-Ressourcenmanager

Der Warteschlangenmanager akzeptiert, wenn Datenbankmanager nicht verfügbar sind. Das heißt, Sie können den Warteschlangenmanager unabhängig vom Datenbankserver starten und stoppen. Sobald der Kontakt wiederhergestellt ist, werden Warteschlangenmanager und Datenbank resynchronisiert. Sie können unbestätigte Arbeitseinheiten auch manuell mit dem Befehl `rsvmqtrn` auflösen.

Für normale Operationen ist nach der Konfiguration nur wenig Verwaltung notwendig. Der Verwaltungsjob ist einfacher, da es der Warteschlangenmanager akzeptiert, wenn Datenbankmanager nicht verfügbar sind. Dies bedeutet insbesondere Folgendes:

- Der Warteschlangenmanager kann jederzeit gestartet werden, ohne zuvor die einzelnen Datenbankmanager starten zu müssen.
- Wenn einer der Datenbankmanager nicht mehr verfügbar ist, muss der Warteschlangenmanager nicht gestoppt und erneut gestartet werden.

Somit kann der Warteschlangenmanager unabhängig vom Datenbankserver gestartet und gestoppt werden.

Wenn der Kontakt zwischen dem Warteschlangenmanager und einer Datenbank verloren geht, muss eine Resynchronisation erfolgen, wenn beide wieder verfügbar sind. Resynchronisation bezeichnet den Prozess, mit dem unbestätigte Arbeitseinheiten, die diese Datenbank einbeziehen, abgeschlossen werden. Im Allgemeinen geschieht dies automatisch ohne Benutzereingriff. Der Warteschlangenmanager fragt die Datenbank nach einer Liste der unbestätigten Arbeitseinheiten. Anschließend weist er die Datenbank an, die einzelnen unbestätigten Arbeitseinheiten festzuschreiben oder rückgängig zu machen.

Beim Start wird der Warteschlangenmanager mit den einzelnen Datenbanken resynchronisiert. Wenn eine einzelne Datenbank nicht mehr verfügbar ist, muss nur diese resynchronisiert werden, wenn der Warteschlangenmanager merkt, dass sie wieder verfügbar ist.

Der Kontakt zwischen dem Warteschlangenmanager und einer zuvor nicht verfügbaren Datenbank wird automatisch wiederhergestellt, wenn mit `MQBEGIN` neue globale Arbeitseinheiten gestartet werden. Hierfür wird die Funktion `'xa_open'` in der Datenbankclientbibliothek aufgerufen. Falls dieser Aufruf `'xa_open'` fehlschlägt, gibt `MQBEGIN` den Beendigungscode `MQCC_WARNING` und den Ursachencode `MQRC_PARTICIPANT_NOT_AVAILABLE` zurück. Sie können den Aufruf `MQBEGIN` zu einem späteren Zeitpunkt wiederholen.

Versuchen Sie den Vorgang nicht mehrmals mit einer globalen Arbeitseinheit, die Aktualisierungen einer Datenbank beinhaltet, welche während des Aufrufs `MQBEGIN` einen Fehler angezeigt hat. Es wird keine Verbindung zu dieser Datenbank geben, über die Aktualisierungen erfolgen können. Die einzige Möglichkeit besteht darin, das Programm zu beenden oder den Aufruf `MQBEGIN` von Zeit zu Zeit zu wiederholen, in der Hoffnung, dass die Datenbank möglicherweise wieder verfügbar wird.

Alternativ dazu können Sie auch mit dem Befehl `rsvmqtrn` alle unbestätigten Arbeitseinheiten explizit auflösen.

Multi Unbestätigte Arbeitseinheiten

Möglicherweise gibt es in einer Datenbank unbestätigte Arbeitseinheiten, wenn der Kontakt zum Warteschlangenmanager unterbrochen wird, nachdem der Datenbankmanager aufgefordert wurde, Vorbereitungen zu treffen. Bis der Datenbankserver vom Warteschlangenmanager das Ergebnis (Festschreibung oder Rollback) erhält, muss er die Datenbanksperren im Zusammenhang mit den Aktualisierungen beibehalten.

Da aufgrund dieser Sperren andere Anwendungen Datenbanksätze weder aktualisieren noch lesen können, muss so bald wie möglich eine Resynchronisation erfolgen.

Falls Sie die automatische Resynchronisation des Warteschlangenmanagers mit der Datenbank aus irgendeinem Grund nicht abwarten können, können Sie die Datenbankaktualisierungen mithilfe von Datenbankmanagerfunktionen manuell festschreiben bzw. rückgängig machen. In *X/Open Distributed Transaction Processing: The XA Specification* wird in diesem Zusammenhang auch von einer *heuristischen* Entscheidung gesprochen. Nutzen Sie diese Möglichkeit nur als letzten Abhilfemaßnahme, da dabei die Datenintegrität beeinträchtigt werden kann; so könnten Sie beispielsweise versehentlich die Datenbankaktualisierungen rückgängig machen, wenn alle anderen Beteiligten ihre Aktualisierungen festgeschrieben haben.

Weitaus besser ist es, den Warteschlangenmanager erneut zu starten oder nach dem Neustart der Datenbank mit dem Befehl `rsvmqtrn` eine automatische Resynchronisation einzuleiten.

Multi

Ausstehende Arbeitseinheiten mit dem Befehl 'dspmqtrn' anzeigen

Mit dem Befehl `dspmqtrn` in Verbindung mit dem Parameter `-i` können Sie unbestätigte Transaktionen internen Ursprungs anzeigen.

Wenn ein Datenbankmanager nicht verfügbar ist, können Sie mit dem Befehl `dspmqtrn` den Status ausstehender globaler Arbeitseinheiten überprüfen, die diese Datenbank mit einbeziehen.

Mit dem Befehl `dspmqtrn` werden nur Arbeitseinheiten mit mindestens einem unbestätigten Beteiligten angezeigt. Die Beteiligten warten auf die Entscheidung des Warteschlangenmanagers über die Festschreibung der das Rollback der vorbereiteten Aktualisierungen.

Für jede dieser globalen Arbeitseinheiten wird in der Ausgabe von `dspmqtrn` der Status der einzelnen Beteiligten angezeigt. Falls die Arbeitseinheit die Ressourcen eines bestimmten Ressourcenmanagers nicht aktualisiert hat, wird sie nicht angezeigt.

Bezüglich einer unbestätigten Arbeitseinheit ist angegeben, dass ein Ressourcenmanager eine der folgenden Aktionen durchgeführt hat:

Vorbereitet

Der Ressourcenmanager ist bereit, die Aktualisierungen festzuschreiben.

Festgeschrieben

Der Ressourcenmanager hat die Aktualisierungen festgeschrieben.

Rückgängig gemacht

Der Ressourcenmanager hat die Aktualisierungen rückgängig gemacht.

Teilgenommen

Der Ressourcenmanager ist Beteiligter, hat jedoch die Aktualisierungen weder vorbereitet, noch festgeschrieben oder rückgängig gemacht.

Beim Neustart fragt der Warteschlangenmanager jede Datenbank mit einer XAResourceManager-Zeilengruppe nach einer Liste ihrer unbestätigten globalen Arbeitseinheiten. Falls die Datenbank nicht erneut gestartet wurde oder aus anderen Gründen nicht verfügbar ist, kann der Warteschlangenmanager die endgültigen Ergebnisse für die betreffenden Arbeitseinheiten noch nicht an die Datenbank übergeben. Das Ergebnis der unbestätigten Arbeitseinheiten wird der Datenbank bei der nächsten Gelegenheit übergeben, wenn die Datenbank wieder verfügbar ist.

In diesem Fall wird für den Datenbankmanager der Status `Vorbereitet` gemeldet, bis eine Resynchronisation stattgefunden hat.

Wenn vom Befehl `dspmqtrn` eine unbestätigte Arbeitseinheit angezeigt wird, werden zunächst alle möglicherweise beteiligten Ressourcenmanager aufgeführt. Diesen wird eine eindeutige Kennung, `RMId`, zugeordnet, die anstelle des *Namens* der Ressourcenmanager verwendet wird, wenn ihr Status in Bezug auf eine unbestätigte Arbeitseinheit gemeldet wird.

Im Abschnitt [Beispielausgabe für 'dspmqtrn'](#) ist das Ergebnis der folgenden Befehlsausgabe dargestellt:

```
dspmqtrn -m MY_QMGR
```

```

AMQ7107: Resource manager 0 is MQSeries.
AMQ7107: Resource manager 1 is DB2 MQBankDB.
AMQ7107: Resource manager 2 is DB2 MQFeeDB.

AMQ7056: Transaction number 0,1.
      XID: formatID 5067085, gtrid_length 12, bqual_length 4
           gtrid [3291A5060000201374657374]
           bqual [00000001]
AMQ7105: Resource manager 0 has committed.
AMQ7104: Resource manager 1 has prepared.
AMQ7104: Resource manager 2 has prepared.

```

Dabei ist *Transaction number* die ID der Transaktion, die mit dem Befehl **rsvmqtrn** verwendet werden kann. Weitere Informationen finden Sie im Abschnitt **AMQ7xxx: IBM MQ-Produktnachrichten**. Die Variablen *XID* sind Teil der *X/Open XA-Spezifikation*; aktuelle Informationen zu dieser Spezifikation finden Sie unter <https://publications.opengroup.org/c193>.

Abbildung 11. Beispielausgabe für 'dspmqtrn'

Aus der Ausgabe im Abschnitt [Beispielausgabe für 'dspmqtrn'](#) geht hervor, dass dem Warteschlangenmanager drei Ressourcenmanager zugeordnet sind. Bei dem ersten Ressourcenmanager 0 handelt es sich um den Warteschlangenmanager selbst. Die beiden anderen Ressourcenmanagerinstanzen sind die MQBankDB -und MQFeeDB Db2 -Datenbanken.

In dem Beispiel ist eine einzelne unbestätigte Arbeitseinheit dargestellt. Für alle drei Ressourcenmanager wird eine Nachricht ausgegeben; dies bedeutet, dass sowohl am Warteschlangenmanager als auch an den beiden Db2-Datenbanken innerhalb der Arbeitseinheit Aktualisierungen vorgenommen wurden.

Die Aktualisierungen am Warteschlangenmanager (Ressourcenmanager 0) wurden festgeschrieben. Die Aktualisierungen an den Db2-Datenbanken befinden sich im Status *Vorbereitet*, Db2 war also nicht mehr verfügbar, als der Aufruf zur Festschreibung der Aktualisierungen der Datenbanken MQBankDB und MQFeeDB erfolgt ist.

Die unbestätigte Arbeitseinheit verfügt über eine externe Kennung, eine sogenannte *XID (Transaktions-ID)*. Hierbei handelt es sich um einen vom Warteschlangenmanager an Db2 vergebenen Datenteil zur Kennung seines Abschnitts der globalen Arbeitseinheit.

Weitere Informationen finden Sie in [dspmqtrn](#).

Multi

Ausstehende Arbeitseinheiten mit dem Befehl 'rsvmqtrn' auflösen

Ausstehende Arbeitseinheiten werden bei der Resynchronisation von Db2 und dem Warteschlangenmanager abgeschlossen.

In der Ausgabe in [Abbildung 11 auf Seite 79](#) ist eine einzelne unbestätigte Arbeitseinheit angegeben, bei der die Festschreibungsentscheidung noch an beide Db2-Datenbanken übermittelt werden muss.

Damit diese Arbeitseinheit abgeschlossen wird, müssen der Warteschlangenmanager und Db2 resynchronisiert werden, sobald Db2 wieder verfügbar wird. Der Warteschlangenmanager nutzt den Start neuer Arbeitseinheiten als Gelegenheit, den Kontakt zu Db2 wiederherzustellen. Alternativ dazu können Sie den Warteschlangenmanager auch zu einer expliziten Resynchronisation mit dem Befehl **rsvmqtrn** anweisen.

Führen Sie diese Maßnahme möglichst bald nach dem Neustart von Db2 durch, damit alle Datenbanksperren im Zusammenhang mit der unbestätigten Arbeitseinheit so schnell wie möglich freigegeben werden. Verwenden Sie die Option '-a', um den Warteschlangenmanager aufzufordern, alle unbestätigten Arbeitseinheiten aufzulösen. Im folgenden Beispiel wurde Db2 erneut gestartet, damit der Warteschlangenmanager die unbestätigte Arbeitseinheit auflösen kann:

```

> rsvmqtrn -m MY_QMGR -a
Any in-doubt transactions have been resolved.

```


Multi **Heterogene Ergebnisse und Fehler**

Auch wenn der Warteschlangenmanager ein Protokoll für zweiphasige Festschreibung verwendet, lässt sich nicht völlig ausschließen, dass Arbeitseinheiten mit heterogenen Ergebnissen abgeschlossen werden. Hierzu kommt es, wenn einige Beteiligte ihre Aktualisierungen festschreiben, während andere sie zurücksetzen.

Wenn Arbeitseinheiten mit heterogenen Ergebnissen abgeschlossen werden, hat dies schwerwiegende Konsequenzen, da sich gemeinsam genutzte Ressourcen, die als einzelne Arbeitseinheit hätten aktualisiert werden müssen, nicht mehr in einem konsistenten Zustand befinden.

Zu heterogenen Ergebnissen kommt es vor allem, wenn heuristische Entscheidungen zu Arbeitseinheiten getroffen werden und der Warteschlangenmanager unbestätigte Arbeitseinheiten nicht selbst auflösen kann. Über diese Entscheidungen hat der Warteschlangenmanager keine Kontrolle.

Wenn der Warteschlangenmanager heterogene Ergebnisse feststellt, generiert er FFST-Informationen und dokumentiert den Fehler mit einer der beiden folgenden Nachrichten in seinen Fehlerprotokollen:

- Falls ein Datenbankmanager anstelle einer Festschreibung ein Rollback vornimmt:

```
AMQ7606 A transaction has been committed but one or more resource
managers have rolled back.
```

- Falls ein Datenbankmanager anstelle eines Rollbacks eine Festschreibung durchführt:

```
AMQ7607 A transaction has been rolled back but one or more resource
managers have committed.
```

In weiteren Nachrichten ist angegeben, welche Datenbanken aufgrund der heuristischen Entscheidung beschädigt sind. Es ist nun Ihre Aufgabe, die Konsistenz der betreffenden Datenbanken lokal wiederherzustellen. Hierbei handelt es sich um ein kompliziertes Verfahren, bei dem die fälschlicherweise festgeschriebene bzw. zurückgesetzte Aktualisierung zunächst isoliert und die Datenbankänderung anschließend manuell rückgängig gemacht bzw. wiederhergestellt werden muss.

Multi **Konfigurationsinformationen ändern**

Ändern Sie die Ressourcenmanager-Konfigurationsinformationen nicht mehr, nachdem der Warteschlangenmanager erfolgreich mit der Koordination globaler Arbeitseinheiten begonnen hat.

Wenn Sie die Konfigurationsinformationen ändern müssen, können Sie dies jederzeit tun, die Änderungen treten allerdings erst nach dem Neustart des Warteschlangenmanagers in Kraft.

Wenn Sie die Ressourcenmanager-Konfigurationsinformationen für eine Datenbank entfernen, nehmen Sie dem Warteschlangenmanager damit praktisch die Möglichkeit, Kontakt zu dem betreffenden Datenbankmanager herzustellen.

Ändern Sie **nie** das Attribut *Name* in Ihren Ressourcenmanager-Konfigurationsinformationen. Mit diesem Attribut wird die betreffende Datenbankmanagerinstanz dem Warteschlangenmanager gegenüber eindeutig angegeben. Wenn Sie diese eindeutige Kennung ändern, geht der Warteschlangenmanager davon aus, dass die Datenbank entfernt und eine komplett neue Instanz hinzugefügt wurde. Der Warteschlangenmanager ordnet ausstehende Arbeitseinheiten weiterhin dem alten *Namen* zu, sodass die Datenbank möglicherweise in einem unbestätigten Zustand bleibt.

Multi **Datenbankmanagerinstanzen entfernen**

Wenn eine Datenbank permanent aus Ihrer Konfiguration entfernt werden muss, vergewissern Sie sich, dass die Datenbank nicht unbestätigt ist, bevor Sie den Warteschlangenmanager erneut starten.

Von den Datenbankprodukten werden Befehle zur Auflistung unbestätigter Transaktionen zur Verfügung gestellt. Ermöglichen Sie im Falle unbestätigter Transaktionen dem Warteschlangenmanager zunächst eine Resynchronisation mit der Datenbank. Starten Sie hierfür den Warteschlangenmanager. Mit dem Befehl **rsvmqtrn** oder dem datenbankeigenen Befehl zur Anzeige unbestätigter Arbeitseinheiten können Sie prüfen, ob die Resynchronisation tatsächlich erfolgt ist. Wenn Sie sicher sind, dass die Resynchronisation

stattgefunden hat, beenden Sie den Warteschlangenmanager und entfernen Sie die Konfigurationsinformationen der Datenbank.

Wenn Sie sich nicht an diese Vorgehensweise halten, merkt sich der Warteschlangenmanager alle unbestätigten Arbeitseinheiten, die die betreffende Datenbank einbeziehen. Es wird dann jedes Mal, wenn der Warteschlangenmanager erneut gestartet wird, eine Warnung, AMQ7623, ausgegeben. Wenn Sie diese Datenbank überhaupt nicht mehr mit dem Warteschlangenmanager konfigurieren möchten, verwenden Sie im Befehl **rsvmqtrn** die Option '-r' und weisen Sie den Warteschlangenmanager damit an, die Beteiligung der Datenbank an den unbestätigten Transaktionen zu übergehen. Der Warteschlangenmanager übergeht solche Transaktionen nur, wenn die unbestätigten Transaktionen mit allen Beteiligten abgeschlossen wurden.

Es kann vorkommen, dass Ressourcenmanager-Konfigurationsinformationen vorübergehend entfernt werden müssen. Auf Systemen mit AIX and Linux wird hierfür am besten die betreffende Zeilengruppe auf Kommentar gesetzt, sodass sie später leicht wiederhergestellt werden kann. Vielleicht entscheiden Sie sich für diese Vorgehensweise, wenn Sie immer Fehler erhalten, wenn der Warteschlangenmanager einen Kontakt zu einer bestimmten Datenbank bzw. einem Datenbankmanager herstellt. Wenn die betreffenden Ressourcenmanager-Konfigurationsinformationen vorübergehend entfernt werden, kann der Warteschlangenmanager die globalen Arbeitseinheiten für alle anderen Beteiligten starten. Nachfolgend sehen Sie ein Beispiel für eine auskommentierte XAResourceManager-Zeilengruppe:

```
# This database has been temporarily removed
#XAResourceManager:
# Name=mydb2
# SwitchFile=db2swit
# XAOpenString=mydbname,myuser,mypassword,toc=t
# ThreadOfControl=THREAD
```

Abbildung 12. Auskommentierte XAResourceManager-Zeilengruppe auf Systemen mit AIX and Linux

Verwenden Sie auf Windows-Systemen IBM MQ Explorer zum Löschen der Informationen über die Datenbankmanagerinstanz. Achten Sie ganz besonders darauf, bei der Wiederherstellung den richtigen Namen in das Feld *Name* einzugeben. Falls Sie den Namen falsch eingeben, kann es wie im Abschnitt „Konfigurationsinformationen ändern“ auf Seite 80 beschrieben zu Fehlern bezüglich eines unbestätigten Status kommen.

Multi **Dynamische XA-Registrierung**

Die XA-Spezifikation bietet eine Möglichkeit, die Anzahl der Aufrufe `xa_*` eines Transaktionsmanagers an einen Ressourcenmanager zu verringern. Diese Optimierung wird als *dynamische Registrierung* bezeichnet.

Db2 unterstützt die dynamische Registrierung. Auch andere Datenbanken bieten möglicherweise Unterstützung hierfür; weitere Informationen finden Sie in der Dokumentation zu Ihrem Datenbankprodukt.

Warum ist die Optimierung durch dynamische Registrierung sinnvoll? Es kann sein, dass in Ihrer Anwendung einige globale Arbeitseinheiten Aktualisierungen von Datenbanktabellen umfassen, andere nicht. Wenn keine permanenten Aktualisierungen der Tabellen einer Datenbank vorgenommen wurden, muss die betreffende Datenbank nicht in das Festschreibungsprotokoll während MQCMIT aufgenommen werden.

Unabhängig davon, ob Ihre Datenbank die dynamische Registrierung unterstützt, ruft Ihre Anwendung `xa_open` während des ersten MQBEGIN-Aufrufs in einer IBM MQ -Verbindung auf. Im nachfolgenden Aufruf MQDISC ruft die Anwendung `xa_close` auf. Das Muster weiterer XA-Aufrufe hängt davon ab, ob die Datenbank die dynamische Registrierung unterstützt:

Falls Ihre Datenbank die dynamische Registrierung nicht unterstützt...

Jede globale Arbeitseinheit umfasst mehrere XA-Funktionsaufrufe, die von IBM MQ-Code an die Datenbankclientbibliothek erfolgen, wobei es keine Rolle spielt, ob Sie die Tabellen der betreffenden

Datenbank innerhalb Ihrer Arbeitseinheit permanent aktualisiert haben. Hierzu gehören folgende Aufrufe:

- `xa_start` und `xa_end` aus dem Anwendungsprozess. Mit diesen Aufrufen werden Anfang und Ende einer globalen Arbeitseinheit deklariert.
- `xa_prepare`, `xa_commit` und `xa_rollback` aus dem Warteschlangenmanageragentenprozess 'amqzlaa0'. Mit diesen Aufrufen wird das Ergebnis der globalen Arbeitseinheit, also die Entscheidung über Festschreibung oder Rollback übermittelt.

Darüber hinaus ruft der Warteschlangenmanageragentenprozess beim ersten Aufruf `MQBEGIN` auch `xa_open` auf.

Falls Ihre Datenbank die dynamische Registrierung unterstützt...

Der IBM MQ-Code führt nur die XA-Funktionsaufrufe durch, die nötig sind. Bei einer globalen Arbeitseinheit, die **keine** permanenten Aktualisierungen von Datenbankressourcen einbezieht, erfolgen **keine** XA-Aufrufe an die Datenbank. Bei einer globalen Arbeitseinheit, die solche permanenten Aktualisierungen **einbezieht**, werden folgende Aufrufe durchgeführt:

- `xa_end` aus dem Anwendungsprozess, um das Ende der globalen Arbeitseinheit zu deklarieren.
- `xa_prepare`, `xa_commit` und `xa_rollback` aus dem Warteschlangenmanageragentenprozess 'amqzlaa0'. Mit diesen Aufrufen wird das Ergebnis der globalen Arbeitseinheit, also die Entscheidung über Festschreibung oder Rollback übermittelt.

Damit die dynamische Registrierung richtig funktioniert, muss die Datenbank die Möglichkeit haben, IBM MQ nach einer permanenten Aktualisierung mitzuteilen, dass sie in die aktuelle globale Arbeitseinheit einbezogen werden möchte. IBM MQ stellt zu diesem Zweck die Funktion `ax_reg` bereit.

Der in Ihrem Anwendungsprozess ausgeführte Client-Code der Datenbank findet die Funktion `ax_reg` und ruft sie auf, um die Tatsache, dass innerhalb der globalen Arbeitseinheit eine permanente Aktion erfolgt ist, *dynamisch zu registrieren*. Als Antwort auf den Aufruf `ax_reg` zeichnet IBM MQ die Teilnahme der Datenbank auf. Falls es sich um den ersten `ax_reg`-Aufruf auf der betreffenden IBM MQ-Verbindung handelt, ruft der Warteschlangenmanageragentenprozess `xa_open` auf.

Dieser Aufruf `ax_reg` wird vom Code für den Datenbankclient durchgeführt, wenn der Code in Ihrem Prozess ausgeführt wird, also beispielsweise während eines Aufrufs `SQL UPDATE` oder eines anderen entsprechenden Aufrufs in der Client-API der Datenbank.

Multi Fehlerbedingungen

Bei der dynamischen XA-Registrierung kann es zu einem irreführenden Fehler im Warteschlangenmanager kommen.

Ein allgemeines Beispiel hierfür: wenn Sie vergessen, die Datenbankumgebungsvariablen vor dem Start des Warteschlangenmanagers ordnungsgemäß zu definieren, schlagen die Aufrufe `xa_open` des Warteschlangenmanagers fehl. Es können keine globalen Arbeitseinheiten verwendet werden.

Um dies zu vermeiden, sollten Sie sich vergewissern, dass die relevanten Umgebungsvariablen definiert sind, bevor Sie den Warteschlangenmanager starten. Beachten Sie die Informationen in der Dokumentation zu Ihrem Datenbankprodukt sowie die Ratschläge in den Abschnitten „[Db2 konfigurieren](#)“ auf Seite 65, „[Oracle konfigurieren](#)“ auf Seite 68 und „[Sybase-Konfiguration](#)“ auf Seite 73.

Bei allen Datenbankprodukten ruft der Warteschlangenmanager `xa_open` einmal beim Warteschlangenmanagerstart im Rahmen der Wiederherstellungssitzung auf (wie im Abschnitt „[Überlegungen zur Vorgehensweise bei Verlust des Kontakts zum XA-Ressourcenmanager](#)“ auf Seite 77 erläutert). Dieser Aufruf `xa_open` schlägt fehl, wenn Sie die Datenbankumgebungsvariablen nicht ordnungsgemäß definieren, der Warteschlangenmanager kann jedoch trotzdem gestartet werden. Dies liegt daran, dass mit demselben `xa_open`-Fehlercode von der Datenbankclientbibliothek angegeben wird, dass der Datenbankserver nicht verfügbar ist. In IBM MQ wird dies nicht als schwerwiegender Fehler behandelt, da es möglich sein muss, den Warteschlangenmanager zu starten, um die Verarbeitung von Daten außerhalb der globalen Arbeitseinheiten, die diese Datenbank einbeziehen, fortzusetzen.

Nachfolgende Aufrufe von `xa_open` werden vom Warteschlangenmanager während des ersten `MQBEGIN` auf einer IBM MQ-Verbindung (wenn die dynamische Registrierung nicht genutzt wird) oder während

eines Aufrufs der von IBM MQ bereitgestellten Funktion `ax_reg` durch den Datenbankclientcode (bei Nutzung der dynamischen Registrierung) durchgeführt.

Das **Timing** der Fehlerbedingungen (bzw. gelegentlich der FFST-Berichte) hängt davon ab, ob dynamische Registrierung genutzt wird:

- Bei Verwendung der dynamischen Registrierung könnte es sein, dass Ihr Aufruf `MQBEGIN` erfolgreich ist, der Datenbankaufruf `SQL UPDATE` (oder ähnlich) schlägt jedoch fehl.
- Wenn keine dynamische Registrierung verwendet wird, schlägt der Aufruf `MQBEGIN` fehl.

Vergewissern Sie sich, dass Ihre Umgebungsvariablen in Ihren Anwendungs- und Warteschlangenmanagerprozessen korrekt definiert sind.

Übersicht über die XA-Aufrufe

Hier finden Sie eine Liste der Aufrufe der XA-Funktionen in einer Datenbankclientbibliothek, die infolge der verschiedenen MQI-Aufrufe zur Steuerung globaler Arbeitseinheiten durchgeführt werden. Hierbei handelt es sich nicht um eine vollständige Beschreibung des in der XA-Spezifikation beschriebenen Protokolls, sondern um eine kurze Übersicht.

Beachten Sie, dass die Aufrufe `xa_start` und `xa_end` immer vom IBM MQ-Code im Anwendungsprozess aus erfolgen, während `xa_prepare`, `xa_commit` und `xa_rollback` immer vom Warteschlangenmanageragentenprozess 'amqzlaa0' aufgerufen werden.

Die in dieser Tabelle aufgeführten Aufrufe `xa_open` und `xa_close` erfolgen alle vom Anwendungsprozess aus. Vom Warteschlangenmanageragentenprozess wird `xa_open` unter den im Abschnitt „Fehlerbedingungen“ auf Seite 82 beschriebenen Bedingungen aufgerufen.

Tabelle 4. Übersicht über die XA-Funktionsaufrufe		
MQI-Aufruf	XA-Aufrufe mit dynamischer Registrierung	XA-Aufrufe ohne dynamische Registrierung
Erster Aufruf <code>MQBEGIN</code>	<code>xa_open</code>	<code>xa_open</code> <code>xa_start</code>
Nachfolgende Aufrufe <code>MQBEGIN</code>	Keine XA-Aufrufe	<code>xa_start</code>
<code>MQCMIT</code> (ohne Aufruf von <code>ax_reg</code> während der aktuellen globalen Arbeitseinheit)	Keine XA-Aufrufe	<code>xa_end</code> <code>xa_prepare</code> <code>xa_commit</code> <code>xa_rollback</code>
<code>MQCMIT</code> (mit Aufruf von <code>ax_reg</code> während der aktuellen globalen Arbeitseinheit)	<code>xa_end</code> <code>xa_prepare</code> <code>xa_commit</code> <code>xa_rollback</code>	Nicht zutreffend. Im nicht dynamischen Modus wird 'ax_reg' nicht aufgerufen.
<code>MQBACK</code> (ohne Aufruf von <code>ax_reg</code> während der aktuellen globalen Arbeitseinheit)	Keine XA-Aufrufe	<code>xa_end</code> <code>xa_rollback</code>
<code>MQBACK</code> (mit Aufruf von <code>ax_reg</code> während der aktuellen globalen Arbeitseinheit)	<code>xa_end</code> <code>xa_rollback</code>	Nicht zutreffend. Im nicht dynamischen Modus wird 'ax_reg' nicht aufgerufen.

Tabelle 4. Übersicht über die XA-Funktionsaufrufe (Forts.)

MQI-Aufruf	XA-Aufrufe mit dynamischer Registrierung	XA-Aufrufe ohne dynamische Registrierung
MQDISC, wenn zuerst MQCMIT oder MQBACK aufgerufen wurde. Wurden diese nicht aufgerufen, erfolgt die Verarbeitung von MQCMIT zum ersten Mal während MQDISC.	xa_close	xa_close

Anmerkungen:

1. Bei MQCMIT wird xa_commit aufgerufen, wenn xa_prepare erfolgreich ist. Andernfalls wird xa_rollback aufgerufen.

Szenario 2: Für die Koordination ist andere Software zuständig

In Szenario 2 werden globale Arbeitseinheiten von einem externen Transaktionsmanager koordiniert, der diese unter der Steuerung der Transaktionsmanager-API startet und festschreibt. Die Verben MQBEGIN, MQCMIT und MQBACK sind nicht verfügbar.

In diesem Abschnitt wird das Szenario einschließlich folgender Maßnahmen beschrieben:

- „Externe Synchronisationspunktkoordination“ auf Seite 84
- „CICS verwenden“ auf Seite 87
- „Microsoft Transaction Server (COM+) verwenden“ auf Seite 91

Externe Synchronisationspunktkoordination

Eine globale Arbeitseinheit kann auch von einem externen X/Open XA-kompatiblen Transaktionsmanager koordiniert werden. In diesem Fall ist der IBM MQ-Warteschlangenmanager an der Arbeitseinheit beteiligt, koordiniert diese jedoch nicht.

Der Steuerungsfluss in einer von einem externen Transaktionsmanager koordinierten globalen Arbeitseinheit sieht wie folgt aus:

1. Eine Anwendung teilt dem externen Synchronisationspunktkoordinator (z. B. TXSeries) mit, dass sie eine Transaktion starten möchte.
2. Der Synchronisationspunktkoordinator informiert bekannte Ressourcenmanager, z. B. IBM MQ, über die aktuelle Transaktion.
3. Die Anwendung gibt Aufrufe an Ressourcenmanager aus, die der aktuellen Transaktion zugeordnet sind. Sie könnte zum Beispiel MQGET-Aufrufe an IBM MQ ausgeben.
4. Die Anwendung setzt eine Festschreibungs- oder Zurücksetzungsanforderung an den externen Synchronisationspunktkoordinator ab.
5. Der Synchronisationspunktkoordinator schließt die Transaktion ab, indem er die entsprechenden Aufrufe an die einzelnen Ressourcenmanager ausgibt, wobei er normalerweise Protokolle für zweiphasige Festschreibung verwendet.

Die unterstützten Versionen externer Synchronisationspunktkoordinatoren, die eine zweiphasige Festschreibung für Transaktionen mit Beteiligung von IBM MQ bereitstellen können, sind unter Systemvoraussetzungen für IBM MQ definiert.

Im weiteren Teil dieses Abschnitts wird die Vorgehensweise zum Aktivieren externer Arbeitseinheiten beschrieben.

XA-Switchstruktur von IBM MQ

Jeder an einer extern koordinierten Arbeitseinheit beteiligte Ressourcenmanager muss eine XA-Switchstruktur bereitstellen. Diese Struktur definiert sowohl die Funktionalität des Ressourcenmanagers als auch die Funktionen, die vom Synchronisationspunktkoordinator aufgerufen werden müssen.

IBM MQ stellt zwei Versionen dieser Struktur bereit:

- *MQRMIXASwitch* für die statische XA-Ressourcenverwaltung
- *MQRMIXASwitchDynamic* für die dynamische XA-Ressourcenverwaltung

Schlagen Sie in der Dokumentation zu Ihrem Transaktionsmanager nach, ob die Schnittstelle für die statische oder die dynamische Ressourcenverwaltung verwendet werden soll. Sofern der Transaktionsmanager dies unterstützt, empfehlen wir die Verwendung der dynamischen XA-Ressourcenverwaltung.

Der Datentyp *long* in der XA-Spezifikation wird von manchen 64-Bit-Transaktionsmanagern als 64-Bit und von manchen als 32-Bit behandelt. IBM MQ unterstützt beide Modelle:

- Wenn es sich bei Ihrem Transaktionsmanager um einen 32-Bit-Transaktionsmanager oder um einen 64-Bit-Transaktionsmanager handelt, der den Typ *long* als 32-Bit behandelt, verwenden Sie die in [Tabelle 5 auf Seite 85](#) aufgelistete Switchloaddatei.
- Verwenden Sie bei einem 64-Bit-Transaktionsmanager verwenden, der den Datentyp *long* als 64-Bit behandelt, die in [Tabelle 6 auf Seite 86](#) aufgeführte Switchloaddatei.

Einige 64-Bit-Transaktionsmanager behandeln den Datentyp *long* als 64-Bit-Version. Für folgende 64-Bit-Transaktionsmanager ist bekanntermaßen die alternative 64-Bit-Switchloaddatei erforderlich:

- Tuxedo
-   64-Bit-TXSeries

Wenn Sie nicht sicher sind, welches Modell Ihr Transaktionsmanager verwendet, konsultieren Sie die Dokumentation zu Ihrem Transaktionsmanager.










Plattform	Name der Switchloaddatei (Server)	Name der Switchloaddatei (erweiterter transaktionsorientierter Client)
 Windows	<i>mqmx.dll</i>	<i>mqcxa.dll</i>
 AIX (ohne Threads)	<i>libmqmx.a</i>	<i>libmqcxa.a</i>
 AIX (ohne Threads)	<i>libmqmx_r.a</i>	<i>libmqcxa_r.a</i>
 Linux (ohne Threads)	<i>libmqmx.so</i>	<i>libmqcxa.so</i>
 Linux (ohne Threads)	<i>libmqmx_r.so</i>	<i>libmqcxa_r.so</i>

Tabelle 6. Namen alternativer 64-Bit-XA-Switchloaddateien

Plattform	Name der Switchloaddatei (Server)	Name der Switchloaddatei (erweiterter transaktionsorientierter Client)
 AIX (ohne Threads)	<i>libmqmxa64.a</i>	<i>libmqcxa64.a</i>
 AIX (ohne Threads)	<i>libmqmxa64_r.a</i>	<i>libmqcxa64_r.a</i>
 Linux (ohne Threads)	<i>libmqmxa64.so</i>	<i>libmqcxa64.so</i>
 Linux (ohne Threads)	<i>libmqmxa64_r.so</i>	<i>libmqcxa64_r.so</i>

Bei einigen externen Synchronisationspunktkoordinatoren (nicht CICS) müssen die einzelnen Ressourcenmanager, die an einer Arbeitseinheit beteiligt sind, ihren Namen im Namensfeld der XA-Switchstruktur angeben. Der Name des IBM MQ-Ressourcenmanagers lautet MQSeries_XA_RMI.

Der Synchronisationspunktordinator legt fest, wie die XA-Switchstruktur von IBM MQ eine Verknüpfung zu ihm herstellt. Informationen zur Verknüpfung der XA-Switchstruktur von IBM MQ mit CICS finden Sie im Abschnitt „CICS verwenden“ auf Seite 87. Informationen zur Verknüpfung der XA-Switchstruktur von IBM MQ mit anderen XA-kompatiblen Synchronisationspunktkoordinatoren finden Sie in der Dokumentation zu den betreffenden Produkten.

Die folgenden Überlegungen gelten für die Verwendung von IBM MQ mit allen XA-kompatiblen Synchronisationspunktkoordinatoren:

- Es wird erwartet, dass der Bibliothekscode des Transaktionsmanagers, der als Teil der vom Anwendungsprogrammierer aufgerufenen API ausgeführt wird, noch vor dem Aufruf von MQCONN einen **xa_open**-Aufruf für IBM MQ ausgibt.

Der **xa_open**-Aufruf muss im gleichen Thread erfolgen wie der MQCONN-Aufruf. Erforderlich macht dies die XA-Spezifikation, in der festgelegt ist, dass der Thread den Kontext angibt.

Diesen Ansatz verfolgt auch das Beispielprogramm `amqstxsx.c`. Dieses Beispielprogramm setzt voraus, dass ein **xa_open**-Aufruf in IBM MQ aus dem Bibliothekscode des Transaktionsmanagers in ihrer Funktion `tpopenerfolgt`.

Wenn vor dem MQCONN-Aufruf kein **xa_open**-Aufruf im selben Thread erfolgt, wird die IBM MQ-Warteschlangenmanagerverbindung keinem XA-Kontext zugeordnet.

Weitere Informationen hierzu finden Sie im Abschnitt [MQCTL](#).

- In der in einem Aufruf 'xa_open' vom Synchronisationspunktordinator übergebenen Struktur 'xa_info' ist der Name eines IBM MQ-Warteschlangenmanagers enthalten. Der Name hat dasselbe Format wie der an den Aufruf MQCONN übergebene Warteschlangenmanagername. Falls im Aufruf 'xa_open' kein Name übergeben wird, wird der Standardwarteschlangenmanager verwendet.

Alternativ dazu können in der Struktur 'xa_info' Werte für die Parameter *TPM* und *AXLIB* enthalten sein. Mit dem Parameter *TPM* wird der verwendete Transaktionsmanager angegeben. Gültige Werte sind CICS, TUXEDO und ENCINA. Der Parameter *AXLIB* gibt den Namen der Bibliothek an, in der die Funktionen 'ax_reg' und 'ax_unreg' des Transaktionsmanagers enthalten sind. Weitere Informationen zu diesen Parametern finden Sie im Abschnitt [Erweiterten transaktionsorientierten Client konfigurieren](#). Falls einer dieser Parameter in der Struktur 'xa_info' enthalten ist, wird der Warteschlangenmanagername im Parameter *QMNAME* angegeben, sofern nicht der Standardwarteschlangenmanager verwendet wird.

- An einer Transaktion, die von einer Instanz eines externen Synchronisationspunktkoordinators koordiniert wird, kann immer nur jeweils ein Warteschlangenmanager beteiligt sein. Der Synchronisationspunktkoordinator wird mit dem Warteschlangenmanager verbunden und für ihn gilt die Regel, dass nur jeweils eine Verbindung unterstützt wird.
- Alle Anwendungen, die Aufrufe an einen externen Synchronisationspunktkoordinator umfassen, können nur zu dem Warteschlangenmanager eine Verbindung herstellen, der an der vom externen Koordinator verwalteten Transaktion beteiligt ist (da sie bereits mit diesem Warteschlangenmanager verbunden sind). Allerdings müssen diese Anwendungen einen Aufruf MQCONN zur Anforderung einer Verbindungskennung und einen Aufruf MQDISC ausgeben, bevor sie beendet werden.
- Ein Warteschlangenmanager mit Ressourcenaktualisierungen, die von einem externen Synchronisationspunktkoordinator koordiniert werden, muss vor dem externen Synchronisationspunktkoordinator gestartet werden. Ebenso muss der Synchronisationspunktkoordinator vor dem Warteschlangenmanager beendet werden.
- Wenn der externe Synchronisationspunktkoordinator abnormal beendet wird, stoppen und starten Sie den Warteschlangenmanager erneut, bevor Sie den Synchronisationspunktkoordinator erneut starten, um sicherzustellen, dass alle Messagingoperationen, die zum Zeitpunkt des Fehlers nicht festgeschrieben waren, ordnungsgemäß aufgelöst werden.

CICS verwenden

CICS ist eines der Elemente von TXSeries.

Die XA-kompatiblen Versionen von TXSeries (die eine zweiphasige Festschreibung verwenden) sind im Abschnitt [Systemvoraussetzungen für IBM MQ](#) definiert.

IBM MQ unterstützt auch andere Transaktionsmanager. Links zu Informationen über unterstützte Software finden Sie auf der Webseite [Systemvoraussetzungen für IBM MQ](#).

Anforderungen der zweiphasigen Festschreibung

Anforderungen der zweiphasigen Festschreibung bei Verwendung der zweiphasigen Festschreibung von CICS mit IBM MQ. Diese Anforderungen gelten nicht für z/OS.

Beachten Sie die folgenden Anforderungen:

- IBM MQ und CICS müssen sich auf derselben physischen Maschine befinden.
- IBM MQ unterstützt nicht CICS auf einem IBM MQ MQI client.
- Sie müssen den Warteschlangenmanager starten und dabei seinen Name in der Zeilengruppe für die XAD-Ressourcendefinition angeben, **bevor** Sie versuchen, CICS zu starten. Wenn Sie dies nicht tun, können Sie CICS nicht starten, wenn Sie eine Zeilengruppe zur XAD-Ressourcendefinition für IBM MQ zur CICS-Region hinzugefügt haben.
- Es kann immer nur auf einen IBM MQ-Warteschlangenmanager aus einer einzelnen CICS-Region zugegriffen werden.
- Eine CICS-Transaktion kann erst nach Ausgabe einer MQCONN-Anforderung auf IBM MQ-Ressourcen zugreifen. Im Aufruf MQCONN muss der Name des IBM MQ-Warteschlangenmanagers angegeben sein, der im XAOpen-Eintrag der Zeilengruppe der XAD-Ressourcendefinition für die CICS-Region angegeben ist. Ist dieser Eintrag leer, muss in der Anforderung MQCONN der Standardwarteschlangenmanager angegeben sein.
- Eine CICS-Transaktion, die auf IBM MQ-Ressourcen zugreift, muss von der Transaktion aus einen Aufruf MQDISC ausgeben, bevor sie zu CICS zurückkehrt. Wenn dies nicht geschieht, ist der CICS-Anwendungsserver möglicherweise weiterhin verbunden und es bleiben Warteschlangen geöffnet. Wenn Sie keinen Exit für die Taskbeendigung installieren (siehe Abschnitt [„Beispiel für einen Taskbeendigungsexit“](#) auf Seite 90) kann es darüber hinaus passieren, dass der CICS-Anwendungsserver später bei einer nachfolgenden Transaktion abnormal beendet wird.
- Sie müssen sicherstellen, dass die CICS-Benutzer-ID (cics) Mitglied der Gruppe 'mqm' ist, sodass der CICS-Code zum Aufruf von IBM MQ berechtigt ist.

Bei Transaktionen, die in einer CICS-Umgebung ausgeführt werden, passt der Warteschlangenmanager seine Berechtigungsmethoden und den entscheidenden Kontext wie folgt an:

- Der Warteschlangenmanager fragt die Benutzer-ID ab, unter der CICS die Transaktion ausführt. Hierbei handelt es sich um die Benutzer-ID, die vom Objektberechtigungsmanager geprüft und für Kontextinformationen verwendet wird.
- Im Nachrichtenkontext lautet der Anwendungstyp MQAT_CICS.
- Der Anwendungsname im Kontext wird aus dem CICS-Transaktionsnamen kopiert.

ALW Allgemeine XA-Unterstützung

Es wird ein XA-Switchloadmodul bereitgestellt, mit dem Sie CICS mit IBM MQ for AIX, Linux, and Windows-Systemen verknüpfen können. Darüber hinaus sind Beispielquellcodedateien verfügbar, mit deren Hilfe Sie die XA-Switches für andere Transaktionsnachrichten entwickeln können. **IBM i** Allgemeine XA-Unterstützung wird unter IBM i nicht unterstützt.

Die Namen der bereitgestellten Switchloadmodule lauten wie folgt:

C (Quelle)	C (Exec) - fügen Sie eine der folgenden Komponenten zu Ihrer XAD.Stanza
amqzscix.c	AIX amqzsc - TXSeries für AIX 5.1
amqzscin.c	Windows mqmc4swi - TXSeries für Windows 5.1

Multi Bibliotheken für die Verwendung mit TXSeries for Multiplatforms erstellen

Verwenden Sie diese Informationen, wenn Sie Bibliotheken für die Verwendung mit TXSeries for Multiplatforms erstellen.

Bei *vordefinierten Switchloaddateien* handelt es sich um gemeinsam genutzte Bibliotheken (DLLs unter Windows) für die Verwendung mit CICS-Programmen, in denen eine 2-Phasen-Commit-Transaktion mit dem XA-Protokoll erforderlich ist. Die Namen dieser vordefinierten Bibliotheken sind in der Tabelle Wesentlicher Code für CICS-Anwendungen: XA-Initialisierungsroutine aufgeführt. Beispielquellcode ist außerdem in den folgenden Verzeichnissen enthalten:

Plattform	Directory	Quellendatei
Linux and Linux	AIX MQ_INSTALLATION_PATH/samp/	amqzscix.c
Windows Windows	MQ_INSTALLATION_PATH\Tools\c \ Beispiele	amqzscin.c

Dabei ist `MQ_INSTALLATION_PATH` das Verzeichnis, in dem IBM MQ installiert wurde.

Anmerkung: Sie müssen eine CICS -Switchloaddatei verwenden, die für die Version von IBM MQ erstellt wurde, mit der die Switchloaddatei verwendet wird.

Befolgen Sie die entsprechenden Anweisungen für Ihr Betriebssystem, um die Switchloaddatei aus der Beispielquelle zu erstellen:

AIX AIX

Geben Sie den folgenden Befehl ein:

```
export MQM_HOME=/usr/mqm
echo "amqzscix" > tmp.exp
```



```

xlc_r $MQM_HOME/samp/amqzscix.c -I/usr/lpp/cics/include -I$MQM_HOME/inc -e amqzscix -bE:tmp.exp -bM:SRE
-o amqzsc /usr/lpp/cics/lib/regxa_swxa.o -L$MQM_HOME/lib -L/usr/lpp/cics/lib -lcicsrt -lEncina
-lEncServer -lthreads -lsarpc -lmqmcics_r -lmqmx_r -lmqzi_r -lmqmc_r
rm tmp.exp

```

Linux Linux-Plattformen

Geben Sie den folgenden Befehl ein:

```

gcc -m32 -shared -fPIC -o amqzscix amqzscix.c
\ -IMQ_INSTALLATION_PATH/inc -I CICS_INSTALLATION_PATH/include
\ -LMQ_INSTALLATION_PATH/lib -Wl,-rpath=MQ_INSTALLATION_PATH/lib
\ -Wl,-rpath=/usr/lib -Wl,-rpath-link,/usr/lib -Wl,--no-undefined
-Wl,--allow-shlib-undefined -L CICS_LIB_PATH/regxa_swxa.o \-lpthread -ldl -lc
-shared -lmqzi_r -lmqmx_r -lmqmcics_r -ldl -lc

```

V9.4.0 Linux V9.4.0 Linux (64 Bit) TXSeries

Geben Sie den folgenden Befehl ein:

```

gcc -shared -fPIC -o amqzscix amqzscix.c
\ -IMQ_INSTALLATION_PATH/inc -I CICS_INSTALLATION_PATH/include
\ -LMQ_INSTALLATION_PATH/lib64 -Wl,-rpath=MQ_INSTALLATION_PATH/lib64
\ -Wl,-rpath=/usr/lib64 -Wl,-rpath-link,/usr/lib64 -Wl,--no-undefined -Wl,--allow-shlib-undefined
\ -L CICS_LIB_PATH/regxa_swxa.o \-lpthread -ldl -lc
-shared -lmqzi_r -lmqmx64_r -lmqmcics_r -ldl -lc

```

Windows Windows

Gehen Sie wie folgt vor:

1. Verwenden Sie den Befehl 'cl', um 'amqzscin.obj' durch Kompilation zu erstellen. Geben Sie dabei mindestens Folgendes an:

```
cl.exe -c -I EncinaPath\include -I MQ_INSTALLATION_PATH\include -Gz -LD amqzscin.c
```

2. Erstellen Sie eine Moduldefinitionsdatei mit dem Namen mqmc1415.def, die die folgenden Zeilen enthält:

```
LIBRARY MQMC4SWI
EXPORTS
CICS_XA_Init
```

3. Verwenden Sie den Befehl **lib**, um eine Exportdatei und eine Importbibliothek zu erstellen. Geben Sie dabei mindestens die folgenden Optionen an:

```
lib -def:mqmc4swi.def -out:mqmc4swi.lib
```

Wenn der Befehl lib erfolgreich ausgeführt wird, wird auch eine mqmc4swi.exp-Datei erstellt.

4. Verwenden Sie den Link-Befehl zum Erstellen von mqmc4swi.dll, indem Sie mindestens die folgende Option verwenden:

```
link.exe -dll -nod -out:mqmc4swi.dll
amqzscin.obj CicsPath\lib\regxa_swxa.obj
mqmc4swi.exp mqmcics4.lib
CicsPath\lib\libcicsrt.lib
DcePath\lib\libdce.lib DcePath\lib\pthread.lib
EncinaPath\lib\libEncina.lib
EncinaPath\lib\libEncServer.lib
msvcrt.lib kernel32.lib
```

IBM MQ-XA-Unterstützung und Tuxedo

IBM MQ auf AIX, Linux, and Windows-Systemen können Tuxedo-koordinierte XA-Anwendungen unendlich in "xa_start" blockieren.

Dies kann nur passieren, wenn zwei oder mehr Prozesse, die von Tuxedo in einer einzelnen globalen Transaktion koordiniert werden, versuchen, unter Verwendung derselben Transaktionsverzweigungs-ID (XID) auf IBM MQ zuzugreifen. Wenn jeder Prozess in der globalen Transaktion von Tuxedo eine andere XID zur Verwendung mit IBM MQ erhält, kann diese Situation nicht auftreten.

Konfigurieren Sie zur Vermeidung des Problems jede Anwendung in Tuxedo, die auf IBM MQ unter einer einzelnen globalen Transaktions-ID (gtrid) zugreift, innerhalb ihrer eigenen Tuxedo-Servergruppe. Prozesse in derselben Servergruppe verwenden beim Zugriff auf Ressourcenmanager im Auftrag einer einzelnen gtrid dieselbe XID, es besteht daher die Gefahr, dass sie bei xa_start in IBM MQ blockiert werden. Prozesse in verschiedenen Servergruppen verwenden beim Zugriff auf Ressourcenmanager separate XIDs und müssen daher ihre Transaktionsarbeit in IBM MQ nicht serialisieren.

Zweiphasige Festschreibung von CICS aktivieren

Damit CICS zur Koordinierung von Transaktionen mit MQI-Aufrufen eine zweiphasige Festschreibung verwenden kann, fügen Sie einen CICS-Zeilengruppeneintrag für die XAD-Ressourcendefinition zur CICS-Region hinzu. Die Informationen in diesem Abschnitt gelten nicht für z/OS.

Im folgenden Beispiel wird ein XAD-Zeilengruppeneintrag für IBM MQ for Windowshinzugefügt, wobei *Drive* das Laufwerk ist, auf dem IBM MQ installiert ist (z. B. D:).

```
cicsadd -cxad -rcics_region \  
ResourceDescription="MQM XA Product Description" \  
SwitchLoadFile="Drive:\Program Files\IBM\IBM MQ\bin\mqmc4swi.dll" \  
XAOpen=queue_manager_name
```

Verwenden Sie für erweiterte transaktionsorientierte Clients die Switchloaddatei 'mqcc4swi.dll'.

Im Folgenden sehen Sie ein Beispiel für das Hinzufügen eines XAD-Zeilengruppeneintrags für IBM MQ for AIX or Linux-Systeme, wobei *MQ_INSTALLATION_PATH* für das übergeordnete Verzeichnis steht, in dem IBM MQ installiert ist:

```
cicsadd -cxad -rcics_region \  
ResourceDescription="MQM XA Product Description" \  
SwitchLoadFile="MQ_INSTALLATION_PATH/lib/amqzsc" \  
XAOpen=queue_manager_name
```

Verwenden Sie für erweiterte transaktionsorientierte Clients die Switchloaddatei amqczsc.

Informationen zur Verwendung des Befehls **cicsadd** finden Sie in der Dokumentation zu CICS .

Aufrufe an IBM MQ können in eine CICS-Transaktion aufgenommen werden und die IBM MQ-Ressourcen werden gemäß CICS-Anweisung festgeschrieben oder rückgängig gemacht. Diese Unterstützung ist für Clientanwendungen nicht verfügbar.

Sie müssen einen MQCONN -Befehl von Ihrer CICS -Transaktion ausgeben, um auf IBM MQ -Ressourcen zuzugreifen, gefolgt von einem entsprechenden MQDISC -Aufruf beim Exit.

CICS-Benutzerexits aktivieren

Ein CICS-Benutzerexit-Punkt (normalerweise als *Benutzerexit* bezeichnet) ist eine Stelle in einem CICS-Modul, an der CICS die Kontrolle an ein von Ihnen geschriebenes Programm (ein Benutzerexit-Programm) übertragen kann und an dem CICS die Kontrolle wieder übernehmen kann, nachdem Ihr Exitprogramm seine Arbeit abgeschlossen hat.

Lesen Sie vor der Verwendung eines CICS-Benutzerexits den *CICS Administration Guide* für Ihre Plattform.

Beispiel für einen Taskbeendigungsexit

IBM MQ stellt Beispielquellcode für einen CICS-Taskbeendigungsexit bereit.

Der Beispielquellcode befindet sich in den folgenden Verzeichnissen:

Tabelle 9. CICS-Taskbeendigungsexits		
Plattform	Directory	Quellendatei
Systeme mit AIX and Linux	MQ_INSTALLATION_PATH/samp	amqzscgx.c
Windows	MQ_INSTALLATION_PATH\Tools\c \ Beispiele	amqzscgn.c

MQ_INSTALLATION_PATH steht für das übergeordnete Verzeichnis, in dem IBM MQ installiert ist.

Die Buildanweisungen für den Beispiel-Taskbeendigungsexit sind in den Kommentaren der einzelnen Quellendateien enthalten.

Dieser Exit wird von CICS bei einer normalen und einer abnormalen Taskbeendigung (nach einem Synchronisationspunkt) aufgerufen. Im Exitprogramm ist keine wiederherstellbare Arbeit zulässig.

Diese Funktionen werden nur in einem IBM MQ- und CICS-Kontext verwendet, in dem die CICS-Version die XA-Schnittstelle unterstützt. CICS bezieht sich auf diese Bibliotheken als `programs` oder `user exits`.

CICS verfügt über eine Reihe von Benutzerexits und `amqzscgx` wird, falls verwendet, unter CICS als `Task termination user exit (UE014015)` definiert und aktiviert, d. h. Exitnummer 15.

Wenn der Taskbeendigungsexit von CICS aufgerufen wird, hat CICS bereits IBM MQ über den Beendigungsstatus der Task informiert und IBM MQ hat die entsprechende Aktion (Commit oder Rollback) ausgeführt. Der Exit gibt lediglich einen Aufruf `MQDISC` für eine Bereinigung aus.

Ein Zweck der Installation und Konfiguration des CICS-Systems für die Verwendung eines Taskbeendigungsexits besteht darin, Ihr System vor einigen Konsequenzen fehlerhaften Anwendungs-codes zu schützen. Wenn beispielsweise eine CICS-Transaktion abnormal beendet wird, ohne zuerst `MQDISC` aufzurufen, und kein Taskbeendigungsexit installiert ist, wird möglicherweise (innerhalb von ca. 10 Sekunden) ein nachfolgender nicht behebbarer Fehler der CICS-Region angezeigt. Dies liegt daran, dass der IBM MQ-Thread für den fehlerfreien Zustand, der in dem `cicas`-Prozess ausgeführt wird, nicht gesendet wurde und keine Zeit zur Bereinigung und Rückgabe hatte. Dies könnte sich daran zeigen, dass der `cicas`-Prozess umgehend beendet wird, nachdem er `FFST`-Berichte in das Verzeichnis `'/var/mqm/errors'` oder ein äquivalentes Verzeichnis unter Windows geschrieben hat.

Microsoft Transaction Server (COM+) verwenden

COM+ (Microsoft Transaction Server) soll Benutzer bei der Ausführung von Geschäftslogikanwendungen auf einem typischen Middle-Tier-Server unterstützen.

Sehen [Funktionen, die nur mit der Primärinstallation auf Windows](#) für wichtige Informationen.

COM + unterteilt die Arbeit in *Aktivitäten*, bei denen es sich in der Regel um kurze unabhängige Geschäftslogikblöcke handelt, z. B. *Transferfonds von Konto A auf Konto B*. COM + stützt sich stark auf Objektorientierung und insbesondere auf COM; lose wird eine COM + -Aktivität wird durch ein COM (business) -Objekt repräsentiert.

COM+ ist im Betriebssystem integriert.

Mit COM+ erhält der Geschäftsobjektadministrator drei Services, die drei für den Programmierer von Geschäftsobjekten wichtige Bereiche abdecken:

- Transaktionsmanagement
- Sicherheit
- Ressourcenpool

COM+ wird in der Regel mit Front-End-Code verwendet, bei dem es sich um einen COM-Client für die in COM+ enthaltenen Objekte handelt, und mit Back-End-Services wie beispielsweise einer Datenbank, wobei IBM MQ die Brücke zwischen dem COM+-Geschäftsobjekt und dem Back-End darstellt.

Der Front-End-Code kann ein Standalone-Programm oder eine Active Server Page (ASP) auf dem Microsoft Internet Information Server (IIS) sein. Dieser Front-End-Code kann sich auf demselben Computer wie COM+ und die Geschäftsobjekte befinden, wobei die Verbindung über COM erfolgt. Der Front-End-Code kann sich auch auf einem anderen Computer befinden; in diesem Fall erfolgt die Verbindung über DCOM. Sie können mehrere Clients verwenden, die in unterschiedlichen Situationen jeweils auf dasselbe COM+-Geschäftsobjekt zugreifen.

Der Back-End-Code kann sich auf demselben Computer wie COM+ und die Geschäftsobjekte befinden oder auf einem anderen Computer; dabei erfolgt die Verbindung über eines der von IBM MQ unterstützten Protokolle.

Globale Arbeitseinheiten beenden

Der Warteschlangenmanager kann so konfiguriert werden, dass globale Arbeitseinheiten nach einem Inaktivitätsintervall ablaufen.

Sie sollten einen Ablaufscan aktivieren, wenn Ihre externe Transaktionsmanagersoftware Probleme hat, wo sie endet, ohne zuerst alle von ihren Anwendungen begonnenen Transaktionen abzuschließen.

Legen Sie zum Aktivieren des Ablaufscans die folgenden Variablen in der Umgebung fest, in der Ihr Warteschlangenmanager gestartet wird:

- **AMQ_TRANSACTION_EXPIRY_RESCAN**=*Neuscan-Intervall in Millisekunden*
- **AMQ_XA_TRANSACTION_EXPIRY**=*Zeitlimitüberschreitungsintervall in Millisekunden*

Beispiel für Linux und AIX:

```
# Scan runs every 30 seconds.  
# Value is given in milliseconds.  
export AMQ_TRANSACTION_EXPIRY_RESCAN=30000  
  
# Transactions of age 60 seconds in the Idle state are regarded as expired.  
# Value is given in milliseconds.  
export AMQ_XA_TRANSACTION_EXPIRY=60000
```

Beispiel für Windows:

```
rem Scan runs every 30 seconds.  
rem Value is given in milliseconds.  
set AMQ_TRANSACTION_EXPIRY_RESCAN=30000  
  
rem Transactions of age 60 seconds in the Idle state are regarded as expired.  
rem Value is given in milliseconds.  
set AMQ_XA_TRANSACTION_EXPIRY=60000
```

Anmerkung: Die Ablaufprüfung wirkt sich nur auf Transaktionen aus, die von externer Transaktionsmanagersoftware gestartet wurden und sich in Tabelle 6-4 der XA-Spezifikation aus der OPEN-Gruppe im Leerlauf befinden.

Transaktionen im Leerlauf sind keinem Anwendungsthread zugeordnet und die externe Transaktionsmanagersoftware hat den Funktionsaufruf **xa_prepare** noch nicht aufgerufen, um die Transaktion in den Status 'Vorbereitet' zu versetzen.

Wenn die externe Transaktionsmanagersoftware abstürzt und erneut gestartet wird, stellt sie normalerweise kein Commit oder Rollback für Transaktionen in diesem Status bereit.

Der Warteschlangenmanager führt einen Rollback für Puts und Gets durch, die in einer Transaktion ausgeführt wurden, die den Status 'Inaktiv' erreicht hat, wenn Sie Folgendes ausführen:

- Starten Sie den Warteschlangenmanager erneut.
- Aktivieren Sie den Ablaufscan und es wird eine inaktive Transaktion gefunden, die älter als das angegebene Alter ist.

Legen Sie **AMQ_XA_TRANSACTION_EXPIRY** fest, um das erwartete Intervall zwischen einer Anwendung, die ihre Arbeit innerhalb der Transaktion ausführt, und dem Abschluss der Transaktion zuzulassen.

Setzen Sie den Wert für **AMQ_TRANSACTION_EXPIRY_RESCAN** auf einen niedrigeren Wert als den Wert für **AMQ_XA_TRANSACTION_EXPIRY** . Führen Sie diese Aktion aus, damit die Scans der Transaktionslisten innerhalb des **AMQ_XA_TRANSACTION_EXPIRY** -Intervalls mehr als einmal ausgeführt werden.

Vorgehensweise zur Entscheidung, ob Sie Ablaufscans aktivieren müssen

Ablaufscans sind nur erforderlich, wenn Sie über einen externen Transaktionsmanager verfügen und dieser beendet oder abgestürzt ist, ohne seine inaktiven Transaktionen abzuschließen.

Wenn der externe Transaktionsmanager auf diese Weise beendet wird, werden die Puts und Gets von Nachrichten Ihrer Anwendung nicht festgeschrieben oder rückgängig gemacht, es sei denn, Sie starten den Warteschlangenmanager erneut oder haben Ablaufscans konfiguriert.

Wenn sich Transaktionen in diesem Status befinden, enthält die Ausgabe von **dspmqrtn -m QMNAME -q -a** eine Ausgabe wie im folgenden Beispiel. Es gibt ein paar Dinge zu beachten, um sicher zu sein, dass dies eine Transaktion von Interesse ist:

- Der Wert für TRANSTATE ist ACTIVE, nicht PREPARED.
- Die Zeitmarken zeigen an, dass die Transaktion lange vor der Ausführung von **dspmqrtn** gestartet wurde. Wenn die Zeitmarken sehr aktuell sind, ist dies keine bedenkliche Transaktion.
- Die EXTURID zeigt, dass es sich um eine XA-Transaktion handelt, da sie nicht leere Werte für XA_FORMATID, XA_GTRID und XA_BQUAL enthält.
- Es gibt einen oder mehrere Anwendungswarteschlangennamen, die als OBJECT-Datensätze aufgelistet sind und zeigen, dass in diesen Warteschlangen unter dieser Transaktion Puts oder Gets ausgeführt wurden.
- There is no live-connection information in this record (for example: CONN, PID, TID, APPLTAG, USERID) because the application is no longer connected.

```
TranNum(0,513)
TRANSTATE(ACTIVE)
UOWLOGDA(2024-05-23) UOWLOGTI(17.03.22)
UOWSTDA(2024-05-23) UOWSTTI(17.03.22)
UOWLOG( )
EXTURID(XA_FORMATID[WASD] XA_GTRID[0000018] XA_BQUAL[0000018])
OBJECT(MY.QUEUE)
```

Unit of recovery disposition

IBM MQ for z/OS provides unit of recovery dispositions. This feature allows you to configure whether the second phase of 2-phase commit transactions can be driven, for example, during recovery, when connected to another queue manager within the same queue sharing group (QSG).

IBM MQ for z/OS V7.0.1 and later supports the unit of recovery disposition.

Unit of recovery disposition

The unit of recovery disposition is related to an application's connection and subsequently any transactions that it starts. There are two possible unit of recovery dispositions.

- A GROUP unit of recovery disposition identifies that a transactional application is logically connected to the queue sharing group and does not have an affinity to any specific queue manager. Any 2-phase commit transactions that it starts that have completed phase-1 of the commit process, that is, they are in doubt, can be inquired and resolved, when connected to any queue manager within the QSG. In a recovery scenario this means that the transaction coordinator does not have to reconnect to the same queue manager, which might be unavailable.
- A QMGR unit of recovery disposition identifies that an application has a direct affinity to the queue manager to which it is connected and any transactions that it starts also have this disposition.

In a recovery scenario the transaction coordinator must reconnect to the same queue manager to inquire, and resolve, any in-doubt transactions, irrespective of whether the queue manager belongs to a queue sharing group.

z/OS For details of how to implement this feature, see [Unit of recovery disposition in a queue sharing group](#).

Sicherheitsszenarios

Dies sind Szenarios, die die Anwendung von Sicherheitsfunktionen auf unterschiedliche Konfigurationen veranschaulicht.

Die verfügbaren Sicherheitsszenarios werden in folgenden Unterabschnitten beschrieben:

Zugehörige Tasks

z/OS Sicherheit unter z/OS einrichten

z/OS Security scenario: two queue managers on z/OS

In this scenario, an application uses the **MQPUT1** call to put messages to queues on queue manager QM1. Some of the messages are then forwarded to queues on QM2, using TCP and LU 6.2 channels. The TCP channels can either use SSL/TLS or not. The application could be a batch application or a CICS application, and the messages are put using the **MQPMO_SET_ALL_CONTEXT** option.

This is illustrated in Figure 13 on page 94.

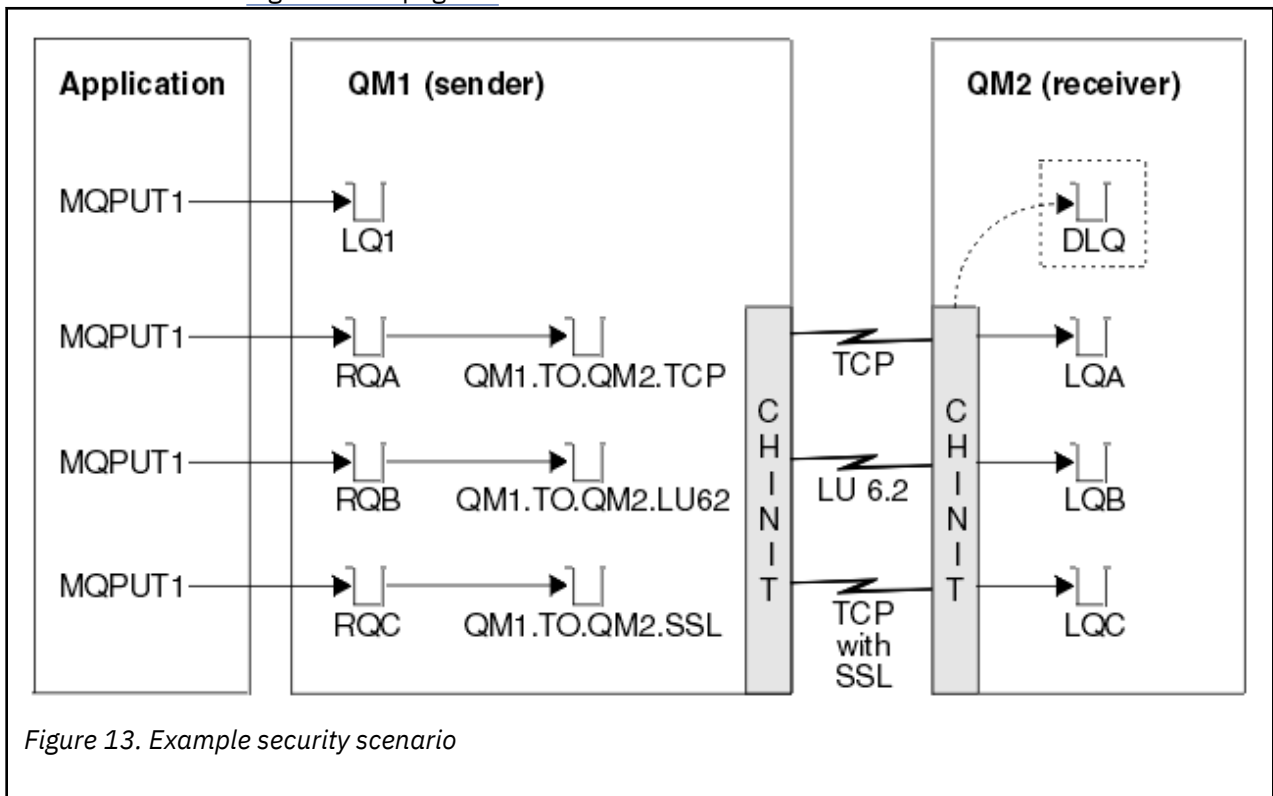


Figure 13. Example security scenario

The following assumptions are made about the queue managers:

- All the required IBM MQ definitions have been predefined or have been made through the CSQINP2 data set processed at queue manager startup.


If they have not, you need the appropriate access authority to the commands needed to define these objects.

- All the RACF® profiles required have been defined and appropriate access authorities have been granted, before the queue manager and channel initiators started.

If they have not, you need the appropriate authority to issue the RACF commands required to define all the profiles needed and grant the appropriate access authorities to those profiles. You also need the appropriate authority to issue the MQSC security commands to start using the new security profiles.

- All digital certificates required have been created and connected to key rings. The digital certificate sent by QM1 as part of the SSL/TLS handshake is recognized by RACF on QM2's system, either because it is also installed in that RACF profile, or because a matching Certificate Name File (CNF) filter exists.

Related tasks

 [Setting up security on z/OS](#)

Sicherheitsschaltereinstellungen für Szenario mit zwei Warteschlangenmanagern

Schaltereinstellungen und RACF-Profile

Folgende Sicherheitsschalter werden für beide Warteschlangenmanager gesetzt:

- Subsystemsicherheit ein
- Sicherheit der Warteschlange ein
- Sicherheit der alternativen Benutzer-IDs ein
- Kontextsicherheit ein
- Prozesssicherheit aus
- Namenslistensicherheit aus
- Themensicherheit aus
- Verbindungssicherheit ein
- Befehlssicherheit ein
- Sicherheit der Befehlsressourcen ein

Folgende Profile werden in der Klasse MQADMIN definiert, um die Prozess-, Namenslisten- und Themensicherheit zu inaktivieren:

```
QM1.NO.PROCESS.CHECKS
QM1.NO.NLIST.CHECKS
QM1.NO.TOPIC.CHECKS
QM2.NO.PROCESS.CHECKS
QM2.NO.NLIST.CHECKS
QM2.NO.TOPIC.CHECKS
```

Queue manager QM1 in two queue manager scenario

Queues and channels for QM1.

The following queues are defined on queue manager QM1:

LQ1

A local queue.

RQA

A remote queue definition, with the following attributes:

- RNAME(LQA)
- RQMNAME(QM2)
- XMITQ(QM1.TO.QM2.TCP)

RQB

A remote queue definition, with the following attributes:

- RNAME(LQB)
- RQMNAME(QM2)
- XMITQ(QM1.TO.QM2.LU62)

RQC

A remote queue definition, with the following attributes:

- RNAME(LQC)
- RQMNAME(QM2)
- XMITQ(QM1.TO.QM2.TLS)

QM1.TO.QM2.TCP

A transmission queue.

QM1.TO.QM2.LU62

A transmission queue.

QM1.TO.QM2.TLS

A transmission queue.

The following channels are defined on QM1:

QM1.TO.QM2.TCP

A sender channel definition, with the following attributes:

- CHLTYPE(SDR)
- TRPTYPE(TCP)
- XMITQ(QM1.TO.QM2.TCP)
- CONNAME(QM2TCP)

QM1.TO.QM2.LU62

A sender channel definition, with the following attributes:

- CHLTYPE(SDR)
- TRPTYPE(LU62)
- XMITQ(QM1.TO.QM2.LU62)
- CONNAME(QM2LU62)

(See [Security considerations for the channel initiator on z/OS](#) for information about setting up APPC security.)

QM1.TO.QM2.TLS

A sender channel definition, with the following attributes:

- CHLTYPE(SDR)
- TRPTYPE(TCP)
- XMITQ(QM1.TO.QM2.TLS)
- CONNAME(QM2TCP)
- SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256)

 **Queue manager QM2 in two queue manager scenario**

Queues and channels for QM2.

The following queues have been defined on queue manager QM2:

LQA

A local queue.

LQB

A local queue.

LQC

A local queue.

DLQ

A local queue that is used as the dead-letter queue.

The following channels have been defined on QM2:

QM1.TO.QM2.TCP

A receiver channel definition, with the following attributes:

- CHLTYPE(RCVR)
- TRPTYPE(TCP)
- PUTAUT(CTX)
- MCAUSER(MCATCP)

QM1.TO.QM2.LU62

A receiver channel definition, with the following attributes:

- CHLTYPE(RCVR)
- TRPTYPE(LU62)
- PUTAUT(CTX)
- MCAUSER(MCALU62)

(See [Security considerations for the channel initiator on z/OS](#) for information about setting up APPC security.)

QM1.TO.QM2.TLS

A receiver channel definition, with the following attributes:

- CHLTYPE(RCVR)
- TRPTYPE(TCP)
- PUTAUT(CTX)
- MCAUSER(MCASSL)
- SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256)

User IDs used in two queue manager scenario

Explanation of the user IDs in the scenario.

The following user IDs are used:

BATCHID

Batch application (Job or TSO ID)

MSGUSR

UserIdentifier in MQMD (context user ID)

MOVER1

QM1 channel initiator address space user ID

MOVER2

QM2 channel initiator address space user ID

MCATCP

MCAUSER specified on the TCP/IP without SSL/TLS receiver channel definition

MCALU62

MCAUSER specified on the LU 6.2 receiver channel definition

MCASSL

MCAUSER specified on the TCP/IP with SSL/TLS receiver channel definition

CICSAD1

CICS address space ID

CICSTX1

CICS task user ID

CERTID

The user ID associated by RACF with the flowed certificate.

Security profiles and accesses required for the two queue manager scenario

Information about the security profiles and accesses required for either a batch or CICS implementation of the two queue manager scenario.

The following table shows the security profiles that are required to enable the two queue manager scenario to work. Additional security profiles are also needed, depending on whether you are doing a batch or a CICS implementation of the scenario. For further information see [“Security profiles required for a batch application”](#) on page 98 and [“Security profiles required for a CICS application”](#) on page 100.

Table 10. Security profiles for the example scenario.

The four columns in this table show the class, the profile, the user ID, and access for the two queue manager scenario.

Class	Profile	User ID	Access
MQCONN	QM1.CHIN	MOVER1	READ
MQADMIN	QM1.RESLEVEL	BATCHID CICSAD1 MOVER1	NONE
MQADMIN	QM1.CONTEXT.**	MOVER1	CONTROL
MQQUEUE	QM1.SYSTEM.COMMAND.INPUT	MOVER1	UPDATE
MQQUEUE	QM1.SYSTEM.CHANNEL.SYNCQ	MOVER1	UPDATE
MQQUEUE	QM1.SYSTEM.CHANNEL.INITQ	MOVER1	UPDATE
MQQUEUE	QM1.SYSTEM.COMMAND.REPLY.MODEL	MOVER1	UPDATE
MQQUEUE	QM1.SYSTEM.ADMIN.CHANNEL.EVENT	MOVER1	UPDATE
MQQUEUE	QM1.QM1.TO.QM2.TCP	MOVER1	ALTER
MQQUEUE	QM1.QM1.TO.QM2.LU62	MOVER1	ALTER
MQQUEUE	QM1.QM1.TO.QM2.TLS	MOVER1	ALTER
MQCONN	QM2.CHIN	MOVER2	READ
MQADMIN	QM2.RESLEVEL	MOVER2	NONE
MQADMIN	QM2.CONTEXT.**	MOVER2	CONTROL
MQQUEUE	QM2.SYSTEM.COMMAND.INPUT	MOVER2	UPDATE
MQQUEUE	QM2.SYSTEM.CHANNEL.SYNCQ	MOVER2	UPDATE
MQQUEUE	QM2.SYSTEM.CHANNEL.INITQ	MOVER2	UPDATE
MQQUEUE	QM2.SYSTEM.COMMAND.REPLY.MODEL	MOVER2	UPDATE
MQQUEUE	QM2.SYSTEM.ADMIN.CHANNEL.EVENT	MOVER2	UPDATE
MQQUEUE	QM2.DLQ	MOVER2	UPDATE

Security profiles required for a batch application

Additional security profiles required for a batch implementation of the two queue manager scenario.

The batch application runs under user ID BATCHID on QM1. It connects to queue manager QM1 and puts messages to the following queues:

- LQ1

- RQA
- RQB
- RQC

It uses the MQPMO_SET_ALL_CONTEXT option. The alternate user ID found in the *UserIdentifier* field of the message descriptor (MQMD) is MSGUSR.

The following profiles are required on queue manager QM1:

<i>Table 11. Sample security profiles for the batch application on queue manager QM1</i>			
Class	Profile	User ID	Access
MQCONN	QM1.BATCH	BATCHID	READ
MQADMIN	QM1.CONTEXT.**	BATCHID	CONTROL
MQQUEUE	QM1.LQ1	BATCHID	UPDATE
MQQUEUE	QM1.RQA	BATCHID	UPDATE
MQQUEUE	QM1.RQB	BATCHID	UPDATE
MQQUEUE	QM1.RQC	BATCHID	UPDATE

The following profiles are required on queue manager QM2 for messages put to queue RQA on queue manager QM1 (for the TCP/IP channel not using TLS):

<i>Table 12. Sample security profiles for queue manager QM2 using TCP/IP and not TLS</i>			
Class	Profile	User ID	Access
MQADMIN	QM2.ALTERNATE.USER.MSGUSR	MCATCP MOVER2	UPDATE
MQADMIN	QM2.CONTEXT.**	MCATCP MOVER2	CONTROL
MQQUEUE	QM2.LQA	MOVER2 MSGUSR	UPDATE
MQQUEUE	QM2.DLQ	MOVER2 MSGUSR	UPDATE

Notes:

1. The user ID passed in the MQMD of the message is used as the user ID for the MQPUT1 on queue manager QM2 because the receiver channel was defined with PUTAUT(CTX) and MCAUSER(MCATCP).
2. The MCAUSER field of the receiver channel definition is set to MCATCP; this user ID is used in addition to the channel initiator address space user ID for the checks carried out against the alternate user ID and context profile.
3. The MOVER2 user ID and the *UserIdentifier* in the message descriptor (MQMD) are used for the resource checks against the queue.
4. The MOVER2 and MSGUSR user IDs both need access to the dead-letter queue so that messages that cannot be put to the destination queue can be sent there.
5. Two user IDs are checked on all three checks performed because RESLEVEL is set to NONE.

The following profiles are required on queue manager QM2 for messages put to queue RQB on queue manager QM1 (for the LU 6.2 channel):

<i>Table 13. Sample security profiles for queue manager QM2 using LU 6.2</i>			
Class	Profile	User ID	Access
MQADMIN	QM2.ALTERNATE.USER.MSGUSR	MICALU62 MOVER1	UPDATE

<i>Table 13. Sample security profiles for queue manager QM2 using LU 6.2 (continued)</i>			
Class	Profile	User ID	Access
MQADMIN	QM2.CONTEXT.**	MCALU62 MO- VER1	CONTROL
MQQUEUE	QM2.LQB	MOVER1 MSGUSR	UPDATE
MQQUEUE	QM2.DLQ	MOVER1 MSGUSR	UPDATE

Notes:

1. The user ID passed in the MQMD of the message is used as the user ID for the MQPUT1 on queue manager QM2 because the receiver channel was defined with PUTAUT(CTX) and MCAUSER(MCALU62).
2. The MCA user ID is set to the value of the MCAUSER field of the receiver channel definition (MCALU62).
3. Because LU 6.2 supports security on the communications system for the channel, the user ID received from the network is used as the channel user ID (MOVER1).
4. Two user IDs are checked on all three checks performed because RESLEVEL is set to NONE.
5. MCALU62 and MOVER1 are used for the checks performed against the alternate user ID and Context profiles, and MSGUSR and MOVER1 are used for the checks against the queue profile.
6. The MOVER1 and MSGUSR user IDs both need access to the dead-letter queue so that messages that cannot be put to the destination queue can be sent there.

The following profiles are required on queue manager QM2 for messages put to queue RQC on queue manager QM1 (for the TCP/IP channel using TLS):

<i>Table 14. Sample security profiles for queue manager QM2 using TCP/IP and TLS</i>			
Class	Profile	User ID	Access
MQADMIN	QM2.ALTERNATE.USER.MSGUSR	MCASSL CERTID	UPDATE
MQADMIN	QM2.CONTEXT.**	MCASSL CERTID	CONTROL
MQQUEUE	QM2.LQC	CERTID MSGUSR	UPDATE
MQQUEUE	QM2.DLQ	CERTID MSGUSR	UPDATE

Notes:

1. The user ID passed in the MQMD of the message is used as the user ID for the MQPUT1 on queue manager QM2 because the receiver channel was defined with PUTAUT(CTX) and MCAUSER(MCASSL).
2. The MCA user ID is set to the value of the MCAUSER field of the receiver channel definition (MCASSL).
3. Because the certificate flowed by the channel from QM1 as part of the TLS handshake might be installed on QM2's system, or might match a certificate name filter on QM2's system, the user ID found during that matching is used as the channel user ID (CERTID).
4. Two user IDs are checked on all three checks performed because RESLEVEL is set to NONE.
5. MCASSL and CERTID are used for the checks performed against the alternate user ID and Context profiles, and MSGUSR and MOVER1 are used for the checks against the queue profile.
6. The CERTID and MSGUSR user IDs both need access to the dead-letter queue so that messages that cannot be put to the destination queue can be sent there.

 **Security profiles required for a CICS application**

Additional security profiles required for a CICS implementation of the two queue manager scenario.

The CICS application uses a CICS address space user ID of CICSAD1 and a CICS task user ID of CICSTX1. The security profiles required on queue manager QM1 are different from those profiles required for the batch application. The profiles required on queue manager QM2 are the same as for the batch application.

The following profiles are required on queue manager QM1:

Class	Profile	User ID	Access
MQCONN	QM1.CICS	CICSAD1	READ
MQADMIN	QM1.CONTEXT.**	CICSAD1 CICSTX1	CONTROL
MQQUEUE	QM1.LQ1	CICSAD1 CICSTX1	UPDATE
MQQUEUE	QM1.RQA	CICSAD1 CICSTX1	UPDATE
MQQUEUE	QM1.RQB	CICSAD1 CICSTX1	UPDATE

z/OS Security scenario: queue sharing group on z/OS

In this scenario, an application uses the **MQPUT1** call to put messages to queues on queue manager QM1. Some of the messages are then forwarded to queues on QM2, using TCP and LU 6.2 channels. The application is a batch application, and the messages are put using the **MQPMO_SET_ALL_CONTEXT** option.

This is illustrated in [Figure 13 on page 94](#).

The following assumptions are made about the queue managers:

- All the required IBM MQ definitions have been predefined or have been made through the CSQINP2 data set processed at queue manager startup.

If they have not, you need the appropriate access authority to the commands needed to define these objects.

- All the RACF profiles required have been defined and appropriate access authorities have been granted, before the queue manager and channel initiators started.

If they have not, you need the appropriate authority to issue the RACF commands required to define all the profiles needed and grant the appropriate access authorities to those profiles. You also need the appropriate authority to issue the MQSC security commands to start using the new security profiles.

Related tasks

[z/OS Setting up security on z/OS](#)

z/OS Security switch settings for queue sharing group scenario

Switch settings and RACF profiles.

The following security switches are set for the queue sharing group:

- Subsystem security on
- Queue sharing group security on
- Queue manager security off
- Queue security on
- Alternate user security on
- Context security on
- Process security off
- Namelist security off
- Topic security off

- Connection security on
- Command security on
- Command resource security on

The following profiles are defined in the MQADMIN class to turn process, namelist, topic and queue manager level security off:

```
QSGA.NO.PROCESS.CHECKS
QSGA.NO.NLIST.CHECKS
QSGA.NO.TOPIC.CHECKS
QSGA.NO.QMGR.CHECKS
```

Queue manager QM1 in queue sharing group scenario

Queues and channels for QM1.

The following queues are defined on queue manager QM1:

LQ1

A local queue.

RQA

A remote queue definition, with the following attributes:

- RNAME(LQA)
- RQMNAME(QM2)
- XMITQ(QM1.TO.QM2.TCP)

RQB

A remote queue definition, with the following attributes:

- RNAME(LQB)
- RQMNAME(QM2)
- XMITQ(QM1.TO.QM2.LU62)

QM1.TO.QM2.TCP

A transmission queue.

QM1.TO.QM2.LU62

A transmission queue.

The following channels are defined on QM1:

QM1.TO.QM2.TCP

A sender channel definition, with the following attributes:

- CHLTYPE(SDR)
- TRPTYPE(TCP)
- XMITQ(QM1.TO.QM2.TCP)
- CONNAME(QM2TCP)

QM1.TO.QM2.LU62

A sender channel definition, with the following attributes:

- CHLTYPE(SDR)
- TRPTYPE(LU62)
- XMITQ(QM1.TO.QM2.LU62)
- CONNAME(QM2LU62)

(See [Security considerations for the channel initiator on z/OS](#) for information about setting up APPC security.)

Queue manager QM2 in queue sharing group scenario

Queues and channels for QM2.

The following queues have been defined on queue manager QM2:

LQA

A local queue.

LQB

A local queue.

DLQ

A local queue that is used as the dead-letter queue.

The following channels have been defined on QM2:

QM1.TO.QM2.TCP

A receiver channel definition, with the following attributes:

- CHLTYPE(RCVR)
- TRPTYPE(TCP)
- PUTAUT(CTX)
- MCAUSER(MCATCP)

QM1.TO.QM2.LU62

A receiver channel definition, with the following attributes:

- CHLTYPE(RCVR)
- TRPTYPE(LU62)
- PUTAUT(CTX)
- MCAUSER(MCALU62)

(See [Security considerations for the channel initiator on z/OS](#) for information about setting up APPC security.)

User IDs used in queue sharing group scenario

Explanation of the user IDs in the scenario.

The following user IDs are used:

BATCHID

Batch application (Job or TSO ID)

MSGUSR

UserIdentifier in MQMD (context user ID)

MOVER1

QM1 channel initiator address space user ID

MOVER2

QM2 channel initiator address space user ID

MCA TCP

MCAUSER specified on the TCP/IP receiver channel definition

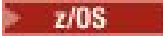
MCA LU62

MCAUSER specified on the LU 6.2 receiver channel definition

Security profiles and accesses required for queue sharing group scenario

Security profiles and accesses for either a batch or CICS implementation of the queue sharing group scenario.

The following table shows the security profiles that are required to enable the queue sharing group scenario to work. A batch implementation of this scenario also requires the additional security profiles that are described in “Security profiles required for a batch application” on page 104.

Table 16. Security profiles for the example scenario. 

The four columns in this table show the class, the profile, the user ID, and access for the queue sharing group scenario.

Class	Profile	User ID	Access
MQCONN	QSGA.CHIN	MOVER1 MO- VER2	READ
MQADMIN	QSGA.RESLEVEL	BATCHID MOVER1 MO- VER2	NONE
MQADMIN	QSGA.CONTEXT.**	MOVER1 MO- VER2	CONTROL
MQQUEUE	QSGA.SYSTEM.COMMAND.INPUT	MOVER1 MO- VER2	UPDATE
MQQUEUE	QSGA.SYSTEM.CHANNEL.SYNCQ	MOVER1 MO- VER	UPDATE
MQQUEUE	QSGA.SYSTEM.CHANNEL.INITQ	MOVER1 MO- VER2	UPDATE
MQQUEUE	QSGA.SYSTEM.COMMAND.REPLY.MODEL	MOVER1 MO- VER2	UPDATE
MQQUEUE	QSGA.SYSTEM.ADMIN.CHANNEL.EVENT	MOVER1 MO- VER2	UPDATE
MQQUEUE	QSGA.SYSTEM.QSG.CHANNEL.SYNCQ	MOVER1 MO- VER2	UPDATE
MQQUEUE	QSGA.SYSTEM.QSG.TRANSMIT.QUEUE	MOVER1 MO- VER2	UPDATE
MQQUEUE	QSGA.QM1.TO.QM2.TCP	MOVER1	ALTER
MQQUEUE	QSGA.QM1.TO.QM2.LU62	MOVER1	ALTER
MQQUEUE	QSGA.DLQ	MOVER2	UPDATE

Security profiles required for a batch application

Additional security profiles required for a batch implementation of the queue sharing group scenario.

The batch application runs under user ID BATCHID on QM1. It connects to queue manager QM1 and puts messages to the following queues:

- LQ1
- RQA
- RQB

It uses the MQPMO_SET_ALL_CONTEXT option. The user ID found in the *UserIdentifier* field of the message descriptor (MQMD) is MSGUSR.

The following profiles are required on queue manager QM1:

<i>Table 17. Sample security profiles for the batch application on queue manager QM1</i>			
Class	Profile	User ID	Access
MQCONN	QSGA.BATCH	BATCHID	READ
MQADMIN	QSGA.CONTEXT.**	BATCHID	CONTROL
MQQUEUE	QSGA.LQ1	BATCHID	UPDATE
MQQUEUE	QSGA.RQA	BATCHID	UPDATE
MQQUEUE	QSGA.RQB	BATCHID	UPDATE

The following profiles are required on queue manager QM2 for messages put to queue RQA on queue manager QM1 (for the TCP/IP channel):

<i>Table 18. Sample security profiles for queue manager QM2 using TCP/IP</i>			
Class	Profile	User ID	Access
MQADMIN	QSGA.ALTERNATE.USER.MSGUSR	MCATCP MOVER2	UPDATE
MQADMIN	QSGA.CONTEXT.**	MCATCP MOVER2	CONTROL
MQQUEUE	QSGA.LQA	MOVER2 MSGUSR	UPDATE
MQQUEUE	QSGA.DLQ	MOVER2 MSGUSR	UPDATE

Notes:

1. The user ID passed in the MQMD of the message is used as the user ID for the MQPUT1 on queue manager QM2 because the receiver channel was defined with PUTAUT(CTX) and MCAUSER(MCATCP).
2. The MCAUSER field of the receiver channel definition is set to MCATCP; this user ID is used in addition to the channel initiator address space user ID for the checks carried out against the alternate user ID and context profile.
3. The MOVER2 user ID and the *UserIdentifier* in the message descriptor (MQMD) are used for the resource checks against the queue.
4. The MOVER2 and MSGUSR user IDs both need access to the dead-letter queue so that messages that cannot be put to the destination queue can be sent there.
5. Two user IDs are checked on all three checks performed because RESLEVEL is set to NONE.

The following profiles are required on queue manager QM2 for messages put to queue RQB on queue manager QM1 (for the LU 6.2 channel):

<i>Table 19. Sample security profiles for queue manager QM2 using LU 6.2</i>			
Class	Profile	User ID	Access
MQADMIN	QSGA.ALTERNATE.USER.MSGUSR	MCALU62 MOVER1	UPDATE
MQADMIN	QSGA.CONTEXT.**	MCALU62 MOVER1	CONTROL
MQQUEUE	QSGA.LQB	MOVER1 MSGUSR	UPDATE

Class	Profile	User ID	Access
MQQUEUE	QSGA.DLQ	MOVER1 MSGUSR	UPDATE

Notes:

1. The user ID passed in the MQMD of the message is used as the user ID for the MQPUT1 on queue manager QM2 because the receiver channel was defined with PUTAUT(CTX) and MCAUSER(MCALU62).
2. The MCA user ID is set to the value of the MCAUSER field of the receiver channel definition (MCALU62).
3. Because LU 6.2 supports security on the communications system for the channel, the user ID received from the network is used as the channel user ID (MOVER1).
4. Two user IDs are checked on all three checks performed because RESLEVEL is set to NONE.
5. MCALU62 and MOVER1 are used for the checks performed against the alternate user ID and Context profiles, and MSGUSR and MOVER1 are used for the checks against the queue profile.
6. The MOVER1 and MSGUSR user IDs both need access to the dead-letter queue so that messages that cannot be put to the destination queue can be sent there.

Server-to-server message channel interception example configurations

Server-to-server message channel interception requires configuration of channel definitions, as well as Advanced Message Security policies, to ensure that inbound and outbound messages can be correctly protected or unprotected. The configuration varies depending on whether the channel is inbound or outbound.

Inbound channel

The following example shows a typical configuration for an inbound channel of type receiver, and provides details of the AMS policy required to protect unprotected inbound messages:



Figure 14. Inbound configuration

The example shows:

- Queue manager QMA
- Channel TO.QMA
- Local queue DESTQ

Use the following code:

```
DEFINE CHANNEL(TO.QMA) CHLTYPE(RCVR) SSLCAUTH(REQUIRED) SSLCIPH(ANY_TLS12) TRPTYPE(TCP)
SPLPROT(ASPOLICY)

DEFINE QLOCAL(DESTQ) DESCR('AMS PROTECTED QUEUE')

setmqsp1 -m QMA -p DESTQ -e AES256 -x CN=TEST,0=ORG,C=US
```

Note: The policy described in the preceding text encrypts messages only; that is, AMS Confidentiality.

See [setmqsp1](#) and [the message security policy \(CSQOUTIL\)](#) for information on using **setmqsp1** on z/OS.

Outbound channel

The following example shows a typical configuration for an outbound channel of type sender. The example provides details of the AMS policies required to protect messages put to the remote queue, and to unprotect and send messages got from the transmission queue:

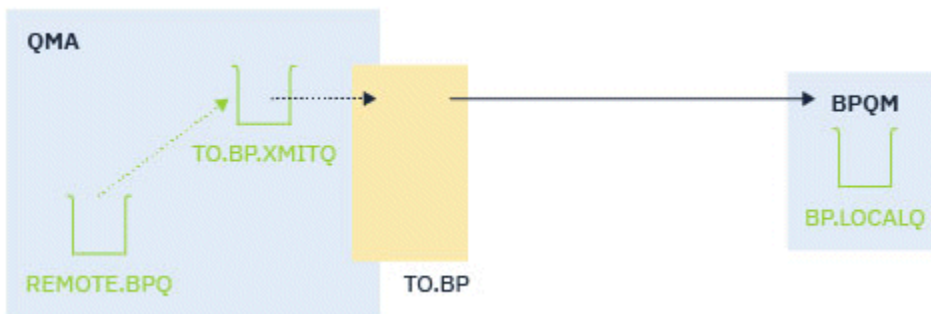


Figure 15. Outbound configuration

The example shows a:

- Queue manager QMA
- Channel TO.BP
- Local transmission queue TO.BP.XMITQ
- Remote queue REMOTE.BPQ

use the following code:

```
DEFINE CHANNEL(TO.BP) CHLTYPE(SDR) SSLCAUTH(REQUIRED) SSLCIPH(ANY_TLS12) TRPTYPE(TCP)
SPLPROT(REMOVE) CONNAME('server(1414)') XMITQ(TO.BP.XMITQ)

DEFINE QLOCAL(TO.BP.XMITQ) DESCR('TRANSMISSION QUEUE FOR TO.BP') USAGE(XMITQ)

DEFINE QREMOTE(REMOTE.BPQ) DESCR('REMOTE QUEUE TO BP') RNAME(BP.LOCALQ) RQMANME(BPQM)
XMITQ(TO.BP.XMITQ)

setmqsp1 -m QMA -p TO.BP.XMITQ -e AES256 -r CN=TEST,O=ORG,C=US
setmqsp1 -m QMA -p REMOTE.BPQ -e AES256 -r CN=TEST,O=ORG,C=US
```

Note: The policy described in the preceding text encrypts messages only; that is, AMS Confidentiality.

Zwei Warteschlangenmanager über SSL/TLS verbinden

Für eine sichere Kommunikation, bei der die verschlüsselten Sicherheitsprotokolle TLS verwendet werden, müssen Kommunikationskanäle eingerichtet und die digitalen Zertifikate verwaltet werden, die Sie zur Authentifizierung verwenden werden.

Beim Konfigurieren Ihrer SSL/TLS-Installation müssen Sie Ihre Kanäle so definieren, dass sie TLS verwenden. Zudem müssen Sie Ihre digitalen Zertifikate anfordern und verwalten. Auf einem Testsystem können Sie selbst signierte Zertifikate verwenden, die von einer lokalen Zertifizierungsstelle ausgegeben wurden. Verwenden Sie selbst signierte Zertifikate nicht auf einem Produktionssystem.


Sie finden ausführliche Informationen zur Erstellung und Verwaltung von Zertifikaten in den folgenden Abschnitten:

- **IBM i** [Unter IBM i mit SSL oder TLS arbeiten](#)
- **ALW** [Auf AIX, Linux, and Windows-Systemen mit SSL oder TLS arbeiten](#)
- **z/OS** [Unter z/OS mit SSL oder TLS arbeiten](#)

In diesen Themen werden die Aufgaben bei der Einrichtung der SSL/TLS-Kommunikation erläutert, zudem erhalten Sie dort eine schrittweise Anleitung zum Ausführen dieser Aufgaben.

Testen Sie gegebenenfalls die SSL/TLS-Clientauthentifizierung, die ein optionaler Bestandteil der Protokolle ist. Während des SSL/TLS-Handshakes fordert der SSL/TLS-Client immer ein digitales Zertifikat beim Server an und prüft es. Bei der IBM MQ-Implementierung fordert der SSL/TLS-Server immer ein Zertifikat beim Client an.

Anmerkungen:

1. In diesem Kontext bezieht sich ein SSL/TLS-Client auf die Verbindung, die den Handshake initialisiert.
2.  Wenn ein z/OS-Warteschlangenmanager in der Rolle eines SSL/TLS-Clients handelt, sendet er nur ein Zertifikat.

Der SSL/TLS-Client sendet nur dann ein Zertifikat, wenn er ein Zertifikat mit einer übereinstimmenden Bezeichnung findet. Weitere Informationen finden Sie im Artikel [Digitale Zertifikatsbezeichnungen](#).

Wenn ein Clientzertifikat gesendet wird, wird es immer vom SSL/TLS-Server geprüft. Sendet der Client kein Zertifikat, schlägt die Authentifizierung nur fehl, wenn der Parameter **SSLCAUTH** für das Ende des Kanals, das als SSL/TLS-Server agiert, auf **REQUIRED** gesetzt ist oder ein Wert für den Parameter **SSLPEER** angegeben ist. Weitere Informationen zur anonymen Verbindung eines Warteschlangenmanagers (d. h. wenn der SSL/TLS-Client kein Zertifikat sendet) finden Sie im Abschnitt „Zwei Warteschlangenmanager über unidirektionale Authentifizierung verbinden“ auf Seite 113.

Selbst signierte Zertifikate für die gegenseitige Authentifizierung zweier Warteschlangenmanager verwenden

Folgen Sie diesen Beispielanweisungen, um die gegenseitige Authentifizierung zweier Warteschlangenmanager mithilfe von selbst signierten TLS-Zertifikaten zu implementieren.

Informationen zu diesem Vorgang

Szenario:

- Sie haben zwei Warteschlangenmanager, QM1 und QM2, die sicher miteinander kommunizieren müssen. Deswegen sollen sich QM1 und QM2 gegenseitig authentifizieren.
- Die sichere Kommunikation möchten Sie mit selbst signierten Zertifikaten testen.

Die Konfiguration wird danach so aussehen:

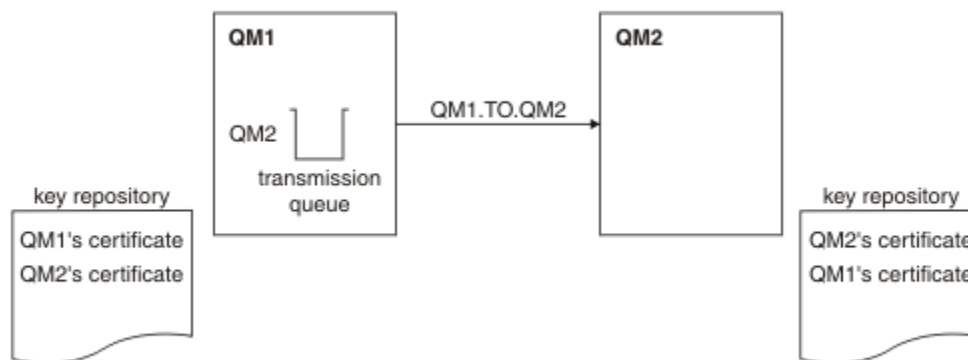











Abbildung 16. Aus dieser Task entstehende Konfiguration

In [Abbildung 16](#) auf Seite 108 enthält das Schlüsselrepository von QM1 das Zertifikat für QM1 und das öffentliche Zertifikat von QM2. Das Schlüsselrepository von QM2 enthält das Zertifikat für QM2 und das öffentliche Zertifikat von QM1.

Vorgehensweise

1. Bereiten Sie das Schlüsselrepository auf beiden Warteschlangenmanagern entsprechend des Betriebssystems vor:
 -  Auf [AIX, Linux, and Windows-Systemen](#).
 -  Auf [z/OS-Systemen](#).
2. Erstellen Sie für jeden Warteschlangenmanager ein selbst signiertes Zertifikat:
 -  Auf [AIX, Linux, and Windows-Systemen](#).
 -  Auf [z/OS-Systemen](#).
3. Extrahieren Sie eine Kopie jedes Zertifikats:
 -  Auf [AIX, Linux, and Windows-Systemen](#).
 -  Auf [z/OS-Systemen](#).
4. Übertragen Sie mit einem Tool wie FTP den öffentlichen Teil des QM1-Zertifikats auf das System von QM2 (und umgekehrt) , wie im Abschnitt [Selbst signierte Zertifikate austauschen](#) beschrieben ist.
5. Fügen Sie auf jedem Warteschlangenmanager das Partnerzertifikat zum Schlüsselrepository hinzu:
 -  Auf [AIX, Linux, and Windows-Systemen](#).
 -  Auf [z/OS-Systemen](#).
6. Definieren Sie auf QM1 einen Senderkanal und die zugehörige Übertragungswarteschlange, indem Sie einen Befehl wie den folgenden ausgeben:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QM1.MACH.COM) XMITQ(QM2)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA) DESCR('Sender channel using TLS from QM1 to QM2')
DEFINE QLOCAL(QM2) USAGE(XMITQ)
```


Dieses Beispiel verwendet die CipherSpec TLS_RSA. Die CipherSpecs an jedem Ende des Kanals müssen identisch sein.

7. Definieren Sie auf QM2 einen Empfängerkanal, indem Sie einen Befehl wie den folgenden ausgeben:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using TLS from QM1 to QM2')
```

Der Kanal muss denselben Namen wie der in Schritt „6“ auf Seite 109 definierte Senderkanal haben und die gleiche CipherSpec verwenden.

8. Starten Sie den Kanal.

 Auf [z/OS](#), siehe [Senderkanal starten](#).

Ergebnisse

Die Schlüsselrepositorys und Kanäle werden erstellt, wie in [Abbildung 16 auf Seite 108](#) gezeigt.

Nächste Schritte

Überprüfen Sie die erfolgreiche Ausführung der Task durch Ausgabe von DISPLAY-Befehlen. Bei erfolgreichem Abschluss der Task sieht die Ausgabe in etwa wie in den folgenden Beispielen aus.

Geben Sie auf dem Warteschlangenmanager QM1 den folgenden Befehl ein:

```
DISPLAY CHS(QM1.TO.QM2) SSLPEER SSLCERTI
```

Die Ausgabe sieht in etwa wie folgt aus:

```
DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
 4 : DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
CHANNEL(QM1.TO.QM2)          CHLTYPE(SDR)
CONNAME(9.20.25.40)          CURRENT
RQMNAME(QM2)
SSLCERTI("CN=QM2,OU=<Department>,O=<Organization>,ST=<State>,C=<Country>")
SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5E:02,CN=QM2,OU=<Department>,O=<Organizati
on>,ST=<State>,C=<Country>")
STATUS(RUNNING)              SUBSTATE(MQGET)
XMITQ(QM2)
```

Geben Sie auf dem Warteschlangenmanager QM2 den folgenden Befehl ein:

```
DISPLAY CHS(QM1.TO.QM2) SSLPEER SSLCERTI
```

Die Ausgabe sieht in etwa wie folgt aus:

```
DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
 5 : DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
CHANNEL(QM2.TO.QM1)          CHLTYPE(RCVR)
CONNAME(9.20.35.92)          CURRENT
RQMNAME(QM1)
SSLCERTI("CN=QM1,OU=<Department>,O=<Organization>,ST=<State>,C=<Country>")
SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QM1,OU=<Department>,O=<Organizati
on>,ST=<State>,C=<Country>")
STATUS(RUNNING)              SUBSTATE(RECEIVE)
XMITQ( )
```

In beiden Fällen muss der Wert von SSLPEER mit dem Wert des DN des in Schritt „2“ auf Seite 109 erstellten Partnerzertifikats übereinstimmen. Da es sich um ein selbst signiertes Zertifikat handelt, entspricht der Name des Ausstellers dem Namen der Partnerwarteschlange.

SSLPEER ist optional. Wenn dieses Attribut allerdings angegeben ist, muss sein Wert den DN des in Schritt „2“ auf Seite 109 erstellten Partnerzertifikat zulassen. Weitere Informationen zur Verwendung von SSLPEER finden Sie im Abschnitt [IBM MQ-Regeln für SSLPEER-Werte](#).

Von Zertifizierungsstelle signierte Zertifikate für die gegenseitige Authentifizierung zweier Warteschlangenmanager verwenden

Folgen Sie diesen Beispielanweisungen, um die gegenseitige Authentifizierung zweier Warteschlangenmanager mithilfe von TLS-Zertifikaten einer Zertifizierungsstelle zu implementieren.

Informationen zu diesem Vorgang

Szenario:

- Sie haben zwei Warteschlangenmanager mit den Namen QM1 und QM2, die auf sichere Weise miteinander kommunizieren müssen. Deswegen sollen sich QM1 und QM2 gegenseitig authentifizieren.
- Dieses Netz wollen Sie später in einer Produktionsumgebung verwenden, weshalb Sie von Anfang an CA-signierte Zertifikate verwenden möchten.

Die Konfiguration wird danach so aussehen:

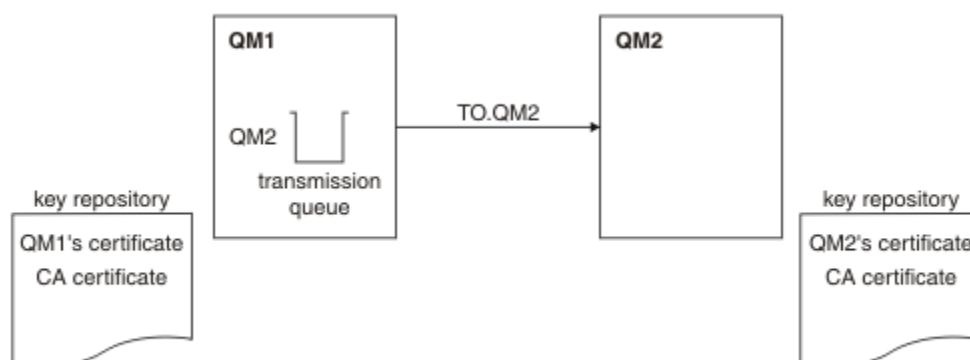














Abbildung 17. Aus dieser Task entstehende Konfiguration

In [Abbildung 17](#) auf [Seite 111](#) enthält das Schlüsselrepository von QM1 das Zertifikat für QM1 sowie das Zertifikat der Zertifizierungsstelle. Das Schlüsselrepository für QM2 enthält das QM2-Zertifikat und das CA-Zertifikat.

In diesem Beispiel wurden die Zertifikate von QM1 und QM2 von der gleichen Zertifizierungsstelle ausgestellt. Wenn die beiden CA-Zertifikate von zwei verschiedenen Zertifizierungsstellen ausgestellt worden wären, müssten die Schlüsselrepositories von QM1 und QM2 beide Zertifikate enthalten.

Vorgehensweise

1. Bereiten Sie das Schlüsselrepository auf beiden Warteschlangenmanagern entsprechend des Betriebssystems bzw. der Systeme vor, die in Ihrem Unternehmen verwendet werden:
 -  [Auf IBM i-Systemen](#)
 -  [Auf AIX, Linux, and Windows-Systemen.](#)
 -  [Auf z/OS-Systemen.](#)
2. Fordern Sie für jeden Warteschlangenmanager ein von einer Zertifizierungsstelle signiertes Zertifikat an.
Sie können die Zertifikate von zwei verschiedenen Zertifizierungsstellen anfordern.
 -  [Auf IBM i-Systemen](#)
 -  [Auf AIX, Linux, and Windows-Systemen.](#)
 -  [Auf z/OS-Systemen.](#)
3. Fügen Sie das CA-Zertifikat zum Schlüsselrepository des jeweiligen Warteschlangenmanagers hinzu:
Wenn Sie für die Warteschlangenmanager verschiedene Zertifizierungsstellen verwenden, müssen Sie das CA-Zertifikat jeder Zertifizierungsstelle beiden Schlüsselrepositories hinzufügen.
 -  [Führen Sie diesen Schritt nicht auf IBM i-Systemen aus.](#)
 -  [Auf AIX, Linux, and Windows-Systemen.](#)
 -  [Auf z/OS-Systemen.](#)
4. Laden Sie das von einer Zertifizierungsstelle signierte Zertifikat in das Schlüsselrepository für den jeweiligen Warteschlangenmanager herunter:
 -  [Auf IBM i-Systemen](#)
 -  [Auf AIX, Linux, and Windows-Systemen.](#)

-  Auf z/OS-Systemen.
5. Definieren Sie auf QM1 einen Senderkanal und die zugehörige Übertragungswarteschlange, indem Sie einen Befehl wie den folgenden ausgeben:

```
DEFINE CHANNEL(TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP)
CONNNAME(QM2.MACH.COM) XMITQ(QM2) SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256)
DESCR('Sender channel using TLS from QM1 to QM2')

DEFINE QLOCAL(QM2) USAGE(XMITQ)
```




Dieses Beispiel verwendet die CipherSpec TLS_RSA_WITH_AES_128_CBC_SHA256. Die CipherSpecs an jedem Ende des Kanals müssen identisch sein.

6. Definieren Sie auf QM2 einen Empfängerkanal, indem Sie einen Befehl wie den folgenden ausgeben:

```
DEFINE CHANNEL(TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256) SSLCAUTH(REQUIRED)
DESCR('Receiver channel using TLS to QM2')
```

Der Kanal muss denselben Namen wie der in Schritt „5“ auf Seite 112 definierte Senderkanal haben und die gleiche CipherSpec verwenden.

7. Starten Sie den Kanal:

-  Auf IBM i-Systemen
-  Auf AIX, Linux, and Windows-Systemen.
-  Auf z/OS-Systemen.

Ergebnisse

Die Schlüsselrepositorys und Kanäle werden erstellt, wie in [Abbildung 17 auf Seite 111](#) gezeigt.

Nächste Schritte

Überprüfen Sie die erfolgreiche Ausführung der Task durch Ausgabe von DISPLAY-Befehlen. Bei erfolgreichem Abschluss der Task sieht die Ausgabe in etwa wie in den folgenden Beispielen aus.

Geben Sie auf dem Warteschlangenmanager QM1 den folgenden Befehl ein:

```
DISPLAY CHS(TO.QM2) SSLPEER SSLCERTI
```

Die Ausgabe sieht in etwa wie folgt aus:

```
DISPLAY CHSTATUS(TO.QM2) SSLPEER SSLCERTI
4 : DISPLAY CHSTATUS(TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
CHANNEL(TO.QM2) CHLTYPE(SDR)
CONNNAME(192.0.0.2) CURRENT
QMNAME(QM2)
SSLCERTI("CN=<Division> CA,OU=<Department>,O=<Organization>,ST=<State>,C=<Country>")
SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QM2,OU=<Department>,O=<Organizati
on>,ST=<State>,C=<Country>")
STATUS(RUNNING) SUBSTATE(MQGET)
XMITQ(QM2)
```

Geben Sie auf dem Warteschlangenmanager QM2 den folgenden Befehl ein:

```
DISPLAY CHS(TO.QM2) SSLPEER SSLCERTI
```

Die Ausgabe sieht in etwa wie folgt aus:


```

DISPLAY CHSTATUS(TO.QM2) SSLPEER SSLCERTI
5 : DISPLAY CHSTATUS(TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
CHANNEL (TO.QM2)                CHLTYPE(RCVR)
CONNNAME(192.0.0.1)              CURRENT
RQMNAME(QM1)
SSLCERTI("CN=<Division> CA,OU=<Department>,O=<Organization>,ST=<State>,C=<Country>")
SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QM1,OU=<Department>,O=<Organization>,ST=<State>,C=<Country>")
STATUS(RUNNING)                  SUBSTATE(RECEIVE)
XMITQ( )

```

In beiden Fällen muss der Wert von SSLPEER mit dem Wert des definierten Namens (DN) des in Schritt „2“ auf Seite 111 erstellten Partnerzertifikats übereinstimmen. Der Name des Ausstellers entspricht dem registrierten Namen des Zertifikatsinhabers des CA-Zertifikats, durch das das in Schritt „4“ auf Seite 111 hinzugefügte persönliche Zertifikat signiert wurde.

Zwei Warteschlangenmanager über unidirektionale Authentifizierung verbinden

Folgen Sie diesen Beispielanweisungen, um ein System mit gegenseitiger Authentifizierung so zu ändern, dass sich Warteschlangenmanager über die unidirektionale Authentifizierung miteinander verbinden können (für den Fall, dass der SSL/TLS-Client kein Zertifikat sendet).

Informationen zu diesem Vorgang

Szenario:

- Ihre beiden Warteschlangenmanager (QM1 und QM2) sind eingerichtet, wie in „Von Zertifizierungsstelle signierte Zertifikate für die gegenseitige Authentifizierung zweier Warteschlangenmanager verwenden“ auf Seite 110 beschrieben.
- Sie möchten QM1 so ändern, dass er sich über die unidirektionale Authentifizierung mit QM2 verbinden kann.

Die Konfiguration wird danach so aussehen:

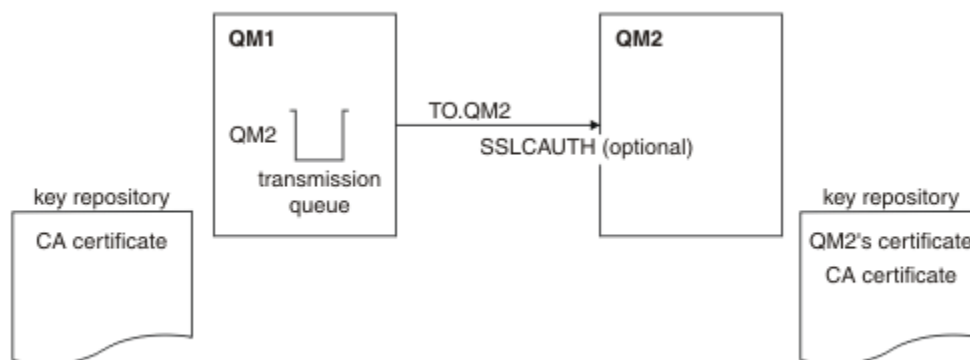



Abbildung 18. Warteschlangenmanager lassen unidirektionale Authentifizierung zu

Vorgehensweise

1. Entfernen Sie das persönliche Zertifikat von QM1 aus seinem Schlüsselrepository:

- Zertifikat auf IBM i -Systemen entfernen.
- Zertifikat entfernen von AIX, Linux, and Windows.

-  Entfernen eines Zertifikats auf z/OS-Systemen. Führen Sie diesen Schritt zweimal aus, um sowohl das persönliche Zertifikat für QMA als auch das Standardzertifikat zu entfernen.

Informationen zur Bezeichnung von Zertifikaten finden Sie im Abschnitt [Bezeichnungen für digitale Zertifikate](#).

2. Optional: Aktualisieren Sie auf QM1 die SSL/TLS-Umgebung, wenn bereits zuvor SSL/TLS-Kanäle aktiv waren, wie im Abschnitt [TLS-Umgebung aktualisieren](#) beschrieben .
3. Lassen Sie anonyme Verbindungen auf dem Empfängerkanal zu (siehe [Anonyme Verbindungen auf dem Empfängerkanal zulassen](#)) .

Die Schlüsselrepositorys und Kanäle werden geändert, wie in [Abbildung 18](#) auf Seite 113 gezeigt.

4. Wenn der Senderkanal inaktiv war, starten Sie ihn.

Anmerkung: Wenn der Senderkanal aktiv war und Sie in Schritt 2 den Befehl REFRESH SECURITY TYPE(SSL) ausgegeben haben, wird der Kanal automatisch gestartet.

Das Vorhandensein des Parameterwerts für den Peer-Namen im Kanalstatus am Serverende des Kanals ist ein Hinweis darauf, dass ein Clientzertifikat übertragen wurde.

5. Überprüfen Sie die erfolgreiche Ausführung der Task durch Ausgabe einiger DISPLAY-Befehle. Bei erfolgreichem Abschluss der Task sieht die Ausgabe in etwa wie in den folgenden Beispielen aus:

- Geben Sie auf dem Warteschlangenmanager QM1 den folgenden Befehl ein:

```
DISPLAY CHS(TO.QM2) SSLPEER SSLCERTI
```

Die Ausgabe sieht in etwa wie folgt aus:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
 4 : DISPLAY CHSTATUS(TO.QMB) SSLPEER
AMQ8417: Display Channel Status details.
CHANNEL(TO.QM2)                CHLTYPE(SDR)
CONNAME(192.0.0.1)             CURRENT
RQMNAME(QM2)
SSLCERTI("CN=IBM MQ CA,OU=IBM MQ Devt,O=IBM,ST=Hampshire,C=UK")
SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMB,OU=IBM MQ Development,O=IBM,ST=Hampshire,C=UK")
STATUS(RUNNING)                SUBSTATE(MQGET)
XMITQ(QM2)
```

- Geben Sie auf dem Warteschlangenmanager QM2 den folgenden Befehl ein:

```
DISPLAY CHS(TO.QM2) SSLPEER SSLCERTI
```

Die Ausgabe sieht in etwa wie folgt aus:

```
DISPLAY CHSTATUS(TO.QM2) SSLPEER SSLCERTI
 5 : DISPLAY CHSTATUS(TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
CHANNEL(TO.QM2)                CHLTYPE(RCVR)
CONNAME(192.0.0.2)             CURRENT
RQMNAME(QMA)                   SSLCERTI( )
SSLPEER( )                     STATUS(RUNNING)
SUBSTATE(RECEIVE)              XMITQ( )
```

Auf QM2 ist das Feld SSLPEER leer, da von QM1 kein Zertifikat gesendet wurde. Auf QM1 stimmt der Wert von SSLPEER mit dem DN im persönlichen Zertifikat von QM2 überein.

Client sicher mit einem WS-Manager verbinden

Für eine sichere Kommunikation, bei der die verschlüsselten Sicherheitsprotokolle TLS verwendet werden, müssen Kommunikationskanäle eingerichtet und die digitalen Zertifikate verwaltet werden, die Sie zur Authentifizierung verwenden werden.

Beim Konfigurieren Ihrer SSL/TLS-Installation müssen Sie Ihre Kanäle so definieren, dass sie TLS verwenden. Zudem müssen Sie Ihre digitalen Zertifikate anfordern und verwalten. Auf einem Testsystem können Sie selbst signierte Zertifikate verwenden, die von einer lokalen Zertifizierungsstelle ausgegeben wurden. Verwenden Sie selbst signierte Zertifikate nicht auf einem Produktionssystem.

Sie finden ausführliche Informationen zur Erstellung und Verwaltung von Zertifikaten in den folgenden Abschnitten:

- **IBM i** [Unter IBM i mit SSL oder TLS arbeiten](#)
- **ALW** [Auf AIX, Linux, and Windows-Systemen mit SSL oder TLS arbeiten](#)
- **z/OS** [Unter z/OS mit SSL oder TLS arbeiten](#)

In diesen Themen werden die Aufgaben bei der Einrichtung der SSL/TLS-Kommunikation erläutert, zudem erhalten Sie dort eine schrittweise Anleitung zum Ausführen dieser Aufgaben.

Testen Sie gegebenenfalls die SSL/TLS-Clientauthentifizierung, die ein optionaler Bestandteil der Protokolle ist. Während des SSL/TLS-Handshakes fordert der SSL/TLS-Client immer ein digitales Zertifikat beim Server an und prüft es. Bei der IBM MQ-Implementierung fordert der SSL/TLS-Server immer ein Zertifikat beim Client an.

IBM i **ALW** Auf IBM i, AIX, Linux, and Windows-Systemen sendet der SSL/TLS-Client nur dann ein Zertifikat, wenn es im richtigen IBM MQ-Format, d. h. entweder `ibmwebsphere.mq`, gefolgt von Ihrer Anmeldebenutzer-ID in Kleinbuchstaben oder dem Wert des Attributs **CERTLABL**, eine Kennzeichnung enthält. Siehe [Bezeichnungen für digitale Zertifikate](#).

Wenn ein Clientzertifikat gesendet wird, wird es immer vom SSL/TLS-Server geprüft. Wenn der Client kein Zertifikat sendet, schlägt die Authentifizierung nur dann fehl, wenn für das Ende des Kanals, das als SSL/TLS-Server fungiert, entweder der Parametersatz **SSLCAUTH** auf REQUIRED oder ein **SSLPEER**-Parameterwert eingestellt ist. Weitere Informationen zur anonymen Verbindung eines Warteschlangenmanagers finden Sie unter [„Client anonym mit einem Warteschlangenmanager verbinden“](#) auf Seite 119.

Zugehörige Konzepte

[TLS CipherSpecs und Cipher-Suites in IBM MQ classes for Java](#)

[TLS CipherSpecs und Cipher-Suites in IBM MQ classes for JMS](#)

Zugehörige Tasks

[Zertifikate für den verwalteten .NET-Client verwenden](#)

Selbst signierte Zertifikate für die gegenseitige Authentifizierung von Client und Warteschlangenmanager verwenden

Folgen Sie diesen Beispielanweisungen, um die gegenseitige Authentifizierung zwischen einem Client und einem Warteschlangenmanager mithilfe von selbst signierten TLS-Zertifikaten zu implementieren.

Informationen zu diesem Vorgang

IBM i Unter IBM i unterstützt DCM keine selbst signierten Zertifikate, weshalb diese Task auf IBM i-Systemen nicht anwendbar ist.

Szenario:

- Sie haben Client C1 und Warteschlangenmanager QM1, die sicher miteinander kommunizieren müssen. Deswegen sollen sich C1 und QM1 gegenseitig authentifizieren.
- Die sichere Kommunikation möchten Sie mit selbst signierten Zertifikaten testen.

Die Konfiguration wird danach so aussehen:

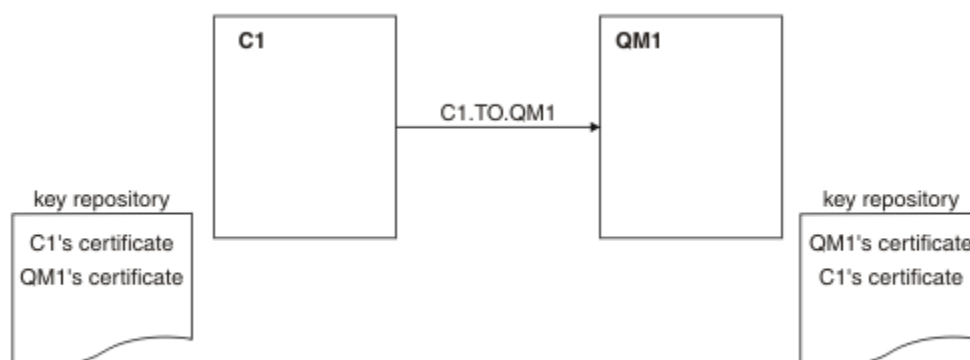


Abbildung 19. Aus dieser Task entstehende Konfiguration

In [Abbildung 19](#) auf Seite 116 enthält das Schlüsselrepositorium von QM1 das Zertifikat für QM1 und das öffentliche Zertifikat von C1. Das Schlüsselrepositorium von C1 enthält das Zertifikat für C1 und das öffentliche Zertifikat von QM1.

Vorgehensweise

1. Bereiten Sie das Schlüsselrepositorium auf dem Client und dem Warteschlangenmanager entsprechend des Betriebssystems vor:
 - **ALW** Auf AIX, Linux, and Windows-Systemen.
 - **z/OS** Auf z/OS-Systemen (nur Warteschlangenmanager).
2. Erstellen Sie für den Client und den Warteschlangenmanager ein selbst signiertes Zertifikat:
 - **ALW** Auf AIX, Linux, and Windows-Systemen.
 - **z/OS** Auf z/OS-Systemen (nur Warteschlangenmanager).
3. Extrahieren Sie eine Kopie jedes Zertifikats:
 - **ALW** Auf AIX, Linux, and Windows-Systemen.
 - **z/OS** Auf z/OS-Systemen.
4. Übertragen Sie den öffentlichen Teil des C1-Zertifikats auf das QM1-System und umgekehrt, indem Sie ein Dienstprogramm wie FTP verwenden.
 - **z/OS** Informationen zu z/OS finden Sie unter [Selbst signierte Zertifikate ändern](#).
5. Fügen Sie auf dem Client und dem Warteschlangenmanager jeweils das Partnerzertifikat zum Schlüsselrepositorium hinzu:
 - **ALW** Auf AIX, Linux, and Windows-Systemen.
 - **z/OS** Auf z/OS-Systemen.
6. Geben Sie den Befehl `REFRESH SECURITY TYPE(SSL)` auf dem Warteschlangenmanager aus.
7. Definieren Sie mit einer der folgenden Methoden einen Clientverbindungskanal:
 - Verwendung des `MQCONN`-Aufrufs mit der `MQSCo`-Struktur auf C1, wie im Abschnitt [Clientverbindungskanal auf dem IBM MQ MQI client mit MQCNO erstellen](#) beschrieben.
 - Mit einer Clientkanal-Definitionstabelle, wie in [Serververbindungs- und Clientverbindungsdefinitionen auf dem Server erstellen](#) beschrieben.
8. Definieren Sie auf QM1 einen Serververbindungskanal, indem Sie einen Befehl wie den folgenden ausgeben:

```
DEFINE CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using TLS from C1 to QM1')
```

Der Kanal muss denselben Namen wie der in Schritt 6 definierte Clientverbindungskanal haben und die gleiche CipherSpec verwenden.

Ergebnisse

Die Schlüsselrepositorys und Kanäle werden erstellt, wie in [Abbildung 19](#) auf Seite 116 gezeigt.

Nächste Schritte

Überprüfen Sie, ob die Task erfolgreich ausgeführt wurde, indem Sie **DISPLAY**-Befehle verwenden. Bei erfolgreichem Abschluss der Task sieht die Ausgabe in etwa wie im folgenden Beispiel aus.

Geben Sie auf dem Warteschlangenmanager QM1 den folgenden Befehl ein:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
```

Die Ausgabe sieht in etwa wie folgt aus:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
CHANNEL(C1.TO.QM1)          CHLTYPE(SVRCONN)
CONNAME(192.0.0.1)          CURRENT
SSLCERTI("CN=QM1,OU=IBM MQ Development,O=IBM,ST=Hampshire,C=UK")
SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5E:02,CN=QM2,OU=IBM MQ Deve
lopment,O=IBM,ST=Hampsh
shire,C=UK")
STATUS(RUNNING)             SUBSTATE(RECEIVE)
```

Er ist optional, um das **SSLPEER**-Filterattribut der Kanaldefinitionen festzulegen. Wenn die Kanaldefinition **SSLPEER** festgelegt ist, muss ihr Wert mit dem SubjektDN in dem Partnerzertifikat übereinstimmen, das in Schritt 2 erstellt wurde. Nach einer erfolgreichen Verbindung zeigt das Feld **SSLPEER** in der Ausgabe von **DISPLAY CHSTATUS** den SubjektDN des fernen Clientzertifikats an.

Von Zertifizierungsstelle signierte Zertifikate für die gegenseitige Authentifizierung von Client und Warteschlangenmanager verwenden

Folgen Sie diesen Beispielanweisungen, um die gegenseitige Authentifizierung zwischen einem Client und einem Warteschlangenmanager mithilfe von TLS-Zertifikaten einer Zertifizierungsstelle zu implementieren.

Informationen zu diesem Vorgang

Szenario:

- Sie haben Client C1 und Warteschlangenmanager QM1, die sicher miteinander kommunizieren müssen. Deswegen sollen sich C1 und QM1 gegenseitig authentifizieren.
- Dieses Netz wollen Sie später in einer Produktionsumgebung verwenden, weshalb Sie von Anfang an CA-signierte Zertifikate verwenden möchten.

Die Konfiguration wird danach so aussehen:

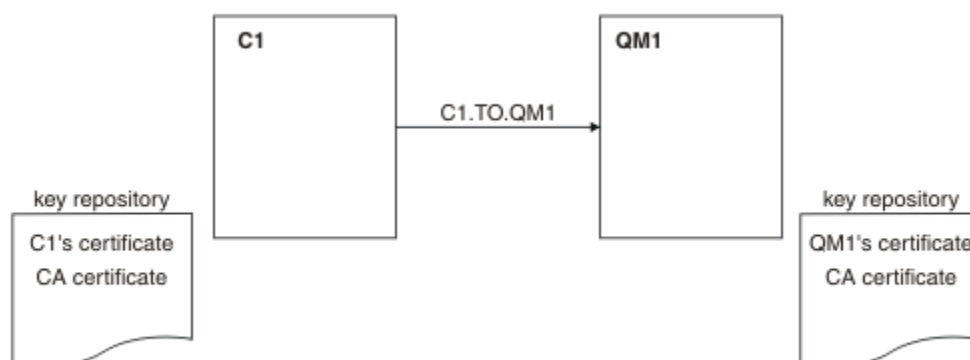


Abbildung 20. Aus dieser Task entstehende Konfiguration

In [Abbildung 20](#) auf Seite 118 enthält das Schlüsselrepository von C1 das Zertifikat für C1 sowie das Zertifikat der Zertifizierungsstelle. Das Schlüsselrepository von QM1 enthält das Zertifikat für QM1 und das Zertifikat der Zertifizierungsstelle. In diesem Beispiel wurden die Zertifikate von C1 und QM1 von der gleichen Zertifizierungsstelle ausgestellt. Wenn die beiden CA-Zertifikate von zwei verschiedenen Zertifizierungsstellen ausgestellt worden wären, müssten die Schlüsselrepositorys von C1 und QM1 beide Zertifikate enthalten.

Vorgehensweise

1. Bereiten Sie das Schlüsselrepository auf dem Client und dem Warteschlangenmanager entsprechend des Betriebssystems vor:

- **IBM i** [Auf IBM i-Systemen](#)
- **ALW** [Auf AIX, Linux, and Windows-Systemen.](#)
- **z/OS** [Auf z/OS-Systemen](#) (nur Warteschlangenmanager).

2. Fordern Sie für den Client und den Warteschlangenmanager ein von einer Zertifizierungsstelle signiertes Zertifikat an.

Sie können die Zertifikate von zwei verschiedenen Zertifizierungsstellen anfordern.

- **IBM i** [Auf IBM i-Systemen](#)
- **ALW** [Auf AIX, Linux, and Windows-Systemen.](#)
- **z/OS** [Auf z/OS-Systemen](#) (nur Warteschlangenmanager).


3. Fügen Sie auf dem Client und dem Warteschlangenmanager jeweils das CA-Zertifikat zum Schlüsselrepository hinzu.

Wenn Sie für den Client und den Warteschlangenmanager verschiedene Zertifizierungsstellen verwenden, müssen Sie das CA-Zertifikat jeder Zertifizierungsstelle beiden Schlüsselrepositorys hinzufügen.

- **IBM i** Führen Sie diesen Schritt nicht auf IBM i-Systemen aus.
- **ALW** [Auf AIX, Linux, and Windows-Systemen.](#)
- **z/OS** [Auf z/OS-Systemen](#) (nur Warteschlangenmanager).

4. Laden Sie das von einer Zertifizierungsstelle signierte Zertifikat in das Schlüsselrepository für den Client und den Warteschlangenmanager herunter:

- **IBM i** [Auf IBM i-Systemen](#)
- **ALW** [Auf AIX, Linux, and Windows-Systemen.](#)

-  Auf z/OS-Systemen (nur Warteschlangenmanager).
- Definieren Sie mit einer der folgenden Methoden einen Clientverbindungskanal:
 - Verwendung des MQCONN-Aufrufs mit der MQSCo-Struktur auf C1, wie im Abschnitt [Clientverbindungskanal auf dem IBM MQ MQI client mit MQCNO erstellen](#) beschrieben.
 - Mit einer Clientkanal-Definitionstabelle, wie in [Serververbindungs- und Clientverbindungsdefinitionen auf dem Server erstellen](#) beschrieben.
 - Definieren Sie auf QM1 einen Serververbindungskanal, indem Sie einen Befehl wie den folgenden ausgeben:

```
DEFINE CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) TRPTYPE(TCP)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA) SSLCAUTH(REQUIRED)
DESCR('Receiver channel using TLS from C1 to QM1')
```

Der Kanal muss denselben Namen wie der in Schritt 6 definierte Clientverbindungskanal haben und die gleiche CipherSpec verwenden.

Ergebnisse

Die Schlüsselrepositorys und Kanäle werden erstellt, wie in [Abbildung 20 auf Seite 118](#) gezeigt.

Nächste Schritte

Überprüfen Sie die erfolgreiche Ausführung der Task durch Ausgabe von DISPLAY-Befehlen. Bei erfolgreichem Abschluss der Task sieht die Ausgabe in etwa wie im folgenden Beispiel aus.

Geben Sie auf dem Warteschlangenmanager QM1 den folgenden Befehl ein:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
```

Die Ausgabe sieht in etwa wie folgt aus:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
CHANNEL(C1.TO.QM1)                CHLTYPE(SVRCONN)
CONNNAME(192.0.0.1)                CURRENT
SSLCERTI("CN=IBM MQ CA,OU=IBM MQ Devt,O=IBM,ST=Hampshire,C=UK")
SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMA,OU=IBM MQ Development,O=IBM,ST=Hampshire,C=UK")
STATUS(RUNNING)                    SUBSTATE(RECEIVE)
```

Das Feld SSLPEER in der Ausgabe DISPLAY CHSTATUS zeigt den SubjektDN des fernen Clientzertifikats an, der in Schritt 2 erstellt wurde. Der Name des Ausstellers entspricht dem DN des CA-Zertifikats, das das persönliche Zertifikat, das in Schritt 4 hinzugefügt wurde, signiert hat.

Client anonym mit einem Warteschlangenmanager verbinden

Folgen Sie diesen Beispielanweisungen, um ein System mit gegenseitiger Authentifizierung so zu ändern, dass sich Warteschlangenmanager anonym miteinander verbinden können.

Informationen zu diesem Vorgang

Szenario:

- Warteschlangenmanager (QM1) und Client (C1) sind eingerichtet, wie in [„Von Zertifizierungsstelle signierte Zertifikate für die gegenseitige Authentifizierung von Client und Warteschlangenmanager verwenden“ auf Seite 117](#) beschrieben.
- Sie möchten C1 so ändern, dass er sich anonym mit QM1 verbinden kann.

Die Konfiguration wird danach so aussehen:

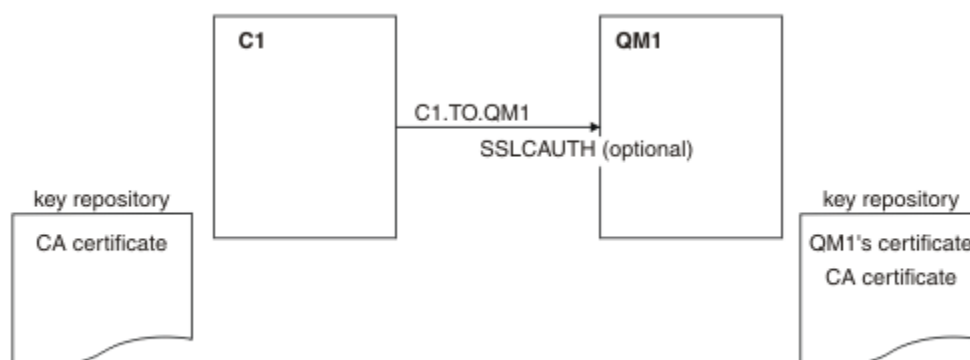




Abbildung 21. Client und Warteschlangenmanager lassen eine anonyme Verbindung zu

Vorgehensweise

1. Entfernen Sie das persönliche Zertifikat entsprechend des Betriebssystems aus dem Schlüsselrepositorium von C1:

-  IBM i -Systeme.
-  AIX, Linux, and Windows -Systeme.

Die Zertifikatsbezeichnung lautet entweder `ibmwebspheremq` gefolgt von Ihrer Benutzer-ID in Kleinbuchstaben oder sie ist der Wert des Attributs **CERTLABL**. Siehe [Bezeichnungen für digitale Zertifikate](#).

2. Starten Sie die Clientanwendung neu oder sorgen Sie dafür, dass die Clientanwendung geschlossen wird, und öffnen Sie dann alle SSL/TLS-Verbindungen neu.
3. Erlauben Sie für den Warteschlangenmanager anonyme Verbindungen, indem Sie den folgenden Befehl ausgeben:

```
ALTER CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) SSLCAUTH(OPTIONAL)
```

Ergebnisse

Die Schlüsselrepositoriums und Kanäle werden geändert, wie in [Abbildung 21 auf Seite 120](#) gezeigt.

Nächste Schritte

Das Vorhandensein des Parameterwerts für den Peer-Namen im Kanalstatus am Serverende des Kanals ist ein Hinweis darauf, dass ein Clientzertifikat übertragen wurde.

Überprüfen Sie die erfolgreiche Ausführung der Task durch Ausgabe einiger DISPLAY-Befehle. Bei erfolgreichem Abschluss der Task sieht die Ausgabe in etwa wie im folgenden Beispiel aus:

Geben Sie auf dem Warteschlangenmanager QM1 den folgenden Befehl ein:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
```

Die Ausgabe sieht in etwa wie folgt aus:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
CHANNEL(C1.TO.QM1)                CHLTYPE(SVRCONN)
CONNNAME(192.0.0.1)                CURRENT
```


SSLCERTI ()
STATUS (RUNNING)

SSLPEER ()
SUBSTATE (RECEIVE)

Die Felder SSLCERTI und SSLPEER sind leer, da von C1 kein Zertifikat gesendet wurde.

Windows Migration unter Windows

Ausgehend von einer vorhandenen IBM MQ 9.3-Installation werden Ihnen in diesem Szenario die wichtigsten Tasks zur Aktualisierung und Migration von Daten auf IBM MQ 9.4 erläutert. Beide Versionen sind in der gleichen Windows-Umgebung installiert.

Windows Lösung planen

Lesen Sie die Themen in diesem Abschnitt, um sich mit den Inhalten dieses Szenarios vertraut zu machen. Außerdem erfahren Sie, weshalb das Szenario für ein Unternehmen hilfreich sein kann und Sie erhalten eine Übersicht über die Lösung, die sich aus diesem Szenario ergibt.

Zugehörige Tasks

[Installation planen](#)

Windows Voraussetzungen

Bei diesem Szenario wird von verschiedenen Voraussetzungen in Bezug auf das System ausgegangen, auf dem die IT-Beispielkonfiguration erstellt und verwendet werden soll. Diese beziehen sich zum Beispiel auf das Betriebssystem und die Version der verwendeten Produkte und auf die Frage, ob Sie Sicherheitsfunktionen für IBM MQ konfiguriert haben.

Bei diesem Szenario wird von Folgendem ausgegangen:

- Sie verwenden für dieses Szenario einen Computer mit einem Windows-Betriebssystem, unter dem die IBM MQ 9.3-Ausgangskonfiguration und anschließend IBM MQ 9.4 installiert werden.

Anmerkung: Auf die Erstellung eines Clusters wird in diesem Szenario nicht eingegangen. Mit den Anweisungen wird eine IBM MQ-Beispielkonfiguration mit nur einem Server installiert, die als Ausgangspunkt zum Erkunden dieses Szenarios (so, wie es ursprünglich entwickelt wurde) dienen kann.

- Sie verwenden folgende Versionen von IBM MQ:
 - Für die Beispielausgangskonfiguration verwenden Sie IBM MQ 9.3.
 - Für die Konfiguration nach der Migration verwenden Sie IBM MQ 9.4.
- In diesem Szenario wird keine Sicherheitskonfiguration für IBM MQ beschrieben. Falls Sicherheitsfunktionen konfiguriert sind, sollten Sie das Szenario aber trotzdem ausführen können.
- Mit der Windows-Eingabeaufforderung und der grafischen Benutzerschnittstelle von IBM MQ Explorer können Sie die in diesem Szenario beschriebenen Tasks ausführen.

Zugehörige Konzepte

[Migrationspfade](#)

Windows Das Unternehmen

Ein Unternehmen möchte die vorhandene IBM MQ 9.3 IT-Konfiguration auf dem Betriebssystem Windows auf IBM MQ 9.4 migrieren.

Das Unternehmen beschließt eine Migration ihrer Geschäftslösung auf IBM MQ 9.4, um Vorteile wie die folgenden nutzen können:

- Nutzung neuer und aktualisierter Funktionen, die mit IBM MQ 9.4 verfügbar sind.
- Den neuen, ab IBM MQ 9.4 verfügbaren Releasetyp 'Continuous Delivery Release (CDR)' kennen lernen.
- Nutzung der LDAP-Berechtigung auf Windows-Plattformen.

Zugehörige Konzepte

[IBM MQ -Releasetypen und -Versionssteuerung](#)

Zugehörige Informationen

[IBM MQ FAQ für langfristigen Support und Continuous Delivery-Releases](#)

Migrationspfad auswählen

Bei einer Migration von IBM MQ 9.3 auf IBM MQ 9.4 können Sie zwischen mehreren Migrationspfaden wählen.

Dieser Abschnitt enthält eine Übersicht über die folgenden Migrationspfade:

- Einstufige Migration (auch als Standalone-Migration bezeichnet)
- Parallelmigration
- Parallelmigration

Hinweis: In diesem Szenario wird nur auf die einstufige Migration und die Parallelmigration eingegangen.

Wägen Sie die Vor- und Nachteile der beiden Migrationsverfahren ab, um festzustellen, welches Verfahren Ihren Anforderungen am ehesten entspricht:

Einstufige Migration

Bei einer einstufigen Migration wird eine Produktinstallation in demselben Installationsverzeichnis durch die neueste Produktversion ersetzt.

Die Einstufenmigration hat den Vorteil, dass die Konfiguration eines Warteschlangenmanagers der früheren Version so wenig wie möglich geändert wird. Bereits vorhandene Anwendungen laden statt der Bibliotheken der älteren Version automatisch die Bibliotheken der neuesten Version. Bei Anwendung dieser Methode ist Ihr System während des gesamten Prozesses nicht verfügbar.

Parallelmigration

Bei einer Parallelmigration wird die neueste Version von IBM MQ installiert, während die bereits vorhandenen Warteschlangenmanager weiterhin der früheren Version zugeordnet sind.

Wenn der richtige Zeitpunkt gekommen ist, werden die Warteschlangenmanager und Anwendungen ebenfalls auf die neueste Version migriert.

Da bei dieser Vorgehensweise die frühere Version vor dem Start eines Warteschlangenmanagers deinstalliert wird, können Sie eine Installation der neuesten Version als Primärinstallation definieren.

Weitere Informationen finden Sie im Abschnitt [Primäre Installation auswählen](#).

Mehrstufige Migration

Bei der mehrstufigen Migration wird die neueste Produktversion installiert, während die bereits vorhandenen Warteschlangenmanager weiterhin der früheren Version zugeordnet sind. Sie haben die Möglichkeit, Warteschlangenmanager zu erstellen und neue Anwendungen auszuführen, die die neueste Installationsversion verwenden. Wenn Sie zur Migration der Warteschlangenmanager und Anwendungen von der älteren Version bereit sind, können Sie sie nacheinander migrieren. Deinstallieren Sie nach Abschluss der Migration auf die neueste Version die ältere Version und definieren Sie die Installation der neuesten Version als Primärinstallation.

Bei der mehrstufigen Migration müssen Sie bis zur Deinstallation der älteren Version eine Umgebung für die Ausführung von Anwendungen konfigurieren, die eine Verbindung zu einem Warteschlangenmanager der neuesten Version herstellen. Sie müssen außerdem einen Pfad für die Ausführung von IBM MQ-Befehlen bereitstellen. Beide dieser Tasks werden mit dem Befehl `setmqenv` ausgeführt.

Zugehörige Konzepte

[Übersicht über die Konzepte und Methoden zur Migration](#)

Zugehörige Tasks

[Migration unter AIX and Linux: einstufig](#)

[Migration unter AIX and Linux: parallel](#)

[Migration unter AIX and Linux: mehrstufig](#)

Zugehörige Verweise

Liste der Änderungen, die Auswirkungen auf eine Migration haben

Windows Technische Lösung

Anhand dieses Szenarios werden zwei Methoden zur Migration von einer früheren Version von IBM MQ auf eine höhere Version beschrieben, wobei beide Versionen auf einem Windows-Betriebssystem ausgeführt werden und sich auf dem gleichen Server befinden.

Windows Übersicht: Die IT-Ausgangskonfiguration

In einem Unternehmen wird eine bestehende IT-Konfiguration verwendet, die von IBM MQ 9.3 auf einem Server mit einem Windows-Betriebssystem bereitgestellt wird. In diesem Szenario wird die Migration der IT-Ausgangskonfiguration auf eine funktional entsprechende IT-Konfiguration beschrieben, die von IBM MQ 9.4 auf dem gleichen Server bereitgestellt wird.

Die IT-Ausgangskonfiguration umfasst mehrere Komponenten, die von einem Administrator konfiguriert bzw. verwendet werden (siehe [Abbildung 22](#) auf Seite 123):

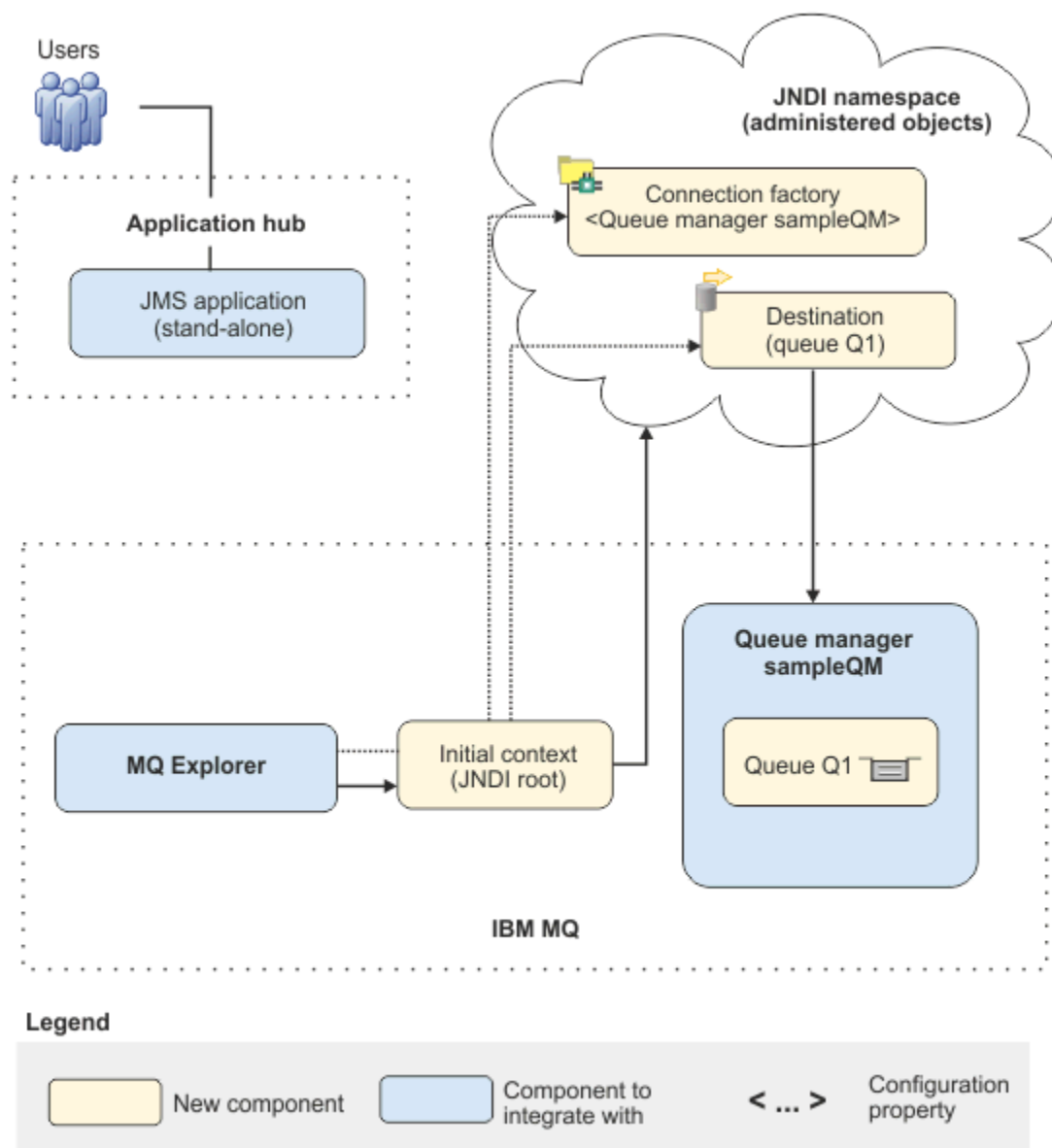


Abbildung 22. IT-Ausgangskonfiguration

JMS-Anwendung

Eine eigenständige Anwendung, mit der Fachanwender interagieren, beispielsweise um eine Bestellung einzugeben. Die Anwendung verwendet Java Message Service (JMS) für die asynchrone Nachrichtenübertragung.

- JMS ist der Java EE-Messaging-Standard, der weithin unterstützt wird. JMS-basierte Anwendungen sind daher für viele Messaging-Produkte tragbar.
- Die Abstraktion der Details der Nachrichtenübertragungsschicht erleichtert JMS die Anwendungsentwicklung.
- JMS ermöglicht die asynchrone Übertragung, sodass Anwendungen (anders als bei eng gekoppelten Systemen wie beispielsweise RPC (Remote Procedure Call)) ausgeführt werden können, ohne dass auf eine Antwort gewartet werden muss.
- Anwendungen, die JMS verwenden, geben die Details für den Zugriff auf Ressourcen nicht direkt an, sondern suchen nach verwalteten JMS-Objekten (beispielsweise eine Verbindungsfactory und ein Ziel), die sie dann verwenden.

In einigen Fällen sind andere Nachrichtenübertragungsstandards unter Umständen geeigneter als JMS. IBM Message Service Clients for C, C++ und .NET (auch als XMS bezeichnet) beispielsweise sind APIs, die ähnliche Vorteile wie JMS für Anwendungen bieten, bei denen es sich nicht um Java-Anwendungen handelt. Bei Verwendung der .NET-Plattform oder beim kombinierten Einsatz von vorhandenen C++-Anwendungen und neuen Java EE-Anwendungen ist XMS daher eher geeignet.

Die Anwendung sendet Nachrichten über Punkt-zu-Punkt-Messaging an eine Warteschlange in der Infrastruktur und verarbeitet Antwortnachrichten, um dem Fachanwender eine entsprechende Antwort bereitzustellen.

In diesem Messaging-Modell sendet eine Anwendung eine Nachricht an eine Warteschlange; eine andere Anwendung erhält diese Nachricht aus der Warteschlange und bestätigt ihren Empfang. Hierbei handelt es sich um das einfachste Messaging-Modell mit lediglich zwei Endpunkten. Es ist zudem für die Beispielanwendung des Szenarios am besten geeignet, da ein einzelner Client Informationen von einem einzelnen Server anfordert.

In einem anderen Messaging-Modell (Publish/Subscribe) veröffentlicht ein Publisher eine Nachricht für ein Nachrichtenthema. Subskribenten melden sich für das Thema an, um Nachrichten zu erhalten. Publisher und Subskribent verfügen über keine Informationen übereinander; die Nachricht geht an keinen oder auch einen oder mehrere Empfänger.

Warteschlangenmanager sampleQM

Der IBM MQ-Warteschlangenmanager, der die Ausgangsinfrastruktur für die Nachrichtenübertragung bereitstellt. Er hostet die Warteschlange, die von der JMS-Anwendung verwendet wird.

Q1 [Nachrichtenwarteschlange]

Die IBM MQ-Warteschlange, an die die JMS-Anwendung Nachrichten sendet.

JNDI-Namensbereich

Ein Java Naming Directory Interface JNDI-Namensbereich enthält verwaltete JMS-Objekte, über die Anwendungen eine Verbindung zu IBM MQ herstellen und auf Ziele zugreifen können, um Nachrichten zu senden bzw. zu empfangen.

JNDI ist Bestandteil von Java EE und ermöglicht Anwendungen ein Standardverfahren für den Zugriff auf verschiedene Namens- und Verzeichnisservices zum Abrufen von Anwendungskomponenten. So können Sie beispielsweise mithilfe von JNDI auf einen Namensservice in einem Dateisystem zugreifen, um die Adresse eines Druckerobjekts abzurufen oder um auf einen Verzeichnisservice auf einem LDAP-Server zuzugreifen, um ein Benutzerobjekt abzurufen, das ID- und Kennwortinformationen enthält. JNDI sorgt daher für eine bessere Portierbarkeit von JMS-basierten Anwendungen und erleichtert das Zusammenspiel dieser Anwendung untereinander sowie ihre Integration in bereits vorhandene Systeme. Beim JMS-Messaging werden mithilfe von JNDI Objekte gespeichert, die das Ziel einer Nachricht oder die Verbindungsfactory, über die die Verbindung zwischen Ihrer Anwendung und deren Messaging-Ziel hergestellt wird, darstellen.

Alle Anwendungen oder Prozesse mit Zugriff auf den JNDI-Namensbereich können dieselben verwalteten Objekte verwenden. Die Eigenschaften der verwalteten Objekte können in JNDI geändert werden; alle Anwendungen oder Prozesse profitieren gleichermaßen von diesen Änderungen.

Ausgangskontext

Der Ausgangskontext gibt den Startpunkt des JNDI-Namensbereichs vor. Um mit IBM MQ Explorer verwaltete Objekte erstellen und konfigurieren zu können, müssen Sie zunächst einen Ausgangskontext hinzufügen, der das Stammelement des JNDI-Namensbereichs definiert. Ebenso benötigt eine JMS-Anwendung einen Ausgangskontext, damit sie verwaltete Objekte aus dem JNDI-Namensbereich abrufen kann.

Verbindungsfactory (myCF)

Ein JMS-Verbindungsfactoryobjekt definiert eine Reihe von Standardkonfigurationseigenschaften für Verbindungen. Eine Anwendung verwendet eine Verbindungsfactory, um eine Verbindung zu IBM MQ herzustellen.

Ziel (myQueue)

Bei einem JMS-Ziel kann es sich um ein Thema oder eine Warteschlange handeln. In diesem Szenario ist eine Warteschlange das Ziel; es ist die IBM MQ-Warteschlange, an die Anwendungen Nachrichten senden und/oder aus der Anwendungen Nachrichten abrufen. Eine Anwendung durchsucht das Ziel im JNDI-Namensbereich, um eine Verbindung zur IBM MQ-Warteschlange herzustellen.

Windows Übersicht: bereitgestellte logische Topologie

Im Unternehmen wurde eine Migration von IBM MQ 9.3 auf IBM MQ 9.4 durchgeführt.

Die im Anschluss an die Migration durchgeführte IT-Konfiguration bleibt unverändert, wie im Abschnitt [Abbildung 23 auf Seite 126](#) gezeigt. Es werden der Warteschlangenmanager und die Beispielanwendung migriert, und somit kann das Unternehmen die Vorteile der neuen Funktionen in der aktuellsten Version von IBM MQ nutzen.

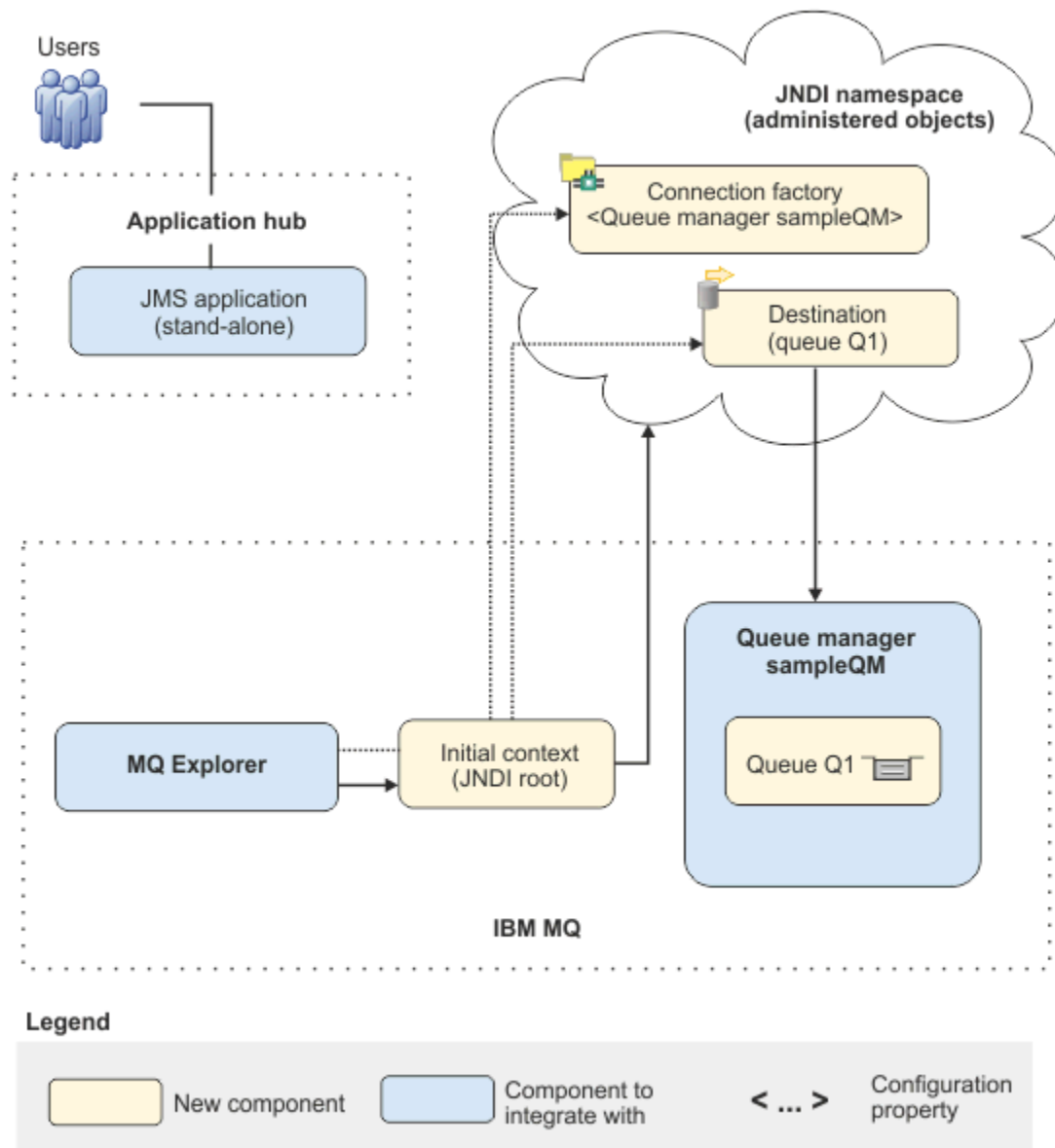


Abbildung 23. Übergebene IT-Konfiguration

Windows Lösung über die grafische Benutzerschnittstelle implementieren

Zur Implementierung der Lösung in diesem Szenario gehört die Verwendung der grafischen Benutzerschnittstelle zur Migration aus einem früheren Release von IBM MQ, das auf einem Windows-Betriebssystem ausgeführt wird, auf ein höheres Release, das ebenfalls auf einem Windows-Betriebssystem ausgeführt wird. Anschließend müssen Sie überprüfen, ob die Warteschlangenmanager und Warteschlangen erfolgreich auf das höhere Release migriert wurden.

Vorbereitende Schritte

Wenn Sie das Szenario testen möchten, führen Sie zuerst die Anweisungen zum Erstellen einer Kopie der Beispiel-Messaging-Infrastruktur im Abschnitt „Ursprüngliche IT-Konfiguration erstellen“ auf Seite 127 aus. Diese Beispielkonfiguration basiert auf IBM MQ 9.3.

Informationen zu diesem Vorgang

Bei dem in diesem Szenario beschriebenen Migrationsprozess wird die Migration eines Warteschlangenmanagers aus IBM MQ 9.3 auf IBM MQ 9.4 gezeigt.

Anhand dieses Szenarios werden zwei Migrationsverfahren veranschaulicht. Sie können entscheiden, ob Sie eines der Verfahren oder beide Verfahren testen möchten.

Option 1: Einstufenmigration

Bei der einstufigen Migration wird die frühere Version in demselben Installationsverzeichnis durch die neuere Version ersetzt.

Die Einstufenmigration hat den Vorteil, dass die Konfiguration eines Warteschlangenmanagers der früheren Version so wenig wie möglich geändert wird. Bereits vorhandene Anwendungen laden statt der Bibliotheken der älteren Version automatisch die Bibliotheken der neuesten Version. Bei Anwendung dieser Methode ist Ihr System während des gesamten Prozesses nicht verfügbar.

Option 2: Parallelmigration

Bei der Parallelmigration wird die frühere Version von IBM MQ parallel zu einer früheren Version installiert. Die Warteschlangenmanager sind weiterhin der früheren Version zugeordnet, bis Sie sie auf die neuere Version migrieren.

Da bei der Parallelmigration die frühere Version vor dem Start eines Warteschlangenmanagers in der neueren Version deinstalliert wird, können Sie die neuere Version der Installation als primäre Installation definieren.

Vorgehensweise

1. Erstellen Sie eine ursprüngliche IT-Beispielkonfiguration, die als Ausgangspunkt für das Szenario verwendet wird (siehe [„Ursprüngliche IT-Konfiguration erstellen“](#) auf Seite 127).
2. Wählen Sie das Verfahren für die Produktmigration aus und gehen Sie entsprechend den Anweisungen für das jeweilige Verfahren vor:
 - [„Option 1: Einstufenmigration“](#) auf Seite 137
 - [„Option 2: Parallelmigration“](#) auf Seite 144

Zugehörige Tasks

[Migration unter AIX and Linux: einstufig](#)

[Migration unter AIX and Linux: parallel](#)

[Einen Warteschlangenmanager unter Windows von einer früheren auf die neueste Version migrieren](#)

[Primäre Installation auswählen](#)

Ursprüngliche IT-Konfiguration erstellen

Dieses Szenario wurde mit einer Beispielausgangskonfiguration (IT-Konfiguration; "IT" steht hier für engl. "initial") erstellt. Folgen Sie den Anweisungen, um diese Beispielkonfiguration einzurichten und das Szenario so zu testen, wie es ursprünglich entwickelt wurde.

Informationen zu diesem Vorgang

Die Ausgangskonfiguration (IT-Konfiguration) für dieses Szenario, die im Abschnitt [„Übersicht: Die IT-Ausgangskonfiguration“](#) auf Seite 123 erläutert wird, umfasst einen Ausgangskontext, der hinzugefügt wurde, damit IBM MQ Explorer eine Verbindung zum Stamm des JNDI-Namespaces herstellen kann. Der JNDI-Namensbereich umfasst eine Verbindungsfactory, die zur Verwendung durch die JMS-Beispielanwendung für die Verbindung zu IBM MQ hinzugefügt wurde, sowie ein Ziel, das für die Verbindung der JMS-Beispielanwendung zu der IBM MQ-Warteschlange hinzugefügt wurde. Diese IBM MQ-Warteschlange wurde ebenfalls in die Ausgangskonfiguration (IT-Konfiguration) aufgenommen und wird von der JMS-Beispielanwendung verwendet.

Vorgehensweise

1. Installieren Sie IBM MQ 9.3 und überprüfen Sie die Installation.
2. Konfigurieren Sie den JNDI-Namensbereich und verwaltete Objekte.
3. Prüfen Sie die IT-Beispielkonfiguration.

IBM MQ 9.4 über das Launchpad installieren

Verwenden Sie den Installationsassistenten und die Assistenten, um die Version von IBM MQ zu installieren, die Sie als ursprüngliche IT-Konfiguration einrichten möchten, die als Ausgangspunkt für dieses Szenario verwendet werden soll.

Vorbereitende Schritte

Führen Sie vor Beginn dieser Aufgabe folgende Überprüfungen durch:

- Beim Installieren müssen Sie über eine lokale Administratorberechtigung verfügen.
- Achten Sie darauf, dass der Maschinenname keine Leerzeichen enthält.
- Stellen Sie sicher, dass ausreichender Plattenspeicherplatz vorhanden ist. Weitere Informationen finden Sie unter Plattenspeicherbedarf auf Multiplattformen.

Bei diesem Szenario müssen Sie nicht feststellen, ob Sie Windows-Domänenbenutzer-IDs für IBM MQ-Benutzer definieren müssen, da diese Anforderung in diesem Szenario nicht berücksichtigt wird. Weitere Informationen finden Sie im Abschnitt Active Directory und DNS-Domäne für IBM MQ erstellen.

Überprüfen Sie vor der Installation von IBM MQ, ob Ihr System die Hardware- und Softwarevoraussetzungen erfüllt. Aktuelle Informationen zu den Hardware- und Softwarevoraussetzungen finden Sie unter Systemvoraussetzungen für IBM MQ.

Informationen zu diesem Vorgang

In dieser Task werden die grundlegenden Schritte zur Installation von IBM MQ auf einem Windows-Betriebssystem beschrieben.

Die Installationsprogramme enthalten Links zu weiteren Informationen. Der Installationsprozess umfasst folgende Teile:

1. Starten Sie den Installationsprozess.
2. Verwenden Sie das Installationslaunchpad zur Prüfung und Installation der Softwarevoraussetzungen, zur Angabe der Netzinformationen sowie zum Starten des IBM MQ-Installationsassistenten.
3. Installieren Sie die Software mithilfe des IBM MQ-Installationsassistenten und starten Sie den Prepare IBM MQ Wizarden.
4. Verwenden Sie den Prepare IBM MQ Wizarden zum Starten des IBM MQ-Service.

Vorgehensweise

1. Starten Sie den Installationsprozess.

Navigieren Sie in Windows Explorer zu dem temporären Ordner, in den das Installationsimage heruntergeladen wurde, und doppelklicken Sie anschließend auf `setup.exe`.

Das Installationslaunchpad wird gestartet.

2. Prüfen und ändern Sie gegebenenfalls die Softwarevoraussetzungen und die Netzkonfiguration über das Launchpad.
 - a) Klicken Sie auf **Softwarevoraussetzungen**, um die Registerkarte **Softwarevoraussetzungen** anzuzeigen.
 - b) Prüfen Sie, ob die Softwarevoraussetzungen erfüllt werden. Der jeweilige Eintrag für die Voraussetzung muss mit einem grünen Häkchen und dem Wort OK angezeigt werden. Nehmen Sie alle angegebenen Korrekturen vor.

Anmerkung: Klicken Sie auf die Plusschaltfläche (+), um weitere Informationen zu den Voraussetzungen anzuzeigen.

- c) Klicken Sie auf **Netzkonfiguration**, um die Registerkarte **Netzkonfiguration** anzuzeigen.
- d) Klicken Sie auf das Optionsfeld **Nein**.

Anmerkung: Bei diesem Szenario wird vorausgesetzt, dass Sie keine Domänenbenutzer-ID für IBM MQ konfigurieren müssen. Falls Sie weitere Informationen zur Konfiguration von IBM MQ for Windows-Domänenbenutzern benötigen, klicken Sie auf die Schaltfläche **Weitere Informationen**.

- e) Wählen Sie auf der Registerkarte **IBM MQ-Installation** des Launchpads die Installationsprache aus und klicken Sie auf die Schaltfläche **Installationsprogramm von IBM MQ starten**, um den IBM MQ-Installationsassistenten zu starten.

Sie haben die Prüfung der Installationsanforderungen von IBM MQ abgeschlossen, alle erforderlichen Änderungen vorgenommen und den IBM MQ-Installationsassistenten gestartet.

3. Verwenden Sie den IBM MQ-Installationsassistenten zur Installation der Software und starten Sie den Prepare IBM MQ Wizarden.
 - a) Lesen Sie im IBM MQ-Installationsassistenten die Lizenzvereinbarung, aktivieren Sie das Kontrollkästchen **Lizenzvereinbarung akzeptieren** und klicken Sie dann auf **Weiter**.
 - b) Klicken Sie auf **Standard** und anschließend auf **Weiter**.
 - c) Prüfen Sie auf der Seite **Bereit zum Installieren von IBM MQ** die Installationsinformationen und klicken Sie auf **Installieren**.

Anmerkung: Beachten Sie die folgenden Angaben:

- Installationsname
- Ordner der höchsten Ebene für Programmdateien
- Ordner der höchsten Ebene für Datendateien

Folgende Komponenten werden installiert:

- IBM MQ-Server
- IBM MQ: eine grafische Schnittstelle für die Verwaltung und Überwachung der IBM MQ-Ressourcen
- Java™ und .NET Messaging und Web Services
- IBM MQ-Entwicklungstoolkit

Der Installationsprozess wird gestartet. Je nach Ihrem System kann der Installationsprozess mehrere Minuten dauern.

Am Ende des Installationsprozesses wird im Fenster IBM MQ Setup die Nachricht **Installation Wizard Completed Successfully** angezeigt.

- d) Klicken Sie auf **Fertigstellen**.

Sie haben IBM MQ erfolgreich installiert. Der Prepare IBM MQ Wizard startet automatisch und zeigt dann die Seite **'Willkommen beim Prepare IBM MQ Wizarden'** an.

4. Verwenden Sie den Assistenten zur Vorbereitung von MQ, um den IBM MQ-Dienst zu starten.

- a) Wählen Sie auf der Seite 'Willkommen beim Prepare IBM MQ Wizarden' die Option **Weiter** aus.

Die Prepare IBM MQ Wizard zeigt die Nachricht **Status: Checking IBM MQ Configuration** und einen Fortschrittsanzeiger an. Sobald der Prozess abgeschlossen ist, wird die Seite 'IBM MQ-Netzkonfiguration' angezeigt.

- b) Wählen Sie auf der Seite 'IBM MQ -Netzkonfiguration' der Prepare IBM MQ Wizard **Nein** aus.
- c) Klicken Sie auf **Weiter**.

Prepare IBM MQ Wizard zeigt eine Nachricht **Status: starting the IBM MQ Service** und einen Fortschrittsanzeiger an. Wenn der Prozess abgeschlossen ist, zeigt der Assistent die Seite 'Prepare IBM MQ Wizard wird beendet' an.

d) Wählen Sie **IBM MQ Explorer starten** aus, geben Sie an, ob Notepad zur Anzeige der Releaseinformationen gestartet werden soll, und klicken Sie anschließend auf **Fertigstellen**.

IBM MQ Explorer wird gestartet.

Sie haben IBM MQ installiert und IBM MQ Explorer gestartet.

Ergebnisse

IBM MQ ist jetzt auf Ihrem Computer installiert.

Nächste Schritte

Sie sind jetzt bereit, die in diesem Szenario verwendeten verwalteten Objekte wie im Abschnitt „[JNDI-Namensbereich und verwaltete Objekte konfigurieren](#)“ auf Seite 130 beschrieben zu erstellen.

Zugehörige Konzepte

[Hardware- und Softwarevoraussetzungen auf Windows-Systemen](#)

[Einführung in IBM MQ](#)

Zugehörige Tasks

[IBM MQ-Server unter Windows installieren](#)

[IBM MQ-Server konfigurieren](#)

JNDI-Namensbereich und verwaltete Objekte konfigurieren

Definieren Sie einen Ausgangskontext für den JNDI-Namensbereich in IBM MQ Explorer und definieren Sie anschließend im Namensbereich die verwalteten Objekte, die die Beispielanwendung verwenden kann.

Informationen zu diesem Vorgang

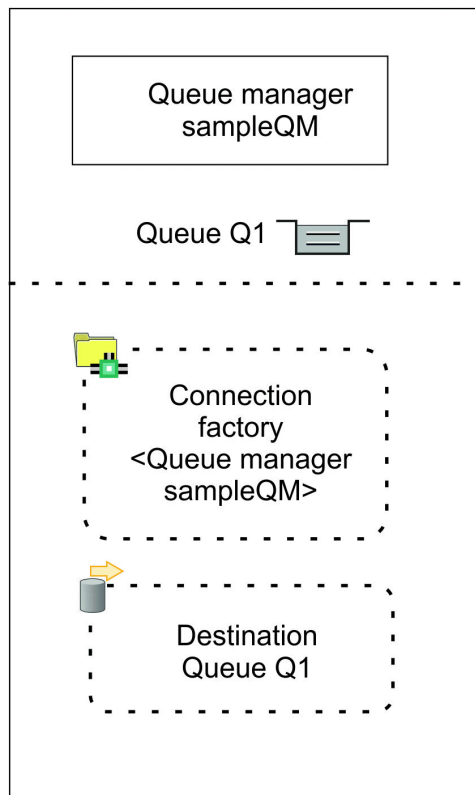
In dieser Task erstellen Sie die folgenden Objekte in IBM MQ:

- Ein JNDI-Namensbereich in einem lokalen Dateisystem. Es wird ein Dateisystem verwendet, da es sich dabei um den einfachsten JNDI-Mechanismus für ein Beispielszenario handelt.

Der JNDI-Namensbereich kann sich auf einem Dateisystem, einem LDAP-Server (LDAP = Lightweight Directory Access Protocol) oder einer anderen JNDI-Implementierung befinden. Wenn Sie einen JNDI-Namensbereich auf einem LDAP-Server oder einer anderen JNDI-Implementierung verwenden möchten, müssen Sie den JNDI-Namensbereich konfigurieren und die Beispielanwendung so ändern, dass auf den JNDI-Namensbereich verwiesen wird, da dies für die Implementierung erforderlich ist.

- Verwaltete Objekte im JNDI-Namensbereich. Die JMS-Anwendung kann die verwalteten Objekte durchsuchen, um eine Verbindung zu IBM MQ herzustellen und auf die IBM MQ-Ziele zuzugreifen, über die Nachrichten gesendet oder empfangen werden.

WebSphere MQ



WebSphere MQ JNDI Namespace

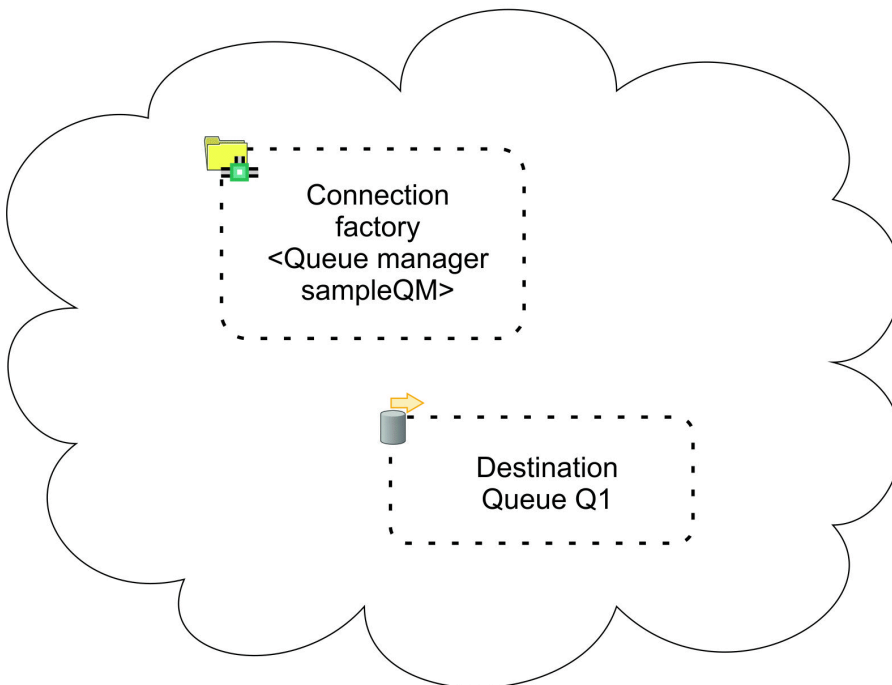


Abbildung 24. In IBM MQ erstellte Objekte

Vorgehensweise

1. Starten Sie den IBM MQ Explorer, falls dies noch nicht geschehen ist, indem Sie auf **Start > Programme > IBM MQ > IBM MQ Explorer** klicken.

Wenn IBM MQ Explorer ausgeführt wird und die Begrüßungsseite angezeigt wird, schließen Sie die Begrüßungsseite, um mit der Verwaltung der IBM MQ-Objekte zu beginnen.

2. Erstellen Sie einen Warteschlangenmanager, der von der Beispielanwendung verwendet werden kann.
 - a) Klicken Sie mit der rechten Maustaste auf **Warteschlangenmanager** und wählen Sie **Neu > Warteschlangenmanager...** aus.

Der Assistent 'Warteschlangenmanager erstellen' wird gestartet.

- b) Geben Sie im Feld **Name des Warteschlangenmanagers** den Namen `sampleQM` ein.

Sie können einen anderen Namen für den Warteschlangenmanager auswählen. müssen diesen aber auch in späteren Konfigurationsschritten anstelle von `sampleQM` verwenden.

Anmerkung: Der Name darf höchstens 48 Zeichen aus der folgenden Gruppe enthalten:

- Groß- oder Kleinbuchstaben (A-Z a-z)
- Numerische Zeichen (0-9)
- Punkt (.)
- Schrägstrich (/)
- Unterstrich (_)
- Prozentzeichen (%)

Bei Namen muss die Groß-/Kleinschreibung beachtet werden. Objekte des gleichen Typs müssen unterschiedlich benannt werden. Beispielsweise können zwei Warteschlangen nicht den gleichen Namen haben, bei einem Warteschlangenmanager und einer Warteschlange ist dies allerdings möglich.

- c) Geben Sie im Feld **Warteschlange für nicht zustellbare Nachrichten** den Namen `SYS-TEM.DEAD.LETTER.QUEUE` ein.

In diesem Feld wird der Name der Warteschlange für nicht zustellbare Nachrichten angegeben, die beim Erstellen des Warteschlangenmanagers automatisch erstellt wird.

In einer Warteschlange für nicht zustellbare Nachrichten werden Nachrichten gespeichert, die nicht an ihre eigentliche Zieladresse zugestellt werden können, weil beispielsweise die Warteschlange voll ist. Allen Warteschlangenmanagern sollte eine Warteschlange für nicht zustellbare Nachrichten zugeordnet sein.

- d) Lassen Sie die übrigen Felder leer und klicken Sie auf **Fertigstellen** oder, falls diese Schaltfläche inaktiviert ist, auf **Weiter**.

Die Schaltfläche **Fertigstellen** ist inaktiviert, wenn es bei der Portnummer einen Konflikt mit einem vorhandenen Warteschlangenmanager gibt, z. B. mit dem Warteschlangenmanager, der im Rahmen der Standardkonfiguration erstellt wird. Sie müssen mit dem Assistenten fortfahren, um die Standardportnummer zu ändern.

- e) Wenn Sie auf **Weiter** geklickt haben, akzeptieren Sie die Standardwerte und klicken Sie auf jeder Seite auf **Weiter**, bis Sie zur letzten Seite des Assistenten gelangen, auf der die Schaltfläche **Fertigstellen** dann verfügbar ist. Ändern Sie die angegebene Portnummer, z. B. in 1415, und klicken Sie auf **Fertigstellen**.

IBM MQ zeigt beim Erstellen und Starten des Warteschlangenmanagers ein Dialogfeld an.

3. Fügen Sie dem *JNDI-Namensbereich* einen Ausgangskontext hinzu und verbinden Sie IBM MQ Explorer anschließend mit diesem Kontext

Bevor Sie IBM MQ Explorer zum Erstellen und Konfigurieren von verwalteten JMS-Objekten verwenden können, müssen Sie einen Ausgangskontext zum Definieren des Stammverzeichnisses für den JNDI-Namensbereich hinzufügen, in dem die verwalteten Objekte gespeichert werden.

Sobald Sie IBM MQ Explorer zum Erstellen oder Steuern von verwalteten Objekten im JNDI-Namensbereich verwenden möchten, müssen Sie IBM MQ Explorer mit dem Ausgangskontext des JNDI-Namensbereichs verbinden.

- a) Klicken Sie im Fenster IBM MQ Explorer - **Navigator** mit der rechten Maustaste auf **JMS Verwaltete Objekte** und wählen Sie anschließend **Add Initial Context...** (Ausgangskontext hinzufügen) aus. Daraufhin wird die Seite " **Verbindungsdetails** " angezeigt.
- b) Unter " **Wo befindet sich der JNDI-Namensbereich?** " Aktivieren Sie das Kontrollkästchen **Dateisystem** .
- c) Geben Sie im Feld **Bindungsverzeichnis** C:\JNDI-Directoryein.
Dieser Wert entspricht der Position des JNDI-Namensbereichs, der in der JMS-Beispielanwendung angegeben wurde. Wenn Sie ein anderes JNDI-Verzeichnis angeben müssen, müssen Sie die Anwendung entsprechend ändern.
Wenn das Verzeichnis in Ihrem System nicht vorhanden ist, wird im Fenster die Nachricht **Specified location does not exist or is not readable** (Angegebene Position ist nicht vorhanden oder nicht lesbar) angezeigt. Klicken Sie auf **Durchsuchen ...** . Navigieren Sie zum Öffnen eines Dateisystemfensters zu **Local Disk (C:)** und klicken dann auf **Neuen Ordner erstellen** , um den Ordner **JNDI-Directory** zu erstellen. Klicken Sie auf **OK**.

Klicken Sie auf **Weiter**.

- d) Lassen Sie die Standardeinstellungen auf der Seite **User preferences** (Benutzervorgaben) unverändert.
 - **Context nickname** (Kurzname des Kontexts): Die Position des JNDI-Namensbereichs wird als Kurzname zur Anzeige des Ausgangskontexts in IBM MQ Explorer verwendet.
 - **Connect immediately on finish** (Sofort nach Fertigstellung verbinden): Durch diese Option wird IBM MQ Explorer mit dem JNDI-Namensbereich verbunden, sobald das Erstellen des Ausgangskontexts beendet ist, damit Sie sofort verwaltete Objekte erstellen können.
 - **Automatically reconnect to context on startup** (Verbindung zu Kontext bei Start automatisch wiederherstellen): Diese Option ist nicht ausgewählt, da IBM MQ Explorer normalerweise nicht automatisch bei jedem Schließen und erneuten Öffnen von IBM MQ Explorer erneut mit dem Ausgangskontext verbunden werden muss.

Wenn Sie IBM MQ Explorer regelmäßig zum Erstellen oder Steuern von verwalteten Objekten im JNDI-Namensbereich verwenden, können Sie das Kontrollkästchen **Automatically reconnect to context on startup** auswählen, damit IBM MQ Explorer automatisch mit dem Ausgangskontext verbunden wird, sobald IBM MQ Explorer gestartet wird. Durch diese Option müssen Sie die Verbindung zwischen IBM MQ Explorer und dem Ausgangskontext nicht manuell herstellen.

Klicken Sie auf **Fertigstellen**, um den Ausgangskontext zu erstellen und anzuzeigen.

4. Erstellen Sie ein von einer Verbindungsfactory verwaltetes Objekt.

Mit einem von einer Verbindungsfactory verwaltetes Objekt wird eine Gruppe von Standardkonfigurationseigenschaften für Verbindungen definiert. Eine Anwendung verwendet eine Verbindungsfactory, um eine Verbindung zu IBM MQ herzustellen.

- a) Erweitern Sie im Fenster IBM MQ Explorer - **Navigator** den Eintrag **JMS Verwaltete Objekte** und anschließend den Ausgangskontext mit der Bezeichnung **file:/C:/JNDI-Verzeichnis/**.
- b) Klicken Sie mit der rechten Maustaste auf **Connection Factories** (Verbindungsfactorys) und wählen Sie anschließend **New > Connection Factory...** (Neu > Verbindungsfactory...) aus. Diese Aktion wird im Assistenten **New Connection Factory** (Neue Verbindungsfactory) angezeigt.
- c) Geben Sie im Namensfeld myCF ein.
Die JMS-Beispielanwendung enthält Code, der eine Verbindungsfactory mit der Bezeichnung myCF sucht. Wenn Sie einen anderen Namen verwenden müssen, müssen Sie die Anwendung entsprechend ändern.

IBM MQ wird für den Messaging-Provider verwendet, da die Beispielanwendung das Punkt-zu-Punkt-Messaging nutzt.

Klicken Sie auf **Weiter**.

- d) Lassen Sie den Typ der Verbindungsfactory unverändert als **Connection Factory**, da diese Option die flexibelste Option für die allgemeine Verwendung von JMS ist.

Mit einer domänenunabhängigen Verbindungsfactory können JMS-Anwendungen das Punkt-zu-Punkt-Messaging sowie das Publish/Subscribe-Messaging verwenden. Dies ist besonders hilfreich, wenn die JMS-Anwendung beide Messaging-Typen in der gleichen Transaktion ausführen soll.

Wenn eine JMS-Anwendung nur das Punkt-zu-Punkt-Messaging oder nur das Publish/Subscribe-Messaging verwenden soll, können Sie beim Erstellen der Verbindungsfactory die spezielle Messaging-Domäne auswählen, damit eine domänenspezifische Verbindungsfactory (Warteschlangen oder Thema) erstellt wird.

- e) Die Unterstützung von XA-Transaktionen soll nicht ausgewählt sein.

Die Beispielanwendung verwendet keine XA-konformen Transaktionen.

IBM MQ JMS unterstützt XA-konforme Transaktionen im Bindungsmodus. Wenn die Beispielanwendung XA-konforme Transaktionen verwenden soll, müssen Sie die Beispielanwendung entsprechend ändern.

Klicken Sie auf **Weiter**.

- f) Behalten Sie den Transportmodus **Bindings** (Bindungen) bei.

Die JMS-Beispielanwendung, die die Verbindungsfactory verwendet, wird auf dem gleichen Computer wie der Warteschlangenmanager ausgeführt, deshalb können Sie den Transportmodus für Bindungen verwenden. Mit dieser Option kann die JMS-Anwendung eine direkte Verbindung zum Warteschlangenmanager herstellen, was ein Vorteil hinsichtlich der Leistung gegenüber dem alternativen Clientmodus darstellt.

Klicken Sie auf **Weiter** und anschließend erneut auf **Weiter**.

- g) Wählen Sie auf der Seite **Change properties** (Eigenschaften ändern) im linken Menü **Connection** (Verbindung) aus und wählen Sie anschließend im Fenster **Connection** den Eintrag 'sampleQM' als **Basiswarteschlangenmanager** aus.

Beim Basiswarteschlangenmanager handelt es sich um den Warteschlangenmanager, zu dem die Anwendung eine Verbindung herstellt. Lassen Sie diesen Wert leer, wenn die Anwendung eine Verbindung zu mehreren Warteschlangenmanagern herstellen soll.

- h) Klicken Sie auf **Fertigstellen**.

IBM MQ zeigt ein Dialogfenster an, um zu betätigen, dass das Objekt erfolgreich erstellt wurde. Klicken Sie auf **OK**, um das Dialogfenster zu schließen.

5. Erstellen Sie ein vom Ziel verwaltetes Objekt.

Mit einem vom Ziel verwalteten Objekt wird die IBM MQ-Warteschlange angegeben, an die Anwendungen Nachrichten senden und/oder von der eine Anwendung Nachrichten empfängt. Eine Anwendung durchsucht das Ziel im JNDI-Namensbereich, um eine Verbindung zur IBM MQ-Warteschlange herzustellen.

Beim Publish/Subscribe-Messaging ermittelt das Ziel keine Warteschlange, sondern ein Thema.

- a) Erweitern Sie im Fenster IBM MQ Explorer - **Navigator** den Eintrag **JMS Verwaltete Objekte** und anschließend den Ausgangskontext mit der Bezeichnung **file:/C:/JNDI-Verzeichnis/**.
- b) Klicken Sie mit der rechten Maustaste auf **Destinations** (Ziele) und wählen Sie anschließend **New > Destination...** (Neu > Ziel...) aus.
- Der Assistent **New Destination** (Neues Ziel) wird angezeigt.
- c) Geben Sie im Feld **Name** den Wert myQueueein.

Behalten Sie den **Typ Queue** (Warteschlange) bei.

Die JMS-Beispielanwendung enthält Code, der ein Ziel mit der Bezeichnung myQueue sucht. Die JMS-Beispielanwendung verwendet das Punkt-zu-Punkt-Messaging, deshalb ist ein Ziel mit dem Typ 'Queue' erforderlich. Ziele mit dem Typ 'Topic' (Thema) werden für das Publish/Subscribe-Messaging verwendet.

- d) Wählen Sie das Kontrollkästchen **Start wizard to create a matching MQ Queue** (Assistenten zum Erstellen einer übereinstimmenden MQ-Warteschlange starten) aus.

Das Zielobjekt benötigt eine übereinstimmende IBM MQ-Warteschlange, und mit IBM MQ Explorer können beide Warteschlangen gemeinsam erstellt werden. Wenn Sie den Assistenten **New Destination** abgeschlossen haben, wird der Assistent **Create an MQ Queue** (MQ-Warteschlange erstellen) geöffnet, wobei viele Zieldetails der IBM MQ-Warteschlange zugeordnet sind.

Klicken Sie auf **Weiter**.

Klicken Sie erneut auf **Weiter**.

- e) Klicken Sie auf der Seite "**Eigenschaften ändern**" auf **Auswählen ...** neben **Warteschlangenmanager**. Wählen Sie den Warteschlangenmanager 'sampleQM' aus, den Sie bereits erstellt haben, und klicken Sie anschließend auf **OK**.

- f) Geben Sie Q1 als Name für die IBM MQ-Warteschlange an.

Sie können einen anderen Namen für die Warteschlange auswählen. müssen diesen aber auch in späteren Konfigurationsschritten anstelle von Q1 verwenden.

Anmerkung: Der Name darf höchstens 48 Zeichen aus der folgenden Gruppe enthalten:

- Groß- oder Kleinbuchstaben (A-Z a-z)
- Numerische Zeichen (0-9)
- Punkt (.)
- Schrägstrich (/)
- Unterstrich (_)
- Prozentzeichen (%)

Bei Namen muss die Groß-/Kleinschreibung beachtet werden. Objekte des gleichen Typs müssen unterschiedlich benannt werden. Beispielsweise können zwei Warteschlangen nicht den gleichen Namen haben, bei einem Warteschlangenmanager und einer Warteschlange ist dies allerdings möglich.

- g) Klicken Sie auf **Fertigstellen**.

Der Assistent **Create an MQ Queue** (MQ-Warteschlange erstellen) wird gestartet.

Wenn der Assistent nicht gestartet wird, haben Sie möglicherweise in einem früheren Schritt das Kontrollkästchen **Start wizard to create a matching MQ Queue** (Assistent zum Erstellen einer übereinstimmenden MQ-Warteschlange starten) nicht ausgewählt. Erweitern Sie im Teilfenster 'IBM MQ Explorer - **Navigator**' den **BeispielQM** -Warteschlangenmanager, klicken Sie mit der rechten Maustaste auf **Warteschlangen** und wählen Sie dann **Neu > Lokale Warteschlange ...** aus.

6. Erstellen Sie eine übereinstimmende IBM MQ-Warteschlange.

Das bereits erstellte, von einem Ziel verwaltete Objekt stellt eine IBM MQ-Warteschlange dar. In dieser Warteschlange werden die JMS-Nachrichten gespeichert.

- a) Klicken Sie auf **Weiter**, um den bereits angegebenen Warteschlangenmanager 'sampleQM' zu akzeptieren.
- b) Klicken Sie auf **Weiter**.
- c) Klicken Sie auf **Fertigstellen**, um die IBM MQ-Warteschlange mit den Informationen aus dem vom Ziel verwalteten Objekt zu erstellen, das zu einem früheren Zeitpunkt erstellt wurde.

IBM MQ zeigt ein Dialogfenster mit der Nachricht an, dass das Objekt erfolgreich erstellt wurde.

Die neue Warteschlange wird im Abschnitt **Queues** (Warteschlangen) unter dem Warteschlangenmanager angezeigt.

Ergebnisse

Sie haben nun die IBM MQ-Objekte erstellt, die zur Verwendung der JMS-Beispielanwendung erforderlich sind.

Nächste Schritte

Sie können jetzt überprüfen, ob IBM MQ ordnungsgemäß für die Verwendung der Beispielanwendung wie unter „[Beispiel-IT-Konfiguration überprüfen](#)“ auf Seite 136 beschrieben konfiguriert wurde.

Beispiel-IT-Konfiguration überprüfen

Führen Sie die eigenständige JMS-Beispielanwendung aus, um Nachrichten über IBM MQ zu senden und zu empfangen, und überprüfen Sie, ob Sie IBM MQ für die Verwendung mit der Beispielanwendung ordnungsgemäß konfiguriert haben.

Vorbereitende Schritte

Laden Sie das Paket mit der Beispielanwendung herunter. Klicken Sie auf den folgenden Link und speichern Sie die Datei auf dem Computer, auf dem Sie IBM MQ installieren: [sampleJMSApp.zip](#). Extrahieren Sie anschließend die Inhalte. Das Paket enthält eine Beispieldatei der JMS -Anwendung `.jar` und Stapeldateien für die Ausführung der Anwendung.

- Die `sampleJMSApp.jar`-Beispieldatei und die `.cmd`-Dateien müssen sich im selben Verzeichnis befinden.
- Die `.cmd`-Dateien verwenden Umgebungsvariablen, um den Klassenpfad für die Ausführung der JMS -Anwendung festzulegen. Wenn bei der Ausführung der JMS-Anwendung der Java-Fehler `java.lang.NoClassDefFoundError` ausgegeben wird, müssen Sie möglicherweise die Klassenpfadzeile in der Befehlsdatei anpassen.

Informationen zu diesem Vorgang

Die JMS-Anwendung enthält einen Requester-Client, der die Startnachricht sendet, sowie einen Responder-Client, der die Nachricht empfängt und eine Antwort sendet. Die bereitgestellten Batchdateien führen die folgenden Aktionen aus:

- `runresponder.cmd` öffnet ein Fenster mit Eingabeaufforderung, in dem der Responder-Client gestartet wird und dann auf eine Nachricht wartet.
- `runrequester.cmd` öffnet ein separates Fenster mit Eingabeaufforderung, in dem der Requester-Client startet und eine Anforderungsnachricht sendet und eine Antwort erhält.

Mit zwei Fenstern mit Eingabeaufforderungen können Sie die Aktionen von Requester und Responder getrennt und eindeutiger anzeigen.

Vorgehensweise

1. Klicken Sie doppelt auf die Datei `runresponder.cmd`.

Im Eingabeaufforderungsfenster mit der Bezeichnung **Responder-Fenster** wird der Responder-Client gestartet, der dann auf eine Nachricht wartet.

```
> Connection factory located in JNDI.> Destination located in JNDI.> Creating connection to
QueueManager.> Created connection.
> Waiting for message.
```

2. Klicken Sie doppelt auf die Datei `runrequester.cmd`.

Überwachen Sie im **Requester-Fenster** die Requester-Nachrichten. Überprüfen Sie im **Responder window** (Responder-Fenster) die aktualisierten Responder-Nachrichten (die Nachricht vom Requester-Client, die empfangen wurde, und die Antwortnachricht, die gesendet wurde).

Ergebnisse

Im Fenster mit Eingabeaufforderung mit der Bezeichnung **Requester-Fenster** zeigt der Requester-Client den Verbindungsstatus, die gesendete Nachricht und anschließend die Antwortnachricht an, die vom Responder-Client empfangen wurde:

```
> Connection factory located in JNDI.> Destination located in JNDI.> Creating connection to
QueueManager.> Connection created.
> Sending stock request for 'BakedBeans'> Sent Message
```



```

ID=ID:414d5120514d5f4c33344c3238482020c3cd094d20002b02
> Received Message ID=ID:414d5120514d5f4c33344c3238482020c3cd094d20002902 for 'B
akedBeans - 15 tins in stock'
> Closing connection to QueueManager.> Closed Connection.
-----
In this window, observe the messages sent through IBM MQ:
- The request message sent
- The reply message received
-----
When ready, press any key to close this window
Press any key to continue . . .

```

Überwachen Sie im **Responder-Fenster** die aktualisierten Responder-Nachrichten: die Nachricht, die empfangen wird (vom Requester-Client), und die Antwortnachricht, die gesendet wird:

```

> Connection factory located in JNDI.> Destination located in JNDI.> Creating connection to
QueueManager.> Created connection.
> Waiting for message.

> Received Message ID=ID:414d5120514d5f4c33344c3238482020c3cd094d20002b02 for 'B
akedBeans'
> Sending Reply Message 'BakedBeans - 15 tins in stock'> Sent Message
ID=ID:414d5120514d5f4c33344c3238482020c3cd094d20002902
> Closing connection to QueueManager.> Closed connection.
-----
In this window, observe the updated responder messages
- The request message received (from the requester)
- The reply message sent
-----
When ready, press any key to close this window
Press any key to continue . . .

```

Aus den in den beiden Befehlsfenstern angezeigten Nachrichten geht hervor, dass der Requester-Client und der Responder-Client der Beispielanwendung über IBM MQ miteinander kommunizieren können.

Nächste Schritte

Sie können jetzt die Migration Ihrer IBM MQ 9.3-Beispielinstallation auf ein späteres Release von IBM MQ mit einer der beiden folgenden Migrationsoptionen starten:

- Für die Migration mit der einstufigen Migrationsmethode befolgen Sie die Anweisungen unter „[Option 1: Einstufenmigration](#)“ auf Seite 137.
- Für die Migration mit der Parallelmigrationsmethode befolgen Sie die Anweisungen unter „[Option 2: Parallelmigration](#)“ auf Seite 144.

Option 1: Einstufenmigration

In Option 1 dieses Szenarios wird die Migration aus einer früheren Version von IBM MQ auf eine spätere Version gezeigt, wenn die Methode zur einstufigen Migration verwendet wird. Bei der einstufigen Migration ersetzt die Installation einer späteren Version von IBM MQ eine frühere Version an der gleichen Installationsposition.

Vorbereitende Schritte

Der Ausgangspunkt für dieses Szenario ist die ursprüngliche IT-Konfiguration, die in „[Übersicht: Die IT-Ausgangskonfiguration](#)“ auf Seite 123 beschrieben wird.

Bevor Sie mit dieser Task beginnen, befolgen Sie die Anweisungen in „[Ursprüngliche IT-Konfiguration erstellen](#)“ auf Seite 127, um eine ursprüngliche IT-Konfiguration zu einzurichten.

Informationen zu diesem Vorgang

Bei einer einstufigen Migration können Sie die frühere Version von IBM MQ vor der Installation der neueren Version deinstallieren oder die neuere Version ohne Deinstallation der früheren Version installieren (d. h. migrieren durch Ersetzen). In beiden Fällen wird das höhere Release in demselben Verzeichnis wie das Vorabrelease installiert. Option 1 dieses Szenarios veranschaulicht eine einstufige Migration, bei der die frühere Version vor Installation der neueren Version deinstalliert wird. Die Warteschlangenmana-

gerdaten bleiben bei der Deinstallation erhalten, d. h., die in diesem Szenario verwendeten Beispielwarteschlangenmanager werden beibehalten und nach der Installation der neueren Version von IBM MQ erkannt.

Vorgehensweise

1. Stoppen Sie die Warteschlangenmanager, die in der früheren Version von IBM MQ ausgeführt wurden, und sichern Sie die Warteschlangenmanagerdaten.
2. Deinstallieren Sie die frühere Version von IBM MQ, aus der Sie die Migration ausführen, ohne die Warteschlangenmanagerdaten zu löschen.
3. Installieren Sie IBM MQ 9.4 über das Launchpad.
4. Verwenden Sie den IBM MQ Explorer, um die neue IBM MQ 9.4-Installation zu überprüfen.
Stellen Sie sicher, dass die Warteschlangenmanager erfolgreich aus dem früheren Release migriert wurden und Sie Nachrichten in die migrierten Warteschlangen einreihen und von dort abrufen können.

Zugehörige Tasks

[Migration unter AIX and Linux: einstufig](#)

Migration vorbereiten

Vor der Migration auf eine neuere Version von IBM MQ müssen Sie zunächst den Warteschlangenmanager stoppen und ein Backup für die Warteschlangenmanagerdaten durchführen.

Informationen zu diesem Vorgang

Wenn Sie eine Migration von einer früheren Version von IBM MQ vornehmen, ohne zuvor ein Systembackup durchzuführen, haben Sie keine Möglichkeit mehr, das System auf die vorherige Version zurückzusetzen, wenn die Migration doch nicht fortgesetzt werden soll. Wenn Sie vor der Installation der neuen Version ein Systembackup vornehmen, können Sie den Upgrade bei Bedarf zurücksetzen. Aktionen wie beispielsweise Änderungen an Nachrichten und Objekten, die von der neueren Version von IBM MQ vorgenommen wurden, können beim Zurücksetzen eines Upgrades allerdings nicht wiederhergestellt werden.

Stoppen Sie vor dem Backup den Warteschlangenmanager, für den der Backup durchgeführt werden soll (in diesem Szenario 'sampleQM'). Wenn Sie ein Backup für einen aktiven Warteschlangenmanager durchführen, kann es aufgrund von Aktualisierungen, die während dem Kopieren der Dateien vorgenommen wurden, zu Inkonsistenzen kommen.

Vorgehensweise

1. Öffnen Sie IBM MQ Explorer.
Klicken Sie auf **Start > Alle Anwendungen > IBM MQ > IBM MQ Explorer**.
2. Stoppen Sie den Warteschlangenmanager 'sampleQM'.
 - a) Klicken Sie in der Navigatoransicht mit der rechten Maustaste auf den Warteschlangenmanager 'sampleQM'.
 - b) Klicken Sie auf **Stoppen**.
Das Fenster **Warteschlangenmanager beenden** wird geöffnet.
 - c) Wählen Sie **Gesteuert** aus und klicken Sie auf **OK**.
Bei Auswahl der Option Gesteuert wird der Warteschlangenmanager ordnungsgemäß heruntergefahren. Die Option **Sofort**, bei der eine Beendigung des Warteschlangenmanagers erzwungen wird, wird normalerweise nur verwendet, wenn keine gesteuerte Beendigung möglich ist.
Der Warteschlangenmanager wird gestoppt. In IBM MQ wird das Symbol neben dem Warteschlangenmanager 'sampleQM' jetzt mit einem roten Abwärtspfeil angezeigt.
3. Schließen Sie IBM MQ Explorer.
4. Erstellen Sie eine Sicherungskopie der Warteschlangenmanagerdaten.

Erstellen Sie eine Kopie der folgenden Daten und stellen Sie sicher, dass auch alle Sicherungsverzeichnisse berücksichtigt werden. Einige dieser Verzeichnisse sind möglicherweise leer; wenn das Backup aber wiederhergestellt werden sollte, benötigen Sie alle Verzeichnisse, daher sollten Sie die leeren Verzeichnisse ebenfalls sichern.

- Die in C:\ProgramData\IBM\MQ\Qmgrs enthaltene(n) WS-Manager-Daten.
- Die Protokolldateiverzeichnisse für die Warteschlangenmanager, die sich in C:\ProgramData\IBM\MQ\log befinden, einschließlich der Protokollsteuerdatei amqh1ctl.lfh.
- Die Konfigurationsdateien, die sich in C:\ProgramData\IBM\MQ\Config befinden.
- Die INI-Datei und die Registry-Einträge von IBM MQ 9.3. Die Warteschlangenmanagerinformationen werden in der INI-Datei gespeichert; anhand dieser Informationen ist es möglich, eine frühere Version des Produkts wiederherzustellen.

5. Halten Sie IBM MQ an.

- a) Stoppen Sie den Service IBM MQ.

Klicken Sie dazu in der Taskleiste mit der rechten Maustaste auf das Symbol für **IBM MQ** und klicken Sie anschließend auf **Stoppen IBM MQ**.

Ein Dialogfenster mit der folgenden Nachricht wird angezeigt:

Beim Herunterfahren von IBM MQ werden alle aktiven Warteschlangenmanager und IBM MQ-Prozesse beendet. Soll der Vorgang fortgesetzt werden? (AMQ4102)

- b) Klicken Sie auf **Ja**; IBM MQ wird jetzt gestoppt.

- c) Klicken Sie nach der Beendigung von IBM MQ in der Taskleiste mit der rechten Maustaste auf das Symbol von **IBM MQ** und klicken Sie anschließend auf **Beenden**.

Ergebnisse

Der Warteschlangenmanager, der auf ein höheres Release von IBM MQ migriert werden soll, wurde gestoppt und es wurde ein Backup für die Warteschlangenmanagerdaten durchgeführt.

Nächste Schritte

Sie können IBM MQ jetzt wie im Abschnitt „[Frühere Version deinstallieren](#)“ auf Seite 139 beschrieben deinstallieren.

Zugehörige Tasks

[WS-Manager-Daten sichern](#)

Frühere Version deinstallieren

Deinstallieren Sie die frühere Version über die Systemsteuerung. Bei einer einstufigen Migration unter Windows ist die Deinstallation der früheren Version des Produkts vor der Installation der aktuellen Version optional.

Vorbereitende Schritte

Vor Ausführung dieser Schritte müssen Sie zunächst die Warteschlangenmanager stoppen, IBM MQ Explorer schließen und IBM MQ stoppen (siehe „[Migration vorbereiten](#)“ auf Seite 138).

Informationen zu diesem Vorgang

In dieser Task deinstallieren Sie IBM MQ mit dem Steuerfenster Windows. Die Warteschlangenmanagerdaten werden während des Deinstallationsprozesses nicht entfernt, d. h., die in diesem Szenario verwendeten Beispielwarteschlangenmanager bleiben bestehen und können bei der Installation der neueren Version des Produkts erkannt werden.

Ob Sie die frühere Version des Produkts deinstallieren müssen, bevor Sie die höhere Version installieren, hängt von Ihrem Betriebssystem ab. Auf Windows-Systemen ist die Deinstallation optional, d. h., Sie können die neuere Version installieren, ohne zuerst die frühere Version zu deinstallieren. In diesem Fall

werden allerdings bei der Installation teilweise andere Optionen und Nachrichten als bei einer vorherigen Deinstallation der früheren Version angezeigt. Weitere Informationen zu den Plattformen, auf denen Sie die frühere Version deinstallieren müssen, bevor Sie die neuere Version installieren, finden Sie unter [Migration auf AIX and Linux: einstufig](#).

Vorgehensweise

1. Öffnen Sie die Windows-Systemsteuerung, indem Sie auf **Start > Systemsteuerung > Programme deinstallieren**.
2. Suchen Sie im Fenster **Programme und Funktionen** den Eintrag für die Installation, die entfernt werden soll, wie beispielsweise IBM WebSphere MQ (Installation1), und klicken Sie auf **Deinstallieren**.
Der Deinstallationsprozess wird durchgeführt. Wenn der Prozess abgeschlossen ist, wird die frühere Version von IBM MQ von Ihrem Computer entfernt und wird nicht mehr in der Liste der Programme angezeigt.

Ergebnisse

Die frühere Version von IBM MQ wurde von Ihrem Computer entfernt. Die Warteschlangenmanagerdaten allerdings sind nach wie vor vorhanden.

Nächste Schritte

Sie sind jetzt bereit, die neuere Version von IBM MQ wie in „[IBM MQ 9.4 über das Launchpad installieren](#)“ auf Seite 140 beschrieben zu installieren.

Zugehörige Tasks

[IBM MQ auf Windows-Systemen deinstallieren](#)

IBM MQ 9.4 über das Launchpad installieren

Verwenden Sie den Installationsassistenten und die Assistenten, um die neuere Version von IBM MQ auf demselben Windows-Computer zu installieren, auf dem auch die frühere Version installiert wurde.

Vorbereitende Schritte

Bevor Sie mit dieser Task beginnen, müssen Sie die komprimierte Datei, die das Installationsimage enthält, herunterladen und anschließend in einem temporären Verzeichnis dekomprimieren.

Bei diesen Schritten wird davon ausgegangen, dass die ältere Version von IBM MQ, von der migriert werden soll, wie im Abschnitt „[Frühere Version deinstallieren](#)“ auf Seite 139 beschrieben deinstalliert wurde. Wenn Sie die neuere Version installieren, ohne zuvor die frühere Version zu deinstallieren, werden einige der Optionen und Nachrichten, die während des Installationsprozesses angezeigt werden, von denen abweichen, die in dieser Task beschrieben sind.

Führen Sie vor Beginn dieser Aufgabe folgende Überprüfungen durch:

- Beim Installieren müssen Sie über eine lokale Administratorberechtigung verfügen. Definieren Sie diese Berechtigung über die Windows-Funktionen.
- Achten Sie darauf, dass der Maschinenname keine Leerzeichen enthält.
- Stellen Sie sicher, dass ausreichender Plattenspeicherplatz vorhanden ist. Weitere Informationen finden Sie unter [Plattenspeicherbedarf auf Multiplattformen](#).

Bei diesem Szenario müssen Sie nicht feststellen, ob Sie Windows-Domänenbenutzer-IDs für IBM MQ-Benutzer definieren müssen, da diese Anforderung in diesem Szenario nicht berücksichtigt wird. Weitere Informationen finden Sie im Abschnitt [Active Directory und DNS-Domäne für IBM MQ erstellen](#).

Überprüfen Sie vor der Installation von IBM MQ, ob Ihr System die Hardware- und Softwarevoraussetzungen erfüllt. Aktuelle Informationen zu den Hardware- und Softwarevoraussetzungen finden Sie unter [Systemvoraussetzungen für IBM MQ](#).

Informationen zu diesem Vorgang

In dieser Task werden die grundlegenden Schritte für die Installation von IBM MQ auf einem Windows-Betriebssystem beschrieben, wenn Sie eine Migration von einer früheren Version durchführen.

Anmerkung: Die Standardpositionen für die Programm- und Datenverzeichnisse sind für IBM MQ 9.0 und höhere Versionen identisch. Daher müssen Sie die Spezifikation des Programms und der Datenverzeichnisse nicht ändern, wenn Sie von IBM MQ 9.0 auf eine neuere Version migrieren. Bei der Migration von Versionen vor IBM MQ 9.0 gibt es jedoch Unterschiede in den Standardpositionen, die berücksichtigt werden müssen. Weitere Informationen finden Sie unter [Positionen von Programm- und Datenverzeichnis unter Windows](#).

Die Installationsprogramme enthalten Links zu weiteren Informationen, die während des Installationsprozesses hilfreich sein können. Der Installationsprozess umfasst folgende Teile:

1. Überprüfen Sie über das Launchpad die Softwarevoraussetzungen und installieren Sie alle erforderlichen Komponenten, geben Sie die Netzinformationen an und starten Sie den IBM MQ-Installationsassistenten.
2. Verwenden Sie den IBM MQ-Installationsassistenten zur Installation der Software und starten Sie den Prepare IBM MQ Wizarden.
3. Verwenden Sie den Prepare IBM MQ Wizarden zum Starten des IBM MQ-Service.

Vorgehensweise

1. Starten Sie den Installationsprozess.

Navigieren Sie in Windows Explorer zu dem temporären Ordner, in den das Installationsimage heruntergeladen wurde, und doppelklicken Sie anschließend auf `setup.exe`.

Das Installationslaunchpad wird gestartet.

2. Starten Sie das Launchpad und prüfen Sie die Softwarevoraussetzungen und die Netzkonfiguration, damit Sie eventuell erforderliche Änderungen vornehmen können.
 - a) Navigieren Sie zum IBM MQ -Software-Verzeichnis, und klicken Sie doppelt auf die Datei `Setup.exe`, um das Launchpad zu starten.
 - b) Klicken Sie auf **Softwarevoraussetzungen**, um die Registerkarte **Softwarevoraussetzungen** anzuzeigen.
 - c) Prüfen Sie, ob die Softwarevoraussetzungen erfüllt werden. Der jeweilige Eintrag für die Voraussetzung muss mit einem grünen Häkchen und dem Wort OK angezeigt werden. Nehmen Sie alle angegebenen Korrekturen vor.

Anmerkung: Falls Sie Details zu einer Voraussetzung erfahren möchten, klicken Sie auf das Kontrollkästchen, um eine Registerkarte mit Informationen einzublenden.
 - d) Klicken Sie auf **Netzkonfiguration**, um die Registerkarte **Netzkonfiguration** anzuzeigen.
 - e) Klicken Sie auf das Optionsfeld **Nein**.

Anmerkung: Bei diesem Szenario wird vorausgesetzt, dass Sie keine Domänenbenutzer-ID für IBM MQ konfigurieren müssen. Falls Sie weitere Informationen zur Konfiguration von IBM MQ for Windows-Domänenbenutzern benötigen, klicken Sie auf die Schaltfläche **Weitere Informationen**.
 - f) Wählen Sie im Launchpad auf der Registerkarte **IBM MQ-Installation** die Installationssprache aus und klicken Sie dann auf **Installationsprogramm von IBM MQ starten**, um den Installationsassistenten von IBM MQ zu starten.

Sie haben die Einrichtung von IBM MQ abgeschlossen, indem Sie Ihre Installationsvoraussetzungen erfüllt oder angegeben haben, und Sie haben den Installationsassistenten von IBM MQ gestartet.

3. Verwenden Sie den IBM MQ-Installationsassistenten zur Installation der Software und starten Sie den Prepare IBM MQ Wizarden.

- a) Lesen Sie im IBM MQ-Installationsassistenten die Lizenzvereinbarung und klicken Sie auf das Kontrollkästchen **Lizenzvereinbarung akzeptieren** und anschließend auf **Weiter**.
- b) Klicken Sie auf **Standard** und anschließend auf **Weiter**.
- c) Überprüfen Sie auf der Seite **Bereit zum Installieren von IBM MQ** die Installationsinformationen und klicken Sie auf **Installieren**.

Zu den Installationsinformationen gehören folgende:

- Installationsname
- Ordner der höchsten Ebene für Programmdateien
- Ordner der höchsten Ebene für Datendateien

Die folgenden Komponenten werden installiert:

- IBM MQ-Server
- IBM MQ: eine grafische Schnittstelle für die Verwaltung und Überwachung der IBM MQ-Ressourcen
- Java™ und .NET Messaging und Web Services
- IBM MQ-Entwicklungstoolkit

Der Installationsprozess wird gestartet. Je nach Ihrem System kann der Installationsprozess mehrere Minuten dauern.

Am Ende des Installationsprozesses wird im Fenster IBM MQ Setup die Nachricht Installation Wizard Completed Successfully angezeigt.

- d) Klicken Sie auf **Fertigstellen**.

Sie haben IBM MQ erfolgreich installiert. Der Prepare IBM MQ Wizard startet automatisch und zeigt dann die Seite '**Willkommen beim Prepare IBM MQ Wizarden**' an.

4. Verwenden Sie den Prepare IBM MQ Wizard zum Starten des IBM MQ-Service.

- a) Wählen Sie auf der Seite 'Willkommen beim Prepare IBM MQ Wizarden' die Option **Weiter** aus.

Die Prepare IBM MQ Wizard zeigt die Nachricht Status: Checking IBM MQ Configuration und einen Fortschrittsanzeiger an. Sobald der Prozess abgeschlossen ist, wird die Seite 'IBM MQ-Netzkonfiguration' angezeigt.

- b) Wählen Sie auf der Seite 'IBM MQ -Netzkonfiguration' der Prepare IBM MQ Wizard **Neinaus**.

- c) Klicken Sie auf **Weiter**.

Prepare IBM MQ Wizard zeigt eine Nachricht Status: starting the IBM MQ Service und einen Fortschrittsanzeiger an. Wenn der Prozess abgeschlossen ist, zeigt der Assistent die Seite 'Prepare IBM MQ Wizard wird beendet' an.

- d) Wählen Sie **IBM MQ Explorer starten** aus, geben Sie an, ob Notepad zur Anzeige der Releaseinformationen gestartet werden soll, und klicken Sie auf **Fertigstellen**.

IBM MQ Explorer wird gestartet.

Ergebnisse

IBM MQ wurde installiert und IBM MQ Explorer wurde gestartet.

Nächste Schritte

Nachdem Sie nun die neuere Version von IBM MQ installiert haben, können Sie überprüfen, ob die Beispiel-Warteschlangenmanager erfolgreich migriert wurden und ob Sie in der Lage sind, Nachrichten in die migrierten Warteschlangen einzugeben und Nachrichten von ihnen abzurufen, wie in [„IBM MQ 9.4-Installation überprüfen“](#) auf Seite 143 beschrieben.

Zugehörige Konzepte

[Adressen von für den Download verfügbaren Installationsimages](#)

Zugehörige Tasks

IBM MQ-Server unter Windows installieren

IBM MQ 9.4-Installation überprüfen

Überprüfen Sie nach der Installation von IBM MQ 9.4 mithilfe von IBM MQ Explorer, ob die Warteschlangenmanager und Warteschlangen erfolgreich vom früheren Release migriert wurden. Stellen Sie anschließend sicher, dass Sie die Beispielanwendung ausführen können.

Informationen zu diesem Vorgang

Wenn der migrierte Warteschlangenmanager 'sampleQM' in der Navigatoransicht von IBM MQ Explorer angezeigt wird, überprüfen Sie, ob Sie eine Nachricht in die migrierte Warteschlange einreihen und von dort abrufen können und ob Sie die Beispielanwendung weiterhin ausführen können.

Vorgehensweise

1. Falls IBM MQ Explorer nicht aktiv ist, starten Sie es jetzt.
Klicken Sie auf **Start > Alle Programme > IBM MQ > IBM MQ Explorer**.
2. Stellen Sie sicher, dass Ihre Warteschlangenmanager erfolgreich auf die neuere Version von IBM MQ migriert wurden:
 - a) Erweitern Sie in der Navigatoransicht den Ordner **Warteschlangenmanager**.
 - b) Überprüfen Sie, ob der Warteschlangenmanager 'sampleQM' im Ordner **Warteschlangenmanager** enthalten ist.
 - c) Erweitern Sie den Warteschlangenmanager 'sampleQM', klicken Sie auf den Ordner **Warteschlangen** und stellen Sie fest, ob die Warteschlange Q1 in der 'Inhaltsansicht' angezeigt wird.
3. Starten Sie den Warteschlangenmanager 'sampleQM', falls er noch nicht aktiv ist.
 - a) Erweitern Sie in der Navigatoransicht den Knoten 'Warteschlangenmanager'.
 - b) Klicken Sie mit der rechten Maustaste auf den Warteschlangenmanager 'sampleQM' und klicken Sie auf **Starten**.
4. Überprüfen Sie, ob Sie eine Nachricht an Warteschlange Q1 senden können.
 - a) Erweitern Sie in der Navigatoransicht den Ordner **Warteschlangenmanager**.
 - b) Erweitern Sie den Warteschlangenmanager 'sampleQM' und klicken Sie auf den Ordner **Warteschlangen**.
 - c) Klicken Sie in der 'Inhaltsansicht' mit der rechten Maustaste auf die Warteschlange Q1 und klicken Sie auf **Testnachricht einreihen**.
Der Dialog **Testnachricht einreihen** wird geöffnet.
 - d) Geben Sie im Feld **Nachrichtendaten** Text ein, z. B. Hello queue!, und klicken Sie dann auf **Nachricht einreihen**.
Der Inhalt des Felds **Nachrichtendaten** wird gelöscht und die Nachricht in die Warteschlange eingereiht.
 - e) Klicken Sie auf **Schließen**.
Beachten Sie in der Inhaltsansicht, dass der **Aktuelle Warteschlangenlänge** -Wert der Warteschlange jetzt 1 ist. Wenn die Spalte "Aktuelle Warteschlangenlänge" nicht sichtbar ist, müssen Sie unter Umständen nach rechts in der Inhaltsansicht blättern.
5. Überprüfen Sie, ob Sie die Nachricht aus der Warteschlange Q1 abrufen können.
 - a) Erweitern Sie in der Navigatoransicht den Ordner **Warteschlangenmanager**.
 - b) Erweitern Sie den Warteschlangenmanager 'sampleQM' und klicken Sie auf den Ordner **Warteschlangen**.
 - c) Klicken Sie in der 'Inhaltsansicht' mit der rechten Maustaste auf die Warteschlange Q1 und klicken Sie auf **Nachrichten durchsuchen**.

Im Nachrichten-Browser wird eine Liste der Nachrichten angezeigt, die aktuell in der Warteschlange enthalten sind.

d) Doppelklicken Sie auf die letzte Nachricht, um den Eigenschaftendialog zu öffnen.

Im Feld **Nachrichtendaten** auf der Seite **Daten** des Eigenschaftendialogs wird der Inhalt der Nachricht in lesbarer Form angezeigt.

6. Überprüfen Sie, ob Sie die Beispielanwendung ausführen können.

a) Klicken Sie doppelt auf die Datei `runresponder.cmd`.

Im Eingabeaufforderungsfenster mit der Bezeichnung **Responder-Fenster** wird der Responder-Client gestartet, der dann auf eine Nachricht wartet.

```
> Connection factory located in JNDI.> Destination located in JNDI.> Creating connection to QueueManager.> Created connection.  
> Waiting for message.
```

b) Klicken Sie doppelt auf die Datei `runrequester.cmd`.

Überwachen Sie im **Requester-Fenster** die Requester-Nachrichten. Überprüfen Sie im **Responder window** (Responder-Fenster) die aktualisierten Responder-Nachrichten (die Nachricht vom Requester-Client, die empfangen wurde, und die Antwortnachricht, die gesendet wurde).

Im Fenster mit Eingabeaufforderung mit der Bezeichnung **Requester-Fenster** zeigt der Requester-Client den Verbindungsstatus, die gesendete Nachricht und anschließend die Antwortnachricht an, die vom Responder-Client empfangen wurde:

```
> Connection factory located in JNDI.> Destination located in JNDI.> Creating connection to QueueManager.> Connection created.  
> Sending stock request for 'BakedBeans'> Sent Message  
ID=ID:414d5120514d5f4c33344c3238482020c3cd094d20002b02  
> Received Message ID=ID:414d5120514d5f4c33344c3238482020c3cd094d20002902 for 'BakedBeans - 15 tins in stock'  
> Closing connection to QueueManager.> Closed Connection.
```

Beobachten Sie in diesem Fenster die Nachrichten, die über IBM MQ gesendet werden:

```
- The request message sent  
- The reply message received  
-----
```

When ready, press any key to close this window
Zum Fortfahren eine beliebige Taste drücken ...

Überwachen Sie im **Responder-Fenster** die aktualisierten Responder-Nachrichten: die Nachricht, die empfangen wird (vom Requester-Client), und die Antwortnachricht, die gesendet wird:

```
> Connection factory located in JNDI.> Destination located in JNDI.> Creating connection to QueueManager.> Created connection.  
> Waiting for message.
```

```
> Received Message ID=ID:414d5120514d5f4c33344c3238482020c3cd094d20002b02 for 'BakedBeans'  
> Sending Reply Message 'BakedBeans - 15 tins in stock'> Sent Message  
ID=ID:414d5120514d5f4c33344c3238482020c3cd094d20002902  
> Closing connection to QueueManager.> Closed connection.
```

```
-----  
In this window, observe the updated responder messages  
- The request message received (from the requester)  
- The reply message sent  
-----
```

When ready, press any key to close this window
Zum Fortfahren eine beliebige Taste drücken ...

Aus den in den beiden Befehlsfenstern angezeigten Nachrichten geht hervor, dass der Requester-Client und der Responder-Client der Beispielanwendung über IBM MQ miteinander kommunizieren können.

Ergebnisse

Sie haben erfolgreich eine Migration auf die neuere Version von IBM MQ durchgeführt.

Option 2: Parallelmigration

In Option 2 dieses Szenarios wird die Migration aus einem früheren Release von IBM MQ auf ein späteres Release gezeigt, wenn die Methode zur Parallelmigration verwendet wird. Bei der Parallelmigration wird

die neueste Version von IBM MQ neben der früheren Version installiert, aus der die Migration durchgeführt werden soll. Die Warteschlangenmanager und Anwendungen sind solange dem früheren Release zugeordnet, bis sie auf das aktuelle Release migriert werden.

Vorbereitende Schritte

Der Ausgangspunkt für dieses Szenario ist die ursprüngliche IT-Konfiguration, die in „[Übersicht: Die IT-Ausgangskonfiguration](#)“ auf Seite 123 beschrieben wird.

Bevor Sie mit dieser Task beginnen, befolgen Sie die Anweisungen in „[Ursprüngliche IT-Konfiguration erstellen](#)“ auf Seite 127, um eine ursprüngliche IT-Konfiguration zu einzurichten.

Informationen zu diesem Vorgang

Bei der in diesem Szenario beschriebenen Parallelmigration wird die neuere Version parallel zu der bereits vorhandenen früheren Version in einem anderen Verzeichnis installiert. Da die frühere Version vor dem Starten von Warteschlangenmanagern in der neueren Version deinstalliert wird, können Sie die Installation der neueren IBM MQ-Version als primäre Installation definieren. Weitere Informationen zur primären Installation finden Sie unter [Primäre Installation](#).

Vorgehensweise

1. [Installieren Sie IBM MQ 9.4 über das Launchpad und überprüfen Sie die Installation anschließend.](#)
2. [Stoppen Sie die Warteschlangenmanager, die in der früheren Version von IBM MQ ausgeführt werden.](#)
3. [Deinstallieren Sie die frühere Version von IBM MQ.](#)
4. [Legen Sie IBM MQ 9.4 als primäre Installation fest.](#)
5. Optional: [Ordnen Sie die Warteschlangenmanager IBM MQ 9.4 zu.](#)
6. [Verwenden Sie IBM MQ Explorer, um die IBM MQ 9.4-Installation zu überprüfen.](#)

Überprüfen Sie, ob die Warteschlangenmanager aus dem Vorabrelease erfolgreich migriert wurden und ob Sie in der Lage sind, Nachrichten in die migrierten Warteschlangen einzureihen bzw. aus den migrierten Warteschlangen abzurufen.

Zugehörige Tasks

[Migration unter Windows: parallel](#)

IBM MQ 9.4 über das Launchpad installieren

Verwenden Sie die Installationsassistenten und die Assistenten, um die neuere Version von IBM MQ neben der früheren Version auf Windows zu installieren.

Vorbereitende Schritte

Prüfen Sie vor Ausführung dieser Task zunächst folgende Punkte:

- Beim Installieren müssen Sie über eine lokale Administratorberechtigung verfügen. Definieren Sie diese Berechtigung über die Windows-Funktionen.
- Achten Sie darauf, dass der Maschinename keine Leerzeichen enthält.
- Stellen Sie sicher, dass der Plattenspeicherplatz (bis zu 1.005 MB) für eine vollständige Installation von IBM MQ for Windows ausreicht.
- Bestimmen Sie, ob Sie Windows -Domänenbenutzer-IDs für IBM MQ -Benutzer definieren müssen.

Überprüfen Sie vor der Installation von IBM MQ, ob Ihr System die Hardware- und Softwarevoraussetzungen erfüllt. Aktuelle Informationen über Hardware- und Softwarevoraussetzungen auf allen unterstützten Plattformen finden Sie unter [Systemvoraussetzungen für IBM MQ](#).

Informationen zu diesem Vorgang

Anhand dieses Szenarios werden die grundlegenden Schritte bei der Installation von IBM MQ unter Windows erläutert (falls das Produkt noch nicht auf dem System installiert ist).

Bei dieser Task wird vorausgesetzt, dass Sie die Standardpositionen für die Programm- und Datendateien von IBM MQ verwenden.

Anmerkung: Die Standardpositionen für die Programm- und Datenverzeichnisse sind für IBM MQ 9.0 und höhere Versionen identisch. Daher müssen Sie die Spezifikation des Programms und der Datenverzeichnisse nicht ändern, wenn Sie von IBM MQ 9.0 auf eine neuere Version migrieren. Bei der Migration von Versionen vor IBM MQ 9.0 gibt es jedoch Unterschiede in den Standardpositionen, die berücksichtigt werden müssen. Weitere Informationen finden Sie unter [Positionen von Programm- und Datenverzeichnis unter Windows](#).

Die Installationsprogramme enthalten Links zu weiteren Informationen, die während des Installationsprozesses hilfreich sein können. Der Installationsprozess umfasst folgende Teile:

1. Überprüfen Sie über das Launchpad die Softwarevoraussetzungen und installieren Sie alle erforderlichen Komponenten, geben Sie die Netzinformationen an und starten Sie den IBM MQ-Installationsassistenten.
2. Verwenden Sie den IBM MQ-Installationsassistenten zur Installation der Software und starten Sie den Prepare IBM MQ Wizarden.
3. Verwenden Sie den Prepare IBM MQ Wizarden zum Starten des IBM MQ-Service.

Vorgehensweise

1. Starten Sie den Installationsprozess.

Navigieren Sie in Windows Explorer zu dem temporären Ordner, in den das Installationsimage heruntergeladen wurde, und doppelklicken Sie anschließend auf `setup.exe`.

Das Installationslaunchpad wird gestartet.

2. Prüfen und ändern Sie gegebenenfalls die Softwarevoraussetzungen und die Netzkonfiguration über das Launchpad.
 - a) Klicken Sie auf **Softwarevoraussetzungen**, um die Registerkarte **Softwarevoraussetzungen** anzuzeigen.
 - b) Prüfen Sie, ob die Softwarevoraussetzungen erfüllt werden. Der jeweilige Eintrag für die Voraussetzung muss mit einem grünen Häkchen und dem Wort OK angezeigt werden. Nehmen Sie alle angegebenen Korrekturen vor.

Anmerkung: Klicken Sie auf die Plusschaltfläche (+), um weitere Informationen zu den Voraussetzungen anzuzeigen.
 - c) Klicken Sie auf **Netzkonfiguration**, um die Registerkarte **Netzkonfiguration** anzuzeigen.
 - d) Klicken Sie auf das Optionsfeld **Nein**.

Anmerkung: Bei diesem Szenario wird vorausgesetzt, dass Sie keine Domänenbenutzer-ID für IBM MQ konfigurieren müssen. Falls Sie weitere Informationen zur Konfiguration von IBM MQ for Windows-Domänenbenutzern benötigen, klicken Sie auf die Schaltfläche **Weitere Informationen**.
 - e) Wählen Sie auf der Registerkarte **IBM MQ-Installation** des Launchpads die Installationssprache aus und klicken Sie auf die Schaltfläche **Installationsprogramm von IBM MQ starten**, um den IBM MQ-Installationsassistenten zu starten.

Sie haben die Prüfung der Installationsanforderungen von IBM MQ abgeschlossen, alle erforderlichen Änderungen vorgenommen und den IBM MQ-Installationsassistenten gestartet.

3. Verwenden Sie den IBM MQ-Installationsassistenten zur Installation der Software und starten Sie den Prepare IBM MQ Wizarden.

Der IBM MQ-Installationsassistent überprüft, ob andere Installationen vorhanden sind, und zeigt die Upgrade- bzw. Installationsoptionen an, die für Sie zur Auswahl stehen. Bei diesem Szenario sind zwei Optionen verfügbar:

- Installieren und die vorhandene(n) Installation(en) dabei unberührt lassen
- Upgrade der 8.0.0.5-Installation 'Installation 1'
- a) Wählen Sie **Installieren und die vorhandene(n) Installation(en) dabei unberührt lassen** aus und klicken Sie auf **Weiter**.
- b) Lesen Sie die Lizenzvereinbarung und klicken Sie auf das Kontrollkästchen **I accept the terms in the license agreement** (Lizenzvereinbarung akzeptieren) und anschließend auf **Weiter**.
- c) Klicken Sie auf **Standard** und anschließend auf **Weiter**.
- d) Überprüfen Sie auf der Seite **Bereit zum Installieren von IBM MQ** die angezeigten Installationsinformationen und klicken Sie auf **Installieren**.

Zu den Installationsinformationen gehören folgende:

- Installationsname
- Ordner der höchsten Ebene für Programmdateien
- Ordner der höchsten Ebene für Datendateien

Die folgenden Komponenten werden installiert:

- IBM MQ-Server
- IBM MQ: eine grafische Schnittstelle für die Verwaltung und Überwachung der IBM MQ-Ressourcen
- Java™ und .NET Messaging und Web Services
- IBM MQ-Entwicklungstoolkit

Der Installationsprozess wird gestartet. Je nach Ihrem System kann der Installationsprozess mehrere Minuten dauern.

Am Ende des Installationsprozesses wird im Fenster IBM MQ Setup die Nachricht **Installation Wizard Completed Successfully** angezeigt.

- e) Klicken Sie auf **Fertigstellen**.

Sie haben IBM MQ erfolgreich installiert. Der Prepare IBM MQ Wizard startet automatisch und zeigt dann die Seite **'Willkommen beim Prepare IBM MQ Wizarden'** an.

4. Verwenden Sie den Prepare IBM MQ Wizarden zum Starten des IBM MQ-Service.

- a) Wählen Sie auf der Seite 'Willkommen beim Prepare IBM MQ Wizarden' die Option **Weiter** aus.

Die Prepare IBM MQ Wizard zeigt die Nachricht **Status: Checking IBM MQ Configuration** und einen Fortschrittsanzeiger an. Sobald der Prozess abgeschlossen ist, wird die Seite 'IBM MQ-Netzkonfiguration' angezeigt.

- b) Wählen Sie auf der Seite 'IBM MQ -Netzkonfiguration' der Prepare IBM MQ Wizard **Neinaus**.

- c) Klicken Sie auf **Weiter**.

Prepare IBM MQ Wizard zeigt eine Nachricht **Status: starting the IBM MQ Service** und einen Fortschrittsanzeiger an. Wenn der Prozess abgeschlossen ist, zeigt der Assistent die Seite 'Prepare IBM MQ Wizard wird beendet' an.

- d) Wählen Sie **IBM MQ Explorer starten** aus, geben Sie an, ob Notepad zur Anzeige der Releaseinformationen gestartet werden soll, und klicken Sie auf **Fertigstellen**.

IBM MQ Explorer wird gestartet.

Nächste Schritte

Sie haben die neuere Version von IBM MQ neben der früheren Version installiert, aber in einem anderen Installationsverzeichnis, und Sie haben IBM MQ Explorer gestartet.

Sie können jetzt die in der früheren Version von IBM MQ ausgeführten Warteschlangenmanager stoppen, wie im Abschnitt [„Warteschlangenmanager stoppen“](#) auf Seite 148 beschrieben.

Windows Warteschlangenmanager stoppen

Vor der Migration auf eine neuere Version von IBM MQ müssen Sie zunächst den Warteschlangenmanager stoppen und ein Backup für die Warteschlangenmanagerdaten durchführen.

Informationen zu diesem Vorgang

Stoppen Sie vor dem Backup den Warteschlangenmanager, für den das Backup durchgeführt werden soll. Wenn Sie ein Backup für einen aktiven Warteschlangenmanager durchführen, kann es aufgrund von Aktualisierungen, die während dem Kopieren der Dateien vorgenommen wurden, zu Inkonsistenzen kommen.

Vorgehensweise

1. Öffnen Sie IBM MQ Explorer.

Klicken Sie auf **Start > Alle Anwendungen > IBM MQ > IBM MQ Explorer**.

2. Stoppen Sie den Warteschlangenmanager 'sampleQM'.

- a) Klicken Sie in der Navigatoransicht mit der rechten Maustaste auf den Warteschlangenmanager 'sampleQM'.

- b) Klicken Sie auf **Stoppen**.

Das Fenster **Warteschlangenmanager beenden** wird geöffnet.

- c) Wählen Sie **Gesteuert** aus und klicken Sie auf **OK**.

Bei Auswahl der Option **Gesteuert** wird der Warteschlangenmanager ordnungsgemäß heruntergefahren. Die Option **Sofort**, bei der eine Beendigung des Warteschlangenmanagers erzwungen wird, wird normalerweise nur verwendet, wenn keine gesteuerte Beendigung möglich ist.

Der Warteschlangenmanager wird gestoppt. In IBM MQ wird das Symbol neben dem Warteschlangenmanager 'sampleQM' jetzt mit einem roten Abwärtspeil angezeigt.

3. Schließen Sie IBM MQ Explorer.

4. Erstellen Sie eine Sicherungskopie der Warteschlangenmanagerdaten.

Erstellen Sie eine Kopie der folgenden Daten und stellen Sie sicher, dass auch alle Sicherungsverzeichnisse berücksichtigt werden. Einige dieser Verzeichnisse sind möglicherweise leer; wenn das Backup aber wiederhergestellt werden sollte, benötigen Sie alle Verzeichnisse, daher sollten Sie die leeren Verzeichnisse ebenfalls sichern.

- Die in C:\ProgramData \IBM \MQ\Qmgrs enthaltene WS-Manager-Daten.
- Die Protokolldateiverzeichnisse für die Warteschlangenmanager, die sich in C:\ProgramData \IBM \MQ\log befinden, einschließlich der Protokollsteuerdatei amqh1ctl.lfh.
- Die Konfigurationsdateien, die sich in C:\ProgramData \IBM \MQ\Config befinden.
- Die INI-Datei und die Registry-Einträge von IBM MQ 9.3. Die Warteschlangenmanagerinformationen werden in der INI-Datei gespeichert; anhand dieser Informationen ist es möglich, eine frühere Version des Produkts wiederherzustellen.

5. Halten Sie IBM MQ an.

- a) Stoppen Sie den Service IBM MQ.

Klicken Sie dazu in der Taskleiste mit der rechten Maustaste auf das Symbol für **IBM MQ** und klicken Sie anschließend auf **Stoppen IBM MQ**.

Ein Dialogfenster mit der folgenden Nachricht wird angezeigt:

Beim Herunterfahren von IBM MQ werden alle aktiven Warteschlangenmanager und IBM MQ-Prozesse beendet. Soll der Vorgang fortgesetzt werden? (AMQ4102)

- b) Klicken Sie auf **Ja**; IBM MQ wird jetzt gestoppt.

- c) Klicken Sie nach der Beendigung von IBM MQ in der Taskleiste mit der rechten Maustaste auf das Symbol von **IBM MQ** und klicken Sie anschließend auf **Beenden**.

Nächste Schritte

Nachdem Sie die Warteschlangenmanager gestoppt haben, sind Sie bereit, sie Ihrer neuen Installation der neueren Version von IBM MQ zuzuordnen, wie in [„Warteschlangenmanager IBM MQ 9.4 zuordnen“](#) auf Seite 150 beschrieben.

Frühere Version deinstallieren

Deinstallieren Sie die frühere Version des Produkts über die Windows-Systemsteuerung.

Vorbereitende Schritte

Vor Ausführung dieser Schritte müssen Sie zunächst die Warteschlangenmanager stoppen, IBM MQ Explorer schließen und IBM MQ stoppen (siehe [„Warteschlangenmanager stoppen“](#) auf Seite 148).

Informationen zu diesem Vorgang

In dieser Task deinstallieren Sie IBM MQ mit dem Steuerfenster Windows . Die Warteschlangenmanagerdaten werden während des Deinstallationsprozesses nicht entfernt, d. h., die in diesem Szenario verwendeten Beispielwarteschlangenmanager bleiben bestehen und können bei der Installation der neueren Version des Produkts erkannt werden.

Vorgehensweise

1. Öffnen Sie die Windows-Systemsteuerung, indem Sie auf **Start > Systemsteuerung > Programme deinstallieren**.
2. Suchen Sie im Fenster **Programme und Funktionen** den Eintrag für die Installation, die entfernt werden soll, wie beispielsweise IBM WebSphere MQ (Installation1), und klicken Sie auf **Deinstallieren**.
Der Deinstallationsprozess wird durchgeführt. Wenn der Prozess abgeschlossen ist, wird die frühere Version von IBM MQ von Ihrem Computer entfernt und wird nicht mehr in der Liste der Programme angezeigt.

Ergebnisse

Die frühere Version des Produkts wurde von Ihrem Computer entfernt. Die Warteschlangenmanagerdaten allerdings sind nach wie vor vorhanden.

Nächste Schritte

Sie sind jetzt bereit, die neuere Version von IBM MQ zu der primären Installation zu machen, wie in [„IBM MQ 9.4 als primäre Installation festlegen“](#) auf Seite 149 beschrieben.

Zugehörige Tasks

[IBM MQ auf Windows-Systemen deinstallieren](#)

IBM MQ 9.4 als primäre Installation festlegen

Bevor Sie Warteschlangenmanager in Ihrer neuen Installation der neueren Version von IBM MQ starten, können Sie optional die neuere Version zur primären Installation machen.

Informationen zu diesem Vorgang

Auf Systemen, die das Vorhandensein mehrerer IBM MQ-Installationen unterstützen, handelt es sich bei der primären Installation um die Installation, auf die systemweite IBM MQ-Positionen verweisen. Die Verwendung einer primären Installation ist optional, jedoch praktisch.

Wenn Sie die in diesem Szenario beschriebene Parallelmigration durchführen, können Sie Ihre Installation der höheren Version des Produkts als primäre Installation definieren, weil bei einer Parallelmigration die frühere Version deinstalliert wird, bevor Warteschlangenmanager gestartet werden.

Weitere Informationen zur primären Installation finden Sie unter [Primäre Installation](#).

Vorgehensweise

1. Überprüfen Sie die primäre Installation, indem Sie den Befehl **dspmqinst** in der Eingabeaufforderung eingeben.

In der Eingabeaufforderung werden die Details aller aktuellen Installationen angezeigt. Die aktuelle primäre Installation enthält die folgende Zeile: `Primary: Yes`.

2. Ändern Sie die aktuelle primäre Installation mit dem Befehl **setmqinst**.

Geben Sie in der Eingabeaufforderung Folgendes ein:

```
setmqinst -x -n Installation_Name
```

Hierbei steht *Installation_Name*> für den Namen der aktuellen primären Installation.

Wenn der Befehl erfolgreich ist, zeigt die Eingabeaufforderung die Nachricht '*Installation_Name*' (*Filepath*) has been unset as the Primary Installation.

3. Mit dem Befehl **setmqinst** legen Sie die neue IBM MQ 9.4-Installation als primäre Installation fest. Geben Sie in der Eingabeaufforderung Folgendes ein:

```
setmqinst -i -n V9_Installation
```

Dabei ist *V9_Installation* der Name der IBM MQ 9.4 -Installation.

Wenn der Befehl erfolgreich ausgeführt wurde, wird in der Eingabeaufforderung die Nachricht '*V9_Installation*' (*Filepath*) has been set as the primary installation. You must restart the operating system to complete the update. angezeigt.

Anmerkung: Sie müssen wie in der Nachricht über die erfolgreiche Ausführung angegeben das Betriebssystem neu starten, um die Aktualisierung abzuschließen.

Nächste Schritte

Sie sind bereit, die migrierten Warteschlangenmanager der neueren Version von IBM MQ zuzuordnen, wie in „[Warteschlangenmanager IBM MQ 9.4 zuordnen](#)“ auf Seite 150 beschrieben.

Warteschlangenmanager IBM MQ 9.4 zuordnen

Verwenden Sie den Assistenten für die Übertragungswarteschlangenmanager, um den Warteschlangenmanager "sampleQM" mit Ihrer Installation der neueren Version von IBM MQ zuzuordnen.

Vorbereitende Schritte

Vor dem Starten dieser Task müssen Sie sicherstellen, dass der Warteschlangenmanager wie unter „[Warteschlangenmanager stoppen](#)“ auf Seite 148 beschrieben gestoppt wurden, da die Übertragung sonst nicht ausgeführt werden kann.

Informationen zu diesem Vorgang

Mit dem Assistenten 'Warteschlangenmanager übertragen' von IBM MQ Explorer können Sie einen oder mehrere Warteschlangenmanager aus anderen Installationen in die aktuelle Installation übertragen. Dieser Assistent hat die gleiche Funktion wie der Befehl **setmqm**, allerdings sparen Sie sich hier die Eingabe der erforderlichen Pfade und Parameter. Nur gestoppte Warteschlangenmanager können übertragen werden. Aktive Warteschlangenmanager werden nur als Referenz angezeigt.

Nachdem Sie einen Warteschlangenmanager in einer Installation für eine höhere Version des Produkts übertragen und gestartet haben, ist eine Migration zurück auf eine frühere Version nicht mehr möglich.

Vorgehensweise

1. Starten Sie IBM MQ Explorer.
Klicken Sie auf **Start > Alle Anwendungen > IBM MQ > IBM MQ Explorer**.
2. Klicken Sie in der Navigatoransicht mit der rechten Maustaste auf den Warteschlangenmanagerknoten und wählen Sie **Warteschlangenmanager übertragen** aus.
3. Klicken Sie mit der rechten Maustaste, wählen Sie den Warteschlangenmanager 'sampleQM' aus und klicken Sie anschließend auf **Übertragen**.
Der Befehl **setmqm** wird mit den ausgewählten Warteschlangenmanagern aufgerufen. Bei einer erfolgreichen Übertragung werden die übertragenen Warteschlangenmanager anschließend in der Navigationsstruktur angezeigt. Bei Problemen wird ein Dialog mit der vom Befehl ausgegebenen Fehlermeldung angezeigt.
4. Starten Sie den Warteschlangenmanager 'sampleQM'.
 - a) Erweitern Sie in der Navigatoransicht den Knoten 'Warteschlangenmanager'.
 - b) Klicken Sie mit der rechten Maustaste auf den Warteschlangenmanager und klicken Sie anschließend auf **Start**.

Ergebnisse

Sie haben den Warteschlangenmanager "sampleQM" der neueren Version von IBM MQ erfolgreich zugeordnet.

Nächste Schritte

Überprüfen Sie, ob der Warteschlangenmanager 'sampleQM' erfolgreich migriert wurde, indem Sie sicherstellen, dass Sie, wie im Abschnitt „[IBM MQ 9.4-Installation überprüfen](#)“ auf Seite 151 beschrieben, eine Nachricht an die Warteschlange senden und von dort abrufen können.

IBM MQ 9.4-Installation überprüfen

Nachdem Sie die neuere Version von IBM MQ installiert haben, verwenden Sie IBM MQ Explorer, um zu überprüfen, ob die Warteschlangenmanager und Warteschlangen erfolgreich aus dem früheren Release migriert wurden, und stellen Sie dann sicher, dass Sie die Beispielanwendung verwenden können.

Informationen zu diesem Vorgang

Wenn der migrierte Warteschlangenmanager 'sampleQM' in der Navigatoransicht von IBM MQ Explorer angezeigt wird, überprüfen Sie, ob Sie eine Nachricht in die migrierte Warteschlange einreihen und von dort abrufen können und ob Sie die Beispielanwendung weiterhin ausführen können.

Vorgehensweise

1. Falls IBM MQ Explorer nicht aktiv ist, starten Sie es jetzt.
Klicken Sie auf **Start > Alle Programme > IBM MQ > IBM MQ Explorer**.
2. Stellen Sie sicher, dass Ihre Warteschlangenmanager erfolgreich auf die neuere Version von IBM MQ migriert wurden:
 - a) Erweitern Sie in der Navigatoransicht den Ordner **Warteschlangenmanager**.
 - b) Überprüfen Sie, ob der Warteschlangenmanager 'sampleQM' im Ordner **Warteschlangenmanager** enthalten ist.
 - c) Erweitern Sie den Warteschlangenmanager 'sampleQM', klicken Sie auf den Ordner **Warteschlangen** und stellen Sie fest, ob die Warteschlange Q1 in der 'Inhaltsansicht' angezeigt wird.
3. Starten Sie den Warteschlangenmanager 'sampleQM', falls er noch nicht aktiv ist.
 - a) Erweitern Sie in der Navigatoransicht den Knoten 'Warteschlangenmanager'.

- b) Klicken Sie mit der rechten Maustaste auf den Warteschlangenmanager 'sampleQM' und klicken Sie auf **Starten**.
4. Überprüfen Sie, ob Sie eine Nachricht an Warteschlange Q1 senden können.
- Erweitern Sie in der Navigatoransicht den Ordner **Warteschlangenmanager**.
 - Erweitern Sie den Warteschlangenmanager 'sampleQM' und klicken Sie auf den Ordner **Warteschlangen**.
 - Klicken Sie in der 'Inhaltsansicht' mit der rechten Maustaste auf die Warteschlange Q1 und klicken Sie auf **Testnachricht einreihen**.
Der Dialog **Testnachricht einreihen** wird geöffnet.
 - Geben Sie im Feld **Nachrichtendaten** Text ein, z. B. Hello queue!, und klicken Sie dann auf **Nachricht einreihen**.
Der Inhalt des Felds **Nachrichtendaten** wird gelöscht und die Nachricht in die Warteschlange eingereiht.
 - Klicken Sie auf **Schließen**.
Beachten Sie in der Inhaltsansicht, dass der **Aktuelle Warteschlangenlänge** -Wert der Warteschlange jetzt 1 ist. Wenn die Spalte "Aktuelle Warteschlangenlänge" nicht sichtbar ist, müssen Sie unter Umständen nach rechts in der Inhaltsansicht blättern.
5. Überprüfen Sie, ob Sie die Nachricht aus der Warteschlange Q1 abrufen können.
- Erweitern Sie in der Navigatoransicht den Ordner **Warteschlangenmanager**.
 - Erweitern Sie den Warteschlangenmanager 'sampleQM' und klicken Sie auf den Ordner **Warteschlangen**.
 - Klicken Sie in der 'Inhaltsansicht' mit der rechten Maustaste auf die Warteschlange Q1 und klicken Sie auf **Nachrichten durchsuchen**.
Im Nachrichten-Browser wird eine Liste der Nachrichten angezeigt, die aktuell in der Warteschlange enthalten sind.
 - Doppelklicken Sie auf die letzte Nachricht, um den Eigenschaftendialog zu öffnen.
Im Feld **Nachrichtendaten** auf der Seite **Daten** des Eigenschaftendialogs wird der Inhalt der Nachricht in lesbarer Form angezeigt.
6. Überprüfen Sie, ob Sie die Beispielanwendung ausführen können.
- Klicken Sie doppelt auf die Datei `runresponder.cmd`.
Im Eingabeaufforderungsfenster mit der Bezeichnung **Responder-Fenster** wird der Responder-Client gestartet, der dann auf eine Nachricht wartet.

```
> Connection factory located in JNDI.> Destination located in JNDI.> Creating connection to QueueManager.> Created connection.
> Waiting for message.
```
 - Klicken Sie doppelt auf die Datei `runrequester.cmd`.
Überwachen Sie im **Requester-Fenster** die Requester-Nachrichten. Überprüfen Sie im **Responder window** (Responder-Fenster) die aktualisierten Responder-Nachrichten (die Nachricht vom Requester-Client, die empfangen wurde, und die Antwortnachricht, die gesendet wurde).
Im Fenster mit Eingabeaufforderung mit der Bezeichnung **Requester-Fenster** zeigt der Requester-Client den Verbindungsstatus, die gesendete Nachricht und anschließend die Antwortnachricht an, die vom Responder-Client empfangen wurde:

```
> Connection factory located in JNDI.> Destination located in JNDI.> Creating connection to QueueManager.> Connection created.
> Sending stock request for 'BakedBeans'> Sent Message
ID=ID:414d5120514d5f4c33344c3238482020c3cd094d20002b02
> Received Message ID=ID:414d5120514d5f4c33344c3238482020c3cd094d20002902 for 'BakedBeans - 15 tins in stock'
> Closing connection to QueueManager.> Closed Connection.
-----
Beobachten Sie in diesem Fenster die Nachrichten, die über IBM MQ gesendet werden:
- The request message sent
- The reply message received
-----
```


When ready, press any key to close this window
Zum Fortfahren eine beliebige Taste drücken ...

Überwachen Sie im **Responder-Fenster** die aktualisierten Responder-Nachrichten: die Nachricht, die empfangen wird (vom Requester-Client), und die Antwortnachricht, die gesendet wird:

```
> Connection factory located in JNDI.> Destination located in JNDI.> Creating connection to
QueueManager.> Created connection.
> Waiting for message.
```

```
> Received Message ID=ID:414d5120514d5f4c33344c3238482020c3cd094d20002b02 for 'B
akedBeans'
> Sending Reply Message 'BakedBeans - 15 tins in stock'> Sent Message
ID=ID:414d5120514d5f4c33344c3238482020c3cd094d20002902
> Closing connection to QueueManager.> Closed connection.
```

```
-----
In this window, observe the updated responder messages
- The request message received (from the requester)
- The reply message sent
-----
```

When ready, press any key to close this window
Zum Fortfahren eine beliebige Taste drücken ...

Aus den in den beiden Befehlsfenstern angezeigten Nachrichten geht hervor, dass der Requester-Client und der Responder-Client der Beispielanwendung über IBM MQ miteinander kommunizieren können.

Ergebnisse

Sie haben erfolgreich eine Migration auf die neuere Version von IBM MQ durchgeführt.

Nächste Schritte

Weitere Informationen zum Migrieren und Durchführen eines Upgrades finden Sie im Abschnitt [Wartung und Migration](#).

Höhere Version von IBM MQ installieren, die mit einer früheren Version unter Windows koexistieren soll

In diesem Szenario sind alle Schritte zur Installation einer Long Term Support-(LTS-)Version von IBM MQ aufgeführt, die mit einer früheren Version des Produkts koexistieren (parallel bestehen) soll. Die Schritte beinhalten auch die Installation eines Fixpacks für die höhere Version. Sie sollten immer die neueste Fixpackstufe installieren.

Übersicht über die Verwendung mehrerer Installationen

Eine Beschreibung der gemeinsamen Verwendung mehrerer Installationen sowie der in einem solchen Szenario verwendeten Hardware und Software.

Informationen zu diesem Vorgang


Wichtig bei Verwendung mehrerer Installationen von IBM MQ (multi-install) auf demselben Host ist, dass bei diesem Installationstyp die Warteschlangenmanager und Anwendungen in der früheren Version des Produkts nicht gestoppt werden müssen, wenn Aktivitäten mit der höheren Version des Produkts ausgeführt werden.

Die Installation der höheren Version des Produkts wirkt sich also nicht auf die Ausführung von Anwendungen in der früheren Version des Produkts aus. Dies ist hilfreich, wenn Sie eine mehrstufige Migration der Warteschlangenmanager der früheren Version des Produkts auf die höhere Version durchführen möchten.

In diesem Szenario ist IBM MQ 8.0.0 die frühere Version des Produkts und IBM MQ 9.1.0 die höhere Version.

Long Term Support-Version von IBM MQ verwenden

In diesem Szenario wird für die höhere Version des Produkts eine LTS-Version von IBM MQ verwendet.

 Wenn Sie eine Continuous Delivery-(CD-)Version von IBM MQ als höhere Version des Produkts verwenden, müssen Sie die verwendete CD-Version (z. B. IBM MQ 9.1.1) deinstallieren, bevor Sie eine höhere Version (z. B. IBM MQ 9.1.5) installieren. Weitere Informationen finden Sie unter [Migration von einem Continuous Delivery-Release auf ein anderes](#).

Für dieses Szenario verwendete Hardware und Software

Betriebssystem

Windows 10

Hostname: johndoe1.fyre.<yourdomainname>.com

Warteschlangenmanager

QM80

Mit IBM MQ 8.0.0 erstellt; soll auf IBM MQ 8.0.0 bleiben

QMMIG

Mit IBM MQ 8.0.0 erstellt; soll zu IBM MQ 9.1.0 migriert werden

QM910

Mit IBM MQ 9.1.0 erstellt; soll auf IBM MQ 9.1.0 bleiben

Installation einer neueren Version von IBM MQ parallel zu einer früheren Version.

Hier erfahren Sie, wie Sie IBM MQ 9.4 parallel zu der vorhandenen Version des Produkts auf demselben System installieren. Die Installation von IBM MQ 9.1 wird nicht als primäre Installation bezeichnet.

Vorbereitende Schritte

Vergewissern Sie sich, dass IBM MQ 9.1 auf Ihrem System installiert ist. Installieren Sie das Produkt unter Beachtung der Anweisungen im Abschnitt [IBM MQ-Server unter Windows installieren](#), wenn IBM MQ 9.1 noch nicht installiert ist.

Sie müssen die Option *Benutzerdefiniert* ausgewählt haben, damit Sie den Client explizit auswählen können. Weitere Informationen finden Sie unter [Installationsmethoden für Windows](#).

Außerdem müssen Sie den Befehl **setmqenv** mit dem Parameter **-n** aufgerufen und auf diese Weise als Installationsname `Installation1` festgelegt haben. Es ist hilfreich, eine Stapeldatei zum Aufrufen des Befehls zu verwenden.

Informationen zu diesem Vorgang

Alle Warteschlangenmanagerdaten werden in einer allgemeinen Verzeichnisstruktur gespeichert. Obwohl der ausführbare Code für jede Version von IBM MQ in einer anderen Verzeichnisstruktur gespeichert ist, werden die Daten für alle Warteschlangenmanager, unabhängig von der Version, gespeichert in: MQ_DATA_PATH=C:\ProgramData\IBM\MQ.

Zu installieren IBM MQ 9.4:

Vorgehensweise

1. Melden Sie sich als Administrator an.
IBM MQ 9.4 wird im Standardverzeichnis `C:\Program Files\IBM\MQ` installiert.
2. Wechseln Sie in das Verzeichnis, in dem sich die Downloaddatei befindet, z. B. `C:\downloads\mq9300`.
3. Dekomprimieren Sie die heruntergeladene Datei.
Die Dateien werden in ein neues Unterverzeichnis namens `MQServer` extrahiert.

4. Wechseln Sie zu dem neuen Verzeichnis und geben Sie den Befehl **setup.exe** aus, um das Installationsprogramm zu starten.
- a) Klicken Sie auf *Softwarevoraussetzungen*, um zu überprüfen, ob Ihr Unternehmen die erforderliche Software installiert hat.
Weitere Informationen finden Sie im Abschnitt Anforderungen unter Windows überprüfen. In diesem Szenario erfüllt das System die erforderlichen Voraussetzungen.
- b) Klicken Sie auf *Netzwerkkonfiguration*.
In diesem Szenario ist die Maschine nicht Teil einer Domäne, sodass kein Domänenbenutzer angegeben werden muss und die Antwort auf die Frage Nolautes.
- c) Klicken Sie auf *IBM MQ Installation*.
- d) Klicken Sie auf *IBM MQ-Installationsprogramm starten*.
Das Installationsprogramm erkennt, dass weitere Installationen auf dem System vorhanden sind, und zeigt die folgende Nachricht an:
Upgrade oder Installation
Upgrade der vorhandenen Installation vornehmen oder die neue Version daneben installieren
In diesem Szenario sollen die anderen Installationen intakt bleiben. Wählen Sie daher den ersten Eintrag aus: *Installieren und die vorhandene(n) Installation(en) dabei unberührt lassen*.
- e) Klicken Sie auf **Weiter** und akzeptieren Sie die Lizenzbedingungen.
- f) Wählen Sie *Benutzerdefiniert* als Installationsoption aus.
Weitere Informationen finden Sie unter Installationsmethoden für Windows.
- g) Klicken Sie auf **Weiter**.
Unter
Installationsdetails
Installationsdetails definieren
finden Sie folgende Angaben:
Installationsname
Installation2
Installationsordner für Programmdateien
C:\ProgramFiles\IBM\MQ
Die Liste der zu installierenden Komponenten wird angezeigt. Beachten Sie Folgendes:
- Manche Elemente werden nicht standardmäßig installiert.
 - Manche Elemente müssen ausgewählt werden, wenn sie installiert werden sollen.
- Wählen Sie für dieses Szenario *MQI Client* aus. Sie sollten diese Option ohnehin auswählen.
- h) Klicken Sie auf **Weiter**.
In der Anzeige *Bereit zur Installation von IBM MQ* sehen Sie eine Übersicht über den Installationsnamen, die Position, die zu installierenden Komponenten usw.
- i) Klicken Sie auf **Installieren**, um fortzufahren.
Daraufhin wird mit dem Kopieren der Dateien in die Installationsverzeichnisstruktur begonnen. Nach dem Kopieren der Dateien wird das folgende Dialogfeld angezeigt: The IBM MQ Installation Wizard has successfully installed IBM MQ.
- j) Klicken Sie auf **Fertigstellen**.
Nach der Installation wird das folgende Dialogfenster angezeigt: Welcome to the Prepare IBM MQ Wizard.
Da in diesem Szenario keine Windows -Domäne verwendet wird, akzeptieren Sie den Standardwert **Nein**.
Akzeptieren Sie den Start von IBM MQ Explorer. Wenn kein wichtiger Grund dagegen spricht, können Sie die Standardeinstellung *MQ Explorer mit einem neuen Arbeitsbereich starten* akzeptieren.

Ergebnisse

Sie haben erfolgreich eine andere Version von IBM MQ for Windows neben einer vorhandenen Version des Produkts installiert.

Nächste Schritte

Sie müssen den Befehl **setmqenv** ausführen, um die Befehle in beiden Versionen verwenden zu können. Details hierzu finden Sie im Abschnitt [„Befehl setmqenv zur Ausführung mit beiden Versionen von IBM MQ verwenden“](#) auf Seite 156.

Befehl setmqenv zur Ausführung mit beiden Versionen von IBM MQ verwenden

Die Installationsaktivitäten sind abgeschlossen und Sie können jetzt die Verzeichnisse prüfen, in denen der IBM MQ-Code enthalten ist.

Vorbereitende Schritte

Stellen Sie sicher, dass das IBM MQ 8.0.0 -Produkt Installation1 in C:\Program Files\IBM\WebSphere MQ\ und das IBM MQ 9.1 -Produkt Installation2 in C:\Program Files\IBM\MQ\ordnungs-gemäß installiert wurden.

Informationen zu diesem Vorgang

Verwenden Sie den Befehl **dspmqlinst**, um die Installationsinformationen der im System installierten Versionen anzuzeigen, ohne dass ein Blick auf die Verzeichnisse erfolgen muss, und der Befehl **dspmqlver**, um die Versionsinformationen anzuzeigen.

Auf Windows -Systemen werden die Informationen in der Registry gespeichert, und die Informationen zur Installationskonfiguration werden in der folgenden Taste gespeichert: Computer\HKEY_LOCAL_MACHINE\SOFTWARE\IBM\WebSphere MQ\Installation.

Wichtig: Sie dürfen diesen Schlüssel nicht direkt bearbeiten oder referenzieren.

Vorgehensweise

1. Zeigen Sie mit dem Befehl **dspmqlinst** die Installationsinformationen an.

```
InstName:      Installation1
InstDesc:
Identifizier:  1
InstPath:      C:\Program Files\IBM\WebSphere MQ
Version:       8.0.0.9
Primary:       Yes
State:         Available
MSIProdCode:   {74F6B169-7CE6-4EFB-8A03-2AA7B2DBB57C}
MSIMedia:      8.0 Server
MSIInstanceId: 1

InstName:      Installation2
InstDesc:
Identifizier:  2
InstPath:      C:\Program Files\IBM\MQ
Version:       9.1.0.0
Primary:       No
State:         Available
MSIProdCode:   {5D3ECA81-BF8D-4E80-B36C-CBB1D69BC110}
MSIMedia:      9.1 Server
MSIInstanceId: 1
```

Anmerkung: Die Namen der betreffenden Installationen (InstName) sind wichtig.

2. Zeigen Sie die Version von IBM MQ mit **dspmqlver** für die Standardinstallation (oder primäre Installation) an:

```
C:\> dspmqver
Name:      WebSphere MQ
Version:   8.0.0.9
Level:    p800-009-180321.1
BuildType: IKAP - (Production)
Platform: WebSphere MQ for Windows (x64 platform)
Mode:     64-bit
O/S:      Windows 10 Professional x64 Edition, Build 18363
InstName: Installation1
InstDesc:
Primary:   Yes
InstPath: C:\Program Files\IBM\WebSphere MQ
DataPath: C:\ProgramData\IBM\MQ
MaxCmdLevel: 802
LicenseType: Production
```

Nach der Ausführung des Befehls erhalten Sie die folgende Nachricht: Note there are a number (1) of other installations, use the '-i' parameter to display them.

3. Geben Sie den Befehl `C:\> where dspmqver` aus. Sie erhalten daraufhin Informationen über die primäre Installation:

```
C:\> where dspmqver
C:\Program Files\IBM\WebSphere MQ\bin64\dspmqver.exe
C:\Program Files\IBM\WebSphere MQ\bin\dspmqver.exe
```

4. Um die Informationen zum IBM MQ 9.1-Produkt anzuzeigen, geben Sie den folgenden Befehl aus:
`C:\> setmqenv -n Installation2.`
5. Geben Sie den Befehl `C:\> where dspmqver` erneut aus. Sie erhalten daraufhin Informationen über die zweite Installation:

```
C:\> where dspmqver
C:\Program Files\IBM\MQ\bin64\dspmqver.exe
C:\Program Files\IBM\MQ\bin\dspmqver.exe
```

6. Geben Sie den Befehl `C:\> dspmqver` erneut aus.
Daraufhin wird Folgendes angezeigt:

```
C:\> dspmqver
Name:      IBM MQ
Version:   9.1.0.0
Level:    p910-L180705
BuildType: IKAP - (Production)
Platform: IBM MQ for Windows (x64 platform)
Mode:     64-bit
O/S:      Windows 10 Professional x64 Edition, Build 18363
InstName: Installation2
InstDesc:
Primary:   No
InstPath: C:\Program Files\IBM\MQ
DataPath: C:\ProgramData\IBM\MQ
MaxCmdLevel: 910
LicenseType: Production
```

7. Geben Sie den Befehl `C:\> set MQ` aus. Nach Verwendung von **setmqenv** werden Informationen zur zweiten Installation angezeigt.

```
C:\> set MQ
MQ_DATA_PATH=C:\ProgramData\IBM\MQ
MQ_ENV_MODE=64
MQ_FILE_PATH=C:\Program Files\IBM\MQ
MQ_INSTALLATION_NAME=Installation2
MQ_INSTALLATION_PATH=C:\Program Files\IBM\MQ
MQ_JAVA_DATA_PATH=C:\ProgramData\IBM\MQ
MQ_JAVA_INSTALL_PATH=C:\Program Files\IBM\MQ\java
MQ_JAVA_LIB_PATH=C:\Program Files\IBM\MQ\java\lib64
MQ_JRE_PATH=C:\Program Files\IBM\MQ\java\jre
```

Sie können eine Stapeldatei erstellen, die den Befehl **setmqenv** mit der angegebenen Syntax ausführt. Stellen Sie sicher, dass diese Stapeldatei in einem Verzeichnis in Ihrem Pfad (PATH) vorhanden ist, z. B. `C:\WinTools`.

Sie können z. B. die Stapeldatei `set-mq-910.bat` mit dem Inhalt erstellen:

```
REM Setup the environment to run MQ 9.1
CALL "C:\Program Files\IBM\MQ\bin\setmqenv" -n Installation2
REM Adding Samples to the path
SET PATH=%PATH%;%MQ_FILE_PATH%\Tools\c\Samples\Bin;%MQ_FILE_PATH%\Tools\c\Samples\Bin64
;%MQ_FILE_PATH%\Tools\jms\samples;%MQ_JAVA_INSTALL_PATH%\bin\ dspmqr -f 2
```

Anmerkungen:

- Beim Aufruf von **setmqenv** müssen Sie das Argument "CALL" verwenden. Ohne dieses Argument wird der Stapel bei der Verarbeitung von **setmqenv** beendet und die folgenden Anweisungen können nicht ausgeführt werden. Mit dem Argument CALL lassen Sie also die Verarbeitung anderer Anweisungen in der Stapeldatei zu.
- Wenn Sie ein Verzeichnis IBM MQ zu Ihrem Pfad hinzufügen, z. B. die Position für die C-Beispiele: `PATH= ...;C:\Program Files\IBM\MQ\tools\c\Samples\Bin; ...` Dieses Verzeichnis wird von **setmqenv** bei der nächsten Ausführung des Befehls ENTFERNT.

Wenn Sie die C-Beispieleausführen möchten, ist die letzte Zeile in der vorherigen Stapeldatei erforderlich, um das Verzeichnis für die Beispiele wieder in den Pfad (PATH) zu stellen.

Beachten Sie auch, dass `MQ_FILE_PATH` verwendet wird, um die richtige Verzeichnisstruktur für IBM MQ 9.1 zu verwenden: `SET PATH=%PATH%;%MQ_FILE_PATH%\tools\c\Samples\Bin.`

Einen Warteschlangenmanager erstellen

Hier erfahren Sie, wie Sie mit dem Befehl **crtmqm** einen Warteschlangenmanager erstellen können. Für diese Task können Sie IBM MQ Explorer verwenden.

Vorbereitende Schritte

Verwenden Sie den Befehl **dspmqr** mit den Parametern **-o installation** und **-s**, um den Installationsnamen und Status der aktuellen Warteschlangenmanager anzuzeigen.

```
C:\> dspmqr -o installation -s
QMNAME(QM80)                STATUS(Running) INSTNAME(Installation1)
INSTPATH(C:\Program Files\IBM\WebSphere MQ) INSTVER(9.1.0.9)
QMNAME(QMMIG)               STATUS(Running) INSTNAME(Installation1)
INSTPATH(C:\Program Files\IBM\WebSphere MQ) INSTVER(9.1.0.9)
```

Informationen zu diesem Vorgang

Um den folgenden Prozess ausführen zu können, müssen Sie eine Windows-Eingabeaufforderung öffnen und sich als *Administrator* festlegen. Wenn Sie versuchen, den Befehl **crtmqm** auszugeben, ohne hierfür die Eingabeaufforderung zu verwenden, erhalten Sie die Nachricht AMQ7077: Sie sind nicht zur Ausführung der angeforderten Operation berechtigt.

Vorgehensweise

- Wählen Sie **Start > Windows-System > Eingabeaufforderung > Mehr > Als Administrator ausführen** aus.

Der Titel des daraufhin erstellten Fensters lautet *Administrator: Eingabeaufforderung*.

Anmerkung: Die Version von IBM MQ in der Eingabeaufforderung lautet IBM MQ 9.1.

- Führen Sie den Befehl **setmqenv** oder die Stapeldatei **set-mq-930** aus, die Sie möglicherweise erstellt haben.

Ausführliche Informationen finden Sie in [„Befehl setmqenv zur Ausführung mit beiden Versionen von IBM MQ verwenden“](#) auf Seite 156.

In beiden Fällen wird Version 9.3.0.0 angezeigt.

- Geben Sie den Befehl `C:\> crtmqm -u SYSTEM.DEAD.LETTER.QUEUE QM930` aus.
Sie sehen die folgende Ausgabe:

```
IBM MQ queue manager created.
Directory 'C:\ProgramData\IBM\MQ\qmgrs\QM930' created.
The queue manager is associated with installation 'Installation2'.
Creating or replacing default objects for queue manager 'QM930'.
Default objects statistics : 87 created. 0 replaced. 0 failed.
Completing setup.
Setup completed.
```

4. Geben Sie den folgenden Befehl aus, um den Warteschlangenmanager C:\> `strmqm QM930` zu starten

Sie sehen die folgende Ausgabe. Beachten Sie die Zeilen, in denen die Installation und Version angegeben sind, unter denen der Warteschlangenmanager ausgeführt wird:

```
IBM MQ queue manager 'QM930' starting.
The queue manager is associated with installation 'Installation2'.
5 log records accessed on queue manager 'QM930' during the log replay phase.
Log replay for queue manager 'QM930' complete.
Transaction manager state recovered for queue manager 'QM910'.
IBM MQ queue manager 'QM930' started using V9.3.0.0.
```

5. Geben Sie den Befehl C:\> `dspmq -o installation -s` erneut aus, um die installierten Warteschlangenmanager anzuzeigen:

```
C:\> dspmq -o installation -s
QMNAME(QM80) STATUS(Running) INSTNAME(Installation1)
INSTPATH(C:\Program Files\IBM\WebSphere MQ) INSTVER(9.1.0.9)
QMNAME(QMMIG) STATUS(Running) INSTNAME(Installation1)
INSTPATH(C:\Program Files\IBM\WebSphere MQ) INSTVER(9.1.0.9)
QMNAME(QM910) STATUS(Running) INSTNAME(Installation2)
INSTPATH(C:\Program Files\IBM\MQ) INSTVER(9.3.0.0)
++ Cannot use MQ 9.3.0 administrative commands to run an MQ 9.1 queue manager
```

Wichtig: Sie können die Verwaltungsbefehle nicht auf einer anderen IBM MQ-Version als ihrer aktuellen Version verwenden.

Wenn Sie versuchen, dies zu tun, erhalten Sie die Nachricht AMQ5691E: Der Warteschlangenmanager <qmname>st einer anderen Installation zugeordnet (<installation name>)..

Warteschlangenmanager zu einer höheren Version von IBM MQ migrieren

Nach der Installation von IBM MQ 9.4 möchten Sie nun eine Migration und ein Upgrade für einen Warteschlangenmanager von IBM MQ 9.1 durchführen, der mit IBM MQ 9.4 verwendet werden soll.

Informationen zu diesem Vorgang

Hierfür sind im Wesentlichen zwei Schritte erforderlich:

1. Weisen Sie den Warteschlangenmanager mit dem Befehl **setmqm** der gewünschten Installation, d. h. der IBM MQ-Version zu.
2. Verwenden Sie den Befehl **strmqm** unter der gewünschten Version. Mit diesem Befehl werden die Daten des Warteschlangenmanagers für die neue Version aktualisiert.

In diesem Szenario wird der Warteschlangenmanager QMMIG, der mit IBM MQ 9.1 erstellt wurde, zur Veranschaulichung der Migrationsprozedur verwendet.



Achtung: Nach der Migration eines Warteschlangenmanagers zu einer höheren Version von IBM MQ ist es nicht mehr möglich, den Warteschlangenmanager mit der vorherigen Version von IBM MQ zu verwenden. Bei der Migration werden eine Reihe von Dateien und Objektdefinitionen geändert, es ist nicht möglich, zurückzukehren.

Vorgehensweise

1. Erstellen Sie mit dem Befehl **dmpmqcfg** eine Sicherung des Warteschlangenmanagers.

Weitere Informationen finden Sie in den Abschnitten IBM MQ-Warteschlangenmanagerdaten sichern und wiederherstellen und Warteschlangenmanagerkonfiguration sichern.

- a) Geben Sie den folgenden Befehl an, um alle Attribute anzugeben, einschließlich der Standardattribute (mit Ausnahme von **setmqaut**, das nicht in der Ausgabe enthalten ist):

```
dmpmqcfg -m QMgr -a > QMgr.dmpmqcfg.out.all.mqsc
```

- b) Geben Sie den folgenden Befehl aus, um die Attribute im **setmqaut**-Format zu erfassen:

```
dmpmqcfg -m QMgr -o setmqaut > QMgr.dmpmqcfg.setmqaut.bat
```

Anmerkung: In der Ausgabedatei mit den **setmqaut**-Befehlen ist bei jedem Befehl der Name des Warteschlangenmanagers enthalten. Wenn Sie die Befehle daher für einen anderen Warteschlangenmanager wiederherstellen möchten, müssen Sie die Datei daher bearbeiten und den Namen des gewünschten Warteschlangenmanagers angeben.

2. Zur Wiederherstellung:

- a) der Befehle für **runmqsc** geben Sie Folgendes aus:

```
runmqsc Qmgr < QMgr.dmpmqcfg.out.mqsc
```

oder

```
runmqsc Qmgr < QMgr.dmpmqcfg.out.all.mqsc
```

- b) der **setmqaut**-Befehle geben Sie Folgendes aus:

```
QMgr.dmpmqcfg.setmqaut.bat
```

3. Die Version des zu migrierenden Warteschlangenmanagers ist IBM MQ 9.1, daher müssen Sie das Script ausführen, mit dem IBM MQ 9.4 als aktive Umgebung festgelegt wird:

```
C:\> set-mq-80
```

- a) Geben Sie den Befehl C:\> dspmqr aus, um die Version zu überprüfen, die der Warteschlangenmanager in IBM MQ 9.1 ausführt.
- b) Geben Sie den Befehl C:\> where dspmqr aus, um zu prüfen, ob der Warteschlangenmanager ausgeführt wird:

```
C:\Program Files\IBM\WebSphere MQ\bin64\dspmqr.exe  
C:\Program Files\IBM\WebSphere MQ\bin\dspmqr.exe
```

- c) Geben Sie den Befehl C:\> dspmqr -m QMMIG -o installation -s aus.

```
QMNAME(QMMIG)           STATUS(Running) INSTNAME(Installation1)  
INSTPATH(C:\Program Files\IBM\WebSphere MQ) INSTVER(9.1.0.9)
```

- d) Geben Sie den Befehl C:\> runmqsc QMMIG aus.

```
display qmgr cmdlevel version  
1 : display qmgr cmdlevel  
AMQ8408: Display Queue Manager details.  
QMNAME(QMMIG)           CMDLEVEL(910)  
VERSION(09001009)  
end
```

und beachten Sie, dass CMDLEVEL auf IBM MQ 9.1 festgelegt ist.

- e) Stoppen Sie den Warteschlangenmanager mit dem Befehl C:\ endmqm -i QMMIG
Sie erhalten folgende Nachrichten:

```
WebSphere MQ queue manager 'QMMIG' ending.  
WebSphere MQ queue manager 'QMMIG' ended.
```


- Ändern Sie die Umgebung für die Ausführung der IBM MQ 9.4 -Befehle, indem Sie entweder den Befehl `C:\> set-mq-930` ausgeben, wenn Sie die Stapeldatei erstellt haben, oder indem Sie den Befehl **setmqenv** verwenden, und überprüfen Sie dann die Version, indem Sie den Befehl **dspmqlver** absetzen.
- Zeigen Sie den Status der Warteschlangenmanager an, indem Sie den Befehl `C:\> dspmq -o installation -s` ausgeben.

Sie erhalten die folgende Ausgabe:

```
QMNAME(QM91)                STATUS(Running)INSTNAME(Installation1)
INSTPATH(C:\Program Files\IBM\WebSphere MQ) INSTVER(9.1.0.9)

QMNAME(QMMIG)                STATUS(Ended immediately) INSTNAME(Installation1)
INSTPATH(C:\Program Files\IBM\WebSphere MQ) INSTVER(9.1.0.9)

QMNAME(QM910)                STATUS(Running) INSTNAME(Installation2)
INSTPATH(C:\Program Files\IBM\MQ) INSTVER(9.3.0.0)
```



Achtung: Der Warteschlangenmanager QMMIG ist immer noch Installation1 (IBM MQ 9.1) zugeordnet, während Installation2 IBM MQ 9.4 zugeordnet ist.

Daher müssen Sie die Zuordnung des Warteschlangenmanagers QMMIG zu Installation1 aufheben und den Warteschlangenmanager Installation2 zuordnen.

- Geben Sie den folgenden Befehl aus, um den Warteschlangenmanager QMMIG Installation2 zuzuordnen

```
C:\> setmqm -m QMMIG -n Installation2
```

Sie erhalten die folgende Nachricht:

```
The setmqm command completed successfully.
```

Dies bedeutet, dass QMMIG jetzt IBM MQ 9.4 zugeordnet ist.

- Starten Sie Warteschlangenmanager QMMIG, indem Sie den Befehl `C:\> strmqm QMMIG` ausgeben. Da der IBM MQ 9.4-Befehl **strmqm** zum ersten Mal für einen Warteschlangenmanager ausgegeben wird, der zuvor mit einer älteren Version verwendet wurde, findet eine Migration statt.

Sie sehen eine ungefähr wie folgt aussehende Ausgabe:

```
IBM MQ queue manager 'QMMIG' starting.
The queue manager is associated with installation 'Installation2'.
5 log records accessed on queue manager 'QMMIG' during the log replay phase.
Log replay for queue manager 'QMMIG' complete.
Transaction manager state recovered for queue manager 'QMMIG'.
Migrating objects for queue manager 'QMMIG'.
Default objects statistics : 5 created. 0 replaced. 0 failed.
IBM MQ queue manager 'QMMIG' started using V9.3.0.0.
```

- Geben Sie den Befehl `C:\> runmqsc QMMIG` aus, um die Attribute des Warteschlangenmanagers anzuzeigen, und beachten Sie die Felder VERSION und CMDLEVEL:

```
display qmgr cmdlevel version
1 : display qmgr cmdlevel version
AMQ8408I: Display Queue Manager details.
QMNAME(QMMIG)                CMDLEVEL(930)
VERSION(09300000)
end
```

Ergebnisse

Sie haben erfolgreich einen Warteschlangenmanager zu einer höheren Version des Produkts migriert.

Fixpack unter IBM MQ 9.4 installieren

Hier erfahren Sie, wie Sie ein Fixpack auf dem installierten IBM MQ 9.4 auf einem System installieren, auf dem sich Installationen mit mehreren Versionen von IBM MQ befinden.

Vorbereitende Schritte

Stellen Sie sicher, dass Sie den Warteschlangenmanager QMMIG zu IBM MQ 9.4 migriert haben. Weitere Informationen finden Sie unter [„Warteschlangenmanager zu einer höheren Version von IBM MQ migrieren“](#) auf Seite 159.

Informationen zu diesem Vorgang

Für dieses Szenario gibt es eine weitere installierte Version (IBM MQ 9.1) und die Warteschlangenmanager, die unter dieser anderen Version ausgeführt werden, werden nicht gestoppt, um zu demonstrieren, dass Sie diese anderen Versionen weiterhin verwenden können, wenn Sie Wartungsaktivitäten für IBM MQ 9.4.0. ausführen.

Beachten Sie, dass bei der Installation des Updates keine erweiterten Optionen ausgewählt werden.

Vorgehensweise

1. Melden Sie sich als Administrator an.

- a) Geben Sie den Befehl `C:\> set -mq -930` aus, wenn Sie die Stapeldatei erstellt oder den Befehl **setmqenv** verwendet haben, um sicherzustellen, dass Sie sich unter IBM MQ 9.4.0. befinden. In diesem Szenario steht n für 0.
- b) Zeigen Sie den Status der Warteschlangenmanager an, indem Sie den Befehl `C:\> dspmq -o installation -sausgeben`.

Sie erhalten die folgende Ausgabe:

```
QMNAME(QM80) STATUS(Running) INSTNAME(Installation1)
INSTPATH(C:\Program Files\IBM\MQ) INSTVER(9.1.0.0)

QMNAME(QMMIG) STATUS(Ended unexpectedly) INSTNAME(Installation2)
INSTPATH(C:\Program Files\IBM\MQ) INSTVER(9.3.0.0)

QMNAME(QM910) STATUS(Ended immediately) INSTNAME(Installation2)
INSTPATH(C:\Program Files\IBM\MQ) INSTVER(9.3.0.0)
```

2. Stoppen Sie die IBM MQ 9.4.0-Prozesse.

Weitere Informationen zum Stoppen der Prozesse finden Sie im Abschnitt [Wartung unter Windows anwenden und entfernen](#).

Zusammenfassung:

- Setzen Sie den Befehl **endmqm immediate** für die Warteschlangenmanager QmmIG und QM930ab.
- Stoppen Sie den IBM MQ-Service für die Installation. Klicken Sie dazu in der Taskleiste mit der rechten Maustaste auf das IBM MQ-Symbol und dann auf **IBM MQ stoppen**.

3. Suchen Sie das Fixpack. In diesem Szenario wird IBM MQ 9.4.0.5 verwendet.

Rufen Sie [Recommended Fixes for IBM MQ](#) auf, um eine Liste der neuesten Fixes zu erhalten:

- Continuous Delivery -Release und die Programmfixliste für Continuous Delivery -Releases
- Kumulative Sicherheitsaktualisierung für das Release Long Term Support und die Programmfixliste für Long Term Support -Releases

Anmerkung: Sie sollten überprüfen, ob Sie das neueste Fixpack verwenden, indem Sie das entsprechende Fixpack aufrufen und herunterladen.

a) Klicken Sie auf die entsprechende Registerkarte.

Bei diesem Szenario handelt es sich um *V9.3.0.5 LTS*

b) Laden Sie die Windows -Software von Fix Central oder Passport Advantage herunter. wenn Sie das gesamte Produkt herunterladen müssen,

In diesem Szenario stellen Sie die Dateigruppe in das folgende Verzeichnis: `C:\download\mq9305`. Der Dateiname lautet `9.3.0-IBM-MQ-Win64-FP0005.zip`.

4. Extrahieren Sie die Dateien aus der Datei `.zip` und führen Sie den Befehl **IBM-MQ-9.3.0-FP0005.exe** aus.

Das Dialogfenster *Install Anywhere* wird angezeigt, das die folgenden Informationen enthält: `InstallAnywhere is preparing to install ...`

Sie müssen warten, bis die Vorbereitungen abgeschlossen sind. Dies kann einige Minuten dauern.

5. Klicken Sie auf **OK**, um fortzufahren, wenn das Fenster *IBM MQ (Fixpack 9.3.0.5 -Dateien)* angezeigt wird.
- Wenn der Abschnitt *Einführung* angezeigt wird, klicken Sie auf **Weiter**.
 - Wenn der Abschnitt *Installationstyp* angezeigt wird, wählen Sie die am besten für Ihr Unternehmen geeignete Option aus. Dies wird wahrscheinlich **Dateien laden und Fixpack anwenden** sein. Klicken Sie anschließend auf **Weiter**.
 - Wenn der Abschnitt *Informationen* angezeigt wird, klicken Sie auf **Weiter**.
 - Wenn der Abschnitt *Zielordner* angezeigt wird, wählen Sie die Standardposition `C:\Program Files\IBM\source\MQ 9.3.0.5` aus und klicken Sie auf **Weiter**.
 - Wenn der Abschnitt *Erweiterte Optionen* angezeigt wird, klicken Sie auf **Weiter**.
 - Wenn der Abschnitt *Zusammenfassung der Installationsvorbereitung* angezeigt wird, überprüfen Sie die angezeigten Informationen und klicken Sie auf **Installieren**.
 - Warten Sie, während der Code geladen wird.
Im Abschnitt *Laden* ist ein Fortschrittsanzeiger zu sehen. Wenn der Prozess abgeschlossen ist, ist *Laden beendet* markiert und der Abschnitt *Fixpack anwenden* wird angezeigt.
 - Klicken Sie auf **Fertig**.
Da auf Ihrem System mehr als eine Installation vorhanden ist, wird ein Dialog angezeigt, in dem Sie die Installation auswählen können, für die ein Upgrade durchgeführt werden soll. In diesem Fall ist dies *Installation2 (9.3.0.0)*.
 - Klicken Sie auf **OK**.
 - Übernehmen Sie den Standardwert für den *Backup-Ordner* und klicken Sie auf **Anwenden**.
Es werden weitere Dialoge mit Fortschrittsanzeigern angezeigt und der letzte Dialogstatus *Fixpack 9.3.0.5 wurde angewendet. Klicken Sie auf 'Fertigstellen', um den Vorgang zu beenden*.
 - Klicken Sie auf **Fertigstellen**.

Ergebnisse

Sie haben erfolgreich ein Upgrade einer Version von IBM MQ for Windows neben einer vorhandenen Version des Produkts durchgeführt.

Managed File Transfer-Szenario

Eine Einführung in gängige Managed File Transfer-Topologien und ein Szenario, das die Verwendung der Managed File Transfer-Funktionalität veranschaulicht, indem es zeigt, wie das System eingerichtet und eine Testnachricht übertragen wird.

- [Gängige Topologien](#)
- [Basisserver konfigurieren](#)

Gängige MFT-Topologien

In diesem Abschnitt werden gängige Managed File Transfer-Topologien beschrieben. Die doppelseitigen Pfeile in jedem Diagramm stehen für Verbindungen zum Warteschlangenmanager.

Weitere Informationen zu Warteschlangenmanagerverbindungsoptionen finden Sie im Abschnitt [„Überlegungen zur Konnektivität“](#) auf Seite 167.

Basistopologie mit einem einzigen Warteschlangenmanager

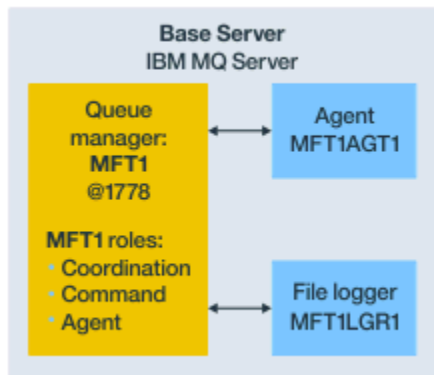


Abbildung 25. Basistopologie mit einem einzigen Warteschlangenmanager

Eine Basistopologie stellt eine vollständige Konfiguration dar, die den Koordinationswarteschlangenmanager einschließt. Der Konfigurationsname entspricht dem Namen des Koordinationswarteschlangenmanagers. Wenn der Koordinationswarteschlangenmanager MFT1 heißt, lautet der Konfigurationsname ebenfalls MFT1.

Die Basistopologie ist die erste von Ihnen erstellte Managed File Transfer-Konfiguration. Nach der Erstellung der Basiskonfiguration werden Partneragenten von fernen Servern zur Basiskonfiguration hinzugefügt, um Dateien auszutauschen.

In der Basistopologie werden außerhalb des Basistopologieservers keine Dateien ausgetauscht. Sie können jedoch in der Basistopologie Dateien an andere Positionen auf demselben Server verschieben, wo sie für Entwicklungszwecke verwendet werden könnten.

Basistopologie mit einem einzigen Partneragenten

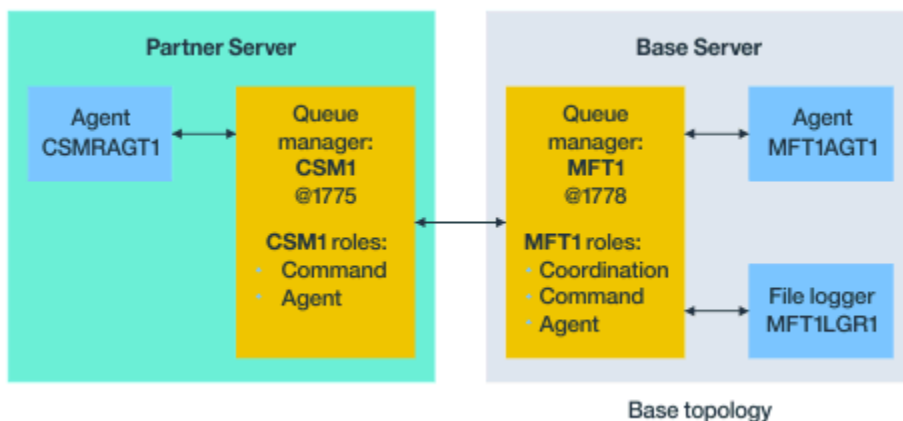


Abbildung 26. Basistopologie mit einem einzigen Partneragenten

In dieser Topologie können Dateien zwischen den zwei Agenten ausgetauscht werden. Es können weitere Partneragenten auf ähnliche Weise wie der erste hinzugefügte Agent hinzugefügt werden.

Sie können einen einzigen Warteschlangenmanager für alle drei Managed File Transfer-Warteschlangenmanagerrollen oder dedizierte Warteschlangenmanager für spezifische Rollen verwenden.

Beispielsweise könnte ein Warteschlangenmanager der Koordinationswarteschlangenmanagerrolle zugeordnet sein und die Befehls- und Agentenrollen könnten einen zweiten Warteschlangenmanager gemeinsam nutzen.

Die Verbindung zwischen einem fernen Agentenwarteschlangenmanager in einem von der Basiskonfiguration separaten Server und dem Koordinationswarteschlangenmanager der Basiskonfiguration muss als IBM MQ-Client oder MQI-Kanal konfiguriert werden.

Die Verbindung zum Koordinationswarteschlangenmanager wird mit dem Befehl **fteSetupCoordination** hergestellt. Wenn die Koordinationswarteschlangenmanagerverbindung nicht als IBM MQ-Clientkanal konfiguriert wird, schlagen Befehle wie **fteListAgents** auf dem Partnerserver fehl, wenn sie vom Partneragentenserver ausgegeben werden.

Basistopologie mit separatem Koordinationswarteschlangenmanager und einem einzigen Partneragenten

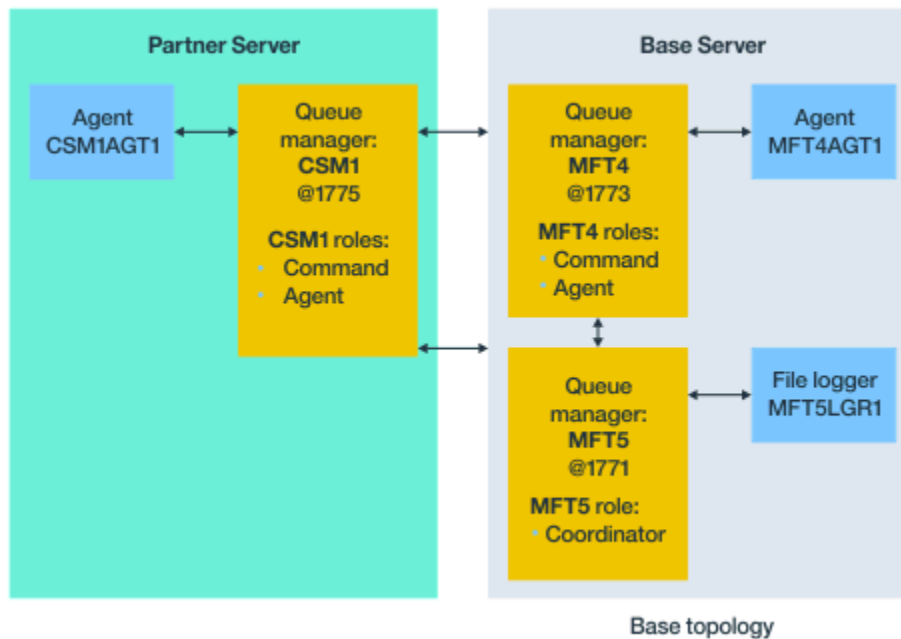


Abbildung 27. Basistopologie mit separatem Koordinationswarteschlangenmanager und einem einzigen Partneragenten

In der Basistopologie in Abbildung 3 wird auf dem Basisserver Warteschlangenmanager MFT4 für die Befehls- und Agentenrollen gemeinsam genutzt und ist Warteschlangenmanager MFT5 der Koordinationswarteschlangenmanagerrolle zugeordnet.

Es muss Konnektivität zwischen allen Warteschlangenmanagern in der Topologie bestehen, einschließlich der Warteschlangenmanager in der Basistopologie (MFT4 und MFT5).

Auf dem Partnerserverwarteschlangenmanager übernimmt Warteschlangenmanager CSM1 die Rollen als Agenten- und Befehlswarteschlangenmanager.

In dieser Topologie können Dateien zwischen den zwei Agenten ausgetauscht werden. Wie im Diagramm gezeigt, muss jeder Partneragent eine Verbindung zu einem Warteschlangenmanager herstellen. Weitere Partneragenten können auf ähnliche Weise wie der erste Partneragent hinzugefügt werden.

Basistopologie mit Managed File Transfer Agent-Partner

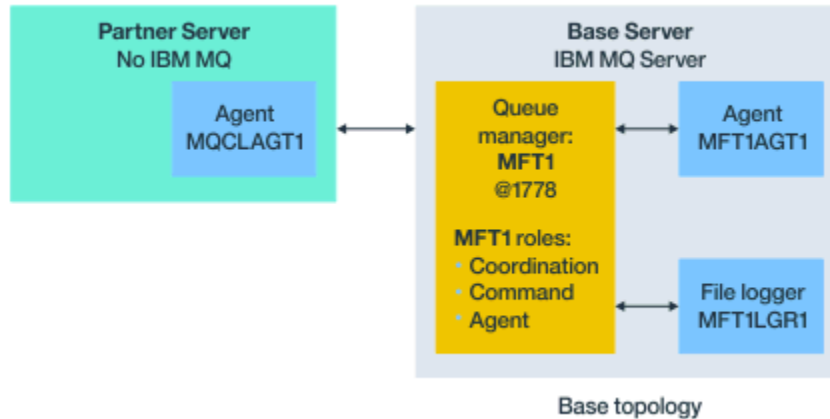


Abbildung 28. Basistopologie mit Managed File Transfer Agent-Partner

In dieser Topologie können Dateien zwischen den zwei Agenten ausgetauscht werden.

Auf dem Server im Partneragenten, im Diagramm als MQCLAGT1 dargestellt, ist der WebSphere MQ-Server nicht installiert.

Der Partneragent wird mit denselben Befehlen wie der installierte IBM MQ-Server konfiguriert, mit folgenden Ausnahmen:

- Die Konfiguration für diesen Partneragenten muss IBM MQ-Clientverbindungen zum Basiswarteschlangenmanager oder zu Warteschlangenmanagern verwenden.
- Die IBM MQ-Definitionen der Koordinationswarteschlangenmanagerrolle, die von den Konfigurationsbefehlen im Partneragentenserver erstellt wurden, müssen nicht ausgeführt werden. Die Koordinationswarteschlangenmanagerdefinitionen sind bereits im Basisserver vorhanden.

Folgendes ist jedoch erforderlich:

- Kopieren Sie die Agentenobjektdefinitionen, die beim Erstellen des Agenten generiert werden, in den Partnerserver.
- Übertragen Sie die Definitionsdatei an den Basiskonfigurationsserver.
- Erstellen Sie die Definitionen in dem Warteschlangenmanager, der als Agentenwarteschlangenmanager im Basisserver angegeben ist.

In diesem Fall übernimmt MFT1 alle drei Rollen und Sie erstellen die Objekte für Agent MQCLAGT1 im MFT1-Warteschlangenmanager.

Alternativ zum Kopieren der Objektdefinitionen in den Basisserver können Sie den Befehl **fteDefine** für Agent MQCLAGT1 auf dem Basisserver ausführen, auf dem sich der Agentenwarteschlangenmanager befindet. Erstellen Sie mithilfe der mit dem Befehl **fteDefine** generierten Definitionen die erforderlichen Agentendefinitionen auf dem Agentenwarteschlangenmanager.

Zum Beispiel würden Sie im angezeigten Diagramm die Datei MQCLAGT1_create.mqsc aus dem Agentenverzeichnis im Partnerserver auf den Basiskonfigurationsserver kopieren und die erforderlichen Agentendefinitionen im MFT1-Warteschlangenmanager erstellen.

Die Konfiguration, die Sie auf dem Partneragentenserver einrichten, erstellt das Managed File Transfer-Konfigurationsverzeichnis und erforderliche Eigenschaftendateien.

Auf dem Partnerserver können Sie den Managed File Transfer Redistributable Client von Fix Central installieren.

Anmerkung: Der MQMFT Redistributable Client unterscheidet sich vom IBM MQ Redistributable Client dadurch, dass er bereits gepackt ist und das Dienstprogramm **genmqpkg** nicht erforderlich ist. Weitere Informationen finden Sie im Abschnitt Redistributable Clients.

Überlegungen zur Konnektivität

In den oben gezeigten Diagrammen steht jede Linie zwischen den Agenten und Warteschlangenmanagern für eine Verbindung zu einem Warteschlangenmanager.

Dabei kann es sich um folgende Verbindungen handeln:

- eine lokale Verbindung
- eine Bindings- oder Nachrichtenkanalverbindung
- eine IBM MQ-Client- oder MQI-Verbindung

Welchen Verbindungstyp Sie in Ihrer Konfiguration auswählen, hängt von den Parametern ab, die Sie angeben:

- Wenn Sie den Parameter für den Warteschlangenmanagernamen ohne sonstige Verbindungsparameter angeben, geben Sie eine Bindings-Verbindung an.

Wenn der verwendete Warteschlangenmanager für die Managed File Transfer-Konfiguration ein lokaler Warteschlangenmanager ist, steht er bei Verwendung im Basiskonfigurationsserver auch für eine lokale Verbindung.

- Wenn Sie den Parameter für den Warteschlangenmanagernamen zusammen mit den entsprechenden Parametern für Host-, Port- und Kanalname angeben, geben Sie eine IBM MQ-Clientverbindung an.

Wenn sich Agenten auf demselben Host wie der Agentenwarteschlangenmanager befinden, ist die Spezifikation eines Bindings-Typs, die im Ergebnis zu einer lokalen Verbindung führt, effizienter.

Basisserver konfigurieren

Vorgehensweise zum Einrichten des Basisservers mit einem separaten Konfigurationswarteschlangenmanager.

Vorbereitende Schritte

Im folgenden Beispiel wird Folgendes vorausgesetzt:

- Sie haben den Abschnitt „Überlegungen zur Konnektivität“ auf Seite 167 gelesen und verstanden, wie der Typ der Verbindung zu Warteschlangenmanagern in der Konfiguration beeinflusst wird.
- Sie verfügen über eine funktionsfähige IBM MQ-Infrastruktur. Informationen zum Einrichten von Warteschlangenmanagern finden Sie im Abschnitt [IBM MQ-Warteschlangenmanager konfigurieren](#).
- IBM MQ-Sicherheitstasks wurden ausgeführt.

Alle Systemressourcen, z. B. der Zugriff auf Dateien, sind mit einer angemessenen Sicherheit konfiguriert.

Informationen zur Managed File Transfer-Sicherheitskonfiguration finden Sie in den Abschnitten [Managed File Transfer sichern](#) und [Benutzerberechtigungen für MFT-Agentenaktionen beschränken](#).

- Alle IBM MQ-Verbindungen werden nach der Konfiguration von IBM MQ getestet, entweder mit einem Beispielprogramm zum Senden und Empfangen von Nachrichten oder einem **amqscnxc**-Beispiel zum Testen von IBM MQ-Clientverbindungen.

Das **amqscnxc**-Beispiel stellt eine Verbindung zu einem Warteschlangenmanager her, indem es die Kanalverbindung im Beispielcode definiert. Dies ähnelt der Art und Weise, wie Managed File Transfer Verbindungen herstellt, wenn es eine MQI- oder IBM MQ-Clientverbindung verwendet.

- Bei den Anweisungen wird vorausgesetzt, dass auf dem Server, den Sie für die Basiskonfiguration verwenden, eine einzige IBM MQ-Version installiert ist. Wenn auf dem Basisserver mehrere IBM MQ-Installationen vorhanden sind, müssen Sie darauf achten, dass Sie den richtigen Pfad für die Version von IBM MQ angeben, die Sie verwenden möchten.
- Für die in diesen Anweisungen verwendeten Warteschlangenmanager ist keine Verbindungsauthentifizierung erforderlich.

Es dürfte einfacher für Sie sein, Ihre erste Konfiguration zu erstellen, ohne dass eine Verbindungsauthentifizierung erforderlich ist. Falls Ihr Unternehmen jedoch verlangt, dass sofort mit einer Verbindungsauthentifizierung gearbeitet wird, finden Sie in Abschnitt [MFT- und IBM MQ-Verbindungsauthentifizierung](#) Anweisungen zum Konfigurieren einer Berechtigungsnachweisdatei namens `MQMFTCcredentials.xml`.

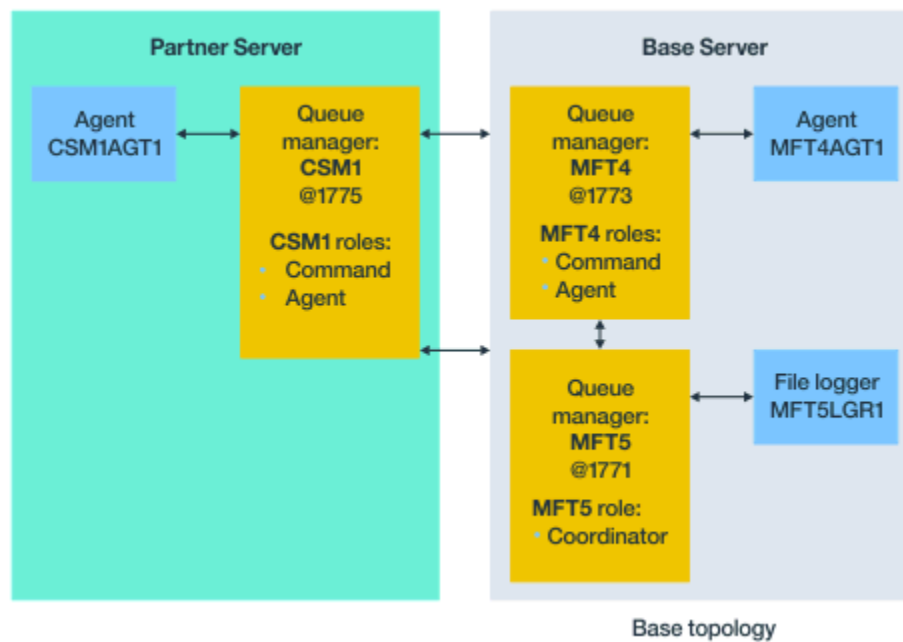


Abbildung 29. Basistopologie mit separatem Koordinationswarteschlangenmanager und einem einzigen Partneragenten

Informationen zu diesem Vorgang

Die Warteschlangenmanagerrollen für die Beispielkonfiguration sind:

- Basisserver
 - Warteschlangenmanager MFT5 ist der Koordinationswarteschlangenmanager.
 - Warteschlangenmanager MFT4 wird als Agentenwarteschlangenmanager für Agent MFT4AGT1 verwendet und dient auch als Befehlswarteschlangenmanager für die MFT5-Konfiguration auf dem Basisserver.
- Partnerserver
 - Warteschlangenmanager CSM1 ist in Doppelfunktion sowohl Agentenwarteschlangenmanager für Agent CSM1AGT1 als auch Befehlswarteschlangenmanager für die MFT5-Konfiguration auf dem Partnerserver.
 - Warteschlangenmanager MFT5 auf dem Basisserver ist der Koordinationswarteschlangenmanager.

Vorgehensweise

1. [Koordinationswarteschlangenmanager konfigurieren](#)
2. [Befehlswarteschlangenmanager konfigurieren](#)
3. [Agenten einrichten](#)
4. [Protokollfunktion einrichten](#)
5. [Partnerserver konfigurieren](#)

Nächste Schritte

Konfigurieren Sie den [MQExplorer mit MQMFT](#), sodass Sie Ihre Beispielinstallation testen können.

Koordinationswarteschlangenmanager konfigurieren

Vorgehensweise zur Konfiguration des Koordinationswarteschlangenmanager für die Koordination von Dateiübertragungen.

Vorbereitende Schritte

Es muss eine vollständige Konnektivität zwischen den für dieses Szenario konfigurierten Warteschlangenmanagern bestehen.

Informationen zu diesem Vorgang

In diesem Abschnitt wird die Konfiguration des Koordinationswarteschlangenmanagers MFT5 beschrieben, wobei die Anweisungen voraussetzen, dass Sie mit einer einzigen IBM MQ-Installation arbeiten.

Wenn Sie über mehrere Installationen verfügen, müssen Sie den Pfad IBM MQ mit dem Befehl [setmqenv](#) auf die erforderliche Version von IBM MQ setzen, bevor Sie eine der Konfigurationstasks starten.

Vorgehensweise

1. Melden Sie sich als Managed File Transfer-Administrator an.
2. Geben Sie folgenden Befehl aus, um den Koordinationswarteschlangenmanager anzugeben und die Konfigurationsverzeichnisstruktur einzurichten:

```
fteSetupCoordination -coordinationQMgr MFT5
```

Verzeichnis des Koordinationswarteschlangenmanagers

```
C:\data\mqft\config\MFT5
```

coordination.properties-Datei

```
C:\data\mqft\config\MFT5\coordination.properties
```

Der Befehl erstellt außerdem eine MQSC-Befehlsdatei, die Sie für den Koordinationswarteschlangenmanager ausführen müssen (C:\data\mqft\config\MFT5\MFT5.mqsc):

3. Wechseln Sie in das Verzeichnis C:\data\mqft\config\MFT5.
4. Konfigurieren Sie den Warteschlangenmanager als den Koordinationswarteschlangenmanager, indem Sie den folgenden Befehl ausführen.

Sie müssen die MQSC-Befehlsdatei angeben, die von dem in Schritt „2“ auf [Seite 169](#) ausgegebenen Befehl erstellt wurde:

```
runmqsc MFT5 < MFT5.mqsc > mft5.txt
```

5. Öffnen Sie die Ergebnisdatei `mft5.txt` in Ihrem bevorzugten Editor und stellen Sie sicher, dass die Definitionen erfolgreich erstellt wurden.

Nächste Schritte

Konfigurieren Sie den [Befehlswarteschlangenmanager](#).

Befehlswarteschlangenmanager konfigurieren

Vorgehensweise zur Konfiguration des Befehlswarteschlangenmanagers.

Vorbereitende Schritte

Der Koordinationswarteschlangenmanager muss bereits konfiguriert sein. Weitere Informationen finden Sie unter „[Koordinationswarteschlangenmanager konfigurieren](#)“ auf Seite 169.

Informationen zu diesem Vorgang

Im Folgenden wird die Erstellung des Befehlswarteschlangenmanagers beschrieben.

Vorgehensweise

Geben Sie den folgenden Befehl ein:

```
fteSetupCommands -connectionQMgr MFT4
```

Sie erhalten folgende BFGCL0245I-Nachricht, dass die Datei erfolgreich erstellt wurde: The file C:\data\mqft\config\MFT4\command.properties has been created successfully.

Für den Befehlswarteschlangenmanager sind keine zusätzlichen IBM MQ-Definitionen erforderlich. Nach der Ausführung von **fteSetupCommands** wird die Datei `command.properties` im Konfigurationsverzeichnis MFT5 erstellt.

Nächste Schritte

Richten Sie den [Agenten](#) ein.

Agenten einrichten

Vorgehensweise zur Vorbereitung des Dateiübertragungsagenten MFT4AGT1, einschließlich der auszuführenden MQSC-Scripts.

Vorbereitende Schritte

Der Befehlswarteschlangenmanager muss bereits konfiguriert sein. Weitere Informationen finden Sie unter „[Befehlswarteschlangenmanager konfigurieren](#)“ auf Seite 169.

Informationen zu diesem Vorgang

Im Folgenden wird die Vorbereitung des Windows-Dateiübertragungsagenten MFT4AGT1 beschrieben.

Vorgehensweise

1. Geben Sie den folgenden Befehl ein:

```
fteCreateAgent -agentName MFT4AGT1 -agentQMgr MFT4
```

Nachdem Sie den Agenten mit dem Befehl **fteCreateAgent** erstellt haben, werden das Verzeichnis `agents` und ein Unterverzeichnis für den Agenten (MFT4AGT1) zum Verzeichnis MFT5 hinzugefügt.

Im Verzeichnis `data\MFT5\agents\MFT4AGT1` finden Sie Folgendes:

- `agent.properties`-Datei
 - Die Datei `MFT4AGT1_create.mqsc` enthält IBM MQ-Definitionen, die für den Agenten erforderlich sind.
2. Wechseln Sie in das `data\MFT5\agents\MFT4AGT1`-Verzeichnis und erstellen Sie die erforderlichen Agentenwarteschlangenmanagerdefinitionen, indem Sie den folgenden Befehl ausgeben:

```
runmqsc MFT4 < MFT4AGT1_create.mqsc > mft4.txt
```

- Öffnen Sie die Ergebnisdatei `mft4.txt` mit Ihrem bevorzugten Editor und stellen Sie sicher, dass die Definitionen erfolgreich erstellt wurden.
- Starten Sie den Agenten, indem Sie folgenden Befehl eingeben: **`fteStartAgent MFT4AGT1`**.
- Geben Sie folgenden Befehl ein, um den Agenten anzuzeigen: **`fteListAgents`**.
Es sollte eine Ausgabe wie die folgende angezeigt werden:

```
5655-MFT, 5724-H72 Copyright IBM Corp. 2008, 2024. ALL RIGHTS RESERVED
BFGPR0127W: No credentials file has been specified to connect to IBM MQ.
Therefore, the assumption is that IBM MQ authentication has been disabled.
Agent Name:      Queue Manager Name:      Status:
MFT4AGT1        MFT4                            READY
```

Anmerkung: Wenn in Ihrer Managed File Transfer-Umgebung keine Verbindungsauthentifizierung aktiviert ist, können Sie die BFGPR0127W-Nachricht ignorieren.

Wenn Sie den Befehl **`ftelistAgents`** ausgeben und die folgende Nachricht empfangen, BFGCL0014W: Es sind keine Agenten vorhanden, die den aktuellen Auswahlkriterien entsprechen., Siehe [Vorgehensweise, wenn Ihr MFT-Agent vom Befehl `fteListAgents` nicht aufgelistet wird](#).

Nächste Schritte

Richten Sie die [Protokollfunktion](#) ein.

Protokollfunktion einrichten

Eine Datei- oder Datenbankprotokollfunktion ist erforderlich, um Protokolldaten und Prüfinformationen zu Übertragungsaktivitäten für die Konfiguration zu speichern. In diesem Beispiel erstellen Sie eine Datei-protokollfunktion.

Vorbereitende Schritte

Folgendes muss bereits konfiguriert sein:

- Konfigurationswarteschlangenmanager
- Befehlswarteschlangenmanager
- Agent

Vorgehensweise

- Geben Sie den folgenden Befehl ein:

```
fteCreateLogger -loggerQMgr MFT5 -loggerType FILE
-fileLoggerMode CIRCULAR -fileSize 5MB -fileCount 3 MFT5lgr1
```

Nach Ausführung des Befehls **`fteCreateLogger`** wird das Verzeichnis `data\mqft\config\MFT5\loggers` mit dem Unterverzeichnis `MFT5LGR1` erstellt.

Das Unterverzeichnis `MFT5LGR1` enthält die Datei `logger.properties`. Außerdem enthält das Verzeichnis eine Datei namens `MFT5LGR1_create.mqsc` mit IBM MQ-Definitionen, die für die Protokollfunktion erforderlich sind.

- Wechseln Sie in das Verzeichnis `data\mqft\config\MFT5\loggers\MFT5LGR1`.
- Führen Sie die zugehörige MQSC-Befehlsdatei aus.

```
runmqsc MFT5 < MFT5_create.mqsc
```

Dadurch werden die für die Protokollfunktion erforderlichen Definitionen erstellt.

- Überprüfen Sie anhand der Ergebnisse der Objektdefinitionen, ob die erforderlichen Objekte erfolgreich erstellt wurden.

4. Starten Sie die Protokollfunktion, indem Sie folgenden Befehl ausgeben: **fteStartLogger** MFT5LGR1.
5. Überprüfen Sie den Inhalt der Datei `output0.log` im Verzeichnis `data\mqft\logs\MFT5\loggers\MFT5LGR1\logs`.

Nach einigen Informationen zu der Protokollfunktion sollte der letzte Eintrag folgende Nachricht enthalten: BFGDB0023I: The logger has completed startup activities and is now running. (Die Protokollfunktion hat den Startvorgang beendet und ist jetzt aktiv.)

Es kann vorkommen, dass beim ersten Start der Protokollfunktion keine Protokolldaten in die Datei `output0.log` geschrieben werden. Wenn die Datei `output0.log` leer ist, starten Sie die Protokollfunktion erneut, indem Sie **fteStopLogger** MFT5LGR1 eingeben und die **Eingabetaste** drücken.

Starten Sie die Protokollfunktion erneut, indem Sie **fteStartLogger** MFT5LGR1 eingeben und die **Eingabetaste** drücken. Die Datei `output0.log` enthält jetzt Daten.

Das gleiche Verhalten gilt auch für die Agentenversion der Datei `output0.log` beim ersten Start eines Agenten.

Stoppen und starten Sie den Agenten mit den Befehlen **fteStopAgent** und **fteStartAgent**. Danach enthält die Agentendatei `output0.log` Daten.

Ergebnisse

Sie haben den Basisserver konfiguriert, der den Koordinationswarteschlangenmanager für diese Konfiguration einschließt.

Nächste Schritte

Führen Sie nun ähnliche Aufgaben für den Partnerserver aus, der einen fernen Agenten enthält.

Partnerserver konfigurieren

Vorgehensweise zur Konfiguration eines Partnerservers, wenn der Basisserver über einen separaten Koordinationswarteschlangenmanager verfügt

Vorbereitende Schritte

Sie müssen alle Aufgaben zum Einrichten eines Basisservers, der einen Konfigurationswarteschlangenmanager einschließt, ausgeführt haben.

Informationen zu diesem Vorgang

Für den Partnerserver gelten dieselben Voraussetzungen wie für IBM MQ und die Sicherheitskonfiguration sowie für den IBM MQ-Pfad.

Beginnen Sie damit, dass Sie das MFT5-Konfigurationsverzeichnis einrichten und mit dem Befehl **fteSetupCoordination** den Koordinationswarteschlangenmanager angeben.

Vorgehensweise

1. Erstellen Sie mit folgendem Befehl das Konfigurationsverzeichnis des Partnerservers:

```
fteSetupCoordination -coordinationQMgr MFT5
-coordinationQMgrHost 177.16.20.15 -coordinationQMgrPort 1771
-coordinationQMgrChannel MQMFT.MFT5.SVRCONN
```

Anmerkungen:

- a. Wenn sich der Koordinationswarteschlangenmanager auf einem anderen Server als dem Partnerserver befindet, muss die Verbindung zum Koordinationswarteschlangenmanager des Basisservers als Clientverbindung definiert werden.

Wird die Verbindung zum Koordinationswarteschlangenmanager nicht als IBM MQ-Clientverbindung definiert, schlagen auf dem Partnerserver alle Managed File Transfer-Befehle zum Herstellen einer Verbindung zum Koordinationswarteschlangenmanager fehl.

Ein Beispiel für einen Befehl zum Herstellen einer Verbindung zum Koordinationswarteschlangenmanager ist **fteListAgents**.

- b. Sie müssen die IBM MQ-Definitionen nicht erstellen, da die für den Koordinationswarteschlangenmanager erforderlichen Definitionen bereits bei der Konfiguration des Basisservers erstellt wurden.
2. Geben Sie mit folgendem Befehl den Befehlswarteschlangenmanager an:

```
fteSetupCommands -connectionQMgr CSM1
```

Für den Befehlswarteschlangenmanager sind keine zusätzlichen IBM MQ-Definitionen erforderlich.

3. Ermitteln Sie den Partneragentenwarteschlangenmanager und erstellen Sie den Partneragentenwarteschlangenmanager, indem Sie folgenden Befehl ausgeben:

```
fteCreateAgent -agentName CSM1AGT1 -agentQMgr CSM1
```

4. Wechseln Sie in das Verzeichnis CSM1AGT1.
5. Erstellen Sie die für den Agenten erforderlichen IBM MQ-Definitionen, indem Sie folgenden Befehl ausgeben:

```
runmqsc CSM1 < CSM1AGT1_create.mqsc > csm1.txt
```

- a) Öffnen Sie die Datei csm1.txt mit Ihrem bevorzugten Editor, um sicherzustellen, dass alle für den Agenten erforderlichen Definitionen erfolgreich erstellt wurden.
6. Starten Sie den Agenten, indem Sie folgenden Befehl ausgeben:

```
fteStartAgent CSM1AGT1
```

7. Geben Sie **fteListAgents** ein, um den Agenten anzuzeigen.

Es sollte eine Ausgabe wie die folgende angezeigt werden:

```
C:\>fteListAgents
5655-MFT, 5724-H72 Copyright IBM Corp. 2008, 2024. ALL RIGHTS RESERVED
BFGPR0127W: No credentials file has been specified to connect to IBM MQ. Therefore, the assumption is that IBM MQ authentication has been disabled.
Agent Name:      Queue Manager Name:  Status:
CSM1AGT1        CSM1                  READY
MFT4AGT1        MFT4                  READY
```

Anmerkung: Wenn in Ihrer Managed File Transfer-Umgebung keine Verbindungsauthentifizierung aktiviert ist, können Sie die BFGPR0127W-Nachricht ignorieren.

Wenn Sie den Befehl **ftelistAgents** ausgeben und die folgende Nachricht empfangen, BFGCL0014W: Es sind keine Agenten vorhanden, die den aktuellen Auswahlkriterien entsprechen., Siehe [Vorgehensweise, wenn Ihr MFT-Agent vom Befehl **fteListAgents** nicht aufgelistet wird](#).

Wenn einer der Agenten den Status UNREACHABLE hat, finden Sie weitere Informationen unter [Was tun, wenn ein Agent in einem unbekanntem Status angezeigt wird](#).

IBM MQ Explorer mit MFT einrichten

Dieser Abschnitt hilft Ihnen dabei, IBM MQ Explorer mit der Managed File Transfer-Konfiguration zu verbinden.

Vorgehensweise

1. Starten Sie IBM MQ Explorer.

2. Blättern Sie in der linken Navigator abwärts und erweitern Sie den Ordner: Managed File Transfer.
Sie sehen den Eintrag für den Koordinationswarteschlangenmanager: MFT5
3. Klicken Sie mit der rechten Maustaste auf MFT5 und wählen Sie **Verbinden** aus.
 - a) Wählen Sie im angezeigten Dropdown-Menü Agents aus und stellen Sie sicher, dass beide Agenten, MFT4AGT1 und CSMAGT1, den Status Ready aufweisen.

Nächste Schritte

Testen Sie Ihre Beispielkonfiguration mit [IBM MQ Explorer](#).

IBM MQ Explorer zum Testen einer Dateiübertragung verwenden

Dieser Abschnitt zeigt in einem Beispiel, wie IBM MQ Explorer zusammen mit Managed File Transfer zum Testen einer Dateiübertragung verwendet werden kann, nachdem IBM MQ Explorer, wie im vorherigen Abschnitt beschrieben, eingerichtet wurde.

Vorbereitende Schritte

Stellen Sie sicher, dass Sie über ein funktionsfähiges System verfügen, dass die Agenten READY sind und IBM MQ Explorer funktioniert. Weitere Informationen finden Sie unter „[IBM MQ Explorer mit MFT einrichten](#)“ auf Seite 173.

Informationen zu diesem Vorgang

Bestimmen Sie die Datei, die zum Testen der Übertragung verwendet, und ein Verzeichnis, in das die Datei kopiert werden soll. In diesem Beispiel wird vorausgesetzt, dass die Datei `test-file.txt` im Verzeichnis `C:\temp\mft` verwendet wird.

```
C:\temp\mft> dir *
Date stamp 61 test-file.txt
1 File(s) 61 bytes
```

Vorgehensweise

1. Starten Sie IBM MQ Explorer unter Windows
2. Erweitern Sie im linken Navigator den Ordner Managed File Transfer.
Sie sehen den Eintrag für den Koordinationswarteschlangenmanager: MFT5
3. Klicken Sie mit der rechten Maustaste auf MFT5 und wählen Sie **Verbinden** aus.
4. Klicken Sie nach Herstellung der Verbindung mit der rechten Maustaste auf MFT5 und wählen Sie **Neue Übertragung** aus.
 - a) Wählen Sie im Pulldown-Menü MFT4AGT1 als Quellenagenten und CSMAGT1 als Zielagenten aus.
 - b) Klicken Sie auf **Weiter**.
 - c) Klicken Sie im nächsten Fenster auf **Hinzufügen**.
Es wird ein breites Dialogfenster geöffnet. Die linke Seite ist für Source und die rechte Seite für Destination.
5. Gehen Sie in der Anzeige Source wie folgt vor:
 - a) Wählen Sie **Textübertragung** aus, da es sich um eine Textdatei handelt.
 - b) Wählen Sie **Durchsuchen** aus, um nach der Datei zu suchen.
In diesem Fall ist die Datei `C:\temp\mft\test-file.txt`.



Achtung: Klicken Sie nicht auf **OK**, da Sie die Destination -Anzeige ausfüllen müssen.

6. Gehen Sie in der Anzeige Destination wie folgt vor:

- a) Geben Sie den Namen ein, den die Datei am Ziel erhalten soll, z. B. `test-file.txt`.
Die Verwendung relativer Pfade wird unterstützt. Am Anfang des vollständigen Dateipfads steht das Ausgangsverzeichnis der Benutzer-ID, die den Zielagenten startet.
- b) Wählen Sie `Overwrite files if present` aus, wenn Sie diese Option benötigen.
- c) Klicken Sie auf **OK**.
Die ausgewählte Datei wird im Teilfenster **Neue Übertragung** angezeigt.
7. Wenn das MFT5-Konfigurationsmenü geschlossen ist und +MFT5 angezeigt wird, erweitern Sie das Menü durch Klicken auf das Pluszeichen (+).
8. Bleiben Sie bei der ausgewählten MFT-Konfiguration.
Überprüfen Sie als Nächstes den Status der Übertragung, indem Sie die folgende Prozedur ausführen.
9. Klicken Sie unter dem Koordinationswarteschlangenmanager MFT5 auf **Übertragungsprotokoll**.
10. Sehen Sie sich den Status in `Managed File Transfer - Current Transfer progress pane` direkt unter der oberen Anzeige **Übertragungsprotokoll** an und warten Sie, bis die Übertragung abgeschlossen ist.
Wird die Übertragung als erfolgreich und mit einem grünen Hintergrund angezeigt, haben Sie den Test Ihrer Konfiguration erfolgreich durchgeführt.
Wird die Übertragung als fehlgeschlagen und mit einem roten Hintergrund angezeigt, ist ein Fehler aufgetreten.
In den meisten Fällen können Sie die Bildlaufleiste unter dem Fenster **Übertragungsprotokoll** verwenden und eine Zusammenfassung der Ursachen für die fehlgeschlagene Übertragung anzeigen.
 - a) Falls Sie nicht feststellen können, warum die Übertragung fehlgeschlagen ist, doppelklicken Sie im Fenster **Übertragungsprotokoll** auf den Eintrag für die Übertragung.
 - b) Wählen Sie XML auf der linken Seite der daraufhin geöffneten Popup-Anzeige aus.
 - c) Blättern Sie durch die Informationen, um die Fehlerursache festzustellen.
 - d) Führen Sie die erforderlichen Korrekturen durch und wiederholen Sie den Test der Übertragung.

Erste Schritte mit IBM MQ Internet Pass-Thru

Diese Szenarios zeigen, wie Sie einige einfache Konfigurationen von IBM MQ Internet Pass-Thru (MQIPT) einrichten. Mit diesem Tasks können Sie auch bestätigen, dass das Produkt erfolgreich installiert wurde.

Vorbereitende Schritte

Bevor Sie mit der Verwendung dieser Szenarios beginnen, müssen Sie sicherstellen, dass die folgenden Voraussetzungen erfüllt sind:

- Sie sind mit dem Definieren von Warteschlangenmanagern, Warteschlangen und Kanälen in IBM MQ vertraut.
- Sie haben bereits einen IBM MQ-Client und -Server installiert.
- MQIPT wird in einem Verzeichnis mit dem Namen `C:\mqipt` auf Windows -Systemen installiert. (Die Beispiele sind für Windows-Systeme geschrieben, werden aber auf jeder unterstützten Plattform ausgeführt.) Weitere Informationen zur Installation von MQIPT finden Sie im Abschnitt [MQIPT installieren](#).
- Der Client, der Server und jede Instanz von MQIPT sind auf separaten Computern installiert.
- Sie sind mit dem Einreihen von Nachrichten in einer Warteschlange mithilfe des Befehls **amqspu** vertraut.
- Sie sind mit dem Abrufen von Nachrichten aus einer Warteschlange mithilfe des Befehls **amqsget** vertraut.
- Sie sind mit dem Festlegen von Clientberechtigungen in IBM MQ vertraut.

Informationen zu diesem Vorgang

Nachdem Sie die Voraussetzungen erfüllt haben, führen Sie die folgenden ersten Schritte aus, um das System für die Arbeit mit den Szenarios zu konfigurieren.

Vorgehensweise

1. Führen Sie auf dem IBM MQ-Server die folgenden Tasks aus:

- Definieren Sie einen Warteschlangenmanager mit der Bezeichnung `MQIPT.QM1`.
- Definieren Sie einen Serververbindungskanal mit der Bezeichnung `MQIPT.CONN.CHANNEL`.
- Definieren Sie eine lokale Warteschlange mit der Bezeichnung `MQIPT.LOCAL.QUEUE`.
- Starten Sie einen TCP/IP-Listener für `MQIPT.QM1` auf Port 1414. Wenn Port 1414 bereits von einer anderen Anwendung verwendet wird, wählen Sie eine freie Portadresse aus und ersetzen Sie den Port in den folgenden Beispielen entsprechend.
- Stellen Sie sicher, dass die Verbindungsauthentifizierung und die Kanalaauthentifizierung so konfiguriert ist, dass Clientverbindungen von der Clientmaschine mit Ihrer Benutzer-ID zulässig sind. Wenn die Verbindungsauthentifizierung so eingestellt ist, dass Anwendungen Authentifizierungsnachweise für Clientverbindungen bereitstellen müssen, müssen Sie eine der folgenden Umgebungsvariablen festlegen, bevor Sie die Befehle `amqsputc` und `amqsgetc` ausführen:

MQSAMP_USER_ID

Wird auf die für die Verbindungsauthentifizierung zu verwendende Benutzer-ID gesetzt, wenn Sie eine Benutzer-ID und ein Kennwort für die Authentifizierung beim Warteschlangenmanager verwenden wollen.



Legen Sie einen Wert fest, der nicht leer ist, wenn für die Authentifizierung beim Warteschlangenmanager ein Authentifizierungstoken verwendet werden soll.

2. Testen Sie die Route vom IBM MQ -Client zum Warteschlangenmanager, indem Sie mit dem Befehl `amqsputc` eine Nachricht in die lokale Warteschlange des Warteschlangenmanagers einreihen und anschließend mit dem Befehl `amqsgetc` abrufen.

Um die Szenarios in diesem Abschnitt vorzubereiten, erstellen und bearbeiten Sie die `mqiplt.conf`-Datei wie folgt:

- a. Kopieren Sie die Datei `mqipltSample.conf`, die sich im Unterverzeichnis `samples` des MQIPT-Installationsverzeichnis befindet, in das Verzeichnis `mqiplt.conf` im ausgewählten MQIPT-Ausgangsverzeichnis. In den folgenden Szenarios wird `C:\mqipltHome` als Ausgangsverzeichnis von MQIPT verwendet.
- b. Erstellen Sie zwei Verzeichnisse neben `mqiplt.conf` mit dem Namen `errors` und `logs`. Legen Sie die Dateiberechtigungen in diesen Verzeichnissen fest, damit die Benutzer-ID, mit der MQIPT ausgeführt wird, über Schreibzugriff verfügt.
- c. Löschen Sie alle Routen aus der Datei `mqiplt.conf`.
- d. Überprüfen Sie im übrigen Abschnitt `[global]`, ob `ClientAccess` vorhanden ist und auf `true` gesetzt ist.

Nächste Schritte

Nach der Einrichtung Ihres Systems können Sie die folgenden Szenarios starten:

- [„Ordnungsgemäße Funktion von MQIPT überprüfen“ auf Seite 177](#)
- [„Fordern Sie ein von einer Zertifizierungsstelle signiertes Zertifikat für MQIPT an.“ auf Seite 180](#)
- [„Testzertifikate erstellen“ auf Seite 178](#)
- [„TLS-Server authentifizieren“ auf Seite 182](#)
- [„TLS-Client authentifizieren“ auf Seite 184](#)

- „[HTTP-Tunnelung konfigurieren](#)“ auf Seite 191
- „[Zugriffssteuerung konfigurieren](#)“ auf Seite 192
- „[SOCKS-Proxy konfigurieren](#)“ auf Seite 195
- „[SOCKS-Client konfigurieren](#)“ auf Seite 197
- „[Clustering-Unterstützung für MQIPT konfigurieren](#)“ auf Seite 199
- „[Portnummern zuordnen](#)“ auf Seite 202
- „[CRLs mithilfe eines LDAP-Servers abrufen](#)“ auf Seite 203
- „[MQIPT im TLS-Proxy-Modus ausführen](#)“ auf Seite 206
- „[MQIPT im TLS-Proxy-Modus mit Sicherheitsmanager ausführen](#)“ auf Seite 208
- „[Sicherheitsexit verwenden](#)“ auf Seite 211
- „[Clientverbindungsanforderungen mithilfe von Sicherheitsexits an IBM MQ-Server für Warteschlangenmanager weiterleiten](#)“ auf Seite 213
- „[Clientverbindungsanforderungen dynamisch weiterleiten](#)“ auf Seite 216
- „[Zertifikatsexit für die Authentifizierung eines TLS-Servers verwenden](#)“ auf Seite 219

Ordnungsgemäße Funktion von MQIPT überprüfen

Verwenden Sie dieses einfache Konfigurationssetup, um sicherzustellen, dass MQIPT ordnungsgemäß installiert ist.

Vorbereitende Schritte

- Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „[Erste Schritte mit IBM MQ Internet Pass-Thru](#)“ auf Seite 175 aufgelisteten vorausgesetzten Tasks abgeschlossen haben.

Informationen zu diesem Vorgang

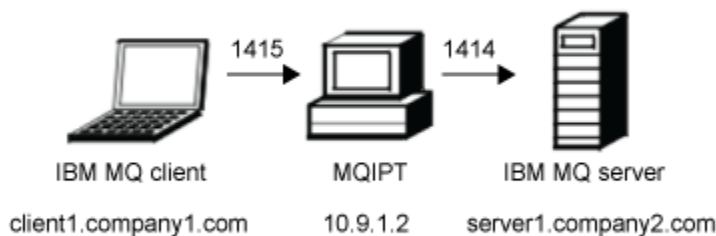


Abbildung 30. Netzdiagramm für Installationsprüftests

Dieses Diagramm zeigt die Verbindung vom IBM MQ-Client (client1.company1.com an Port 1415) über MQIPT zum IBM MQ-Server (server1.company2.com an Port 1414).

Vorgehensweise

Führen Sie die folgenden Schritte aus, um zu prüfen, ob MQIPT ordnungsgemäß funktioniert:

1. Definieren Sie eine MQIPT-Route.

Bearbeiten Sie auf dem MQIPT -Computer `mqipt.conf` und fügen Sie eine Routendefinition hinzu:

```
[route]
ListenerPort=1415
```

```
Destination=server1.company2.com
DestinationPort=1414
```

2. Starten Sie MQIPT.

Öffnen Sie eine Eingabeaufforderung und geben Sie folgenden Befehl ein:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt C:\mqiptHome die Position der MQIPT-Konfigurationsdatei mqipt.conf und ipt1 den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt1
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1415 has started and will forward messages to:
MQCPI034 ...server1.company2.com(1414)
MQCPI035 ...using MQ protocol
MQCPI078 Route 1415 ready for connection requests
```

3. Geben Sie in einer Eingabeaufforderung auf dem IBM MQ-Clientsystem die folgenden Befehle ein:

a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=MQIPT.CONN.CHANNEL/tcp/10.9.1.2(1415)
```

b) Reihnen Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE MQIPT.QM1
Hello world
```

Nachdem Sie die Nachrichtenzeichenfolge eingegeben haben, drücken Sie zweimal die Eingabetaste.

c) Rufen Sie die Nachricht ab:

```
amqsgetc MQIPT.LOCAL.QUEUE MQIPT.QM1
```

Die Nachricht "Hello world" wird zurückgegeben.

4. Geben Sie den folgenden Befehl ein, um IBM MQ zu stoppen:

```
mqiptAdmin -stop -n ipt1
```

Testzertifikate erstellen

Ein selbst signiertes Zertifikat kann von einer MQIPT -Route verwendet werden, um sich bei einem fernen Peer zu identifizieren.

Selbst signierte Zertifikate können in Testszenarios nützlich sein, damit Sie die TLS-Konnektivität sicherstellen können, ohne eine Zertifizierungsstelle (Certificate Authority, CA) für ein Zertifikat bezahlen zu müssen. Verwenden Sie jedoch keine selbst signierten Zertifikate in Produktionsumgebungen. Wenn Sie ein Zertifikat für die Verwendung in einer Produktionsumgebung anfordern müssen, lesen Sie den Abschnitt [„Fordern Sie ein von einer Zertifizierungsstelle signiertes Zertifikat für MQIPT an.“](#) auf Seite 180.

Vorbereitende Schritte

Bevor Sie mit dieser Task beginnen, führen Sie die in [„Erste Schritte mit IBM MQ Internet Pass-Thru“](#) auf Seite 175 aufgelisteten Schritte aus.

Informationen zu diesem Vorgang

V9.4.0 **V9.4.0** Erstellen Sie das Zertifikat mit dem Befehl **mqiptKeytool**. Geben Sie den Namen des Keystores an, der das Zertifikat in der Routeneigenschaft **SSLServerKeyRing** oder **SSLClientKeyRing** MQIPT enthält, je nachdem, ob das Zertifikat von eingehenden oder abgehenden Verbindungen verwendet wird.

Vorgehensweise

Geben Sie den folgenden Befehl ein, um ein selbst signiertes persönliches Zertifikat zu Testzwecken in einem PKCS #12 -Keystore zu erstellen:

```
mqiptKeytool -genkeypair -keystore keystore_name -storetype pkcs12 -storepass password  
-alias label -dname DN_identity  
-keyalg key_algorithm -keysize key_size -sigalg signature_algorithm
```

Dabei gilt:

-keystore *keystore-name*

Gibt den Namen des Keystores an Beispiel: mqiptKeys.p12. Der Keystore wird erstellt, wenn er nicht vorhanden ist.

-storepass *Kennwort*

Gibt das Keystore-Kennwort an

-alias *Bezeichnung*

Gibt die Zertifikatsbezeichnung an.

-dname *DN-Identität*

Gibt den definierten X.500 -Namen für das Zertifikat in Anführungszeichen an. Beispiel: "CN=Test Certificate, OU=Sales, O = Example, C = US"

-keyalg *Schlüsselalgorithmus*

Gibt den Algorithmus an, der zum Erstellen des Schlüsselpaars verwendet wird Beispiel: RSA.

-keysize *Schlüsselgröße*

Gibt die Schlüsselgröße an. Beispiel: 2048.

-sigalg *Signaturalgorithmus*

Gibt den Algorithmus an, der zum Signieren des Zertifikats verwendet wird. Beispiel: SHA256WithRSA.

Wenn Sie die Beispielwerte verwenden, erstellt dieser Befehl ein digitales Zertifikat mit einem öffentlichen 2048-Bit-RSA-Schlüssel und einer digitalen Signatur, die RSA mit dem Hashalgorithmus SHA-256 verwendet.

Wählen Sie einen geeigneten Verschlüsselungsalgorithmus für öffentliche Schlüssel, eine Schlüsselgröße und einen Algorithmus für digitale Signatur für die Sicherheitsanforderungen Ihres Unternehmens aus. Weitere Informationen finden Sie unter [Hinweise zu digitalen Zertifikaten für MQIPT](#).

Nächste Schritte

Setzen Sie den folgenden Befehl ab, um das Keystore-Kennwort zu verschlüsseln:

```
mqiptPW
```

Geben Sie das zu verschlüsselnde Keystore-Kennwort ein, wenn Sie dazu aufgefordert werden. Setzen Sie den Wert der entsprechenden Eigenschaft in der Konfigurationsdatei **mqipt.conf** auf das verschlüsselte Kennwort, das vom **mqiptPW** -Befehl ausgegeben wird. Legen Sie den Wert der Eigenschaft **SSLServerKeyRingPW** oder **SSLClientKeyRingPW** oder das verschlüsselte Kennwort fest, je nachdem, ob das Zertifikat von eingehenden oder abgehenden Verbindungen verwendet wird. Weitere Informationen zur Verschlüsselung von Schlüssel Speicherkennwörtern finden Sie unter [Gespeicherte Kennwörter verschlüsseln](#).

Fordern Sie ein von einer Zertifizierungsstelle signiertes Zertifikat für MQIPT an.

Fordern Sie ein Zertifikat an, das von einer anerkannten Zertifizierungsstelle (CA) signiert wurde, damit MQIPT TLS verwenden kann.

Vorbereitende Schritte

Bevor Sie mit dieser Task beginnen, führen Sie die in „[Erste Schritte mit IBM MQ Internet Pass-Thru](#)“ auf Seite 175 aufgelisteten Schritte aus.

V 9.4.0 > V 9.4.0 Diese Task setzt voraus, dass Sie mit dem Befehl **mqiptKeytool** ein neues Zertifikat von einer Zertifizierungsstelle anfordern und dass Ihr persönliches Zertifikat in einer Datei an Sie zurückgegeben wird. Ein Zertifikat reicht aus, um die Serverauthentifizierung für TLS-Verbindungen zu MQIPT -Routen zu aktivieren. Wenn Sie die Clientauthentifizierung für TLS-Verbindungen benötigen, die von MQIPT zum Routenziel zu verwenden, fordern Sie ein zweites Zertifikat an, indem Sie die Schritte in dieser Task zweimal ausführen, um zwei Keystores zu erstellen.

Informationen zu diesem Vorgang

V 9.4.0 > V 9.4.0 Verwenden Sie den Befehl **mqiptKeytool**, um die Zertifikatsanforderung zu erstellen und das signierte Zertifikat im Keystore zu empfangen. Geben Sie den Namen des Keystores an, der das Zertifikat in der Routeneigenschaft **SSLServerKeyRing** oder **SSLClientKeyRing** MQIPT enthält, je nachdem, ob das Zertifikat von eingehenden oder abgehenden Verbindungen verwendet wird.

Vorgehensweise

1. Geben Sie den folgenden Befehl ein, um ein Schlüsselpaar und ein zugehöriges Zertifikat in einem PKCS #12 -Keystore zu erstellen:

```
mqiptKeytool -genkeypair -keystore keystore_name -storetype pkcs12 -storepass password  
-alias label -dname DN_identity  
-keyalg key_algorithm -keysize key_size -sigalg signature_algorithm
```

Dabei gilt:

-keystore *keystore-name*

Gibt den Namen des Keystores an Beispiel: mqiptKeys.p12. Der Keystore wird erstellt, wenn er nicht vorhanden ist.

-storepass *Kennwort*

Gibt das Keystore-Kennwort an

-alias *Bezeichnung*

Gibt die Zertifikatsbezeichnung an.

-dname *DN-Identität*

Gibt den definierten X.500 -Namen für das Zertifikat in Anführungszeichen an. Beispiel: "CN=Test Certificate, OU=Sales, O = Example, C = US"

-keyalg *Schlüsselalgorithmus*

Gibt den Algorithmus an, der zum Erstellen des Schlüsselpaars verwendet wird Beispiel: RSA.

-keysize *Schlüsselgröße*

Gibt die Schlüsselgröße an. Beispiel: 2048.

-sigalg *Signaturalgorithmus*

Gibt den Algorithmus an, der zum Signieren des Zertifikats verwendet wird. Beispiel: SHA256WithRSA.

Wenn Sie die Beispielwerte verwenden, erstellt dieser Befehl ein digitales Zertifikat mit einem öffentlichen 2048-Bit-RSA-Schlüssel und einer digitalen Signatur, die RSA mit dem Hashalgorithmus SHA-256 verwendet.

Wählen Sie einen geeigneten Verschlüsselungsalgorithmus für öffentliche Schlüssel, eine Schlüsselgröße und einen Algorithmus für digitale Signatur für die Sicherheitsanforderungen Ihres Unternehmens aus. Weitere Informationen finden Sie unter [Hinweise zu digitalen Zertifikaten für MQIPT](#).

2. Geben Sie den folgenden Befehl ein, um eine Zertifikatssignieranforderung zu erstellen:

```
mqiptkeytool -certreq -keystore keystore_name -storetype pkcs12 -storepass password
             -alias label -file certreq_filename
```

Dabei gilt:

-keystore *keystore-name*

Gibt den Namen des Keystores an Beispiel: mqiptKeys.p12. Der Keystore wird erstellt, wenn er nicht vorhanden ist.

-storepass *Kennwort*

Gibt das Keystore-Kennwort an

-alias *Bezeichnung*

Gibt die Zertifikatsbezeichnung an.

-file *Dateiname_Zertanforderung*

Gibt den Dateinamen für die Zertifikatsanforderung an.

Senden Sie die Zertifikatsanforderungsdatei, die der Befehl erstellt, zum Signieren an Ihre Zertifizierungsstelle.

3. Wenn Sie das signierte Zertifikat von der Zertifizierungsstelle empfangen, geben Sie den folgenden Befehl ein, um das Zertifikat zum Schlüsselspeicher hinzuzufügen:

```
mqiptkeytool -importcert -keystore keystore_name -storetype pkcs12 -storepass password
             -file cert_filename
```

Dabei gilt:

-keystore *keystore-name*

Gibt den Namen des Keystores an Beispiel: mqiptKeys.p12. Der Keystore wird erstellt, wenn er nicht vorhanden ist.

-storepass *Kennwort*

Gibt das Keystore-Kennwort an

-file *Zertifikatsdateiname*

Gibt den Namen der Datei an, die das signierte Zertifikat enthält.

4. Stellen Sie sicher, dass das CA-Zertifikat der Zertifizierungsstelle, die das persönliche Zertifikat signiert hat, im CA-Keystore vorhanden ist. Sie können auswählen, ob das CA-Zertifikat demselben Keystore wie das persönliche Zertifikat oder einem separaten Keystore hinzugefügt werden soll, der nur für CA-Zertifikate verwendet wird.

Wenn Sie einen separaten CA-Keystore verwenden möchten, können Sie entweder den mit MQIPTbereitgestellten CA-Beispielkeystore `sslCAdefault.pfx` verwenden oder einen neuen PKCS #12 -Keystore erstellen. Fügen Sie das öffentliche CA-Zertifikat der Zertifizierungsstelle, die Ihr persönliches Zertifikat signiert hat, zum CA-Keystore hinzu, sofern es nicht bereits im Beispiel-Keystore vorhanden ist.

Das öffentliche CA-Zertifikat kann mit Ihrem persönlichen Zertifikat zurückgegeben werden. Wenn das öffentliche CA-Zertifikat nicht mit Ihrem Zertifikat zurückgegeben wird, müssen Sie das CA-Zertifikat von derselben Zertifizierungsstelle anfordern, die Ihr persönliches Zertifikat bereitgestellt hat, und es anschließend zum Schlüsselspeicher hinzufügen.

Geben Sie den folgenden Befehl ein, um das CA-Zertifikat zum CA-Keystore hinzuzufügen:

```
mqiptkeytool -importcert -keystore keystore_name -storetype pkcs12 -storepass password
             -file cert_filename -alias label
```

Dabei gilt:

-keystore *keystore-name*

Gibt den Namen des CA-Keystores an Der Keystore wird erstellt, wenn er nicht vorhanden ist.

-storepass Kennwort

Gibt das Kennwort für den CA-Keystore an

-file Zertifikatsdateiname

Gibt den Namen der Datei an, die das CA-Zertifikat enthält.

-alias Bezeichnung

Gibt die Bezeichnung an, die dem CA-Zertifikat zugeordnet werden soll.

Nächste Schritte

Setzen Sie den folgenden Befehl ab, um das Keystore-Kennwort zu verschlüsseln:

```
mqiptPW
```

Geben Sie das zu verschlüsselnde Keystore-Kennwort ein, wenn Sie dazu aufgefordert werden. Setzen Sie den Wert der entsprechenden Eigenschaft in der Konfigurationsdatei **mqipt.conf** auf das verschlüsselte Kennwort, das vom **mqiptPW**-Befehl ausgegeben wird. Legen Sie den Wert der Eigenschaft **SSLServerKeyRingPW** oder **SSLClientKeyRingPW** oder das verschlüsselte Kennwort fest, je nachdem, ob das Zertifikat von eingehenden oder abgehenden Verbindungen verwendet wird. Weitere Informationen zur Verschlüsselung von Schlüsselspeicherkennwörtern finden Sie unter [Gespeicherte Kennwörter verschlüsseln](#).

Um diese neuen Keystores für die Serverauthentifizierung zu verwenden, stellen Sie die Keystores in ein Verzeichnis mit dem Namen **ssl** im Ausgangsverzeichnis **MQIPT** und legen Sie die folgenden Routeneigenschaften fest:

```
SSLClientCAKeyRing=C:\\mqiptHome\\ssl\\sslCAdefault.pfx  
SSLClientCAKeyRingPW=encrypted_password  
SSLServerKeyRing=C:\\mqiptHome\\ssl\\myServer.pfx  
SSLServerKeyRingPW=encrypted_password  
SSLServerCAKeyRing=C:\\mqiptHome\\ssl\\sslCAdefault.pfx  
SSLServerCAKeyRingPW=encrypted_password
```

Weitere Informationen zum Konfigurieren von **MQIPT** für die Verwendung von **TLS** finden Sie unter [„TLS-Server authentifizieren“](#) auf Seite 182.

TLS-Server authentifizieren

In diesem Szenario können Sie eine **TLS**-Verbindung testen, indem Sie das Testzertifikat in der Schlüsselringdatei (**sslSample.pfx**) verwenden, die mit **MQIPT** im Unterverzeichnis **samples/ssl** des **MQIPT**-Installationsverzeichnisses bereitgestellt wird.

Vorbereitende Schritte

Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in [„Erste Schritte mit IBM MQ Internet Pass-Thru“](#) auf Seite 175 aufgelisteten vorausgesetzten Tasks ausgeführt haben und den Abschnitt [SSL/TLS-Unterstützung in MQIPT](#) gelesen haben.

Informationen zu diesem Vorgang

Die Verbindung wird zwischen einem **IBM MQ-Client** und einem **IBM MQ-Server** über zwei Instanzen von **MQIPT** hergestellt. Die Verbindung zwischen **MQIPT 1** und **MQIPT 2** verwendet **TLS**, wobei **MQIPT 1** als **TLS-Client** und **MQIPT 2** als **TLS-Server** fungiert.

Während des **TLS-Handshakes** sendet der Server sein Testzertifikat an den Client und der Client verwendet seine Kopie des Zertifikats mit dem **Trust-as-Peer-Flag-Set** zur Authentifizierung des Servers. Es wird die Cipher-Suite **SSL_RSA_WITH_AES_256_CBC_SHA256** verwendet. Die **mqipt.conf**-Konfigurationsdatei in diesem Szenario basiert auf der Konfigurationsdatei, die im [„Ordnungsgemäße Funktion von MQIPT überprüfen“](#) auf Seite 177 -Szenario erstellt wurde. Ausführliche Informationen zum Erstellen eines Testzertifikats zur Verwendung in diesem Beispiel finden Sie im Abschnitt [„Testzertifikate erstellen“](#) auf Seite 178.

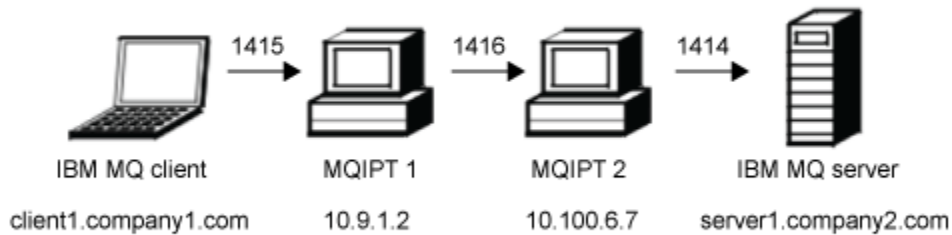


Abbildung 31. SSL/TLS-Server-Netzdiagramm

Dieses Diagramm zeigt die Verbindung vom IBM MQ-Client (client1.company1.com an Port 1415) über zwei Instanzen von MQIPT zum IBM MQ-Server (server1.company2.com an Port 1414).

Vorgehensweise

Führen Sie zur Authentifizierung eines TLS-Servers die folgenden Schritte aus:

1. Auf dem System MQIPT 1:

- a) Bearbeiten Sie `mqipt.conf` und fügen Sie die folgende Routendefinition hinzu:

```
[route]
ListenerPort=1415
Destination=10.100.6.7
DestinationPort=1416
SSLClient=true
SSLClientKeyRing=C:\mqipt\samples\ssl\sslSample.pfx
SSLClientKeyRingPW=<mqiptPW>!PCaB1HWrFM0p43ngjwgArg==!6N/vsbqr7iqMhFN+wozxQ==
SSLClientCipherSuites=SSL_RSA_WITH_AES_256_CBC_SHA256
```

- b) Öffnen Sie eine Eingabeaufforderung und starten Sie MQIPT:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt `C:\mqiptHome` die Position der MQIPT-Konfigurationsdatei `mqipt.conf` und `ipt1` den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt1
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1415 is starting and will forward messages to :
MQCPI034 ...10.100.6.7(1416)
MQCPI035 ...using MQ protocol
MQCPI036 ...SSL Client side enabled with properties :
MQCPI139 .....secure socket protocols <NULL>
MQCPI031 .....cipher suites SSL_RSA_WITH_AES_256_CBC_SHA256
MQCPI032 .....key ring file C:\mqipt\samples\ssl\sslSample.pfx
MQCPI047 .....CA key ring file <NULL>
MQCPI071 .....site certificate uses
UID=*,CN=*,T=*,OU=*,DC=*,O=*,STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*
MQCPI038 .....peer certificate uses
UID=*,CN=*,T=*,OU=*,DC=*,O=*,STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*
MQCPI078 Route 1415 ready for connection requests
```

2. Auf dem System MQIPT 2:

- a) Bearbeiten Sie `mqipt.conf` und fügen Sie die folgende Routendefinition hinzu:

```
[route]
ListenerPort=1416
```

```
Destination=Server1.company2.com
DestinationPort=1414
SSLServer=true
SSLServerKeyRing=C:\mqipt\samples\ssl\sslSample.pfx
SSLServerKeyRingPW=<mqiptPW>1!PCaB1HWiFMOp43ngjwgArg==!6N/vsbqru7iqMhFN+wozQ==
SSLServerCipherSuites=SSL_RSA_WITH_AES_256_CBC_SHA256
```

b) Öffnen Sie eine Eingabeaufforderung und starten Sie MQIPT:

```
C:
cd \mqipt\bin
mqipt .. -n ipt2
```

Dabei zeigt .. an, dass sich die MQIPT-Konfigurationsdatei mqipt.conf im übergeordneten Verzeichnis befindet und ipt2 der Name ist, der der Instanz von MQIPT zugeordnet wurde.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt2
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqipt\logs will be used to store the log files
MQCPI006 Route 1416 is starting and will forward messages to :
MQCPI034 ...Server1.company2.com(1414)
MQCPI035 ...using MQ protocol
MQCPI037 ...SSL Server side enabled with properties :
MQCPI139 .....secure socket protocols <NULL>
MQCPI031 .....cipher suites SSL_RSA_WITH_AES_256_CBC_SHA256
MQCPI032 .....key ring file C:\mqipt\samples\ssl\sslSample.pfx
MQCPI047 .....CA key ring file <NULL>
MQCPI071 .....site certificate uses
UID=*,CN=*,T=*,OU=*,DC=*,O=*,STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*
MQCPI038 .....peer certificate uses
UID=*,CN=*,T=*,OU=*,DC=*,O=*,STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*
MQCPI033 .....client authentication set to false
MQCPI078 Route 1416 ready for connection requests
```

3. Geben Sie bei einer Eingabeaufforderung auf dem IBM MQ-Client die folgenden Befehle ein:

a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=MQIPT.CONN.CHANNEL/tcp/10.9.1.2(1415)
```

b) Reichen Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE MQIPT.QM1
Hello world
```

Drücken Sie zweimal die Eingabetaste, nachdem Sie die Nachrichtenzeichenfolge eingegeben haben.

c) Rufen Sie die Nachricht ab:

```
amqsgetc MQIPT.LOCAL.QUEUE MQIPT.QM1
```

Die Nachricht "Hello world" wird zurückgegeben.

TLS-Client authentifizieren

In diesem Szenario können Sie eine TLS-Verbindung testen, indem Sie mit dem Beispieldtestzertifikat eine Server- und Clientauthentifizierung durchführen.

Vorbereitende Schritte

Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „[Erste Schritte mit IBM MQ Internet Pass-Thru](#)“ auf Seite 175 aufgelisteten vorausgesetzten Tasks ausgeführt haben und den Abschnitt [SSL/TLS-Unterstützung in MQIPT](#) gelesen haben.

Informationen zu diesem Vorgang

Die Verbindung wird zwischen einem IBM MQ-Client und einem IBM MQ-Server über zwei Instanzen von MQIPT hergestellt. Die Verbindung zwischen MQIPT 1 und MQIPT 2 verwendet TLS, wobei MQIPT 1 als TLS-Client und MQIPT 2 als TLS-Server fungiert.

Während des TLS-Handshakes sendet der Server sein Testzertifikat an den Client. Der Client verwendet seine Kopie des Zertifikats mit dem Trust-as-Peer-Flag zur Authentifizierung des Servers. Anschließend sendet der Client sein Testzertifikat an den Server. Der Server verwendet seine Kopie des Zertifikats mit dem Trust-as-Peer-Flag zur Authentifizierung des Clients. Es wird die Cipher-Suite `SSL_RSA_WITH_AES_256_CBC_SHA256` verwendet. Die `mqipt.conf`-Konfigurationsdatei in diesem Szenario basiert auf der Konfigurationsdatei, die im „Ordnungsgemäße Funktion von MQIPT überprüfen“ auf Seite 177 -Szenario erstellt wurde.

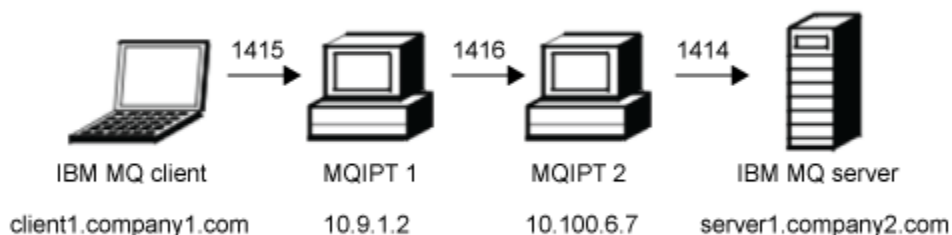


Abbildung 32. SSL/TLS-Client-Netzdiagramm

Dieses Diagramm zeigt die Verbindung vom IBM MQ-Client (client1.company1.com an Port 1415) über zwei Instanzen von MQIPT zum IBM MQ-Server (server1.company2.com an Port 1414).

Vorgehensweise

Führen Sie zur Authentifizierung eines TLS-Clients die folgenden Schritte aus:

1. Auf dem System MQIPT 1:

a) Bearbeiten Sie `mqipt.conf` und fügen Sie die folgende Routendefinition hinzu:

```
[route]
ListenerPort=1415
Destination=10.100.6.7
DestinationPort=1416
SSLClient=true
SSLClientKeyRing=C:\mqipt\samples\ssl\sslSample.pfx
SSLClientKeyRingPW=<mqiptPW>1!PCaB1HWzFM0p43ngjwgArg==!6N/vsbqr7iqMhFN+wozxQ==
SSLClientCipherSuites=SSL_RSA_WITH_AES_256_CBC_SHA256
```

b) Öffnen Sie eine Eingabeaufforderung und starten Sie MQIPT:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt `C:\mqiptHome` die Position der MQIPT-Konfigurationsdatei `mqipt.conf` und `ipt1` den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt1
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1415 is starting and will forward messages to :
MQCPI034 ...10.100.6.7(1416)
MQCPI035 ...using MQ protocol
```

```

MQCPI036 ...SSL Client side enabled with properties :
MQCPI139 .....secure socket protocols <NULL>
MQCPI031 .....cipher suites SSL_RSA_WITH_AES_256_CBC_SHA256
MQCPI032 .....key ring file C:\mqipt\samples\ssl\sslSample.pfx
MQCPI047 .....CA key ring file <NULL>
MQCPI071 .....site certificate uses
UID=*,CN=*,T=*,OU=*,DC=*,O=*,STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*
MQCPI038 .....peer certificate uses
UID=*,CN=*,T=*,OU=*,DC=*,O=*,STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*
MQCPI078 Route 1415 ready for connection requests

```

2. Auf dem System MQIPT 2:

- a) Bearbeiten Sie mqipt.conf und fügen Sie die folgende Routendefinition hinzu:

```

[route]
ListenerPort=1416
Destination=Server1.company2.com
DestinationPort=1414
SSLServer=true
SSLServerAskClientAuth=true
SSLServerKeyRing=C:\mqipt\samples\ssl\sslSample.pfx
SSLServerKeyRingPW=<mqiptPW>!PCaB1HWrFMOp43ngjwgArg==!6N/vsbqru7iqMhFN+wozxQ==
SSLServerCipherSuites=SSL_RSA_WITH_AES_256_CBC_SHA256

```

- b) Öffnen Sie eine Eingabeaufforderung und starten Sie MQIPT:

```

C:
cd \mqipt\bin
mqipt .. -n ipt2

```

Dabei zeigt .. an, dass sich die MQIPT-Konfigurationsdatei mqipt.conf im übergeordneten Verzeichnis befindet und ipt2 der Name ist, der der Instanz von MQIPT zugeordnet wurde.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```

5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt2
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqipt\logs will be used to store the log files
MQCPI006 Route 1416 is starting and will forward messages to :
MQCPI034 ...Server1.company2.com(1414)
MQCPI035 ...using MQ protocol
MQCPI037 ...SSL Server side enabled with properties :
MQCPI139 .....secure socket protocols <NULL>
MQCPI031 .....cipher suites SSL_RSA_WITH_AES_256_CBC_SHA256
MQCPI032 .....key ring file C:\mqipt\samples\ssl\sslSample.pfx
MQCPI047 .....CA key ring file <NULL>
MQCPI071 .....site certificate uses
UID=*,CN=*,T=*,OU=*,DC=*,O=*,STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*
MQCPI038 .....peer certificate uses
UID=*,CN=*,T=*,OU=*,DC=*,O=*,STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*
MQCPI033 .....client authentication set to true
MQCPI078 Route 1416 ready for connection requests

```

3. Geben Sie in einer Eingabeaufforderung auf dem IBM MQ-Clientsystem die folgenden Befehle ein:

- a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=MQIPT.CONN.CHANNEL/tcp/10.9.1.2(1415)
```

- b) Reihnen Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE MQIPT.QM1
Hello world
```

Drücken Sie zweimal die Eingabetaste, nachdem Sie die Nachrichtenzeichenfolge eingegeben haben.

- c) Rufen Sie die Nachricht ab:

```
amqsgetc MQIPT.LOCAL.QUEUE MQIPT.QM1
```

Die Nachricht "Hello world" wird zurückgegeben.

TLS-Client und -Server authentifizieren

Sie können MQIPT als TLS-Server und -Client ausführen, um die eingehende TLS-Sitzung zu beenden und Daten über eine separate TLS-Verbindung an das Ziel weiterzuleiten.

Vorbereitende Schritte

- Führen Sie die unter „Erste Schritte mit IBM MQ Internet Pass-Thru“ auf Seite 175 aufgelisteten Schritte aus.
- Lesen Sie den Abschnitt [SSL/TLS-Unterstützung in MQIPT](#).

Anmerkung: In diesem Szenario werden zur Vereinfachung selbst signierte Zertifikate verwendet. Verwenden Sie in Produktionsumgebungen keine selbst signierten Zertifikate. Fordern Sie stattdessen Zertifikate an, die von einer anerkannten Zertifizierungsstelle signiert wurden.

Informationen zu diesem Vorgang

Die Verbindung wird zwischen einem IBM MQ -Client und einem IBM MQ -Server über eine einzelne Instanz von MQIPT hergestellt. Die Verbindungen zwischen dem IBM MQ -Client und MQIPT sowie zwischen MQIPT und dem IBM MQ -Server verwenden beide TLS. Daher ist die MQIPT -Route sowohl ein TLS-Server als auch ein TLS-Client.

Beim TLS-Handshake zwischen dem Client und MQIPT senden der Client und MQIPT ihre Zertifikate an einander, um die Verbindung zu authentifizieren. Wenn die Verbindung zwischen dem Client und MQIPT hergestellt ist, stellt MQIPT eine separate TLS-Verbindung zum IBM MQ -Server her. MQIPT und der IBM MQ -Server senden ihre Zertifikate an einander, um die Verbindung zu authentifizieren.

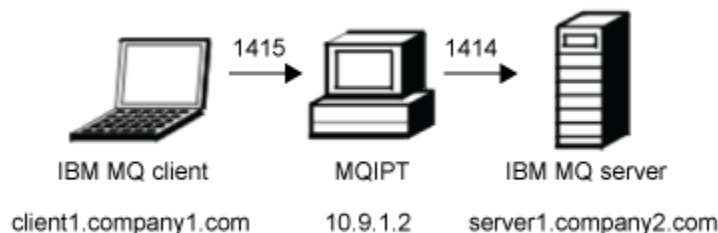


Abbildung 33. SSL/TLS-Server und -Client-Netzdiagramm

Dieses Diagramm zeigt die Verbindung vom IBM MQ -Client (client1.company1.com) über eine einzelne Instanz von MQIPT zum IBM MQ -Server (server1.company2.com an Port 1414).

Vorgehensweise

Führen Sie die folgenden Schritte aus, um eine MQIPT -Route als TLS-Server und als TLS-Client zu konfigurieren:

1. Führen Sie auf dem System, auf dem der Client ausgeführt wird, die folgenden Schritte aus, um den IBM MQ -Client für die Verwendung einer TLS-Verbindung zu konfigurieren.
 - a) Erstellen Sie für den Client ein Schlüsselrepository und Geben Sie den folgenden Befehl ein, um ein neues Schlüsselrepository mit dem Namen client-key.kdb im Verzeichnis C:\ProgramData\IBM\MQ zu erstellen:

```
runmqakm -keydb -create -db C:\ProgramData\IBM\MQ\clientkey.kdb -pw password -stash
```

Dabei ist *Kennwort* das Kennwort des Schlüsselrepositorys.

Weitere Informationen finden Sie unter [Schlüsselrepository unter AIX, Linux, and Windows einrichten](#).

- b) Erstellen Sie im Schlüsselrepository des Clients, das Sie in Schritt „1.a“ auf Seite 187 erstellt haben, ein persönliches Zertifikat für den Client.

Geben Sie den folgenden Befehl ein, um ein neues selbst signiertes Zertifikat mit der Bezeichnung `clientcert` für den Client zu erstellen:

```
runmqakm -cert -create -db C:\ProgramData\IBM\MQ\clientkey.kdb -stashed  
-label clientcert -dn CN=Client
```

Weitere Informationen finden Sie im Abschnitt [Selbst signiertes persönliches Zertifikat unter AIX, Linux, and Windows erstellen](#).

- c) Extrahieren Sie das Zertifikat aus dem Clientschlüsselrepository.

Geben Sie den folgenden Befehl ein, um das Clientzertifikat zu extrahieren:

```
runmqakm -cert -extract -db C:\ProgramData\IBM\MQ\clientkey.kdb -stashed -label clientcert  
-target C:\ProgramData\IBM\MQ\client.crt -format ascii
```

Weitere Informationen finden Sie unter [Öffentlichen Teil eines selbst signierten Zertifikats aus einem Schlüsselrepository unter AIX, Linux, and Windowsextrahieren](#).

- d) Kopieren Sie die extrahierte Zertifikatsdatei auf das System, auf dem MQIPT ausgeführt wird.
2. Führen Sie auf dem System, auf dem der IBM MQ -Server ausgeführt wird, die folgenden Schritte aus, um den Warteschlangenmanager für eine TLS-Verbindung zu konfigurieren.

- a) Erstellen Sie für den Warteschlangenmanager ein Schlüsselrepository.

Geben Sie den folgenden Befehl ein, um ein neues Schlüsselrepository namens `key.kdb` für den Warteschlangenmanager zu erstellen:

```
runmqakm -keydb -create -db C:\ProgramData\IBM\MQ\qmgrs\MQIPT!QM1\ssl\key.kdb -pw password  
-stash
```

Dabei ist *Kennwort* das Kennwort des Schlüsselrepositorys.

Stellen Sie sicher, dass dem Benutzer 'mqm' Lesezugriff auf die Stashdatei `C:\ProgramData\IBM\MQ\qmgrs\MQIPT!QM1\ssl\key.sth` erteilt wird, die der Befehl erstellt, um das Kennwort des verschlüsselten Schlüsselrepositorys zu speichern.

Weitere Informationen finden Sie unter [Schlüsselrepository unter AIX, Linux, and Windows einrichten](#).

- b) Erstellen Sie im Schlüsselrepository des Warteschlangenmanagers, das Sie in Schritt „2.a“ auf Seite 188 erstellt haben, ein persönliches Zertifikat für den Warteschlangenmanager.

Geben Sie den folgenden Befehl ein, um ein neues selbst signiertes Zertifikat mit der Bezeichnung `ibmwebspheremqmqipt.qm1` für den Warteschlangenmanager zu erstellen:

```
runmqakm -cert -create -db C:\ProgramData\IBM\MQ\qmgrs\MQIPT!QM1\ssl\key.kdb -stashed  
-label ibmwebspheremqmqipt.qm1 -dn CN=MQIPT.QM1
```

Weitere Informationen finden Sie im Abschnitt [Selbst signiertes persönliches Zertifikat unter AIX, Linux, and Windows erstellen](#).

- c) Extrahieren Sie das Zertifikat aus dem Schlüsselrepository des Warteschlangenmanagers.

Geben Sie den folgenden Befehl ein, um das Warteschlangenmanagerzertifikat zu extrahieren:

```
runmqakm -cert -extract -db C:\ProgramData\IBM\MQ\qmgrs\MQIPT!QM1\ssl\key.kdb -stashed  
-label ibmwebspheremqmqipt.qm1  
-target C:\ProgramData\IBM\MQ\qmgrs\MQIPT!QM1\ssl\mqipt.qm1.crt -format ascii
```

Weitere Informationen finden Sie unter [Öffentlichen Teil eines selbst signierten Zertifikats aus einem Schlüsselrepository unter AIX, Linux, and Windowsextrahieren](#).



- d) Kopieren Sie die extrahierte Zertifikatsdatei auf das System, auf dem MQIPT ausgeführt wird.

- e) Geben Sie den folgenden MQSC-Befehl aus, um den MQIPT.CONN.CHANNEL -Serververbindungs-kanal zur Verwendung von TLS:

```
ALTER CHANNEL(MQIPT.CONN.CHANNEL) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCIPH(ANY_TLS12_OR_HIGHER)
```

3. Führen Sie auf dem System, auf dem MQIPT ausgeführt wird, die folgenden Schritte aus, um die MQIPT -Route für die Verwendung von TLS zu konfigurieren.

- a) Erstellen Sie ein persönliches Zertifikat für MQIPT in einem PKCS #12 -Schlüsselrepository.

  Geben Sie den folgenden Befehl ein, um ein neues selbst signiertes Zertifikat mit der Bezeichnung `mqiptcert` zu erstellen:

```
mqiptKeytool -genkeypair -keystore C:\mqiptHome\ssl\mqipt.p12 -storetype pkcs12 -storepass password -alias mqiptcert -dname "CN=MQIPT Test Certificate" -keyalg RSA -keysize 2048 -sigalg SHA256WithRSA
```

Dabei ist *Kennwort* das Kennwort des Schlüsselrepositorys.



- b) Geben Sie den folgenden Befehl ein, um das Clientzertifikat und das Warteschlangenmanagerzertifikat zum MQIPT -Schlüsselrepository hinzuzufügen:

```
mqiptKeytool -importcert -keystore C:\mqiptHome\ssl\mqipt.p12 -storetype pkcs12 -storepass password -file client.crt
mqiptKeytool -importcert -keystore C:\mqiptHome\ssl\mqipt.p12 -storetype pkcs12 -storepass password -file mqipt.qm1.crt
```

Dabei ist *Kennwort* das Kennwort des Schlüsselrepositorys, *client.crt* die Clientzertifikatsdatei, die Sie in Schritt „1.c“ auf Seite 188 erstellt haben, und *mqipt.qm1.crt* das in Schritt „2.c“ auf Seite 188 erstellte Warteschlangenmanagerzertifikat.

- c) Extrahieren Sie das Zertifikat MQIPT aus dem Schlüsselrepository.

  Geben Sie den folgenden Befehl ein, um das Zertifikat MQIPT zu extrahieren:

```
mqiptKeytool -exportcert -keystore C:\mqiptHome\ssl\mqipt.p12 -storetype pkcs12 -storepass password -alias mqiptcert -file C:\mqiptHome\ssl\mqipt.crt -rfc
```

Dabei ist *Kennwort* das Kennwort des Schlüsselrepositorys.

- d) Kopieren Sie die extrahierte Zertifikatsdatei auf das System, auf dem der Client ausgeführt wird, und auf das System, auf dem der IBM MQ -Server ausgeführt wird.
e) Geben Sie den folgenden Befehl ein, um das Kennwort für das MQIPT -Schlüsselrepository zu verschlüsseln:

```
mqiptPW
```

Geben Sie bei entsprechender Aufforderung das Kennwort für das Schlüsselrepository ein, das Sie bei der Erstellung des Schlüsselrepositorys in Schritt „3.a“ auf Seite 189 angegeben haben.

- f) Bearbeiten Sie die Datei `mqipt.conf` und fügen Sie die folgende Routendefinition hinzu:

```
[route]
ListenerPort=1415
Destination=server1.company2.com
DestinationPort=1414
SSLServer=true
SSLServerKeyRing=C:\mqiptHome\ssl\mqipt.p12
SSLServerKeyRingPW=encrypted_password
SSLClient=true
SSLClientKeyRing=C:\mqiptHome\ssl\mqipt.p12
SSLClientKeyRingPW=encrypted_password
```

Dabei ist *verschlüsseltes_Kennwort* das Kennwort des verschlüsselten Schlüsselrepositorys, das durch Ausführung des Befehls **mqiptPW** in Schritt „3.e“ auf Seite 189 erstellt wird.

4. Fügen Sie das Zertifikat MQIPT zum Schlüsselrepository des Clients und zum Schlüsselrepository des Warteschlangenmanagers hinzu.

- a) Geben Sie auf dem System, auf dem der Client ausgeführt wird, den folgenden Befehl ein, um das Zertifikat MQIPT zum Clientschlüsselrepository hinzuzufügen:

```
runmqakm -cert -add -db C:\ProgramData\IBM\MQ\clientkey.kdb -stashed  
-label mqiptcert -file mqipt.crt -format ascii
```

Dabei ist *mqipt.crt* die MQIPT -Zertifikatdatei, die Sie in Schritt „3.c“ auf Seite 189 erstellt haben.

- b) Geben Sie auf dem System, auf dem der IBM MQ -Server ausgeführt wird, den folgenden Befehl ein, um das Zertifikat MQIPT zum Schlüsselrepository des Warteschlangenmanagers hinzuzufügen:

```
runmqakm -cert -add -db C:\ProgramData\IBM\MQ\mqgrs\MQIPT!QM1\ssl\key.kdb -stashed  
-label mqiptcert -file mqipt.crt -format ascii
```

Dabei ist *mqipt.crt* die MQIPT -Zertifikatdatei, die Sie in Schritt „3.c“ auf Seite 189 erstellt haben.

Weitere Informationen finden Sie unter CA-Zertifikat (oder öffentlichen Teil eines selbst signierten Zertifikats) zu einem Schlüsselrepository auf AIX, Linux, and Windows -Systemen hinzuzufügen.

5. Öffnen Sie auf dem System, auf dem MQIPT ausgeführt wird, eine Eingabeaufforderung und geben Sie die folgenden Befehle ein, um MQIPT zu starten:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt C:\mqiptHome die Position der MQIPT-Konfigurationsdatei *mqipt.conf* und *ipt1* den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved  
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting  
MQCPI004 Reading configuration information from mqipt.conf  
MQCPI152 MQIPT name is ipt1  
MQCPI021 Password checking has been enabled on the command port  
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files  
MQCPI006 Route 1415 is starting and will forward messages to :  
MQCPI034 ....server1.company2.com(1414)  
MQCPI035 ....using MQ protocol  
MQCPI036 ....SSL Client side enabled with properties :  
MQCPI139 .....secure socket protocols <NULL>  
MQCPI031 .....cipher suites <NULL>  
MQCPI032 .....key ring file C:\mqiptHome\ssl\mqipt.p12  
MQCPI047 .....CA key ring file <NULL>  
MQCPI071 .....site certificate uses  
UID=*,CN=*,T=*,OU=*,DC=*,O=*,STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*  
MQCPI038 .....peer certificate uses  
UID=*,CN=*,T=*,OU=*,DC=*,O=*,STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*  
MQCPI037 ....SSL Server side enabled with properties :  
MQCPI139 .....secure socket protocols <NULL>  
MQCPI031 .....cipher suites <NULL>  
MQCPI032 .....key ring file C:\mqiptHome\ssl\mqipt.p12  
MQCPI047 .....CA key ring file <NULL>  
MQCPI071 .....site certificate uses  
UID=*,CN=*,T=*,OU=*,DC=*,O=*,STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*  
MQCPI038 .....peer certificate uses  
UID=*,CN=*,T=*,OU=*,DC=*,O=*,STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*  
MQCPI033 .....client authentication set to false  
MQCPI078 Route 1415 ready for connection requests
```

6. Geben Sie bei einer Eingabeaufforderung auf dem IBM MQ-Clientsystem den folgenden Befehl ein, um das TLS-Beispielprogramm auszuführen:

```
AMQSSSLC -m MQIPT.QM1 -c MQIPT.CONN.CHANNEL -x 10.9.1.2(1415)  
-k "C:\ProgramData\IBM\MQ\clientkey" -l clientcert -s ANY_TLS12_OR_HIGHER
```

Die folgende Nachricht gibt an, dass die Anwendung erfolgreich eine Verbindung zum Warteschlangenmanager hergestellt hat:

HTTP-Tunnelung konfigurieren

In diesem Szenario können Sie eine einfache Verbindung zwischen zwei Instanzen von MQIPT über HTTP testen.

Vorbereitende Schritte

Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „Erste Schritte mit IBM MQ Internet Pass-Thru“ auf Seite 175 aufgelisteten vorausgesetzten Tasks abgeschlossen haben.

Anmerkung: V9.4.0 V9.4.0 Ab IBM MQ 9.4.0 akzeptieren MQIPT -Routen standardmäßig keine HTTP-Verbindungen mehr. Routen müssen so konfiguriert werden, dass HTTP-Verbindungen über die Eigenschaft **AllowedProtocols** akzeptiert werden.

Informationen zu diesem Vorgang

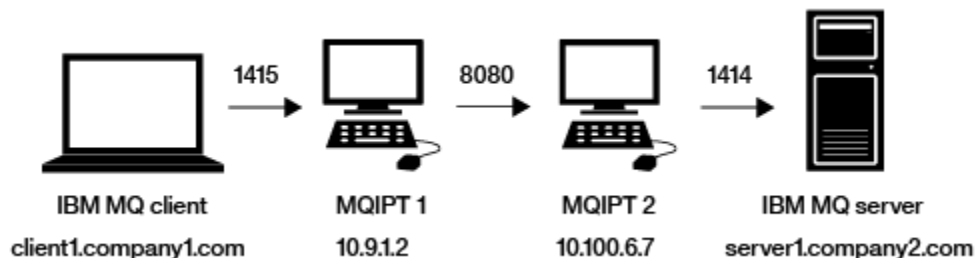


Abbildung 34. HTTP-Tunneling-Netzdiagramm

Dieses Diagramm zeigt die Verbindung vom IBM MQ-Client (client1.company1.com an Port 1415) über zwei Instanzen von MQIPT, mit Tunnelung der Verbindung über HTTP und schließlich zum IBM MQ-Server (server1.company2.com an Port 1414).

Vorgehensweise

Führen Sie die folgenden Schritte aus, um HTTP-Tunneling zwischen zwei Instanzen von MQIPT zu konfigurieren:

1. Auf dem System MQIPT 1:
 - a) Bearbeiten Sie `mqipt.conf` und fügen Sie die folgende Routendefinition hinzu:

```
[route]
ListenerPort=1415
Destination=10.100.6.7
DestinationPort=8080
HTTP=true
HTTPServer=10.100.6.7
HTTPServerPort=8080
```

- b) Öffnen Sie eine Eingabeaufforderung und starten Sie MQIPT:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt `C:\mqiptHome` die Position der MQIPT-Konfigurationsdatei `mqipt.conf` und `ipt1` den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
```

```
MQCPI152 MQIPT name is ipt1
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1415 is starting and will forward messages to :
MQCPI034 ....10.100.6.7(8080)
MQCPI035 ...using HTTP
MQCPI066 ....and HTTP server at 10.100.6.7(8080)
MQCPI078 Route 1415 ready for connection requests
```

2. Auf dem System MQIPT 2:

- a) Bearbeiten Sie `mqipt.conf` und fügen Sie die folgende Routendefinition hinzu:

```
[route]
ListenerPort=8080
Destination=Server1.company2.com
DestinationPort=1414
AllowedProtocols=http
```

- b) Öffnen Sie eine Eingabeaufforderung und starten Sie MQIPT:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt2
```

Dabei gibt `C:\mqiptHome` die Position der MQIPT-Konfigurationsdatei `mqipt.conf` und `ipt2` den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt2
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 8080 is starting and will forward messages to :
MQCPI034 ....Server1.company2.com(1414)
MQCPI035 ...using MQ protocols
MQCPI158 ...connection protocols accepted: HTTP
MQCPI078 Route 8080 ready for connection requests
```

3. Geben Sie bei einer Eingabeaufforderung auf dem IBM MQ-Client die folgenden Befehle ein:

- a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=MQIPT.CONN.CHANNEL/tcp/10.9.1.2(1415)
```

- b) Reihnen Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE MQIPT.QM1
Hello world
```

Drücken Sie zweimal die Eingabetaste, nachdem Sie die Nachrichtenzeichenfolge eingegeben haben.

- c) Rufen Sie die Nachricht ab:

```
amqsgetc MQIPT.LOCAL.QUEUE MQIPT.QM1
```

Die Nachricht "Hello world" wird zurückgegeben.

Zugriffssteuerung konfigurieren

In diesem Szenario können Sie MQIPT so einrichten, dass nur Verbindungen von bestimmten Clients akzeptiert werden. Hierfür fügen Sie unter Verwendung von Java security manager Sicherheitsprüfungen am MQIPT-Listener-Port hinzu.

Vorbereitende Schritte

- Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „[Erste Schritte mit IBM MQ Internet Pass-Thru](#)“ auf Seite 175 aufgelisteten vorausgesetzten Tasks abgeschlossen haben.

Informationen zu diesem Vorgang

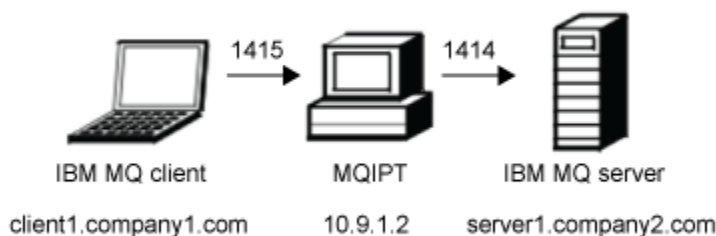


Abbildung 35. Zugriffssteuerungsnetzdiagramm

Dieses Diagramm zeigt die Verbindung vom IBM MQ-Client (client1.company1.com an Port 1415) über MQIPT zum IBM MQ-Server (server1.company2.com an Port 1414).

Vorgehensweise

Führen Sie zur Konfiguration der Zugriffssteuerung die folgenden Schritte aus:

1. Konfigurieren Sie MQIPT:

- a) Kopieren Sie die Java security manager-Beispielrichtlinie in das Ausgangsverzeichnis von MQIPT und geben Sie dazu bei einer Eingabeaufforderung den folgenden Befehl ein:

```
copy C:\mqipt\samples\mqiptSample.policy C:\mqiptHome\mqipt.policy
```

- b) Starten Sie das Dienstprogramm "Policy Tool" mit dem folgenden Befehl:

```
C:\mqipt\java\jre\bin\policytool
```

- c) Klicken Sie auf **Datei > Öffnen** und wählen Sie C:\mqiptHome\mqipt.policy aus

- d) Klicken Sie auf **Edit Policy Entry** (Richtlinieneintrag bearbeiten) und ändern Sie die Codebasis (CodeBase) von:

```
file:/C:/Program Files/IBM/IBM MQ Internet Pass-Thru/lib/com.ibm.mq.ipt.jar
```

in:

```
file:/C:/mqipt/lib/com.ibm.mq.ipt.jar
```

- e) Ändern Sie die Dateiberechtigungen für die Verzeichnisse IBM MQ Internet Pass-Thru, errors und logs von:

```
C:\Program Files\IBM\IBM MQ Internet Pass-Thru
```

in:

```
C:\mqiptHome
```

- f) Ändern Sie die anderen Dateiberechtigungen von:

```
C:\Program Files\IBM\IBM MQ Internet Pass-Thru
```

in:

```
C:\mqipt
```

- g) Klicken Sie auf **Add Permission** (Berechtigung hinzufügen).

Füllen Sie die Felder wie folgt aus:

Permission (Berechtigung): java.net.SocketPermission

Target (Ziel): client1.company1.com:1024-

Actions (Aktionen): accept, listen, resolve

- h) Klicken Sie auf **File > Save** (Datei > Speichern), um die Änderungen an der Richtliniendatei zu speichern.

- i) Bearbeiten Sie mqipt.conf.

- i) Fügen Sie dem Abschnitt [global] die folgenden zwei Eigenschaften hinzu:

```
SecurityManager=true  
SecurityManagerPolicy=C:\mqiptHome\mqipt.policy
```

- ii) Fügen Sie die folgende Routendefinition hinzu:

```
[route]  
ListenerPort=1415  
Destination=server1.company2.com  
DestinationPort=1414
```

2. Starten Sie MQIPT:

Öffnen Sie eine Eingabeaufforderung und geben Sie Folgendes ein:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt C:\mqiptHome die Position der MQIPT-Konfigurationsdatei mqipt.conf und ipt1 den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved  
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting  
MQCPI004 Reading configuration information from mqipt.conf  
MQCPI152 MQIPT name is ipt1  
MQCPI055 Setting the java.security.policy to C:\mqiptHome\mqipt.policy  
MQCPI053 Starting the Java Security Manager  
MQCPI021 Password checking has been enabled on the command port  
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files  
MQCPI006 Route 1415 has started and will forward messages to :  
MQCPI034 ....server1.company2.com(1414)  
MQCPI035 ....using MQ protocol  
MQCPI078 Route 1415 ready for connection requests
```

3. Geben Sie in einer Eingabeaufforderung auf dem IBM MQ-Clientsystem die folgenden Befehle ein:

- a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=MQIPT.CONN.CHANNEL/tcp/10.9.1.2(1415)
```

- b) Reißen Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE MQIPT.QM1  
Hello world
```

Drücken Sie zweimal die Eingabetaste, nachdem Sie die Nachrichtenzeichenfolge eingegeben haben.

- c) Rufen Sie die Nachricht ab:

```
amqsgetc MQIPT.LOCAL.QUEUE MQIPT.QM1
```

Die Nachricht "Hello world" wird zurückgegeben.

SOCKS-Proxy konfigurieren

In diesem Szenario können Sie MQIPT für die Verwendung als SOCKS-Proxy einrichten.

Vorbereitende Schritte

- Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „Erste Schritte mit IBM MQ Internet Pass-Thru“ auf Seite 175 aufgelisteten vorausgesetzten Tasks abgeschlossen haben.
- Aktivieren Sie SOCKS auf dem gesamten IBM MQ-Computer oder nur auf den IBM MQ-Clientanwendungen **amqsputc** und **amqsgetc**.
- Konfigurieren Sie den SOCKS-Client wie folgt:
 1. Verwenden Sie MQIPT als SOCKS-Proxy.
 2. Aktivieren Sie die Unterstützung für SOCKS 5.
 3. Inaktivieren Sie die Benutzerauthentifizierung.
 4. Schränken Sie die Verbindungen zur MQIPT-Netzadresse ein.

Informationen zu diesem Vorgang

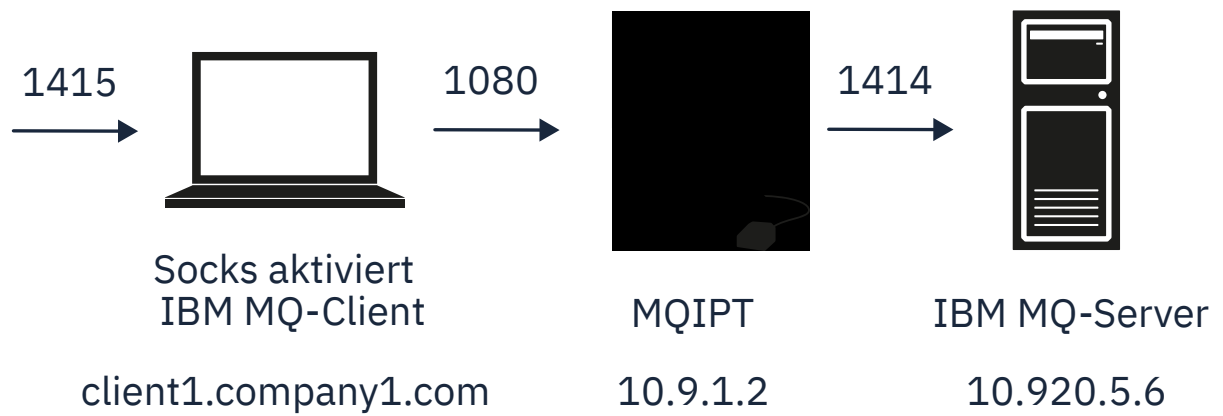


Abbildung 36. SOCKS-Proxy-Netzdiagramm

In diesem Diagramm ist der Verbindungsfluss vom IBM MQ-Client (client1.company1.com an Port 1415) über MQIPT zum IBM MQ-Server (server1.company2.com an Port 1414) dargestellt.

Vorgehensweise

Führen Sie die folgenden Schritte aus, um einen SOCKS-Proxy zu konfigurieren:

1. Konfigurieren und starten Sie MQIPT:

- a) Bearbeiten Sie `mqipt.conf` und fügen Sie die folgende Routendefinition hinzu:

```
[route]
ListenerPort=1080
Destination=server1.company2.com
DestinationPort=1414
SocksServer=true
```

Die Werte der Routeneigenschaften **Destination** und **DestinationPort** werden ignoriert, da das tatsächliche Ziel beim SOCKS-Handshakeverfahren vom IBM MQ-Client abgerufen wird.

- b) Öffnen Sie eine Eingabeaufforderung und starten Sie MQIPT:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt `C:\mqiptHome` die Position der MQIPT-Konfigurationsdatei `mqipt.conf` und `ipt1` den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt1
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1080 has started and will forward messages to :
MQCPI034 ...server1.company2.com(1414)
MQCPI035 ...using MQ protocol
MQCPI052 ...SOCKS server side enabled
MQCPI078 Route 1080 ready for connection requests
```

2. Geben Sie in einer Eingabeaufforderung auf dem IBM MQ-Clientsystem die folgenden Befehle ein:

- a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=MQIPT.CONN.CHANNEL/tcp/10.20.5.6(1414)
```

- b) Reihen Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE MQIPT.QM1
Hello world
```

Drücken Sie zweimal die Eingabetaste, nachdem Sie die Nachrichtenzeichenfolge eingegeben haben.

- c) Rufen Sie die Nachricht ab:

```
amqsgetc MQIPT.LOCAL.QUEUE MQIPT.QM1
```

Die Nachricht "Hello world" wird zurückgegeben.

SOCKS-Client konfigurieren

In diesem Szenario können Sie MQIPT unter Verwendung eines vorhandenen SOCKS-Proxys so ausführen, als wäre es SOCKS-fähig.

Das Verfahren ist ähnlich wie im Szenario „SOCKS-Proxy konfigurieren“ auf Seite 195, außer dass MQIPT anstelle des IBM MQ-Clients eine SOCKS-fähige Verbindung herstellt.

Vorbereitende Schritte

Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „Erste Schritte mit IBM MQ Internet Pass-Thru“ auf Seite 175 aufgelisteten vorausgesetzten Tasks abgeschlossen haben.

Informationen zu diesem Vorgang

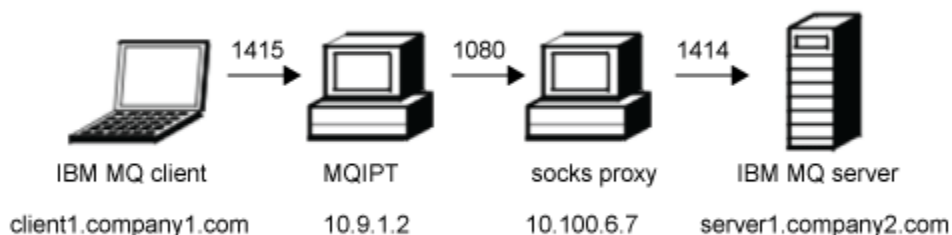


Abbildung 37. SOCKS-Client-Netzdiagramm

Dieses Diagramm zeigt die Netzverbindung vom IBM MQ-Client (`client1.company1.com` an Port 1415) über MQIPT und den SOCKS-Proxy (an Port 1080) zum IBM MQ-Server (`server1.company2.com` an Port 1414).

Vorgehensweise

Führen Sie die folgenden Schritte aus, um einen SOCKS-Client zu konfigurieren:

1. Konfigurieren Sie MQIPT.

Bearbeiten Sie auf dem MQIPT -Computer `mqipt.conf` und fügen Sie eine Routendefinition hinzu:

```
[route]
ListenerPort=1415
Destination=server1.company2.com
DestinationPort=1414
SocksClient=true
SocksProxyHost=10.9.6.7
SocksProxyPort=1080
```

2. Starten Sie MQIPT.

Öffnen Sie eine Eingabeaufforderung und geben Sie Folgendes ein:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt `C:\mqiptHome` die Position der MQIPT-Konfigurationsdatei `mqipt.conf` und `ipt1` den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt1
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1415 has started and will forward messages to :
MQCPI034 ....server1.company2.com(1414)
MQCPI035 ....using MQ protocol
MQCPI039 ....and SOCKS proxy at 10.9.6.7(1080)
MQCPI078 Route 1415 ready for connection requests
```

3. Geben Sie bei einer Eingabeaufforderung auf dem IBM MQ-Client die folgenden Befehle ein:

- a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=MQIPT.CONN.CHANNEL/tcp/10.9.1.2(1415)
```

b) Reihen Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE MQIPT.QM1  
Hello world
```

Drücken Sie zweimal die Eingabetaste, nachdem Sie die Nachrichtenzeichenfolge eingegeben haben.

c) Rufen Sie die Nachricht ab:

```
amqsgetc MQIPT.LOCAL.QUEUE MQIPT.QM1
```

Die Nachricht "Hello world" wird zurückgegeben.

Clustering-Unterstützung für MQIPT konfigurieren

In diesem Szenario können Sie eine Clustering-Umgebung einrichten.

Vorbereitende Schritte

- Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „[Erste Schritte mit IBM MQ Internet Pass-Thru](#)“ auf Seite 175 aufgelisteten vorausgesetzten Tasks abgeschlossen haben.
- Auf dem IBM MQ-Server LONDON:
 - Definition eines Warteschlangenmanagers mit der Bezeichnung LONDON.
 - Definition eines Serververbindungskanals mit der Bezeichnung MQIPT.CONN.CHANNEL.
 - Start eines TCP/IP-Listeners für LONDON an Port 1414.
 - SOCKS-fähiger Warteschlangenmanager.
- Auf dem IBM MQ-Server NEWYORK:
 - Definition eines Warteschlangenmanagers mit der Bezeichnung NEWYORK.
 - Definition eines Serververbindungskanals mit der Bezeichnung MQIPT.CONN.CHANNEL.
 - Start eines TCP/IP-Listeners für NEWYORK an Port 1414.
 - SOCKS-fähiger Warteschlangenmanager.

Anmerkung: Damit ein Warteschlangenmanager SOCKS verwenden kann, aktivieren Sie den gesamten Computer oder nur die IBM MQ-Serveranwendung. Konfigurieren Sie den SOCKS-Client wie folgt:

- Verweisen Sie den Client auf MQIPT als SOCKS-Proxy.
- Aktivieren Sie die Unterstützung für SOCKS V5.
- Inaktivieren Sie die Benutzerauthentifizierung.
- Stellen Sie nur ferne Verbindungen zu MQIPT her.

Informationen zu diesem Vorgang

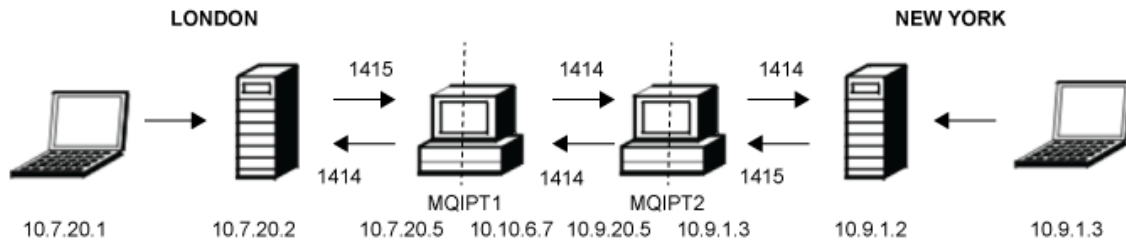


Abbildung 38. Diagramm des Clusternetzes

In diesem Diagramm werden die Verbindungen zwischen den IBM MQ-Clients über MQIPT zu den IBM MQ-Servern gezeigt.

Es kann nur eine Anwendung für einen angegebenen Port auf dem gleichen Computer empfangsbereit sein. Wenn Port 1414 bereits verwendet wird, wählen Sie einen freien Port und ersetzen Sie diesen in den Beispielen entsprechend.

Anschließend können Sie die Routen zwischen den Warteschlangenmanagern testen, indem Sie eine Nachricht in die lokale Warteschlange auf dem Server LONDON einreihen und sie vom Server NEWYORK abrufen.

Vorgehensweise

Führen Sie zur Konfiguration der Clustering-Unterstützung für MQIPT die folgenden Schritte aus:

1. Richten Sie den den Server LONDON ein.

Öffnen Sie eine Eingabeaufforderung und geben Sie die folgenden Befehle ein:

```
runmqsc
DEFINE CHANNEL(TO.LONDON) +
  CHLTYPE(CLUSRCVR) TRPTYPE(TCP) +
  CLUSTER(INVENTORY) +
  CONNAME('10.10.6.7(1414)')
DEFINE CHANNEL(TO.NEWYORK) +
  CHLTYPE(CLUSSDR) TRPTYPE(TCP) +
  CLUSTER(INVENTORY) +
  CONNAME('10.9.20.5(1414)')
```

2. Richten Sie den Server NEWYORK ein.

Öffnen Sie eine Eingabeaufforderung und geben Sie die folgenden Befehle ein:

```
runmqsc
ALTER QMGR REPOS(INVENTORY)
DEFINE QLOCAL(MQIPT.LOCAL.QUEUE) +
  CLUSTER(INVENTORY)
DEFINE CHANNEL(TO.NEWYORK) +
  CHLTYPE(CLUSRCVR) TRPTYPE(TCP) +
  CLUSTER(INVENTORY) +
  CONNAME('10.9.20.5(1414)')
DEFINE CHANNEL(TO.LONDON) +
  CHLTYPE(CLUSSDR) TRPTYPE(TCP) +
  CLUSTER(INVENTORY) +
  CONNAME('10.10.6.7(1414)')
```

3. Konfigurieren Sie MQIPT 1.

Bearbeiten Sie `mqipt.conf` und fügen Sie die folgenden Routendefinitionen hinzu:

```
[route]
Name=LONDON to NEWYORK
ListenerPort=1415
Destination=10.9.20.5
DestinationPort=1414
SocksServer=true
```



```
[route]
Name=MQIPT1 to LONDON
ListenerPort=1414
Destination=10.7.20.2
DestinationPort=1414
```

4. Starten Sie MQIPT 1.

Öffnen Sie eine Eingabeaufforderung und geben Sie Folgendes ein:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt C:\mqiptHome die Position der MQIPT-Konfigurationsdatei mqipt.conf und ipt1 den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt1
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1415 has started and will forward messages to :
MQCPI034 ....10.9.20.5(1414)
MQCPI035 ....using MQ protocol
MQCPI052 ....SOCKS server side enabled
MQCPI078 Route 1415 ready for connection requests
MQCPI006 Route 1414 has started and will forward messages to :
MQCPI034 ....10.7.20.2(1414)
MQCPI035 ....using MQ protocol
MQCPI078 Route 1414 ready for connection requests
```

5. Konfigurieren Sie MQIPT 2.

Bearbeiten Sie mqipt.conf und fügen Sie die folgenden Routendefinitionen hinzu:

```
[route]
Name=NEWYORK to LONDON
ListenerPort=1415
Destination=10.10.6.7
DestinationPort=1414
SocksServer=true
```

```
[route]
Name=MQIPT2 to NEWYORK
ListenerPort=1414
Destination=10.9.1.2
DestinationPort=1414
```

6. Starten Sie MQIPT 2.

Öffnen Sie eine Eingabeaufforderung und geben Sie die folgenden Befehle ein:

```
C:
cd \mqipt\bin
mqipt .. -n ipt2
```

Dabei zeigt .. an, dass sich die MQIPT-Konfigurationsdatei mqipt.conf im übergeordneten Verzeichnis befindet und ipt2 der Name ist, der der Instanz von MQIPT zugeordnet wurde.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt2
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqipt\logs will be used to store the log files
MQCPI006 Route 1415 has started and will forward messages to :
MQCPI034 ....10.10.6.7(1414)
MQCPI035 ....using MQ protocol
MQCPI052 ....SOCKS server side enabled
MQCPI078 Route 1415 ready for connection requests
MQCPI006 Route 1414 has started and will forward messages to :
MQCPI034 ....10.9.1.2(1414)
```

```
MQCPI035 ...using MQ protocol
MQCPI078 Route 1414 ready for connection requests
```

7. Geben Sie in einer Eingabeaufforderung im IBM MQ-Client von LONDON (10.7.20.1) die folgenden Befehle ein:

a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=MQIPT.CONN.CHANNEL/tcp/10.7.20.2(1414)
```

b) Reihen Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE LONDON
Hello world
```

Drücken Sie zweimal die Eingabetaste, nachdem Sie die Nachrichtenzeichenfolge eingegeben haben.

Dadurch sendet der Warteschlangenmanager LONDON Nachrichten an die Warteschlange auf dem Warteschlangenmanager NEW YORK.

8. Geben Sie in einer Eingabeaufforderung im IBM MQ-Client für NEW YORK (10.9.1.3) die folgenden Befehle ein:

a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=MQIPT.CONN.CHANNEL/TCP/10.9.1.2(1414)
```

b) Rufen Sie die Nachricht ab:

```
amqsgetc MQIPT.LOCAL.QUEUE NEWYORK
```

Die Nachricht "Hello world" wird zurückgegeben.

Portnummern zuordnen

Sie können steuern, welche lokalen Portadressen bei abgehenden Verbindungen verwendet werden. Wenn Ihre Firewall beispielsweise nur bestimmte Bereiche von Portnummern zulässt, können Sie mithilfe von MQIPT sicherstellen, dass die Ausgabe von einem gültigen Port stammt.

Vorbereitende Schritte

- Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „[Erste Schritte mit IBM MQ Internet Pass-Thru](#)“ auf Seite 175 aufgelisteten vorausgesetzten Tasks abgeschlossen haben.
- Installieren Sie MQIPT auf einem mit mehreren Netzen verbundenen Computer.

Informationen zu diesem Vorgang

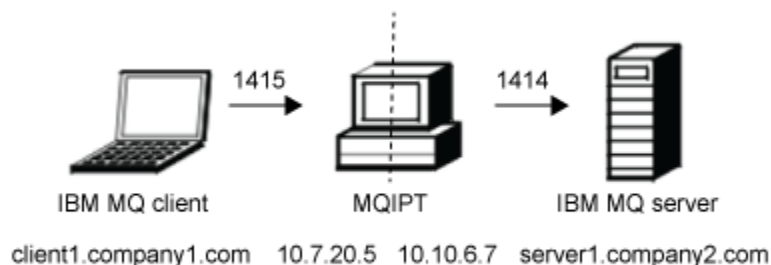


Abbildung 39. Portzuordnungs-Netzdiagramm

Dieses Diagramm zeigt die Verbindung von einem IBM MQ-Client (client1.company1.com an Port 1415) über MQIPT zu einem IBM MQ-Server (server1.company2.com an Port 1414).

Vorgehensweise

Führen Sie die folgenden Schritte aus, um Portnummern zuzuordnen:

1. Konfigurieren Sie MQIPT.

Bearbeiten Sie `mqipt.conf` und fügen Sie die folgende Routendefinition hinzu:

```
[route]
ListenerPort=1415
Destination=server1.company2.com
DestinationPort=1414
LocalAddress=10.10.6.7
OutgoingPort=2000
MaxConnectionThreads=20
```

2. Starten Sie MQIPT.

Öffnen Sie die Eingabeaufforderung auf dem IBM MQ-System und geben Sie folgenden Befehl ein:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt `C:\mqiptHome` die Position der MQIPT-Konfigurationsdatei `mqipt.conf` und `ipt1` den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt1
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1415 is starting and will forward messages to :
MQCPI034 ....server1.company2.com(1414)
MQCPI035 ....using MQ protocol
MQCPI069 ....binding to local address 10.10.6.7 when making new connections
MQCPI070 ....using local port address range 2000-2019 when making new connections
MQCPI078 Route 1415 ready for connection requests
```

3. Geben Sie in einer Eingabeaufforderung auf dem IBM MQ-Clientsystem die folgenden Befehle ein:

a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=MQIPT.CONN.CHANNEL/tcp/10.7.20.5(1415)
```

b) Reihem Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE MQIPT.QM1
Hello world
```

Drücken Sie zweimal die Eingabetaste, nachdem Sie die Nachrichtenzeichenfolge eingegeben haben.

c) Rufen Sie die Nachricht ab:

```
amqsgetc MQIPT.LOCAL.QUEUE MQIPT.QM1
```

Die Nachricht "Hello world" wird zurückgegeben.

CRLs mithilfe eines LDAP-Servers abrufen

Sie können MQIPT für die Verwendung eines LDAP-Servers zum Abrufen von Zertifikatswiderrufslisten (CRLs) konfigurieren.

Vorbereitende Schritte

- Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „Erste Schritte mit IBM MQ Internet Pass-Thru“ auf Seite 175 aufgelisteten vorausgesetzten Tasks abgeschlossen haben.
- Stellen Sie sicher, dass MQIPT 2 über ein persönliches Zertifikat verfügt, das von der anerkannten Zertifizierungsstelle (Certificate Authority, CA) ausgegeben und in einer Schlüsselringdatei mit dem Namen `myCert.pfx` gespeichert wird.
- Stellen Sie sicher, dass MQIPT 1 über eine Kopie des anerkannten CA-Zertifikats verfügt, das für die Authentifizierung des von MQIPT 2 gesendeten Zertifikats verwendet wird. Dieses Zertifikat wird in einer Schlüsselringdatei mit dem Namen `caCerts.pfx` gespeichert.
- Die Kennwörter für den Zugriff auf die Schlüsselringe wurde mit dem Befehl `mqiptPW` verschlüsselt.

Informationen zu diesem Vorgang

In diesem Szenario können Sie den IBM MQ-Client mit einem Warteschlangenmanager verbinden und eine IBM MQ-Nachricht in die Zielwarteschlange stellen. Wenn Sie einen MQIPT-Trace in MQIPT 1 ausführen, wird angezeigt, welcher LDAP-Server verwendet wird.

Um die Funktionsweise von CRLs zu demonstrieren, stellen Sie sicher, dass das von MQIPT 2 verwendete persönliche Zertifikat von der vertrauenswürdigen Zertifizierungsstelle widerrufen wird. Der IBM MQ-Client ist dann nicht berechtigt, eine Verbindung zum Warteschlangenmanager herzustellen, da die Verbindung von MQIPT 1 zu MQIPT 2 abgelehnt wird.

In diesem Szenario geht es nicht darum, zu erläutern, wie ein LDAP-Server installiert und eingerichtet wird oder wie eine Schlüsselringdatei mit persönlichen oder vertrauenswürdigen Zertifikaten erstellt wird. Es wird davon ausgegangen, dass der LDAP-Server von einer bekannten und vertrauenswürdigen Zertifizierungsstelle (CA) verfügbar ist. Es wird kein LDAP-Sicherungsserver verwendet, dieser könnte jedoch durch Hinzufügen der entsprechenden Routeneigenschaften implementiert werden.

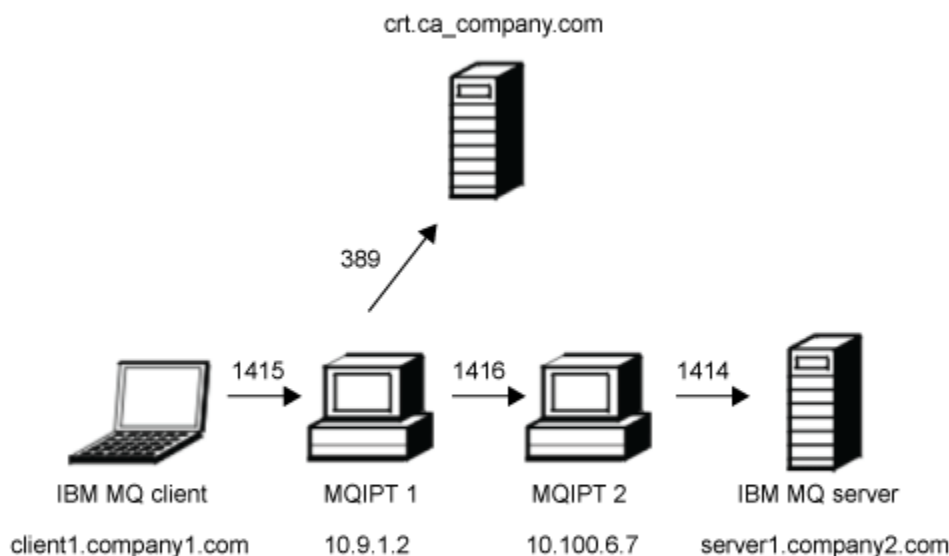


Abbildung 40. LDAP-Server-Netzdiagramm

Dieses Diagramm zeigt die Verbindung vom IBM MQ-Client (client1.company1.com an Port 1415) über zwei Instanzen von MQIPT zum IBM MQ-Server (server1.company2.com an Port 1414). Der erste MQIPT verfügt über eine Verbindung zu einem LDAP-Server (crl.ca_company.com an Port 389).

Vorgehensweise

Führen Sie die folgenden Schritte aus, um CRLs mithilfe eines LDAP-Servers abzurufen:

1. Auf dem System MQIPT 1:

- a) Bearbeiten Sie `mqipt.conf` und fügen Sie die folgende Routendefinition hinzu:

```
[route]
ListenerPort=1415
Destination=10.100.6.7
DestinationPort=1416
SSLClient=true
SSLClientCAKeyRing=C:\mqiptHome\ssl\caCerts.pfx
SSLClientCAKeyRingPW=encrypted_key_ring_password
LDAP=true
LDAPServer1=crl.ca_company.com
LDAPServer1Timeout=4
```

Datei ist *Kennwort_verschlüsselter_Schlüsselring* das Kennwort für den Schlüsselring `ca-Certs.pfx`, der mit dem Befehl **`mqiptPW`** verschlüsselt wurde.

- b) Öffnen Sie eine Eingabeaufforderung und starten Sie MQIPT:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt `C:\mqiptHome` die Position der MQIPT-Konfigurationsdatei `mqipt.conf` und `ipt1` den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt1
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1415 has started and will forward messages to :
MQCPI034 ...10.100.6.7(1416)
MQCPI035 ...using MQ protocol
MQCPI036 ...SSL Client side enabled with properties :
MQCPI031 .....CipherSuites <NULL>
MQCPI032 .....key ring file <NULL>
MQCPI047 .....CA key ring file C:\mqiptHome\ssl\caCerts.pfx
MQCPI071 .....site certificate uses UID=*,CN=*,T=*,OU=*,DC=*,O=*,
                                     STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*
MQCPI038 .....peer certificate uses UID=*,CN=*,T=*,OU=*,DC=*,O=*,
                                     STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*
MQCPI075 ...LDAP main server at crl.ca_company.com(389)
MQCPI086 .....timeout of 4 second(s)
MQCPI084 ...CRL cache expiry timeout is 1 hour(s)
MQCPI085 ...CRLs will be saved in the key-ring file(s)
MQCPI078 Route 1415 ready for connection requests
```

2. Auf dem System MQIPT 2:

- a) Bearbeiten Sie `mqipt.conf` und fügen Sie die folgende Routendefinition hinzu:

```
[route]
ListenerPort=1416
Destination=server1.company2.com
DestinationPort=1414
SSLServer=true
SSLServerKeyRing=C:\mqipt\ssl\myCert.pfx
SSLServerKeyRingPW=encrypted_key_ring_password
```

Datei ist *Kennwort_verschlüsselter_Schlüsselring* das Kennwort für den Schlüsselring `myCert.pfx`, der mit dem Befehl **`mqiptPW`** verschlüsselt wurde.

- b) Öffnen Sie eine Eingabeaufforderung und starten Sie MQIPT:

```
C:
cd \mqipt\bin
mqipt .. -n ipt2
```

Dabei zeigt `..` an, dass sich die MQIPT-Konfigurationsdatei `mqipt.conf` im übergeordneten Verzeichnis befindet und `ipt2` der Name ist, der der Instanz von MQIPT zugeordnet wurde.

Die folgende Nachricht zeigt die erfolgreiche Ausführung an:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt2
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqipt\logs will be used to store the log files
MQCPI006 Route 1416 is starting and will forward messages to :
MQCPI034 ....server1.company2.com(1414)
MQCPI035 ....using MQ protocol
MQCPI037 ....SSL Server side enabled with properties :
MQCPI031 .....CipherSuites <NULL>
MQCPI032 .....key ring file C:\mqipt\ssl\myCert.pfx
MQCPI047 .....CA key ring file <NULL>
MQCPI071 .....site certificate uses UID=*,CN=*,T=*,OU=*,DC=*,O=*,
                    STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*
MQCPI038 .....peer certificate uses UID=*,CN=*,T=*,OU=*,DC=*,O=*,
                    STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*
MQCPI033 .....client authentication set to false
MQCPI078 Route 1416 ready for connection requests
```

3. Geben Sie in einer Eingabeaufforderung auf dem IBM MQ-Clientsystem die folgenden Befehle ein:

a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=MQIPT.CONN.CHANNEL/tcp/10.9.1.2(1415)
```

b) Reihem Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE MQIPT.QM1
Hello world
```

Drücken Sie zweimal die Eingabetaste, nachdem Sie die Nachrichtenzeichenfolge eingegeben haben.

c) Rufen Sie die Nachricht ab:

```
amqsgetc MQIPT.LOCAL.QUEUE MQIPT.QM1
```

Die Nachricht "Hello world" wird zurückgegeben.

MQIPT im TLS-Proxy-Modus ausführen

Sie können MQIPT im TLS-Proxy-Modus ausführen, sodass eine TLS-Verbindungsanforderung von einem IBM MQ-TLS-Client akzeptiert und im Tunnelungsverfahren an einen IBM MQ-TLS-Server übertragen wird.

Vorbereitende Schritte

Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „Erste Schritte mit IBM MQ Internet Pass-Thru“ auf Seite 175 aufgelisteten vorausgesetzten Tasks abgeschlossen haben.

Informationen zu diesem Vorgang

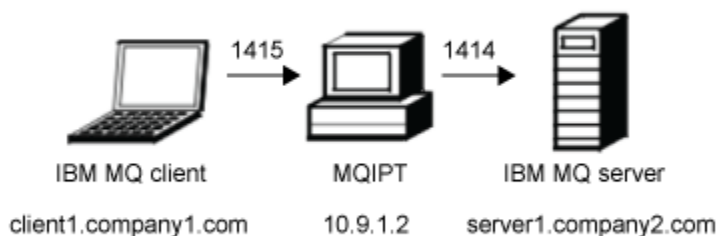


Abbildung 41. SSL/TLS-Proxy-Modus - Netzdiagramm

Dieses Diagramm zeigt den Verbindungsfluss vom IBM MQ-Client (client1.company1.com an Port 1415) über MQIPT zum IBM MQ-Server (server1.company2.com an Port 1414).

Weitere Informationen zur Konfiguration von TLS für IBM MQ finden Sie im Abschnitt [Mit SSL/TLS arbeiten](#).

Vorgehensweise

Führen Sie die folgenden Schritte aus, um MQIPT im TLS-Proxy-Modus auszuführen:

1. Konfigurieren Sie den IBM MQ-Client und -Server für die Verwendung einer TLS-Verbindung.
 - a) Erstellen Sie für den Warteschlangenmanager ein Schlüsselrepository.
Weitere Informationen finden Sie unter [Schlüsselrepository unter AIX, Linux, and Windows einrichten](#).
 - b) Erstellen Sie ein Schlüsselrepository für den Client in dem Verzeichnis C:\ProgramData\IBM\MQ. weisen Sie diesem den Namen *clientkey.kdb* zu.
 - c) Erstellen Sie im Schlüsselrepository des Warteschlangenmanagers, das Sie in Schritt „1.a“ auf Seite 207 erstellt haben, ein persönliches Zertifikat für den Warteschlangenmanager.
Weitere Informationen finden Sie im Abschnitt [Selbst signiertes persönliches Zertifikat unter AIX, Linux, and Windows erstellen](#).
 - d) Erstellen Sie im Schlüsselrepository des Clients, das Sie in Schritt „1.b“ auf Seite 207 erstellt haben, ein persönliches Zertifikat für den Client.
 - e) Rufen Sie das persönliche Zertifikat aus dem Serverschlüsselrepository ab und fügen Sie es dem Client-Repository hinzu.
Weitere Informationen finden Sie im Abschnitt [Öffentlicher Teil eines selbst signierten Zertifikats aus einem Schlüsselrepository unter AIX, Linux, and Windows extrahieren und Zertifikat einer Zertifizierungsstelle \(oder den öffentlichen Teil eines selbst signierten Zertifikats\) in einem Schlüsselrepository auf AIX, Linux, and Windows-Systemen hinzufügen](#).
 - f) Rufen Sie das persönliche Zertifikat aus dem Clientschlüsselrepository ab und fügen Sie es dem Serverschlüsselrepository hinzu.
 - g) Ändern Sie den Serververbindungskanal MQIPT.CONN.CHANNEL mit dem folgenden MQSC-Befehl so, dass TLS verwendet wird:

```
ALTER CHANNEL(MQIPT.CONN.CHANNEL) CHLTYPE(SVRCONN) TRPTYPE(TCP)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256)
```

2. Führen Sie die folgenden Schritte aus, um MQIPT im TLS-Proxy-Modus auszuführen:

- a) Bearbeiten Sie *mqipt.conf* und fügen Sie die folgende Routendefinition hinzu:

```
[route]
ListenerPort=1415
```

```
Destination=server1.company2.com
DestinationPort=1414
SSLProxyMode=true
```

b) Starten Sie MQIPT.

Öffnen Sie eine Eingabeaufforderung und geben Sie folgenden Befehl ein:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt C:\mqiptHome die Position der MQIPT-Konfigurationsdatei mqipt.conf und ipt1 den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt1
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1415 has started and will forward messages to :
MQCPI034 ...server1.company2.com(1414)
MQCPI035 ...using SSLProxyMode protocol
MQCPI078 Route 1415 ready for connection requests
```

3. Geben Sie bei einer Eingabeaufforderung auf dem IBM MQ-Clientsystem den folgenden Befehl ein, um das TLS-Beispielprogramm auszuführen:

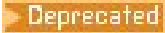
```
AMQSSSLC -m MQIPT.QM1 -c MQIPT.CONN.CHANNEL -x 10.9.1.2(1415)
          -k "C:\ProgramData\IBM\MQ\clientkey" -l cert_label -s
          TLS_RSA_WITH_AES_128_CBC_SHA256
```

cert_label steht dabei für die Bezeichnung des Clientzertifikats, das Sie in Schritt „1.d“ auf Seite 207 erstellt haben.

MQIPT im TLS-Proxy-Modus mit Sicherheitsmanager ausführen

Sie können MQIPT im TLS-Proxy-Modus ausführen, sodass eine TLS-Verbindungsanforderung von einem IBM MQ-TLS-Client akzeptiert und im Tunnelungsverfahren an einen IBM MQ-TLS-Server übertragen wird. Bei Verwendung eines Sicherheitsmanagers mit MQIPT können Sie die Adressen beschränken, an die Nachrichten gesendet werden können.

Vorbereitende Schritte

Anmerkung:  Die Verwendung von Java security manager mit MQIPT wird nicht weiter unterstützt, da Java security manager in einem zukünftigen Release von Javanicht mehr verwendet wird.

Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „[Erste Schritte mit IBM MQ Internet Pass-Thru](#)“ auf Seite 175 aufgelisteten vorausgesetzten Tasks abgeschlossen haben.

Informationen zu diesem Vorgang

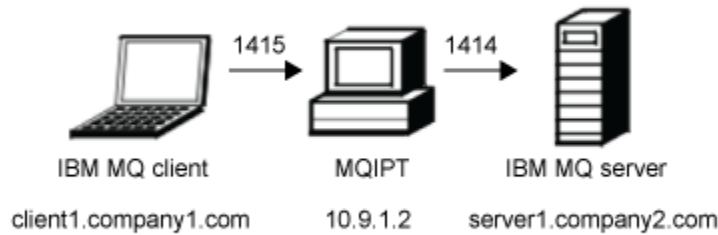


Abbildung 42. SSL/TLS-Proxy-Modus - Netzdiagramm

Dieses Diagramm zeigt den Verbindungsfluss vom IBM MQ-Client (client1.company1.com an Port 1415) über MQIPT zum IBM MQ-Server (server1.company2.com an Port 1414).

Weitere Informationen zur Konfiguration von TLS für IBM MQ finden Sie im Abschnitt [Mit SSL/TLS arbeiten](#).

Vorgehensweise

Führen Sie die folgenden Schritte aus, um MQIPT im TLS-Proxy-Modus mit Sicherheitsmanager auszuführen:

1. Konfigurieren Sie den IBM MQ-Client und -Server für die Verwendung einer TLS-Verbindung.
 - a) Erstellen Sie für den Warteschlangenmanager ein Schlüsselrepository.
Weitere Informationen finden Sie unter [Schlüsselrepository unter AIX, Linux, and Windows einrichten](#).
 - b) Erstellen Sie ein Schlüsselrepository für den Client in dem Verzeichnis C:\ProgramData\IBM\MQ. weisen Sie diesem den Namen *clientkey.kdb* zu.
 - c) Erstellen Sie im Schlüsselrepository des Warteschlangenmanagers, das Sie in Schritt „1.a“ auf Seite 209 erstellt haben, ein persönliches Zertifikat für den Warteschlangenmanager.
Weitere Informationen finden Sie im Abschnitt [Selbst signiertes persönliches Zertifikat unter AIX, Linux, and Windows erstellen](#).
 - d) Erstellen Sie im Schlüsselrepository des Clients, das Sie in Schritt „1.b“ auf Seite 209 erstellt haben, ein persönliches Zertifikat für den Client.
 - e) Rufen Sie das persönliche Zertifikat aus dem Serverschlüsselrepository ab und fügen Sie es dem Client-Repository hinzu.
Weitere Informationen finden Sie im Abschnitt [Öffentlicher Teil eines selbst signierten Zertifikats aus einem Schlüsselrepository unter AIX, Linux, and Windows extrahieren und Zertifikat einer Zertifizierungsstelle \(oder den öffentlichen Teil eines selbst signierten Zertifikats\) in einem Schlüsselrepository auf AIX, Linux, and Windows-Systemen hinzufügen](#).
 - f) Rufen Sie das persönliche Zertifikat aus dem Clientschlüsselrepository ab und fügen Sie es dem Serverschlüsselrepository hinzu.
 - g) Ändern Sie den Serververbindungskanal MQIPT.CONN.CHANNEL mit dem folgenden MQSC-Befehl so, dass TLS verwendet wird:

```
ALTER CHANNEL(MQIPT.CONN.CHANNEL) CHLTYPE(SVRCONN) TRPTYPE(TCP)
SSLICIPH(TLS_RSA_WITH_AES_128_CBC_SHA256)
```

2. Kopieren Sie auf dem MQIPT-Computer (siehe Diagramm) die Java security manager-Beispielrichtlinie in das Ausgangsverzeichnis von MQIPT und geben Sie dazu bei einer Eingabeaufforderung den folgenden Befehl ein:

```
copy C:\mqipt\samples\mqiptSample.policy C:\mqiptHome\mqipt.policy
```

3. Starten Sie das Dienstprogramm "Policy Tool" mit dem folgenden Befehl:

```
C:\mqipt\java\jre\bin\policytool
```

Führen Sie im Richtlinienwerkzeug folgende Schritte aus:

- a) Klicken Sie auf **Datei > Öffnen** und wählen Sie C:\mqiptHome\mqipt.policy aus
- b) Wählen Sie Folgendes aus:

```
file:/C:/Program Files/IBM/IBM MQ Internet Pass-Thru/lib/com.ibm.mq.ipt.jar
```

und klicken Sie anschließend auf **Edit Policy Entry** (Richtlinieneintrag bearbeiten).

- c) Ändern Sie CodeBase von:

```
file:/C:/Program Files/IBM/IBM MQ Internet Pass-Thru/lib/com.ibm.mq.ipt.jar
```

in:

```
file:/C:/mqipt/lib/com.ibm.mq.ipt.jar
```

- d) Ändern Sie die Dateiberechtigungen für die Verzeichnisse IBM MQ Internet Pass-Thru, errors und logs von:

```
C:\Program Files\IBM\IBM MQ Internet Pass-Thru
```

in:

```
C:\mqiptHome
```

- e) Ändern Sie die anderen Dateiberechtigungen von:

```
C:\Program Files\IBM\IBM MQ Internet Pass-Thru
```

in:

```
C:\mqipt
```

- f) Klicken Sie auf **Add Permission** (Berechtigung hinzufügen).

Füllen Sie die Felder wie folgt aus:

Permission (Berechtigung): java.net.SocketPermission

Target (Ziel): client1.company1.com:1024-

Actions (Aktionen): accept, listen, resolve

- g) Klicken Sie auf **File > Save** (Datei > Speichern), um die Änderungen an der Richtliniendatei zu speichern.

4. Bearbeiten Sie mqipt.conf. Fügen Sie im Abschnitt [global] die folgenden Eigenschaften sowie die nachfolgende Routendefinition hinzu:

```
[global]
SecurityManager=true
SecurityManagerPolicy=C:\mqiptHome\mqipt.policy

[route]
ListenerPort=1415
Destination=server1.company2.com
DestinationPort=1414
SSLProxyMode=true
```

5. Starten Sie MQIPT.

Öffnen Sie eine Eingabeaufforderung und geben Sie den folgenden Befehl ein:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt C:\mqiptHome die Position der MQIPT-Konfigurationsdatei mqipt.conf und ipt1 den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt1
MQCPI055 Setting the java.security.policy to C:\mqiptHome\mqipt.policy
MQCPI053 Starting the Java Security Manager
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\mqipt\logs will be used to store the log files
MQCPI006 Route 1415 has started and will forward messages to :
MQCPI034 ...server1.company2.com(1414)
MQCPI035 ...using SSLProxyMode protocol
MQCPI078 Route 1415 ready for connection requests
```

6. Geben Sie bei einer Eingabeaufforderung auf dem IBM MQ-Clientsystem den folgenden Befehl ein, um das TLS-Beispielprogramm auszuführen:

```
AMQSSSLC -m MQIPT.QM1 -c MQIPT.CONN.CHANNEL -x 10.9.1.2(1415)
          -k "C:\ProgramData\IBM\MQ\clientkey" -l cert_label -s
          TLS_RSA_WITH_AES_128_CBC_SHA256
```

cert_label steht dabei für die Bezeichnung des Clientzertifikats, das Sie in Schritt „1.d“ auf Seite 209 erstellt haben.

Sicherheitsexit verwenden

In diesem Szenario können Sie einen mitgelieferten Beispielsicherheitsexit namens 'SampleSecurityExit' verwenden, sodass nur Clientverbindungen, die einen mit den Zeichen MQIPT . beginnenden Kanalnamen verwenden, zulässig sind.

Vorbereitende Schritte

- Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „[Erste Schritte mit IBM MQ Internet Pass-Thru](#)“ auf Seite 175 aufgelisteten vorausgesetzten Tasks abgeschlossen haben.
- Installieren Sie das JavaJava 8.0 JDK.
- Fügen Sie das Unterverzeichnis Java bin zur Umgebungsvariablen **PATH** hinzu.

Informationen zu diesem Vorgang

Der Beispielexit, der in diesem Szenario verwendet wird, ist SampleSecurityExit.java. Sie wird mit MQIPT im Unterverzeichnis samples/exits des MQIPT -Installationsverzeichnis bereitgestellt.

Wenn Sie den vorgeschlagenen Namen für den Serververbindungskanal, MQIPT.CONN.CHANNEL, verwenden (der in den meisten dieser Szenarios verwendet wird), kann die Clientverbindung abgeschlossen werden und eine IBM MQ-Nachricht kann in die Warteschlange eingereicht werden.

Um zu demonstrieren, dass der Sicherheitsexit wie erwartet funktioniert, definieren Sie einen anderen Serververbindungskanal mit einem beliebigen Namen, der nicht mit den Zeichen MQIPT . beginnt (z. B. TEST.CONN.CHANNEL), und wiederholen Sie den Befehl **amqspu**tc , nachdem Sie die Umgebungsvariable **MQSERVER** geändert haben, um den neuen Kanalnamen zu verwenden. In diesem Fall wird die Verbindung zurückgewiesen und der Fehler 2059 (MQRC_Q_MGR_NOT_AVAILABLE) zurückgegeben.

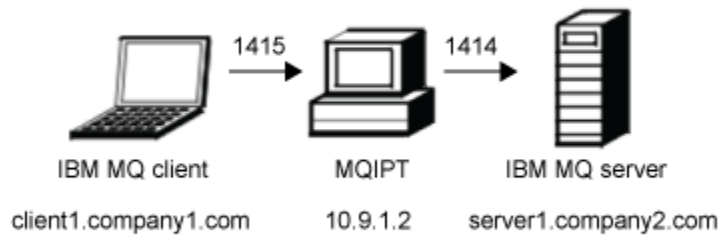


Abbildung 43. Netzdiagramm des Sicherheitsexits

In diesem Diagramm ist der Verbindungsfluss vom IBM MQ-Client (client1.company1.com an Port 1415) über MQIPT zum IBM MQ-Server (server1.company2.com an Port 1414) dargestellt.

Vorgehensweise

Führen Sie für die Verwendung eines Sicherheitsexits die folgenden Schritte aus:

1. Auf dem MQIPT-Computer:

- a) Erstellen Sie ein Verzeichnis mit dem Namen `exits` im Ausgangsverzeichnis von MQIPT, indem Sie den folgenden Befehl in einer Eingabeaufforderung eingeben:

```
md C:\mqiptHome\exits
```

- b) Geben Sie die folgenden Befehle ein, um den Exit zu kompilieren. Dieser Schritt ist nicht notwendig, wenn Sie den Exitcode nicht geändert haben, da der kompilierte Beispiexit mit MQIPT bereitgestellt wird.

```
C:
cd \mqipt\samples\exits
javac -classpath C:\mqipt\lib\com.ibm.mq.ipt.jar;. SampleSecurityExit.java
```

- c) Geben Sie den folgenden Befehl ein, um die kompilierte Exitklassendatei `SampleSecurityExit.class` in das Verzeichnis `C:\mqiptHome\exits` zu kopieren:

```
copy C:\mqipt\samples\exits\SampleSecurityExit.class C:\mqiptHome\exits
```

- d) Bearbeiten Sie `mqipt.conf` und fügen Sie eine Routendefinition hinzu:

```
[route]
ListenerPort=1415
Destination=server1.company2.com
DestinationPort=1414
SecurityExit=true
SecurityExitName=SampleSecurityExit
```

- e) Öffnen Sie eine Eingabeaufforderung und starten Sie MQIPT:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt `C:\mqiptHome` die Position der MQIPT-Konfigurationsdatei `mqipt.conf` und `ipt1` den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt1
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
```

```
MQCPI006 Route 1415 has started and will forward messages to :
MQCPI034 ....server1.company2.com(1414)
MQCPI035 ...using MQ protocol
MQCPI079 ...using security exit C:\mqiptHome\exits\SampleSecurityExit
MQCPI080 .....and timeout of 30 seconds
MQCPI078 Route 1415 ready for connection requests
```

2. Geben Sie in einer Eingabeaufforderung auf dem IBM MQ-Clientensystem die folgenden Befehle ein:

a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=MQIPT.CONN.CHANNEL/tcp/10.9.1.2(1415)
```

b) Reihen Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE MQIPT.QM1
Hello world
```

Drücken Sie zweimal die Eingabetaste, nachdem Sie die Nachrichtenzeichenfolge eingegeben haben.

c) Rufen Sie die Nachricht ab:

```
amqsgetc MQIPT.LOCAL.QUEUE MQIPT.QM1
```

Die Nachricht "Hello world" wird zurückgegeben.

Clientverbindungsanforderungen mithilfe von Sicherheitsexits an IBM MQ-Server für Warteschlangenmanager weiterleiten

In diesem Szenario können Sie Clientverbindungsanforderungen dynamisch im Umlaufverfahren an eine Gruppe von drei IBM MQ-Servern für Warteschlangenmanager weiterleiten. Der Warteschlangenmanager muss auf jedem Server in der Gruppe identisch sein.

Vorbereitende Schritte

- Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „[Erste Schritte mit IBM MQ Internet Pass-Thru](#)“ auf Seite 175 aufgelisteten vorausgesetzten Tasks abgeschlossen haben.
- Installieren Sie das Java 8.0 JDK.
- Fügen Sie das Unterverzeichnis Java bin zur Umgebungsvariablen **PATH** hinzu.

Informationen zu diesem Vorgang

Der Beispiexit, der in diesem Szenario verwendet wird, ist `SampleRoutingExit.java`. Sie wird mit MQIPT im Unterverzeichnis `samples/exits` des MQIPT -Installationsverzeichnisses bereitgestellt.

Der Name und die Position der kompilierten Exitklassendatei werden mit den Eigenschaften **MQIPTSecurityExitName** und **MQIPTSecurityExitPath** definiert.

Die Liste der zu verwendenden Warteschlangenmanager-und Servernamen wird aus einer Konfigurationsdatei mit dem Namen `SampleRoutingExit.conf` gelesen. Der Exit erwartet, dass sich die Konfigurationsdatei in demselben Verzeichnis wie die Exitklassendatei befindet.

Bei der ersten Ausführung des Befehls **amqsputc** wird die IBM MQ-Nachricht in die Warteschlange MQIPT.LOCAL.QUEUE auf dem ersten Server gestellt. Bei der zweiten Ausführung des Befehls wird die Nachricht in die Warteschlange auf dem zweiten Server gestellt usw. Bei dieser Konfiguration kann der Befehl **amqsgetc** nicht die Nachricht abrufen, die gerade in die Warteschlange gestellt wurde, da die vom Befehl **amqsgetc** verwendete Clientverbindungsanforderung an die nächste Warteschlange in der Liste übergeben wird. Durch die dreimalige Ausführung des Befehls **amqsputc** gefolgt von drei **amqsgetc**-Befehlen, wird jedoch sichergestellt, dass jede Nachricht in der gleichen Reihenfolge abgerufen wird.

Wenn Sie einen anderen IBM MQ-Client verwenden, mit dem eine direkte Verbindung zu einem Warteschlangenmanager hergestellt wird (d. h., Sie verwenden MQIPT in diesem Beispiel nicht), können Sie Nachrichten gezielt von einem beliebigen Warteschlangenmanager abrufen.

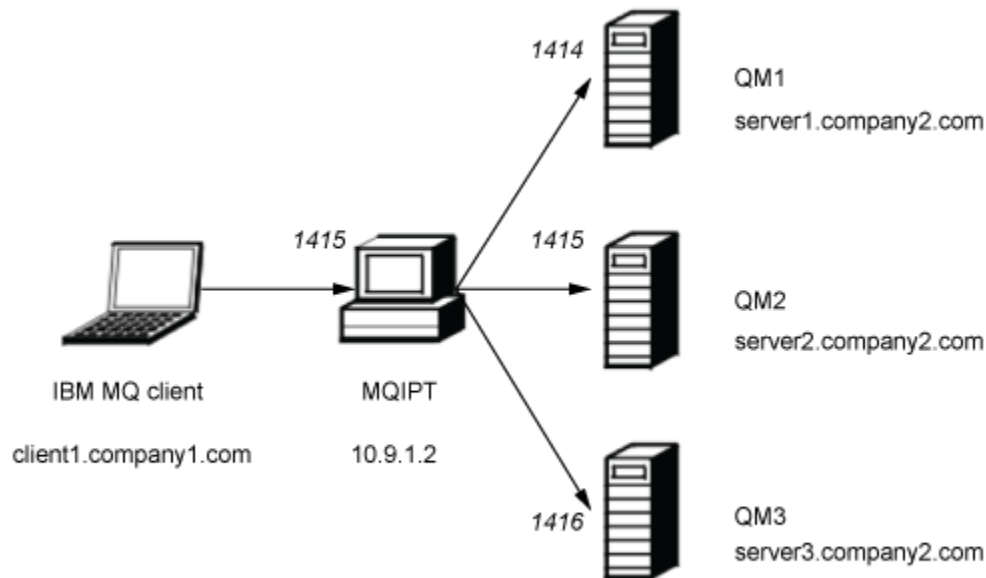


Abbildung 44. Netzdiagramm zur Weiterleitung eines Sicherheitsexits

In diesem Diagramm ist der Verbindungsfluss vom IBM MQ-Client (client1.company1.com an Port 1415) über MQIPT zu drei IBM MQ-Servern (server1.company2.com, server2.company2.com und server3.company2.com) zu sehen.

Vorgehensweise

Führen Sie die folgenden Schritte aus, um Clientverbindungsanforderungen nacheinander an drei unterschiedliche IBM MQ-Server für die Warteschlangenmanager mithilfe von Sicherheitsexits weiterzuleiten:

1. Erstellen Sie drei identische Warteschlangenmanager mit dem Namen MQIPT.QM1 auf drei separaten Servern.

Jeder Warteschlangenmanager verfügt über einen SVRCONN-Kanal mit dem Namen MQIPT.CONN.CHANNEL und eine leere lokale Warteschlange mit dem Namen MQIPT.LOCAL.QUEUE.

2. Auf dem MQIPT-Server:

- a) Erstellen Sie ein Verzeichnis mit dem Namen exits im Ausgangsverzeichnis von MQIPT, indem Sie den folgenden Befehl in einer Eingabeaufforderung eingeben:

```
md C:\mqiptHome\exits
```

- b) Erstellen Sie im Verzeichnis C:\mqiptHome\exits (wobei C:\mqiptHome für das Verzeichnis steht, in dem sich die mqipt.conf-Datei befindet) eine Beispielkonfigurationsdatei mit dem Namen SampleRoutingExit.conf, die die Namen Ihrer drei Warteschlangenmanager enthält.

Die Konfigurationsdatei kann beispielsweise die folgenden Einträge enthalten:

```
server1.company2.com:1414  
server2.company2.com:1415  
server3.company2.com:1416
```

Stellen Sie sicher, dass vor dem ersten Eintrag in der Datei keine leeren Zeilen sind und dass es sich bei jedem Eintrag um einen gültigen Servernamen handelt. Wenn Sie unterschiedliche Servernamen verwendet haben, ändern Sie diese Namen, damit sie mit Ihrer Umgebung übereinstimmen.

- c) Öffnen Sie eine Eingabeaufforderung und geben Sie die folgenden Befehle ein, um den Exit zu kompilieren. Dieser Schritt ist nicht notwendig, wenn Sie den Exitcode nicht geändert haben, da der kompilierte Beispiexit mit MQIPT bereitgestellt wird.

```
C:
cd \mqipt\samples\exits
javac -classpath C:\mqipt\lib\com.ibm.mq.ipt.jar;. SampleRoutingExit.java
```

- d) Geben Sie den folgenden Befehl ein, um die kompilierte Exitklassendatei `SampleRoutingExit.class` in das Verzeichnis `C:\mqiptHome\exits` zu kopieren:

```
copy C:\mqipt\samples\exits\SampleRoutingExit.class C:\mqiptHome\exits
```

- e) Bearbeiten Sie `mqipt.conf` und fügen Sie eine Routendefinition hinzu:

```
[route]
ListenerPort=1415
Destination=server1.company2.com
DestinationPort=1414
SecurityExit=true
SecurityExitPath=C:\mqiptHome\exits
SecurityExitName=SampleRoutingExit
```

Wenn Sie `SampleRoutingExit.conf` in das Standardverzeichnis `C:\mqiptHome\exits` stellen, müssen Sie **SecurityExitPath** nicht festlegen.

- f) Starten Sie MQIPT.

Öffnen Sie eine Eingabeaufforderung und geben Sie folgenden Befehl ein:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt `C:\mqiptHome` die Position der MQIPT-Konfigurationsdatei `mqipt.conf` und `ipt1` den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt1
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1415 has started and will forward messages to :
MQCPI034 ...server1.company2.com(1414)
MQCPI035 ...using MQ protocol
MQCPI079 ...using security exit C:\mqiptHome\exits\SampleRoutingExit
MQCPI080 .....and timeout of 30 seconds
MQCPI078 Route 1415 ready for connection requests
```

3. Geben Sie in einer Eingabeaufforderung auf dem IBM MQ-Clientsystem die folgenden Befehle ein:

- a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=MQIPT.CONN.CHANNEL/TCP/10.9.1.2(1415)
```

- b) Reihen Sie drei Nachrichten ein:

```
amqsputc MQIPT.LOCAL.QUEUE MQIPT.QM1
Hello world 1
amqsputc MQIPT.LOCAL.QUEUE MQIPT.QM1
Hello world 2
amqsputc MQIPT.LOCAL.QUEUE MQIPT.QM1
Hello world 3
```

Nachdem Sie die jeweilige Nachrichtenzeichenfolge eingegeben haben, drücken Sie zweimal die Eingabetaste.

c) Rufen Sie die Nachrichten ab:

```
amqsgetc MQIPT.LOCAL.QUEUE MQIPT.QM1
amqsgetc MQIPT.LOCAL.QUEUE MQIPT.QM1
amqsgetc MQIPT.LOCAL.QUEUE MQIPT.QM1
```

Die Nachrichten Hello world 1, Hello world 2 und Hello world 3 werden zurückgegeben.

Clientverbindungsanforderungen dynamisch weiterleiten

In diesem Szenario können Sie Clientverbindungsanforderungen basierend auf dem Namen des verwendeten Kanals dynamisch an einen Zielsever weiterleiten.

Vorbereitende Schritte

- Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „Erste Schritte mit IBM MQ Internet Pass-Thru“ auf Seite 175 aufgelisteten vorausgesetzten Tasks abgeschlossen haben.
- Installieren Sie das Java 8.0 JDK.
- Fügen Sie das Unterverzeichnis Java bin zur Umgebungsvariablen **PATH** hinzu.

Informationen zu diesem Vorgang

Wenn Sie den Namen des Warteschlangenmanagers als ersten Teil des Kanalnamens verwenden, müssen Sie für alle Verbindungsanforderungen nur eine MQIPT-Route verwenden. Um beispielsweise eine Verbindung zu QM1 herzustellen, könnte der Name eines SVRCONN-Kanals QM1.MQIPT.CHANNEL lauten.

Der Beispiexit, der in diesem Szenario verwendet wird, ist `SampleOneRouteExit.java`. Sie wird mit MQIPT im Unterverzeichnis `samples/exits` des MQIPT-Installationsverzeichnis bereitgestellt.

Der Name und die Position der kompilierten Exitklassendatei werden mit den Eigenschaften MQIPT **SecurityExitName** und **SecurityExitPath** definiert.

Die Liste der zu verwendenden Warteschlangenmanager- und Servernamen wird aus einer Konfigurationsdatei mit dem Namen `SampleOneRouteExit.conf` gelesen. Der Exit erwartet, dass sich die Konfigurationsdatei in demselben Verzeichnis wie die Exitklassendatei befindet.

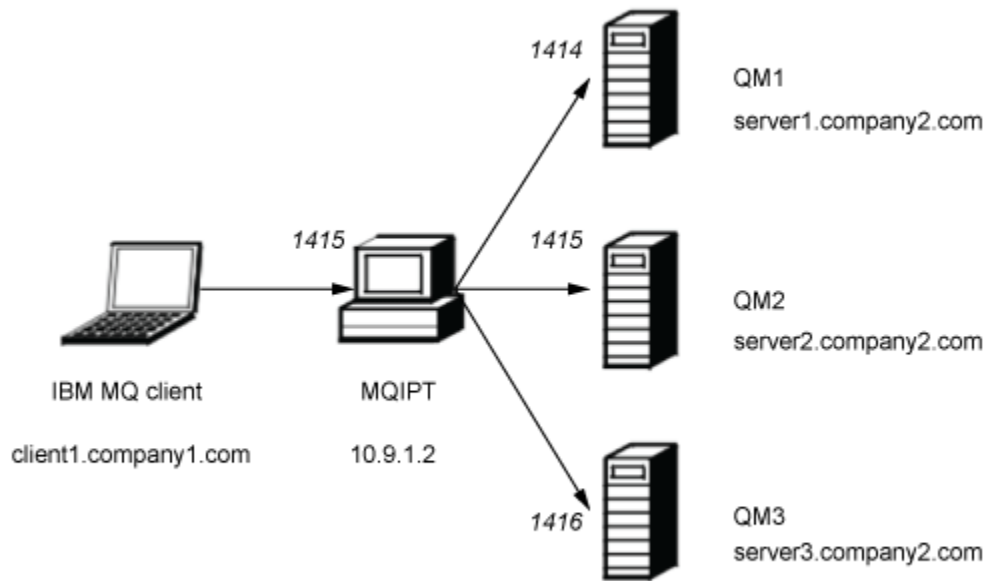


Abbildung 45. Dynamisches Ein-Routen-Exit-Netzdiagramm

In diesem Diagramm ist der Verbindungsfluss vom IBM MQ-Client (client1.company1.com an Port 1415) über MQIPT zu drei IBM MQ-Servern (server1.company2.com, server2.company2.com und server3.com-pany2.com) zu sehen.

Vorgehensweise

Führen Sie für die dynamische Weiterleitung von Clientverbindungsanforderungen die folgenden Schritte aus:

1. Erstellen Sie drei verschiedene Warteschlangenmanager auf drei separaten Servern.
Jeder Warteschlangenmanager verfügt über einen nach ihm selbst benannten SVRCONN-Kanal, z. B. QM1.MQIPT.CHANNEL auf Warteschlangenmanager QM1, und eine leere lokale Warteschlange namens MQIPT.LOCAL.QUEUE.
2. Auf dem MQIPT-Server:
 - a) Erstellen Sie ein Verzeichnis mit dem Namen exits im Ausgangsverzeichnis von MQIPT , indem Sie den folgenden Befehl in einer Eingabeaufforderung eingeben:

```
md C:\mqiptHome\exits
```

- b) Erstellen Sie im Verzeichnis C:\mqiptHome\exits (wobei C:\mqiptHome für das Verzeichnis steht, in dem sich die mqipt.conf -Datei befindet) eine Beispielkonfigurationsdatei mit dem Namen SampleOneRouteExit.conf , die die Namen Ihrer drei Warteschlangenmanager enthält.

Die Konfigurationsdatei kann beispielsweise die folgenden Einträge enthalten:

```
server1.company2.com:1414
server2.company2.com:1415
server3.company2.com:1416
```

Stellen Sie sicher, dass vor dem ersten Eintrag in der Datei keine leeren Zeilen sind und dass es sich bei jedem Eintrag um einen gültigen Servernamen handelt. Wenn Sie unterschiedliche Servernamen verwendet haben, ändern Sie diese Namen, damit sie mit Ihrer Umgebung übereinstimmen.

Beachten Sie, dass alle Warteschlangenmanagernamen in der Liste eindeutig sein müssen. Wenn Sie denselben Namen mehrmals aufführen, wird nur der letzte Eintrag für diesen Namen registriert. Dies gilt auch, wenn sich die Warteschlangenmanager auf verschiedenen Servern befinden.

- c) Öffnen Sie eine Eingabeaufforderung und geben Sie die folgenden Befehle ein, um den Exit zu kompilieren. Dieser Schritt ist nicht notwendig, wenn Sie den Exitcode nicht geändert haben, da der kompilierte Beispiexit mit MQIPT bereitgestellt wird.

```
C:
cd \mqipt\samples\exits
javac -classpath C:\mqipt\lib\com.ibm.mq.ipt.jar;. SampleOneRouteExit.java
```

- d) Geben Sie den folgenden Befehl ein, um die kompilierte Exitklassendatei `SampleOneRouteExit.class` in das Verzeichnis `C:\mqiptHome\exits` zu kopieren:

```
copy C:\mqipt\samples\exits\SampleOneRouteExit.class C:\mqiptHome\exits
```

- e) Bearbeiten Sie `mqipt.conf` und fügen Sie die folgende Routendefinition hinzu:

```
[route]
ListenerPort=1415
Destination=server1.company2.com
DestinationPort=1414
SecurityExit=true
SecurityExitName=SampleOneRouteExit
```

- f) Öffnen Sie eine Eingabeaufforderung und starten Sie MQIPT:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt `C:\mqiptHome` die Position der MQIPT-Konfigurationsdatei `mqipt.conf` und `ipt2` den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt2
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1415 has started and will forward messages to :
MQCPI034 ...server1.company2.com(1414)
MQCPI035 ...using MQ protocol
MQCPI079 ...using security exit C:\mqiptHome\exits\SampleOneRouteExit
MQCPI080 .....and timeout of 5 seconds
MQCPI078 Route 1415 ready for connection requests
```

3. Geben Sie in einer Eingabeaufforderung auf dem IBM MQ-Clientsystem die folgenden Befehle ein:

- a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=QM1.MQIPT.CHANNEL/TCP/10.9.1.2(1415)
```

- b) Reihnen Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE QM1
Hello world 1
```

Drücken Sie zweimal die Eingabetaste, nachdem Sie die Nachrichtenzeichenfolge eingegeben haben.

Die Nachricht wird von MQIPT an QM1 weitergeleitet, da der SVRCONN-Kanalname mit QM1 beginnt.

- c) Rufen Sie die Nachricht von QM1 ab:

```
amqsgetc MQIPT.LOCAL.QUEUE QM1
```

Die Nachricht `Hello world 1` wird zurückgegeben.

d) Legen Sie die Umgebungsvariable **MQSERVER** erneut fest:

```
SET MQSERVER=QM2.MQIPT.CHANNEL/TCP/10.9.1.2(1415)
```

e) Reihen Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE QM2  
Hello world 2
```

Drücken Sie zweimal die Eingabetaste, nachdem Sie die Nachrichtenzeichenfolge eingegeben haben.

Die Nachricht wird von MQIPT an QM2 weitergeleitet, da der SVRCONN-Kanalname mit QM2 beginnt.

f) Rufen Sie die Nachricht von QM2 ab:

```
amqsgetc MQIPT.LOCAL.QUEUE QM2
```

Die Nachricht `Hello world 2` wird zurückgegeben.

g) Legen Sie die Umgebungsvariable **MQSERVER** nochmals fest:

```
SET MQSERVER=QM3.MQIPT.CHANNEL/TCP/10.9.1.2(1415)
```

h) Reihen Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE QM3  
Hello world 3
```

Drücken Sie zweimal die Eingabetaste, nachdem Sie die Nachrichtenzeichenfolge eingegeben haben.

Die Nachricht wird von MQIPT an QM3 weitergeleitet, da der SVRCONN-Kanalname mit QM3 beginnt.

i) Rufen Sie die Nachricht von QM3 ab:

```
amqsgetc MQIPT.LOCAL.QUEUE QM3
```

Die Nachricht `Hello world 3` wird zurückgegeben.

Zertifikatsexit für die Authentifizierung eines TLS-Servers verwenden

In diesem Szenario können Sie eine TLS-Verbindung mithilfe eines Zertifikatsexits authentifizieren.

Vorbereitende Schritte

- Bevor Sie mit der Verwendung dieses Szenarios beginnen, stellen Sie sicher, dass Sie die in „[Erste Schritte mit IBM MQ Internet Pass-Thru](#)“ auf Seite 175 aufgelisteten vorausgesetzten Tasks abgeschlossen haben.
- Installieren Sie das Java 8.0 JDK.
- Fügen Sie das Unterverzeichnis `Java bin` zur Umgebungsvariablen **PATH** hinzu.

Informationen zu diesem Vorgang

In diesem Szenario wird dieselbe Funktion wie in dem Szenario im Abschnitt „[TLS-Server authentifizieren](#)“ auf Seite 182 ausgeführt, allerdings kommt hier ein Zertifikatsexit hinzu.

Der Beispielexit, der in diesem Szenario verwendet wird, ist `SampleCertificateExit.java`. Sie wird mit MQIPT im Unterverzeichnis `samples/exits` des MQIPT-Installationsverzeichnisses bereitgestellt.

Durch Ändern des Werts der Eigenschaft **SSLExitData** kann die TLS-Verbindung zwischen den beiden MQIPT-Servern zugelassen bzw. abgelehnt werden.

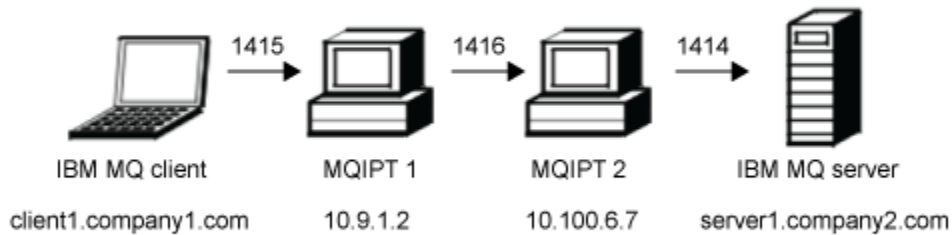


Abbildung 46. SSL/TLS-Server-Netzdiagramm

Dieses Diagramm zeigt die Verbindung vom IBM MQ-Client (`client1.company1.com` an Port 1415) über zwei Instanzen von MQIPT zum IBM MQ-Server (`server1.company2.com` an Port 1414).

Vorgehensweise

Führen Sie die folgenden Schritte aus, um einen TLS-Server unter Verwendung eines Zertifikatsexits zu authentifizieren:

1. Auf dem System MQIPT 1:

- a) Erstellen Sie ein Verzeichnis mit dem Namen `exits` im Ausgangsverzeichnis von MQIPT, indem Sie den folgenden Befehl in einer Eingabeaufforderung eingeben:

```
md C:\mqiptHome\exits
```

- b) Öffnen Sie eine Eingabeaufforderung und geben Sie die folgenden Befehle ein, um den Exit zu kompilieren. Dieser Schritt ist nicht notwendig, wenn Sie den Exitcode nicht geändert haben, da der kompilierte Beispiexit mit MQIPT bereitgestellt wird.

```
C:
cd \mqipt\samples\exits
javac -classpath C:\mqipt\lib\com.ibm.mq.ipt.jar;. SampleCertificateExit.java
```

- c) Geben Sie den folgenden Befehl ein, um die kompilierte Exitklassendatei `SampleCertificateExit.class` in das Verzeichnis `C:\mqiptHome\exits` zu kopieren:

```
copy C:\mqipt\samples\exits\SampleCertificateExit.class C:\mqiptHome\exits
```

- d) Bearbeiten Sie `mqipt.conf` und fügen Sie die folgende Routendefinition hinzu:

```
[route]
ListenerPort=1415
Destination=9.100.6.7
DestinationPort=1416
SSLClient=true
SSLClientKeyRing=C:\mqipt\samples\ssl\sslSample.pfx
SSLClientKeyRingPW=<mqiptPW>1!PCaB1HWzFMOp43ngjwgArg==!6N/vsbqru7iqMhFN+wozxQ==
SSLClientExit=true
SSLExitName=SampleCertificateExit
SSLExitPath=C:\mqiptHome\exits
SSLExitData=allow
```

- e) Öffnen Sie eine Eingabeaufforderung und starten Sie MQIPT:

```
C:\mqipt\bin\mqipt C:\mqiptHome -n ipt1
```

Dabei gibt `C:\mqiptHome` die Position der MQIPT-Konfigurationsdatei `mqipt.conf` und `ipt1` den Namen an, der der Instanz von MQIPT zugeordnet werden soll.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt1
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1415 has started and will forward messages to :
MQCPI034 ...9.100.6.7(1416)
MQCPI035 ...using MQ protocol
MQCPI036 ...SSL Client side enabled with properties :
MQCPI031 .....CipherSuites <null>
MQCPI032 .....keyring file C:\mqipt\samples\ssl\sslSample.pfx
MQCPI047 .....CA keyring file <null>
MQCPI038 .....peer certificate uses UID=*,CN=*,T=*,OU=*,DC=*,O=*,
                                STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*
MQCPI129 .....using certificate exit C:\mqiptHome\exits\SampleCertificateExit
MQCPI131 .....and certificate exit data 'allow'
MQCPI078 Route 1415 ready for connection requests
```

2. Auf dem System MQIPT 2:

a) Bearbeiten Sie mqipt.conf und fügen Sie die folgende Routendefinition hinzu:

```
[route]
ListenerPort=1416
Destination=Server1.company2.com
DestinationPort=1414
SSLServer=true
SSLServerKeyRing=C:\mqipt\samples\ssl\sslSample.pfx
SSLServerKeyRingPW=C:\mqipt\samples\ssl\sslSample.pwd
```

b) Öffnen Sie eine Eingabeaufforderung und starten Sie MQIPT:

```
C:
cd \mqipt\bin
mqipt .. -n ipt2
```

Dabei zeigt .. an, dass sich die MQIPT-Konfigurationsdatei mqipt.conf im übergeordneten Verzeichnis befindet und ipt2 der Name ist, der der Instanz von MQIPT zugeordnet wurde.

Die folgenden Nachrichten geben an, dass MQIPT erfolgreich gestartet wurde:

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.4.0.0 starting
MQCPI004 Reading configuration information from mqipt.conf
MQCPI152 MQIPT name is ipt2
MQCPI021 Password checking has been enabled on the command port
MQCPI011 The path C:\mqipt\logs will be used to store the log files
MQCPI006 Route 1416 has started and will forward messages to :
MQCPI034 ...server1.company2.com(1414)
MQCPI035 ...using MQ protocol
MQCPI037 ...SSL Server side enabled with properties :
MQCPI031 .....CipherSuites <null>
MQCPI032 .....key ring file C:\mqipt\samples\ssl\sslSample.pfx
MQCPI047 .....CA key ring file <null>
MQCPI038 .....peer certificate uses UID=*,CN=*,T=*,OU=*,DC=*,O=*,
                                STREET=*,L=*,ST=*,PC=*,C=*,DNQ=*
MQCPI033 .....client authentication set to false
MQCPI078 Route 1416 ready for connection requests
```

3. Geben Sie in einer Eingabeaufforderung auf dem IBM MQ-Clientsystem die folgenden Befehle ein:

a) Legen Sie die Umgebungsvariable **MQSERVER** fest:

```
SET MQSERVER=MQIPT.CONN.CHANNEL/tcp/10.9.1.2(1415)
```

b) Reihnen Sie eine Nachricht ein:

```
amqsputc MQIPT.LOCAL.QUEUE MQIPT.QM1
Hello world
```

Drücken Sie zweimal die Eingabetaste, nachdem Sie die Nachrichtenzeichenfolge eingegeben haben.

c) Rufen Sie die Nachricht ab:

```
amqsgetc MQIPT.LOCAL.QUEUE MQIPT.QM1
```

Die Nachricht "Hello world" wird zurückgegeben.

MQ Adv.

V 9.4.0

MQ Adv. VUE

Kafka Connect-Szenarios

Mit IBM MQ und Apache Kafka, die sich auf verschiedene Aspekte des Messaging-Spektrums spezialisiert haben (einer auf Konnektivität und der andere auf Daten), erfordern Lösungen häufig den Datenfluss zwischen den beiden. Dies können Sie mit Kafka Connect erreichen.

Kafka Connect stellt ein Framework zum Versetzen von Daten von einem externen System in einen Kafka-Cluster oder von einem Kafka-Cluster in ein externes System bereit. Dies wird durch Steckverbinder erreicht.

Es stehen viele verschiedene Connectortypen zur Verfügung und IBM stellt Connectors für die Verwendung mit IBM MQ bereit. Es gibt zwei verschiedene Connectortypen:

- Quellenconnectors übertragen Daten von einem externen System in Kafka.

Der IBM MQ-Quellenconnector konsumiert Nachrichten aus einer IBM MQ-Warteschlange und veröffentlicht sie als Ereignisse in einem Kafka-Thema.

- Senkenanschlüsse übertragen Daten aus Kafka in ein externes System.

Der IBM MQ-Senkenconnector konsumiert Ereignisse aus einem Kafka-Thema und sendet sie als Nachrichten an eine MQ-Warteschlange.

Weitere Informationen finden Sie unter [Kafka Connect und Connectors](#).

Kafka Connect-Szenarios können Folgendes umfassen:

- Ein zentrales Banksystem mit IBM MQ als Konnektivitätsbackbone. Sie möchten eine Kopie der Nachrichten erstellen, die IBM MQ durchlaufen, und sie zur Analyse mit einer Push-Operation in Kafka übertragen.
- Sie möchten Ihr Kernbankensystem so erweitern, dass Daten in Kafka ausgegeben werden, aber Sie möchten nur, dass Daten in Kafka eingegeben werden, wenn die Banktransaktion erfolgreich abgeschlossen wird. Verwenden Sie daher IBM MQ als transaktionsorientierte Bridge.
- **z/OS** Sie müssen Daten von Multiplatforms in z/OS abrufen. Das Entwicklungsteam für mehrere Plattformen verfügt über Erfahrung mit Kafka, das z/OS-Team möchte die IBM MQ-Integration mit CICS / IMS nutzen.

Ab IBM MQ 9.4.0 erhalten Sie, wenn Ihr Unternehmen über IBM MQ Advanced Berechtigung, IBM MQ Advanced for z/OS VUE, IBM MQ Advanced for Multiplatforms Berechtigung oder IBM MQ Appliance Berechtigung verfügt, Zugriff auf die bereitgestellten IBM und unterstützte Quellen- und Senkenconnectors.

Anmerkungen:

1. Diese Ansätze können mit jeder Variante von Kafka verwendet werden, z. B. Apache Kafka und IBM Event Streams.
2. Unterstützung wird nur für die beiden IBM-Connectors bereitgestellt, nicht für das Kafka Connect-Framework selbst.

MQ Adv.

V 9.4.0

MQ Adv. VUE

Kafka Connect-Allgemeine Topologien

In diesem Abschnitt werden die drei Ansätze beschrieben, die bei der Integration von IBM MQ mit Kafka über die IBM-Connectors verwendet werden können.

Weitere Informationen zum Abrufen der Connectors und „Connectors verwenden“ auf Seite 229 zu den Verbindungs- und Konfigurationsoptionen des Warteschlangenmanagers finden Sie im Abschnitt „Connectors abrufen“ auf Seite 229.

Direkt an Warteschlange (Quelle)

Anwendungen, die Daten über IBM MQ an Kafka senden möchten, können diese Nachrichten an die Warteschlange senden, die vom IBM MQ -Quellenconnector verwendet wird. Der IBM MQ -Quellenconnector überträgt diese Nachrichten dann an das relevante Kafka -Topic.

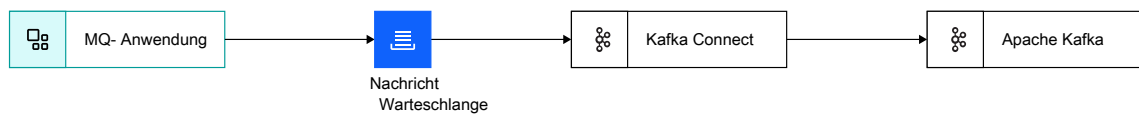


Abbildung 47. Direkt an Warteschlange (Quelle)

Dieser Ansatz sollte verwendet werden, wenn eine Anwendung Daten an Kafka senden muss und diese Daten noch nicht an IBM MQ gesendet wurden.

Das Senden von Daten mit IBM MQ bedeutet, dass das Senden der Nachricht innerhalb einer Transaktion erfolgen kann, die mit anderen Aktualisierungen, z. B. an eine Datenbank, koordiniert wird. Bei diesem Ansatz entfällt auch die Notwendigkeit, eine potenziell kurzlebige Verbindung zu Kafka einzurichten und stattdessen eine vorhandene Verbindung zu IBM MQ zu verwenden.

Streaming-Warteschlangenkopie (Quelle)

In many cases there is a need to take a copy of existing data moving through IBM MQ and sending it into Kafka, for example, for analytics. Ab IBM MQ 9.3 kann dies mithilfe von Streaming-Warteschlangen erreicht werden. Mit Streaming-Warteschlangen können Nachrichten, die in eine Warteschlange eingereicht werden, vom Warteschlangenmanager in eine zweite Warteschlange kopiert werden, ohne dass sich dies auf die Anwendungen auswirkt, die die erste Warteschlange verwenden. Weitere Informationen finden Sie unter [„Streaming-Warteschlangen“ auf Seite 32](#).

For example:

```
DEF QL(TO.APP) STREAMQ(TO.KAFKA) STRMQOS(MUSTDUP)
DEF QL(TO.KAFKA)
```

bedeutet, dass beim Senden einer Nachricht an TO.APP eine Kopie dieser Nachricht an TO.KAFKA gesendet werden muss. Der IBM MQ -Quellenconnector übernimmt dann diese Nachrichten aus TO.KAFKA und überträgt sie an das relevante Kafka -Thema.

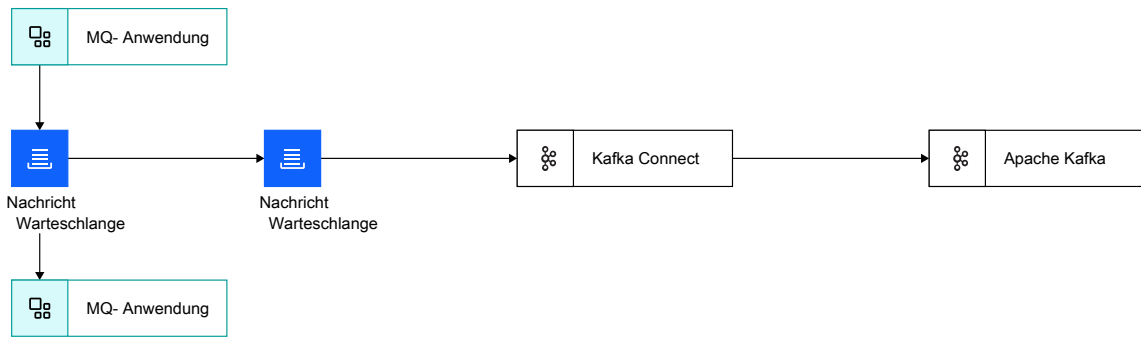


Abbildung 48. Streaming-Warteschlangenkopie (Quelle)

Die Aktivierung von Streaming-Warteschlangen hat keine Auswirkungen auf vorhandene Anwendungen, da sich die ursprüngliche Nachricht nicht ändert. Die an die zweite Warteschlange gesendete Nachricht ist mit der ursprünglichen Nachricht mit denselben Nutzdaten, derselben Nachrichten-ID, derselben Korrelations-ID usw. identisch.

Direkt an Warteschlange (Senke)

Wie beim Quellenconnector kann der Senkenconnector so konfiguriert werden, dass er Daten aus einem Kafka -Topic direkt in eine Warteschlange empfängt.

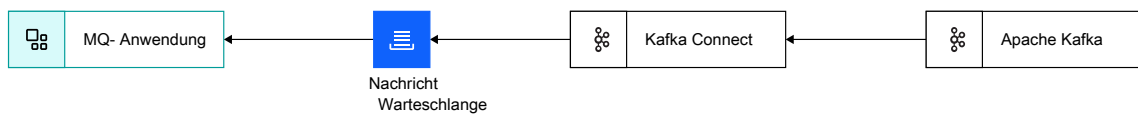


Abbildung 49. Direkt an Warteschlange (Senke)

Receiving data through IBM MQ means that the receive of the message can be done inside an transaction coordinated with other updates, for example to a database.

Bei diesem Ansatz wird auch vermieden, dass eine potenziell kurzlebige Verbindung zu Kafka eingerichtet werden muss und stattdessen eine vorhandene Verbindung zu IBM MQ verwendet werden muss.

Connectors abrufen

V 9.4.0 Die Connectors der Version 2, die im Lieferumfang von IBM MQ 9.4.0 enthalten sind, ermöglichen die Zustellung von Nachrichten mindestens einmal und genau einmal.

V 9.4.0 Weitere Informationen zu den Unterschieden zwischen der Zustellung mindestens einmal und genau einmal sowie zur Konfiguration der Zustellung genau einmal finden Sie unter „Unterstützung genau einmal“ auf Seite 231.

z/OS In IBM MQ Advanced for z/OS Value Unit Edition und IBM MQ Advanced for z/OS werden die Connectors und ihre Beispiele im Verzeichnis kafka-connect der Connector Pack-Komponente in z/OS UNIX System Services (USS) bereitgestellt.

In IBM MQ Advanced for Multiplatforms und IBM MQ Appliance können diese Connectors und erforderlichen Konfigurationsdateien abgerufen werden, indem Sie sich bei Fix Central anmelden und nach V.R.M.F-IBM-MQ-Kafka-Connectors.tar.gz suchen, z. B. 9.4.0.0-IBM-MQ-Kafka-Connectors.tar.gz.

MQ Adv. **MQ Adv. VUE** **MQ Adv. z/OS** Im Lieferumfang der einzelnen IBM MQ -Versionen sind die folgenden Connectors enthalten:

IBM MQ-Versionsnummer	IBM MQ for Multiplatforms -TAR-Dateiname	Version des Quellenconnectors	Sink-Connector-Version	Unterstützung bei der Bereitstellung genau einmal
V 9.4.0 V 9.4.0 9.4.0	9.4.0.0-IBM-MQ-Kafka_Connectors.tar.gz	2.0.0	2.1.0	Ja

Anmerkungen:

1. Sie sollten immer die neueste Version der Connectors verwenden und regelmäßig nach Aktualisierungen suchen. Die mit IBM MQ bereitgestellten Connectors sind die aktuellsten Connectors zu dem Zeitpunkt, zu dem das Produkt ausgeliefert wird, und werden regelmäßig auf die neueste Version aktualisiert.
2. Wenn die Unterstützung für die IBM MQ -Connectors über eine Berechtigung für IBM MQ Advanced for z/OS Value Unit Edition, IBM MQ Advanced for Multiplatforms oder IBM MQ Appliance bereitgestellt wird, müssen die Connectors mit einem Warteschlangenmanager verbunden sein, der mit dieser Berechtigung ausgeführt wird.

Connectors verwenden

Die Connectors werden mithilfe von Eigenschaften oder JSON-Dateien konfiguriert. Beispieldateien werden mit den Connectors bereitgestellt.

Details zu den Konfigurationsoptionen sowie zum Einrichten der Connectors finden Sie unter:

Quellenconnector: [Kafka Connect source connector for IBM MQ](#)

Sink-Connector: [Kafka Connect sink connector for IBM MQ](#)

V 9.4.0 Informationen zum Aktivieren der Unterstützung für genau einmal im Quellenconnector finden Sie unter [MQ -Quellenconnector ausführen](#) und für den Senkenconnector unter [MQ -Senkenconnector ausführen](#).

V 9.4.0 Weitere Informationen zu den Unterschieden zwischen der Zustellung mindestens einmal und genau einmal sowie zur Konfiguration der Zustellung genau einmal finden Sie unter „[Unterstützung genau einmal](#)“ auf Seite 231.

Damit Kafka Connect die IBM MQ Connectors ausführen kann, müssen die Connector-JAR-Dateien und verschiedene IBM MQ -JAR-Dateien im Klassenpfad enthalten sein. Die folgenden JAR-Dateien sind erforderlich:

```
jms.jar
com.ibm.mq.allclient.jar
org.json.jar
V 9.4.0 bcpkix-jdk18on.jar
V 9.4.0 bcprov-jdk18on.jar
V 9.4.0 bcutil-jdk18on.jar
```

For example:

Quellenconnector

```
V 9.4.0 V 9.4.0 :
```

```
export CLASSPATH=$CLASSPATH:/path-to-kafka-jars/kafka-connect-mq-source-2.0.0.jar:
/path-to-mq-jars/jms.jar:/path-to-mq-jars/com.ibm.mq.allclient.jar:/path-to-mq-jars/org.json.jar:
/path-to-mq-jars/bcpkix-jdk18on.jar:/path-to-mq-jars/bcprov-jdk18on.jar:/path-to-mq-jars/bcutil-
jdk18on.jar
```

Zielconnector

```
V 9.4.0 V 9.4.0 :
```

```
export CLASSPATH=$CLASSPATH:/path-to-kafka-jars/kafka-connect-mq-sink-2.1.0.jar:
/path-to-mq-jars/jms.jar:/path-to-mq-jars/com.ibm.mq.allclient.jar:/path-to-mq-jars/org.json.jar:
/path-to-mq-jars/bcpkix-jdk18on.jar:/path-to-mq-jars/bcprov-jdk18on.jar:/path-to-mq-jars/bcutil-
jdk18on.jar
```

Dabei gilt:

path-to-kafka-jars ist der Pfad zu der Position, an der die IBM MQ -Connectors installiert sind.
path-to-mq-jars ist der Pfad zu der Position, an der der IBM JMS -Client installiert ist.

V 9.4.0 **V 9.4.0** Die 2.1.0 -Version des Senkenconnectors bietet die Möglichkeit, Werte zum MQMD hinzuzufügen, bevor Nachrichten an IBM MQ gesendet werden. Weitere Informationen finden Sie unter [MQMD im IBM MQ -Senkenconnector aktivieren v2](#).

z/OS Bei Ausführung unter z/OS verweist USS_ROOT/kafka-connect/source/kafka-connect-mq-source.jar in der Connector-Pack-Komponente auf die neueste Version des Quellenconnectors und USS_ROOT/kafka-connect/sink/kafka-connect-mq-sink.jar auf die neueste Version des Sink-Connectors.

Kafka Connect und die IBM MQ -Connectors können auf jeder Plattform mit einer Java Virtual Machine ausgeführt werden. Sie müssen nicht auf derselben Plattform wie die Warteschlangenmanager oder der Kafka -Cluster ausgeführt werden, zu denen sie eine Verbindung herstellen.

Wenn es jedoch eine große Entfernung zwischen den Warteschlangenmanagern und den Kafka -Clustern gibt, sollten Sie die Connectors relativ nah an den Warteschlangenmanagern positionieren, idealerweise in derselben Verfügbarkeitszone oder in demselben Rechenzentrum.

Connectors unter z/OS verwenden



Die Connectors werden vollständig unterstützt, wenn Warteschlangenmanager auf allen Plattformen ausgeführt werden, auch unter z/OS. Verbindungen zu z/OS -Warteschlangenmanagern können entweder über einen Serververbindungskanal oder über lokale Bindungen erfolgen.

In Leistungstestumgebungen unter IBM z/OS und IBM MQ for z/OS wurde die optimale Leistung erzielt, indem die Connectors unter z/OS in z/OS UNIX System Services (USS) ausgeführt wurden und über lokale Bindungen eine Verbindung zu Warteschlangenmanagern hergestellt wurde. Details zu diesen Ergebnissen sind hier verfügbar: [Kafka Connectors for IBM MQ -eine Perspektive von MQ for z/OS](#).

Die Ausführung von Kafka Connect in USS unter z/OS erfordert einige zusätzliche Konfigurationsschritte. Die Dokumentation zu diesen Schritten finden Sie hier: [Connectors unter IBM z/OS ausführen](#).

MQ Adv. V 9.4.0 MQ Adv. VUE MQ Adv. z/OS Unterstützung genau einmal

Es gibt zwei Versionen der IBM MQ Kafka -Connectors: 1 und 2. Die Connectors der Version 2 unterstützen die Nachrichtenzustellung genau einmal und mindestens einmal, während die Connectors der Version 1 die Nachrichtenzustellung genau einmal unterstützen.

At-least-once-Nachrichtenübermittlung bedeutet, dass im Falle eines Fehlers in IBM MQ, dem IBM MQ Kafka -Connector oder KafkaFolgendes gilt:

- Für den Quellenconnector gehen IBM MQ -Nachrichten nicht verloren, werden jedoch möglicherweise mehrmals an Kafka zugestellt, was zu doppelten Kafka -Nachrichten führt.
- Für den Sink-Connector gehen Kafka -Nachrichten nicht verloren, werden jedoch möglicherweise mehrmals an IBM MQ zugestellt, was zu doppelten IBM MQ -Nachrichten führt.

Eine genau einmalige Nachrichtenübermittlung bedeutet, dass bei einem Fehler in IBM MQ, im IBM MQ Kafka -Connector oder in KafkaFolgendes gilt:

- Für den Quellenconnector gehen IBM MQ -Nachrichten nicht verloren und werden an Kafka zugestellt, ohne dass es zu doppelten Kafka -Nachrichten kommt.
- Für den Sink-Connector gehen Kafka Nachrichten nicht verloren und werden an IBM MQ zugestellt, ohne dass es zu doppelten IBM MQ Nachrichten kommt.

Genau-Einmal-Unterstützung ist nur im Connector der Version 2 verfügbar, der mit IBM MQ oder mit IBM Event Streamsgeliefert wird. Sie ist im Connector der Version 1 nicht verfügbar.

Der Connector der Version 2 kann im Modus "at-least-once" oder "exactly-once" ausgeführt werden. Die Unterstützung von genau einmal wird durch die entsprechende Konfiguration von Kafka und durch die Verwendung einer "Statuswarteschlange" aktiviert. Jede Instanz eines Connectors, die im Modus für genau einmal ausgeführt wird, benötigt eine eigene Statuswarteschlange.

Der Durchsatz und die Skalierbarkeit der Connectors, die im Modus 'Genau einmal' ausgeführt werden, sind geringer als im Modus 'Mindestens einmal'. Aktivieren Sie den Modus "Genau einmal" nur, wenn Ihre Anwendungen nicht für die Arbeit mit doppelten Nachrichten konzipiert sind.

Weitere Informationen zum Konfigurieren des Modus "Genau einmal" im Quellenconnector finden Sie unter [MQ -Quellenconnector ausführen](#) und unter [MQ -Senkenconnector ausführen](#).

Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden.

Möglicherweise bietet IBM die in diesem Dokument beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim zuständigen IBM Ansprechpartner erhältlich. Hinweise auf Produkte, Programme oder Services von IBM bedeuten nicht, dass nur Produkte, Programme oder Services von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Services können auch andere, ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder andere Schutzrechte der IBM verletzen. Die Verantwortung für den Betrieb von Fremdprodukten, Fremdprogrammen und Fremdservices liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

IBM Director of Licensing
IBM Europe, Middle East & Africa
Tour Descartes
2, avenue Gambetta
92066 Paris La Défense
U.S.A.

Bei Lizenzanforderungen zu Double-Byte-Information (DBCS) wenden Sie sich bitte an die IBM Abteilung für geistiges Eigentum in Ihrem Land oder senden Sie Anfragen schriftlich an folgende Adresse:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die hier enthaltenen Informationen werden in regelmäßigen Zeitabständen aktualisiert und als Neuausgabe veröffentlicht. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter werden lediglich als Service für den Kunden bereitgestellt und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängigen, erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

IBM Europe, Middle East & Africa
Software Interoperability Coordinator, Department 49XA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des in diesen Informationen beschriebenen Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung bzw. der Allgemeinen Geschäftsbedingungen von IBM, der IBM Internationalen Nutzungsbedingungen für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer kontrollierten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Gewährleistung, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können davon abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Aussagen über Pläne und Absichten von IBM unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele von IBM.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufes. Sie sollen nur die Funktionen des Lizenzprogramms illustrieren und können Namen von Personen, Firmen, Marken oder Produkten enthalten. Sämtliche dieser Namen sind fiktiv. Ähnlichkeiten mit Namen und Adressen tatsächlicher Unternehmen oder Personen sind zufällig.

COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Beispielanwendungsprogramme, die in Quellsprache geschrieben sind und Programmieretechniken in verschiedenen Betriebsumgebungen veranschaulichen. Sie dürfen diese Beispielprogramme kostenlos ohne Zahlung an IBM in jeder Form kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle für die Betriebsumgebung konform sind, für die diese Beispielprogramme geschrieben sind. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten.

Wird dieses Buch als Softcopy (Book) angezeigt, erscheinen keine Fotografien oder Farbabbildungen.

Informationen zu Programmierschnittstellen

Die bereitgestellten Informationen zur Programmierschnittstelle sollen Sie bei der Erstellung von Anwendungssoftware für dieses Programm unterstützen.

Dieses Handbuch enthält Informationen zu geplanten Programmierschnittstellen, die es dem Kunden ermöglichen, Programme zum Abrufen der Services von IBM MQ zu schreiben.

Diese Informationen können jedoch auch Angaben über Diagnose, Bearbeitung und Optimierung enthalten. Die Informationen zu Diagnose, Bearbeitung und Optimierung sollten Ihnen bei der Fehlerbehebung für die Anwendungssoftware helfen.

Wichtig: Verwenden Sie diese Diagnose-, Änderungs- und Optimierungsinformationen nicht als Programmierschnittstelle, da sie Änderungen unterliegen.

Marken

IBM, das IBM Logo, ibm.com, sind Marken der IBM Corporation in den USA und/oder anderen Ländern. Eine aktuelle Liste der IBM Marken finden Sie auf der Webseite "Copyright and trademark information" www.ibm.com/legal/copytrade.shtml. Weitere Produkt- und Servicennamen können Marken von IBM oder anderen Unternehmen sein.

Microsoft und Windows sind eingetragene Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

UNIX ist eine eingetragene Marke von The Open Group in den USA und anderen Ländern.

Linux ist eine eingetragene Marke von Linus Torvalds in den USA und/oder anderen Ländern.

Dieses Produkt enthält Software, die von Eclipse Project (<https://www.eclipse.org/>) entwickelt wurde.

Java und alle auf Java basierenden Marken und Logos sind Marken oder eingetragene Marken der Oracle Corporation und/oder ihrer verbundenen Unternehmen.



Teilenummer:

(1P) P/N: