

7.5

IBM WebSphere MQ konfigurieren

IBM

Hinweis

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen unter „Bemerkungen“ auf Seite 489 gelesen werden.

Diese Ausgabe bezieht sich auf Version 7 Release 5 von IBM® WebSphere MQ und auf alle nachfolgenden Releases und Modifikationen, bis dieser Hinweis in einer Neuausgabe 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

konfigurieren.....	5
Mehrere Installationen unter UNIX, Linux und Windows konfigurieren.....	5
Anwendungen in einer Umgebung mit mehreren Installationen verbinden.....	6
Primäre Installation ändern.....	16
WS-Manager einer Installation zuordnen.....	18
Installationen von IBM WebSphere MQ auf einem System suchen.....	19
Warteschlangenmanager erstellen und verwalten.....	20
Standardwarteschlangenmanager erstellen.....	23
Vorhandenen WS-Manager als Standardwert erstellen.....	25
Konfigurationsdateien nach der Erstellung eines Warteschlangenmanagers sichern.....	26
WS-Manager starten.....	26
Stoppen eines Warteschlangenmanagers.....	27
WS-Manager erneut starten.....	28
Löschen eines Warteschlangenmanagers.....	28
Anwendungen mithilfe der verteilten Steuerung von Warteschlangen verbinden.....	29
IBM WebSphere MQ -Verfahren für verteiltes Messaging.....	30
Einführung in die verteilte Warteschlangenverwaltung.....	52
Kanäle unter Unix, Linux und Windows überwachen und steuern.....	80
Verbindungen zwischen dem Server und den Clients konfigurieren.....	106
Auswahl des zu verwendenden Kommunikationstyps.....	108
Einen erweiterten transaktionsorientierten Client konfigurieren.....	110
Definieren von MQI-Kanälen.....	120
Serververbindungs- und Clientverbindungsdefinitionen auf verschiedenen Plattformen erstellen.....	122
Serververbindungs- und Clientverbindungsdefinitionen auf dem Server erstellen.....	125
Kanalexits.....	130
Verbindung eines Clients mit einer Gruppe mit gemeinsamer Warteschlange herstellen.....	135
Client mit einer Konfigurationsdatei konfigurieren.....	136
Client mit Umgebungsvariablen konfigurieren.....	156
Eingereihtes Publish/Subscribe steuern.....	164
Publish/Subscribe-Nachrichtenattribute in der Warteschlange festlegen.....	164
In Warteschlange eingereihtes Publish/Subscribe starten.....	165
In Warteschlange eingereihtes Publish/Subscribe stoppen.....	166
Datenstrom hinzufügen.....	167
Datenstrom löschen.....	168
Einen Subskriptionspunkt hinzufügen.....	168
Warteschlangenmanager mit einer Hierarchie verbinden.....	170
Verbindung eines Warteschlangenmanagers zu einer Hierarchie trennen.....	171
WS-Manager-Cluster konfigurieren.....	171
Zugriffsteuerung und mehrere Clusterübertragungswarteschlangen.....	173
Vergleich mit der verteilten Steuerung von Warteschlangen.....	174
Komponenten eines Clusters.....	176
Clusterwarteschlangenmanager für die Aufnahme von vollständigen Repositories auswählen.....	192
Cluster verwalten.....	193
Namenskonventionen für Cluster.....	194
Überlappende Cluster.....	195
Tipps zum Clustering.....	196
Kommunikation in einem Cluster einrichten.....	198
Aufbewahrungsdauer für Repository-Informationen.....	199
IBM WebSphere MQ -Cluster verwalten.....	200
Nachrichten an und von Clustern weiterleiten.....	268
Cluster für Workload-Management verwenden.....	284
Clustering: Bewährte Verfahren.....	301

Verfügbarkeit, Wiederherstellung und Neustart.....	335
Automatische Clientverbindungswiederholung.....	336
Überwachung der Konsolnachricht.....	342
Hochverfügbarkeitskonfigurationen.....	343
Sicherstellen, dass Nachrichten nicht verloren gehen (Protokollierung).....	431
IBM WebSphere MQ -Warteschlangenmanagerdaten sichern und wiederherstellen.....	448
Konfigurationsinformationen ändern.....	454
Konfigurationsinformationen auf UNIX-, Linux-und Windows-Systemen ändern.....	454
Attribute zum Ändern von IBM WebSphere MQ -Konfigurationsdaten.....	460
Konfigurationsdaten des Warteschlangenmanagers ändern.....	467
HP Integrity NonStop Server konfigurieren.....	485
Übersicht über den Gateway-.....	486
Gateway für die Ausführung unter Pathway konfigurieren.....	486
Clientinitialisierungsdatei konfigurieren.....	488
Berechtigungen für Kanäle erteilen.....	488
Bemerkungen.....	489
Informationen zu Programmierschnittstellen.....	490
Marken.....	491

Konfiguration

Erstellen Sie einen oder mehrere Warteschlangenmanager auf einem oder mehreren Computern, und konfigurieren Sie sie auf Ihren Entwicklungs-, Test- und Produktionssystemen, um Nachrichten zu verarbeiten, die Ihre Geschäftsdaten enthalten.

Bevor Sie IBM WebSphere MQ konfigurieren, lesen Sie die Informationen zu den Konzepten von IBM WebSphere MQ in der [IBM WebSphere MQ Technical Overview](#). Informationen zur Planung Ihrer IBM WebSphere MQ -Umgebung finden Sie unter [Planung](#).

Es gibt eine Reihe verschiedener Methoden, die Sie für die Erstellung, Konfiguration und Verwaltung Ihrer Warteschlangenmanager und der zugehörigen Ressourcen in IBM WebSphere MQ verwenden können. Zu diesen Methoden gehören Befehlszeilenschnittstellen, eine grafische Benutzerschnittstelle und eine Verwaltungs-API. Weitere Informationen zu diesen Schnittstellen finden Sie im Abschnitt [Verwaltung von IBM WebSphere MQ](#).

Anweisungen zum Erstellen, Starten, Stoppen und Löschen eines Warteschlangenmanagers finden Sie im Abschnitt [„Warteschlangenmanager erstellen und verwalten“](#) auf Seite 20.

Weitere Informationen zum Erstellen der Komponenten, die für eine gemeinsame Verbindung zu Ihren IBM WebSphere MQ-Installationen und -Anwendungen erforderlich sind, finden Sie im Abschnitt [„Anwendungen mithilfe der verteilten Steuerung von Warteschlangen verbinden“](#) auf Seite 29.

Anweisungen zum Verbinden Ihrer Clients mit einem IBM WebSphere MQ -Server unter Verwendung unterschiedlicher Methoden finden Sie in [„Verbindungen zwischen Client und Server konfigurieren“](#) auf Seite 106.

Anweisungen zum Konfigurieren eines WS-Manager-Clusters finden Sie in [„WS-Manager-Cluster konfigurieren“](#) auf Seite 171.

Sie können das Verhalten von IBM WebSphere MQ oder eines Warteschlangenmanagers ändern, indem Sie Konfigurationsinformationen ändern. Weitere Informationen finden Sie unter [„Konfigurationsinformationen für IBM WebSphere MQ und Warteschlangenmanager ändern“](#) auf Seite 454. Im Allgemeinen müssen Sie einen Warteschlangenmanager nicht erneut starten, damit die Konfigurationsänderungen wirksam werden, mit Ausnahme der in dieser Produktdokumentation genannten Änderungen.

Zugehörige Konzepte

[WebSphere MQ - Technische Übersicht](#)

Zugehörige Tasks

[Lokale WebSphere MQ -Objekte verwalten](#)

[Verwaltung ferner WebSphere MQ-Objekte](#)

[Planung](#)

Mehrere Installationen unter UNIX, Linux, and Windows konfigurieren

Wenn Sie mehrere Installationen auf demselben System verwenden, müssen Sie die -Installationen und -Warteschlangenmanager konfigurieren.

Diese Informationen gelten für UNIX, Linux®, and Windows.

Verwenden Sie die Informationen in den folgenden Links, um Ihre -Installationen zu konfigurieren:

- [„Primäre Installation ändern“](#) auf Seite 16
- [„WS-Manager einer Installation zuordnen“](#) auf Seite 18
- [„Anwendungen in einer Umgebung mit mehreren Installationen verbinden“](#) auf Seite 6

Zugehörige Konzepte

[Mehrere Installationen](#)

Zugehörige Tasks

Primäre Installation auswählen

Auswählen eines Installationsnamens

Anwendungen in einer Umgebung mit mehreren Installationen verbinden

Wenn auf UNIX, Linux, and Windows -Systemen IBM WebSphere MQ Version 7.1 oder höher Bibliotheken geladen werden, verwendet IBM WebSphere MQ automatisch die entsprechenden Bibliotheken, ohne dass Sie weitere Maßnahmen ergreifen müssen. IBM WebSphere MQ verwendet Bibliotheken aus der Installation, die dem Warteschlangenmanager zugeordnet ist, zu dem die Anwendung eine Verbindung herstellt.

Die folgenden Konzepte erläutern die Verbindungsherstellung von Anwendungen zu IBM WebSphere MQ:

Verlinken

Wenn die Anwendung kompiliert wird, wird sie mit den IBM WebSphere MQ-Bibliotheken verbunden, um die Funktionsexporte abzurufen, die dann bei Ausführung der Anwendung geladen werden.

wird geladen

Bei Ausführung der Anwendung werden die IBM WebSphere MQ-Bibliotheken lokalisiert und geladen. Der spezifische Mechanismus, der zum Lokalisieren der Bibliotheken verwendet wird, variiert je nach Betriebssystem und wie die Anwendung erstellt wird. Weitere Informationen zum Suchen und Laden von Bibliotheken in einer Umgebung mit mehreren Installationsumgebungen finden Sie in [„Bibliotheken von IBM WebSphere MQ Version 7.1 oder höher laden“](#) auf Seite 8.

Verbindung wird hergestellt

Wenn die Anwendung beispielsweise mit dem Aufruf MQCONN oder MQCONNX eine Verbindung zu einem aktiven Warteschlangenmanager herstellt, verwendet sie dazu die geladenen IBM WebSphere MQ-Bibliotheken.

Wenn eine Serveranwendung eine Verbindung zu einem WS-Manager herstellt, müssen die geladenen Bibliotheken aus der Installation stammen, die dem Warteschlangenmanager zugeordnet ist. Bei mehreren Installationen auf einem System bringt diese Einschränkung neue Herausforderungen bei der Auswahl des Mechanismus mit sich, mit dem das Betriebssystem die zu ladenden IBM WebSphere MQ-Bibliotheken lokalisiert:

- Wenn der Befehl **setmqm** verwendet wird, um die einem WS-Manager zugeordnete Installation zu ändern, ändern sich die Bibliotheken, die geladen werden müssen.
- Wenn eine Anwendung eine Verbindung zu mehreren WS-Managern herstellt, deren Eigner verschiedene Installationen sind, müssen mehrere Gruppen von Bibliotheken geladen werden.

Werden jedoch Bibliotheken von IBM WebSphere MQ Version 7.1 oder höher gefunden und geladen, lädt und verwendet IBM WebSphere MQ die entsprechenden Bibliotheken, ohne dass weitere Maßnahmen des Benutzers erforderlich sind. Wenn die Anwendung eine Verbindung zu einem WS-Manager herstellt, lädt IBM WebSphere MQ Bibliotheken aus der Installation, die dem Warteschlangenmanager zugeordnet ist.

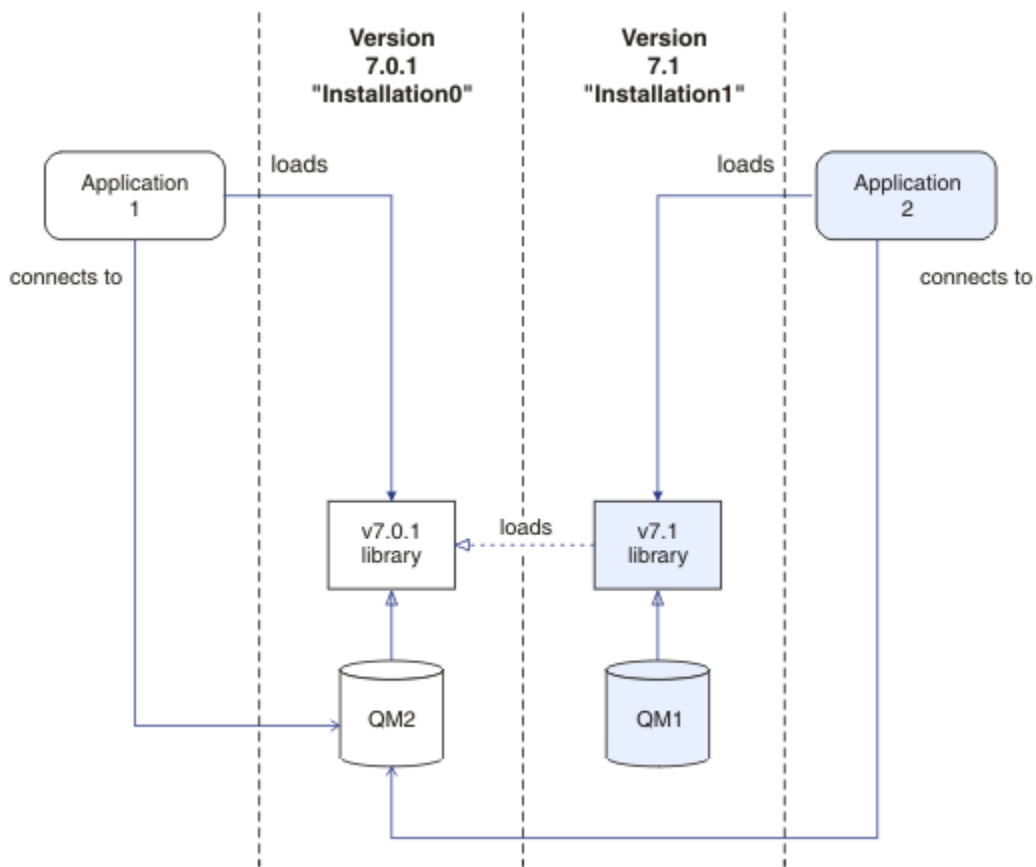


Abbildung 1. Anwendungen in einer Umgebung mit mehreren Installationen verbinden

Abbildung 1 auf Seite 7 zeigt beispielsweise eine Umgebung mit mehreren Installationen mit einer Installation der Version 7.0.1 (Installation0) und einer Installation der Version 7.1 (Installation1). Zwei Anwendungen sind mit diesen Installationen verbunden, aber sie laden unterschiedliche Bibliotheksversionen.

Application 1 lädt eine Bibliothek der Version 7.0.1 direkt. Wenn application 1 eine Verbindung zu QM2herstellt, werden die Bibliotheken der Version 7.0.1 verwendet. Wenn application 1 versucht, eine Verbindung zu QM1herzustellen, bzw. wenn QM2 Installation1zugeordnet ist, schlägt application 1 mit dem Fehler 2059 (080B) (RC2059) fehl: MQRC_Q_MGR_NOT_AVAILABLE Die Anwendung schlägt fehl, da die Bibliothek der Version 7.0.1 keine anderen Bibliotheksversionen laden kann. Wenn also Bibliotheken der Version 7.0.1 direkt geladen werden, können Sie keinen Warteschlangenmanager verwenden, der einer Installation einer höheren Version von IBM WebSphere MQzugeordnet ist.

Application 2 lädt eine Bibliothek der Version 7.1 direkt. Wenn application 2 eine Verbindung zu QM2herstellt, lädt und verwendet die Bibliothek der Version 7.1 die Bibliothek der Version 7.0.1 . Wenn application 2 eine Verbindung zu QM1herstellt oder wenn QM2 Installation1zugeordnet ist, wird die Bibliothek von Version 7.1 geladen und die Anwendung funktioniert wie erwartet.

Migrationsszenarios und das Verbinden von Anwendungen mit mehreren Installationen werden ausführlicher in Koexistenz mehrerer Warteschlangenmanager-Installationen unter UNIX, Linuxund Windowsbehandelt.

Weitere Informationen zum Laden von IBM WebSphere MQ Version 7.1 -Bibliotheken finden Sie in „Bibliotheken von IBM WebSphere MQ Version 7.1 oder höher laden“ auf Seite 8.

Unterstützung und Einschränkungen

Wenn Bibliotheken der folgenden Version 7.1oder höher gefunden und geladen werden, kann IBM WebSphere MQ die entsprechenden Bibliotheken automatisch laden und verwenden:

- Die C-Serverbibliotheken
- Die C++-Serverbibliotheken
- Die XA-Serverbibliotheken
- Die COBOL-Serverbibliotheken
- Die COM+-Serverbibliotheken
- .NET im nicht verwalteten Modus

IBM WebSphere MQ lädt und verwendet außerdem automatisch die entsprechenden Bibliotheken für Java- und JMS-Anwendungen im Bindungsmodus.

Es gibt eine Reihe von Einschränkungen für Anwendungen, die mehrere Installationen verwenden. Weitere Informationen finden Sie im Abschnitt [„Einschränkungen für Anwendungen mit mehreren Installationen“](#) auf Seite 12.

Zugehörige Konzepte

[„WS-Manager einer Installation zuordnen“](#) auf Seite 18

Wenn Sie einen WS-Manager erstellen, wird er automatisch der Installation zugeordnet, die den `crtmqm`-Befehl ausgegeben hat. Unter UNIX, Linux, und Windows können Sie die Installation ändern, die einem Warteschlangenmanager zugeordnet ist, indem Sie den Befehl `setmqm` verwenden.

[„Einschränkungen für Anwendungen mit mehreren Installationen“](#) auf Seite 12

Es gibt Einschränkungen bei der Verwendung von CICS-Serverbibliotheken, Direktaufteilungsverbindungen, Nachrichtenkennungen und Exits in einer Umgebung mit mehreren Installationen.

[„Bibliotheken von IBM WebSphere MQ Version 7.1 oder höher laden“](#) auf Seite 8

Bei der Entscheidung, wie IBM WebSphere MQ-Bibliotheken geladen werden, müssen Sie eine Reihe von Faktoren berücksichtigen, unter anderem Ihre Umgebung, ob Sie Ihre bestehenden Anwendungen ändern können, ob Sie eine primäre Installation möchten, wo IBM WebSphere MQ installiert ist und ob sich der Speicherort von IBM WebSphere MQ möglicherweise ändert.

Zugehörige Tasks

[Primäre Installation auswählen](#)

[„Primäre Installation ändern“](#) auf Seite 16

Mit dem Befehl `setmqinst` können Sie eine Installation als primäre Installation festlegen bzw. ihre Festlegung aufheben.

Bibliotheken von IBM WebSphere MQ Version 7.1 oder höher laden

Bei der Entscheidung, wie IBM WebSphere MQ-Bibliotheken geladen werden, müssen Sie eine Reihe von Faktoren berücksichtigen, unter anderem Ihre Umgebung, ob Sie Ihre bestehenden Anwendungen ändern können, ob Sie eine primäre Installation möchten, wo IBM WebSphere MQ installiert ist und ob sich der Speicherort von IBM WebSphere MQ möglicherweise ändert.

Wie IBM WebSphere MQ Version 7.1-Bibliotheken lokalisiert und geladen werden, hängt von der Installationsumgebung ab:

- Wenn auf UNIX and Linux -Systemen eine Kopie von IBM WebSphere MQ Version 7.1 an der Standardposition installiert ist, funktionieren vorhandene Anwendungen weiterhin auf dieselbe Weise wie frühere Versionen. Wenn die Anwendungen jedoch symbolische Links in `/usr/lib` benötigen, müssen Sie entweder eine Installation der Version 7.1 als primäre Installation auswählen oder die symbolischen Links manuell erstellen.
- Wenn IBM WebSphere MQ Version 7.1 nicht an der Standardposition installiert ist, was der Fall ist, wenn auch IBM WebSphere MQ Version 7.0.1 installiert ist, müssen Sie möglicherweise Ihre vorhandenen Anwendungen ändern, damit die richtigen Bibliotheken geladen werden.

Wie Bibliotheken von IBM WebSphere MQ Version 7.1 oder höher lokalisiert und geladen werden, hängt auch davon ab, wie vorhandene Anwendungen zum Laden von Bibliotheken konfiguriert sind. Weitere Informationen zum Laden von Bibliotheken finden Sie unter [„Lademechanismen für die Betriebssystembibliothek“](#) auf Seite 11.

Optimal sollten Sie sicherstellen, dass der WS-Manager der IBM WebSphere MQ -Bibliothek zugeordnet ist, die vom Betriebssystem geladen wird.

Die Lademethoden für IBM WebSphere MQ-Bibliotheken variieren auch je nach Plattform. Jede Methode hat dabei ihre Vor- und ihre Nachteile.

<i>Tabelle 1. Vorteile und Nachteile der Optionen zum Laden von Bibliotheken</i>			
Plattform	Option	Leistungen	Rückzugsschragen
UNIX and Linux-Systeme	<p>Legen Sie den eingebetteten Laufzeitsuchpfad (RPath) der Anwendung fest oder ändern Sie diesen.</p> <p>Mit dieser Option müssen Sie die Anwendung erneut kompilieren und verknüpfen. Weitere Informationen zum Kompilieren und Verknüpfen von Anwendungen finden Sie im Abschnitt WebSphere MQ -Anwendung erstellen.</p>	<ul style="list-style-type: none"> • Der Umfang der Änderung ist klar. 	<ul style="list-style-type: none"> • Sie müssen in der Lage sein, die Anwendung erneut zu kompilieren und zu verlinken. • Wenn sich die Speicherposition von IBM WebSphere MQ ändert, müssen Sie den Laufzeit-Suchpfad ändern.

Tabelle 1. Vorteile und Nachteile der Optionen zum Laden von Bibliotheken (Forts.)

Plattform	Option	Leistungen	Rückzugsschragen
UNIX and Linux-Systeme	Legen Sie die Umgebungsvariable <code>LD_LIBRARY_PATH</code> (<code>LIBPATH</code> unter AIX) unter Verwendung von <code>setmqenv</code> oder <code>crtmqenv</code> mit der Option <code>-k</code> oder <code>-l</code> fest.	<ul style="list-style-type: none"> • Es sind keine Änderungen an vorhandenen Anwendungen erforderlich. • Überschreibt eingebettete RPaths in einer Anwendung. • Einfache Änderung der Variablen, wenn sich die Position von IBM WebSphere MQ ändert. 	<ul style="list-style-type: none"> • <code>setuid</code> -und <code>setgid</code> -Anwendungen oder Anwendungen, die auf andere Weise erstellt wurden, können <code>LD_LIBRARY_PATH</code> aus Sicherheitsgründen ignorieren. • Umgebungsspezifisch, daher muss in jeder Umgebung, in der die Anwendung ausgeführt wird, festgelegt werden. • Mögliche Auswirkungen auf andere Anwendungen, die auf <code>LD_LIBRARY_PATH</code> basieren. • HP-UX: Optionen, die beim Kompilieren der Anwendungen verwendet wurden, könnten die Verwendung von <code>LD_LIBRARY_PATH</code> inaktivieren. Weitere Informationen finden Sie unter Hinweise zur Laufzeitverknüpfung für HP-UX. • Linux: Der zum Erstellen der Anwendung verwendete Compiler könnte die Verwendung von <code>LD_LIBRARY_PATH</code> inaktivieren. Weitere Informationen finden Sie im Abschnitt Überlegungen zur Laufzeitverknüpfung für Linux.
Windows-Systeme	Legen Sie die Variable <code>PATH</code> mit <code>setmqenv</code> oder <code>crtmqenv</code> fest.	<ul style="list-style-type: none"> • Für vorhandene Anwendungen sind keine Änderungen erforderlich. • Einfache Änderung der Variablen, wenn sich die Position von IBM WebSphere MQ ändert. 	<ul style="list-style-type: none"> • Umgebungsspezifisch, daher muss in jeder Umgebung, in der die Anwendung ausgeführt wird, festgelegt werden. • Mögliche Auswirkungen auf andere Anwendungen.

Tabelle 1. Vorteile und Nachteile der Optionen zum Laden von Bibliotheken (Forts.)

Plattform	Option	Leistungen	Rückzugsschragen
UNIX, Linux, and Windows-Systeme	<p>Setzen Sie die primäre Installation auf eine Installation der Version 7.1 oder höher. Siehe „Primäre Installation ändern“ auf Seite 16.</p> <p>Weitere Informationen zur primären Installation finden Sie unter <u>Primäre Installation auswählen</u>.</p>	<ul style="list-style-type: none"> Für vorhandene Anwendungen sind keine Änderungen erforderlich. Einfache Änderung der primären Installation, wenn sich die Position von IBM WebSphere MQ ändert. Gibt ein ähnliches Verhalten gegenüber früheren Versionen von IBM WebSphere MQ an. 	<ul style="list-style-type: none"> Wenn WebSphere MQ Version 7.0.1 installiert ist, können Sie die primäre Installation nicht auf Version 7.1 oder höher setzen. UNIX and Linux: Funktioniert nicht, wenn <code>/usr/lib</code> nicht der Standardsuchpfad ist.

Hinweise zum Laden von Bibliotheken für HP-UX

Die Beispielkompilierbefehle in der Produktdokumentation für Vorgängerversionen von IBM WebSphere MQ enthalten die Linkoption `-W1`, `+noenvvar` für 64-Bit-Anwendungen. Mit dieser Option wird die Verwendung von `LD_LIBRARY_PATH` zum Laden von gemeinsam genutzten Bibliotheken inaktiviert. Wenn Ihre Anwendungen IBM WebSphere MQ -Bibliotheken von einer anderen Position als der in RPath angegebenen Position laden möchten, müssen Sie Ihre Anwendungen aktualisieren. Sie können die Anwendungen aktualisieren, indem Sie ohne die Linkoption `-W1`, `+noenvvar` erneut kompilieren und verknüpfen oder den Befehl `chatx` verwenden.

Wenn Sie wissen möchten, wie Ihre Anwendungen derzeit Bibliotheken laden, lesen Sie den Abschnitt „Lademechanismen für die Betriebssystembibliothek“ auf Seite 11.

Hinweise zum Laden von Bibliotheken für Linux

Anwendungen, die mit einigen Versionen von gcc, z. B. Version 3.2.x kompiliert wurden, können über einen eingebetteten RPath verfügen, der nicht mit der Umgebungsvariablen `LD_LIBRARY_PATH` überschrieben werden kann. Mit dem Befehl `readelf -d applicationName` können Sie feststellen, ob eine Anwendung betroffen ist. Der RPath kann nicht überschrieben werden, wenn das RPATH-Symbol vorhanden ist und das RUNPATH-Symbol nicht vorhanden ist.

Hinweise zum Laden von Bibliotheken für Solaris

Die Beispielkompilierbefehle in der Produktdokumentation für frühere Versionen von IBM WebSphere MQ enthalten die Link-Optionen für `-lmqmc` `-lmqmzse`. Die entsprechenden Versionen dieser Bibliotheken werden nun automatisch von IBM WebSphere MQ geladen. Wenn IBM WebSphere MQ in einer anderen Position als der Standardposition installiert ist oder wenn mehrere Installationen auf dem System vorhanden sind, müssen Sie Ihre Anwendungen aktualisieren. Sie können die Anwendungen aktualisieren, indem Sie sie erneut kompilieren und verknüpfen, ohne dass die `-lmqmc` `-lmqmzse` -Linkoptionen angegeben sind.

Lademechanismen für die Betriebssystembibliothek

Auf Windows-Systemen werden mehrere Verzeichnisse nach den Bibliotheken durchsucht:

- Das Verzeichnis, aus dem die Anwendung geladen wird.
- Das aktuelle Verzeichnis.
- Die Verzeichnisse in der Umgebungsvariablen `PATH`, sowohl die globale `PATH`-Variable als auch die `PATH`-Variable des aktuellen Benutzers.

Auf UNIX and Linux-Systemen können verschiedene Methoden zur Lokalisierung der zu ladenden Bibliotheken verwendet worden sein:

- Verwenden Sie die Umgebungsvariable `LD_LIBRARY_PATH` (auch `LIBPATH` auf AIX und `SHLIB_PATH` unter HP-UX). Wenn diese Variable festgelegt ist, definiert sie eine Gruppe von Verzeichnissen, die nach den erforderlichen WebSphere MQ -Bibliotheken durchsucht werden. Wenn in diesen Verzeichnissen Bibliotheken gefunden werden, werden sie bevorzugt für alle Bibliotheken verwendet, die mit den anderen Methoden gefunden werden können.
- Verwenden eines eingebetteten Suchpfads (RPath). Die Anwendung kann eine Reihe von Verzeichnissen enthalten, die nach den IBM WebSphere MQ-Bibliotheken durchsucht werden. Wenn der `LD_LIBRARY_PATH` nicht definiert ist oder wenn die erforderlichen Bibliotheken nicht mit der Variablen gefunden wurden, wird der RPath für die Bibliotheken durchsucht. Wenn Ihre vorhandenen Anwendungen einen RPath-Pfad verwenden, aber Sie die Anwendung nicht erneut kompilieren und verlinken können, müssen Sie entweder IBM WebSphere MQ Version 7.1 in der Standardposition installieren oder eine andere Methode zum Suchen der Bibliotheken verwenden.
- Der Standardbibliothekspfad wird verwendet. Wenn die WebSphere MQ -Bibliotheken nach dem Durchsuchen der Variablen `LD_LIBRARY_PATH` und der RPath-Positionen nicht gefunden werden, wird der Standardbibliothekspfad durchsucht. In der Regel enthält dieser Pfad `/usr/lib` oder `/usr/lib64`. Wenn die Bibliotheken nach dem Durchsuchen des Standardbibliothekspfads nicht gefunden werden, kann die Anwendung aufgrund fehlender Abhängigkeiten nicht gestartet werden.

Sie können Betriebssystemmechanismen verwenden, um zu ermitteln, ob Ihre Anwendungen über einen eingebetteten Suchpfad verfügen. Beispiel:

- AIX: **dump**
- HP-UX: **chatr**
- Linux: **readelf**
- Solaris: **elfdump**

Zugehörige Konzepte

[„WS-Manager einer Installation zuordnen“ auf Seite 18](#)

Wenn Sie einen WS-Manager erstellen, wird er automatisch der Installation zugeordnet, die den **cxrtmqm**-Befehl ausgegeben hat. Unter UNIX, Linux, and Windows können Sie die Installation ändern, die einem Warteschlangenmanager zugeordnet ist, indem Sie den Befehl **setmqm** verwenden.

[„Einschränkungen für Anwendungen mit mehreren Installationen“ auf Seite 12](#)

Es gibt Einschränkungen bei der Verwendung von CICS -Serverbibliotheken, Direktaufteilungsverbindungen, Nachrichtenkennungen und Exits in einer Umgebung mit mehreren Installationen.

[„Anwendungen in einer Umgebung mit mehreren Installationen verbinden“ auf Seite 6](#)

Wenn auf UNIX, Linux, and Windows -Systemen IBM WebSphere MQ Version 7.1 oder höher Bibliotheken geladen werden, verwendet IBM WebSphere MQ automatisch die entsprechenden Bibliotheken, ohne dass Sie weitere Maßnahmen ergreifen müssen. IBM WebSphere MQ verwendet Bibliotheken aus der Installation, die dem Warteschlangenmanager zugeordnet ist, zu dem die Anwendung eine Verbindung herstellt.

Zugehörige Tasks

[Primäre Installation auswählen](#)

[„Primäre Installation ändern“ auf Seite 16](#)

Mit dem Befehl **setmqinst** können Sie eine Installation als primäre Installation festlegen bzw. ihre Festlegung aufheben.

Einschränkungen für Anwendungen mit mehreren Installationen

Es gibt Einschränkungen bei der Verwendung von CICS -Serverbibliotheken, Direktaufteilungsverbindungen, Nachrichtenkennungen und Exits in einer Umgebung mit mehreren Installationen.

CICS-Serverbibliotheken

Wenn Sie die CICS-Serverbibliotheken verwenden, wählt IBM WebSphere MQ nicht automatisch die richtige Bibliotheksversion aus. Sie müssen Ihre Anwendungen mit der entsprechenden Bibliotheksebene für den Warteschlangenmanager kompilieren und verknüpfen, zu dem die Anwendung eine Verbindung herstellt. Weitere Informationen finden Sie unter [Bibliotheken zur Verwendung mit TXSeries for Multiplatforms Version 5 erstellen](#).

Nachrichtenkennungen

Nachrichtenkennungen, die den Sonderwert MQHC_UNASSOCIATED_HCONN verwenden, sind begrenzt auf die Verwendung mit der ersten Installation, die in einem Prozess geladen wird. Wenn die Nachrichtenkennung nicht von einer bestimmten Installation verwendet werden kann, wird der Ursachencode MQRC_HMSG_NOT_AVAILABLE zurückgegeben.

Diese Einschränkung betrifft Nachrichteneigenschaften. Sie können keine Nachrichtenkennungen verwenden, um Nachrichteneigenschaften von einem Warteschlangenmanager in einer Installation abzurufen und sie in einen Warteschlangenmanager in einer anderen Installation zu versetzen. Weitere Informationen zu Nachrichtenkennungen finden Sie in [MQCRTMH-Create message handle](#).

Exits

In einer Umgebung mit mehreren Installationsumgebungen müssen vorhandene Exits für die Verwendung mit IBM WebSphere MQ Version 7.1 oder höher aktualisiert werden, um Installationen zu verwenden. Datenkonvertierungsexits, die mit dem Befehl `crtmqcvx` generiert werden, müssen mit dem aktualisierten Befehl neu generiert werden.

Alle Exits müssen unter Verwendung der Struktur MQIEP geschrieben werden, dürfen keinen integrierten Laufzeit-Suchpfad (RPATH) zur Suche nach den IBM WebSphere MQ-Bibliotheken verwenden und können keine Verbindung zu den IBM WebSphere MQ-Bibliotheken herstellen. Weitere Informationen finden Sie im Abschnitt [Exits und installierbare Services schreiben und kompilieren](#).

Direktaufruf

Auf einem Server mit Mehrfachinstallationen müssen Anwendungen, die eine Direktaufrufverbindung mit IBM WebSphere MQ Version 7.1 oder höher verwenden, folgende Regeln befolgen:

1. Der Warteschlangenmanager muss mit derselben Installation verknüpft sein, aus der die Anwendung die Laufzeitbibliotheken von IBM WebSphere MQ geladen hat. Die Anwendung darf keine Direktaufrufverbindung zu einem Warteschlangenmanager verwenden, der mit einer anderen Installation verknüpft ist. Ein Versuch, die Verbindung herzustellen, führt zu einem Fehler mit Ursachencode MQRC_INSTALLATION_MISMATCH.
2. Die Verbindungsherstellung über einen Nicht-Direktaufruf mit einem Warteschlangenmanager, der mit derselben Installation verknüpft ist, aus der die Anwendung die Laufzeitbibliotheken von IBM WebSphere MQ geladen hat, verhindert die Verbindungsherstellung der Anwendung per Direktaufruf – ausgenommen, eine der folgenden Bedingungen wird erfüllt:
 - Die Anwendung stellt ihre erste Verbindung zu einem Warteschlangenmanager, der derselben Installation zugeordnet ist, als Direktaufrufverbindung her.
 - Die Umgebungsvariable AMQ_SINGLE_INSTALLATION ist festgelegt.
3. Die Herstellung einer Verbindung zu einem Warteschlangenmanager einer Version 7.1-Installation oder höher ohne Direktaufruf hat keinerlei Auswirkungen darauf, ob eine Anwendung eine Verbindung per Direktaufruf herstellen kann.
4. Sie können die Verbindungsherstellung zu einem mit einer Version 7.0.1-Installation verknüpften Warteschlangenmanager und die Verbindungsherstellung über Direktaufruf zu einem Warteschlangenmanager, der mit der Installation von Version 7.1 oder höher verknüpft ist, nicht kombinieren.

Mit der Einstellung AMQ_SINGLE_INSTALLATION können Sie jede Verbindung zu einem Warteschlangenmanager als Direktaufrufverbindung herstellen. Anderenfalls gelten nahezu dieselben Einschränkungen:

- Die Installation muss dieselbe sein, von der aus auch die Laufzeitbibliotheken von IBM WebSphere MQ geladen wurden.
- Jede Verbindung im demselben Prozess muss zu derselben Installation hergestellt werden. Wenn Sie versuchen, eine Verbindung zu einem Warteschlangenmanager herzustellen, der einer anderen Installation zugeordnet ist, schlägt die Verbindung mit dem Ursachencode MQRC_INSTALLATION_MISMATCH fehl. Wenn AMQ_SINGLE_INSTALLATION festgelegt ist, gilt diese Einschränkung für alle Verbindungen, nicht nur für Direktaufrufverbindungen.
- Stellen Sie Direktaufrufverbindungen nur zu einem einzigen Warteschlangenmanager her.

Zugehörige Verweise

[MQCONN – Verbindung mit Warteschlangenmanager herstellen \(erweitert\)](#)

[MQIEP-Struktur](#)

[2583 \(0A17\) \(RC2583\): MQRC_INSTALLATION_MISMATCH](#)

[2587 \(0A1B\) \(RC2587\): MQRC_HMSG_NOT_AVAILABLE](#)

[2590 \(0A1E\) \(RC2590\): MQRC_FASTPATH_NOT_AVAILABLE](#)

.NET-Anwendungen in einer Umgebung mit Mehrfachinstallationen verbinden

Standardmäßig verwenden Anwendungen die .NET-Assemblys aus der primären Installation. Wenn keine primäre Installation vorhanden ist oder Sie die primären Installationsassemblys nicht verwenden möchten, müssen Sie die Anwendungskonfigurationsdatei oder die *DEVPATH* -Umgebungsvariable aktualisieren.

Wenn auf dem System eine primäre Installation vorhanden ist, werden die .NET-Assemblys und Richtliniendateien dieser Installation im Global Assembly Cache (GAC) registriert. Die .NET-Assemblys für alle anderen Installationen befinden sich im Installationspfad jeder Installation, aber die Assemblys sind nicht im GAC registriert. Daher werden Anwendungen standardmäßig mit den .NET-Assemblys aus der primären Installation ausgeführt. Sie müssen die Anwendungskonfigurationsdatei aktualisieren, wenn einer der folgenden Fälle zutrifft:

- Sie verfügen nicht über eine primäre Installation.
- Sie möchten, dass die Anwendung die primären Installationsassemblys nicht verwendet.
- Bei der primären Installation handelt es sich um eine niedrigere Version von IBM WebSphere MQ als die Version, mit der die Anwendung kompiliert wurde.

Informationen zum Aktualisieren der Anwendungskonfigurationsdatei finden Sie unter [„.NET-Anwendungen mithilfe der Anwendungskonfigurationsdatei verbinden“](#) auf Seite 14.

Sie müssen die Umgebungsvariable *DEVPATH* aktualisieren, wenn der folgende Fall wahr ist:

- Sie möchten, dass Ihre Anwendung die Assemblys von einer nicht primären Installation aus verwendet, die primäre Installation jedoch dieselbe Version wie die nicht primäre Installation hat.

Weitere Informationen zum Aktualisieren der Variablen *DEVPATH* finden Sie unter [„Verbindung zu .NET-Anwendungen mit DEVPATH herstellen“](#) auf Seite 15.

.NET-Anwendungen mithilfe der Anwendungskonfigurationsdatei verbinden

Innerhalb der Anwendungskonfigurationsdatei müssen Sie verschiedene Tags für die Umleitung von Anwendungen festlegen, um Assemblys verwenden zu können, die nicht von der primären Installation aus verwendet werden.

Die folgende Tabelle enthält die spezifischen Änderungen, die an der Anwendungskonfigurationsdatei vorgenommen werden müssen, damit .NET-Anwendungen eine Verbindung mit bestimmten Assemblys herstellen können:

Tabelle 2. Anwendungen für die Verwendung bestimmter Baugruppen konfigurieren		
	Mit einer niedrigeren Version von IBM WebSphere MQ kompilierte Anwendungen	Mit einer höheren Version von IBM WebSphere MQ kompilierte Anwendungen
Zum Ausführen einer Anwendung mit einer primären IBM WebSphere MQ -Installation einer höheren Version. (Baugruppen der höheren Version in GAC)	Keine Änderungen erforderlich	Keine Änderungen erforderlich
Zum Ausführen einer Anwendung mit einer primären Installation einer niedrigeren Version von IBM WebSphere MQ . (Unterbaugruppen in GAC):	Keine Änderungen erforderlich	In der Anwendungskonfigurationsdatei: <ul style="list-style-type: none"> • Verwenden Sie den Tag <code><bindingRedirect></code> , um die Verwendung der niedrigeren Version der Assemblys anzugeben, die sich im GAC befinden.
Ausführen einer Anwendung mit einer höheren Version der nicht primären IBM WebSphere MQ -Installation. (Assemblys der höheren Version im Installationsordner)	In der Anwendungskonfigurationsdatei: <ul style="list-style-type: none"> • Verwenden Sie den Tag <code><codebase></code> , um auf die Position der Assemblys der höheren Version zu verweisen • Verwenden Sie den Tag <code><bindingRedirect></code> , um die Verwendung der Assemblys der höheren Version anzugeben 	In der Anwendungskonfigurationsdatei: <ul style="list-style-type: none"> • Verwenden Sie den Tag <code><codebase></code> , um auf die Position der Assemblys der höheren Version zu verweisen
Zum Ausführen einer Anwendung mit einer älteren Version der nicht primären IBM WebSphere MQ -Installation. (ältere Version der Assemblys im Installationsordner)	In der Anwendungskonfigurationsdatei: <ul style="list-style-type: none"> • Verwenden Sie den Tag <code><codebase></code> , um auf die Position der Assemblys der niedrigeren Version zu verweisen. • Tag <code><publisherpolicy Apply=no></code> einschließen 	In der Anwendungskonfigurationsdatei: <ul style="list-style-type: none"> • Verwenden Sie den Tag <code><codebase></code> , um auf die Position der Assemblys der niedrigeren Version zu verweisen. • Verwenden Sie den Tag <code><bindingRedirect></code> , um die Verwendung der Assemblys der niedrigeren Version anzugeben. • Tag <code><publisherpolicy Apply=no></code> einschließen

Eine Beispielanwendungskonfigurationsdatei `NonPrimaryRedirect.config` wird im Ordner `MQ_INSTALLATION_PATH\tools\dotnet\samples\base` bereitgestellt. Diese Datei können Sie mit dem IBM WebSphere MQ-Installationspfad einer beliebigen nicht-primären Installation ändern. Die Datei kann mithilfe des Tags `<linkedConfiguration>` auch direkt in andere Konfigurationsdateien eingeschlossen werden. Es werden Muster für `nmqsget.exe.config` und `nmqsput.exe.config` bereitgestellt. In beiden Beispielen wird der Tag `<linkedConfiguration>` verwendet und die Datei `NonPrimaryRedirect.config` eingeschlossen.

Verbindung zu .NET-Anwendungen mit DEVPATH herstellen

Sie können die Assemblys mit der Umgebungsvariablen `DEVPATH` finden. Die durch die Variable `DEVPATH` angegebenen Assemblys werden bevorzugt für alle Baugruppen in der GAC verwendet. Weitere Informati-

onen zur Verwendung dieser Variablen finden Sie in der entsprechenden Microsoft-Dokumentation unter *DEVPATH*.

Um die Assemblys mit der Umgebungsvariablen *DEVPATH* zu suchen, müssen Sie die Variable *DEVPATH* auf den Ordner setzen, der die zu verwendenden Assemblys enthält. Anschließend müssen Sie die Anwendungskonfigurationsdatei aktualisieren und die folgenden Laufzeitkonfigurationsdaten hinzufügen:

```
<configuration>
  <runtime>
    <developmentMode developerInstallation="true" />
  </runtime>
</configuration>
```

Zugehörige Konzepte

„Anwendungen in einer Umgebung mit mehreren Installationen verbinden“ auf Seite 6

Wenn auf UNIX, Linux, and Windows -Systemen IBM WebSphere MQ Version 7.1 oder höher Bibliotheken geladen werden, verwendet IBM WebSphere MQ automatisch die entsprechenden Bibliotheken, ohne dass Sie weitere Maßnahmen ergreifen müssen. IBM WebSphere MQ verwendet Bibliotheken aus der Installation, die dem Warteschlangenmanager zugeordnet ist, zu dem die Anwendung eine Verbindung herstellt.

[Mehrere Installationen](#)

Zugehörige Tasks

[Primäre Installation auswählen](#)

[.NET verwenden](#)

Primäre Installation ändern

Mit dem Befehl **setmqinst** können Sie eine Installation als primäre Installation festlegen bzw. ihre Festlegung aufheben.

Informationen zu diesem Vorgang

Diese Task gilt für UNIX, Linux, and Windows.

Bei der primären Installation handelt es sich um die Installation, auf die systemweite Standorte verweisen. Weitere Informationen zur primären Installation sowie Hinweise zur Auswahl der primären Installation finden Sie im Abschnitt [Primäre Installation auswählen](#).

Wenn eine Installation von IBM WebSphere MQ Version 7.1 oder höher mit einer Installation von IBM WebSphere MQ Version 7.0.1 koexistiert, muss die IBM WebSphere MQ Version 7.0.1 -Installation die primäre Installation sein. Sie wird als primäre Installation markiert, wenn IBM WebSphere MQ Version 7.1 oder eine höhere Version installiert ist und IBM WebSphere MQ Version 7.1 oder höher nicht als primäre Installation ausgewählt werden kann.

Während des Installationsprozesses unter Windows können Sie angeben, dass die Installation die primäre Installation sein soll. Auf UNIX and Linux -Systemen müssen Sie nach der Installation einen **setmqinst** -Befehl absetzen, um die Installation als primäre Installation festzulegen.

„[Primäre Installation festlegen](#)“ auf Seite 16.

„[Deinstallieren der primären Installation](#)“ auf Seite 17.

Primäre Installation festlegen

Vorgehensweise

Gehen Sie wie folgt vor, um eine Installation als primäre Installation festzulegen:

1. Überprüfen Sie, ob eine Installation bereits die primäre Installation ist, indem Sie den folgenden Befehl eingeben:

```
MQ_INSTALLATION_PATH/bin/dspmqinst
```


Dabei ist `MQ_INSTALLATION_PATH` der Installationspfad einer Installation von IBM WebSphere MQ Version 7.1 oder höher.

2. Wenn eine bestehende Installation von IBM WebSphere MQ Version 7.1 oder höher als die primäre Installation festgelegt ist, heben Sie diese Festlegung mithilfe der Anweisungen im Abschnitt „Deinstallieren der primären Installation“ auf Seite 17 auf. Wenn IBM WebSphere MQ Version 7.0.1 auf dem System installiert ist, kann die primäre Installation nicht geändert werden.
3. Geben Sie als Rootbenutzer auf UNIX and Linux-Systemen bzw. als Mitglied der Administratorgruppe auf Windows-Systemen einen der folgenden Befehle ein:

- Gehen Sie wie folgt vor, um die primäre Installation mithilfe des Pfads der Installation festzulegen, die die primäre Installation sein soll:

```
MQ_INSTALLATION_PATH/bin/setmqinst -i -p MQ_INSTALLATION_PATH
```

- Gehen Sie wie folgt vor, um die primäre Installation mit dem Namen der Installation festzulegen, die die primäre Installation sein soll:

```
MQ_INSTALLATION_PATH/bin/setmqinst -i -n installationName
```

4. Starten Sie das System auf Windows -Systemen erneut.

Deinstallieren der primären Installation

Vorgehensweise

Gehen Sie wie folgt vor, um eine Installation als primäre Installation zu dekonfigurieren:

1. Überprüfen Sie, welche Installation die primäre Installation ist, indem Sie den folgenden Befehl eingeben:

```
MQ_INSTALLATION_PATH/bin/dspmqinst
```

Dabei ist `MQ_INSTALLATION_PATH` der Installationspfad einer Installation von IBM WebSphere MQ Version 7.1 oder höher.

Wenn IBM WebSphere MQ Version 7.0.1 die primäre Installation ist, können Sie die primäre Installation nicht aufheben.

2. Geben Sie als Rootbenutzer auf UNIX and Linux-Systemen bzw. als Mitglied der Administratorgruppe auf Windows-Systemen einen der folgenden Befehle ein:

- Gehen Sie wie folgt vor, um die primäre Installation mithilfe des Pfads der Installation zu dekonfigurieren, die Sie nicht mehr als primäre Installation verwenden möchten:

```
MQ_INSTALLATION_PATH/bin/setmqinst -x -p MQ_INSTALLATION_PATH
```

- Gehen Sie wie folgt vor, um die primäre Installation unter Verwendung des Namens der Installation zu dekonfigurieren, die Sie nicht mehr als primäre Installation verwenden möchten:

```
MQ_INSTALLATION_PATH/bin/setmqinst -x -n installationName
```

Zugehörige Konzepte

[Features, die nur mit der primären Installation unter Windows verwendet werden können](#)

[Verknüpfungen von externen Speicherarchiven und Steuerbefehlen zur primären Installation unter UNIX und Linux](#)

Zugehörige Tasks

[Deinstallieren, Durchführen eines Upgrades und Wartung der primären Installation](#)

[Auswählen eines Installationsnamens](#)

Zugehörige Verweise

[setmqinst](#)

WS-Manager einer Installation zuordnen

Wenn Sie einen WS-Manager erstellen, wird er automatisch der Installation zugeordnet, die den **cxrtmqm**-Befehl ausgegeben hat. Unter UNIX, Linux, and Windows können Sie die Installation ändern, die einem Warteschlangenmanager zugeordnet ist, indem Sie den Befehl **setmqm** verwenden.

Den Befehl **setmqm** können Sie wie folgt verwenden:

- Verschieben einzelner WS-Manager zwischen funktional entsprechenden Versionen von WebSphere MQ. Beispiel: Verschieben eines Warteschlangenmanagers von einem Test in ein Produktionssystem.
- Migration einzelner Warteschlangenmanager von einer älteren Version von WebSphere MQ auf eine neuere Version von WebSphere MQ. Die Migration von Warteschlangenmanagern zwischen Versionen hat verschiedene Auswirkungen, die Sie kennen müssen. Weitere Informationen zur Migration finden Sie unter [Migration und Upgrade von WebSphere MQ](#).

So ordnen Sie einen Warteschlangenmanager einer Installation zu:

1. Stoppen Sie den WS-Manager mit dem Befehl **endmqm** aus der Installation, die derzeit dem Warteschlangenmanager zugeordnet ist.
2. Ordnen Sie den Warteschlangenmanager mit dem Befehl **setmqm** aus dieser Installation einer anderen Installation zu.

Geben Sie beispielsweise den folgenden Befehl in Installation2 ein, um den Warteschlangenmanager QMB einer Installation mit dem Namen Installation2 zu setzen:

```
MQ_INSTALLATION_PATH/bin/setmqm -m QMB -n Installation2
```

Dabei steht *MQ_INSTALLATION_PATH* für den Pfad, in dem Installation2 installiert ist.

3. Starten Sie den Warteschlangenmanager mit dem Befehl **strmqm** aus der Installation, die jetzt dem Warteschlangenmanager zugeordnet ist.

Dieser Befehl führt alle erforderlichen Warteschlangenmanager-Migrationen aus und führt dazu, dass der WS-Manager betriebsbereit ist.

Die Installation, der ein Warteschlangenmanager zugeordnet ist, schränkt den Warteschlangenmanager ein, sodass er nur mit Befehlen verwaltet werden kann, die in dieser Installation ausgegeben werden. Es gibt drei wichtige Ausnahmen:

- Der Befehl **setmqm** ändert die dem Warteschlangenmanager zugeordnete Installation. Dieser Befehl muss von der Installation abgesetzt werden, die Sie dem Warteschlangenmanager zuordnen möchten, nicht die Installation, der der WS-Manager derzeit zugeordnet ist. Der mit dem Befehl **setmqm** angegebene Installationsname muss mit der Installation übereinstimmen, von der der Befehl abgesetzt wird.
- Der Befehl **strmqm** muss normalerweise in der Installation ausgegeben werden, die dem Warteschlangenmanager zugeordnet ist. Wenn jedoch ein Warteschlangenmanager der Version 7.0.1 oder einer früheren Version zum ersten Mal in einer Installation der Version 7.1 oder höher gestartet wird, können Sie den Befehl **strmqm** verwenden. In diesem Fall startet **strmqm** den WS-Manager und ordnet ihn der Installation zu, von der aus der Befehl ausgegeben wird.
- Der Befehl **dspmq** zeigt Informationen zu allen Warteschlangenmanagern in einem System an, nicht nur zu den Warteschlangenmanagern, die derselben Installation wie der Befehl **dspmq** zugeordnet sind. Der Befehl `dspmq -o installation` zeigt Informationen dazu an, welche Warteschlangenmanager welchen Installationen zugeordnet sind.

Zuordnung von Warteschlangenmanagern in HA-Umgebungen

Bei HA-Umgebungen ordnet der Befehl **addmqinf** den WS-Manager automatisch der Installation zu, von der der Befehl **addmqinf** ausgegeben wird. Solange der Befehl **strmqm** von derselben Installation wie der **addmqinf**-Befehl ausgegeben wird, ist keine weitere Konfiguration erforderlich. Um den Warteschlangenmanager mit einer anderen Installation zu starten, müssen Sie zuerst die zugehörige Installation mit dem **setmqm**-Befehl ändern.

Gelöschten Installationen zugeordnete Warteschlangenmanager

Wenn die Installation, der ein Warteschlangenmanager zugeordnet ist, gelöscht wurde oder die Statusinformationen des Warteschlangenmanagers nicht verfügbar sind, kann der Befehl **setmqm** den Warteschlangenmanager keiner anderen Installation zuordnen. Nehmen Sie in dieser Situation die folgenden Aktionen vor:

1. Mit dem Befehl **dspmqinst** können Sie die anderen Installationen auf Ihrem System anzeigen.
2. Ändern Sie das `InstallationName` -Feld der Zeilengruppe `QueueManager` in `mqs.ini` manuell, um eine andere Installation anzugeben.
3. Verwenden Sie den Befehl **dlmqm** aus dieser Installation, um den Warteschlangenmanager zu löschen.

Zugehörige Konzepte

„Installationen von IBM WebSphere MQ auf einem System finden“ auf Seite 19

Wenn Sie mehrere Installationen von IBM WebSphere MQ auf einem System verwenden, können Sie überprüfen, welche Versionen an welcher Position installiert sind.

„Die IBM WebSphere MQ -Konfigurationsdatei `mqs.ini`“ auf Seite 456

Die IBM WebSphere MQ-Konfigurationsdatei "`mqs.ini`" enthält Informationen, die für alle Warteschlangenmanager auf dem Knoten relevant sind. Sie wird automatisch während der Installation erstellt.

Zugehörige Tasks

[Primäre Installation auswählen](#)

Zugehörige Verweise

[setmqm](#)

[strmqm](#)

[dspm](#)

[dspmqinst](#)

Installationen von IBM WebSphere MQ auf einem System finden

Wenn Sie mehrere Installationen von IBM WebSphere MQ auf einem System verwenden, können Sie überprüfen, welche Versionen an welcher Position installiert sind.

Sie können die folgenden Methoden verwenden, um die IBM WebSphere MQ -Installationen auf Ihrem System zu suchen:

- Verwenden Sie den Befehl **dspmqver**. Der Befehl liefert keine Details zu allen Installationen auf einem System, wenn er in einer Version 7.0.1-Installation ausgegeben wird.
- Verwenden Sie die Plattforminstallationstools, um die Abfrage zu fragen, wo IBM WebSphere MQ installiert wurde. Verwenden Sie dann den Befehl **dspmqver** in einer Version 7.1-Installation oder höher. Mit den folgenden Beispielbefehlen können Sie abfragen, wo IBM WebSphere MQ installiert ist:

- Auf AIX-Systemen können Sie den Befehl **lsllp** verwenden:

```
lsllp -R ALL -l mqm.base.runtime
```

- Auf HP-UX-Systemen können Sie den Befehl **swlist** verwenden:

```
swlist -a location -a revision -l product MQSERIES
```

- Auf Linux-Systemen können Sie den Befehl **rpm** verwenden:

```
rpm -qa --qf "%{NAME}-%{VERSION}-%{RELEASE}\t%{INSTPREFIXES}\n" | grep MQSeriesRuntime
```

- Unter Solaris verwenden Sie die Befehle **pkginfo** und **pkgparam**:

1. Listen Sie die installierten Pakete auf, indem Sie den folgenden Befehl eingeben:

```
pkginfo | grep -w mqm
```

2. Geben Sie für jedes aufgelistete Paket den folgenden Befehl ein:

```
pkgparam pkgname BASEDIR
```

- Auf Windows-Systemen können Sie den Befehl **wmic** verwenden. Mit diesem Befehl kann der wmic-Client installiert werden:

```
wmic product where "(Name like '%MQ%') AND (not Name like '%bitSupport')" get Name, Version, InstallLocation
```

- Geben Sie auf UNIX and Linux-Systemen den folgenden Befehl aus, um zu ermitteln, wo IBM WebSphere MQ installiert wurde:

```
cat /etc/opt/mqm/mqinst.ini
```

Verwenden Sie anschließend den Befehl **dspmqr** in einer Installation von Version 7.1 oder höher.

- Geben Sie den folgenden Befehl aus, um Details zu Installationen auf dem System auf der 32-Bit-Version von Windows anzuzeigen:

```
reg.exe query "HKEY_LOCAL_MACHINE\SOFTWARE\IBM\WebSphere MQ\Installation" /s
```

- Geben Sie auf der 64-Bit-Version von Windows den folgenden Befehl aus:

```
reg.exe query "HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\IBM\WebSphere MQ\Installation" /s
```

Hinweis: Der Befehl **reg.exe** zeigt nur Informationen für Installationen von Version 7.1 oder höher an.

Zugehörige Konzepte

[Mehrere Installationen](#)

Zugehörige Verweise

[dspmqr](#)

[dspmqrinst](#)

Warteschlangenmanager erstellen und verwalten

Bevor Sie Nachrichten und Warteschlangen verwenden können, müssen Sie mindestens einen WS-Manager und die zugehörigen Objekte erstellen und starten.

Einen Warteschlangenmanager erstellen

Ein Warteschlangenmanager verwaltet die Ressourcen, die ihm zugeordnet sind, insbesondere die Warteschlangen, die er besitzt. Er stellt Warteschlangenservices für Anwendungen für MQI-Aufrufe (Message Queuing Interface) und Befehle zum Erstellen, Ändern, Anzeigen und Löschen von IBM WebSphere MQ-Objekten bereit.

Verwenden Sie zum Erstellen eines Warteschlangenmanagers den IBM WebSphere MQ Steuerbefehl **crtmqm** (siehe [crtmqm](#)). Der Befehl **crtmqm** erstellt automatisch die erforderlichen Standardobjekte und Systemobjekte (siehe [Systemstandardobjekte](#)). Standardobjekte bilden die Basis für alle Objektdefinitionen, die Sie machen. Systemobjekte sind für die WS-Manageroperation erforderlich. Wenn Sie einen Warteschlangenmanager und die zugehörigen Objekte erstellt haben, starten Sie den Warteschlangenmanager über den Befehl **strmqm**.

Anmerkung: IBM WebSphere MQ unterstützt keine Maschinennamen, die Leerzeichen enthalten. Wenn Sie IBM WebSphere MQ auf einem Computer mit einem Maschinennamen installieren, der Leerzeichen enthält, können Sie keine WS-Manager erstellen.

Ein

Bevor Sie einen Warteschlangenmanager erstellen können, müssen Sie verschiedene Punkte beachten (insbesondere in einer Produktionsumgebung). Arbeiten Sie durch die folgende Prüfliste:

Die dem WS-Manager zugeordnete Installation

Der Befehl `crtmqm` ordnet der Installation, von der der Befehl `crtmqm` ausgegeben wurde, automatisch einen Warteschlangenmanager zu. Für Befehle, die auf einem WS-Manager ausgeführt werden, müssen Sie den Befehl von der Installation absetzen, die dem Warteschlangenmanager zugeordnet ist. Sie können die zugeordnete Installation eines Warteschlangenmanagers mit dem Befehl `setmqm` ändern. Beachten Sie, dass das Windows -Installationsprogramm den Benutzer, der die Installation durchführt, nicht zur Gruppe 'mqm' hinzufügt. Weitere Informationen finden Sie unter [Berechtigung zur Verwaltung von IBM WebSphere MQ auf UNIX-, Linux -und Windows -Systemen](#).

Namenskonventionen

Verwenden Sie Namen in Großbuchstaben, so dass Sie mit Warteschlangenmanagern auf allen Plattformen kommunizieren können. Denken Sie daran, dass Namen genau so zugeordnet werden, wie Sie sie eingeben. Verwenden Sie nicht unnötig lange Namen, um die Unannehmlichkeiten zu vermeiden, wenn Sie viele Eingabefehler eingeben.

Geben Sie einen eindeutigen WS-Manager-Namen an

Bei der Erstellung eines Warteschlangenmanagers muss sichergestellt werden, dass er einen Namen erhält, der *keinem* anderen Warteschlangenmanager im Netz zugeordnet ist. WS-Manager-Namen werden beim Erstellen des Warteschlangenmanagers nicht überprüft, und Namen, die nicht eindeutig sind, verhindern, dass Kanäle für die verteilte Steuerung von Warteschlangen erstellt werden.

Eine Möglichkeit, die Eindeutigkeit zu gewährleisten, besteht darin, jedem WS-Managernamen seinen eigenen eindeutigen Knotennamen zu stellen. Wenn ein Knoten beispielsweise ACCOUNTS heißt, können Sie Ihren Warteschlangenmanager ACCOUNTS . SATURN . QUEUE . MANAGER nennen, wobei SATURN einen bestimmten Warteschlangenmanager identifiziert und QUEUE . MANAGER eine Erweiterung ist, die Sie allen Warteschlangenmanagern geben können. Sie können die Erweiterung auch weglassen, allerdings handelt es sich bei ACCOUNTS . SATURN und ACCOUNTS . SATURN . QUEUE . MANAGER um *verschiedene* Warteschlangenmanagernamen.

Wenn Sie IBM WebSphere MQ für die Kommunikation mit anderen Unternehmen verwenden, können Sie auch Ihren eigenen Unternehmensnamen als Präfix einschließen. Dies wird in den hier angegebenen Beispielen nicht gemacht, damit sie übersichtlich bleiben.

Anmerkung: Bei Warteschlangenmanagernamen in Steuerbefehlen muss die Groß-/Kleinschreibung beachtet werden. Dies bedeutet, dass Sie zwei Warteschlangenmanager mit den Namen `jupiter.queue.manager` und `JUPITER.queue.manager` erstellen dürfen. Es ist jedoch besser, solche Komplikationen zu vermeiden.

Anzahl der WS-Manager begrenzen

Sie können so viele WS-Manager wie Ressourcen zulassen. Da jeder WS-Manager jedoch seine eigenen Ressourcen benötigt, ist es im Allgemeinen besser, einen Warteschlangenmanager mit 100 Warteschlangen auf einem Knoten zu haben, als zehn Warteschlangenmanager mit jeweils zehn Warteschlangen zu haben.

In Produktionssystemen können viele Prozessoren mit einem einzigen Warteschlangenmanager genutzt werden, aber größere Servermaschinen werden möglicherweise effizienter mit mehreren Warteschlangenmanagern ausgeführt.

Geben Sie einen Standardwarteschlangenmanager an

Jeder Knoten sollte über einen Standardwarteschlangenmanager verfügen, obwohl es möglich ist, IBM WebSphere MQ auf einem Knoten ohne einen solchen WS-Manager zu konfigurieren. Der Standardwarteschlangenmanager ist der Warteschlangenmanager, zu dem Anwendungen eine Verbindung herstellen, wenn sie keinen Warteschlangenmanagernamen in einem MQCONN -Aufruf angeben. Es ist auch der Warteschlangenmanager, der MQSC-Befehle verarbeitet, wenn Sie den `runmqsc` -Befehl ohne Angabe eines Warteschlangenmanagernamens aufrufen.

Die Angabe eines Warteschlangenmanagers als Standard *ersetzt* alle bestehenden Standardwarteschlangenmanager-Spezifikationen für den Knoten.

Das Ändern der Standardwarteschlangenverwaltung kann sich auf andere Benutzer oder Anwendungen auswirken. Die Änderung hat keine Auswirkungen auf derzeit verbundene Anwendungen, da sie die Kennung aus ihrem ursprünglichen Verbindungsaufruf in allen weiteren MQI-Aufrufen verwenden können. Mit dieser Kennung wird sichergestellt, dass die Aufrufe an denselben Warteschlangenma-

nager übertragen werden. Alle Anwendungen, die *nach* verbinden, haben die Verbindung des Standardwarteschlangenmanagers mit dem neuen Standardwarteschlangenmanager geändert. Dies ist möglicherweise die Absicht, die Sie beabsichtigen, aber Sie sollten dies berücksichtigen, bevor Sie die Standardeinstellung ändern.

Weitere Informationen zum Erstellen eines Standardwarteschlangenmanagers finden Sie unter [„Standardwarteschlangenmanager erstellen“](#) auf Seite 23.

Geben Sie eine Warteschlange für nicht zustellbare Nachrichten an.

Die Warteschlange für nicht zustellbare Nachrichten ist eine lokale Warteschlange, in die Nachrichten gestellt werden, wenn sie nicht an die Zieladresse weitergeleitet werden können.

Es ist wichtig, dass auf jedem Warteschlangenmanager in Ihrem Netzwerk eine Warteschlange für nicht zustellbare Nachrichten vorhanden ist. Wenn Sie keinen Fehler definieren, kann es zu Fehlern in den Anwendungsprogrammen kommen, dass Kanäle geschlossen werden, und Antworten auf Verwaltungsbefehle werden möglicherweise nicht empfangen.

Wenn eine Anwendung beispielsweise versucht, eine Nachricht in eine Warteschlange in einem anderen Warteschlangenmanager zu stellen, aber den falschen Warteschlangennamen angibt, wird der Kanal gestoppt und die Nachricht verbleibt in der Übertragungswarteschlange. Andere Anwendungen können diesen Kanal dann nicht für ihre Nachrichten verwenden.

Die Kanäle sind nicht betroffen, wenn die Warteschlangenmanager Warteschlangen für nicht zustellbare Nachrichten haben. Die unzustellbare Nachricht wird am empfangenden Ende in die Warteschlange für nicht zustellbare Nachrichten gestellt, wobei der Kanal und die zugehörige Übertragungswarteschlange verfügbar sind.

Verwenden Sie bei der Erstellung eines Warteschlangenmanagers die Kennung '-u', um den Namen der Warteschlange für nicht zustellbare Nachrichten anzugeben. Sie können auch einen MQSC-Befehl verwenden, um die Attribute eines Warteschlangenmanagers zu ändern, den Sie bereits definiert haben, um die zu verwendende Warteschlange für nicht zustellbare Nachrichten anzugeben. Ein Beispiel für den MQSC-Befehl ALTER finden Sie im Abschnitt [Mit Warteschlangenmanagern arbeiten](#).

Geben Sie eine Standardübertragungswarteschlange an.

Bei einer Übertragungswarteschlange handelt es sich um eine lokale Warteschlange, in der Nachrichten, die sich im Transit zu einem fernen Warteschlangenmanager befinden, vor der Übertragung in die Warteschlange gestellt werden. Die Standardübertragungswarteschlange ist die Warteschlange, die verwendet wird, wenn keine Übertragungswarteschlange explizit definiert ist. Jedem WS-Manager kann eine Standardübertragungswarteschlange zugeordnet werden.

Verwenden Sie bei der Erstellung eines Warteschlangenmanagers die Kennung '-d', um den Namen der Standardübertragungswarteschlange anzugeben. Dadurch wird die Warteschlange nicht tatsächlich erstellt. Sie müssen dies später explizit tun. Weitere Informationen finden Sie im Abschnitt [Mit lokalen Warteschlangen arbeiten](#).

Geben Sie die erforderlichen Protokollierungsparameter an.

Sie können Protokollierungsparameter im Befehl `crtmqm` angeben, einschließlich des Protokolltyps sowie des Pfads und der Größe der Protokolldateien.

In einer Entwicklungsumgebung sollten die Standardprotokollierungsparameter ausreichend sein. Sie können die Standardwerte jedoch ändern, wenn Sie z. B.:

- Sie verfügen über eine Konfiguration mit niedrigem Endsystem, die keine großen Protokolle unterstützen kann.
- Es wird erwartet, dass eine große Anzahl langer Nachrichten gleichzeitig in den Warteschlangen enthalten ist.
- Sie erwarten eine Menge persistenter Nachrichten, die durch den Warteschlangenmanager passieren.

Nachdem Sie die Protokollierungsparameter festgelegt haben, können einige von ihnen nur geändert werden, indem der Warteschlangenmanager gelöscht und mit demselben Namen, aber mit unterschiedlichen Protokollierungsparametern erneut erstellt wird.

Weitere Informationen zu Protokollierungsparametern finden Sie im Abschnitt „Verfügbarkeit, Wiederherstellung und Neustart“ auf Seite 335.

Nur für IBM WebSphere MQ für UNIX -Systeme

Sie können das Warteschlangenmanager-Verzeichnis `/var/mqm/qmgrs/<qmgr>` auch in einem separaten lokalen Dateisystem erstellen, bevor Sie den Befehl `crtmqm` verwenden. Wenn Sie `crtmqm` verwenden und das Verzeichnis `/var/mqm/qmgrs/<qmgr>` vorhanden, leer und Eigentum von `mqm` ist, wird es für die Warteschlangenmanagerdaten verwendet. Wenn der Eigner des Verzeichnisses nicht `mqm` ist, schlägt die Erstellung mit der Nachricht First Failure Support Technology (FFST) fehl. Wenn das Verzeichnis nicht leer ist, wird ein neues Verzeichnis erstellt.

Zugehörige Konzepte

„Konfiguration“ auf Seite 5

Erstellen Sie einen oder mehrere Warteschlangenmanager auf einem oder mehreren Computern, und konfigurieren Sie sie auf Ihren Entwicklungs-, Test- und Produktionssystemen, um Nachrichten zu verarbeiten, die Ihre Geschäftsdaten enthalten.

„Konfigurationsdateien nach der Erstellung eines Warteschlangenmanagers sichern“ auf Seite 26

IBM WebSphere MQ -Konfigurationsdaten werden in Konfigurationsdateien auf Windows-, UNIX und Linux -Systemen gespeichert.

„WS-Manager starten“ auf Seite 26

Wenn Sie einen WS-Manager erstellen, müssen Sie ihn starten, damit er Befehle oder MQI-Aufrufe verarbeiten kann.

„Stoppen eines Warteschlangenmanagers“ auf Seite 27

Es gibt drei Möglichkeiten, einen Warteschlangenmanager zu stoppen: mittels eines gesteuerten Abschlusses, durch sofortige Beendigung und mittels eines präventiven Abschlusses.

„WS-Manager erneut starten“ auf Seite 28

Sie können den Befehl `strmqm` verwenden, um einen Warteschlangenmanager erneut zu starten, oder auf IBM WebSphere MQ for Windows - und IBM WebSphere MQ for Linux -Systemen (x86 - und x86-64 -Plattformen) einen Warteschlangenmanager aus IBM WebSphere MQ Explorer erneut starten.

„Konfigurationsinformationen für IBM WebSphere MQ und Warteschlangenmanager ändern“ auf Seite 454

Sie können das Verhalten von IBM WebSphere MQ oder einem einzelnen Warteschlangenmanager den Anforderungen Ihrer Installation entsprechend ändern.

System- und Standardobjekte

Zugehörige Tasks

„Vorhandenen WS-Manager als Standardwert erstellen“ auf Seite 25

Sie können einen vorhandenen Warteschlangenmanager als Standardwarteschlangenmanager definieren. Die Vorgehensweise hängt von der von Ihnen verwendeten Plattform ab.

„Löschen eines Warteschlangenmanagers“ auf Seite 28

Sie können einen Warteschlangenmanager mit dem Befehl `dlrmqm` oder mit dem WebSphere MQ Explorer löschen.

Standardwarteschlangenmanager erstellen

Der Standardwarteschlangenmanager ist der Warteschlangenmanager, zu dem Anwendungen eine Verbindung herstellen, wenn er in einem MQCONN-Aufruf keinen Warteschlangenmanagernamen angibt. Es ist auch der Warteschlangenmanager, der MQSC-Befehle verarbeitet, wenn Sie den Befehl `runmqsc` aufrufen, ohne einen Warteschlangenmanagernamen anzugeben. Zum Erstellen eines Warteschlangenmanagers verwenden Sie den IBM WebSphere MQ-Steuerbefehl `crtmqm`.

Vorbereitende Schritte

Bevor Sie einen Standardwarteschlangenmanager erstellen, lesen Sie die in „Warteschlangenmanager erstellen und verwalten“ auf Seite 20 beschriebenen Hinweise.

UNIX

Wenn Sie **crtmqm** verwenden, um einen Warteschlangenmanager unter UNIX and Linux zu erstellen, und das Verzeichnis `/var/mqm/qmgrs/<qmgr>` bereits vorhanden, Eigentum von mqm und leer ist, wird es für die Warteschlangenmanagerdaten verwendet. Wenn der Eigner des Verzeichnisses nicht mqm ist, schlägt die Erstellung des Warteschlangenmanagers mit einer First Failure Support Technology-Nachricht (FFST) fehl. Wenn das Verzeichnis nicht leer ist, wird ein neues Verzeichnis für die WS-Manager-Daten erstellt.

Diese Überlegung gilt auch dann, wenn das Verzeichnis `/var/mqm/qmgrs/<qmgr>` bereits in einem separaten lokalen Dateisystem vorhanden ist.

Informationen zu diesem Vorgang

Wenn Sie einen WS-Manager mit dem Befehl **crtmqm** erstellen, erstellt der Befehl automatisch die erforderlichen Standardobjekte und Systemobjekte. Standardobjekte bilden die Basis für alle Objektdefinitionen, die Sie erstellen, und Systemobjekte sind für die WS-Manageroperation erforderlich.

Durch die Einbeziehung der relevanten Parameter in den Befehl können Sie beispielsweise auch den Namen der Standardübertragungswarteschlange definieren, die vom Warteschlangenmanager verwendet werden soll, und den Namen der Warteschlange für nicht zustellbare Nachrichten.

Windows

Unter Windows können Sie die Option **sax** des Befehls **crtmqm** verwenden, um mehrere Instanzen des Warteschlangenmanagers zu starten.

Weitere Informationen zum Befehl **crtmqm** und seiner Syntax finden Sie unter **crtmqm**.

Prozedur

- Verwenden Sie zum Erstellen eines Standardwarteschlangenmanagers den Befehl **crtmqm** mit dem Flag **-q**.

Im folgenden Beispiel für den Befehl **crtmqm** wird ein Standardwarteschlangenmanager mit dem Namen `SATURN.QUEUE.MANAGER` erstellt:

```
crtmqm -q -d MY.DEFAULT.XMIT.QUEUE -u SYSTEM.DEAD.LETTER.QUEUE SATURN.QUEUE.MANAGER
```

Dabei gilt:

-q

Gibt an, dass dieser WS-Manager der Standardwarteschlangenmanager ist.

-d MY.DEFAULT.XMIT.QUEUE

Der Name der Standardübertragungswarteschlange, die von diesem WS-Manager verwendet werden soll.

Anmerkung: IBM WebSphere MQ erstellt keine Standardübertragungswarteschlange für Sie. Sie müssen sie selbst definieren.

-u SYSTEM.DEAD.LETTER.QUEUE

Ist der Name der Standardwarteschlange für nicht zustellbare Nachrichten, die von IBM WebSphere MQ bei der Installation erstellt wurde.

SATURN.QUEUE.MANAGER

Ist der Name dieses Warteschlangenmanagers. Dies muss der letzte Parameter sein, der im Befehl **crtmqm** angegeben wurde.

Nächste Schritte

Wenn Sie einen Warteschlangenmanager und dessen Objekte erstellt haben, verwenden Sie den Befehl **strmqm** zum Starten des Warteschlangenmanagers.

Zugehörige Konzepte

„Konfigurationsdateien nach der Erstellung eines Warteschlangenmanagers sichern“ auf Seite 26

IBM WebSphere MQ -Konfigurationsdaten werden in Konfigurationsdateien auf Windows-, UNIX and Linux -Systemen gespeichert.

[Mit Warteschlangenmanagern arbeiten](#)

[Mit lokalen Warteschlangen arbeiten](#)

Zugehörige Verweise

[System-und Standardobjekte](#)

Vorhandenen WS-Manager als Standardwert erstellen

Sie können einen vorhandenen Warteschlangenmanager als Standardwarteschlangenmanager definieren. Die Vorgehensweise hängt von der von Ihnen verwendeten Plattform ab.

WebSphere MQ für Windows und WebSphere MQ für Linux -Systeme (x86 -und x86-64 -Plattformen)

Informationen zu diesem Vorgang

Verwenden Sie die folgenden Anweisungen, wenn Sie einen vorhandenen Warteschlangenmanager als Standardwarteschlangenmanager auf WebSphere MQ für Windows -und WebSphere MQ für Linux (x86 -und x86-64 -Plattformen) definieren möchten:

Vorgehensweise

1. Öffnen Sie den IBM WebSphere MQ Explorer.
2. Klicken Sie mit der rechten Maustaste auf IBM WebSphere MQ und wählen Sie dann Properties... aus. Die Anzeige "Eigenschaften für WebSphere MQ" wird geöffnet.
3. Geben Sie den Namen des Standardwarteschlangenmanagers in das Feld "Standardname für Warteschlangenmanager" ein.
4. Klicken Sie auf "OK".

UNIX and Linux-Systeme

Informationen zu diesem Vorgang

Wenn Sie einen Standardwarteschlangenmanager erstellen, wird sein Name in das Attribut Name der Zeilengruppe `DefaultQueueManager` in der Konfigurationsdatei WebSphere MQ (`mqs.ini`) eingefügt. Die Zeilengruppe und ihr Inhalt werden automatisch erstellt, wenn sie nicht vorhanden sind.

Prozedur

- Wenn Sie einen vorhandenen Warteschlangenmanager als Standardwarteschlange verwenden möchten, ändern Sie den Namen des WS-Managers im Attribut Name in den Namen des neuen Standardwarteschlangenmanagers. Dies können Sie mit Hilfe eines Texteditors manuell ausführen.
- Wenn auf dem Knoten kein Standardwarteschlangenmanager vorhanden ist und Sie einen vorhandenen WS-Manager als Standardwarteschlange verwenden möchten, erstellen Sie die Zeilengruppe `DefaultQueueManager` mit dem erforderlichen Namen selbst.
- Wenn Sie versehentlich einen anderen Warteschlangenmanager als Standardwarteschlangenmanager definieren und zum ursprünglichen Standardwarteschlangenmanager zurückkehren möchten, bearbeiten Sie die Zeilengruppe `DefaultQueueManager` in der Datei "`mqs.ini`" und ersetzen Sie den nicht gewünschten durch den gewünschten Standardwarteschlangenmanager.

Nächste Schritte

Weitere Informationen zu Konfigurationsdateien finden Sie im Abschnitt [„Konfigurationsinformationen für IBM WebSphere MQ und Warteschlangenmanager ändern“](#) auf Seite 454.

Konfigurationsdateien nach der Erstellung eines Warteschlangenmanagers sichern

IBM WebSphere MQ -Konfigurationsdaten werden in Konfigurationsdateien auf Windows-, UNIX and Linux -Systemen gespeichert.

Verwenden Sie auf Windows -und Linux -Systemen (x86 und x86-64) IBM WebSphere MQ Explorer , um Änderungen an den Konfigurationsdateien vorzunehmen.

Auf Windows-Systemen können Sie auch mit dem Befehl `amqmdain` Änderungen an den Konfigurationsdateien vornehmen. Beachten Sie die Informationen im Abschnitt [amqmdain](#).

Es gibt zwei Typen von Konfigurationsdateien:

- Bei der Installation des Produkts wird die IBM WebSphere MQ-Konfigurationsdatei (`mqs.ini`) erstellt. Sie enthält eine Liste der Warteschlangenmanager, die bei jedem Erstellen oder Löschen eines Warteschlangenmanagers aktualisiert wird. Pro Knoten gibt es eine Datei "`mqs.ini`".
- Beim Erstellen eines neuen Warteschlangenmanagers wird automatisch auch eine neue Warteschlangenmanager-Konfigurationsdatei (`qm.ini`) erstellt. Dieser Parameter enthält Konfigurationsparameter für den Warteschlangenmanager.

Sichern Sie nach der Erstellung eines Warteschlangenmanagers Ihre Konfigurationsdateien. Wenn Sie dann einen anderen WS-Manager erstellen, der Probleme verursacht, können Sie die Sicherungen erneut erstellen, wenn Sie die Ursache des Problems entfernt haben. Sichern Sie Ihre Konfigurationsdateien in der Regel jedes Mal, wenn Sie einen neuen Warteschlangenmanager erstellen.

Weitere Informationen zu Konfigurationsdateien finden Sie im Abschnitt [„Konfigurationsinformationen für IBM WebSphere MQ und Warteschlangenmanager ändern“](#) auf Seite 454.

WS-Manager starten

Wenn Sie einen WS-Manager erstellen, müssen Sie ihn starten, damit er Befehle oder MQI-Aufrufe verarbeiten kann.

Verwenden Sie den Befehl `strmqm` , um einen Warteschlangenmanager zu starten.

Anmerkung: Sie müssen den Befehl `strmqm` aus der Installation verwenden, die dem Warteschlangenmanager zugeordnet ist, mit dem Sie arbeiten. Um herauszufinden, welcher Installation ein Warteschlangenmanager zugeordnet ist, verwenden Sie den Befehl `dspmq -o installation`.

Soll beispielsweise ein Warteschlangenmanager mit dem Namen QMB gestartet werden, geben Sie folgenden Befehl ein:

```
strmqm QMB
```

Auf WebSphere MQ für Windows und WebSphere MQ für Linux (x86 -und x86-64 -Plattformen) können Sie einen Warteschlangenmanager wie folgt starten:

1. Öffnen Sie den IBM WebSphere MQ Explorer.
2. Wählen Sie den Warteschlangenmanager in der Navigatoransicht aus.
3. Klicken Sie auf **Start**. Der WS-Manager wird gestartet.

Wenn der Start des Warteschlangenmanagers länger als ein paar Sekunden dauert, gibt WebSphere MQ Informationsnachrichten aus, die den Startfortschritt sporadisch detailliert beschreiben.

Der Befehl `strmqm` gibt die Steuerung erst zurück, wenn der WS-Manager gestartet wurde und bereit ist, Verbindungsanforderungen zu akzeptieren.

Warteschlangenmanager automatisch starten

In WebSphere MQ für Windows können Sie einen Warteschlangenmanager automatisch starten, wenn das System mit IBM WebSphere MQ Explorer gestartet wird. Weitere Informationen hierzu finden Sie im Abschnitt [Verwaltung mit dem IBM WebSphere MQ Explorer](#) .

Stoppen eines Warteschlangenmanagers

Es gibt drei Möglichkeiten, einen Warteschlangenmanager zu stoppen: mittels eines gesteuerten Abschlusses, durch sofortige Beendigung und mittels eines präventiven Abschlusses.

Mit dem Befehl **endmqm** können Sie einen Warteschlangenmanager stoppen.

Anmerkung: Sie müssen den Befehl **endmqm** aus der Installation verwenden, die dem Warteschlangenmanager zugeordnet ist, mit dem Sie arbeiten. Um herauszufinden, welcher Installation ein Warteschlangenmanager zugeordnet ist, verwenden Sie den Befehl `dspmqr -o installation`.

Soll beispielsweise ein Warteschlangenmanager mit dem Namen QMB gestoppt werden, geben Sie folgenden Befehl ein:

```
endmqm QMB
```

Auf WebSphere MQ für Windows und WebSphere MQ für Linux (x86 -und x86-64 -Plattformen) können Sie einen Warteschlangenmanager wie folgt stoppen:

1. Öffnen Sie den IBM WebSphere MQ Explorer.
2. Wählen Sie den Warteschlangenmanager in der Navigatoransicht aus.
3. Klicken Sie auf Stop . . . Die Anzeige 'End Queue Manager' wird angezeigt.
4. Wählen Sie 'Gesteuert' oder 'Sofort' aus.
5. Klicken Sie auf OK. Der Warteschlangenmanager wird gestoppt.

Gesteuerter Abschluss

Standardmäßig führt der Befehl **endmqm** einen gesteuerten Abschluss des angegebenen Warteschlangenmanagers durch. Dies kann etwas länger dauern. Bei einem gesteuerten Abschluss wird gewartet, bis alle verbundenen Anwendungen getrennt sind.

Verwenden Sie diese Art der Beendigung, um Anwendungen darüber zu benachrichtigen, dass sie gestoppt werden müssen. Bei Ausgabe des Befehls:

```
endmqm -c QMB
```

erhalten Sie keine Benachrichtigung, wenn alle Anwendungen gestoppt sind. (Ein `endmqm -c QMB`-Befehl entspricht einem `endmqm QMB`-Befehl.)

Bei Ausgabe des Befehls:

```
endmqm -w QMB
```

wartet der Befehl hingegen, bis alle Anwendungen gestoppt sind und der Warteschlangenmanager beendet ist.

Sofortige Beendigung

Bei einer sofortigen Beendigung werden alle aktuellen MQI-Aufrufe zu Ende geführt, neue Aufrufe schlagen jedoch fehl. Dieser Typ des Herunterfahrens wartet nicht darauf, dass Anwendungen vom WS-Manager getrennt werden.

Geben Sie für eine sofortige Beendigung Folgendes ein:

```
endmqm -i QMB
```

Präventiver Systemabschluss

Anmerkung: Diese Abschlussmethode sollten Sie nur verwenden, wenn alle anderen Versuche, den Warteschlangenmanager mit dem Befehl **endmqm** zu stoppen, fehlgeschlagen sind. Die Auswirkungen dieser Methode auf die verbundenen Anwendungen sind unvorhersehbar.

Falls eine sofortige Beendigung nicht funktioniert, müssen Sie mit dem Flag `-p` auf einen *präventiven* Abschluss zurückgreifen. Beispiel:

```
endmqm -p QMB
```

Dadurch wird der Warteschlangenmanager sofort gestoppt. Wenn diese Methode immer noch nicht funktioniert, beachten Sie die Informationen im Abschnitt [Warteschlangenmanager manuell stoppen](#) bezüglich einer alternativen Lösung.

Eine ausführliche Beschreibung des Befehls **endmqm** und seiner Optionen finden Sie im Abschnitt [endmqm](#).

Probleme beim Abschluss eines Warteschlangenmanagers

Probleme beim Abschluss eines Warteschlangenmanagers haben häufig ihre Ursache in einzelnen Anwendungen. Zum Beispiel, wenn Anwendungen:

- MQI-Rückkehrcodes nicht korrekt überprüfen
- Keine Benachrichtigung über Quiesce anfordern
- Beenden, ohne die Verbindung zum Warteschlangenmanager zu trennen (durch Ausgabe eines MQDISC-Aufrufs)

Bei Problemen beim Stoppen eines Warteschlangenmanagers können Sie den Befehl **endmqm** mit der Tastenkombination STRG-C abbrechen. Danach können Sie einen anderen Befehl **endmqm** mit einem Flag für die erforderliche Abschlussmethode ausgeben.

WS-Manager erneut starten

Sie können den Befehl **strmqm** verwenden, um einen Warteschlangenmanager erneut zu starten, oder auf IBM WebSphere MQ for Windows -und IBM WebSphere MQ for Linux -Systemen (x86 -und x86-64 -Plattformen) einen Warteschlangenmanager aus IBM WebSphere MQ Explorer erneut starten.

Geben Sie zum erneuten Starten eines Warteschlangenmanagers Folgendes ein:

```
strmqm saturn.queue.manager
```

Auf IBM WebSphere MQ for Windows -und IBM WebSphere MQ for Linux -Systemen (x86 -und x86-64 -Plattformen) können Sie einen Warteschlangenmanager wie folgt erneut starten:

1. Öffnen Sie den IBM WebSphere MQ Explorer.
2. Wählen Sie den Warteschlangenmanager in der Navigatoransicht aus.
3. Klicken Sie auf Start. Der WS-Manager wird erneut gestartet.

Wenn der Neustart des Warteschlangenmanagers länger als ein paar Sekunden dauert, gibt IBM WebSphere MQ Informationsnachrichten aus, die den Startfortschritt sporadisch detailliert beschreiben.

Löschen eines Warteschlangenmanagers

Sie können einen Warteschlangenmanager mit dem Befehl **dltmqm** oder mit dem WebSphere MQ Explorer löschen.

Vorbereitende Schritte

Stoppen Sie den Warteschlangenmanager.

Prozedur

- Geben Sie den folgenden Befehl aus: `dltmqm QMB`

Anmerkung: Sie müssen den Befehl **dltmqm** aus der Installation verwenden, die dem Warteschlangenmanager zugeordnet ist, mit dem Sie arbeiten. Um herauszufinden, welcher Installation ein Warteschlangenmanager zugeordnet ist, verwenden Sie den Befehl `dspm q -o installation`.

Schritte zum Löschen eines Warteschlangenmanagers

Informationen zu diesem Vorgang

Auf WebSphere MQ für Windows und WebSphere MQ für Linux (x86 -und x86-64 -Plattformen) können Sie einen Warteschlangenmanager wie folgt löschen:

Vorgehensweise

1. Öffnen Sie WebSphere MQ Explorer.
2. Wählen Sie in der Navigatoransicht den WS-Manager aus.
3. Wenn der WS-Manager nicht gestoppt wurde, stoppen Sie ihn.
 - a) Klicken Sie mit der rechten Maustaste auf den Warteschlangenmanager.
 - b) Klicken Sie auf **Stoppen** .
4. Klicken Sie mit der rechten Maustaste auf den Warteschlangenmanager.
5. Klicken Sie auf **Löschen** .

Ergebnisse

Der WS-Manager wird gelöscht.



Achtung:

- Das Löschen eines Warteschlangenmanagers ist ein drastischer Schritt, da Sie auch alle Ressourcen löschen, die dem Warteschlangenmanager zugeordnet sind, einschließlich aller Warteschlangen und deren Nachrichten und aller Objektdefinitionen. Wenn Sie den Befehl **dltmqm** verwenden, wird keine Eingabeaufforderung angezeigt, mit der Sie Ihre Meinung ändern können. Wenn Sie die Eingabetaste drücken, gehen alle zugeordneten Ressourcen verloren.
- Wenn Sie in WebSphere MQ für Windows einen Warteschlangenmanager löschen, wird er auch aus der automatischen Startliste entfernt (siehe „WS-Manager starten“ auf Seite 26). Nach Abschluss des Befehls wird die Nachricht `WebSphere MQ queue manager ending` angezeigt. Sie werden nicht darüber informiert, dass der Warteschlangenmanager gelöscht wurde.
- Wenn Sie einen Cluster-WS-Manager löschen, wird er nicht aus dem Cluster entfernt. Weitere Informationen finden Sie im Hinweis in der Beschreibung des Befehls **dltmqm**.

Eine Beschreibung des Befehls **dltmqm** und seiner Optionen finden Sie unter [dltmqm](#). Stellen Sie sicher, dass nur vertrauenswürdige Administratoren die Berechtigung haben, diesen Befehl zu verwenden. (Informationen zur Sicherheit finden Sie unter [Setting up security on Windows, UNIX and Linux systems](#) .)


Anwendungen mithilfe der verteilten Steuerung von Warteschlangen verbinden

In diesem Abschnitt finden Sie ausführliche Informationen zur übergreifenden Kommunikation zwischen WebSphere MQ-Installationen, einschließlich Informationen zu Warteschlangendefinitionen, Kanaldefinitionen, Auslösungen und Synchronisationspunktprozeduren.

Bevor Sie diesen Abschnitt lesen, sollten Sie sich mit Kanälen, Warteschlangen und sonstigen Konzepten, die im Thema [Konzepte der übergreifenden Kommunikation](#) beschrieben werden, vertraut machen.

Verwenden Sie die Informationen in den folgenden Links, um Ihre Anwendungen mithilfe der verteilten Steuerung von Warteschlangen zu verbinden:

- [„So senden Sie eine Nachricht an einen anderen Warteschlangenmanager“](#) auf Seite 55
- [„Ausgelöste Kanäle“](#) auf Seite 73

- [„Sicherheit von Nachrichten“ auf Seite 71](#)
- [„IBM WebSphere MQ -Verfahren für verteiltes Messaging“ auf Seite 30](#)
- [„Einführung in die verteilte Warteschlangenverwaltung“ auf Seite 52](#)
-  [„Kanäle in UNIX, Linux, and Windows überwachen und steuern“ auf Seite 80](#)

Zugehörige Konzepte

[„Verbindungen zwischen Client und Server konfigurieren“ auf Seite 106](#)

Um die Kommunikationsverbindungen zwischen WebSphere MQ MQI-Clients und -Servern zu konfigurieren, müssen Sie Ihr Kommunikationsprotokoll festlegen, die Verbindungen an beiden Enden der Verbindung definieren, ein Empfangsprogramm starten und Kanäle definieren.

[„Konfigurationsinformationen für IBM WebSphere MQ und Warteschlangenmanager ändern“ auf Seite 454](#)

Sie können das Verhalten von IBM WebSphere MQ oder einem einzelnen Warteschlangenmanager den Anforderungen Ihrer Installation entsprechend ändern.

Zugehörige Tasks

[„WS-Manager-Cluster konfigurieren“ auf Seite 171](#)

Über die Links in diesem Abschnitt erfahren Sie, wie Cluster funktionieren, wie eine Clusterkonfiguration entworfen wird und wie ein einfacher Cluster eingerichtet wird.

IBM WebSphere MQ -Verfahren für verteiltes Messaging

In den Unterabschnitten in diesem Abschnitt werden die Verfahren beschrieben, die bei der Planung von Kanälen verwendet werden. In diesen Unterabschnitten werden Verfahren beschrieben, mit deren Hilfe Sie planen, wie die Warteschlangenmanager miteinander verbunden werden, und den Fluss von Nachrichten zwischen Ihren Anwendungen verwalten.

Informationen zu den Planungsbeispielen für Nachrichtenkanäle finden Sie unter:

- [Beispiel für Nachrichtenkanalplanung für verteilte Plattformen](#)

Zugehörige Konzepte

[„Anwendungen mithilfe der verteilten Steuerung von Warteschlangen verbinden“ auf Seite 29](#)

In diesem Abschnitt finden Sie ausführliche Informationen zur übergreifenden Kommunikation zwischen WebSphere MQ-Installationen, einschließlich Informationen zu Warteschlangendefinitionen, Kanaldefinitionen, Auslösungen und Synchronisationspunktprozeduren.

[Kanäle](#)

[Einführung in die Nachrichtenwarteschlangensteuerung](#)

[Konzepte der übergreifenden Kommunikation](#)

Zugehörige Verweise

[Beispielkonfigurationsdaten](#)

Nachrichtenflusssteuerung

Die Nachrichtenflusssteuerung ist eine Task, die die Einrichtung und Verwaltung von Nachrichtenrouten zwischen Warteschlangenmanagern umfasst. Es ist wichtig für Routen, die Multi-Hop durch viele WS-Manager. In diesem Abschnitt wird beschrieben, wie Sie Warteschlangen, Aliaswarteschlangendefinitionen und Nachrichtenkanäle auf Ihrem System verwenden, um die Nachrichtenflusssteuerung zu erreichen.

Sie steuern den Nachrichtenfluss mit einer Reihe von Verfahren, die in [„Anwendungen mithilfe der verteilten Steuerung von Warteschlangen verbinden“ auf Seite 29](#) eingeführt wurden. Wenn sich Ihr Warteschlangenmanager in einem Cluster befindet, wird der Nachrichtenfluss unter Verwendung unterschiedlicher Verfahren gesteuert, wie in [„Nachrichtenflusssteuerung“ auf Seite 30](#) beschrieben.

Sie können die folgenden Objekte verwenden, um die Nachrichtenflusssteuerung zu erreichen:

- Übertragungswarteschlangen

- Nachrichtenkanäle
- Definition der fernen Warteschlange
- WS-Manager-Aliasdefinition
- Aliasdefinition der Warteschlange für Antwortwarteschlange

Der Warteschlangenmanager und die Warteschlangenobjekte werden in Objekte beschrieben. Nachrichtenkanäle werden im Abschnitt Komponenten für die verteilte Steuerung von Warteschlangen beschrieben. Die folgenden Verfahren verwenden diese Objekte, um Nachrichtenflüsse in Ihrem System zu erstellen:

- Nachrichten in ferne Warteschlangen stellen
- Routing über bestimmte Übertragungswarteschlangen
- Nachrichten empfangen
- Nachrichten über das System übergeben
- Getrennte Nachrichtenflüsse
- Nachrichtenflüsse zu einem anderen Ziel wechseln
- Auflösen des Namens der Empfangswarteschlange für Antworten auf einen Aliasnamen

Hinweis

Alle in diesem Abschnitt beschriebenen Konzepte sind für alle Knoten in einem Netz relevant und schließen das Senden und Empfangen von Nachrichtenkanalenden ein. Aus diesem Grund ist in den meisten Beispielen nur ein Knoten dargestellt. Die Ausnahme ist die explizite Zusammenarbeit des Administrators am anderen Ende eines Nachrichtenkanals durch den Administrator.

Bevor Sie mit den einzelnen Verfahren fortfahren, ist es sinnvoll, die Konzepte der Namensauflösung und die drei Möglichkeiten der Verwendung von Definitionen ferner Warteschlangen zu verwenden. Siehe Konzepte der übergreifenden Kommunikation.

Zugehörige Konzepte

„Warteschlangennamen in Übertragungsheader“ auf Seite 31

Zielwarteschlangennamen werden mit der Nachricht in den Übertragungsheader übertragen, bis die Zielwarteschlange erreicht ist.

„Vorgehensweise zum Erstellen von Warteschlangenmanagern und Antwortaliasnamen“ auf Seite 31

In diesem Thema werden die drei Möglichkeiten erläutert, wie Sie eine Definition einer fernen Warteschlange erstellen können.

Warteschlangennamen in Übertragungsheader

Zielwarteschlangennamen werden mit der Nachricht in den Übertragungsheader übertragen, bis die Zielwarteschlange erreicht ist.

Der von der Anwendung verwendete Warteschlangennamen, der Name der logischen Warteschlange, wird vom WS-Manager in den Namen der Zielwarteschlange aufgelöst. Mit anderen Worten: der Name der physischen Warteschlange. Dieser Zielwarteschlangennamen wird mit der Nachricht in einem separaten Datenbereich, dem Übertragungsheader, übertragen, bis die Zielwarteschlange erreicht ist. Der Übertragungsheader wird dann abgestreift.

Sie ändern den Warteschlangenmanagerteil dieses Warteschlangennamens, wenn Sie parallele Serviceklassen erstellen. Denken Sie daran, den Namen des Warteschlangenmanagers an den ursprünglichen Namen zurückzugeben, wenn das Ende der Serviceklassendiversion erreicht ist.

Vorgehensweise zum Erstellen von Warteschlangenmanagern und Antwortaliasnamen

In diesem Thema werden die drei Möglichkeiten erläutert, wie Sie eine Definition einer fernen Warteschlange erstellen können.

Das Definitionsobjekt für die ferne Warteschlange wird auf drei verschiedene Arten verwendet. Tabelle 3 auf Seite 32 In wird erläutert, wie die drei Methoden definiert werden:

- Verwenden Sie eine Definition einer fernen Warteschlange, um einen lokalen Warteschlangennamen erneut zu definieren.

Die Anwendung stellt beim Öffnen einer Warteschlange nur den Warteschlangennamen zur Verfügung, und dieser Warteschlangename ist der Name der Definition der fernen Warteschlange.

Die Definition der fernen Warteschlange enthält die Namen der Zielwarteschlange und des Warteschlangenmanagers. Optional kann die Definition den Namen der Übertragungswarteschlange enthalten, die verwendet werden soll. Wenn kein Übertragungswarteschlangename angegeben wird, verwendet der Warteschlangenmanager den Namen des Warteschlangenmanagers, der aus der Definition der fernen Warteschlange entnommen wurde, für den Namen der Übertragungswarteschlange. Wenn eine Übertragungswarteschlange mit diesem Namen nicht definiert ist, aber eine Standardübertragungswarteschlange definiert ist, wird die Standardübertragungswarteschlange verwendet.

- Verwenden Sie eine Definition einer fernen Warteschlange, um einen WS-Manager-Namen erneut zu definieren.

Das Anwendungs- oder Kanalprogramm stellt beim Öffnen der Warteschlange einen Warteschlangennamen zusammen mit dem Namen des fernen Warteschlangenmanagers bereit.

Wenn Sie eine ferne Warteschlangendefinition mit demselben Namen wie der Name des Warteschlangenmanagers angegeben haben und den Warteschlangennamen in der Definition leer gelassen haben, ersetzt der Warteschlangenmanager den Namen des Warteschlangenmanagers im offenen Aufruf mit dem Namen des Warteschlangenmanagers in der Definition.

Darüber hinaus kann die Definition den Namen der Übertragungswarteschlange enthalten, die verwendet werden soll. Wenn kein Übertragungswarteschlangename angegeben wird, verwendet der Warteschlangenmanager den Namen des Warteschlangenmanagers, der aus der Definition der fernen Warteschlange für den Namen der Übertragungswarteschlange entnommen wurde. Wenn eine Übertragungswarteschlange mit diesem Namen nicht definiert ist, aber eine Standardübertragungswarteschlange definiert ist, wird die Standardübertragungswarteschlange verwendet.

- Verwenden Sie eine Definition einer fernen Warteschlange, um einen Namen der Empfangswarteschlange für Antworten erneut zu definieren.

Jedes Mal, wenn eine Anwendung eine Nachricht in eine Warteschlange einreicht, kann sie den Namen einer Warteschlange für Antwortnachrichten für Antwortnachrichten angeben, jedoch mit dem Namen des Warteschlangenmanagers.

Wenn Sie eine ferne Warteschlangendefinition mit demselben Namen wie die Empfangswarteschlange für Antworten angeben, ersetzt der lokale WS-Manager den Namen der Warteschlange für die Antwort auf die Warteschlange durch den Namen der Warteschlange aus Ihrer Definition.

Sie können einen Warteschlangenmanagernamen in der Definition angeben, jedoch keinen Namen für die Übertragungswarteschlange.

<i>Tabelle 3. Drei Methoden zur Verwendung des Definitionsobjekts für ferne Warteschlangen</i>			
Verwendung	Name des Warteschlangenmanagers	Warteschlangennamen	Name der Übertragungswarteschlange
1. Definition der fernen Warteschlange (bei OPEN-Aufruf)			
Bereitgestellt im Aufruf	Leer oder lokaler QM	(*) erforderlich	-
Bereitgestellt in der Definition	Erforderlich	Erforderlich	optional
2. WS-Manager-Aliasname (bei OPEN-Aufruf)			
Bereitgestellt im Aufruf	(*) erforderlich und nicht lokaler QM	Erforderlich	-

Tabelle 3. Drei Methoden zur Verwendung des Definitionsobjekts für ferne Warteschlangen (Forts.)			
Verwendung	Name des Warteschlangenmanagers	Warteschlangenna-me	Name der Übertragungswarteschlange
Bereitgestellt in der Definition	Erforderlich	Leer	optional
3. Aliasname für Antwortwarteschlange (bei PUT-Aufruf)			
Bereitgestellt im Aufruf	Leer	(*) erforderlich	-
Bereitgestellt in der Definition	optional	optional	Leer
Anmerkung: (*) bedeutet, dass dieser Name der Name des Definitionsobjekts ist.			

Eine formale Beschreibung finden Sie in [Warteschlangennamensauflösung](#).

Einreihen von Nachrichten in ferne Warteschlangen

Sie können ferne Warteschlangendefinitionsobjekte verwenden, um einen Warteschlangennamen in eine Übertragungswarteschlange in einen benachbarten WS-Manager aufzulösen.

In einer Umgebung mit verteilter Warteschlange sind eine Übertragungswarteschlange und ein Kanal der Sammelpunkt für alle Nachrichten an eine Position, ob die Nachrichten von Anwendungen in Ihrem lokalen System stammen oder die über Kanäle von einem benachbarten System ankommen. [Abbildung 2](#) auf Seite 33 zeigt eine Anwendung, die Nachrichten in eine logische Warteschlange mit dem Namen 'QA_norm' stellt. In der Namensauflösung wird die ferne Warteschlangendefinition 'QA_norm' verwendet, um die Übertragungswarteschlange QMB auszuwählen. Anschließend wird ein Übertragungsheader zu den Nachrichten mit der Nachricht 'QA_norm at QMB' hinzugefügt.

Nachrichten, die vom benachbarten System in 'Channel_back' ankommen, weisen einen Übertragungsheader mit dem Namen der physischen Warteschlange 'QA_norm auf QMB' auf. Beispiel: Diese Nachrichten werden unverändert in die Übertragungswarteschlange QMB gestellt.

Der Kanal verschiebt die Nachrichten in einen benachbarten WS-Manager.

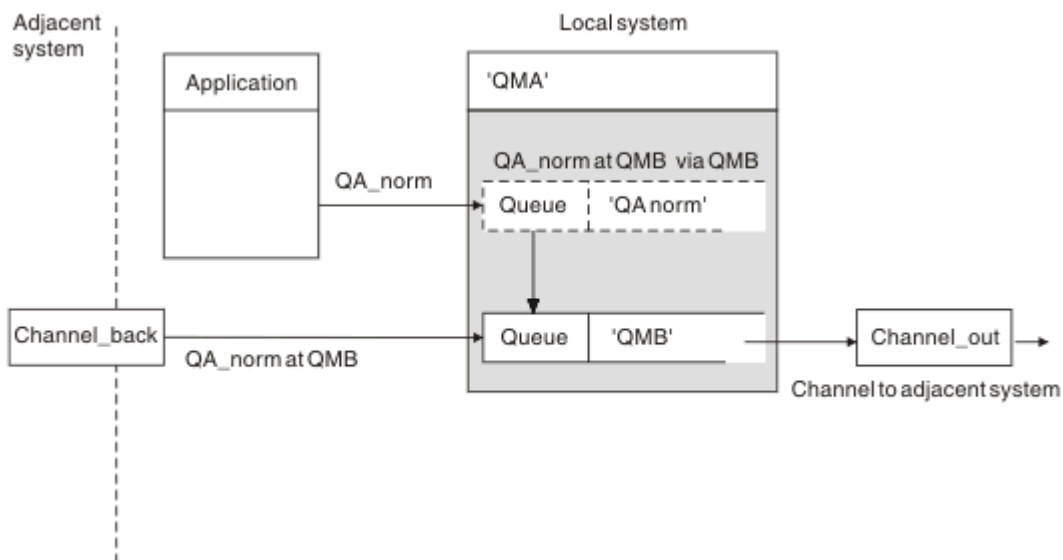


Abbildung 2. Eine Definition einer fernen Warteschlange wird verwendet, um einen Warteschlangennamen in eine Übertragungswarteschlange in einen benachbarten WS-Manager aufzulösen.

Wenn Sie der WebSphere MQ -Systemadministrator sind, müssen Sie Folgendes ausführen:

- Den Nachrichtenkanal aus dem benachbarten System definieren
- Definieren Sie den Nachrichtenkanal für das benachbarte System.
- Erstellen Sie die Übertragungswarteschlange QMB.
- Definieren Sie das ferne Warteschlangenobjekt 'QA_norm', um den Namen der Warteschlange aufzulösen, die von Anwendungen für den Namen der Zielwarteschlange, den Namen des Zielwarteschlangenmanagers und den Namen der Übertragungswarteschlange verwendet wird.

In einer Clustering-Umgebung müssen Sie nur einen Clusterempfängerkanal auf dem lokalen Warteschlangenmanager definieren. Es ist nicht erforderlich, eine Übertragungswarteschlange oder ein fernes Warteschlangenobjekt zu definieren. Weitere Informationen finden Sie unter [Cluster](#).

Weitere Informationen zur Namensauflösung

Der Effekt der Definition einer fernen Warteschlange besteht darin, einen Namen für die physische Zielwarteschlange und den Namen des Warteschlangenmanagers zu definieren. Diese Namen werden in die Übertragungsheader von Nachrichten gestellt.

Eingehende Nachrichten von einem benachbarten System haben bereits diesen Typ von Namensauflösung, die vom ursprünglichen Warteschlangenmanager ausgeführt wurde. Daher weisen sie den Übertragungsheader mit dem Namen der physischen Zielwarteschlange und dem Namen des Warteschlangenmanagers auf. Diese Nachrichten werden von den Definitionen der fernen Warteschlange nicht beeinflusst.

Die Übertragungswarteschlange auswählen

Sie können eine ferne Warteschlangendefinition verwenden, um eine andere Übertragungswarteschlange zuzulassen, um Nachrichten an denselben benachbarten Warteschlangenmanager zu senden.

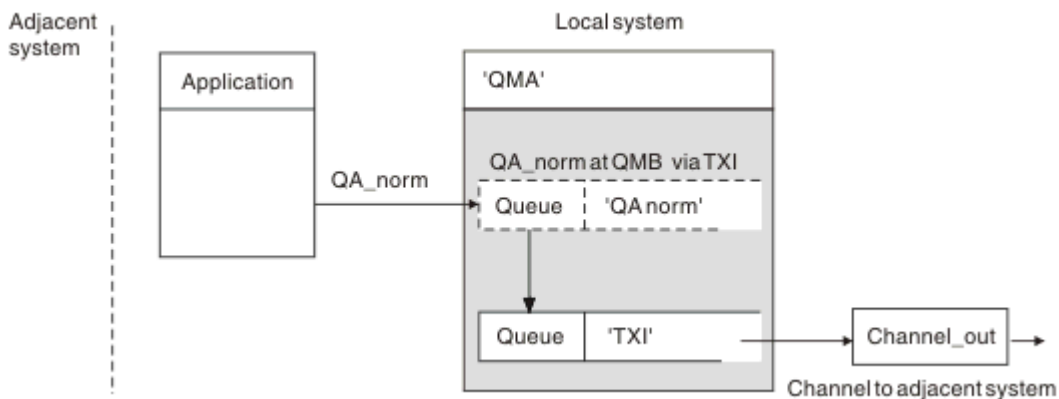


Abbildung 3. Die Definition der fernen Warteschlange ermöglicht, dass eine andere Übertragungswarteschlange verwendet wird.

Wenn Sie in einer Umgebung mit verteilter Warteschlange einen Nachrichtenfluss von einem Kanal in einen anderen ändern müssen, verwenden Sie dieselbe Systemkonfiguration wie in [Abbildung 2 auf Seite 33](#) in „Einreihen von Nachrichten in ferne Warteschlangen“ auf [Seite 33](#) gezeigt. [Abbildung 3 auf Seite 34](#) in diesem Thema zeigt, wie Sie die Definition der fernen Warteschlange verwenden, um Nachrichten über eine andere Übertragungswarteschlange und somit über einen anderen Kanal an denselben benachbarten Warteschlangenmanager zu senden.

Für die in [Abbildung 3 auf Seite 34](#) gezeigte Konfiguration müssen Sie das ferne Warteschlangenobjekt 'QA_norm' und die Übertragungswarteschlange 'TX1' angeben. Sie müssen 'QA_norm' angeben, um die Warteschlange 'QA_norm' auf dem fernen WS-Manager, die Übertragungswarteschlange 'TX1' und den Warteschlangenmanager 'QMB_priority' auszuwählen. Geben Sie 'TX1' in der Definition des Kanals an, der an das System angrenzender Kanal ist.

Nachrichten werden in die Übertragungswarteschlange 'TX1' mit einem Übertragungsheader gestellt, der 'QA_norm bei QMB_priority' enthält, und werden über den Kanal an das benachbarte System gesendet.

Der Kanal 'channel_back' wurde aus dieser Abbildung weggelassen, da er einen WS-Manager-Aliasnamen benötigen würde.

In einer Clustering-Umgebung müssen Sie keine Übertragungswarteschlange oder eine Definition einer fernen Warteschlange definieren. Weitere Informationen finden Sie im Abschnitt „Clusterwarteschlangen“ auf Seite 179.

Nachrichten empfangen

Sie können den Warteschlangenmanager für den Empfang von Nachrichten von anderen Warteschlangenmanagern konfigurieren. Sie müssen sicherstellen, dass eine unbeabsichtigte Namensauflösung nicht auftritt.

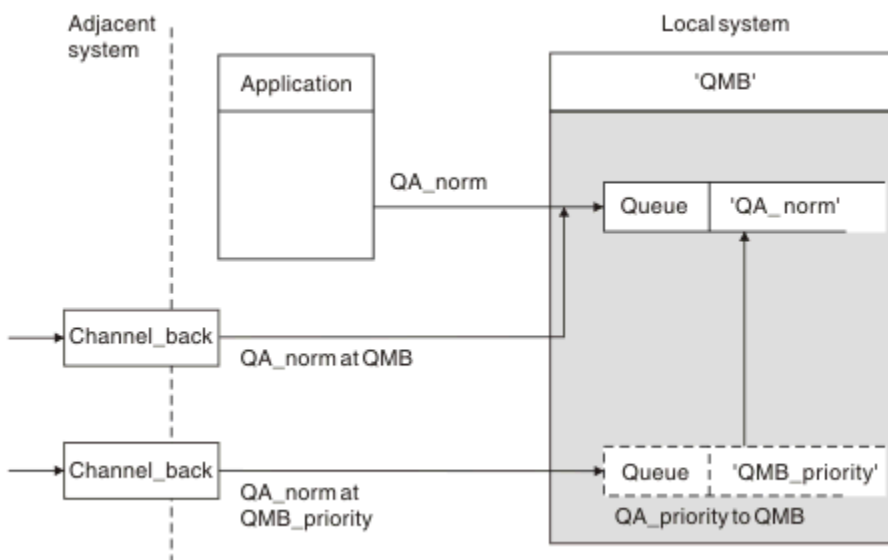


Abbildung 4. Nachrichten direkt empfangen und Aliasnamen-WS-Manager-Namen auflösen

Neben der Anordnung von Nachrichten, die gesendet werden sollen, muss der Systemadministrator auch die Nachrichten veranlassen, die von benachbarten Warteschlangenmanagern empfangen werden. Empfangene Nachrichten enthalten den physischen Namen des Zielwarteschlangenmanagers und die Warteschlange in der Übertragungsheader. Sie werden mit Nachrichten aus einer lokalen Anwendung behandelt, die sowohl den Namen des Warteschlangenmanagers als auch den Warteschlangennamen angibt. Aufgrund dieser Behandlung müssen Sie sicherstellen, dass Nachrichten, die in Ihr System eingegeben werden, nicht über eine unbeabsichtigte Namensauflösung verfügen. Dieses Szenario enthält [Abbildung 4 auf Seite 35](#).

Für diese Konfiguration müssen Sie Folgendes vorbereiten:

- Nachrichtenkanäle zum Empfangen von Nachrichten von benachbarten Warteschlangenmanagern
- Eine WS-Manager-Aliasdefinition zum Auflösen eines eingehenden Nachrichtenflusses, 'QMB_priority', in den lokalen WS-Manager-Namen 'QMB'
- Die lokale Warteschlange 'QA_norm', falls sie nicht vorhanden ist.

Namen von Aliaswarteschlangenmanagern empfangen

Die Verwendung der Definition des WS-Manager-Aliasnamens in dieser Abbildung hat keinen anderen Zielwarteschlangenmanager ausgewählt. Nachrichten, die diesen lokalen WS-Manager durchlaufen und an 'QMB_priority' adressiert sind, sind für den Warteschlangenmanager 'WSMB' bestimmt. Der Name des Aliaswarteschlangenmanagers wird zum Erstellen des separaten Nachrichtenflusses verwendet.

Nachrichten über das System übergeben

Sie können Nachrichten über das System auf drei Arten übergeben—unter Verwendung des Positionsnamens, unter Verwendung eines Aliasnamens für den Warteschlangenmanager oder durch Auswahl einer Übertragungswarteschlange.

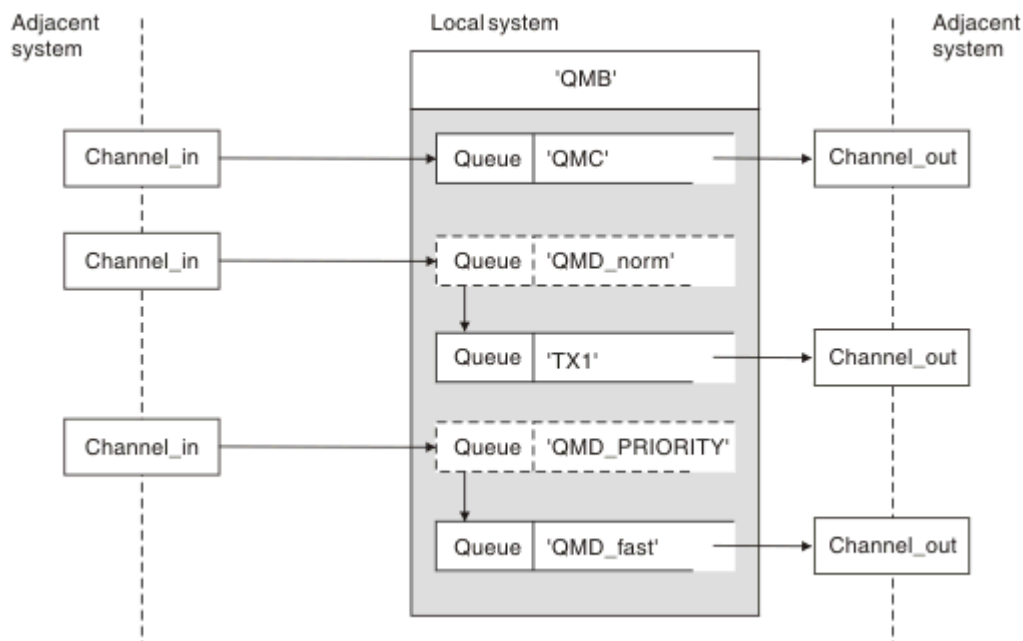


Abbildung 5. Drei Methoden zum Übergeben von Nachrichten durch Ihr System

Das in [Abbildung 4](#) auf Seite 35 in „Nachrichten empfangen“ auf Seite 35 gezeigte Verfahren hat gezeigt, wie ein Aliasfluss erfasst wird. [Abbildung 5](#) auf Seite 36 zeigt, wie Netzwerke aufgebaut werden, indem die zuvor beschriebenen Verfahren zusammengeführt werden.

Die Konfiguration zeigt einen Kanal, der drei Nachrichten mit unterschiedlichen Zieladressen liefert:

1. QB an QMC
2. QB an QMD_norm
3. QB an QMD_PRIORITY

Sie müssen den ersten Nachrichtenfluss unverändert über Ihr System übergeben. Sie müssen den zweiten Nachrichtenfluss über eine andere Übertragungswarteschlange und einen anderen Kanal übergeben. Für den zweiten Nachrichtenfluss müssen Sie auch Nachrichten für den Aliasnamen WS-Managername QMD_norm in den Warteschlangenmanager QMD auflösen. Der dritte Nachrichtenfluss wählt eine andere Übertragungswarteschlange ohne jede andere Änderung aus.

In einer Clustering-Umgebung werden Nachrichten über eine Clusterübertragungswarteschlange übergeben. Normalerweise überträgt eine einzelne Übertragungswarteschlange (SYSTEM.CLUSTER.TRANS-MIT.QUEUE) alle Nachrichten an alle Warteschlangenmanager in allen Clustern, zu denen der Warteschlangenmanager gehört. Weitere Informationen finden Sie im Abschnitt [Cluster von Warteschlangenmanagern](#). Sie können separate Übertragungswarteschlangen für alle oder einige der WS-Manager in den Clustern definieren, zu denen der Warteschlangenmanager gehört.

Die folgenden Methoden beschreiben Verfahren, die auf eine Umgebung mit verteilten Warteschlangen anwendbar sind.

Verwenden Sie diese Methoden.

Für diese Konfigurationen müssen Sie Folgendes vorbereiten:

- Eingabekanaldefinitionen

- Ausgabekanaldefinitionen
- Übertragungswarteschlangen:
 - QMC
 - TX1
 - QMD_fast
- Definitionen des WS-Manager-Aliasnamens
 - QMD_norm mit QMD_norm an QMD über TX1
 - QMD_PRIORITY mit QMD_PRIORITY an QMD_PRIORITY über QMD_fast

Anmerkung: Keiner der Nachrichtenflüsse, die im Beispiel angezeigt werden, ändert die Zielwarteschlange. Die Aliasnamen des Warteschlangenmanagers stellen die Trennung von Nachrichtenflüssen zur Verfügung.

Methode 1: Ankommenden Standortnamen verwenden

Sie werden Nachrichten mit einem Übertragungsheader empfangen, der einen anderen Standortnamen enthält, z. B. QMC. Die einfachste Konfiguration besteht darin, eine Übertragungswarteschlange mit diesem Namen zu erstellen, QMC. Der Kanal, der die Übertragungswarteschlange bereitstellt, übergibt die Nachricht unverändert an das nächste Ziel.

Methode 2: Aliasnamen für den WS-Manager verwenden

Die zweite Methode ist die Verwendung der WS-Manager-Aliasobjektdefinition, aber geben Sie einen neuen Standortnamen, QMD und eine bestimmte Übertragungswarteschlange an, TX1. Diese Aktion:

- Beendet den Aliasnachrichtenfluss, der durch den Aliasnamen QMD_norm des Warteschlangenmanagers, d. h. die benannte Serviceklasse QMD_norm, konfiguriert wird.
- Ändert die Übertragungsheader in diesen Nachrichten von QMD_norm in QMD.

Methode 3: Wählen Sie eine Übertragungswarteschlange aus.

Die dritte Methode besteht darin, dass ein WS-Manager-Aliasobjekt mit demselben Namen wie die Zielposition QMD_PRIORITY definiert ist. Verwenden Sie die Definition des WS-Manager-Aliasnamens, um eine bestimmte Übertragungswarteschlange, QMD_fast und somit einen anderen Kanal auszuwählen. Die Übertragungsheader in diesen Nachrichten bleiben unverändert.

Getrennte Nachrichtenflüsse

Sie können einen WS-Manager-Aliasnamen verwenden, um separate Nachrichtenflüsse zu erstellen, um Nachrichten an denselben WS-Manager zu senden.

In einer Umgebung mit verteilten Warteschlangen kann die Notwendigkeit, Nachrichten an denselben WS-Manager in verschiedene Nachrichtenflüsse zu trennen, aus einer Reihe von Gründen auftreten. Beispiel:

- Möglicherweise müssen Sie einen separaten Nachrichtenfluss für große, mittlere und kleine Nachrichten bereitstellen. Diese Notwendigkeit gilt auch in einer Clustering-Umgebung, und in diesem Fall können Sie Cluster erstellen, die sich überschneiden. Es gibt eine Reihe von Gründen, die Sie vielleicht tun könnten, z. B.:
 - Damit andere Organisationen ihre eigene Verwaltung haben können.
 - Damit unabhängige Anwendungen separat verwaltet werden können.
 - So erstellen Sie eine Serviceklasse. Sie könnten z. B. einen Cluster mit dem Namen STAFF haben, der eine Untergruppe des Clusters mit dem Namen STUDENTS ist. Wenn Sie eine Nachricht in eine Warteschlange stellen, die im STAFF-Cluster zugänglich gemacht wird, wird ein eingeschränkter

Kanal verwendet. Wenn Sie eine Nachricht in eine Warteschlange stellen, die im STUDENTS-Cluster zugänglich gemacht wird, kann entweder ein allgemeiner Kanal oder ein eingeschränkter Kanal verwendet werden.

- So erstellen Sie Test- und Produktionsumgebungen.
- Es kann erforderlich sein, eingehende Nachrichten durch unterschiedliche Pfade aus dem Pfad der lokal generierten Nachrichten weiterzuleiten.
- Ihre Installation muss unter Umständen die Verschiebung von Nachrichten zu bestimmten Zeiten (z. B. über Nacht) planen, und die Nachrichten müssen dann in reservierten Warteschlangen gespeichert werden, bis sie geplant sind.

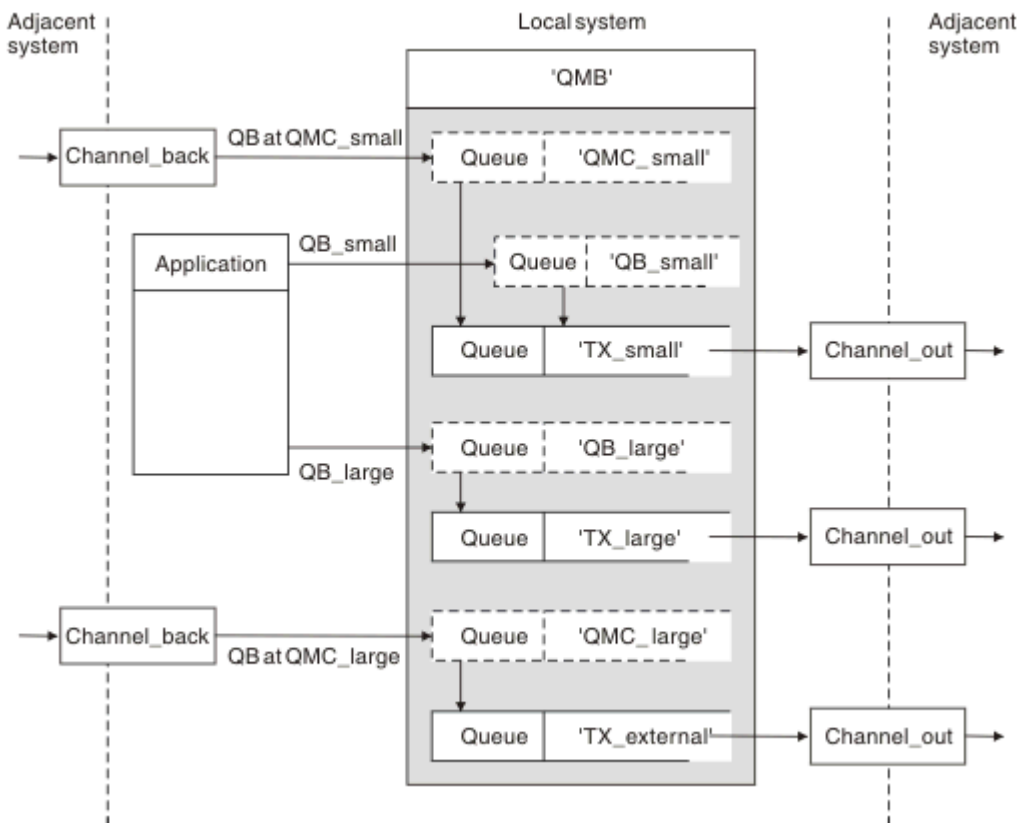


Abbildung 6. Nachrichtenflüsse trennen

In dem in [Abbildung 6](#) auf Seite 38 gezeigten Beispiel werden die beiden eingehenden Datenflüsse den Aliasnamen 'QMC_small' und 'QMC_large' des Aliaswarteschlangenmanagers verwendet. Sie stellen diese Flüsse mit einer WS-Manager-Aliasdefinition bereit, um diese Nachrichtenflüsse für den lokalen WS-Manager zu erfassen. Sie verfügen über eine Anwendung, die zwei ferne Warteschlangen adressieren soll, und Sie benötigen diese Nachrichtenflüsse getrennt voneinander. Sie stellen zwei Definitionen für ferne Warteschlangen bereit, die dieselbe Position angeben, 'QMC', aber geben Sie andere Übertragungswarteschlangen an. Diese Definition behält die Abläufe bei und ist am Ende nicht mehr erforderlich, da sie denselben Namen für den Zielwarteschlangenmanager in den Übertragungsheadern haben. Sie stellen Folgendes bereit:

- Die eingehenden Kanaldefinitionen
- Die beiden fernen Warteschlangendefinitionen QB_small und QB_large
- Die beiden WS-Manager-Aliasnamendefinitionen QMC_small und QMC_large
- Die drei sendenden Kanaldefinitionen
- Drei Übertragungswarteschlangen: TX_small, TX_large und TX_external

Koordination mit benachbarten Systemen

Wenn Sie einen WS-Manager-Aliasnamen verwenden, um einen separaten Nachrichtenfluss zu erstellen, müssen Sie diese Aktivität mit dem Systemadministrator am fernen Ende des Nachrichtenkanals koordinieren, um sicherzustellen, dass der entsprechende WS-Manager-Aliasname dort verfügbar ist.

Konzentration von Botschaften an verschiedene Standorte

Sie können Nachrichten, die für verschiedene Orte bestimmt sind, auf einen einzigen Kanal konzentrieren.

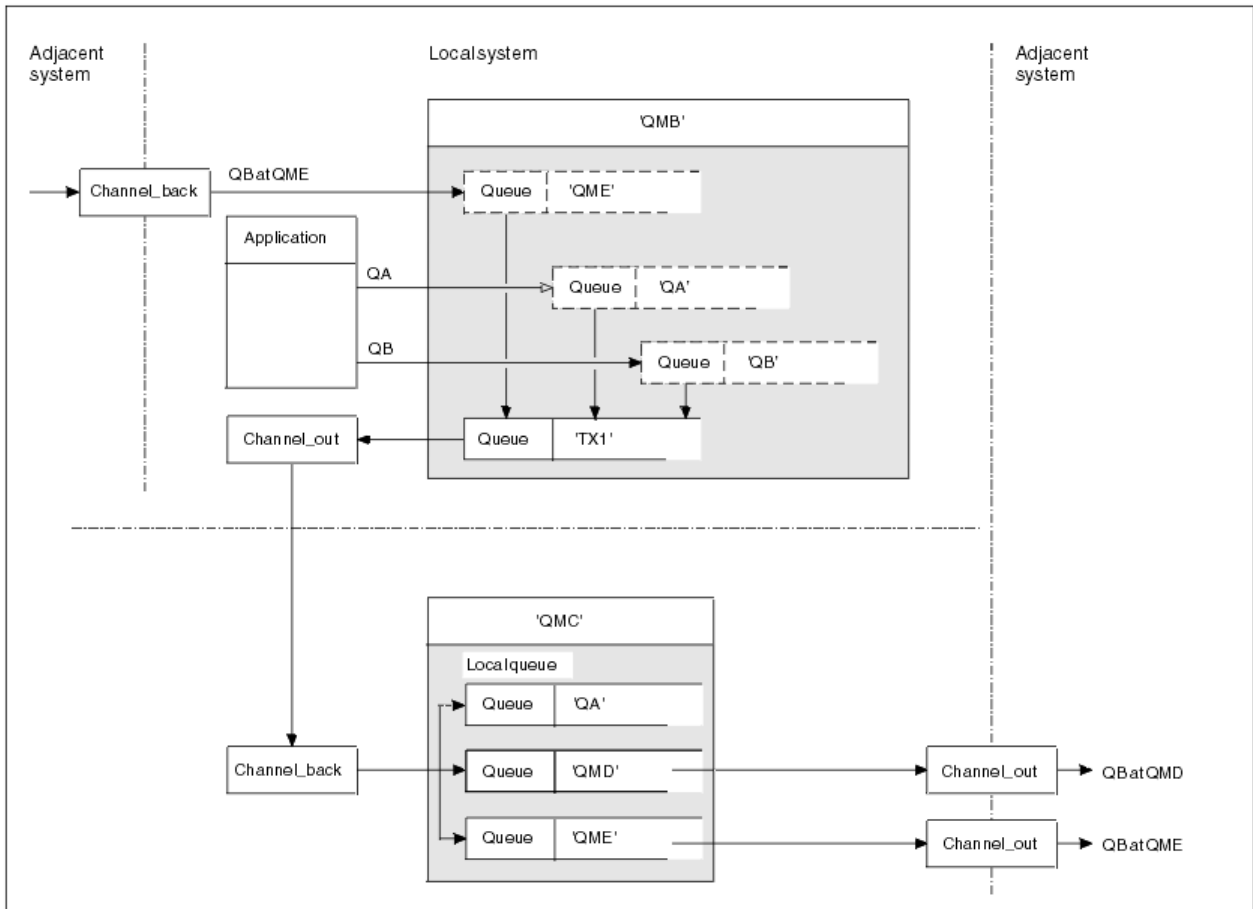


Abbildung 7. Kombinieren von Nachrichtenflüssen auf einem Kanal

Abbildung 7 auf Seite 39 veranschaulicht ein Verfahren zur verteilten Steuerung von Nachrichten, das für die Konzentration von Nachrichten bestimmt ist, die für verschiedene Positionen in einem Kanal bestimmt sind. Zwei Verwendungsmöglichkeiten wären denkbar:

- Nachrichtenverkehr durch ein Gateway konzentrieren
- Verwenden von Hochgeschwindigkeits-Highways zwischen Knoten

In diesem Beispiel werden Nachrichten aus verschiedenen Quellen, lokalen und benachbarten und mit unterschiedlichen Zielwarteschlangen und Warteschlangenmanagern über die Übertragungswarteschlange 'TX1' in den Warteschlangenmanager QMC übertragen. Der WS-Manager QMC stellt die Nachrichten entsprechend den Zielen bereit. Eine Übertragungswarteschlange 'QMD' wurde für die Weiterübertragung an Warteschlangenmanager QMD festgelegt. Eine andere Gruppe in eine Übertragungswarteschlange 'QME' für die Weiterleitung an Warteschlangenmanager QME gesetzt. Andere Nachrichten werden in die lokale Warteschlange 'QA' gestellt.

Sie müssen Folgendes angeben:

- Kanal- definitionen
- Übertragungswarteschlange TX1

- Definitionen ferner Warteschlangen:
 - QA mit 'QA bei QMC über TX1'
 - QB mit 'QB auf QMD bis TX1'
- Definition des WS-Manager-Aliasnamens
 - QME mit 'QME bis TX1'

Der ergänzende Administrator, der die Konfiguration von QMC konfiguriert, muss Folgendes bereitstellen:

- Kanaldefinition mit demselben Kanalnamen empfangen
- Übertragungswarteschlange QMD mit zugeordneter Sendekanaldefinition
- Übertragungswarteschlange QME mit zugeordneter Sendekanaldefinition
- Lokales Warteschlangenobjekt QA.

Nachrichtenflüsse zu einem anderen Ziel umwählen

Sie können die Zieladresse bestimmter Nachrichten mit Hilfe von WS-Manager-Aliasnamen und -Übertragungswarteschlangen neu definieren.

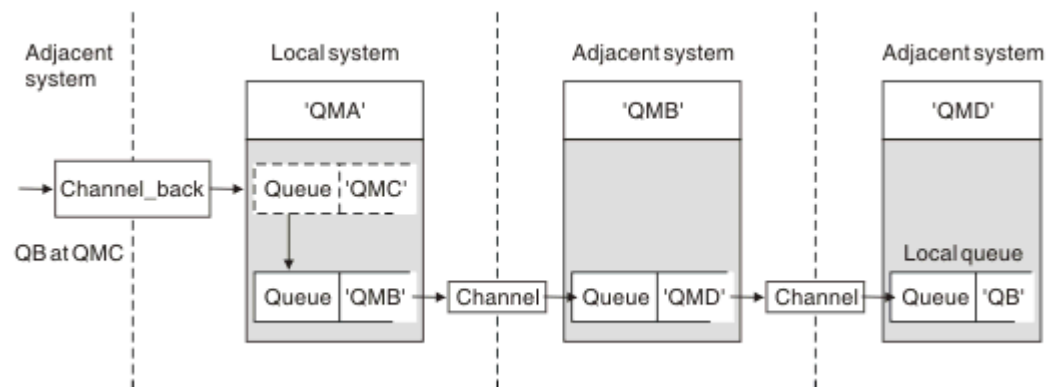


Abbildung 8. Nachrichtenströme zu einem anderen Ziel verwählen

In [Abbildung 8](#) auf Seite 40 ist dargestellt, wie Sie das Ziel bestimmter Nachrichten neu definieren können. Eingehende Nachrichten an QMA sind für 'QB at QMC' bestimmt. Sie kommen normalerweise bei QMA an und werden in eine Übertragungswarteschlange mit dem Namen QMC gestellt, die Teil eines Kanals zu QMC war. Die QMA muss die Nachrichten in QMD umleiten, aber die WSMD nur über WSMB erreichen. Diese Methode ist nützlich, wenn Sie einen Service von einer Position in eine andere versetzen müssen, und es den Subskribenten ermöglichen, Nachrichten temporär zu senden, bis sie an die neue Adresse angepasst wurden.

Die Methode, eingehende Nachrichten, die für einen bestimmten Warteschlangenmanager bestimmt sind, an einen anderen Warteschlangenmanager zu umleiten:

- Ein WS-Manager-Aliasname zum Ändern des Zielwarteschlangenmanagers in einen anderen Warteschlangenmanager und zum Auswählen einer Übertragungswarteschlange für das benachbarte System.
- Eine Übertragungswarteschlange, die dem benachbarten WS-Manager dient
- Eine Übertragungswarteschlange auf dem benachbarten Warteschlangenmanager für die Weiterleitung an den Zielwarteschlangenmanager.

Sie müssen Folgendes angeben:

- Channel_back-Definition
- WS-Manager-Aliasobjektdefinition QMC mit QB auf QMD über WSMB
- Channel_out-Definition
- Die zugeordnete Übertragungswarteschlange (WSMB)

Der ergänzende Administrator, der die Konfiguration von WSMB konfiguriert, muss Folgendes bereitstellen:

- Die entsprechende Channel_back-Definition
- Die Übertragungswarteschlange, QMD
- Die zugeordnete Kanaldefinition zu QMD

Sie können Aliasnamen in einer Clustering-Umgebung verwenden. Weitere Informationen finden Sie in [„WS-Manager-Aliasnamen und-Cluster“](#) auf Seite 280.

Nachrichten an eine Verteilerliste senden

Sie können einen einzigen MQPUT-Aufruf verwenden, damit eine Anwendung eine Nachricht an mehrere Ziele sendet.

In WebSphere MQ auf allen Plattformen außer z/OS kann eine Anwendung eine Nachricht mit einem einzigen MQPUT-Aufruf an mehrere Ziele senden. Sie können dies sowohl in einer Umgebung mit verteilter Warteschlange als auch in einer Clusterumgebung tun. Sie müssen die Ziele in einer Verteilerliste definieren, wie in [Verteilerlisten](#) beschrieben.

Nicht alle Warteschlangenmanagern unterstützen Verteilerlisten. Wenn ein MCA eine Verbindung mit einem Partner herstellt, bestimmt er, ob der Partner Verteilerlisten unterstützt und dementsprechend eine Markierung in der Übertragungswarteschlange setzt. Wenn eine Anwendung versucht, eine Nachricht zu senden, die für eine Verteilerliste bestimmt ist, der Partner jedoch Verteilerlisten nicht unterstützt, fängt der sendende MCA die Nachricht ab und stellt sie einmal für jedes geplante Ziel in die Übertragungswarteschlange.

Durch einen empfangenden MCA wird sichergestellt, dass Nachrichten, die an eine Verteilerliste gesendet werden, sicher an allen vorgesehenen Zieladressen empfangen werden. Wenn ein Ziel fehlschlägt, stellt der MCA fest, welche Fehler aufgetreten sind. Anschließend kann sie Ausnahmebedingungenberichte für sie generieren und versuchen, die Nachrichten erneut an sie zu senden.

Empfangswarteschlange für Antworten

Sie können eine vollständige ferne Warteschlangenverarbeitungsschleife mit Hilfe einer Warteschlange für Antwortantworten erstellen.

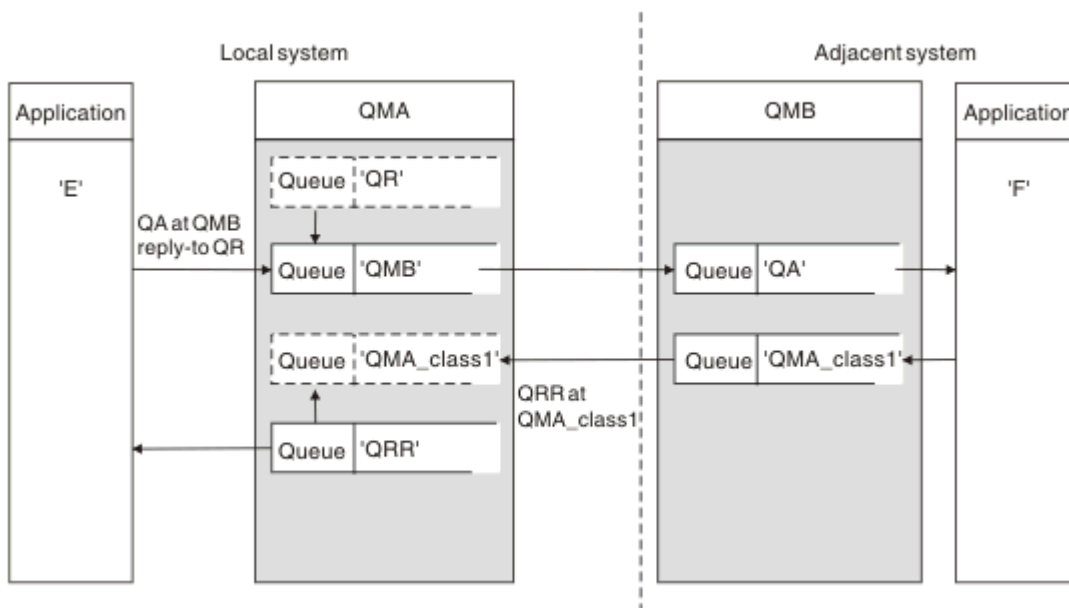


Abbildung 9. Name der Warteschlange für Antwortwarteschlangen beim PUT-Aufruf

In [Abbildung 9 auf Seite 41](#) wird eine vollständige Verarbeitungsschleife der fernen Warteschlange mit Hilfe einer Warteschlange für Antwortantworten angezeigt. Diese Schleife gilt sowohl in einer Umgebung mit verteilter Warteschlange als auch in einer Clustering-Umgebung. Die Details sind wie in [Tabelle 7 auf Seite 50](#) dargestellt.

Die Anwendung öffnet QA auf WSMB und reiht Nachrichten in diese Warteschlange ein. Die Nachrichten erhalten einen Antwortwarteschlangennamen von QR, ohne dass der Name des Warteschlangenmanagers angegeben wurde. WS-Manager QMA sucht das Antwortwarteschlangenobjekt QR und extrahiert daraus den Aliasnamen QRR und den WS-Manager-Namen QMA_class1. Diese Namen werden in die Antwortfelder der Nachrichten gestellt.

Antwortnachrichten von Anwendungen auf WSMB werden an QRR in QMA_class1 adressiert. Die WS-Manager-Aliasnamendefinition QMA_class1 wird vom WS-Manager verwendet, um die Nachrichten an sich selbst und die Warteschlange QRR zu fließen.

In diesem Szenario wird die Art und Weise dargestellt, in der Sie Anwendungen die Möglichkeit geben, eine Serviceklasse für Antwortnachrichten auszuwählen. Die Klasse wird von der Übertragungswarteschlange QMA_class1 auf WSMB zusammen mit der WS-Manager-Aliasdefinition QMA_klasse1 bei QMA implementiert. Auf diese Weise können Sie die Antwortwarteschlange einer Anwendung ändern, so dass die Abläufe getrennt werden, ohne dass die Anwendung einbezogen wird. Die Anwendung wählt immer QR für diese bestimmte Serviceklasse aus. Sie haben die Möglichkeit, die Serviceklasse mit der Antwortwarteschlange-QR-Definition zu ändern.

Sie müssen Folgendes erstellen:

- Antwortwarteschlangendefinition QR
- Übertragungswarteschlangenobjekt (WSMB)
- Channel_out-Definition
- Channel_back-Definition
- WS-Manager-Aliasnamendefinition QMA_class1
- Lokales Warteschlangenobjekt QRR, wenn es nicht vorhanden ist

Der ergänzende Administrator auf dem benachbarten System muss Folgendes erstellen:

- Kanaldefinition wird empfangen
- Übertragungswarteschlangenobjekt QMA_class1
- Zugeordneter Sendechannel
- Lokales Warteschlangenobjekt QA.

Ihre Anwendungsprogramme verwenden:

- Name der Reply-to-Warteschlange QR in Anrufen
- Warteschlangennamen QRR in get-Aufrufen

Auf diese Weise können Sie die Serviceklasse nach Bedarf ändern, ohne dass die Anwendung einbezogen wird. Sie ändern den Antwortalias 'QR' zusammen mit der Übertragungswarteschlange 'QMA_class1' und dem WS-Manager-Aliasnamen 'QMA_klasse1'.

Wenn beim Einlegen der Nachricht in die Warteschlange kein Objekt "reply-to alias" gefunden wird, wird der Name des lokalen Warteschlangenmanagers in das Feld für den Namen des leeren Antwortwarteschlangenmanagers (Name des Warteschlangenmanagers) eingefügt. Der Name der Empfangswarteschlange für Antworten bleibt unverändert.

Einschränkung für Namensauflösung

Da die Namensauflösung für die Warteschlange für Antwortnachrichten bei 'QMA' ausgeführt wurde, als die ursprüngliche Nachricht gestellt wurde, ist keine weitere Namensauflösung bei 'WSMB' zulässig. Die Nachricht wird mit dem physischen Namen der Empfangswarteschlange für Antworten von der Anwendung "Antwort" in die Warteschlange gestellt.

Die Anwendungen müssen sich bewusst sein, dass der Name, den sie für die Empfangswarteschlange für Antworten verwenden, sich von dem Namen der tatsächlichen Warteschlange unterscheidet, in der die Rückgabennachrichten zu finden sind.

Wenn beispielsweise zwei Serviceklassen für die Verwendung von Anwendungen mit den Aliasnamen 'C1_alias' für Antwortwarteschlangen und 'C2_alias' zur Verfügung gestellt werden, verwenden die Anwendungen diese Namen als Antwort-in-Warteschlange-Namen in den Nachrichteneinträgen. Tatsächlich erwarten die Anwendungen jedoch, dass Nachrichten in den Warteschlangen 'C1' für 'C1_alias' und 'C2' für 'C2_alias' angezeigt werden.

Eine Anwendung kann jedoch in der Warteschlange für Antwortnachrichten einen Anfragenaufruf vornehmen, um den Namen der realen Warteschlange, die zum Abrufen der Antwortnachrichten verwendet werden muss, selbst zu überprüfen.

Zugehörige Konzepte

„Vorgehensweise zum Erstellen von Warteschlangenmanagern und Antwortaliasnamen“ auf Seite 31
In diesem Thema werden die drei Möglichkeiten erläutert, wie Sie eine Definition einer fernen Warteschlange erstellen können.

„Beispiel für Antwortwarteschlangenalias“ auf Seite 43

Dieses Beispiel veranschaulicht die Verwendung eines Antwortalias für Antworten, um eine andere Route (Übertragungswarteschlange) für zurückgegebene Nachrichten auszuwählen. Für die Verwendung dieser Funktion ist es erforderlich, dass der Name der Empfangswarteschlange für Antworten in Zusammenarbeit mit den Anwendungen geändert wird.

„Funktionsweise des Beispiels“ auf Seite 45

Eine Erläuterung des Beispiels und die Verwendung des Aliasnamens der Empfangswarteschlange für Antworten durch den Warteschlangenmanager.

„Durchlauf der Warteschlange für Antwortwarteschlangen“ auf Seite 46

Ein Walkthrough des Prozesses von einer Anwendung, die eine Nachricht in eine ferne Warteschlange einreihen soll, bis zu derselben Anwendung, die die Antwortnachricht aus der Warteschlange für Aliasantwortnachrichten entfernt.

Beispiel für Antwortwarteschlangenalias

Dieses Beispiel veranschaulicht die Verwendung eines Antwortalias für Antworten, um eine andere Route (Übertragungswarteschlange) für zurückgegebene Nachrichten auszuwählen. Für die Verwendung dieser Funktion ist es erforderlich, dass der Name der Empfangswarteschlange für Antworten in Zusammenarbeit mit den Anwendungen geändert wird.

Wie in Abbildung 10 auf Seite 44 gezeigt, muss die Rückgaberoute für die Antwortnachrichten, einschließlich der Übertragungswarteschlange, des Kanals und des Aliasnamens des Warteschlangenmanagers, verfügbar sein.

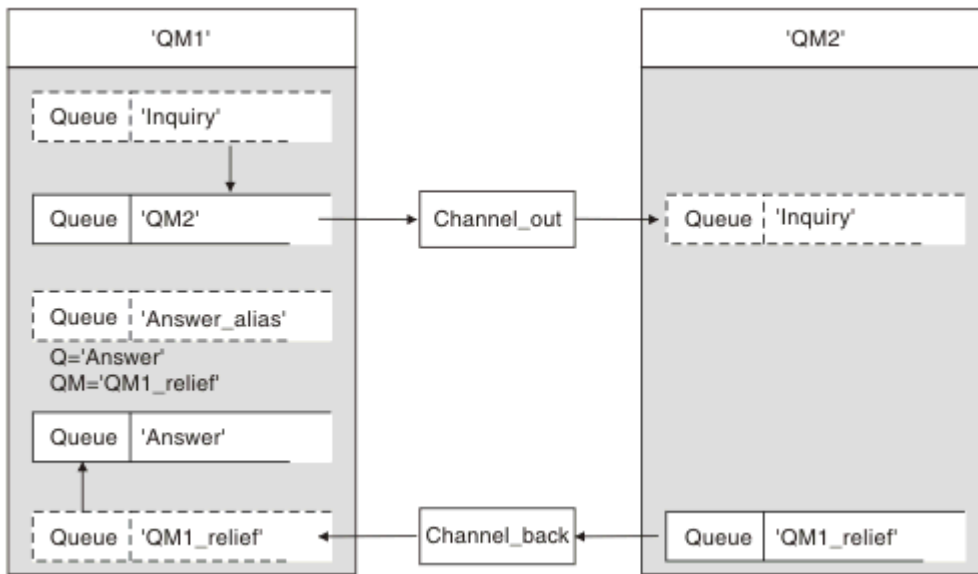


Abbildung 10. Beispiel für Antwortwarteschlangenalias

Dieses Beispiel gilt für Requesteranwendungen in 'QM1', die Nachrichten an Serveranwendungen in 'QM2' senden. Die Nachrichten auf dem Server sollen über einen alternativen Kanal unter Verwendung der Übertragungswarteschlange 'QM1_relief' zurückgegeben werden. (Der Standardrückgabekanal wird mit einer Übertragungswarteschlange 'QM1' bereitgestellt).

Der Aliasname der Empfangswarteschlange für Antworten ist eine bestimmte Verwendung der Definition der fernen Warteschlange mit dem Namen 'Answer_alias'. Anwendungen auf WSM1 enthalten diesen Namen, 'Answer_alias', im Feld 'reply-to' aller Nachrichten, die sie in die Warteschlange 'Abfrage' stellen.

Die Antwortwarteschlangendefinition 'Answer_alias' wird als 'Answer at QM1_relief' definiert. Anwendungen auf WSM1 erwarten, dass ihre Antworten in der lokalen Warteschlange mit dem Namen 'Answer' angezeigt werden.

Serveranwendungen bei QM2 verwenden das Antwortfeld der empfangenen Nachrichten, um die Namen der Warteschlange und des Warteschlangenmanagers für die Antwortnachrichten an den Requester auf QM1 zu erhalten.

Definitionen, die in diesem Beispiel auf QM1 verwendet werden

Der Systemadministrator für WebSphere MQ auf QM1 muss sicherstellen, dass die Antwortwarteschlange 'Answer' zusammen mit den anderen Objekten erstellt wird. Der Name des WS-Manager-Aliasnamens, der mit einem Stern (*) gekennzeichnet ist, muss mit dem Namen des Warteschlangenmanagers in der Aliasdefinition für die Antwortwarteschlange übereinstimmen, die ebenfalls mit einem Stern (*) markiert ist.

Objekt	Definition								
Lokale Übertragungswarteschlange	QM2								
Definition der fernen Warteschlange	<table border="0"> <tr> <td>Objektname</td> <td>Anfrage</td> </tr> <tr> <td>Name des fernen Warteschlangenmanagers</td> <td>QM2</td> </tr> <tr> <td>Name der fernen Warteschlange</td> <td>Anfrage</td> </tr> <tr> <td>Name der Übertragungswarteschlange</td> <td>WSM2 (DEFAULT)</td> </tr> </table>	Objektname	Anfrage	Name des fernen Warteschlangenmanagers	QM2	Name der fernen Warteschlange	Anfrage	Name der Übertragungswarteschlange	WSM2 (DEFAULT)
Objektname	Anfrage								
Name des fernen Warteschlangenmanagers	QM2								
Name der fernen Warteschlange	Anfrage								
Name der Übertragungswarteschlange	WSM2 (DEFAULT)								

Objekt	Definition	
Aliasname des Warteschlangenmanagers	Objektname	QM1_relief *
	Name des Warteschlangenmanagers	QM1
	Warteschlangenname	(leer)
Aliasname der Antwortwarteschlange	Objektname	Antwortaliasname
	Name des fernen Warteschlangenmanagers	QM1_relief *
	Name der fernen Warteschlange	Antworte

Definition des Einreihens auf QM1

Anwendungen füllen die Antwortfelder mit dem Aliasnamen der Warteschlange für Antwortwarteschlangen aus und lassen das Feld für den Namen des WS-Managers leer.

Feld	Inhalt
Warteschlangenname	Anfrage
Name des Warteschlangenmanagers	(leer)
Name der Empfangswarteschlange für Antworten	Antwortaliasname
Warteschlangenmanager für Antwortwarteschlange	(leer)

Definitionen, die in diesem Beispiel auf WSM2 verwendet werden

Der WebSphere MQ auf QM2 muss sicherstellen, dass die lokale Warteschlange für die eingehenden Nachrichten vorhanden ist und dass die ordnungsgemäß benannte Übertragungswarteschlange für die Antwortnachrichten verfügbar ist.

Objekt	Definition
Lokale Warteschlange	Anfrage
Übertragungswarteschlange	QM1_relief

Definition in WSM2 definieren

Anwendungen auf QM2 rufen den Namen der Empfangswarteschlange für Antworten und den Namen des WS-Managers aus der ursprünglichen Nachricht ab und verwenden sie, wenn sie die Antwortnachricht in die Warteschlange für Antwortnachrichten einreihen.

Feld	Inhalt
Warteschlangenname	Antworte
Name des Warteschlangenmanagers	QM1_relief

Funktionsweise des Beispiels

Eine Erläuterung des Beispiels und die Verwendung des Aliasnamens der Empfangswarteschlange für Antworten durch den Warteschlangenmanager.

In diesem Beispiel verwenden Requester-Anwendungen auf WSM1 immer 'Answer_alias' als Antwort-Warteschlange in das relevante Feld des put-Aufrufs. Sie rufen ihre Nachrichten immer aus der Warteschlange mit dem Namen 'Answer' ab.

Die Definitionen der Warteschlange für Antwortwarteschlangen sind für die Verwendung durch den QM1-Systemadministrator verfügbar, um den Namen der Antwort-Warteschlange 'Answer' und der Rückgaberoute 'QM1_relief' zu ändern.

Die Änderung des Warteschlangennamens 'Answer' ist in der Regel nicht hilfreich, da die Anwendungen QM1 ihre Antworten in dieser Warteschlange erwarten. Der Systemverwalter QM1 kann jedoch die Rückgaberoute (Serviceklasse) je nach Bedarf ändern.

Verwendung des Aliasnamens "reply-to queue" des Warteschlangenmanagers

Warteschlangenmanager QM1 ruft die Definitionen aus dem Aliasnamen der Empfangswarteschlange für Antworten ab, wenn der Name der Antwortwarteschlange, die in den von der Anwendung aufgerufenen Aufruf eingeschlossen ist, mit dem Aliasnamen für die Antwortwarteschlange identisch ist, und der Teil des Warteschlangenmanagers leer ist.

Der WS-Manager ersetzt den Namen der Empfangswarteschlange für Antworten in dem Aufruf der Warteschlange mit dem Namen der Warteschlange aus der Definition. Er ersetzt den leeren WS-Manager-Namen im put-Aufruf mit dem Namen des WS-Managers aus der Definition.

Diese Namen werden mit der Nachricht im Nachrichtendeskriptor übertragen.

Tabelle 4. Aliasname der Antwortwarteschlange		
Name des Felds	Put-Aufruf	Übertragungsheader
Name der Empfangswarteschlange für Antworten	Antwortaliasname	Antworten
Name des Antwortwarteschlangenmanagers	(leer)	QM1_relief

Durchlauf der Warteschlange für Antwortwarteschlangen

Ein Walkthrough des Prozesses von einer Anwendung, die eine Nachricht in eine ferne Warteschlange einreihen soll, bis zu derselben Anwendung, die die Antwortnachricht aus der Warteschlange für Aliasantwortnachrichten entfernt.

Um dieses Beispiel zu vervollständigen, lassen Sie uns den Prozess betrachten.

1. Die Anwendung öffnet eine Warteschlange mit dem Namen 'Abfrage' und reiht Nachrichten in diese Warteschlange ein. Die Anwendung setzt die Felder "reply-to" des Nachrichtendeskriptors auf:

Name der Empfangswarteschlange für Antworten	Antwortaliasname
Name des Antwortwarteschlangenmanagers	(leer)

2. Der Warteschlangenmanager 'QM1' antwortet auf den Namen des leeren Warteschlangenmanagers, indem er die Definition einer fernen Warteschlangendefinition mit dem Namen 'Answer_alias' (Antwortalias) überprüft. Wenn keine gefunden wird, stellt der Warteschlangenmanager seinen eigenen Namen 'QM1' in das Feld für den Antwortwarteschlangenmanager des Nachrichtendeskriptors.
3. Wenn der Warteschlangenmanager eine Definition einer fernen Warteschlange mit dem Namen 'Answer_alias' findet, extrahiert er den Warteschlangennamen und die Namen des WS-Managers aus der Definition (Warteschlangennamen = 'Answer' und WS-Manager-Name = 'QM1_relief'). Anschließend werden sie in die Antwortfelder des Nachrichtendeskriptors gestellt.
4. Der Warteschlangenmanager 'QM1' verwendet die Definition 'Inquiry' für die ferne Warteschlange, um festzustellen, ob sich die beabsichtigte Zielwarteschlange im Warteschlangenmanager 'QM2' befindet, und die Nachricht wird in die Übertragungswarteschlange 'QM2' gestellt. 'QM2' ist der Name

der Standardübertragungswarteschlange für Nachrichten, die für Warteschlangen beim Warteschlangenmanager 'QM2' bestimmt sind.

5. Wenn der WS-Manager 'QM1' die Nachricht in die Übertragungswarteschlange einreicht, wird der Nachricht ein Übertragungsheader hinzugefügt. Dieser Header enthält den Namen der Zielwarteschlange, 'Inquiry' und den Zielwarteschlangenmanager 'QM2'.
6. Die Nachricht wird beim WS-Manager 'QM2' empfangen und in die lokale Warteschlange 'Inquiry' gestellt.
7. Eine Anwendung ruft die Nachricht aus dieser Warteschlange ab und verarbeitet die Nachricht. Die Anwendung bereitet eine Antwortnachricht vor und reiht diese Antwortnachricht aus dem Nachrichtendeskriptor der ursprünglichen Nachricht in den Namen der Antwort auf die Antwort ein:

Name der Empfangswarteschlange für Antworten	Antworten
Name des Antwortwarteschlangenmanagers	QM1_relief

8. WS-Manager 'QM2' führt den Befehl put aus. Wenn der Warteschlangenmanagername 'QM1_relief' ein ferner Warteschlangenmanager ist, wird die Nachricht in die Übertragungswarteschlange mit dem gleichen Namen 'QM1_relief' versetzt. Die Nachricht erhält einen Übertragungsheader, der den Namen der Zielwarteschlange, 'Answer' und den Zielwarteschlangenmanager 'QM1_relief' enthält.
9. Die Nachricht wird an WS-Manager 'QM1' übertragen. Der Warteschlangenmanager erkennt, dass der WS-Manager-Name 'QM1_relief' ein Aliasname ist. Er extrahiert aus der Aliasdefinition 'QM1_relief' den Namen des physischen Warteschlangenmanagers 'QM1'.
10. WS-Manager 'QM1' reiht dann die Nachricht in den Namen der Warteschlange ein, die im Übertragungsheader 'Answer' enthalten ist.
11. Die Anwendung extrahiert ihre Antwortnachricht aus der Warteschlange 'Answer'.

Überlegungen zum Netzbetrieb

In einer Umgebung mit verteilten Warteschlangensteuerung, da Nachrichtenziele mit nur einem Warteschlangennamen und einem Warteschlangenmanagernamen adressiert werden, gelten bestimmte Regeln.

1. Wenn der Name des Warteschlangenmanagers angegeben wird und der Name sich von dem Namen des lokalen WS-Managers unterscheidet:
 - Es muss eine Übertragungswarteschlange mit dem gleichen Namen verfügbar sein. Diese Übertragungswarteschlange muss Teil eines Nachrichtenkanals sein, der Nachrichten in einen anderen WS-Manager versetzt, oder
 - Eine WS-Manager-Aliasnamendefinition muss vorhanden sein, um den Namen des Warteschlangenmanagers in denselben Namen oder einen anderen WS-Manager-Namen und eine optionale Übertragungswarteschlange aufzulösen.
 - Wenn der Name der Übertragungswarteschlange nicht aufgelöst werden kann und eine Standardübertragungswarteschlange definiert wurde, wird die Standardübertragungswarteschlange verwendet.
2. Wird nur der Warteschlangename angegeben, muss eine Warteschlange eines beliebigen Typs mit demselben Namen auf dem lokalen WS-Manager verfügbar sein. Bei dieser Warteschlange kann es sich um eine Definition einer fernen Warteschlange handeln, die in eine Übertragungswarteschlange in einen benachbarten Warteschlangenmanager, einen WS-Manager-Namen und eine optionale Übertragungswarteschlange aufgelöst wird.

Informationen zur Funktionsweise in einer Clusterumgebung finden Sie in den entsprechenden Abschnitten im Abschnitt [Funktionsweise von Clustern](#) der Produktdokumentation.

Betrachten Sie das Szenario eines Nachrichtenkanals, der Nachrichten von einem WS-Manager in einen anderen in einer Umgebung mit verteilten Warteschlangen versetzt.

Die Nachrichten, die verschoben werden, stammen von einem anderen Warteschlangenmanager im Netz, und einige Nachrichten, die einen unbekanntem Warteschlangenmanagernamen als Ziel haben, kommen

möglicherweise an. Dieses Problem kann auftreten, wenn ein WS-Manager-Name geändert wurde oder beispielsweise aus dem System entfernt wurde.

Das Kanalprogramm erkennt diese Situation, wenn es keine Übertragungswarteschlange für diese Nachrichten finden kann, und stellt die Nachrichten in die Warteschlange für nicht zustellbare Nachrichten (dead-letter). Es liegt in Ihrer Verantwortung, nach diesen Nachrichten zu suchen und die Weiterleitung an die richtige Zieladresse zu veranlassen. Alternativ können Sie sie an den Absender zurückgeben, wo der Urheber ermittelt werden kann.

Ausnahmebedingungsberichte werden unter diesen Umständen generiert, wenn Berichtsnachrichten in der ursprünglichen Nachricht angefordert wurden.

Namensauflösungskonvention

Die Namensauflösung, die die Identität der Zielwarteschlange (d. a. logisch in eine Änderung des physischen Namens) ändert, tritt nur einmal und nur im Ursprungswarteschlangenmanager auf.

Die nachfolgende Verwendung der verschiedenen Alias-Möglichkeiten darf nur bei der Trennung und Kombination von Nachrichtenflüssen verwendet werden.

Routing zurückgeben

Nachrichten können eine Rückgabeadresse in Form des Namens einer Warteschlange und eines Warteschlangenmanagers enthalten. Dieses Formular für die Rückgabeadresse kann sowohl in einer Umgebung mit verteilter Warteschlange als auch in einer Clusterumgebung verwendet werden.

Diese Adresse wird normalerweise von der Anwendung angegeben, die die Nachricht erstellt. Sie kann von jeder Anwendung geändert werden, die dann die Nachricht verarbeitet, einschließlich Benutzerexitanwendungen.

Unabhängig von der Quelle dieser Adresse kann jede Anwendung, die die Nachricht bearbeitet, diese Adresse verwenden, um Antworten, Status oder Berichtsnachrichten an die Ursprungsanwendung zurückzugeben.

Die Art und Weise, wie diese Antwortnachrichten weitergeleitet werden, unterscheidet sich nicht von der Art und Weise, wie die ursprüngliche Nachricht weitergeleitet wird. Sie müssen wissen, dass die Nachrichtenflüsse, die Sie zu anderen Warteschlangenmanagern erstellen, entsprechende Rückkehrabläufe benötigen.

Konflikte mit physischen Namen

Der Name der Zielwarteschlange für die Antwort auf die Warteschlange wurde in den Namen der physischen Warteschlange des ursprünglichen Warteschlangenmanagers aufgelöst. Er darf beim antwortenden Warteschlangenmanager nicht erneut aufgelöst werden.

Es ist eine wahrscheinliche Möglichkeit für Namenskonfliktprobleme, die nur durch eine netzwerkweite Vereinbarung über Namen von physischen und logischen Warteschlangen verhindert werden können.

Übersetzungen von Warteschlangennamen verwalten

Wenn Sie eine Warteschlangenmanageraliasdefinition oder eine Definition einer fernen Warteschlange erstellen, wird die Namensauflösung für jede Nachricht ausgeführt, die diesen Namen trägt. Diese Situation muss verwaltet werden.

Diese Beschreibung wird für Anwendungsdesigner und Kanalplaner mit einem einzelnen System bereitgestellt, das Nachrichtenkanäle zu benachbarten Systemen hat. Sie nimmt eine lokale Ansicht der Kanalplanung und -steuerung vor.

Wenn Sie eine Warteschlangenmanageraliasdefinition oder eine Definition einer fernen Warteschlange erstellen, wird die Namensauflösung für jede Nachricht ausgeführt, die diesen Namen trägt, unabhängig von der Quelle der Nachricht. Um diese Situation zu überwachen, die möglicherweise eine große Anzahl von Warteschlangen in einem WS-Manager-Netz enthält, behalten Sie die folgenden Tabellen bei:

- Die Namen der Quellenwarteschlangen und der Quellenwarteschlangenmanager in Bezug auf aufgelöste Warteschlangennamen, aufgelöste Warteschlangenmanagernamen und aufgelöste Übertragungswarteschlangennamen mit der Methode der Auflösung.
- Die Namen der Quellenwarteschlangen in Bezug auf:
 - Namen der aufgelösten Zielwarteschlangen
 - Namen von Zielwarteschlangenmanagern aufgelöst
 - Übertragungswarteschlangen
 - Nachrichtenkanalnamen
 - Angrenzende Systemnamen
 - Namen der Antwortwarteschlange

Anmerkung: Die Verwendung des Begriffs *Quelle* in diesem Kontext bezieht sich auf den Warteschlangennamen oder den Namen des Warteschlangenmanagers, der von der Anwendung bereitgestellt wird, oder ein Kanalprogramm, wenn eine Warteschlange zum Einreihen von Nachrichten geöffnet wird.

Ein Beispiel für jede dieser Tabellen finden Sie in [Tabelle 5 auf Seite 49](#), [Tabelle 6 auf Seite 49](#) und [Tabelle 7 auf Seite 50](#).

Die Namen in diesen Tabellen werden aus den Beispielen in diesem Abschnitt abgeleitet, und diese Tabelle ist nicht als praktisches Beispiel für die Warteschlangennamensauflösung in einem Knoten gedacht.

Tabelle 5. Warteschlangennamensauflösung auf WS-Manager QMA

Quellenwarteschlange angegeben, wenn Warteschlange geöffnet wird	Der Quellenwarteschlangenmanager wurde beim Öffnen der Warteschlange angegeben.	Aufgelöster Warteschlangennamen	Aufgelöster Name des Warteschlangenmanagers	Name der aufgelösten Übertragungswarteschlange	Auflösungstyp
QA_norm	-	QA_norm	WSMB	WSMB	Ferne Warteschlange
(beliebig)	WSMB	-	-	WSMB	(keine)
QA_norm	-	QA_norm	WSMB	TX1	Ferne Warteschlange
QB	QMC	QB	WSMD	WSMB	Aliasname des Warteschlangenmanagers

Tabelle 6. Warteschlangennamensauflösung auf WS-Manager WSMB

Quellenwarteschlange angegeben, wenn Warteschlange geöffnet wird	Der Quellenwarteschlangenmanager wurde beim Öffnen der Warteschlange angegeben.	Aufgelöster Warteschlangennamen	Aufgelöster Name des Warteschlangenmanagers	Name der aufgelösten Übertragungswarteschlange	Auflösungstyp
QA_norm	-	QA_norm	WSMB	-	(keine)
QA_norm	WSMB	QA_norm	WSMB	-	(keine)
QA_norm	QMB_PRIORITÄT	QA_norm	WSMB	-	Aliasname des Warteschlangenmanagers

Tabelle 6. Warteschlangennamensauflösung auf WS-Manager WSMB (Forts.)

Quellenwarteschlange angegeben, wenn Warteschlange geöffnet wird	Der Quellenwarteschlangenmanager wurde beim Öffnen der Warteschlange angegeben.	Aufgelöster Warteschlangenname	Aufgelöster Name des Warteschlangenmanagers	Name der aufgelösten Übertragungswarteschlange	Auflösungstyp
(beliebig)	QMC	(beliebig)	QMC	QMC	(keine)
(beliebig)	QMD_norm	(beliebig)	QMD_norm	TX1	Aliasname des Warteschlangenmanagers
(beliebig)	QMD_PRIORITÄT	(beliebig)	QMD_PRIORITÄT	QMD_schnell	Aliasname des Warteschlangenmanagers
(beliebig)	QMC_small	(beliebig)	QMC_small	TX_klein	Aliasname des Warteschlangenmanagers
(beliebig)	QMC_large	(beliebig)	QMC_large	TX_extern	Aliasname des Warteschlangenmanagers
QB_small	QMC	QB_small	QMC	TX_klein	Ferne Warteschlange
QB_large	QMC	QB_large	QMC	TX_groß	Ferne Warteschlange
(beliebig)	QME	(beliebig)	QME	TX1	Aliasname des Warteschlangenmanagers
QA	QMC	QA	QMC	TX1	Ferne Warteschlange
QB	WSMD	QB	WSMD	TX1	Ferne Warteschlange

Tabelle 7. Umsetzung der Namen der Warteschlange für Antwortwarteschlangen auf WS-Manager QMA

Anwendungsdesign		Antwort-Aliasdefinition	
Lokaler QMGR	Warteschlangenname für Nachrichten	Aliasname der Antwort-Warteschlange	Redefined to
QMA	QRR	QR	QRR bei QMA_class1

Numerierung der Kanalnachrichtenfolge

Der Kanal verwendet Folge-nummern, um sicherzustellen, dass Nachrichten zugestellt, ohne Duplizierung zugestellt und in derselben Reihenfolge gespeichert werden, in der sie aus der Übertragungswarteschlange entnommen wurden.

Die Folgenummer wird am sendenden Ende des Kanals generiert und vor der Verwendung um eins erhöht, was bedeutet, dass die aktuelle Folgenummer die Nummer der letzten gesendeten Nachricht ist. Diese Informationen können mit DISPLAY CHSTATUS angezeigt werden (siehe [WebSphere MQ-Scriptbefehlsre-](#)

ferenz (MQSC)). Die Folgenummer und eine Kennung, die als LUWID bezeichnet wird, werden im persistenten Speicher für die letzte Nachricht gespeichert, die in einem Stapel übertragen wurde. Diese Werte werden beim Kanalstart verwendet, um sicherzustellen, dass beide Enden des Links übereinstimmen, welche Nachrichten erfolgreich übertragen wurden.

Sequenzielles Abrufen von Nachrichten

Wenn eine Anwendung eine Folge von Nachrichten in dieselbe Zielwarteschlange einreicht, können diese Nachrichten in Folge durch eine **Single**-Anwendung mit einer Folge von MQGET-Operationen abgerufen werden, wenn die folgenden Bedingungen erfüllt sind:

- Alle put-Anforderungen wurden von derselben Anwendung ausgeführt.
- Alle put-Anforderungen waren entweder von derselben Arbeitseinheit oder alle gestellten Anforderungen wurden außerhalb einer UO- Unit gestellt.
- Die Nachrichten haben alle dieselbe Priorität.
- Die Nachrichten weisen alle dieselbe Persistenz auf.
- Bei der fernen Warteschlangensteuerung ist die Konfiguration so, dass es nur einen Pfad von der Anwendung gibt, die die put-Anforderung stellt, über ihren Warteschlangenmanager über die übergreifende Kommunikation mit dem Zielwarteschlangenmanager und der Zielwarteschlange.
- Die Nachrichten werden nicht in eine Warteschlange für dead-letter gestellt (z. B. wenn eine Warteschlange temporär voll ist).
- Die Anwendung, die die Nachricht erhält, ändert nicht absichtlich die Reihenfolge des Abrufs, z. B. durch die Angabe eines bestimmten *MsgId* oder *CorrelId* oder durch die Verwendung von Nachrichtenprioritäten.
- Es werden nur eine Anwendung get-Operationen ausgeführt, um die Nachrichten aus der Zielwarteschlange abzurufen. Wenn mehr als eine Anwendung vorhanden ist, müssen diese Anwendungen so konzipiert sein, dass sie alle Nachrichten in jeder Sequenz, die von einer sendenden Anwendung gestellt wird, abrufen können.

Anmerkung: Nachrichten von anderen Aufgaben und Arbeitseinheiten können mit der Sequenz interspergt werden, selbst wenn die Sequenz aus einer einzigen Arbeitseinheit eingestellt wurde.

Wenn diese Bedingungen nicht erfüllt werden können und die Reihenfolge der Nachrichten in der Zielwarteschlange wichtig ist, kann die Anwendung so codiert werden, dass sie ihre eigene Nachrichtenfolgenummer als Teil der Nachricht verwendet, um die Reihenfolge der Nachrichten zu gewährleisten.

Sequenz des Abrufs von schnellen, nicht persistenten Nachrichten

Nicht persistente Nachrichten in einem schnellen Kanal können persistente Nachrichten auf demselben Kanal überdauern und so aus der Reihenfolge kommen. Der empfangende MCA reiht die nicht persistenten Nachrichten sofort in die Zielwarteschlange ein und macht sie sichtbar. Persistente Nachrichten werden bis zum nächsten Synchronisationspunkt nicht sichtbar gemacht.

Loopback-Tests

Loopback-Tests ist ein Verfahren auf Nicht- z/OS -Plattformen, mit dem Sie eine Kommunikationsverbindung testen können, ohne tatsächlich eine Verbindung zu einer anderen Maschine herzustellen.

Sie haben eine Verbindung zwischen zwei WS-Managern eingerichtet, als ob sie sich auf separaten Maschinen befinden, aber Sie testen die Verbindung, indem Sie eine Schleife zu einem anderen Prozess auf derselben Maschine zurückführen. Dieses Verfahren bedeutet, dass Sie Ihren Kommunikationscode testen können, ohne dass ein aktives Netz erforderlich ist.

Die Art und Weise, in der Sie dies tun, hängt davon ab, welche Produkte und Protokolle Sie verwenden.

Auf Systemen mit Windows können Sie den Loopback-Adapter verwenden.

Weitere Informationen finden Sie in der Dokumentation zu den Produkten, die Sie verwenden.

Trace-und Aktivitätsaufzeichnung weiterleiten

Sie können die Weiterleitung einer Nachricht durch eine Reihe von Warteschlangenmanagern auf zwei Arten bestätigen.

Sie können die WebSphere MQ -Anwendung zur Routenanzeige, die über den Steuerbefehl `dspmqrtever` verfügbar ist, oder die Aktivitätsaufzeichnung verwenden. Beide Themen werden im Abschnitt [Referenzinformationen zur Überwachung](#) beschrieben.

Einführung in die verteilte Warteschlangenverwaltung

DQM (Distributed Queue Management) wird zum Definieren und Steuern der Kommunikation zwischen Warteschlangenmanagern verwendet.

Verwaltung verteilter Warteschlangen:


- Ermöglicht es Ihnen, Kommunikationskanäle zwischen Warteschlangenmanagern zu definieren und zu steuern.
- Stellt einen Nachrichtenkanaldienst zur Verfügung, mit dem Nachrichten von einem Typ von *lokaler Warteschlange*, der als Übertragungswarteschlange bezeichnet wird, in Kommunikationsverbindungen auf einem lokalen System und von Kommunikationsverbindungen zu lokalen Warteschlangen an einem Zielwarteschlangenmanager verschoben werden können.
- Stellt Funktionen zur Überwachung des Betriebs von Kanälen und zur Diagnose von Problemen mit Hilfe von Anzeigen, Befehlen und Programmen bereit.

Kanaldefinitionen ordnen Kanalnamen zu Übertragungswarteschlangen, Kommunikationsverbindungskennungen und Kanalattributen zu. Kanaldefinitionen werden auf verschiedenen Plattformen auf unterschiedliche Weise implementiert. Das Senden und Empfangen von Nachrichten wird durch Programme gesteuert, die als *Nachrichtenkanalagenten* (MCAs - Message Channel Agents) bezeichnet werden und die Kommunikation anhand der Kanaldefinitionen initialisieren und steuern.


Die MCAs wiederum werden von DQM selbst gesteuert. Die Struktur ist plattformabhängig, enthält jedoch in der Regel Empfangsprogramme und Auslösemonitore sowie Bedienerbefehle und Anzeigen.

Ein *Nachrichtenkanal* ist eine Einwegpipe für das Versetzen von Nachrichten von einem WS-Manager in einen anderen. Somit weist ein Nachrichtenkanal zwei Endpunkte auf, die durch ein MCAs-Paar dargestellt werden. Jeder Endpunkt verfügt über eine Definition seines Endes des Nachrichtenkanals. Beispiel: Ein Ende würde einen Sender, das andere Ende einen Empfänger definieren.

Weitere Informationen zum Definieren von Kanälen finden Sie unter:

-  [„Kanäle in UNIX, Linux, and Windows überwachen und steuern“ auf Seite 80](#)

Informationen zu den Planungsbeispielen für Nachrichtenkanäle finden Sie unter:

-  [Beispiel für Nachrichtenkanalplanung für verteilte Plattformen](#)

Informationen zu Kanalexits finden Sie unter [Channel-Exit-Programme für Messaging-Kanäle](#).

Zugehörige Konzepte

[„Senden und Empfangen von Nachrichten“ auf Seite 53](#)

Die folgende Abbildung zeigt das Management des verteilten Warteschlangenmanagements mit Details zu den Beziehungen zwischen Entitäten, wenn Nachrichten übertragen werden. Es zeigt auch den Ablauf der Steuerung an.

[„Kanalsteuerfunktion“ auf Seite 58](#)

Die Kanalsteuerfunktion stellt Funktionen zur Verfügung, mit der Sie Kanäle definieren, überwachen und steuern können.

[„Was passiert, wenn eine Nachricht nicht zugestellt werden kann?“ auf Seite 71](#)

Wenn eine Nachricht nicht zugestellt werden kann, kann der MCA sie auf mehrere Arten verarbeiten. Sie kann es erneut versuchen, sie kann an den Absender zurückkehren oder sie in die Warteschlange für dead-Mail setzen.

„Initialisierungs- und Konfigurationsdateien“ auf Seite 76

Die Handhabung von Kanalinitialisierungsdaten hängt von Ihrer WebSphere MQ -Plattform ab.

„Datenkonvertierung für Nachrichten“ auf Seite 77

WebSphere MQ -Nachrichten erfordern möglicherweise eine Datenkonvertierung, wenn sie zwischen Warteschlangen in verschiedenen Warteschlangenmanagern gesendet werden.

„Schreiben eigener Nachrichtenkanalagenten“ auf Seite 77

WebSphere MQ ermöglicht Ihnen, eigene MCA-Programme (Message Channel Agent, Nachrichtenkanalagent) zu schreiben oder eines von einem unabhängigen Softwareanbieter zu installieren.

„Andere Aspekte, die für die verteilte Warteschlangenverwaltung zu berücksichtigen sind“ auf Seite 78

Weitere Themen, die bei der Vorbereitung von WebSphere MQ für das verteilte Warteschlangenmanagement zu beachten sind In diesem Abschnitt finden Sie Informationen zu Nicht zugestellten Nachrichtenwarteschlangen, Warteschlangen in Verwendung, Systemerweiterungen und Benutzerexitprogrammen sowie zur Ausführung von Kanälen und Empfangsprogrammen als vertrauenswürdige Anwendungen.

Zugehörige Verweise

Beispielkonfigurationsdaten

Senden und Empfangen von Nachrichten

Die folgende Abbildung zeigt das Management des verteilten Warteschlangenmanagements mit Details zu den Beziehungen zwischen Entitäten, wenn Nachrichten übertragen werden. Es zeigt auch den Ablauf der Steuerung an.

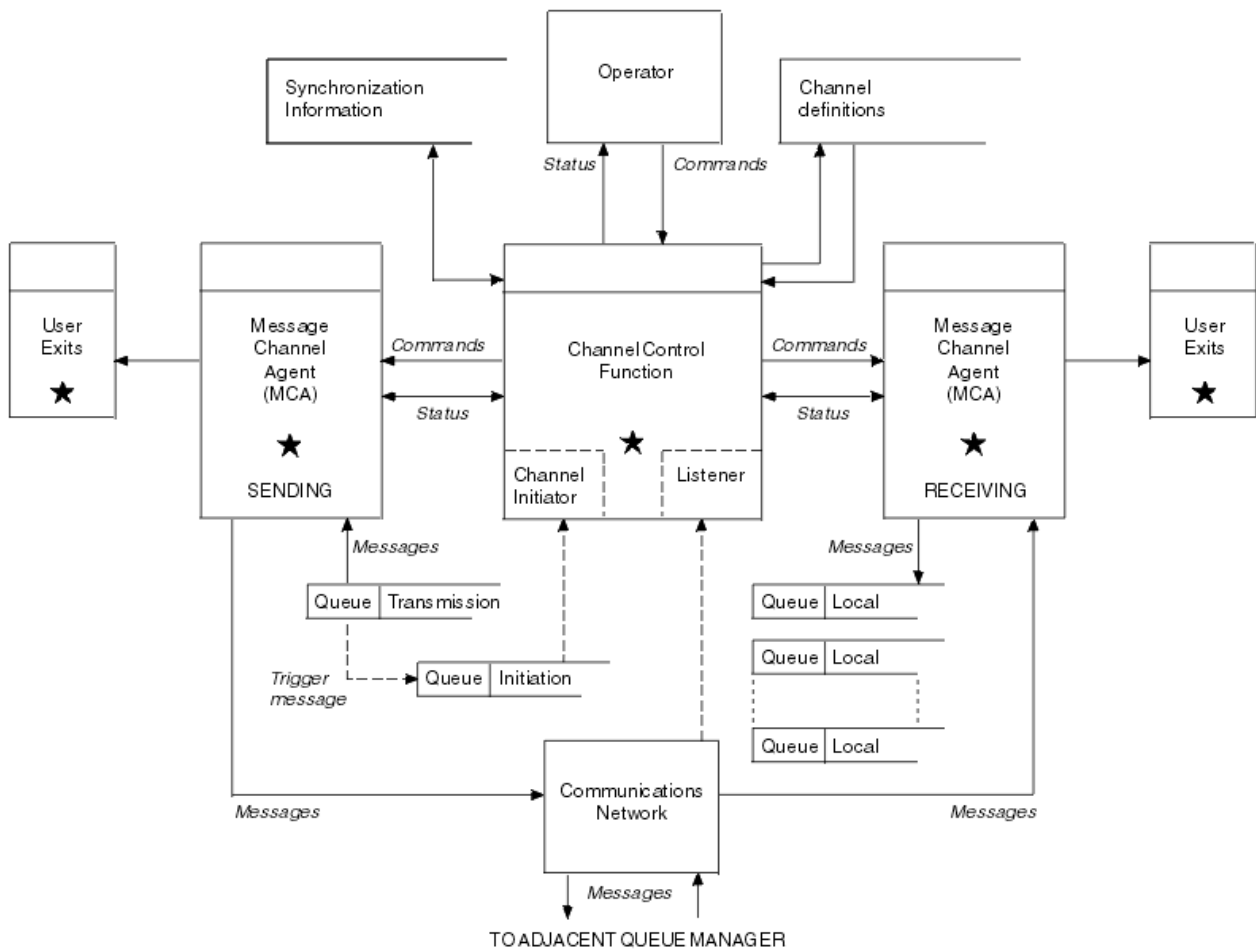


Abbildung 11. Verteiltes Warteschlangenmanagementmodell

Anmerkung:

1. Abhängig von der Plattform gibt es je einen MCA pro Kanal. Es können eine oder mehrere Kanalsteuerfunktionen für einen bestimmten Warteschlangenmanager vorhanden sein.
2. Die Implementierung von MCAs und Kanalsteuerfunktionen ist stark plattformabhängig. Sie können Programme oder Prozesse oder Threads sein, und sie können eine einzelne Entität oder viele verschiedene unabhängige oder verknüpfte Teile umfassen.
3. Alle Komponenten, die mit einem Stern gekennzeichnet sind, können die MQI verwenden.

Kanalparameter

Ein MCA erhält seine Parameter auf eine der folgenden Arten:

- Wird der Kanalname durch einen Befehl gestartet, wird er in einem Datenbereich übergeben. Der MCA liest dann die Kanaldefinition direkt, um die zugehörigen Attribute abzurufen.
- Der MCA kann für den Sender und in einigen Fällen Serverkanäle automatisch vom Warteschlangenmanager des Warteschlangenmanagers gestartet werden. Der Kanalname wird, sofern zutreffend, aus der Auslöserprozessdefinition abgerufen und an den MCA übergeben. Die übrige Verarbeitung ist mit der zuvor beschriebenen Verarbeitung identisch. Serverkanäle müssen nur so konfiguriert werden, dass sie ausgelöst werden, wenn sie vollständig qualifiziert sind, d. a. sie geben einen CONNAME an, zu dem eine Verbindung hergestellt werden soll.
- Wenn der Kanalname über Remotezugriff von einem Sender-, Server-, Requester- oder Clientanschluss gestartet wird, wird der Kanalname in den Anfangsdaten vom Partner-Nachrichtenkanalagenten übergeben. Der MCA liest die Kanaldefinition direkt, um die zugehörigen Attribute abzurufen.

Bestimmte Attribute, die in der Kanaldefinition nicht definiert sind, sind ebenfalls verhandelbar:

Nachrichten teilen

Wenn ein Ende keine geteilten Nachrichten unterstützt, werden die geteilten Nachrichten nicht gesendet.

Konvertierungsfunktion

Wenn ein Ende die erforderliche Codepagekonvertierung oder die Konvertierung der numerischen Codierung bei Bedarf nicht ausführen kann, muss das andere Ende sie verarbeiten. Wenn das Ende nicht unterstützt wird, kann der Kanal bei Bedarf nicht gestartet werden.

Unterstützung Verteilerliste

Wenn ein Ende Verteilerlisten nicht unterstützt, setzt der Partner-MCA ein Flag in seiner Übertragungswarteschlange, so dass es Nachrichten abfangen kann, die für mehrere Ziele bestimmt sind.

Kanalstatus und Folgenummern

Nachrichtenkanalagentenprogramme führen Datensätze der aktuellen Folgenummer und der logischen Arbeitseinheit für jeden Kanal und des allgemeinen Status des Kanals auf. Auf einigen Plattformen können Sie diese Statusinformationen anzeigen, um die Kanäle zu steuern.

So senden Sie eine Nachricht an einen anderen Warteschlangenmanager

In diesem Abschnitt wird die einfachste Methode zum Senden einer Nachricht zwischen Warteschlangenmanagern beschrieben, einschließlich der erforderlichen Voraussetzungen und Berechtigungen. Es können auch andere Methoden verwendet werden, um Nachrichten an einen fernen Warteschlangenmanager zu senden.

Bevor Sie eine Nachricht von einem WS-Manager an einen anderen Warteschlangenmanager senden, müssen Sie die folgenden Schritte ausführen:

1. Überprüfen Sie, ob Ihr ausgewähltes Kommunikationsprotokoll verfügbar ist.
2. Starten Sie die Warteschlangenmanager.
3. Starten Sie die Kanalinitiatoren.
4. Starten Sie die Empfangsprogramme.

Außerdem benötigen Sie die korrekte WebSphere MQ -Sicherheitsberechtigung, um die erforderlichen Objekte erstellen zu können.

Gehen Sie wie folgt vor, um Nachrichten von einem WS-Manager an einen anderen zu

- Definieren Sie die folgenden Objekte auf dem Quellenwarteschlangenmanager:
 - Senderkanal
 - Definition der fernen Warteschlange
 - Initialisierungswarteschlange (optional)
 - Übertragungswarteschlange
 - Warteschlange für nicht zustellbare Nachrichten
- Definieren Sie die folgenden Objekte auf dem Ziel-WS-Manager:
 - Empfängerkanal
 - Zielwarteschlange
 - Warteschlange für nicht zustellbare Nachrichten

Sie können diese Objekte je nach WebSphere MQ -Plattform mit verschiedenen Methoden definieren:

- Auf allen Plattformen können Sie die WebSphere MQ -Scriptbefehle (MQSC) verwenden, die unter [Die MQSC-Befehle](#) den PCF-Befehlen (Programmable Command Format) beschrieben sind, die unter [Verwaltungstasks automatisieren](#) beschrieben sind, oder den WebSphere MQ Explorer.

Weitere Informationen zum Erstellen der Komponenten zum Senden von Nachrichten an einen anderen WS-Manager finden Sie in den folgenden Unterabschnitten:

Zugehörige Konzepte

„Warteschlangenmanager erstellen und verwalten“ auf Seite 20

Bevor Sie Nachrichten und Warteschlangen verwenden können, müssen Sie mindestens einen WS-Manager und die zugehörigen Objekte erstellen und starten.

„IBM WebSphere MQ -Verfahren für verteiltes Messaging“ auf Seite 30

In den Unterabschnitten in diesem Abschnitt werden die Verfahren beschrieben, die bei der Planung von Kanälen verwendet werden. In diesen Unterabschnitten werden Verfahren beschrieben, mit deren Hilfe Sie planen, wie die Warteschlangenmanager miteinander verbunden werden, und den Fluss von Nachrichten zwischen Ihren Anwendungen verwalten.

„Einführung in die verteilte Warteschlangenverwaltung“ auf Seite 52

DQM (Distributed Queue Management) wird zum Definieren und Steuern der Kommunikation zwischen Warteschlangenmanagern verwendet.

„Ausgelöste Kanäle“ auf Seite 73

WebSphere MQ stellt eine Funktion zum automatischen Starten einer Anwendung bereit, wenn bestimmte Bedingungen in einer Warteschlange erfüllt sind. Diese Funktion wird als Triggerung bezeichnet.

„Sicherheit von Nachrichten“ auf Seite 71

Zusätzlich zu den typischen Wiederherstellungsfunktionen von WebSphere MQ stellt die verteilte Warteschlangenverwaltung sicher, dass Nachrichten ordnungsgemäß zugestellt werden, indem eine Synchronisationspunktprozedur verwendet wird, die zwischen den beiden Enden des Nachrichtenkanals koordiniert wird. Wenn diese Prozedur einen Fehler feststellt, wird der Kanal geschlossen, so dass Sie das Problem untersuchen und die Nachrichten sicher in der Übertragungswarteschlange behalten können, bis der Kanal erneut gestartet wird.

„Kanäle in UNIX, Linux, and Windows überwachen und steuern“ auf Seite 80

Für DQM müssen Sie die Kanäle zu fernen Warteschlangenmanagern erstellen, überwachen und steuern. Sie können Kanäle mit Befehlen, Programmen, IBM WebSphere MQ Explorer, Dateien für die Kanaldefinitionen und einem Speicherbereich für Synchronisationsinformationen steuern.

„Verbindungen zwischen Client und Server konfigurieren“ auf Seite 106

Um die Kommunikationsverbindungen zwischen WebSphere MQ MQI-Clients und -Servern zu konfigurieren, müssen Sie Ihr Kommunikationsprotokoll festlegen, die Verbindungen an beiden Enden der Verbindung definieren, ein Empfangsprogramm starten und Kanäle definieren.

Zugehörige Tasks

„WS-Manager-Cluster konfigurieren“ auf Seite 171

Über die Links in diesem Abschnitt erfahren Sie, wie Cluster funktionieren, wie eine Clusterkonfiguration entworfen wird und wie ein einfacher Cluster eingerichtet wird.

Kanäle definieren

Wenn Sie Nachrichten von einem WS-Manager an einen anderen Warteschlangenmanager senden möchten, müssen Sie zwei Kanäle definieren. Sie müssen einen Kanal auf dem Quellenwarteschlangenmanager und einen Kanal auf dem Zielwarteschlangenmanager definieren.

Auf dem Quellenwarteschlangenmanager

Definieren Sie einen Kanal mit dem Kanaltyp SENDER. Sie müssen Folgendes angeben:

- Der Name der Übertragungswarteschlange, die verwendet werden soll (das Attribut XMITQ).
- Der Verbindungsname des Partnersystems (Attribut CONNAME).
- Der Name des Kommunikationsprotokolls, das Sie verwenden (Attribut TRPTYPE). Unter WebSphere MQ für z/OS muss das Protokoll TCP oder LU6.2 sein. Auf anderen Plattformen müssen Sie dies nicht angeben. Sie können den Wert aus der Standardkanaldefinition übernehmen lassen.

Details zu allen Kanalattributen finden Sie unter [Kanalattribute](#).

Auf dem Zielwarteschlangenmanager

Definieren Sie einen Kanal mit einem Kanaltyp von RECEIVER und denselben Namen wie der Senderkanal.

Den Namen des von Ihnen verwendeten Übertragungsprotokolls (Attribut TRPTYPE) angeben. Unter WebSphere MQ für z/OS muss das Protokoll TCP oder LU6.2 sein. Auf anderen Plattformen müssen Sie dies nicht angeben. Sie können den Wert aus der Standardkanaldefinition übernehmen lassen.

Empfängerkanaldefinitionen können generisch sein. Dies bedeutet Folgendes: Wenn Sie mehrere Warteschlangenmanager mit demselben Empfänger kommunizieren, können die sendenden Kanäle alle denselben Namen für den Empfänger angeben, und es gilt eine Empfängerdefinition für alle.

Anmerkung: Der Wert des Parameters TRPTYPE wird vom antwortenden Nachrichtenkanalagenten ignoriert. Ein TRPTYPE mit dem Wert 'TCP' in der Senderkanaldefinition beispielsweise wird erfolgreich mit einem TRPTYPE-Wert 'LU62' in der Empfängerkanaldefinition als Partner gestartet.

Wenn Sie den Kanal definiert haben, können Sie ihn mit dem Befehl PING CHANNEL testen. Mit diesem Befehl wird eine spezielle Nachricht vom Senderkanal an den Empfängerkanal gesendet und überprüft, ob die Nachricht zurückgegeben wird.

Definieren der Warteschlangen

Wenn Sie Nachrichten von einem Warteschlangenmanager an einen anderen Warteschlangenmanager senden möchten, müssen Sie bis zu sechs Warteschlangen definieren. Sie müssen bis zu vier Warteschlangen auf dem Quellenwarteschlangenmanager und bis zu zwei Warteschlangen auf dem Ziel-WS-Manager definieren.

Auf dem Quellenwarteschlangenmanager

- Definition der fernen Warteschlange

Geben Sie in dieser Definition Folgendes an:

Name des fernen Warteschlangenmanagers

Der Name des Zielwarteschlangenmanagers.

Name der fernen Warteschlange

Der Name der Zielwarteschlange auf dem Ziel-WS-Manager.

Name der Übertragungswarteschlange

Der Name der Übertragungswarteschlange. Sie müssen diesen Namen der Übertragungswarteschlange nicht angeben. Ist dies nicht der Fall, wird eine Übertragungswarteschlange mit demselben Namen wie der Zielwarteschlangenmanager verwendet. Wenn diese Option nicht vorhanden ist, wird die Standardübertragungswarteschlange verwendet. Es wird empfohlen, der Übertragungswarteschlange denselben Namen wie der Zielwarteschlangenmanager zu geben, so dass die Warteschlange standardmäßig gefunden wird.

- Definition der Initialisierungswarteschlange

Erforderlich unter z/OS und optional auf anderen Plattformen. Überlegen Sie die Benennung der Initialisierungswarteschlange SYSTEM.CHANNEL.INITQ. auf anderen Plattformen.

- Definition der Übertragungswarteschlange

Eine lokale Warteschlange, bei der das Attribut USAGE auf XMITQ gesetzt ist.

- Definition der Warteschlange für nicht zustellbare

Definieren Sie eine Warteschlange für dead-Mail, in die unzustellbare Nachrichten geschrieben werden können.

Auf dem Zielwarteschlangenmanager

- Definition der lokalen Warteschlange

Die Zielwarteschlange. Der Name dieser Warteschlange muss mit dem Namen dieser Warteschlange übereinstimmen, der im Feld für den fernen Warteschlangennamen der Definition der fernen Warteschlange im Quellenwarteschlangenmanager angegeben ist.

- Definition der Warteschlange für nicht zustellbare

Definieren Sie eine Warteschlange für dead-Mail, in die unzustellbare Nachrichten geschrieben werden können.

Zugehörige Konzepte

„Übertragungswarteschlange erstellen“ auf Seite 58

Bevor ein Kanal (außer einem Requesterkanal) gestartet werden kann, muss die Übertragungswarteschlange wie in diesem Abschnitt beschrieben definiert werden. Die Übertragungswarteschlange muss in der Kanaldefinition angegeben werden.

Übertragungswarteschlange erstellen

Bevor ein Kanal (außer einem Requesterkanal) gestartet werden kann, muss die Übertragungswarteschlange wie in diesem Abschnitt beschrieben definiert werden. Die Übertragungswarteschlange muss in der Kanaldefinition angegeben werden.

Definieren Sie für jeden sendenden Nachrichtenkanal eine lokale Warteschlange mit dem Attribut USAGE, das auf XMITQ gesetzt ist. Wenn Sie eine bestimmte Übertragungswarteschlange in Ihren fernen Warteschlangendefinitionen verwenden möchten, erstellen Sie eine ferne Warteschlange wie angezeigt.

Verwenden Sie zum Erstellen einer Übertragungswarteschlange die WebSphere MQ -Befehle (MQSC), wie in den folgenden Beispielen gezeigt:

Beispiel für die Erstellung einer Übertragungswarteschlange

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

Beispiel für eine ferne Warteschlange erstellen

```
DEFINE QREMOTE(PAYROLL) DESCR('Remote queue for QM2') +  
XMITQ(QM2) RNAME(PAYROLL) RQMNAME(QM2)
```

Erwäge die Benennung der Übertragungswarteschlange den Namen des WS-Managers auf dem fernen System, wie in den Beispielen gezeigt.

Kanal starten

Wenn Sie Nachrichten in die ferne Warteschlange einlegen, die auf dem Quellenwarteschlangenmanager definiert ist, werden sie in der Übertragungswarteschlange gespeichert, bis der Kanal gestartet wird. Wenn der Kanal gestartet wurde, werden die Nachrichten an die Zielwarteschlange auf dem fernen WS-Manager zugestellt.

Starten Sie den Kanal auf dem sendenden Warteschlangenmanager mit dem Befehl START CHANNEL. Wenn Sie den sendenden Kanal starten, wird der empfangende Kanal automatisch gestartet (durch den Listener) und die Nachrichten werden an die Zielwarteschlange gesendet. Beide Enden des Nachrichtenkanals müssen aktiv sein, damit Nachrichten übertragen werden können.

Da sich die beiden Kanäle auf verschiedenen Warteschlangenmanagern befinden, konnten sie mit anderen Attributen definiert werden. Um Unterschiede zu beheben, gibt es beim Start des Kanals eine erste Datenverhandlung zwischen den beiden Enden. Im Allgemeinen arbeiten die beiden Enden des Kanals mit den Attributen, die die weniger Ressourcen benötigen. Auf diese Weise können größere Systeme die geringeren Ressourcen kleinerer Systeme am anderen Ende des Nachrichtenkanals aufnehmen.

Der sendende MCA teilt große Nachrichten auf, bevor er sie über den Kanal sendet. Sie werden auf dem fernen Warteschlangenmanager erneut assemblierassembliert. Dies ist für den Benutzer nicht erkennbar.

Ein MCA kann Nachrichten mit mehreren Threads übertragen. Dieser Prozess, der als *Pipelining* bezeichnet wird, ermöglicht es dem MCA, Nachrichten effizienter zu übertragen, wobei weniger Wartestatus vorhanden sind. Pipelining verbessert die Kanalleistung.

Kanalsteuerfunktion

Die Kanalsteuerfunktion stellt Funktionen zur Verfügung, mit der Sie Kanäle definieren, überwachen und steuern können.

Befehle werden über Anzeigen, Programme oder von einer Befehlszeile an die Kanalsteuerfunktion ausgegeben. In der Anzeige-Schnittstelle werden auch Kanalstatus- und Kanaldefinitionsdaten angezeigt. Sie können Programmable Command Formats oder die WebSphere MQ -Befehle (MQSC) und Steuerbefehle verwenden, die in „Kanäle in UNIX, Linux, and Windows überwachen und steuern“ auf Seite 80 beschrieben sind.

Die Befehle fallen in die folgenden Gruppen:

- Kanalverwaltung
- Kanalsteuerung
- Kanalstatusüberwachung

Kanalverwaltungsbefehle befassen sich mit den Definitionen der Kanäle. Sie ermöglichen Ihnen Folgendes:

- Kanaldefinition erstellen
- Kanaldefinition kopieren
- Kanaldefinition ändern
- Kanaldefinition löschen

Kanalsteuerungsbefehle verwalten den Betrieb der Kanäle. Sie ermöglichen Ihnen Folgendes:

- Kanal starten
- Kanal stoppen
- Erneutes Synchronisieren mit Partner (in einigen Implementierungen)
- Nachrichtenfolgennummern zurücksetzen
- Auflösen eines unbestätigten Nachrichtenaufbaus von Nachrichten
- Ping; Senden einer Testkommunikation über den Kanal

Die Kanalüberwachung zeigt den Status der Kanäle an, z. B.:

- Aktuelle Kanaleinstellungen
- Ob der Kanal aktiv oder inaktiv ist
- Gibt an, ob der Kanal in einem synchronisierten Status beendet wurde

Weitere Informationen zum Definieren, Steuern und Überwachen von Kanälen finden Sie in den folgenden Unterabschnitten:

Vorbereiten von Kanälen

Bevor Sie versuchen, einen Nachrichtenkanal oder einen MQI-Kanal zu starten, müssen Sie den Kanal vorbereiten. Sie müssen sicherstellen, dass alle Attribute der lokalen und fernen Kanaldefinitionen korrekt und kompatibel sind.

In Kanalattribute werden die Kanaldefinitionen und Attribute beschrieben.

Obwohl Sie explizite Kanaldefinitionen konfiguriert haben, können die Kanalverhandlungen, die bei einem Kanalstart ausgeführt werden, einen oder einen anderen der definierten Werte überschreiben. Dieses Verhalten ist normal und nicht für den Benutzer erkennbar und wurde auf diese Weise so angeordnet, dass ansonsten inkompatible Definitionen zusammenarbeiten können.

Automatische Definition von Empfänger- und Serververbindungskanälen

Wenn in WebSphere MQ auf allen Plattformen außer z/OS keine geeignete Kanaldefinition vorhanden ist, wird für einen Empfänger- oder Serververbindungskanal mit aktivierter automatischer Definition automatisch eine Definition erstellt. Die Definition wird wie folgt erstellt:

1. Die entsprechende Modellkanaldefinition, SYSTEM.AUTO.RECEIVER oder SYSTEM.AUTO.SVRCONN. Die Modellkanaldefinitionen für die automatische Definition sind mit den Systemstandardwerten SYSTEM.DEF.RECEIVER und SYSTEM.DEF.SVRCONN identisch, mit Ausnahme des Beschreibungsfelds, das "Auto-defined by" ist, gefolgt von 49 Leerzeichen. Der Systemadministrator kann einen beliebigen Teil der bereitgestellten Modellkanaldefinitionen ändern.
2. Informationen aus dem Partnersystem. Die Werte aus dem Partner werden für den Kanalnamen und den Folgenummernumbruch verwendet.
3. Ein Kanalexitprogramm, das Sie zum Ändern der Werte verwenden können, die von der automatischen Definition erstellt wurden. Siehe [Exitprogramm für die automatische Kanaldefinition \(Channel Auto-Definition\)](#)

Anschließend wird die Beschreibung geprüft, um festzustellen, ob sie durch einen Exit für die automatische Definition geändert wurde oder weil die Modelldefinition geändert wurde. Wenn die ersten 44 Zeichen immer noch "Automatisch definiert durch" gefolgt von 29 Leerzeichen sind, wird der Name des WS-Managers hinzugefügt. Wenn die letzten 20 Zeichen noch immer alle Leerzeichen sind, werden die lokale Uhrzeit und das Datum hinzugefügt.

Wenn die Definition erstellt und gespeichert wurde, beginnt der Kanalstart so, als ob die Definition immer vorhanden war. Die Stapelgröße, die Übertragungsgröße und die Nachrichtenlänge werden mit dem Partner ausgehandelt.

Andere Objekte definieren

Bevor ein Nachrichtenkanal gestartet werden kann, müssen beide Enden auf ihren Warteschlangenmanagern definiert sein (oder für die automatische Definition aktiviert sein). Die Übertragungswarteschlange, die sie bereitstellen soll, muss für den Warteschlangenmanager auf der sendenden Seite definiert sein. Die Kommunikationsverbindung muss definiert und verfügbar sein. Möglicherweise müssen Sie weitere WebSphere MQ -Objekte wie Definitionen für ferne Warteschlangen, Definitionen für Warteschlangenmanager-Aliasnamen und Definitionen für Antwortwarteschlangen vorbereiten, um die im Abschnitt „Anwendungen mithilfe der verteilten Steuerung von Warteschlangen verbinden“ auf Seite 29 beschriebenen Szenarios zu implementieren.

Informationen zum Definieren von MQI-Kanälen finden Sie in [„Definieren von MQI-Kanälen“](#) auf Seite 120.

Mehrere Nachrichtenkanäle pro Übertragungswarteschlange

Es ist möglich, mehr als einen Kanal pro Übertragungswarteschlange zu definieren, aber nur einer dieser Kanäle kann zu einem beliebigen Zeitpunkt aktiv sein. Berücksichtigen Sie diese Option bei der Bereitstellung alternativer Routen zwischen WS-Managern für die Fehlerberichtigung bei der Datenverkehrsverteilung und dem Verbindungsfehler. Eine Übertragungswarteschlange kann nicht von einem anderen Kanal verwendet werden, wenn der vorherige Kanal beendet wurde und einen Stapel von Nachrichten im unbestätigen Fall an der sendenden Seite hinterlässt. Weitere Informationen finden Sie in [„Unbestätigte Kanäle“](#) auf Seite 69.

Kanal starten

Ein Kanal kann dazu veranlasst werden, Nachrichten auf eine von vier Arten zu übertragen. Es kann sein:

- Gestartet durch einen Operator (nicht Empfänger, Clusterempfänger oder Serververbindungskanäle).
- Ausgelöst aus der Übertragungswarteschlange. Diese Methode gilt nur für Senderkanäle und vollständig qualifizierte Serverkanäle (die Kanäle, die einen CONNAME angeben). Sie müssen die erforderlichen Objekte für die Auslösung von Kanälen vorbereiten.
- Gestartet von einem Anwendungsprogramm (nicht Empfänger, Clusterempfänger oder Serververbindungskanäle).
- Fernes Starten aus dem Netz durch einen Sender-, Cluster-Sender-, Requester-, Server- oder Client-Kanal-Kanal. Empfänger-, Cluster-Empfänger- und möglicherweise Server- und Requesterkanalübertragung-

gen werden auf diese Weise gestartet; es handelt sich also um Serververbindungskanäle. Die Kanäle selbst müssen bereits gestartet sein (d. a. aktiviert).

Anmerkung: Da ein Kanal 'gestartet' ist, sendet er nicht unbedingt Nachrichten. Stattdessen kann es 'aktiviert' sein, die Übertragung zu starten, wenn eine der vier zuvor beschriebenen Ereignisse eintritt. Die Aktivierung und Deaktivierung eines Kanals wird mit den Bedienerbefehlen START und STOP erzielt.

Kanalstatus

Ein Kanal kann zu einem beliebigen Zeitpunkt in einem von vielen Status sein. Einige Staaten haben auch Unterzustände. Aus einem bestimmten Zustand kann ein Kanal in andere Zustände übergehen.

Abbildung 12 auf Seite 61 zeigt die Hierarchie aller möglichen Kanalstatus und die Unterzustände, die für jeden Kanalstatus gelten.

Abbildung 13 auf Seite 62 zeigt die Links zwischen Kanalstatus. Diese Links gelten für alle Arten von Nachrichtenkanal- und Serververbindungskanälen.

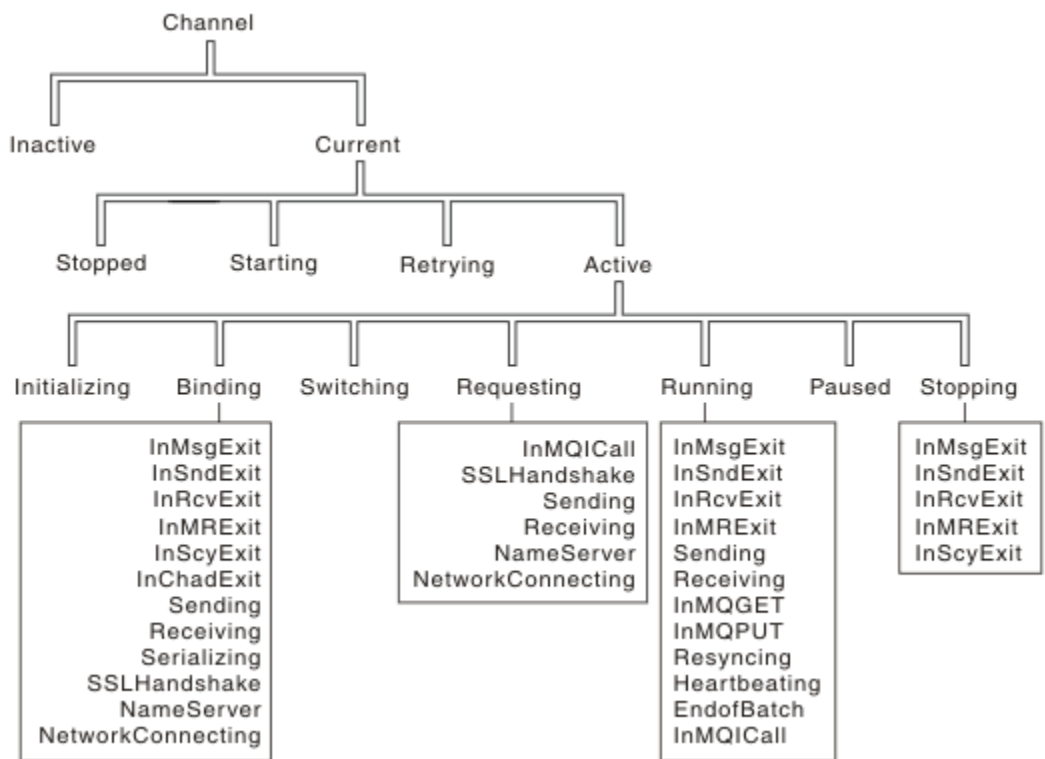


Abbildung 12. Kanalstatus und Unterzustände

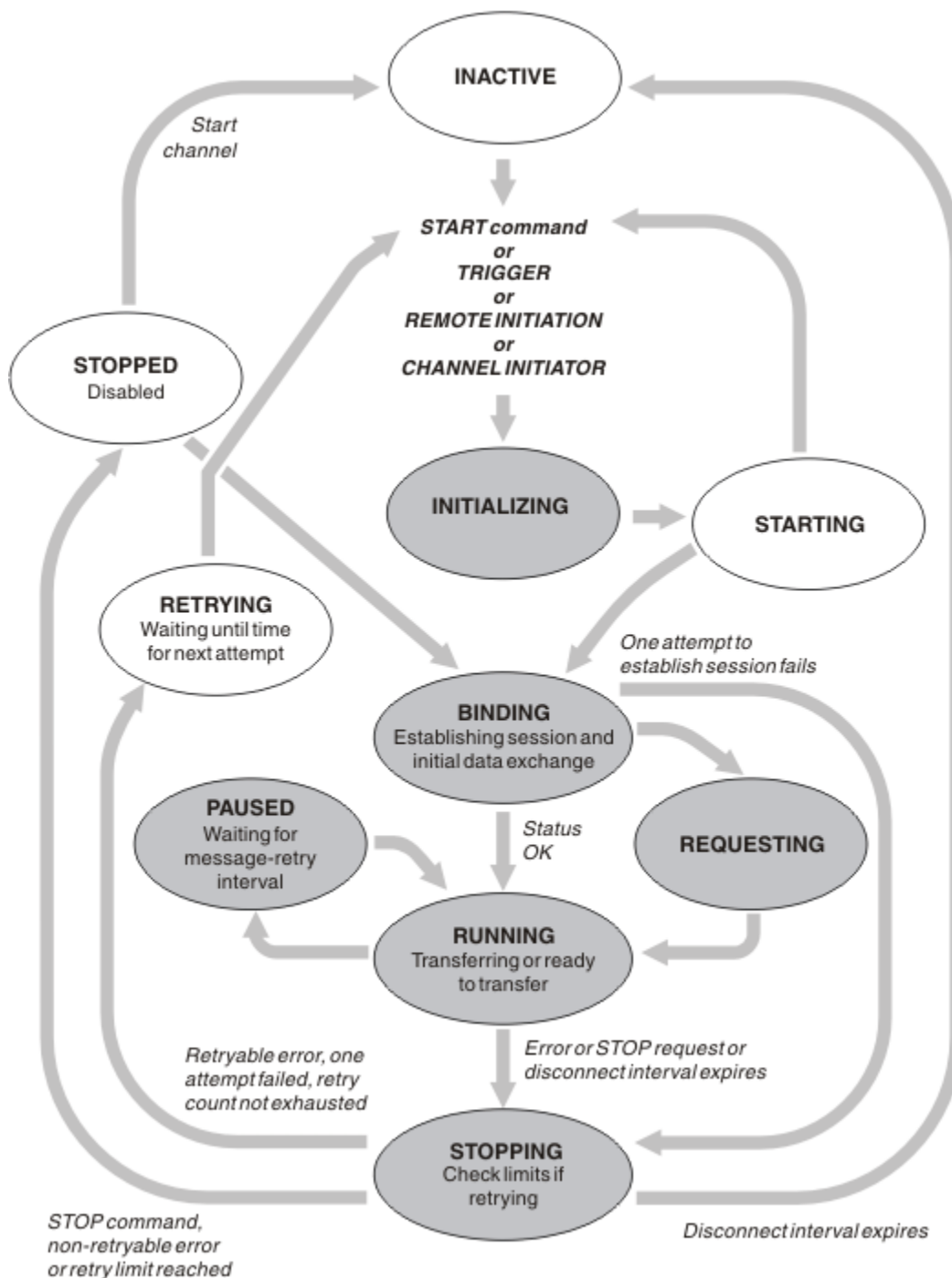


Abbildung 13. Flows zwischen Kanalstatus

Aktuell und aktiv

Ein Kanal ist *aktuell*, wenn er sich in einem anderen Status als "inaktiv" befindet. Ein aktueller Kanal ist *aktiv*, es sei denn, er befindet sich im Status **RETRYING**, **STOPPED** oder **STARTING**. Wenn ein Kanal aktiv ist, verbraucht er Ressourcen, und ein Prozess oder Thread wird ausgeführt. Die sieben möglichen Status eines aktiven Kanals (**INITIALIZING**, **BINDING**, **SWITCHING**, **REQUESTING**, **RUNNING**, **PAUSED** oder **STOPPING**) werden in [Abbildung 13](#) auf Seite 62 hervorgehoben.

Ein aktiver Kanal kann auch einen Substatus anzeigen, der mehr Details zu dem, was der Kanal gerade tut, gibt. Die Unterzustände für die einzelnen Status werden in [Abbildung 12](#) auf Seite 61 angezeigt.

Aktuell und aktiv

Der Kanal ist " aktuell " , wenn er sich in einem anderen Status als 'inaktiv' befindet. Ein aktueller Kanal ist " aktiv " , es sei denn, er befindet sich im Status RE TRYING, STOPPED oder STARTING.

Wenn ein Kanal "aktiv" ist, kann er auch einen Substatus anzeigen, der mehr Details zu dem, was der Kanal gerade tut, gibt.

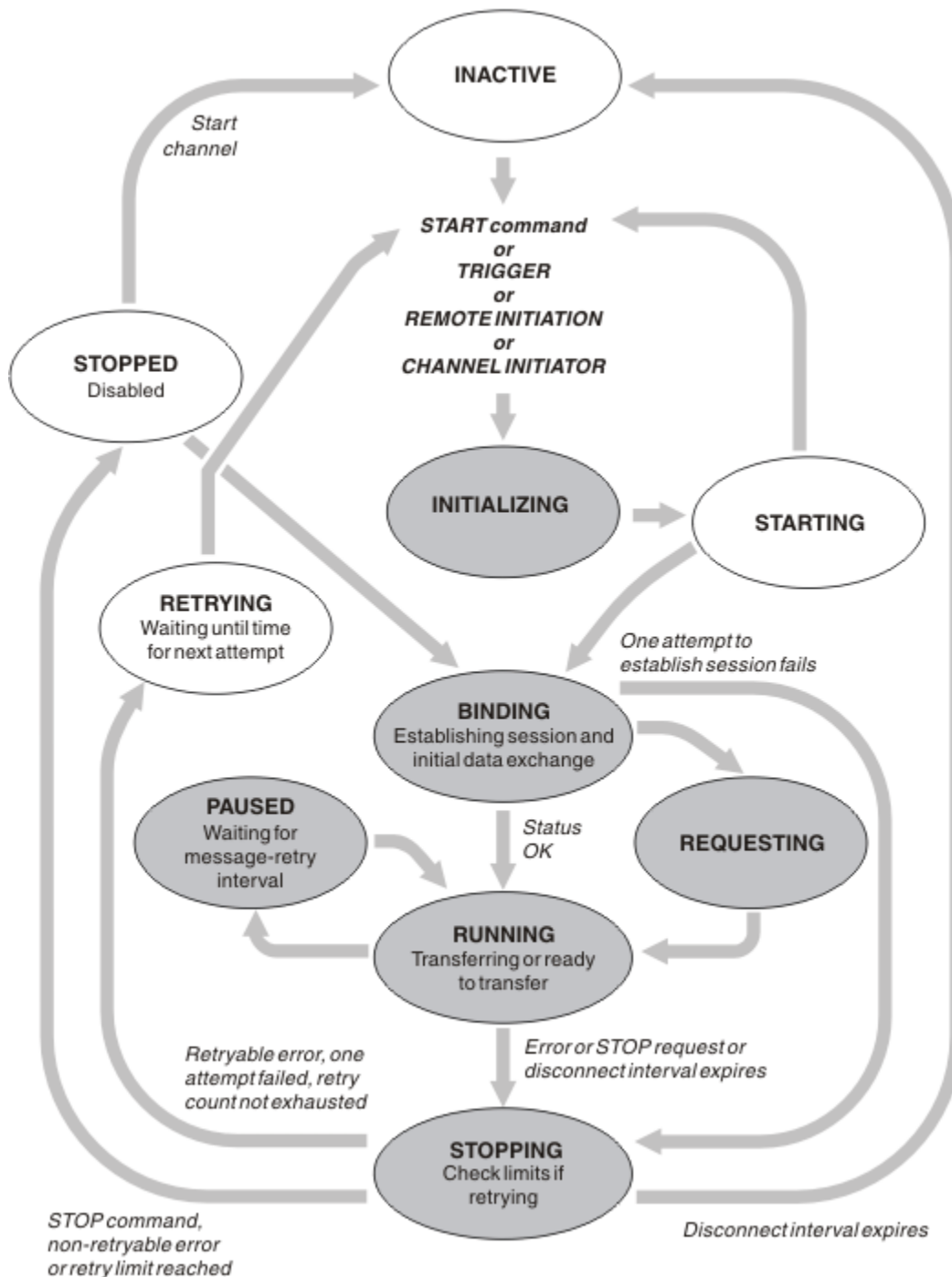


Abbildung 14. Flows zwischen Kanalstatus

Anmerkung:

1. Wenn sich ein Kanal in einem der sechs in Abbildung 14 auf Seite 63 hervorgehobenen Status befindet (INITIALIZING, BINDING, REQUESTING, RUNNING, PAUSED oder STOPPING), wird die Ressource verbraucht, und ein Prozess oder Thread wird ausgeführt; der Kanal ist *aktiv* .

2. Wenn sich ein Kanal im Status STOPPED befindet, kann die Sitzung aktiv sein, da der nächste Status noch nicht bekannt ist.

Angeben der maximalen Anzahl aktueller Kanäle

Sie können die maximale Anzahl Kanäle angeben, die gleichzeitig ausgeführt werden können. Diese Zahl gibt die Anzahl der Kanäle an, die Einträge in der Kanalstatustabelle enthalten, einschließlich der Kanäle, die erneut versucht werden, und Kanäle, die gestoppt wurden. Geben Sie dies mit der Konfigurationsdatei des Warteschlangenmanagers für UNIX and Linux -Systeme oder dem WebSphere MQ Explorer an. Weitere Informationen zu den Werten, die mit der Initialisierung oder der Konfigurationsdatei festgelegt wurden, finden Sie im Abschnitt Zeilengruppen für Konfigurationsdateien für verteilte Steuerung von Warteschlangen . Weitere Informationen zur Angabe der maximalen Anzahl von Kanälen finden Sie unter Verwaltung von IBM WebSphere MQ für WebSphere MQ für UNIX and Linux -Systeme und Windows -Systeme.

Anmerkung:

1. Serververbindungskanäle sind in dieser Nummer enthalten.
2. Ein Kanal muss aktiv sein, bevor er aktiv werden kann. Wenn ein Kanal gestartet wird, aber nicht aktuell werden kann, schlägt der Start fehl.

Angeben der maximalen Anzahl aktiver Kanäle

Sie können auch die maximale Anzahl aktiver Kanäle angeben, um zu verhindern, dass Ihr System von vielen Startkanälen überlastet wird. Wenn Sie diese Methode verwenden, legen Sie das Attribut "Unterbrechungsintervall" auf einen niedrigen Wert fest, damit die Wartezeitkanäle gestartet werden können, sobald andere Kanäle beendet werden.

Jedes Mal, wenn ein Kanal versucht, eine Verbindung zu seinem Partner herzustellen, muss er zu einem aktiven Kanal werden. Wenn der Versuch fehlschlägt, bleibt es ein aktueller Kanal, der nicht aktiv ist, bis es Zeit für den nächsten Versuch ist. Die Anzahl der Wiederholungen eines Kanalretries und die Häufigkeit, die durch den Wiederholungszähler und die Kanalattribute des Wiederholungsintervalls festgelegt wird. Es gibt kurze und lange Werte für diese beiden Attribute. Weitere Informationen finden Sie unter Kanalattribute .

Wenn ein Kanal zu einem aktiven Kanal werden muss (da ein Startbefehl abgesetzt wurde oder weil er ausgelöst wurde oder weil es Zeit für einen anderen Wiederholungsversuch ist), kann dies jedoch nicht möglich sein, da die Anzahl der aktiven Kanäle bereits den Maximalwert hat, wartet der Kanal, bis einer der aktiven Bereiche von einer anderen Kanalinstanz, die nicht mehr aktiv ist, freigegeben wird. Wenn jedoch ein Kanal gestartet wird, weil er über Remotezugriff eingeleitet wird und zu diesem Zeitpunkt keine aktiven Steckplätze zur Verfügung stehen, wird die ferne Initialisierung zurückgewiesen.

Wenn ein anderer Kanal als ein Requesterkanal versucht, aktiv zu werden, geht er in den Status STARTING. Dieser Status tritt auch dann auf, wenn ein aktiver Steckplatz sofort verfügbar ist, obwohl er nur im Status STARTING für einen kurzen Zeitraum vorhanden ist. Wenn der Kanal jedoch auf einen aktiven Steckplatz warten muss, befindet er sich im Status STARTING, während er wartet.

Requesterkanäle werden nicht in den Status STARTING (STARTING) gestartet. Wenn ein Requesterkanal nicht gestartet werden kann, weil die Anzahl der aktiven Kanäle bereits an der Grenze liegt, wird der Kanal abnormal beendet.

Wenn ein Kanal, bei dem es sich nicht um einen Requesterkanal handelt, keinen aktiven Slot abrufen kann und daher auf einen Slot wartet, wird eine Nachricht in das Protokoll geschrieben und ein Ereignis generiert. Wenn ein Slot später freigegeben wird und der Kanal ihn anfordern kann, werden eine weitere Nachricht und ein weiteres Ereignis generiert. Keines dieser Ereignisse und Nachrichten wird generiert, wenn der Kanal in der Lage ist, einen Slot direkt zu erhalten.

Wenn ein Befehl STOP CHANNEL abgesetzt wird, während der Kanal darauf wartet, aktiv zu werden, wird der Kanal in den Status STOPPED (STOPPED) wechselt. Ein Ereignis Channel-Stopped wird ausgelöst.

Serververbindungskanäle sind in der maximalen Anzahl aktiver Kanäle enthalten.

Weitere Informationen zur Angabe der maximalen Anzahl aktiver Kanäle finden Sie unter [Verwaltung IBM WebSphere MQ für WebSphere MQ für UNIX and Linux -Systeme und Windows -Systeme](#).

Kanalfehler

Fehler auf Kanälen bewirken, dass der Kanal weitere Übertragungen stoppen kann. Wenn es sich bei dem Kanal um einen Sender oder einen Server handelt, geht er in den Status WIEDERHOLUNG, da es möglich ist, dass sich das Problem möglicherweise selbst abbildet. Wenn der Kanal nicht in den Status RETRY wechseln kann, wechselt der Kanal in den Status STOPPED.

Für sendende Kanäle wird die zugeordnete Übertragungswarteschlange auf GET (DISABLED) gesetzt und die Auslösung ist ausgeschaltet. (Ein Befehl STOP mit STATUS (STOPPED) übernimmt die Seite, die sie in den Status STOPPED ausgegeben hat; nur das Auslaufen des Unterbrechungsintervalls oder ein STOP-Befehl mit STATUS (INACTIVE) macht sie normal beendet und wird inaktiv.) Kanäle, die sich im Status STOPPED befinden, benötigen Bedienereingriffe, bevor sie erneut gestartet werden können (siehe [„Gestoppte Kanäle erneut starten“](#) auf Seite 69).

Anmerkung: Bei UNIX-, Linux- und Windows-Systemen muss ein Kanalinitiator aktiv sein, damit eine Wiederholung versucht werden kann. Wenn der Kanalinitiator nicht verfügbar ist, wird der Kanal inaktiv und muss manuell erneut gestartet werden. Wenn Sie ein Script zum Starten des Kanals verwenden, stellen Sie sicher, dass der Kanalinitiator aktiv ist, bevor Sie versuchen, das Script auszuführen.

[Länger Wiederholungszähler \(LONGRTY\)](#) beschreibt, wie die Wiederholung funktioniert. Wenn der Fehler gelöscht wird, wird der Kanal automatisch erneut gestartet, und die Übertragungswarteschlange wird erneut aktiviert. Wenn der Wiederholungsgrenzwert ohne die Fehlerbereinigung erreicht wird, wechselt der Kanal in den Status STOPPED (STOPPED). Ein gestoppter Kanal muss manuell durch den Bediener erneut gestartet werden. Ist der Fehler weiterhin vorhanden, wiederholt er die Operation nicht erneut. Wenn die Übertragung erfolgreich gestartet wird, wird die Übertragungswarteschlange erneut aktiviert.

Wenn der Warteschlangenmanager gestoppt wird, während sich ein Kanal im Status RETRYING oder STOPPED befindet, wird der Kanalstatus beim Neustart des -Warteschlangenmanagers gespeichert. Der Kanalstatus für den Kanaltyp SVRCONN wird jedoch zurückgesetzt, wenn der -Warteschlangenmanager gestoppt wird, während sich der Kanal im Status STOPPED befindet.

Wenn ein Kanal keine Nachricht in die Zielwarteschlange einlegen kann, weil die Warteschlange voll ist oder unterdrückt wird, kann der Kanal die Operation in einem Zeitintervall (das im Attribut für den Nachrichtenwiederholungsintervall angegeben ist) wiederholt versuchen, die Operation zu wiederholen (die im Attribut für die Anzahl der Wiederholungen für Nachrichten angegeben ist). Alternativ können Sie einen eigenen Nachrichtenwiederholungsexit schreiben, der festlegt, welche Umstände eine Wiederholung bewirken, und die Anzahl der Versuche, die ausgeführt wurden. Der Kanal wechselt in den Status PAUSED, während er darauf wartet, dass das Nachrichtenwiederholungsintervall beendet wird.

Informationen zu den Kanalattributen und [Kanalexitprogramme für Messaging-Kanäle](#) finden Sie unter [Kanalattribute](#). Informationen zum Nachrichtenwiederholungsexit finden Sie im Abschnitt.

Grenzwerte für Serververbindungskanäle

Sie können Grenzwerte für Serververbindungskanäle festlegen, um zu verhindern, dass Clientanwendungen die Kanalressourcen des Warteschlangenmanagers (**MAXINST**) ausschöpfen, und um zu verhindern, dass eine einzelne Clientanwendung die Kapazität des Serververbindungskanals (**MAXINSTC**) ausschöpft.

Eine maximale Gesamtanzahl an Kanälen kann zu einem beliebigen Zeitpunkt auf einem einzelnen Warteschlangenmanager aktiv sein. Die Gesamtzahl der Serververbindungskanalinstanzen ist in der maximalen Anzahl aktiver Kanäle enthalten.

Wenn Sie nicht die maximale Anzahl simultaner Instanzen eines Serververbindungskanals angeben, der gestartet werden kann, ist es möglich, dass eine einzelne Clientanwendung eine Verbindung zu einem einzigen Serververbindungskanal herstellen kann, um die maximale Anzahl aktiver Kanäle zu erschöpfen, die verfügbar sind. Wenn die maximale Anzahl aktiver Kanäle erreicht ist, wird verhindert, dass andere Kanäle auf dem WS-Manager gestartet werden. Um diese Situation zu vermeiden, müssen Sie die Anzahl der simultanen Instanzen eines einzelnen Serververbindungskanals begrenzen, der unabhängig von dem Client gestartet werden kann, der gestartet wurde.

Wenn der Wert des Grenzwerts auf unter die aktuell laufende Anzahl der Instanzen des Serververbindungskanals reduziert wird, selbst auf null, sind die aktiven Kanäle nicht betroffen. Neue Instanzen können erst gestartet werden, wenn genügend vorhandene Instanzen nicht mehr ausgeführt werden, so dass die Anzahl der Instanzen, die momentan ausgeführt werden, kleiner als der Grenzwert ist.

Außerdem können viele verschiedene Client-Verbindungskanäle eine Verbindung zu einem einzelnen Serververbindungskanal herstellen. Der Grenzwert für die Anzahl simultaner Instanzen eines einzelnen Serververbindungskanals, der gestartet werden kann, unabhängig davon, welcher Client sie gestartet hat, verhindert, dass ein Client die maximale Kapazität des aktiven Kanals des Warteschlangenmanagers anstrengt. Wenn Sie nicht auch die Anzahl der simultanen Instanzen eines einzelnen Serververbindungskanals begrenzen, der von einem einzelnen Client aus gestartet werden kann, ist es möglich, dass eine einzelne fehlerhafte Clientanwendung so viele Verbindungen öffnet, dass sie die Kanalkapazität ausschöpft, die einem einzelnen Serververbindungskanal zugeordnet ist, und verhindert daher, dass andere Clients, die den Kanal verwenden müssen, keine Verbindung zu ihm herstellen müssen. Um diese Situation zu vermeiden, müssen Sie die Anzahl der simultanen Instanzen eines einzelnen Serververbindungskanals begrenzen, der von einem einzelnen Client aus gestartet werden kann.

Wenn der Wert der individuellen Clientgrenze unter die Anzahl der Instanzen des Serververbindungskanals reduziert wird, die momentan von einzelnen Clients ausgeführt werden, selbst auf null, sind die aktiven Kanäle nicht betroffen. Neue Instanzen des Serververbindungskanals können jedoch nicht von einem einzelnen Client gestartet werden, der den neuen Grenzwert überschreitet, bis genügend vorhandene Instanzen von diesem Client nicht mehr ausgeführt werden, so dass die Anzahl der Instanzen, die momentan ausgeführt werden, kleiner als der Wert dieses Parameters ist.

Überprüfen, ob das andere Ende des Kanals noch verfügbar ist

Sie können das Intervall der Überwachungssignale, das Intervall für die Nutzung des Überwachungszeitlimits und das Empfangszeitlimit verwenden, um zu überprüfen, ob das andere Ende des Kanals verfügbar ist.

Überwachungssignale

Sie können das Attribut "heartbeat interval channel" verwenden, um anzugeben, dass Flüsse von dem sendenden Nachrichtenkanalsystem übergeben werden sollen, wenn keine Nachrichten in der Übertragungswarteschlange vorhanden sind, wie im Heartbeat-Intervall (HBINT) beschrieben.

Keep Alive

Wenn Sie in WebSphere MQ für UNIX-, Linux- und Windows -Systeme TCP als Transportprotokoll verwenden, können Sie `keepalive=yes` festlegen. Wenn Sie diese Option angeben, prüft TCP regelmäßig, ob das andere Ende der Verbindung noch verfügbar ist. Ist dies nicht der Kanal, wird der Kanal beendet. Diese Option wird in Keepalive Interval (KAINT) beschrieben.

Wenn Sie über unzuverlässige Kanäle verfügen, die TCP-Fehler melden, verwenden Sie die Option **Keepalive**, um die Wahrscheinlichkeit zu beheben, dass die Kanäle wiederhergestellt werden.

Sie können Zeitintervalle angeben, um das Verhalten der Option **Keepalive** zu steuern. Wenn Sie das Zeitintervall ändern, werden nur die TCP/IP-Kanäle gestartet, die nach der Änderung gestartet wurden. Stellen Sie sicher, dass der Wert, den Sie für das Zeitintervall auswählen, kleiner ist als der Wert des Unterbrechungsintervalls für den Kanal.

Weitere Informationen zur Verwendung der Option **Keepalive** finden Sie unter dem Parameter KAINT im Befehl DEFINE CHANNEL.

Zeitlimit für Empfang

Wenn Sie TCP als Transportprotokoll verwenden, wird auch das empfangende Ende einer inaktiven Nicht-MQI-Kanalverbindung geschlossen, wenn für einen Zeitraum keine Daten empfangen werden. Dieser Zeitraum, der Wert *Zeitlimitüberschreitung bei Empfang*, wird gemäß dem Wert von HBINT (Überwachungssignalintervall) bestimmt.

In WebSphere MQ für UNIX-, Linux- und Windows -Systeme wird der Wert für *Empfangszeitlimit* wie folgt festgelegt:

1. Für eine anfängliche Anzahl von Datenflüssen, bevor eine Vereinbarung stattfindet, ist der Wert *receive time-out* doppelt so oft wie der Wert für HBINT aus der Kanaldefinition.
2. Wenn die Kanäle einen HBINT-Wert aushandeln, wenn HBINT auf weniger als 60 Sekunden gesetzt ist, wird der Wert von *receive time-out* auf das Doppelte dieses Werts gesetzt. Wenn HBINT auf 60 Sekunden oder mehr gesetzt ist, wird der Wert von *receive time-out* auf 60 Sekunden größer als der Wert von HBINT gesetzt.

Anmerkung:

1. Wenn einer der Werte null ist, gibt es kein Zeitlimit.
2. Bei Verbindungen, die keine Überwachungssignale unterstützen, wird der Wert von HBINT in Schritt 2 mit null ausgehandelt, und daher gibt es kein Zeitlimit. Daher müssen Sie TCP/IP KEEPALIVE verwenden.
3. Für Clientverbindungen, die gemeinsame Dialoge verwenden, können Heartbeats über den Kanal (von beiden Enden) die gesamte Zeit fließen, nicht nur, wenn ein MQGET-Aufruf aussteht.
4. Bei Clientverbindungen, bei denen die gemeinsamen Dialoge nicht verwendet werden, werden Überwachungssignale vom Server nur dann aus dem Server fließen, wenn der Client einen MQGET-Aufruf mit einem Wartestatus ausgibt. Daher wird es nicht empfohlen, das Intervall der Überwachungssignale für Clientkanäle zu klein zu setzen. Wenn das Überwachungssignal beispielsweise auf 10 Sekunden gesetzt wird, schlägt ein MQCMIT-Aufruf fehl (mit MQRC_CONNECTION_BROKEN), wenn die Festbeschreibung länger als 20 Sekunden dauert, da während dieser Zeit keine Daten geflossen sind. Dies kann bei großen Arbeitseinheiten der Fall sein. Es tritt jedoch nicht auf, wenn geeignete Werte für das Intervall der Überwachungssignale ausgewählt werden, da nur MQGET mit Wartezeit einen signifikanten Zeitraum in Anspruch nimmt.

Wenn SHARECNV ungleich null ist, verwendet der Client eine Vollduplexverbindung, d. -er., dass der Client während aller MQI-Aufrufe Überwachungssignale (und macht) kann.

5. In Clientkanälen von WebSphere MQ Version 7 können Überwachungssignale sowohl vom Server als auch von der Clientseite fließen. Das Zeitlimit an beiden Enden basiert auf $2 * \text{HBINT}$ für HBINTs von weniger als 60 Sekunden und $\text{HBINT} + 60$ für HBINTs von mehr als 60 Sekunden.
6. Das Abbrechen der Verbindung nach dem doppelten Intervall der Überwachungssignale ist gültig, da mindestens in jedem Intervall der Überwachungssignale ein Daten- oder Überwachungssignalfluss erwartet wird. Das Festlegen des Überwachungssignalintervalls ist zu klein, kann jedoch zu Problemen führen, insbesondere wenn Sie Kanalexits verwenden. Wenn der Wert für HBINT beispielsweise eine Sekunde ist und ein Sende- oder Empfangsexit verwendet wird, wartet das empfangende Ende nur 2 Sekunden, bevor der Kanal abgebrochen wird. Wenn der MCA eine Task wie das Verschlüsseln der Nachricht ausführt, ist dieser Wert möglicherweise zu kurz.

MCA-Adopting

Die Funktion "Adopt MCA" ermöglicht es IBM WebSphere MQ Explorer, einen Empfängerkanal abzubreaken und an seiner Stelle einen neuen zu starten.

Wenn ein Kanal einen Kommunikationsfehler aufweist, könnte der Empfängerkanal im Status 'Kommunikationsempfang' verbleiben. Wenn die Kommunikation erneut aufgebaut wird, versucht der Senderkanal, die Verbindung wieder herzustellen. Wenn der ferne Warteschlangenmanager feststellt, dass der Empfängerkanal bereits aktiv ist, kann keine andere Version desselben Empfängerkanals gestartet werden. Für dieses Problem ist ein Benutzereingriff erforderlich, um das Problem oder die Verwendung des System-keepalive zu beheben.

Mit der Funktion "Adopt MCA" wird das Problem automatisch gelöst. Sie ermöglicht es IBM WebSphere MQ Explorer, einen Empfängerkanal abzubreaken und an seiner Stelle einen neuen zu starten.

Die Funktion kann mit verschiedenen Optionen eingerichtet werden. **distributed** Informationen zu verteilten Plattformen finden Sie unter Verwaltung.

Kanäle stoppen und in den Quiescemodus versetzt

In diesem Abschnitt wird erläutert, wie Sie einen Kanal stoppen und in den Wartemodus versetzen können, bevor das Unterbrechungszeitintervall abläuft.

Nachrichtenkanäle sind so konzipiert, dass sie lange Verbindungen zwischen Warteschlangenmanagern mit ordnungsgemäße gesteuert Beendigung, die nur durch das Kanalattribut "Unterbrechungsintervall" gesteuert werden, bestehen. Dieser Mechanismus funktioniert gut, es sei denn, der Bediener muss den Kanal beenden, bevor das Unterbrechungszeitintervall abläuft. Dieser Bedarf kann in den folgenden Situationen auftreten:

- Systemstilllegung
- Ressourcenschonung
- Unilaterale Aktion an einem Ende eines Kanals

In diesem Fall können Sie den Kanal stoppen. Dazu können Sie folgende Optionen verwenden:

- MQSC-Befehl STOP CHANNEL
- Befehl "Stop Channel PCF"
- IBM WebSphere MQ Explorer

Es gibt drei Optionen zum Stoppen von Kanälen mit den folgenden Befehlen:

QUIESCE

Die Option QUIESCE versucht, den aktuellen Stapel von Nachrichten zu beenden, bevor der Kanal gestoppt wird.

FORCE

Die Option FORCE versucht, den Kanal sofort zu stoppen, und möglicherweise muss der Kanal beim Neustart erneut synchronisiert werden, da der Kanal möglicherweise unbestätigt bleibt.

BEENDEN

Die Option TERMINATE versucht, den Kanal sofort zu stoppen, und beendet den Thread oder den Prozess des Kanals.

Alle diese Optionen verlassen den Kanal in einem STOPPED-Status, der einen Bedienereingriff erfordert, um ihn erneut zu starten.

Das Stoppen des Kanals auf der Senderseite ist zwar wirksam, erfordert jedoch einen Bedienereingriff, um den Neustart zu starten. Am empfangenden Ende des Kanals sind die Dinge sehr viel schwieriger, da der MCA auf Daten von der sendenden Seite wartet und es keine Möglichkeit gibt, einen *ordnungsgemäßen* Abschluss des Kanals von der empfangenden Seite zu initiieren; der Stoppbefehl steht an, bis der MCA aus seinem Wartestatus für Daten zurückkehrt.

Abhängig von den erforderlichen Betriebsmerkmalen gibt es daher drei empfohlene Methoden für die Verwendung von Kanälen:

- Wenn die Kanäle lange ausgeführt werden sollen, beachten Sie, dass die Beendigung nur von der sendenden Seite aus ordnungsgemäß beendet werden kann. Wenn Kanäle unterbrochen werden, d. h. gestoppt sind, ist ein Bedienereingriff (ein Befehl START CHANNEL) erforderlich, um sie erneut starten zu können.
- Wenn die Kanäle nur dann aktiv sein sollen, wenn Nachrichten für die Übertragung vorhanden sind, setzen Sie das Unterbrechungsintervall auf einen relativ niedrigen Wert. Die Standardeinstellung ist hoch und wird daher nicht für Kanäle empfohlen, in denen diese Steuerungsstufe erforderlich ist. Da es schwierig ist, den empfangenden Kanal zu unterbrechen, ist es die wirtschaftlichste Option, den Kanal automatisch zu trennen und die Verbindung zu den Workloadanforderungen wieder herzustellen. Für die meisten Kanäle kann die entsprechende Einstellung des Unterbrechungsintervalls heuristisch festgelegt werden.
- Sie können das Attribut "heartbeat-interval" verwenden, um zu bewirken, dass der sendende MCA einen Überwachungssignalfuss an den empfangenden MCA sendet, wenn er keine Nachrichten zum Senden hat. Diese Aktion gibt den empfangenden MCA aus seinem Wartestatus frei und gibt ihm die Möglichkeit, den Kanal in den Quiescemodus zu setzen, ohne zu warten, bis das Unterbrechungsinter-

vall abgelaufen ist. Geben Sie dem Überwachungssignalintervall einen niedrigeren Wert als den Wert des Unterbrechungsintervalls an.

Anmerkung:

1. Es wird empfohlen, das Unterbrechungsintervall für Serverkanäle auf einen niedrigen Wert oder auf Überwachungssignale zu setzen. Dieser niedrige Wert soll den Fall zulassen, dass der Requesterkanal abnormal beendet wird (z. B. weil der Kanal abgebrochen wurde), wenn keine Nachrichten für den zu sendenden Serverkanal vorhanden sind. Wenn das Unterbrechungsintervall hoch ist und keine Überwachungssignale verwendet werden, erkennt der Server nicht, dass der Anforderer beendet wurde (was er nur beim nächsten Versuch, eine Nachricht an den Requester zu senden), beendet hat. Während der Server noch aktiv ist, hält er die Übertragungswarteschlange für exklusive Eingabe bereit, um weitere Nachrichten zu erhalten, die in die Warteschlange eintreffen. Wenn versucht wird, den Kanal vom anfordernden Benutzer erneut zu starten, empfängt die Startanforderung einen Fehler, da der Server immer noch die Übertragungswarteschlange für die exklusive Eingabe geöffnet hat. Es ist erforderlich, den Serverkanal zu stoppen und den Kanal anschließend erneut vom Requester erneut zu starten.

Gestoppte Kanäle erneut starten

Wenn ein Kanal in den Status STOPPED wechselt, müssen Sie den Kanal manuell erneut starten.

Geben Sie einen der folgenden Befehle aus, um den Kanal erneut zu starten:

- Der MQSC-Befehl START CHANNEL
- Der PCF-Befehl 'Kanal starten'
- IBM WebSphere MQ Explorer

Für Sender-oder Serverkanäle wurde die zugeordnete Übertragungswarteschlange auf GET (DISABLED) gesetzt und die Auslösung wurde inaktiviert, wenn der Kanal in den Status STOPPED (STOPPED) eingetreten ist. Wenn die Startanforderung empfangen wird, werden diese Attribute automatisch zurückgesetzt.

Wenn der -Warteschlangenmanager (auf verteilten Plattformen) gestoppt wird, während sich ein Kanal im Status RETRYING oder STOPPED befindet, wird der Kanalstatus beim Neustart des -Warteschlangenmanagers gespeichert. Der Kanalstatus für den Kanaltyp SVRCONN wird jedoch zurückgesetzt, wenn der Warteschlangenmanager gestoppt wird, während sich der Kanal im Status STOPPED befindet.

Unbestätigte Kanäle

Ein unbestätigter Kanal ist ein Kanal, der sich im Zweifel mit einem fernen Kanal befindet, über den Nachrichten gesendet und empfangen wurden.

Beachten Sie die Unterscheidung zwischen diesem und einem Warteschlangenmanager, der Zweifel daran hat, welche Nachrichten in einer Warteschlange festgeschrieben werden sollen.

Sie können die Möglichkeit, einen Kanal in Frage zu stellen, durch die Verwendung des Kanalparameters 'Batch Heartbeat' (BATCHHB) verringern. Wenn ein Wert für diesen Parameter angegeben wird, überprüft ein Senderkanal, ob der ferne Kanal noch aktiv ist, bevor er weitere Maßnahmen ergreift. Wird keine Antwort empfangen, wird davon ausgegangen, dass der Empfängerkanal nicht mehr aktiv ist. Die Nachrichten können rückgängig gemacht und erneut weitergeleitet werden, und der Sender-Channel wird nicht in Frage gestellt. Dadurch wird die Zeit verringert, in der der Kanal in Zweifel zum Zeitraum zwischen dem Senderkanal gestellt werden konnte, der verifiziert, dass der Empfängerkanal noch aktiv ist, und zu überprüfen, ob der Empfängerkanal die gesendeten Nachrichten empfangen hat. Weitere Informationen zum Parameter für den Stapelüberwachungssignalwert finden Sie unter [Kanalattribute](#).

Unbestätigte Kanalprobleme werden in der Regel automatisch aufgelöst. Auch wenn die Kommunikation verloren geht und ein Kanal im Zweifelsfall mit einem Nachrichtenstapel im Sender in Zweifel steht, wenn der Empfangsstatus unbekannt ist, wird die Situation behoben, wenn die Kommunikation erneut aufgebaut wird. Die Folgenummer und die LUWID-Datensätze werden zu diesem Zweck aufbewahrt. Der Kanal steht im Zweifel, bis LUWID-Informationen ausgetauscht wurden und nur ein Nachrichtenstapel für den Kanal unbestätigt sein kann.

Sie können den Kanal bei Bedarf manuell resynchronisieren. Der Begriff *manuell* schließt die Verwendung von Operatoren oder Programmen ein, die WebSphere MQ -Systemverwaltungsbefehle enthalten. Der

manuelle Resynchronisationsprozess funktioniert wie folgt. Diese Beschreibung verwendet MQSC-Befehle, aber Sie können auch die PCF-Entsprechungen verwenden.

1. Verwenden Sie den Befehl `DISPLAY CHSTATUS`, um die ID der letzten logischen Arbeitseinheit (LUWID) für **jede** Seite des Kanals zu finden. Verwenden Sie dazu die folgenden Befehle:

- Für die unbestätigte Seite des Kanals:

```
DISPLAY CHSTATUS(name) SAVED CURLUWID
```

Sie können die Parameter `CONNNAME` und `XMITQ` verwenden, um den Kanal weiter zu identifizieren.

- Für die Empfängerseite des Kanals:

```
DISPLAY CHSTATUS(name) SAVED LSTLUWID
```

Sie können den Parameter `CONNNAME` verwenden, um den Kanal weiter zu identifizieren.

Die Befehle sind unterschiedlich, da nur die sendende Seite des Kanals unbestätigt sein kann. Die empfangende Seite ist nie im Zweifel.

Unter WebSphere MQ für IBM i kann der Befehl `DISPLAY CHSTATUS` mit dem Befehl `STRMQMMQSC` oder mit dem CL-Befehl `WRKMQMCHST` ausgeführt werden.

2. Wenn die beiden LUWIDs identisch sind, hat die empfangende Seite die Arbeitseinheit festgeschrieben, die vom Absender als unbestätigt betrachtet wird. Die sendende Seite kann nun die unbestätigten Nachrichten aus der Übertragungswarteschlange entfernen und sie erneut aktivieren. Dies wird mit dem folgenden Kanal `RESOLVE` ausgeführt:

```
RESOLVE CHANNEL(name) ACTION(COMMIT)
```

3. Wenn die beiden LUWIDs unterschiedlich sind, hat die empfangende Seite die UOW nicht festgeschrieben, die der Absender als unbestätigt betrachtet. Die sendende Seite muss die unbestätigten Nachrichten in der Übertragungswarteschlange aufbewahren und sie erneut senden. Dies wird mit dem folgenden Kanal `RESOLVE` ausgeführt:

```
RESOLVE CHANNEL(name) ACTION(BACKOUT)
```

Wenn dieser Prozess abgeschlossen ist, ist der Kanal nicht mehr im Zweifel. Die Übertragungswarteschlange kann jetzt, falls erforderlich, von einem anderen Kanal verwendet werden.

Fehlerdiagnose

Es gibt zwei unterschiedliche Aspekte bei der Fehlerbestimmung: Probleme, die bei der Übergabe eines Befehls erkannt werden, und Probleme, die während des Betriebs der Kanäle erkannt werden.

Befehlsüberprüfung

Befehle und Anzeigendaten müssen von Fehlern frei sein, bevor sie zur Verarbeitung akzeptiert werden. Alle Fehler, die von der Validierung festgestellt werden, werden dem Benutzer unverzüglich durch Fehlermeldungen mitgeteilt.

Die Problemdiagnose beginnt mit der Interpretation dieser Fehlermeldungen und ergreift Korrekturmaßnahmen.

Fehler werden verarbeitet

Probleme, die während des normalen Betriebs der Kanäle gefunden werden, werden an die Systemkonsole oder das Systemprotokoll gemeldet. Die Problemdiagnose beginnt mit der Erfassung aller relevanten Informationen aus dem Protokoll und wird mit der Analyse fortgesetzt, um das Problem zu identifizieren.

Bestätigungs- und Fehlernachrichten werden an das Terminal zurückgegeben, das die Befehle nach Möglichkeit eingeleitet hat.

WebSphere MQ erzeugt Abrechnungs- und Statistikdaten, mit denen Sie Nutzungs- und Leistungstrends ermitteln können. **distributed** Auf verteilten Plattformen werden diese Informationen als PCF-Datensätze erzeugt. Details hierzu finden Sie unter [Strukturdatentypen](#).

Nachrichten und Codes

Informationen zu Nachrichten und Codes, die bei der primären Diagnose des Problems hilfreich sind, finden Sie im Abschnitt [Diagnosenachrichten und Ursachencodes](#).

Sicherheit von Nachrichten

Zusätzlich zu den typischen Wiederherstellungsfunktionen von WebSphere MQ stellt die verteilte Warteschlangenverwaltung sicher, dass Nachrichten ordnungsgemäß zugestellt werden, indem eine Synchronisationspunktprozedur verwendet wird, die zwischen den beiden Enden des Nachrichtenkanals koordiniert wird. Wenn diese Prozedur einen Fehler feststellt, wird der Kanal geschlossen, so dass Sie das Problem untersuchen und die Nachrichten sicher in der Übertragungswarteschlange behalten können, bis der Kanal erneut gestartet wird.

Die Synchronisationspunktprozedur hat einen zusätzlichen Vorteil, indem sie versucht, eine *im Zweifel*-Situation wiederherzustellen, wenn der Kanal gestartet wird. (*Im Zweifel* ist der Status einer Arbeitseinheit mit Wiederherstellung, für die ein Synchronisationspunkt angefordert wurde, aber das Ergebnis der Anforderung ist noch nicht bekannt.) Darüber hinaus sind diese Funktion die beiden folgenden Funktionen:

1. Mit Commit oder Backout auflösen
2. Folgenummer zurücksetzen

Die Verwendung dieser Funktionen findet nur in Ausnahmefällen statt, da sich der Kanal in den meisten Fällen automatisch wiederfindet.

Schnelle, nicht persistente Nachrichten

Mit dem Kanalattribut der nicht persistenten Nachrichtengeschwindigkeit (NPMSPEED) kann angegeben werden, dass alle nicht persistenten Nachrichten auf dem Kanal schneller zugestellt werden sollen. Weitere Informationen zu diesem Attribut finden Sie im Abschnitt [Nicht persistente Nachrichtengeschwindigkeit \(NPMSPEED\)](#).

Wenn ein Kanal beendet wird, während sich schnell, nicht persistente Nachrichten im Transit befinden, gehen die Nachrichten möglicherweise verloren, und die Anwendung kann bei Bedarf für die Wiederherstellung sorgen.

Wenn der empfangende Kanal die Nachricht nicht in die Zielwarteschlange einlegen kann, wird sie in die Warteschlange für nicht zustellbare Nachrichten gestellt, wenn eine Warteschlange definiert wurde. Ist dies nicht der Fall, wird die Nachricht gelöscht.

Anmerkung: Wenn das andere Ende des Kanals die Option nicht unterstützt, wird der Kanal mit normaler Geschwindigkeit ausgeführt.

Nicht zugegebene Nachrichten

Informationen darüber, was passiert, wenn eine Nachricht nicht zugestellt werden kann, finden Sie in „Was passiert, wenn eine Nachricht nicht zugestellt werden kann?“ auf Seite 71.

Was passiert, wenn eine Nachricht nicht zugestellt werden kann?

Wenn eine Nachricht nicht zugestellt werden kann, kann der MCA sie auf mehrere Arten verarbeiten. Sie kann es erneut versuchen, sie kann an den Absender zurückkehren oder sie in die Warteschlange für dead-Mail setzen.

Abbildung 15 auf Seite 72 zeigt die Verarbeitung an, die auftritt, wenn ein MCA keine Nachricht in die Zielwarteschlange einlegen kann. (Die angezeigten Optionen gelten nicht für alle Plattformen.)

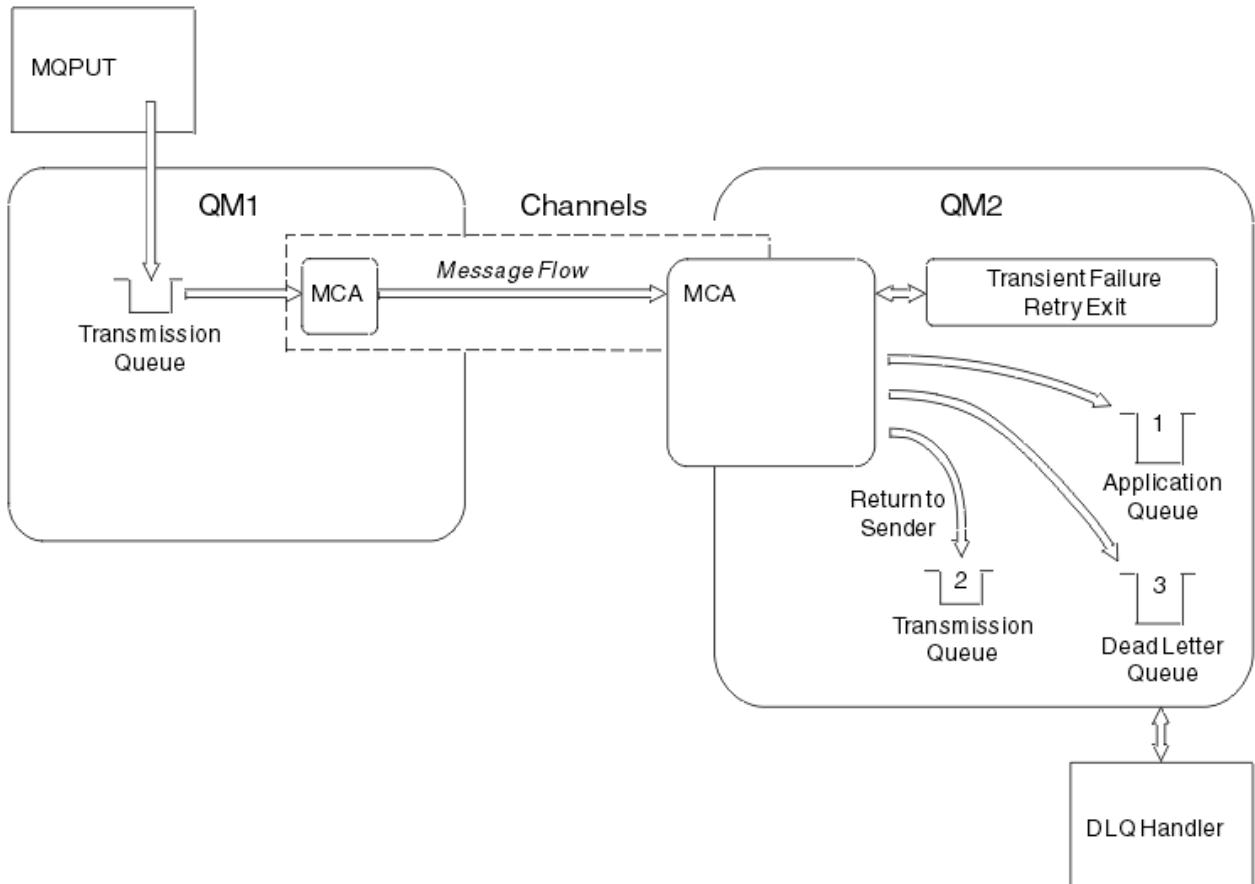


Abbildung 15. Was passiert, wenn eine Nachricht nicht zugestellt werden kann

Wie in der Abbildung dargestellt, kann der MCA mehrere Dinge mit einer Nachricht tun, die er nicht liefern kann. Die Aktion wird durch die Optionen bestimmt, die bei der Definition des Kanals und in den MQPUT-Berichtsoptionen für die Nachricht angegeben sind.

1. Nachrichtenwiederholung

Wenn der Nachrichtenkanalnachrichtenkanalcode (MCA) keine Nachricht aus einem Grund in die Zielwarteschlange einlegen kann (z. B. weil die Warteschlange voll ist), kann der MCA warten und die Operation zu einem späteren Zeitpunkt wiederholen. Sie können feststellen, ob der MCA wartet, wie lange und wie viele Versuche er hat.

- Sie können eine Nachrichtenwiederholungszeit und ein Intervall für MQPUT-Fehler angeben, wenn Sie Ihren Kanal definieren. Wenn die Nachricht nicht in die Zielwarteschlange gestellt werden kann, weil die Warteschlange voll ist oder für die Ausführung gesperrt ist, versucht der MCA die Operation, wie oft angegeben, in dem angegebenen Zeitintervall versucht wird.
- Sie können einen eigenen Nachrichtenwiederholungsexit schreiben. Über den Exit können Sie angeben, unter welchen Bedingungen der MCA die MQPUT- oder MQOPEN-Operation wiederholen soll. Geben Sie den Namen des Exits an, wenn Sie den Kanal definieren.

2. Zurück-zu-Absender

Wenn die Nachrichtenwiederholung nicht erfolgreich war oder ein anderer Typ von Fehler festgestellt wurde, kann der Nachrichtenkanalnachrichtenkanalfehler die Nachricht an den Absender zurücksenden. Zum Aktivieren von "return-to-sender" müssen Sie die folgenden Optionen im Nachrichtendescriptor angeben, wenn Sie die Nachricht in die ursprüngliche Warteschlange stellen:

- Die Berichtsoption MQRO_EXCEPTION_WITH_FULL_DATA

- Die Berichtsoption MQRO_DISCARD_MSG
- Der Name der Empfangswarteschlange für Antworten und der Empfangswarteschlange für Antworten an den Warteschlangenmanager.

Wenn der MCA die Nachricht nicht in die Zielwarteschlange einlegen kann, generiert er einen Ausnahmebericht, der die ursprüngliche Nachricht enthält, und stellt ihn in eine Übertragungswarteschlange, die an die in der ursprünglichen Nachricht angegebene Warteschlange für Antwortnachrichten gesendet werden soll. (Wenn sich die Warteschlange für Antwortnachrichten auf demselben WS-Manager wie der Nachrichtenkanalmanager befindet, wird die Nachricht direkt in diese Warteschlange gestellt, nicht in eine Übertragungswarteschlange.)

3. Warteschlange für nicht zustellbare Nachrichten

Wenn eine Nachricht nicht zugestellt oder zurückgegeben werden kann, wird sie in die Warteschlange für dead-letter (DLQ) gestellt. Sie können den DLQ-Handler verwenden, um die Nachricht zu verarbeiten. Diese Verarbeitung wird unter Behandlung nicht zugestellter Nachrichten mit der WebSphere MQ -Steueroutine der Warteschlange für nicht zustellbare Nachrichten für IBM WebSphere MQ für UNIX-, Linuxbeschrieben. Wenn die Warteschlange für nicht zustellbare Nachrichten nicht verfügbar ist, wird die Nachricht vom sendenden Nachrichtenkanalsystem (MCA) in der Übertragungswarteschlange und der Kanal gestoppt. Bei einem schnellen Kanal gehen nicht persistente Nachrichten verloren, die nicht in eine Warteschlange mit dead-letter geschrieben werden können.

Wenn in IBM WebSphere MQ Version 7.0 keine lokale Warteschlange für nicht zustellbare Nachrichten definiert ist, die ferne Warteschlange nicht erreichbar oder nicht definiert ist und keine ferne Warteschlange für nicht zustellbare Nachrichten vorhanden ist, schaltet der Sendekanal in den RETRY-Modus um und Nachrichten werden automatisch in die Übertragungswarteschlange zurückgesetzt.

Zugehörige Verweise

Warteschlange für nicht zustellbare Mail verwenden (USEDLQ)

Ausgelöste Kanäle

WebSphere MQ stellt eine Funktion zum automatischen Starten einer Anwendung bereit, wenn bestimmte Bedingungen in einer Warteschlange erfüllt sind. Diese Funktion wird als Triggerung bezeichnet.

Diese Erläuterung soll als Übersicht über die Auslösekonzepte dienen. Eine vollständige Beschreibung finden Sie unter WebSphere MQ -Anwendungen mit Auslösern starten.

Plattformspezifische Informationen finden Sie in den folgenden Informationen:

- Für Windows siehe UNIX and Linux -Systeme, „Auslösen von Kanälen auf UNIX-, Linux -und Windows -Systemen“ auf Seite 75

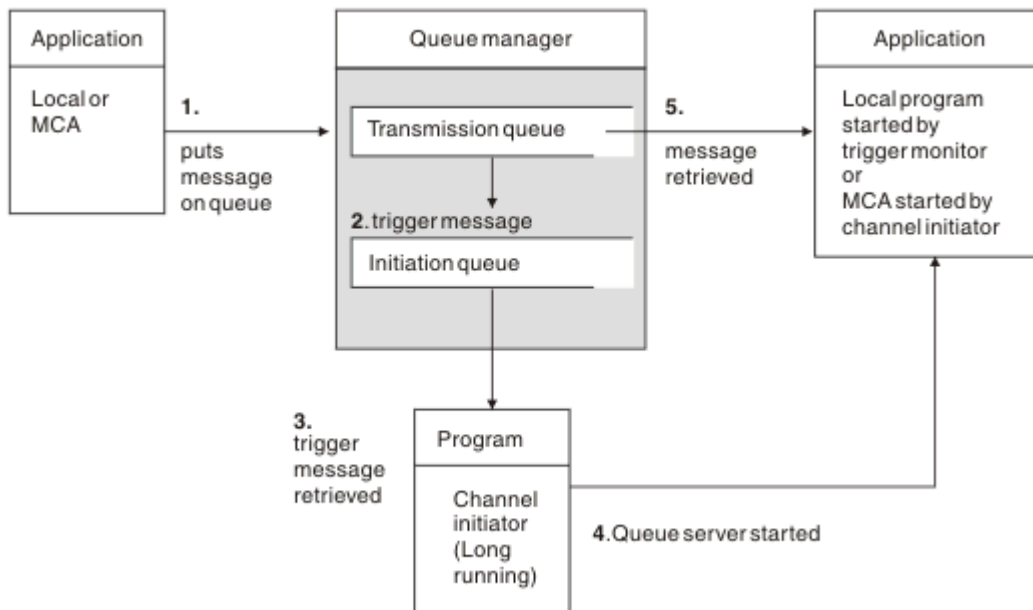


Abbildung 16. Die Konzepte des Triggerns

Die für die Auslösung erforderlichen Objekte werden in [Abbildung 16 auf Seite 74](#) angezeigt. Es zeigt die folgende Abfolge von Ereignissen an:

1. Der lokale WS-Manager stellt eine Nachricht von einer Anwendung oder von einem Nachrichtenkanalagenten (MCA) in die Übertragungswarteschlange.
2. Wenn die Auslöserbedingungen erfüllt sind, stellt der lokale WS-Manager eine Auslösenachricht in die Initialisierungswarteschlange.
3. Das Kanalinitiatorprogramm long-running überwacht die Initialisierungswarteschlange und ruft Nachrichten bei ihrer Ankunft ab.
4. Der Kanalinitiator verarbeitet die Auslösenachrichten in Übereinstimmung mit den darin enthaltenen Informationen. Diese Informationen können den Kanalnamen enthalten, in dem Fall, in dem der entsprechende MCA gestartet wird.
5. Die lokale Anwendung oder der MCA, der ausgelöst wurde, ruft die Nachrichten aus der Übertragungswarteschlange ab.

Um dieses Szenario zu konfigurieren, müssen Sie folgende Schritte ausführen:

- Erstellen Sie die Übertragungswarteschlange mit dem Namen der Initialisierungswarteschlange (das heißt SYSTEM.CHANNEL.INITQ) in dem entsprechenden Attribut.
- Stellen Sie sicher, dass die Initialisierungswarteschlange (SYSTEM.CHANNEL.INITQ) vorhanden ist.
- Stellen Sie sicher, dass das Kanalinitiatorprogramm verfügbar und aktiv ist. Das Kanalinitiatorprogramm muss in seinem Startbefehl mit dem Namen der Initialisierungswarteschlange angegeben werden.
- Optional können Sie die Prozessdefinition für die Triggerung erstellen, falls sie nicht vorhanden ist, und stellen Sie sicher, dass das Feld *UserData* den Namen des Kanals enthält, der verwendet wird. Anstatt eine Prozessdefinition zu erstellen, können Sie den Kanalnamen in dem Attribut *TriggerData* der Übertragungswarteschlange angeben. WebSphere MQ für UNIX, Linux und Windows -Systeme lassen die Angabe des Kanalnamens als leer zu. In diesem Fall wird die erste verfügbare Kanaldefinition mit dieser Übertragungswarteschlange verwendet.
- Stellen Sie sicher, dass die Definition der Übertragungswarteschlange den Namen der Prozessdefinition enthält, die sie bereitstellen soll (falls zutreffend), der Name der Initialisierungswarteschlange und die auslösenden Merkmale, die Sie fühlen, am besten geeignet sind. Das Auslösersteuerattribut ermöglicht es, dass die Auslösung aktiviert wird, oder nicht, falls erforderlich.

Anmerkung:

1. Das Kanalinitiatorprogramm dient als 'Auslösemonitor', der die Initialisierungswarteschlange überwacht, die zum Starten von Kanälen verwendet wird.
2. Es kann eine Initialisierungswarteschlange und ein Auslöserprozess verwendet werden, um eine beliebige Anzahl Kanäle auszulösen.
3. Es können eine beliebige Anzahl von Initialisierungswarteschlangen und Auslöserprozessen definiert werden.
4. Es wird empfohlen, einen Auslösertyp FIRST zu verwenden, um eine Überflutung des Systems mit Kanalstarts zu vermeiden.

Auslösen von Kanälen auf UNIX-, Linux -und Windows -Systemen

Sie können eine Prozessdefinition in WebSphere MQ erstellen und Prozesse definieren, die ausgelöst werden. Verwenden Sie den WebSphere MQ-Scriptbefehl DEFINE PROCESS, um eine Prozessdefinition zu erstellen, die den Prozess benennt, der ausgelöst werden soll, wenn Nachrichten in einer Übertragungswarteschlange ankommen. Das Attribut USERDATA der Prozessdefinition enthält den Namen des Kanals, der von der Übertragungswarteschlange bedient wird.

Definieren Sie die lokale Warteschlange (QM4), und geben Sie an, dass Auslösenachrichten in die Initialisierungswarteschlange (IQ) geschrieben werden sollen, um die Anwendung auszulösen, die den Kanal startet (QM3.TO.QM4):

```
DEFINE QLOCAL(QM4) TRIGGER INITQ(SYSTEM.CHANNEL.INITQ) PROCESS(P1) USAGE(XMITQ)
```

Definieren Sie die Anwendung (Prozess P1), die gestartet werden soll:

```
DEFINE PROCESS(P1) USERDATA(QM3.TO.QM4)
```

Für WebSphere MQ für UNIX-, Linux -und Windows -Systeme können Sie alternativ eine Prozessdefinition überflüssig machen, indem Sie den Kanalnamen im Attribut TRIGDATA der Übertragungswarteschlange angeben.

Definieren Sie die lokale Warteschlange (QM4). Geben Sie an, dass Auslösenachrichten in die Standardinitialisierungswarteschlange SYSTEM.CHANNEL.INITQ geschrieben werden sollen, um die Anwendung (Prozess P1) auszulösen, die den Kanal startet (QM3.TO.QM4):

```
DEFINE QLOCAL(QM4) TRIGGER INITQ(SYSTEM.CHANNEL.INITQ)  
USAGE(XMITQ) TRIGDATA(QM3.TO.QM4)
```

Wenn Sie keinen Kanalnamen angeben, durchsucht der Kanalinitiator die Kanaldefinitionsdateien, bis er einen Kanal findet, der der benannten Übertragungswarteschlange zugeordnet ist.

Zugehörige Konzepte

[„Kanalinitiator starten und stoppen“ auf Seite 75](#)

Die Triggerung wird mithilfe des Kanalinitiatorprozesses implementiert.

[„Anwendungen mithilfe der verteilten Steuerung von Warteschlangen verbinden“ auf Seite 29](#)

In diesem Abschnitt finden Sie ausführliche Informationen zur übergreifenden Kommunikation zwischen WebSphere MQ-Installationen, einschließlich Informationen zu Warteschlangendefinitionen, Kanaldefinitionen, Auslösungen und Synchronisationspunktprozeduren.

Zugehörige Verweise

[Kanalprogramme auf UNIX-, Linux-und Windows-Systemen](#)

Kanalinitiator starten und stoppen

Die Triggerung wird mithilfe des Kanalinitiatorprozesses implementiert.

Dieser Kanalinitiatorprozess wird mit dem MQSC-Befehl START CHINIT gestartet. Wenn Sie nicht die Standardinitialisierungswarteschlange verwenden, geben Sie den Namen der Initialisierungswarteschlange im Befehl an. Wenn Sie beispielsweise den Befehl START CHINIT zum Starten der Warteschlange IQ für den Standardwarteschlangenmanager verwenden möchten, geben Sie Folgendes ein:

```
START CHINIT INITQ(IQ)
```

Standardmäßig wird ein Kanalinitiator automatisch mit der Standardinitialisierungswarteschlange SYSTEM.CHANNEL.INITQ. gestartet. Wenn Sie alle Kanalinitiatoren manuell starten möchten, führen Sie die folgenden Schritte aus:

1. Erstellen und starten Sie den WS-Manager.
2. Ändern Sie die Eigenschaft SCHINIT des WS-Managers in MANUAL.
3. Beenden Sie den WS-Manager und starten Sie ihn erneut

In Linux und Windows wird ein Kanalinitiator automatisch gestartet. Die Anzahl der Kanalinitiatoren, die gestartet werden können, ist begrenzt. Der Standardwert und der Maximalwert sind 3. Sie können dies mithilfe von MAXINITIATORS in der Datei qm.ini für UNIX and Linux -Systeme und in der Registry für Windows-Systeme ändern.

Weitere Informationen zum Kanalinitiatorbefehl **runmqchi** und zu den anderen Steuerbefehlen finden Sie im Abschnitt [WebSphere MQ -Steuerbefehle](#).

Kanalinitiator stoppen

Der Standardkanalinitiator wird beim Starten eines Warteschlangenmanagers automatisch gestartet. Alle Kanalinitiatoren werden automatisch gestoppt, wenn ein WS-Manager gestoppt wird.

Initialisierungs- und Konfigurationsdateien

Die Handhabung von Kanalinitialisierungsdaten hängt von Ihrer WebSphere MQ -Plattform ab.

Windows-, UNIX and Linux -Systeme

In WebSphere MQ für Windows, UNIX and Linux -Systeme gibt es *Konfigurationsdateien*, die grundlegende Konfigurationsinformationen zur WebSphere MQ -Installation enthalten.

Es gibt zwei Konfigurationsdateien: eine gilt für die Maschine, die andere für einen einzelnen WS-Manager.

WebSphere MQ -Konfigurationsdatei

Diese Datei enthält Informationen, die für alle Warteschlangenmanager auf dem WebSphere MQ -System relevant sind. Die Datei heißt mqs.ini. Eine vollständige Beschreibung finden Sie in [Verwaltung für WebSphere MQ für Windows, UNIX and Linux -Systeme](#).

Warteschlangenmanagerkonfigurationsdatei

Diese Datei enthält Konfigurationsdaten, die sich auf einen bestimmten Warteschlangenmanager beziehen. Die Datei heißt qm.ini.

Sie wird während der Erstellung des Warteschlangenmanagers erstellt und kann Konfigurationsinformationen enthalten, die für alle Aspekte des Warteschlangenmanagers relevant sind. Die Informationen in der Datei enthalten Details dazu, wie sich die Konfiguration des Protokolls von der Standardkonfiguration in der Konfigurationsdatei WebSphere MQ unterscheidet.

Die Konfigurationsdatei des Warteschlangenmanagers wird im Stammverzeichnis der Verzeichnisstruktur, die vom Warteschlangenmanager belegt ist, gehalten. Für die Attribute "DefaultPath" müssten die WS-Manager-Konfigurationsdateien für einen Warteschlangenmanager mit dem Namen QMNAME beispielsweise wie folgt sein:

Für UNIX and Linux -Systeme:

```
/var/mqm/qmgrs/QMNAME/qm.ini
```

Es folgt ein Auszug aus einer Datei 'qm.ini'. Sie gibt an, dass der TCP/IP-Listener an Port 2500 empfangsbereit sein soll, die maximale Anzahl der aktuellen Kanäle 200 beträgt und die maximale Anzahl aktiver Kanäle 100 sein soll.

```
TCP:
  Port=2500
CHANNELS:
  MaxChannels=200
  MaxActiveChannels=100
```

Sie können einen Bereich von TCP/IP-Ports angeben, die von einem abgehenden Kanal verwendet werden sollen. Eine Methode besteht darin, die Datei qm.ini zu verwenden, um den Anfang und das Ende eines Bereichs von Portwerten anzugeben. Das folgende Beispiel zeigt eine Datei qm.ini, die eine Reihe von Kanälen angibt:

```
TCP:
  StrPort=2500
  EndPort=3000
CHANNELS:
  MaxChannels=200
  MaxActiveChannels=100
```

Wenn Sie einen Wert für 'StrPort' oder 'EndPort' angeben, müssen Sie einen Wert für beide angeben. Der Wert von 'EndPort' muss immer größer als der Wert von 'StrPort' sein.

Der Kanal versucht, die einzelnen Portwerte in dem angegebenen Bereich zu verwenden. Wenn die Verbindung erfolgreich hergestellt werden kann, ist der Portwert der Port, den der Kanal verwendet.

Für Windows -Systeme:

```
C:\Program Files\IBM\WebSphere MQ\qmgrs\QMNAME\qm.ini
```

Weitere Informationen zu Dateien 'qm.ini' finden Sie im Abschnitt [Zeilengruppen der Konfigurationsdatei für die verteilte Steuerung von Warteschlangen](#).

Datenkonvertierung für Nachrichten

WebSphere MQ -Nachrichten erfordern möglicherweise eine Datenkonvertierung, wenn sie zwischen Warteschlangen in verschiedenen Warteschlangenmanagern gesendet werden.

Eine WebSphere MQ besteht aus zwei Teilen:

- Steuerinformationen in einem Nachrichtendeskriptor
- Anwendungsdaten

Für beide Teile ist möglicherweise eine Datenkonvertierung erforderlich, wenn sie zwischen Warteschlangen in verschiedenen Warteschlangenmanagern gesendet wird. Informationen zur Anwendungsdatenkonvertierung finden Sie unter [Anwendungsdatenkonvertierung](#).

Schreiben eigener Nachrichtenkanalagenten

WebSphere MQ ermöglicht Ihnen, eigene MCA-Programme (Message Channel Agent, Nachrichtenkanalagent) zu schreiben oder eines von einem unabhängigen Softwareanbieter zu installieren.

Sie können eigene MCA-Programme schreiben, damit WebSphere MQ über Ihr eigenes proprietäres Kommunikationsprotokoll interagiert oder Nachrichten über ein Protokoll gesendet werden, das von WebSphere MQ nicht unterstützt wird. (Sie können keinen eigenen MCA schreiben, um mit einem von WebSphere MQ bereitgestellten MCA am anderen Ende zu interagieren.)

Wenn Sie einen Nachrichtenkanalagenten verwenden möchten, der nicht von WebSphere MQ bereitgestellt wurde, müssen Sie die folgenden Punkte beachten.

Senden und Empfangen von Nachrichten

Sie müssen eine sendende Anwendung schreiben, die Nachrichten von überall dort abrufen, wo Ihre Anwendung sie stellt, z. B. aus einer Übertragungswarteschlange, und sendet sie an ein Protokoll, mit dem Sie kommunizieren möchten. Sie müssen außerdem eine empfangende Anwendung schreiben, die Nachrichten aus diesem Protokoll aufnimmt und sie in Zielwarteschlangen stellt. Die sendenden und empfangenden Anwendungen verwenden die MQI-Aufrufe (Message Queue Interface, Nachrichtenwarteschlangenschnittstelle) und nicht alle speziellen Schnittstellen.

Sie müssen sicherstellen, dass Nachrichten nur einmal zugestellt werden. Die Synchronisationspunkt-Koordination kann zur Unterstützung bei dieser Zustellung verwendet werden.

Kanalsteuerfunktion

Sie müssen Ihre eigenen Verwaltungsfunktionen bereitstellen, um Kanäle zu steuern. Sie können die Kanalverwaltungsfunktionen von WebSphere MQ weder für die Konfiguration (z. B. den Befehl DEFINE CHANNEL) noch für die Überwachung (z. B. DISPLAY CHSTATUS) Ihrer Kanäle verwenden.

Initialisierungsdatei

Sie müssen eine eigene Initialisierungsdatei angeben, wenn Sie einen benötigen.

Anwendungsdatenkonvertierung

Möglicherweise möchten Sie die Datenkonvertierung für Nachrichten, die Sie an ein anderes System senden, ermöglichen. Ist dies der Fall, verwenden Sie die Option MQGMO_CONVERT im MQGET-Aufruf, wenn Sie Nachrichten von überall dort abrufen, wo Ihre Anwendung sie einreicht, z. B. die Übertragungswarteschlange.

Benutzerexits

Überlegen Sie, ob Sie Benutzerexits benötigen. Ist dies der Fall, können Sie dieselben Schnittstellendefinitionen verwenden, die auch WebSphere MQ verwendet.

Auslösefunktion

Wenn Ihre Anwendung Nachrichten in eine Übertragungswarteschlange einreicht, können Sie die Attribute der Übertragungswarteschlange so konfigurieren, dass Ihr sendende Nachrichtenkanalgruppe ausgelöst wird, wenn Nachrichten in die Warteschlange eintreffen.

Kanalinitiator

Möglicherweise müssen Sie einen eigenen Kanalinitiator bereitstellen.

Andere Aspekte, die für die verteilte Warteschlangenverwaltung zu berücksichtigen sind

Weitere Themen, die bei der Vorbereitung von WebSphere MQ für das verteilte Warteschlangenmanagement zu beachten sind In diesem Abschnitt finden Sie Informationen zu Nicht zugestellten Nachrichtenwarteschlangen, Warteschlangen in Verwendung, Systemerweiterungen und Benutzerexitprogrammen sowie zur Ausführung von Kanälen und Empfangsprogrammen als vertrauenswürdige Anwendungen.

Nicht zugegebene Nachrichtenwarteschlange

Um sicherzustellen, dass Nachrichten, die in der Warteschlange für nicht zustellbare Nachrichten (auch die Warteschlange für den dead-letter oder DLQ genannt) ankommen, verarbeitet werden, müssen Sie ein Programm erstellen, das in regelmäßigen Abständen ausgelöst oder ausgeführt werden kann, um diese Nachrichten zu verarbeiten. Ein DLQ-Handler wird mit WebSphere MQ auf UNIX and Linux -Systemen bereitgestellt. Weitere Informationen finden Sie unter [The sample DLQ handler, amqsdq](#).

Warteschlangen im Gebrauch

MCAs für Empfängerkanäle können die Zielwarteschlangen auch dann offen halten, wenn Nachrichten nicht übertragen werden. Dies führt dazu, dass die Warteschlangen " im Gebrauch " angezeigt werden.

Maximale Anzahl Kanäle

Siehe [Zeilengruppen der Konfigurationsdatei für die verteilte Steuerung von Warteschlangen](#) .

Systemerweiterungen und Benutzerexitprogramme

In der Kanaldefinition wird eine Funktion zur Verfügung gestellt, damit zusätzliche Programme während der Verarbeitung von Nachrichten zu definierten Zeitpunkten ausgeführt werden können. Diese Programme werden nicht mit WebSphere MQ bereitgestellt, können jedoch von jeder Installation entsprechend den lokalen Anforderungen bereitgestellt werden.

Damit diese Benutzerexitprogramme ausgeführt werden können, müssen sie über vordefinierte Namen verfügen und im Aufruf an die Kanalprogramme verfügbar sein. Die Namen der Benutzerexitprogramme sind in den Nachrichtenkanaldefinitionen enthalten.

Es gibt eine definierte Steuerblockschnittstelle für die Übergabe der Steuerung an diese Programme und für die Steuerung der Rückgabe der Steuerung von diesen Programmen.

Die genauen Stellen, an denen diese Programme aufgerufen werden, sowie Details zu Steuerblöcken und Namen finden Sie im Abschnitt [Kanalexitprogramme für Messaging-Kanäle](#) .

Kanäle und Empfangsprogramme als vertrauenswürdige Anwendungen ausführen

Wenn die Leistung eine wichtige Überlegung in Ihrer Umgebung ist und Ihre Umgebung stabil ist, können Sie Ihre Kanäle und Empfangsprogramme unter Verwendung der FASTPATH-Bindung als vertrauenswürdig ausführen. Es gibt zwei Faktoren, die beeinflussen, ob Kanäle und Empfangsprogramme als vertrauenswürdig ausgeführt werden:

- Die Umgebungsvariable MQ_CONNECT_TYPE=FASTPATH oder MQ_CONNECT_TYPE = STANDARD. Hierbei wird die Groß-/Kleinschreibung beachtet. Wenn Sie einen Wert angeben, der nicht gültig ist, wird er ignoriert.
- MQIBindType in der Zeilengruppe 'Channels' der qm.ini oder der Registry-Datei. Sie können diese Einstellung auf FASTPATH oder STANDARD setzen und die Groß-/Kleinschreibung muss nicht beachtet werden. Der Standardwert ist STANDARD.

Sie können MQIBindType in Verbindung mit der Umgebungsvariablen verwenden, um den erforderlichen Effekt wie folgt zu erzielen:

MQIBindType	Umgebungsvariable	Ergebnis
STANDARD	NICHT DEFINIERT	STANDARD
FASTPATH	NICHT DEFINIERT	FASTPATH
STANDARD	STANDARD	STANDARD
FASTPATH	STANDARD	STANDARD
STANDARD	FASTPATH	STANDARD
FASTPATH	FASTPATH	FASTPATH
STANDARD	CLIENT	CLIENT
FASTPATH	CLIENT	STANDARD
STANDARD	LOKAL	STANDARD
FASTPATH	LOKAL	STANDARD

In der Zusammenfassung gibt es nur zwei Möglichkeiten, Kanäle und Empfangsprogramme als vertrauenswürdig zu nutzen:

1. Geben Sie MQIBindType=FASTPATH in qm.ini oder in der Registrierungsdatenbank an und geben Sie die Umgebungsvariable nicht an.

2. Geben Sie MQIBindType=FASTPATH in qm.ini oder Registry an und setzen Sie die Umgebungsvariable auf FASTPATH.

Ziehen Sie die Ausführung von Empfangsprogrammen als vertrauenswürdig in Betracht, da Empfangsprogramme stabile Prozesse sind. Ziehen Sie die Ausführung von Kanälen als vertrauenswürdig in Betracht, es sei denn, Sie verwenden instabile Kanalexits oder den Befehl STOP CHANNEL MODE (TERMINATE).

Kanäle in UNIX, Linux, and Windows überwachen und steuern

Für DQM müssen Sie die Kanäle zu fernen Warteschlangenmanagern erstellen, überwachen und steuern. Sie können Kanäle mit Befehlen, Programmen, IBM WebSphere MQ Explorer, Dateien für die Kanaldefinitionen und einem Speicherbereich für Synchronisationsinformationen steuern.

Sie können die folgenden Arten von Befehlen verwenden:

Die IBM WebSphere MQ -Befehle (MQSC)

Sie können den MQSC als einzelne Befehle in einer MQSC-Sitzung in Windows-, UNIX and Linux -Systemen verwenden. Wenn Sie komplexere oder mehrere Befehle ausgeben möchten, kann der MQSC in eine Datei integriert werden, die Sie dann über die Befehlszeile ausführen. Weitere Informationen finden Sie in [MQSC-Befehle](#). Dieser Abschnitt enthält einige einfache Beispiele für die Verwendung von MQSC für die verteilte Steuerung von Warteschlangen.

Bei den Kanalbefehlen handelt es sich um eine Untergruppe der IBM WebSphere MQ-Befehle (MQSC). Sie verwenden MQSC und die Steuerbefehle wie folgt:

- Kanaldefinitionen erstellen, kopieren, anzeigen, ändern und löschen
- Kanäle starten und stoppen, Pingsignal absetzen, Kanalfolgenummern zurücksetzen und unbestätigte Nachrichten auflösen, wenn Links nicht erneut aufgebaut werden können
- Statusinformationen zu Kanälen anzeigen

Steuerbefehle

Sie können auch *Steuerbefehle* in der Befehlszeile für einige dieser Funktionen ausgeben. Ausführliche Informationen hierzu finden Sie unter [Steuerbefehle](#).

Programmierbare Befehlsformatbefehle

Weitere Informationen finden Sie in [PCF-Befehle](#).

IBM WebSphere MQ Explorer

Auf UNIX-, Linux -und Windows -Systemen können Sie die IBM WebSphere MQ Explorer verwenden. Dies stellt eine grafische Verwaltungsschnittstelle zur Verfügung, mit der Verwaltungstasks als Alternative zur Verwendung von Steuerbefehlen oder MQSC-Befehlen ausgeführt werden können. Kanaldefinitionen werden als WS-Manager-Objekte gehalten.

Jeder WS-Manager verfügt über eine DQM-Komponente zur Steuerung von Verbindungen zu kompatiblen fernen Warteschlangenmanagern. Ein Speicherbereich enthält Folgenummern und *IDs der logischen Arbeitseinheit (Logical Unit of Work, LUW)*. Diese werden für Kanalsynchronisationszwecke verwendet.

Eine Liste der Funktionen, die Ihnen bei der Einrichtung und Steuerung von Nachrichtenkanälen mit den verschiedenen Befehlstypen zur Verfügung stehen, finden Sie in [Tabelle 8 auf Seite 81](#).

Zugehörige Konzepte

[„Erste Schritte mit Objekten“ auf Seite 83](#)

Kanäle müssen definiert sein, und die zugehörigen Objekte müssen vorhanden und verfügbar sein, bevor ein Kanal gestartet werden kann. In diesem Abschnitt wird gezeigt, wie.

[„Kommunikation für Windows einrichten“ auf Seite 90](#)

Wenn ein Verwaltungskanal für die verteilte Steuerung von Warteschlangen gestartet wird, versucht er, die in der Kanaldefinition angegebene Verbindung zu verwenden. Damit dies gelingt, ist es erforderlich, dass die Verbindung definiert und verfügbar ist. In diesem Abschnitt wird erläutert, wie dies mithilfe einer der vier Kommunikationsformen für WebSphere MQ für Windows -Systeme erfolgen kann.

[„Kommunikation auf UNIX and Linux -Systemen einrichten“ auf Seite 100](#)

DQM ist eine ferne Warteschlangenfunktion für IBM WebSphere MQ. Es stellt Kanalsteuerprogramme für den Warteschlangenmanager zur Verfügung, die die Schnittstelle zu Kommunikationsverbindungen

bilden, die vom Systembediener gesteuert werden können. Die Kanaldefinitionen, die von der Verwaltung der verteilten Steuerung von Warteschlangen gehalten werden, verwenden diese Verbindungen.

Zugehörige Verweise

[Kanalprogramme auf UNIX-, Linux- und Windows-Systemen](#)

[Beispiel einer Nachrichtenkanalplanung für verteilte Plattformen](#)

[Beispielkonfigurationsdaten](#)

[Kanalattribute](#)

Erforderliche Funktionen für die Einrichtung und Steuerung von Kanälen

Es kann eine Reihe von IBM WebSphere MQ-Funktionen erforderlich sein, um Kanäle einzurichten und zu steuern. Die Kanalfunktionen werden in diesem Thema erläutert.

Sie können eine Kanaldefinition mit den von IBM WebSphere MQ bereitgestellten Standardwerten erstellen. Geben Sie dabei den Namen des Kanals, den Typ des zu erstellenden Kanals, die zu verwendende Übertragungsmethode, den Namen der Übertragungswarteschlange und den Verbindungsnamen an.

Der Kanalname muss an beiden Enden des Kanals identisch sein und innerhalb des Netzes eindeutig sein. Die verwendbaren Zeichen müssen jedoch auf diejenigen eingeschränkt werden, die für IBM WebSphere MQ-Objektnamen gültig sind.

Informationen zu anderen kanalbezogenen Funktionen finden Sie in den folgenden Abschnitten:

- [„Erste Schritte mit Objekten“ auf Seite 83](#)
- [„Erstellen von zugeordneten Objekten“ auf Seite 83](#)
- [„Standardobjekte erstellen“ auf Seite 84](#)
- [„Kanal erstellen“ auf Seite 84](#)
- [„Anzeigen eines Kanals“ auf Seite 85](#)
- [„Kanalstatus anzeigen“ auf Seite 85](#)
- [„Links mit Ping überprüfen“ auf Seite 86](#)
- [„Kanal starten“ auf Seite 86](#)
- [„-Kanal stoppen“ auf Seite 88](#)
- [„Kanal umbenennen“ auf Seite 88](#)
- [„Kanal zurücksetzen“ auf Seite 89](#)
- [„Unbestätigte Nachrichten in einem Kanal auflösen“ auf Seite 89](#)

In [Tabelle 8 auf Seite 81](#) ist die komplette Liste der IBM WebSphere MQ-Funktionen angezeigt, die Sie eventuell benötigen.

<i>Tabelle 8. In UNIX, Linux, and Windows-Systemen erforderliche Funktionen</i>			
Funktion	Steuerbefehle	MQSC	WebSphere MQ Explorer-Entsprechung?
WS-Managerfunktionen			
Warteschlangenmanager ändern		ALTER QMGR	Ja
Warteschlangenmanager erstellen	crtmqm		Ja
WS-Manager löschen	dlmqm		Ja
WS-Manager anzeigen		ANZEIGEN QMGR	Ja
WS-Manager beenden	endmqm		Ja
Ping-WS-Manager		PING QMGR	Nein

Tabelle 8. In UNIX, Linux, and Windows-Systemen erforderliche Funktionen (Forts.)			
Funktion	Steuerbefehle	MQSC	WebSphere MQ Explorer-Entsprechung?
WS-Manager starten	strmqm		Ja
Befehlsserverfunktionen			
Befehlsserver anzeigen	dspmqcsv		Nein
Befehlsserver beenden	endmqcsv		Nein
Befehlsserver starten	strmqcsv		Nein
Warteschlangenfunktionen			
Warteschlange ändern		ALTER QALIAS ALTER QLOCAL ALTER QMODEL ALTER QREMOTE Siehe ALTER-Warteschlangen .	Ja
Warteschlange löschen		CLEAR QLOCAL	Ja
Warteschlange erstellen		DEFINE QALIAS DEFINE QLOCAL DEFINE QMODEL DEFINE QREMOTE Siehe DEFINE-Warteschlangen .	Ja
Warteschlange löschen		DELETE QALIAS DELETE QLOCAL DELETE QMODEL DELETE QREMOTE Siehe DELETE-Warteschlangen .	Ja
Warteschlange anzeigen		ANZEIGEN QUEUE	Ja
Prozessfunktionen			
Änderungsprozess		ALTER PROCESS	Ja
Prozess erstellen		DEFINE PROCESS	Ja
Prozess löschen		DELETE PROCESS	Ja
Anzeigeprozess		ANZEIGEN PROZESS	Ja
Kanalfunktionen			
Kanal ändern		ALTER CHANNEL	Ja
Kanal erstellen		DEFINE CHANNEL	Ja
Kanal löschen		DELETE CHANNEL	Ja
Kanal anzeigen		ANZEIGEN CHANNEL	Ja

Tabelle 8. In UNIX, Linux, and Windows-Systemen erforderliche Funktionen (Forts.)

Funktion	Steuerbefehle	MQSC	WebSphere MQ Explorer-Entsprechung?
Kanalstatus anzeigen		<u>ANZEIGEN CHSTATUS</u>	Ja
Kanal beenden		<u>STOP CHANNEL</u>	Ja
Pingkanal		<u>Pingkanal</u>	Ja
Kanal zurücksetzen		<u>Kanal zurücksetzen</u>	Ja
Kanal auflösen		<u>Auflösungskanal</u>	Ja
Kanal ausführen	<u>runmqchl</u>	<u>START CHANNEL</u>	Ja
Kanalinitiator ausführen	<u>runmqchi</u>	<u>START CHINIT</u>	Nein
Listener ausführen ¹	<u>runmqlsr</u>	<u>START LISTENER</u>	Nein
Listener beenden	endmqlsr (nur Windows -Systeme, AIX, HP-UX und Solaris)		Nein
Anmerkung:			
1. Ein Listener kann beim Start des Warteschlangenmanagers automatisch gestartet werden.			

Erste Schritte mit Objekten

Kanäle müssen definiert sein, und die zugehörigen Objekte müssen vorhanden und verfügbar sein, bevor ein Kanal gestartet werden kann. In diesem Abschnitt wird gezeigt, wie.

Verwenden Sie die WebSphere MQ -Befehle (MQSC) oder den IBM WebSphere MQ Explorer für folgende Zwecke:

1. Nachrichtenkanäle und zugehörige Objekte definieren
2. Nachrichtenkanäle überwachen und steuern

Zu den zugeordneten Objekten, die Sie möglicherweise definieren müssen, gehören:

- Übertragungswarteschlangen
- Definitionen ferner Warteschlangen
- WS-Manager-Aliasdefinitionen
- Aliasnamendefinitionen für Antwortwarteschlange
- Antwort-in lokale Warteschlangen
- Prozesse für Triggerung (MCAs)
- Nachrichtenkanaldefinitionen

Die jeweilige Kommunikationsverbindung für jeden Kanal muss definiert und verfügbar sein, bevor ein Kanal ausgeführt werden kann. Eine Beschreibung der Definition von LU 6.2-, TCP/IP-, NetBIOS-, SPX- und DECnet-Links finden Sie in dem jeweiligen Kommunikationshandbuch für Ihre Installation. Siehe auch [Beispielkonfigurationsdaten](#).

Weitere Informationen zum Erstellen und Arbeiten mit Objekten finden Sie in den folgenden Unterabschnitten:

Erstellen von zugeordneten Objekten

MQSC wird zum Erstellen von zugeordneten Objekten verwendet.

Verwenden Sie MQSC zum Erstellen der Warteschlangen- und Aliasobjekte: Übertragungswarteschlangen, Definitionen für ferne Warteschlangen, Definitionen von WS-Manager-Aliasnamen, Antwortwarteschlangen-Aliasdefinitionen und Antwort-in-lokale Warteschlangen.

Erstellen Sie außerdem die Definitionen von Prozessen für die Auslösung (MCAs) auf ähnliche Weise.

Ein Beispiel für die Erstellung aller erforderlichen Objekte finden Sie im Abschnitt [Beispiel für die Nachrichtenkanalplanung für verteilte Plattformen](#).

Standardobjekte erstellen

Standardobjekte werden automatisch erstellt, wenn ein Warteschlangenmanager erstellt wird. Bei diesen Objekten handelt es sich um Warteschlangen, Kanäle, eine Prozessdefinition und Verwaltungswarteschlangen. Nachdem die Standardobjekte erstellt wurden, können Sie sie jederzeit ersetzen, indem Sie den Befehl `strmqm` mit der Option `-c` ausführen.

Wenn Sie den Befehl `crtmqm` zum Erstellen eines Warteschlangenmanagers verwenden, leitet der Befehl auch ein Programm ein, um eine Gruppe von Standardobjekten zu erstellen.

1. Jedes Standardobjekt wird wiederum erstellt. Das Programm protokolliert, wie viele Objekte erfolgreich definiert wurden, wie viele vorhanden waren und ersetzt wurden, und wie viele erfolglose Versuche es gab.
2. Das Programm zeigt die Ergebnisse an, und wenn Fehler aufgetreten sind, leitet Sie das entsprechende Fehlerprotokoll auf die Details zu.

Wenn die Ausführung des Programms beendet ist, können Sie den Warteschlangenmanager mit dem Befehl `strmqm` starten.

Weitere Informationen zu den Befehlen `'crtmqm'` und `'strmqm'` finden Sie unter [Steuerbefehle](#).

Ändern der Standardobjekte

Wenn Sie die Option `-c` angeben, wird der Warteschlangenmanager vorübergehend gestartet, während die Objekte erstellt werden, und wird anschließend wieder heruntergefahren. Wenn Sie `strmqm` mit der Option `-c` absetzen, werden vorhandene Systemobjekte mit den Standardwerten aktualisiert (z. B. wird das Attribut `MCAUSER` einer Kanaldefinition auf Leerzeichen gesetzt). Sie müssen den Befehl `strmqm` erneut verwenden, ohne die Option `-c` zu verwenden, wenn Sie den Warteschlangenmanager starten wollen.

Wenn Sie die Standardobjekte ändern möchten, können Sie eine eigene Version der alten Datei `'amqscoma.tst'` erstellen und diese bearbeiten.

Kanal erstellen

Erstellen Sie **zwei** Kanaldefinitionen, eine an jedem Ende der Verbindung. Sie erstellen die erste Kanaldefinition auf dem ersten Warteschlangenmanager. Anschließend erstellen Sie die zweite Kanaldefinition am zweiten WS-Manager am anderen Ende der Verbindung.

Beide Enden müssen mit **demselben** Kanalnamen definiert werden. Die beiden Enden müssen **kompatible** Kanaltypen aufweisen, z. B. Sender und Empfänger.

Verwenden Sie den MQSC-Befehl `DEFINE CHANNEL`, um eine Kanaldefinition für ein Ende der Verbindung zu erstellen. Geben Sie den Namen des Kanals, den Kanaltyp für dieses Ende der Verbindung, einen Verbindungsnamen, eine Beschreibung (falls erforderlich), den Namen der Übertragungswarteschlange (falls erforderlich) und das Übertragungsprotokoll an. Geben Sie außerdem alle anderen Attribute an, die von den Systemstandardwerten für den erforderlichen Kanaltyp verschieden sein sollen, und verwenden Sie dabei die zuvor erfassten Informationen.

Sie erhalten Hilfe bei der Entscheidung über die Werte der Kanalattribute in [Kanalattribute](#).

Anmerkung: Es wird empfohlen, alle Kanäle in Ihrem Netzwerk eindeutig zu benennen. Dies ist eine gute Möglichkeit, die Namen der Quellen- und Zielwarteschlangenmanager in den Kanalnamen zu berücksichtigen.

Kanalbeispiel erstellen

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) +  
DESCR('Sender channel to QM2') +  
CONNNAME(QM2) TRPTYPE(TCP) XMITQ(QM2) CONVERT(YES)
```

In allen MQSC-Beispielen wird der Befehl so angezeigt, wie er in einer Befehlsdatei erscheint und wie er in Windows -oder UNIX -oder Linux -Systemen eingegeben wird. Die beiden Methoden sehen identisch aus, es sei denn, dass Sie einen Befehl interaktiv absetzen müssen. Sie müssen zunächst eine MQSC-Sitzung starten. Geben Sie `runmqsc` für den Standardwarteschlangenmanager ein, oder `runmqsc qmname`, wobei *qmname* für den Namen des erforderlichen Warteschlangenmanagers steht. Geben Sie dann eine beliebige Anzahl Befehle ein, wie in den Beispielen gezeigt.

Beschränken Sie die Zeilenlänge Ihrer Befehle auf 72 Zeichen, um die Portierbarkeit zu begrenzen. Verwenden Sie das Verkettungszeichen `+`, wie dargestellt, um über mehr als eine Zeile fortzufahren. Unter Windows verwenden Sie `Strg-z`, um den Eintrag in der Befehlszeile zu beenden. Verwenden Sie auf UNIX and Linux -Systemen die Tastenkombination `Strg + d`. Alternativ dazu können Sie auf UNIX-, Linux -oder Windows -Systemen den Befehl **end** verwenden.

Anzeigen eines Kanals

Verwenden Sie den WebSphere MQ-Scriptbefehl `DISPLAY CHANNEL`, um die Attribute eines Kanals anzuzeigen.

Der Parameter `ALL` des Befehls `DISPLAY CHANNEL` wird standardmäßig angenommen, wenn keine bestimmten Attribute angefordert werden, und der angegebene Kanalname nicht generisch ist.

Die Attribute werden in [Kanalattribute](#) beschrieben.

Kanalbeispiele anzeigen

```
DISPLAY CHANNEL(QM1.TO.QM2) TRPTYPE,CONVERT  
DISPLAY CHANNEL(QM1.TO.*) TRPTYPE,CONVERT  
DISPLAY CHANNEL(*) TRPTYPE,CONVERT  
DISPLAY CHANNEL(QM1.TO.QMR34) ALL
```

Kanalstatus anzeigen

Verwenden Sie den WebSphere MQ-Scriptbefehl `DISPLAY CHSTATUS`, geben Sie den Kanalnamen an und geben Sie an, ob der aktuelle Status von Kanälen oder der Status von gespeicherten Informationen angezeigt werden soll.

`DISPLAY CHSTATUS` gilt für alle Nachrichtenkanäle. Sie gilt nicht für MQI-Kanäle, die keine Serververbindungskanäle sind.

Zu den angezeigten Informationen gehören:

- Kanalname
- Kommunikationsverbindungsname
- Unbestätigte Kanalstatus (falls erforderlich)
- Letzte Folgenummer
- Name der Übertragungswarteschlange (falls erforderlich)
- Die unbestätigte ID (falls erforderlich)
- Die zuletzt festgeschriebene Folgenummer
- ID der logischen Arbeitseinheit
- Prozess-ID
- Thread-ID (nur Windows)

Kanalstatusbeispiele anzeigen

```
DISPLAY CHSTATUS(*) CURRENT
DISPLAY CHSTATUS(QM1.TO.*) SAVED
```

Der gespeicherte Status gilt erst dann, wenn mindestens ein Nachrichtenstachsatz auf dem Kanal übertragen wurde. Der Status wird auch gespeichert, wenn ein Kanal gestoppt wird (mit dem Befehl STOP CHL) und wenn der Warteschlangenmanager beendet wird.

Links mit Ping überprüfen

Verwenden Sie den MQSC-Befehl PING CHANNEL, um eine feste Datennachricht mit dem fernen Ende auszutauschen.

Ping gibt dem Systembetreuer ein wenig Vertrauen, dass der Link verfügbar und funktionsfähig ist.

Ping bedeutet nicht die Verwendung von Übertragungswarteschlangen und Zielwarteschlangen. Sie verwendet Kanaldefinitionen, die zugehörige Kommunikationsverbindung und die Netzkonfiguration. Es kann nur verwendet werden, wenn der Kanal momentan nicht aktiv ist.

Es ist nur über Sender- und Serverkanäle verfügbar. Der entsprechende Kanal wird an der fernen Seite des Links gestartet und führt die Startparametervereinbarung aus. Fehler werden normal benachrichtigt.

Das Ergebnis des Nachrichtenaustauschs wird als Ping complete oder als Fehlernachricht dargestellt.

Ping mit LU 6.2

Wenn Ping aufgerufen wird, wird standardmäßig keine Benutzer-ID oder kein Kennwort an das empfangende Ende fließen. Wenn die Benutzer-ID und das Kennwort erforderlich sind, können sie am einleitenden Ende in der Kanaldefinition erstellt werden. Wenn ein Kennwort in die Kanaldefinition eingegeben wird, wird es vor dem Speichern von WebSphere MQ verschlüsselt. Anschließend wird sie entschlüsselt, bevor sie über den Datenaustausch fließt.

Kanal starten

Verwenden Sie den MQSC-Befehl START CHANNEL für Sender-, Server- und Requesterkanäle. Damit Anwendungen Nachrichten austauschen können, müssen Sie ein Empfangsprogramm für eingehende Verbindungen starten.

START CHANNEL ist nicht erforderlich, wenn ein Kanal mit Warteschlangenmanagerauslösung konfiguriert wurde.

Nach dem Start liest der sendende MCA die Kanaldefinitionen und öffnet die Übertragungswarteschlange. Es wird eine Kanalstartsequenz ausgegeben, die den entsprechenden Nachrichtenkanalserver (MCA) des Empfängers oder Serverkanals über Remotezugriff startet. Wenn sie gestartet wurden, warten die Absender- und Serverprozesse auf Nachrichten, die in die Übertragungswarteschlange eintreffen und sie bei ihrer Ankunft übertragen.

Wenn Sie als Threads Trigger- oder Ausführungskanäle verwenden, stellen Sie sicher, dass der Kanalinitiator für die Überwachung der Initialisierungswarteschlange verfügbar ist. Der Kanalinitiator wird standardmäßig als Teil des Warteschlangenmanagers gestartet.

TCP und LU 6.2 stellen jedoch andere Funktionen bereit:

- Für TCP auf UNIX und Linux-Systemen kann inetd so konfiguriert werden, dass ein Kanal gestartet wird. inetd wird als separater Prozess gestartet.
- Konfigurieren Sie für LU 6.2 in UNIX und Linux-Systemen Ihr SNA-Produkt so, dass der LU 6.2-Responder-Prozess gestartet wird.

- Für LU 6.2 in Windows -Systemen können Sie mit dem SNA-Server TpStart (ein Dienstprogramm, das mit dem SNA-Server bereitgestellt wird) einen Kanal starten. TpStart wird als separater Prozess gestartet.

Die Verwendung der Option Start bewirkt, dass der Kanal bei Bedarf immer resynchronisiert wird.

Damit der Start erfolgreich ist:

- Kanaldefinitionen, lokale und ferne, müssen vorhanden sein. Wenn für einen Empfänger-oder Serververbindungskanal keine entsprechende Kanaldefinition vorhanden ist, wird automatisch ein Standardkanal erstellt, wenn der Kanal automatisch definiert wird. Siehe [Exitprogramm für die automatische Kanaldefinition \(Channel Auto-Definition\)](#)
- Die Übertragungswarteschlange muss vorhanden sein, und sie dürfen keine anderen Kanäle verwenden.
- MCAs, lokale und ferne, müssen vorhanden sein.
- Die Kommunikationsverbindung muss verfügbar sein.
- WS-Manager müssen aktiv, lokal und fern ausgeführt werden.
- Der Nachrichtenkanal darf nicht bereits aktiv sein.

Es wird eine Nachricht an die Anzeige zurückgegeben, in der bestätigt wird, dass die Anforderung zum Starten eines Kanals akzeptiert wurde. Überprüfen Sie zur Bestätigung, dass der Startbefehl erfolgreich war, das Fehlerprotokoll, oder verwenden Sie DISPLAY CHSTATUS. Die Fehlerprotokolle sind:

Windows

`MQ_INSTALLATION_PATH\qmgrs\qmname\errors\AMQERR01.LOG` (für jeden WS-Manager mit dem Namen qmname)

`MQ_INSTALLATION_PATH\qmgrs\@SYSTEM\errors\AMQERR01.LOG` (für allgemeine Fehler)

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

Anmerkung: Auf Windows -Systemen wird weiterhin eine Nachricht im Anwendungsereignisprotokoll von Windows angezeigt.

UNIX and Linux-Systeme

`/var/mqm/qmgrs/qmname/errors/AMQERR01.LOG` (für jeden WS-Manager mit dem Namen qmname)

`/var/mqm/qmgrs/@SYSTEM/errors/AMQERR01.LOG` (für allgemeine Fehler)

Verwenden Sie auf Windows- UNIX and Linux -Systemen den Befehl `runmqclsr`, um den WebSphere MQ -Empfangsprogrammprozess zu starten. Standardmäßig bewirkt jeder eingehende Anforderungen für den Kanalanschluss, dass der Listenerprozess MCAs als Threads des Prozesses 'amqrmppa' startet.

```
runmqclsr -t tcp -m QM2
```

Für abgehende Verbindungen müssen Sie den Kanal auf eine der folgenden drei Arten starten:

1. Verwenden Sie den MQSC-Befehl `START CHANNEL`, und geben Sie dabei den Kanalnamen an, um den Kanal als Prozess oder als Thread zu starten, abhängig vom Parameter `MCATYPE`. (Wenn Kanäle als Threads gestartet werden, handelt es sich um Threads eines Kanalinitiators.)

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

2. Verwenden Sie den Steuerbefehl `runmqchl`, um den Kanal als Prozess zu starten.

```
runmqchl -c QM1.TO.QM2 -m QM1
```

3. Verwenden Sie den Kanalinitiator, um den Kanal auszulösen.

-Kanal stoppen

Verwenden Sie den MQSC-Befehl STOP CHANNEL, um den Kanal anzufordern, die Aktivität zu stoppen. Der Kanal startet keinen neuen Stapel von Nachrichten, bis der Bediener den Kanal erneut startet.

Informationen zum erneuten Starten von gestoppten Kanälen finden Sie in [„Gestoppte Kanäle erneut starten“](#) auf Seite 69.

Dieser Befehl kann an einen Kanal jedes Typs mit Ausnahme von MQCHT_CLNTCONN ausgegeben werden.

Sie können den Typ des erforderlichen Stoppes auswählen:

Quiesce-Beispiel stoppen

```
STOP CHANNEL(QM1.TO.QM2) MODE(QUIESCE)
```

Mit diesem Befehl wird der Kanal in geordneter Weise geschlossen, um sie ordnungsgemäß zu schließen. Der aktuelle Nachrichtenstapel ist abgeschlossen, und die Synchronisationspunktprozedur wird mit dem anderen Ende des Kanals ausgeführt. Wenn der Kanal inaktiv ist, beendet dieser Befehl keinen empfangenden Kanal.

Beispiel für Stoppkraft

```
STOP CHANNEL(QM1.TO.QM2) MODE(FORCE)
```

Mit dieser Option wird der Kanal sofort gestoppt, aber der Thread oder der Prozess des Kanals wird nicht beendet. Der Kanal beendet die Verarbeitung des aktuellen Nachrichtenstroms nicht vollständig und kann daher den Kanal im Zweifel lassen. Im Allgemeinen sollten Sie die Option für Stilllegung in den Wartemodus (Quiesce stop

Stoppbeispiel stoppen

```
STOP CHANNEL(QM1.TO.QM2) MODE(TERMINATE)
```

Mit dieser Option wird der Kanal sofort gestoppt und der Thread oder der Prozess des Kanals beendet.

Stoppbeispiel stoppen (Quiesce)

```
STOP CHANNEL(QM1.TO.QM2) STATUS(STOPPED)
```

Für diesen Befehl ist kein MODE angegeben, daher wird standardmäßig MODE (QUIESCE) angenommen. Er fordert den Kanal auf, den Kanal zu stoppen, damit er nicht automatisch erneut gestartet werden kann, sondern manuell gestartet werden muss.

Inaktives Beispiel stoppen (Quiesce)

```
STOP CHANNEL(QM1.TO.QM2) STATUS(INACTIVE)
```

Für diesen Befehl ist kein MODE angegeben, daher wird standardmäßig MODE (QUIESCE) angenommen. Er fordert die Inaktivierung des Kanals an, so dass er bei Bedarf automatisch erneut gestartet wird.

Kanal umbenennen

Verwenden Sie MQSC, um einen Nachrichtenkanal umzubenennen.

Verwenden Sie MQSC, um die folgenden Schritte auszuführen:

1. Verwenden Sie STOP CHANNEL, um den Kanal zu stoppen.
2. Verwenden Sie DEFINE CHANNEL, um eine doppelte Kanaldefinition mit dem neuen Namen zu erstellen.
3. Verwenden Sie DISPLAY CHANNEL, um zu überprüfen, ob er korrekt erstellt wurde.
4. Verwenden Sie DELETE CHANNEL, um die ursprüngliche Kanaldefinition zu löschen.

Wenn Sie einen Nachrichtenkanal umbenennen möchten, denken Sie daran, dass ein Kanal über **zwei** Kanaldefinitionen verfügt, eine an jedem Ende. Stellen Sie sicher, dass Sie den Kanal an beiden Enden gleichzeitig umbenennen.

Kanal zurücksetzen

Verwenden Sie den MQSC-Befehl RESET CHANNEL, um die Nachrichtenfolgennummer zu ändern.

Der Befehl RESET CHANNEL ist für jeden Nachrichtenkanal verfügbar, aber nicht für MQI-Kanäle (Clientverbindung oder Serververbindung). Die erste Nachricht startet die neue Sequenz, wenn der Kanal das nächste Mal gestartet wird.

Wenn der Befehl auf einem Sender-oder Serverkanal abgesetzt wird, informiert er die andere Seite der Änderung, wenn der Kanal erneut gestartet wird.

Zugehörige Konzepte

„Erste Schritte mit Objekten“ auf Seite 83

Kanäle müssen definiert sein, und die zugehörigen Objekte müssen vorhanden und verfügbar sein, bevor ein Kanal gestartet werden kann. In diesem Abschnitt wird gezeigt, wie.

„Kanalsteuerfunktion“ auf Seite 58

Die Kanalsteuerfunktion stellt Funktionen zur Verfügung, mit der Sie Kanäle definieren, überwachen und steuern können.

„Anwendungen mithilfe der verteilten Steuerung von Warteschlangen verbinden“ auf Seite 29

In diesem Abschnitt finden Sie ausführliche Informationen zur übergreifenden Kommunikation zwischen WebSphere MQ-Installationen, einschließlich Informationen zu Warteschlangendefinitionen, Kanaldefinitionen, Auslösungen und Synchronisationspunktprozeduren.

Zugehörige Verweise

[RESET CHANNEL](#)

Unbestätigte Nachrichten in einem Kanal auflösen

Verwenden Sie den MQSC-Befehl RESOLVE CHANNEL, wenn Nachrichten von einem Sender oder Server im Zweifel gehalten werden. Zum Beispiel, weil ein Ende der Verbindung beendet wurde und es keine Aussicht auf eine Wiederherstellung gibt.

Der Befehl RESOLVE CHANNEL akzeptiert einen von zwei Parametern: BACKOUT oder COMMIT. Mit Backout werden Nachrichten in die Übertragungswarteschlange zurückgespeichert, während Commit sie löscht.

Das Kanalprogramm versucht nicht, eine Sitzung mit einem Partner aufzubauen. Stattdessen bestimmt sie die ID der logischen Arbeitseinheit (LUWID), die die unbestätigte_Nachrichten darstellt. Anschließend gibt es, wie angefordert, folgende Probleme aus:

- BACKOUT, um die Nachrichten in die Übertragungswarteschlange zurückzuspeichern; oder
- COMMIT, um die Nachrichten aus der Übertragungswarteschlange zu löschen.

Damit die Auflösung erfolgreich ist:

- Der Kanal muss inaktiv sein.
- Der Kanal muss im Zweifel sein.
- Der Kanaltyp muss "sender" oder "server" sein
- Es muss eine lokale Kanaldefinition vorhanden sein.
- Der lokale WS-Manager muss aktiv sein.

Zugehörige Konzepte

[„Erste Schritte mit Objekten“](#) auf Seite 83

Kanäle müssen definiert sein, und die zugehörigen Objekte müssen vorhanden und verfügbar sein, bevor ein Kanal gestartet werden kann. In diesem Abschnitt wird gezeigt, wie.

[„Kanalsteuerfunktion“](#) auf Seite 58

Die Kanalsteuerfunktion stellt Funktionen zur Verfügung, mit der Sie Kanäle definieren, überwachen und steuern können.

[„Anwendungen mithilfe der verteilten Steuerung von Warteschlangen verbinden“](#) auf Seite 29

In diesem Abschnitt finden Sie ausführliche Informationen zur übergreifenden Kommunikation zwischen WebSphere MQ-Installationen, einschließlich Informationen zu Warteschlangendefinitionen, Kanaldefinitionen, Auslösungen und Synchronisationspunktprozeduren.

Zugehörige Verweise

[GELÖST-CHANNEL](#)

Kommunikation für Windows einrichten

Wenn ein Verwaltungskanal für die verteilte Steuerung von Warteschlangen gestartet wird, versucht er, die in der Kanaldefinition angegebene Verbindung zu verwenden. Damit dies gelingt, ist es erforderlich, dass die Verbindung definiert und verfügbar ist. In diesem Abschnitt wird erläutert, wie dies mithilfe einer der vier Kommunikationsformen für WebSphere MQ für Windows -Systeme erfolgen kann.

Es kann hilfreich sein, den Abschnitt [Beispielkonfiguration- IBM WebSphere MQ für Windows](#) zu lesen.

Informationen für UNIX and Linux-Systeme finden Sie im Abschnitt [„Kommunikation auf UNIX and Linux -Systemen einrichten“](#) auf Seite 100.

Entscheidung über die Verbindungsart

Wählen Sie eine der folgenden vier Formen der Kommunikation für WebSphere MQ für Windows -Systeme aus:

- [„TCP-Verbindung unter Windows definieren“](#) auf Seite 91
- [„Definieren einer LU 6.2 -Verbindung unter Windows“](#) auf Seite 92
- [„NetBIOS -Verbindung unter Windows definieren“](#) auf Seite 94
- [„SPX-Verbindung unter Windows definieren“](#) auf Seite 97 (nur Windows XP und Windows 2003 Server)

Jede Kanaldefinition darf nur ein Protokoll als Attribut für das Übertragungsprotokoll (Transporttyp) angeben. Ein oder mehrere Protokolle können von einem WS-Manager verwendet werden.

Für WebSphere MQ -Clients kann es nützlich sein, alternative Kanäle mit unterschiedlichen Übertragungsprotokollen zu verwenden. Weitere Informationen zu WebSphere MQ -Clients finden Sie unter [Übersicht über Clients](#).

Zugehörige Konzepte

[„Anwendungen mithilfe der verteilten Steuerung von Warteschlangen verbinden“](#) auf Seite 29

In diesem Abschnitt finden Sie ausführliche Informationen zur übergreifenden Kommunikation zwischen WebSphere MQ-Installationen, einschließlich Informationen zu Warteschlangendefinitionen, Kanaldefinitionen, Auslösungen und Synchronisationspunktprozeduren.

[„Kanäle in UNIX, Linux, and Windows überwachen und steuern“](#) auf Seite 80

Für DQM müssen Sie die Kanäle zu fernen Warteschlangenmanagern erstellen, überwachen und steuern. Sie können Kanäle mit Befehlen, Programmen, IBM WebSphere MQ Explorer, Dateien für die Kanaldefinitionen und einem Speicherbereich für Synchronisationsinformationen steuern.

[„Verbindungen zwischen Client und Server konfigurieren“](#) auf Seite 106

Um die Kommunikationsverbindungen zwischen WebSphere MQ MQI-Clients und -Servern zu konfigurieren, müssen Sie Ihr Kommunikationsprotokoll festlegen, die Verbindungen an beiden Enden der Verbindung definieren, ein Empfangsprogramm starten und Kanäle definieren.

TCP-Verbindung unter Windows definieren

Definieren Sie eine TCP-Verbindung, indem Sie einen Kanal auf der sendenden Seite konfigurieren, um die Adresse des Ziels anzugeben, und indem Sie ein Empfangsprogramm auf der Empfangsseite ausführen.

Sendende Beendigung

Geben Sie den Hostnamen oder die TCP-Adresse der Zielmaschine in das Feld Verbindungsname der Kanaldefinition an.

Der Port, zu dem die Verbindung hergestellt werden soll, standardmäßig 1414. Die Portnummer 1414 wird von der Internet Assigned Numbers Authority für IBM WebSphere MQ zugeordnet.

Wenn Sie eine andere Portnummer als die Standardportnummer verwenden möchten, geben Sie sie im Feld für den Verbindungsnamen der Kanalobjektdefinition an:

```
DEFINE CHANNEL('channel name') CHLTYPE(SDR) +  
  TRPTYPE(TCP) +  
  CONNNAME('OS2R0G3(1822)') +  
  XMITQ('XMITQ name') +  
  REPLACE
```

Dabei steht OS2R0G3 für den DNS-Namen des fernen Warteschlangenmanagers und 1822 für den erforderlichen Port. (Dies muss der Port sein, an dem der Listener empfangsbereit ist.)

Ein Laufkanal muss gestoppt und erneut gestartet werden, um eine Änderung an der Kanalobjektdefinition abzuholen.

Sie können die Standardportnummer ändern, indem Sie sie in der Datei `.ini` für IBM WebSphere MQ für Windows angeben:

```
TCP:  
Port=1822
```

Anmerkung: Um die TCP/IP-Portnummer auszuwählen, die verwendet werden soll, verwendet IBM WebSphere MQ die erste Portnummer, die in der folgenden Reihenfolge gefunden wird:

1. Die Portnummer, die explizit in der Kanaldefinition oder Befehlszeile angegeben wurde. Mit dieser Zahl kann die Standardportnummer für einen Kanal außer Kraft gesetzt werden.
2. Das Portattribut, das in der TCP-Zeilengruppe der `.ini`-Datei angegeben ist. Mit dieser Zahl kann die Standardportnummer für einen WS-Manager überschrieben werden.
3. Der Standardwert ist 1414. Dies ist die Nummer, die IBM WebSphere MQ von der Internet Assigned Numbers Authority für eingehende und abgehende Verbindungen zugeordnet ist.

Weitere Informationen zu den Werten, die Sie mit `qm.ini` festlegen, finden Sie unter [Zeilengruppen für Konfigurationsdateien für verteilte Steuerung von Warteschlangen](#).

Empfang auf TCP

Um ein empfangendes Kanalprogramm zu starten, muss ein Empfangsprogramm gestartet werden, um eingehende Netzanforderungen zu erkennen und den zugehörigen Kanal zu starten. Sie können den IBM WebSphere MQ -Listener verwenden.

Das Empfangen von Kanalprogrammen wird als Antwort auf eine Startanforderung vom sendenden Kanal gestartet.

Um ein empfangendes Kanalprogramm zu starten, muss ein Empfangsprogramm gestartet werden, um eingehende Netzanforderungen zu erkennen und den zugehörigen Kanal zu starten. Sie können den IBM WebSphere MQ -Listener verwenden.

Wenn Sie den mit IBM WebSphere MQ bereitgestellten Listener ausführen möchten, der neue Kanäle als Threads startet, verwenden Sie den Befehl `runmqqlsr`.

Ein grundlegendes Beispiel für die Verwendung des Befehls **runmqtsr**:

```
runmqtsr -t tcp [-m QMNAME] [-p 1822]
```

Die eckigen Klammern geben optionale Parameter an; QMNAME ist für den Standard-WS-Manager nicht erforderlich, und die Portnummer ist nicht erforderlich, wenn Sie die Standardeinstellung (1414) verwenden. Die Port-Nummer darf 65535 nicht überschreiten.

Anmerkung: Um die TCP/IP-Portnummer auszuwählen, die verwendet werden soll, verwendet IBM WebSphere MQ die erste Portnummer, die in der folgenden Reihenfolge gefunden wird:

1. Die Portnummer, die explizit in der Kanaldefinition oder Befehlszeile angegeben wurde. Mit dieser Zahl kann die Standardportnummer für einen Kanal außer Kraft gesetzt werden.
2. Das Portattribut, das in der TCP-Zeilengruppe der `.ini`-Datei angegeben ist. Mit dieser Zahl kann die Standardportnummer für einen WS-Manager überschrieben werden.
3. Der Standardwert ist 1414. Dies ist die Nummer, die IBM WebSphere MQ von der Internet Assigned Numbers Authority für eingehende und abgehende Verbindungen zugeordnet ist.

Führen Sie für bestmögliche Leistung den IBM WebSphere MQ-Listener wie im Abschnitt [„Kanäle und Empfangsprogramme als vertrauenswürdige Anwendungen ausführen“](#) auf Seite 79 beschrieben als vertrauenswürdige Anwendung aus. Weitere Informationen zu vertrauenswürdigen Anwendungen finden Sie im Abschnitt [Einschränkungen für vertrauenswürdige Anwendungen](#).

Verwendung der Option "TCP/IP SO_KEEPALIVE"

Wenn Sie die Option Windows SO_KEEPALIVE verwenden möchten, müssen Sie den folgenden Eintrag in Ihrer Registry hinzufügen:

```
TCP:  
KeepAlive=yes
```

Weitere Informationen zur Option SO_KEEPALIVE finden Sie in [„Überprüfen, ob das andere Ende des Kanals noch verfügbar ist“](#) auf Seite 66.

Unter Windows steuert der Registrierungswert `HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters` für die Zeioption `Windows KeepAlive` das Intervall, das vergeht, bevor die Verbindung überprüft wird. Der Standardwert ist zwei Stunden.

Definieren einer LU 6.2-Verbindung unter Windows

SNA muss so konfiguriert werden, dass ein LU 6.2-Dialog zwischen den beiden Maschinen aufgebaut werden kann.

Sobald die SNA konfiguriert ist, fahren Sie wie folgt fort.

Informationen hierzu finden Sie in der folgenden Tabelle.

Ferne Plattform	TPNAME	TPPATH
z/OS oder MVS/ESA ohne CICS	Entsprechendes gilt für die entsprechenden Nebeninformationen zum fernen Warteschlangenmanager.	-
z/OS oder MVS/ESA mit CICS	CKRC (Sender) CKSV (Requester) CKRC (Server)	-
IBM i	Entsprechendes gilt für den Vergleichswert im Routing-Eintrag auf dem IBM i-System.	-

Tabelle 9. Einstellungen auf dem lokalen Windows -System für eine ferne Warteschlangenmanagerplattform (Forts.)

Ferne Plattform	TPNAME	TPPATH
UNIX and Linux-Systeme	Entsprechendes gilt für die entsprechenden Nebeninformationen zum fernen Warteschlangenmanager.	<code>MQ_INSTALLATION_PATH/bin/amqcrs6a</code>
Windows	Wie im Befehl Windows Run Listener angegeben, oder das aufrufbare Transaktionsprogramm, das mit TpSetup unter Windowsdefiniert wurde.	<code>MQ_INSTALLATION_PATH\bin\amqcrs6a</code>

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

Wenn mehrere WS-Manager auf derselben Maschine vorhanden sind, stellen Sie sicher, dass die TPNames in den Kanaldefinitionen eindeutig sind.

Die neuesten Informationen zur Konfiguration von AnyNet SNA über TCP/IP finden Sie in der folgenden Onlinedokumentation zu IBM : [AnyNet SNA über TCP/IP](#) und [SNA-Knotenoperationen](#).

Zugehörige Konzepte

„Sendeseite auf LU 6.2“ auf Seite 93

Erstellen Sie ein CPI-C-Nebenobjekt (symbolisches Ziel) aus der Verwaltungsanwendung des LU 6.2-Produkts, das Sie verwenden. Geben Sie diesen Namen in das Feld Verbindungsname in der Kanaldefinition ein. Erstellen Sie außerdem einen LU 6.2-Link zu dem Partner.

„Empfangen auf LU 6.2“ auf Seite 93

Das Empfangen von Kanalprogrammen wird als Antwort auf eine Startanforderung vom sendenden Kanal gestartet.

Sendeseite auf LU 6.2

Erstellen Sie ein CPI-C-Nebenobjekt (symbolisches Ziel) aus der Verwaltungsanwendung des LU 6.2-Produkts, das Sie verwenden. Geben Sie diesen Namen in das Feld Verbindungsname in der Kanaldefinition ein. Erstellen Sie außerdem einen LU 6.2-Link zu dem Partner.

Geben Sie im CPI-C-Nebenobjekt den Partner-LU-Namen auf der empfangenden Maschine, den TP-Namen und den Modusnamen ein. Beispiel:

```
Partner LU Name      OS2R0G2
Partner TP Name     recv
Mode Name           #INTER
```

Empfangen auf LU 6.2

Das Empfangen von Kanalprogrammen wird als Antwort auf eine Startanforderung vom sendenden Kanal gestartet.

Um ein empfangendes Kanalprogramm zu starten, muss ein Empfangsprogramm gestartet werden, um eingehende Netzanforderungen zu erkennen und den zugehörigen Kanal zu starten. Sie starten dieses Listenerprogramm mit dem Befehl `RUNMQLSR` und geben den `TpName` an, auf dem empfangsbereit ist. Alternativ können Sie `TpStart` unter SNA Server für Windows verwenden.

Befehl RUNMQLSR verwenden

Beispiel für den Befehl zum Starten des Listeners:

```
RUNMQLSR -t LU62 -n RECV [-m QMNAME]
```

Dabei ist RECV der TpName, der auf der anderen Seite (Senden) als "TpName to start on the remote side" angegeben wird. Das letzte Teil in eckigen Klammern ist optional und ist für den Standardwarteschlangenmanager nicht erforderlich.

Es ist möglich, mehr als einen Warteschlangenmanager auf einer Maschine auszuführen. Sie müssen jedem WS-Manager einen anderen TpName zuordnen und anschließend ein Empfangsprogramm für die einzelnen WS-Manager starten. Beispiel:

```
RUNMQLSR -t LU62 -m QM1 -n TpName1  
RUNMQLSR -t LU62 -m QM2 -n TpName2
```

Die beste Leistung erzielen Sie, wenn Sie den WebSphere MQ -Listener als vertrauenswürdige Anwendung ausführen, wie im Abschnitt [Kanäle und Listener als vertrauenswürdige Anwendungen ausführen](#) beschrieben. Informationen zu vertrauenswürdigen Anwendungen finden Sie im Abschnitt [Einschränkungen für vertrauenswürdige Anwendungen](#).

Sie können alle WebSphere MQ -Listener, die auf einem inaktiven Warteschlangenmanager ausgeführt werden, mit folgendem Befehl stoppen:

```
ENDMQLSR [-m QMNAME]
```

Microsoft SNA Server unter Windows verwenden

Sie können TpSetup (über das SNA-Server-SDK) verwenden, um ein aufrufbares TP zu definieren, das dann amqcrs6a.exe steuert, oder Sie können verschiedene Registrierungswerte manuell festlegen. Die Parameter, die an amqcrs6a.exe übergeben werden sollen, sind:

```
-m QM -n TpName
```

Hierbei steht *QM* für den Namen des WS-Managers und *TpName* für den TP-Namen. Weitere Informationen finden Sie im Handbuch *Microsoft SNA Server APPC Programmers Guide* oder im Handbuch *Microsoft SNA Server CPI-C Programmers Guide*.

Wenn Sie keinen Warteschlangenmanagernamen angeben, wird der Standardwarteschlangenmanager angenommen.

NetBIOS -Verbindung unter Windows definieren

WebSphere MQ verwendet drei Typen von NetBIOS -Ressourcen, wenn eine NetBIOS -Verbindung zu einem anderen WebSphere MQ -Produkt hergestellt wird: Sitzungen, Befehle und Namen. Jede dieser Ressourcen hat einen Grenzwert, der entweder standardmäßig oder nach Auswahl während der Installation von NetBIOS festgelegt wird.

Jeder aktive Kanal, unabhängig vom Typ, verwendet eine NetBIOS-Sitzung und einen NetBIOS-Befehl. Die IBM NetBIOS -Implementierung ermöglicht es mehreren Prozessen, denselben lokalen NetBIOS -Namen zu verwenden. Daher muss nur ein NetBIOS -Name für die Verwendung durch WebSphere MQ verfügbar sein. Andere Anbieterimplementierungen, z. B. Novell's NetBIOS-Emulation, erfordern einen anderen lokalen Namen pro Prozess. Überprüfen Sie Ihre Anforderungen in der Dokumentation für das NetBIOS-Produkt, das Sie verwenden.

Stellen Sie in allen Fällen sicher, dass bereits genügend Ressourcen für die einzelnen Typen verfügbar sind, oder erhöhen Sie die in der Konfiguration angegebenen Maximalwerte. Alle Änderungen an den Werten erfordern einen Systemneustart.

Während des Systemstarts zeigt der NetBIOS-Einheitentreiber die Anzahl der Sitzungen, Befehle und Namen an, die für Anwendungen zur Verfügung stehen. Diese Ressourcen stehen für alle NetBIOS-basierten Anwendungen zur Verfügung, die auf demselben System ausgeführt werden. Daher ist es möglich, dass andere Anwendungen diese Ressourcen nutzen, bevor sie von WebSphere MQ angefordert werden müssen. Ihr LAN-Netzadministrator sollte dies für Sie klären können.

Zugehörige Konzepte

[„Lokalen NetBIOS-Namen für IBM WebSphere MQ definieren“](#) auf Seite 95

Der lokale NetBIOS-Name, der von IBM WebSphere MQ -Kanalprozessen verwendet wird, kann auf drei Arten angegeben werden.

„WS-Manager-NetBIOS-Sitzung, -Befehl und -Namensbegrenzungen einrichten“ auf Seite 95

Die Grenzwerte des Warteschlangenmanagers für NetBIOS-Sitzungen, -Befehle und -Namen können auf zwei Arten angegeben werden.

„LAN-Adapternummer erstellen“ auf Seite 96

Damit Kanäle erfolgreich über NetBIOS ausgeführt werden können, muss die Adapterunterstützung an jedem Ende kompatibel sein. IBM WebSphere MQ ermöglicht Ihnen, die Auswahl der LAN-Adapternummer (LANA) zu steuern, indem Sie den Wert AdapterNum in der NETBIOS-Zeilengruppe Ihrer Datei qm.ini verwenden und den Parameter -a im Befehl runmqslr angeben.

„NetBIOS-Verbindung initialisieren“ auf Seite 96

Definieren der Schritte, die zum Einleiten einer Verbindung erforderlich sind.

„Ziellistener für die NetBIOS-Verbindung“ auf Seite 97

Definieren der Schritte, die am empfangenden Ende der NetBIOS-Verbindung ausgeführt werden sollen.

Lokalen NetBIOS-Namen für IBM WebSphere MQ definieren

Der lokale NetBIOS-Name, der von IBM WebSphere MQ -Kanalprozessen verwendet wird, kann auf drei Arten angegeben werden.

In der Reihenfolge der Vorrangstellung sind die drei folgenden Möglichkeiten:

1. Der Wert, der im Parameter -l des Befehls RUNMQLSR angegeben ist, z. B.:

```
RUNMQLSR -t NETBIOS -l my_station
```

2. Die Umgebungsvariable MQNAME mit einem Wert, der durch den folgenden Befehl festgelegt wird:

```
SET MQNAME=my_station
```

Sie können den MQNAME-Wert für jeden Prozess festlegen. Alternativ können Sie ihn auf einer Systemebene in der Windows-Registry festlegen.

Wenn Sie eine NetBIOS-Implementierung verwenden, die eindeutige Namen erfordert, müssen Sie in jedem Fenster, in dem ein IBM WebSphere MQ -Prozess gestartet wird, einen Befehl SET MQNAME absetzen. Der MQNAME-Wert ist beliebig, aber er muss für jeden Prozess eindeutig sein.

3. Die NETBIOS-Zeilengruppe in der WS-Managerkonfigurationsdatei qm.ini. Beispiel:

```
NETBIOS:  
LocalName=my_station
```

Anmerkung:

1. Aufgrund der Unterschiede bei der Implementierung der unterstützten NetBIOS-Produkte wird empfohlen, die einzelnen NetBIOS-Namen im Netz eindeutig zu machen. Wenn dies nicht der Fall ist, kann es zu unvorhersehbaren Ergebnissen kommen. Wenn bei der Einrichtung eines NetBIOS-Kanals Fehler auftreten und im Fehlerprotokoll des Warteschlangenmanagers Fehlermeldungen angezeigt werden, die einen NetBIOS-Rückkehrcode von X'15 ' aufweisen, überprüfen Sie die Verwendung von NetBIOS-Namen.
2. Unter Windows können Sie Ihren Maschinennamen nicht als NetBIOS-Namen verwenden, da Windows sie bereits verwendet.
3. Für die Initialisierung des Senderkanals ist es erforderlich, dass ein NetBIOS-Name entweder über die Umgebungsvariable MQNAME oder über den lokalen Namen in der Datei qm.ini angegeben wird.

WS-Manager-NetBIOS-Sitzung, -Befehl und -Namensbegrenzungen einrichten

Die Grenzwerte des Warteschlangenmanagers für NetBIOS-Sitzungen, -Befehle und -Namen können auf zwei Arten angegeben werden.

In der Reihenfolge der Vorrangstellung sind folgende Möglichkeiten:

1. Die im Befehl RUNMQLSR angegebenen Werte:

```
-s Sessions
-e Names
-o Commands
```

Wenn der Operand -m nicht im Befehl angegeben ist, gelten die Werte nur für den Standardwarteschlangenmanager.

2. Die NETBIOS-Zeilengruppe in der WS-Managerkonfigurationsdatei qm.ini. Beispiel:

```
NETBIOS:
    NumSess=Qmgr_max_sess
    NumCmds=Qmgr_max_cmds
    NumNames=Qmgr_max_names
```

LAN-Adaptornummer erstellen

Damit Kanäle erfolgreich über NetBIOS ausgeführt werden können, muss die Adapterunterstützung an jedem Ende kompatibel sein. IBM WebSphere MQ ermöglicht Ihnen, die Auswahl der LAN-Adaptornummer (LANA) zu steuern, indem Sie den Wert AdapterNum in der NETBIOS-Zeilengruppe Ihrer Datei qm.ini verwenden und den Parameter -a im Befehl runmqslr angeben.

Die von IBM WebSphere MQ für NetBIOS -Verbindungen verwendete LAN-Standardadapternummer ist 0. Stellen Sie sicher, dass die Nummer auf Ihrem System wie folgt verwendet wird:

Unter Windows ist es nicht möglich, die LAN-Adaptornummer direkt über das Betriebssystem abzurufen. Stattdessen verwenden Sie LANACFG.EXE, verfügbar über Microsoft. Die Ausgabe des Tools zeigt die Nummern des virtuellen LAN-Adapters und die zugehörigen effektiven Bindungen an. Weitere Informationen zu LAN-Adaptornummern finden Sie im Microsoft Knowledge Base-Artikel 138037 *HOWTO: Use LANA Numbers in a 32-bit Environment*.

Geben Sie den korrekten Wert in der Zeilengruppe NETBIOS der WS-Manager-Konfigurationsdatei qm.ini an:

```
NETBIOS:
    AdapterNum=n
```

Dabei steht n für die korrekte LAN-Adaptornummer für dieses System.

NetBIOS-Verbindung initialisieren

Definieren der Schritte, die zum Einleiten einer Verbindung erforderlich sind.

Führen Sie die folgenden Schritte aus, um die Verbindung zu starten:

1. Definieren Sie den NetBIOS-Stationsnamen mit dem Wert MQNAME oder LocalName.
2. Überprüfen Sie, ob die LAN-Adaptornummer auf Ihrem System verwendet wird, und geben Sie die korrekte Datei mit dem AdapterNum an.
3. Geben Sie im Feld ConnectionName der Kanaldefinition den NetBIOS-Namen an, der vom Ziel-Listener-Programm verwendet wird. Unter Windows **müssen** NetBIOS -Kanäle als Threads ausgeführt werden. Geben Sie dazu MCATYPE (THREAD) in der Kanaldefinition an.

```
DEFINE CHANNEL (chname) CHLTYPE(SDR) +
    TRPTYPE(NETBIOS) +
    CONNAME(your_station) +
    XMITQ(xmitq) +
    MCATYPE(THREAD) +
    REPLACE
```


Ziellistener für die NetBIOS-Verbindung

Definieren der Schritte, die am empfangenden Ende der NetBIOS-Verbindung ausgeführt werden sollen.

Führen Sie auf der Empfangsseite die folgenden Schritte aus:

1. Definieren Sie den NetBIOS-Stationsnamen mit dem Wert MQNAME oder LocalName.
2. Überprüfen Sie, ob die LAN-Adapternummer auf Ihrem System verwendet wird, und geben Sie die korrekte Datei mit dem AdapterNum an.
3. Definieren Sie den Empfängerkanal:

```
DEFINE CHANNEL (chname) CHLTYPE(RCVR) +  
TRPTYPE(NETBIOS) +  
REPLACE
```

4. Starten Sie das Empfangsprogramm WebSphere MQ , um die Station einzurichten und eine Verbindung zu ihr herzustellen. Beispiel:

```
RUNMQLSR -t NETBIOS -l your_station [-m qmgr]
```

Mit diesem Befehl wird `your_station` als NetBIOS-Station eingerichtet, die darauf wartet, kontaktiert zu werden. Der NetBIOS-Stationsname muss im gesamten NetBIOS-Netz eindeutig sein.

Um die beste Leistung zu erzielen, führen Sie den WebSphere MQ -Listener als vertrauenswürdige Anwendung aus, wie in „Kanäle und Empfangsprogramme als vertrauenswürdige Anwendungen ausführen“ auf Seite 79 beschrieben. Informationen zu vertrauenswürdigen Anwendungen finden Sie im Abschnitt Einschränkungen für vertrauenswürdige Anwendungen .

Sie können alle WebSphere MQ -Listener, die auf einem inaktiven Warteschlangenmanager ausgeführt werden, mit folgendem Befehl stoppen:

```
ENDMQLSR [-m QMNAME]
```

Wenn Sie keinen Warteschlangenmanagernamen angeben, wird der Standardwarteschlangenmanager angenommen.

SPX-Verbindung unter Windows definieren

Eine SPX-Verbindung findet nur bei einem Client und Server Anwendung, auf denen Windows XP und Windows 2003 Server ausgeführt werden.

Die Kanaldefinition auf der sendenden Seite gibt die Adresse des Ziels an. Auf der Empfangsseite muss ein Empfangsprogramm ausgeführt werden.

Zugehörige Konzepte

„Sendeseite auf SPX“ auf Seite 97

Wenn es sich bei der Zielmaschine um eine ferne Maschine handelt, geben Sie die SPX-Adresse der Zielmaschine im Feld Verbindungsname der Kanaldefinition an.

„Empfangen über SPX“ auf Seite 98

Das Empfangen von Kanalprogrammen wird als Antwort auf eine Startanforderung vom sendenden Kanal gestartet.

„IPX-/SPX-Parameter“ auf Seite 99

In den meisten Fällen werden die Standardeinstellungen für die IPX/SPX-Parameter Ihren Anforderungen entsprechen. Möglicherweise müssen Sie jedoch einige von ihnen in Ihrer Umgebung ändern, um ihre Verwendung für WebSphere MQ zu optimieren.

Sendeseite auf SPX

Wenn es sich bei der Zielmaschine um eine ferne Maschine handelt, geben Sie die SPX-Adresse der Zielmaschine im Feld Verbindungsname der Kanaldefinition an.

Die SPX-Adresse wird in folgendem Format angegeben:

```
network.node(socket)
```

Dabei gilt:

network

die Netzadresse (4 Byte) des Netzes, in dem sich die ferne Maschine befindet,

node

die Knotenadresse (6 Byte), die der LAN-Adresse des LAN-Adapters auf der fernen Maschine entspricht.

socket

die Socketnummer (2 Byte), über die die ferne Maschine empfangsbereit ist.

Wenn sich die lokale und die ferne Maschine in demselben Netz befinden, muss die Netzadresse nicht angegeben werden. Wenn das ferne Ende an dem Standardsocket (5E86) empfangsbereit ist, muss das Socket nicht angegeben werden.

Es folgt ein Beispiel für eine vollständig angegebene SPX-Adresse, die im Parameter CONNAME eines MQSC-Befehls angegeben ist:

```
CONNAME('00000001.08005A7161E5(5E87)')
```

Im Standardfall, bei dem sich die Maschinen beide in demselben Netz befinden, ergibt sich Folgendes:

```
CONNAME(08005A7161E5)
```

Die Standardsocketnummer kann geändert werden, indem sie in der Konfigurationsdatei des Warteschlangenmanagers (qm.ini) angegeben wird:

```
SPX:  
Socket=5E87
```

Weitere Informationen zu den Werten, die Sie mit qm.inifestlegen, finden Sie unter [Zeilengruppen der Konfigurationsdatei für verteilte Steuerung von Warteschlangen](#).

Empfangen über SPX

Das Empfangen von Kanalprogrammen wird als Antwort auf eine Startanforderung vom sendenden Kanal gestartet.

Um ein empfangendes Kanalprogramm zu starten, muss ein Empfangsprogramm gestartet werden, um eingehende Netzanforderungen zu erkennen und den zugehörigen Kanal zu starten.

Verwenden Sie das Empfangsprogramm von WebSphere MQ.

Rückstandsoption des SPX-Listeners verwenden

Beim Empfang auf SPX wird eine maximale Anzahl ausstehender Verbindungsanforderungen festgelegt. Dies kann als *Rückstand* von Anforderungen betrachtet werden, die auf den SPX-Port warten, damit der Listener die Anforderung akzeptiert. Die Standardwerte für das Listener-Rückstandsprotokoll werden in [Tabelle 10 auf Seite 98](#) angezeigt.

Plattform	Standardwert für Listenerückstand
Windows -Server	5
Windows -Workstation	5

Wenn der Rückstand die Werte in [Tabelle 10 auf Seite 98](#) erreicht, wird der Ursachencode MQRC_Q_MGR_NOT_AVAILABLE empfangen, wenn versucht wird, über MQCONN oder MQCONNX eine Verbindung zum Warteschlangenmanager herzustellen. In diesem Fall ist es möglich, erneut zu versuchen, die Verbindung herzustellen.

Um diesen Fehler zu vermeiden, können Sie jedoch einen Eintrag in der Datei qm.ini oder in der Registry für Windowshinzufügen:

```
SPX:  
ListenerBacklog = n
```

Dadurch wird die maximale Standardanzahl ausstehender Anforderungen (siehe [Tabelle 10 auf Seite 98](#)) für den SPX-Listener überschrieben.

Anmerkung: Einige Betriebssysteme unterstützen einen größeren Wert als der Standardwert. Dies kann bei Bedarf verwendet werden, um das Erreichen des Verbindungslimits zu vermeiden.

Führen Sie einen der folgenden Schritte aus, um den Listener mit aktivierter backlog-Option auszuführen:

- Verwenden Sie den Befehl `RUNMQLSR -b` oder
- Verwenden Sie den MQSC-Befehl **DEFINE LISTENER** mit dem Attribut `BACKLOG`, das auf den erforderlichen Wert gesetzt ist.

Weitere Informationen zum Befehl **RUNMQLSR** finden Sie im Abschnitt [runmqcls](#). Informationen zum Befehl **DEFINE LISTENER** finden Sie im Abschnitt [DEFINE LISTENER](#).

WebSphere MQ-Empfangsprogramm verwenden

Verwenden Sie den Befehl `RUNMQLSR` zur Ausführung des durch WebSphere MQ bereitgestellten Empfangsprogramms, das neue Kanäle als Threads startet. Beispiel:

```
RUNMQLSR -t spx [-m QMNAME] [-x 5E87]
```

Die eckigen Klammern geben optionale Parameter an. `QMNAME` ist für den Standardwarteschlangenmanager nicht erforderlich und die Socketnummer ist nicht erforderlich, wenn Sie den Standardwert (5E86) verwenden.

Um die beste Leistung zu erzielen, führen Sie den WebSphere MQ -Listener als vertrauenswürdige Anwendung aus, wie in [„Kanäle und Empfangsprogramme als vertrauenswürdige Anwendungen ausführen“](#) auf Seite 79 beschrieben. Weitere Informationen zu vertrauenswürdigen Anwendungen finden Sie unter [Einschränkungen für vertrauenswürdige Anwendungen](#).

Sie können alle WebSphere MQ -Listener, die auf einem inaktiven Warteschlangenmanager ausgeführt werden, mit folgendem Befehl stoppen:

```
ENDMQLSR [-m QMNAME]
```

Wenn Sie keinen Warteschlangenmanagernamen angeben, wird der Standardwarteschlangenmanager angenommen.

IPX-/SPX-Parameter

In den meisten Fällen werden die Standardeinstellungen für die IPX/SPX-Parameter Ihren Anforderungen entsprechen. Möglicherweise müssen Sie jedoch einige von ihnen in Ihrer Umgebung ändern, um ihre Verwendung für WebSphere MQ zu optimieren.

Die tatsächlichen Parameter und die Methode, sie zu ändern, variieren je nach Plattform und Anbieter der SPX-Kommunikationsunterstützung. Im Beispielabschnitt werden einige dieser Parameter beschrieben, insbesondere die Parameter, die den Betrieb von WebSphere MQ -Kanälen und -Clientverbindungen beeinflussen können.

Windows-Systeme

Ausführliche Informationen zur Verwendung und Einstellung der NWLink IPX- und SPX-Parameter finden Sie in der Microsoft-Dokumentation. Die IPX-/SPX-Parameter befinden sich unter folgenden Pfaden in der Registrierung:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Service\NWLinkSPX\Parameters  
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Service\NWLinkIPX\Parameters
```

Kommunikation auf UNIX and Linux -Systemen einrichten

DQM ist eine ferne Warteschlangenfunktion für IBM WebSphere MQ. Es stellt Kanalsteuerprogramme für den Warteschlangenmanager zur Verfügung, die die Schnittstelle zu Kommunikationsverbindungen bilden, die vom Systembediener gesteuert werden können. Die Kanaldefinitionen, die von der Verwaltung der verteilten Steuerung von Warteschlangen gehalten werden, verwenden diese Verbindungen.

Wenn ein Verwaltungskanal für die verteilte Steuerung von Warteschlangen gestartet wird, versucht er, die in der Kanaldefinition angegebene Verbindung zu verwenden. Um erfolgreich zu sein, ist es erforderlich, dass die Verbindung definiert und verfügbar ist. In diesem Abschnitt wird beschrieben, wie dies ausgeführt wird. Darüber hinaus enthalten die folgenden Abschnitte möglicherweise hilfreiche Informationen:

- [Beispielkonfiguration - IBM WebSphere MQ for AIX](#)
- [Beispielkonfiguration - IBM WebSphere MQ for HP-UX](#)
- [Beispielkonfiguration - IBM WebSphere MQ for Solaris](#)
- [Beispielkonfiguration- IBM WebSphere MQ für Linux](#)

Informationen zu Windows finden Sie unter „[Kommunikation für Windows einrichten](#)“ auf Seite 90.

Sie können zwischen zwei Kommunikationsformen für WebSphere MQ auf UNIX and Linux -Systemen wählen:

- [„TCP-Verbindung unter UNIX and Linux definieren“](#) auf Seite 100
- [„LU 6.2 -Verbindung unter UNIX and Linux definieren“](#) auf Seite 104

Jede Kanaldefinition muss nur ein Attribut des Übertragungsprotokolls (Transporttyp) angeben. Ein oder mehrere Protokolle können von einem WS-Manager verwendet werden.

Für IBM WebSphere MQ Explorer MQI-Clients kann es nützlich sein, alternative Kanäle mit unterschiedlichen Übertragungsprotokollen zu verwenden. Weitere Informationen zu IBM WebSphere MQ Explorer MQI-Clients finden Sie im Abschnitt [Übersicht über IBM WebSphere MQ MQI-Clients](#).

Zugehörige Konzepte

„[Anwendungen mithilfe der verteilten Steuerung von Warteschlangen verbinden](#)“ auf Seite 29

In diesem Abschnitt finden Sie ausführliche Informationen zur übergreifenden Kommunikation zwischen WebSphere MQ-Installationen, einschließlich Informationen zu Warteschlangendefinitionen, Kanaldefinitionen, Auslösungen und Synchronisationspunktprozeduren.

„[Kanäle in UNIX, Linux, and Windows überwachen und steuern](#)“ auf Seite 80

Für DQM müssen Sie die Kanäle zu fernen Warteschlangenmanagern erstellen, überwachen und steuern. Sie können Kanäle mit Befehlen, Programmen, IBM WebSphere MQ Explorer, Dateien für die Kanaldefinitionen und einem Speicherbereich für Synchronisationsinformationen steuern.

„[Verbindungen zwischen Client und Server konfigurieren](#)“ auf Seite 106

Um die Kommunikationsverbindungen zwischen WebSphere MQ MQI-Clients und -Servern zu konfigurieren, müssen Sie Ihr Kommunikationsprotokoll festlegen, die Verbindungen an beiden Enden der Verbindung definieren, ein Empfangsprogramm starten und Kanäle definieren.

TCP-Verbindung unter UNIX and Linux definieren

Die Kanaldefinition auf der sendenden Seite gibt die Adresse des Ziels an. Der Listener-oder inet-Dämon ist für die Verbindung an der empfangenden Seite konfiguriert.

Sendende Beendigung

Geben Sie den Hostnamen oder die TCP-Adresse der Zielmaschine in das Feld Verbindungsname der Kanaldefinition an. Der Port, zu dem die Verbindung hergestellt werden soll, standardmäßig 1414. Die Portnummer 1414 wird von der Internet Assigned Numbers Authority für WebSphere MQ zugeordnet.

Wenn Sie eine andere Portnummer als die Standardportnummer verwenden möchten, ändern Sie das Feld für den Verbindungsnamen wie folgt:

```
Connection Name REMHOST(1822)
```

Hierbei steht REMHOST für den Hostnamen der fernen Maschine und 1822 für die erforderliche Portnummer. (Dies muss der Port sein, an dem der Listener empfangsbereit ist.)

Alternativ können Sie die Portnummer ändern, indem Sie sie in der WS-Managerkonfigurationsdatei (qm.ini) angeben:

```
TCP:
Port=1822
```

Weitere Informationen zu den Werten, die Sie mit qm.ini festlegen, finden Sie unter [Zeilengruppen für Konfigurationsdateien für verteilte Steuerung von Warteschlangen](#).

Empfang auf TCP

Sie können entweder den TCP/IP-Listener verwenden, der der inet-Dämon (inetd) ist, oder den WebSphere MQ-Listener.

Einige Linux-Verteilungen verwenden jetzt anstelle des inet-Dämons den erweiterten inet-Dämon (xinetd). Informationen zur Verwendung des erweiterten inet-Dämons auf einem Linux-System finden Sie im Abschnitt [TCP-Verbindung auf Linux aufbauen](#).

Zugehörige Konzepte

[„TCP/IP-Listener verwenden“ auf Seite 101](#)

Um Kanäle in UNIX and Linux zu starten, müssen die Dateien '/etc/services' und 'inetd.conf' bearbeitet werden.

[„Verwendung der Option TCP-Listener-Backlog“ auf Seite 103](#)

In TCP werden die Verbindungen nur unvollständig behandelt, wenn zwischen dem Server und dem Client ein Dreiwege-Handshake nicht stattfindet. Diese Verbindungen werden als ausstehende Verbindungsanforderungen bezeichnet. Für diese ausstehenden Verbindungsanforderungen wird ein Maximalwert festgelegt und kann als Rückstand von Anforderungen betrachtet werden, die auf den TCP-Port warten, damit der Listener die Anforderung akzeptiert.

[„WebSphere MQ-Empfangsprogramm verwenden“ auf Seite 104](#)

Verwenden Sie den Befehl `runmqtsr`, um das mit WebSphere MQ bereitgestellte Empfangsprogramm auszuführen, das neue Kanäle als Threads startet.

[„Verwendung der Option "TCP/IP SO_KEEPALIVE"“ auf Seite 104](#)

Auf einigen UNIX and Linux-Systemen können Sie festlegen, wie lange TCP warten soll, bevor überprüft wird, ob die Verbindung noch verfügbar ist, und wie oft eine Verbindungswiederholung erfolgen soll, wenn die erste Überprüfung nicht erfolgreich ist. Dies ist entweder ein optimierbarer Kernelparameter oder kann in der Befehlszeile eingegeben werden.

TCP/IP-Listener verwenden

Um Kanäle in UNIX and Linux zu starten, müssen die Dateien '/etc/services' und 'inetd.conf' bearbeitet werden.

Befolgen Sie diese Anweisungen:

1. Bearbeiten Sie die Datei '/etc/services':

Anmerkung: Hierzu müssen Sie sich als Superuser oder Root-Benutzer anmelden. Sie können diese Änderung ändern, aber sie muss mit der Portnummer übereinstimmen, die auf der Senderseite angegeben wurde.

Fügen Sie die folgende Zeile zur Datei hinzu:

```
MQSeries 1414/tcp
```

Dabei ist 1414 die Portnummer, die für WebSphere MQ erforderlich ist. Die Port-Nummer darf 65535 nicht überschreiten.

2. Fügen Sie in der Datei `inetd.conf` eine Zeile hinzu, um das Programm `amqcrsta` aufzurufen. Dabei steht `MQ_INSTALLATION_PATH` für das übergeordnete Verzeichnis, in dem WebSphere MQ installiert ist:

```
MQSeries stream tcp nowait mqm MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta  
[-m Queue_Man_Name]
```

Die Aktualisierungen sind aktiv, nachdem 'inetd' die Konfigurationsdateien erneut gelesen hat. Geben Sie dazu die folgenden Befehle von der Rootbenutzer-ID aus:

- Unter AIX:

```
refresh -s inetd
```

- Unter HP-UX von der Benutzer-ID 'mqm':

```
inetd -c
```

- Unter Solaris 10 oder höher:

```
inetconv
```

- Auf anderen UNIX and Linux -Systemen (einschließlich Solaris 9):

```
kill -1 <process number>
```

Wenn das von `inetd` gestartete Empfangsprogramm die Ländereinstellung von `inetd` übernimmt, ist es möglich, dass der MQMDE nicht berücksichtigt wird (zusammengeführt) und als Nachrichtendaten in die Warteschlange gestellt wird. Um sicherzustellen, dass der MQMDE berücksichtigt wird, müssen Sie die Ländereinstellung korrekt festlegen. Die von `inetd` festgelegte Ländereinstellung stimmt möglicherweise nicht mit der für andere Ländereinstellungen überein, die von WebSphere MQ -Prozessen verwendet werden. So legen Sie die Ländereinstellung fest:

1. Erstellen Sie ein Shell-Script, das die Umgebungsvariablen `LANG`, `LC_COLLATE`, `LC_CTYPE`, `LC_MONETARY`, `LC_NUMERIC`, `LC_TIME` und `LC_MESSAGES` auf die Ländereinstellung setzt, die für andere WebSphere MQ -Prozesse verwendet wird.
2. Rufen Sie in demselben Shell-Script das Empfangsprogramm auf.
3. Ändern Sie die Datei '`inetd.conf`', um Ihr Shell-Script anstelle des Listenerprogramms aufzurufen.

Es ist möglich, mehr als einen Warteschlangenmanager auf dem Server zu verwenden. Sie müssen jeder der beiden Dateien eine Zeile für jeden der WS-Manager hinzufügen. Beispiel:

```
MQSeries1    1414/tcp  
MQSeries2    1822/tcp
```

```
MQSeries2 stream tcp nowait mqm MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta -m QM2
```

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

So lässt sich vermeiden, dass Fehlermeldungen ausgegeben werden, wenn eine Beschränkung für die Anzahl der ausstehenden Verbindungsanforderungen gilt, die in der Warteschlange eines einzelnen TCP-Ports stehen können. Informationen zur Anzahl der ausstehenden Verbindungsanforderungen finden Sie in „[Verwendung der Option TCP-Listener-Backlog](#)“ auf Seite 103.

Verwendung der Option TCP-Listener-Backlog

In TCP werden die Verbindungen nur unvollständig behandelt, wenn zwischen dem Server und dem Client ein Dreiwege-Handshake nicht stattfindet. Diese Verbindungen werden als ausstehende Verbindungsanforderungen bezeichnet. Für diese ausstehenden Verbindungsanforderungen wird ein Maximalwert festgelegt und kann als Rückstand von Anforderungen betrachtet werden, die auf den TCP-Port warten, damit der Listener die Anforderung akzeptiert.

Die Werte des Standard-Listener-Rückstandsprotokolls werden in [Tabelle 11](#) auf Seite 103 angezeigt.

Serverplattform	Maximale Verbindungsanforderungen
AIX	100
HP-UX	20
Linux	100
IBM i	255
Solaris	100
Windows -Server	100
Windows -Workstation	100
z/OS	255

Wenn der Rückstand die in [Tabelle 11](#) auf Seite 103 gezeigten Werte erreicht, wird die TCP/IP-Verbindung zurückgewiesen, und der Kanal kann nicht gestartet werden.

Bei MCA-Kanälen führt dies dazu, dass der Kanal in einen RETRY-Status eingeht und die Verbindung zu einem späteren Zeitpunkt erneut versucht.

Um diesen Fehler zu vermeiden, können Sie jedoch einen Eintrag in der Datei `qm.ini` hinzufügen:

```
TCP:
ListenerBacklog = n
```

Dies überschreibt die standardmäßige maximale Anzahl ausstehender Anforderungen (siehe [Tabelle 11](#) auf Seite 103) für den TCP/IP-Listener.

Anmerkung: Einige Betriebssysteme unterstützen einen größeren Wert als der Standardwert. Falls erforderlich, kann dieser Wert verwendet werden, um das Erreichen des Verbindungsgrenzwerts zu vermeiden.

Führen Sie einen der folgenden Schritte aus, um den Listener mit aktivierter `backlog`-Option auszuführen:

- Verwenden Sie den Befehl `runmqclsr -b` oder
- Verwenden Sie den MQSC-Befehl **DEFINE LISTENER** mit dem Attribut `BACKLOG`, das auf den erforderlichen Wert gesetzt ist.

Weitere Informationen zum Befehl `runmqclsr` finden Sie im Abschnitt [runmqclsr](#). Weitere Informationen zum Befehl `DEFINE LISTENER` finden Sie in [DEFINE LISTENER](#).

WebSphere MQ-Empfangsprogramm verwenden

Verwenden Sie den Befehl `runmqclsr`, um das mit WebSphere MQ bereitgestellte Empfangsprogramm auszuführen, das neue Kanäle als Threads startet.

Beispiel:

```
runmqclsr -t tcp [-m QMNAME] [-p 1822]
```

Die eckigen Klammern geben optionale Parameter an; QMNAME ist für den Standardwarteschlangenmanager nicht erforderlich, und die Portnummer ist nicht erforderlich, wenn Sie den Standardwert (1414) verwenden. Die Port-Nummer darf 65535 nicht überschreiten.

Um die beste Leistung zu erzielen, führen Sie den WebSphere MQ -Listener als vertrauenswürdige Anwendung aus, wie in „[Kanäle und Empfangsprogramme als vertrauenswürdige Anwendungen ausführen](#)“ auf Seite 79 beschrieben. Informationen zu vertrauenswürdigen Anwendungen finden Sie im Abschnitt [Einschränkungen für vertrauenswürdige Anwendungen](#).

Sie können alle WebSphere MQ -Listener, die auf einem inaktiven Warteschlangenmanager ausgeführt werden, mit folgendem Befehl stoppen:

```
endmqclsr [-m QMNAME]
```

Wenn Sie keinen Warteschlangenmanagernamen angeben, wird der Standardwarteschlangenmanager angenommen.

Verwendung der Option "TCP/IP SO_KEEPALIVE"

Auf einigen UNIX and Linux-Systemen können Sie festlegen, wie lange TCP warten soll, bevor überprüft wird, ob die Verbindung noch verfügbar ist, und wie oft eine Verbindungswiederholung erfolgen soll, wenn die erste Überprüfung nicht erfolgreich ist. Dies ist entweder ein optimierbarer Kernelparameter oder kann in der Befehlszeile eingegeben werden.

Wenn Sie die Option SO_KEEPALIVE verwenden möchten (weitere Informationen hierzu finden Sie im Abschnitt „[Überprüfen, ob das andere Ende des Kanals noch verfügbar ist](#)“ auf Seite 66), müssen Sie den folgenden Eintrag in Ihrer Warteschlangenmanagerkonfigurationsdatei (qm.ini) hinzufügen:

```
TCP:
  KeepAlive=yes
```

Weitere Informationen finden Sie in der Dokumentation für Ihr UNIX and Linux-System.

LU 6.2 -Verbindung unter UNIX and Linux definieren

SNA muss so konfiguriert werden, dass ein LU 6.2-Dialog zwischen den beiden Maschinen aufgebaut werden kann.

Die neuesten Informationen zur Konfiguration von SNA über TCP/IP finden Sie in der folgenden Online-Dokumentation zu IBM : [Communications Server](#).

SNA muss so konfiguriert werden, dass ein LU 6.2-Dialog zwischen den beiden Systemen aufgebaut werden kann.

Informationen hierzu finden Sie im Handbuch *Multiplatform APPC Configuration Guide* und in der folgenden Tabelle.

Tabelle 12. Einstellungen auf dem lokalen UNIX and Linux -System für eine ferne Warteschlangenmanagerplattform

Ferne Plattform	TPNAME	TPPATH
z/OS ohne CICS	Dasselbe gilt für den entsprechenden TPName-Wert in den Nebeninformationen zum fernen Warteschlangenmanager.	-
z/OS mit CICS	CKRC (Sender) CKSV (Requester) CKRC (Server)	-
IBM i	Entsprechendes gilt für den Vergleichswert im Routing-Eintrag auf dem IBM i-System.	-
UNIX and Linux-Systeme	Dasselbe gilt für den entsprechenden TPName-Wert in den Nebeninformationen zum fernen Warteschlangenmanager.	MQ_INSTALLATION_PATH/bin/amqcrs6a
Windows	Wie im Befehl Windows Run Listener angegeben, oder das aufrufbare Transaktionsprogramm, das mit TpSetup unter Windowsdefiniert wurde.	MQ_INSTALLATION_PATH\bin\amqcrs6a

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

Wenn mehrere WS-Manager auf derselben Maschine vorhanden sind, stellen Sie sicher, dass die TPNames in den Kanaldefinitionen eindeutig sind.

Zugehörige Konzepte

„Sendende Beendigung“ auf Seite 105

Erstellen Sie auf UNIX and Linux-Systemen ein CPI-C-Nebenobjekt (symbolisches Ziel), und geben Sie diesen Namen in das Feld Verbindungsname in der Kanaldefinition ein. Erstellen Sie außerdem einen LU 6.2-Link zu dem Partner.

„Empfangen auf LU 6.2“ auf Seite 106

Erstellen Sie auf UNIX and Linux-Systemen einen empfangsbereiten Anschluss auf der Empfangsseite, ein logisches LU 6.2-Verbindungsprofil und ein TPN-Profil.

Sendende Beendigung

Erstellen Sie auf UNIX and Linux-Systemen ein CPI-C-Nebenobjekt (symbolisches Ziel), und geben Sie diesen Namen in das Feld Verbindungsname in der Kanaldefinition ein. Erstellen Sie außerdem einen LU 6.2-Link zu dem Partner.

Geben Sie im CPI-C-Nebenobjekt den Namen der Partner-LU auf der empfangenden Maschine, den Namen des Transaktionsprogramms und den Modusnamen ein. Beispiel:

```
Partner LU Name          REMHOST
Remote TP Name          recv
Service Transaction Program no
Mode Name                #INTER
```

Verwenden Sie unter HP-UX die Umgebungsvariable APPCLLU, um die lokale LU zu benennen, die der Sender verwenden soll. Setzen Sie unter Solaris die Umgebungsvariable APPC_LOCAL_LU auf den Namen der lokalen LU.

SECURITY PROGRAM wird verwendet, sofern von CPI-C unterstützt, wenn WebSphere MQ versucht, eine SNA-Sitzung aufzubauen.

Empfangen auf LU 6.2

Erstellen Sie auf UNIX and Linux-Systemen einen empfangsbereiten Anschluss auf der Empfangsseite, ein logisches LU 6.2-Verbindungsprofil und ein TPN-Profil.

Geben Sie im TPN-Profil den vollständigen Pfad zu der ausführbaren Datei und den Namen des Transaktionsprogramms ein:

```
Full path to TPN executable   MQ_INSTALLATION_PATH/bin/amqcrs6a
Transaction Program name     recv
User ID                       0
```

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

Geben Sie auf Systemen, auf denen Sie die Benutzer-ID festlegen können, einen Benutzer an, der Mitglied der Gruppe "mqm" ist. Legen Sie unter AIX, Solaris und HP-UX die Umgebungsvariablen APPCTPN (Transaktionsname) und APPCLLU (Name der lokalen LU) fest (Sie können die Konfigurationsanzeigen für das aufgerufene Transaktionsprogramm verwenden).

Möglicherweise müssen Sie einen anderen WS-Manager als den Standardwarteschlangenmanager verwenden. Ist dies der Fall, definieren Sie eine Befehlsdatei, die Folgendes aufruft:

```
amqcrs6a -m Queue_Man_Name
```

Rufen Sie dann die Befehlsdatei auf.

Verbindungen zwischen Client und Server konfigurieren

Um die Kommunikationsverbindungen zwischen WebSphere MQ MQI-Clients und -Servern zu konfigurieren, müssen Sie Ihr Kommunikationsprotokoll festlegen, die Verbindungen an beiden Enden der Verbindung definieren, ein Empfangsprogramm starten und Kanäle definieren.

In WebSphere MQ werden die logischen Kommunikationsverbindungen zwischen Objekten als *Kanäle* bezeichnet. Die Kanäle, über die WebSphere MQ MQI-Clients mit Servern verbunden werden, werden als MQI-Kanäle bezeichnet. Sie definieren Kanaldefinitionen an jedem Ende Ihrer Verbindung, damit Ihre WebSphere MQ -Anwendung auf dem WebSphere MQ MQI-Client mit dem Warteschlangenmanager auf dem Server kommunizieren kann. Eine ausführliche Beschreibung der Vorgehensweise finden Sie im Abschnitt [Benutzerdefinierte Kanäle](#).

Vor der Definition der MQI-Kanäle sind zunächst folgende Schritte erforderlich:

1. Entscheiden Sie, welche Form der Kommunikation Sie verwenden möchten. Informationen hierzu finden Sie unter „[Zu verwendende Übertragungsart](#)“ auf Seite 106.
2. Definieren Sie die Verbindung an jedem Kanalende:

Um die Verbindung zu definieren, müssen Sie folgende Schritte ausführen:

- Konfigurieren Sie die Verbindung.
- Notieren Sie die Werte der Parameter, die Sie für die Kanaldefinitionen benötigen.
- Aktivieren Sie den Server, um eingehende Netzanforderungen von Ihrem WebSphere MQ MQI-Client zu erkennen, indem Sie einen *Listener* starten.

Zu verwendende Übertragungsart

Von den Plattformen werden jeweils unterschiedliche Übertragungsprotokolle unterstützt. Die Auswahl des Übertragungsprotokolls hängt von Ihrer Kombination aus WebSphere MQ MQI-Client und Serverplattformen ab.

Abhängig von Ihren Client- und Serverplattformen gibt es bis zu vier Arten von Übertragungsprotokollen für MQI-Kanäle:

- LU 6.2
- NetBIOS

- SPX
- TCP/IP

Wenn Sie Ihre MQI-Kanäle definieren, muss jede Kanaldefinition ein Übertragungsprotokoll (Transporttyp) angeben. Ein Server ist nicht auf ein Protokoll beschränkt, so dass verschiedene Kanaldefinitionen unterschiedliche Protokolle angeben können. Für WebSphere MQ MQI-Clients kann es nützlich sein, alternative MQI-Kanäle mit unterschiedlichen Übertragungsprotokollen zu verwenden.

Ihre Wahl des Übertragungsprotokolls kann durch Ihre spezielle Kombination von WebSphere MQ MQI-Client- und -Serverplattformen eingeschränkt werden. Die möglichen Kombinationen sind in der folgenden Tabelle aufgeführt.

<i>Tabelle 13. Übertragungsprotokolle-Kombination von WebSphere MQ MQI-Client- und Serverplattformen</i>		
Übertragungsprotokoll	WebSphere MQ-MQI-Client	WebSphere MQ-Server
TCP/IP	UNIX-Systeme Windows	UNIX-Systeme Windows z/OS
LU 6.2	UNIX-Systeme ¹ Windows	UNIX-Systeme ¹ Windows
NetBIOS	Windows	Windows
SPX	Windows	Windows
Anmerkung:		
1. Außer Linux für Power Systems		

Weitere Informationen zur Konfiguration verschiedener Verbindungsarten finden Sie unter den folgenden Links:

- [„TCP-Verbindung unter Windows definieren“](#) auf Seite 91
- [„TCP-Verbindung unter UNIX and Linux definieren“](#) auf Seite 100
- [„TCP/IP-Verbindungslimits“](#) auf Seite 109
- [„Definieren einer LU 6.2 -Verbindung unter Windows“](#) auf Seite 92
- [„LU 6.2 -Verbindung unter UNIX and Linux definieren“](#) auf Seite 104
- [„NetBIOS -Verbindung unter Windows definieren“](#) auf Seite 94
- [„SPX-Verbindung unter Windows definieren“](#) auf Seite 97

Zugehörige Konzepte

[„Einen erweiterten transaktionsorientierten Client konfigurieren“](#) auf Seite 110

In dieser Themensammlung wird beschrieben, wie die erweiterte transaktionsorientierte Funktion für jede Kategorie von Transaktionsmanager konfiguriert wird.

[„Definieren von MQI-Kanälen“](#) auf Seite 120

Um einen neuen Kanal zu erstellen, müssen Sie **zwei** Kanaldefinitionen erstellen, eine für jedes Ende der Verbindung, die denselben Channel-Namen und kompatible Kanaltypen verwenden. In diesem Fall sind die Kanaltypen *Serververbindung* und *Clientverbindung*.

[„Serververbindungs- und Clientverbindungsdefinitionen auf verschiedenen Plattformen erstellen“](#) auf Seite 122

Sie können jede Kanaldefinition auf dem Computer erstellen, auf den sie angewendet wird. Dabei bestehen Einschränkungen hinsichtlich der Erstellung von Kanaldefinitionen auf einem Clientcomputer.

„[Serververbindungs- und Clientverbindungsdefinitionen auf dem Server erstellen](#)“ auf Seite 125
 Sie können beide Definitionen auf dem Server erstellen und dann die Clientverbindungsdefinition für den Client verfügbar machen.

„[Kanalexitprogramme für MQI-Kanäle](#)“ auf Seite 130
 Für die WebSphere MQ MQI-Clientumgebung auf UNIX-, Linux -und Windows -Systemen sind drei Typen von Kanalexits verfügbar.

„[Verbindung eines Clients mit einer Gruppe mit gemeinsamer Warteschlange herstellen](#)“ auf Seite 135
 Sie können einen Client mit einer Gruppe mit gemeinsamer Warteschlange verbinden, indem Sie einen MQI-Kanal zwischen einem Client und einem WS-Manager auf einem Server erstellen, der Mitglied einer Gruppe mit gemeinsamer Warteschlange ist.

„[Client mit einer Konfigurationsdatei konfigurieren](#)“ auf Seite 136
 Konfigurieren Sie Ihre Clients mit Attributen in einer Textdatei. Diese Attribute können durch Umgebungsvariablen oder auf andere plattformspezifische Methoden überschrieben werden.

Zugehörige Tasks

[IBM MQ-MQI-Clientanwendungen mit Warteschlangenmanagern verbinden](#)

Zugehörige Verweise

[ANZEIGEN CHLAUTH](#)

[SET CHLAUTH](#)

Zu verwendende Übertragungsart

Unterschiedliche Plattformen unterstützen unterschiedliche Kommunikationsprotokolle. Die Auswahl des Übertragungsprotokolls hängt von Ihrer Kombination aus WebSphere MQ MQI-Client und Serverplattformen ab.

Es gibt vier Arten der Kommunikation für MQI-Kanäle auf verschiedenen Plattformen:

- LU 6.2
- NetBIOS
- SPX
- TCP/IP

Wenn Sie Ihre MQI-Kanäle definieren, muss jede Kanaldefinition ein Übertragungsprotokoll (Transporttyp) angeben. Ein Server ist nicht auf ein Protokoll beschränkt, so dass verschiedene Kanaldefinitionen unterschiedliche Protokolle angeben können. Für WebSphere MQ MQI-Clients kann es nützlich sein, alternative MQI-Kanäle mit unterschiedlichen Übertragungsprotokollen zu verwenden.

Ihre Wahl des Übertragungsprotokolls hängt auch von Ihrer Kombination von WebSphere MQ -Client- und Serverplattformen ab. Die möglichen Kombinationen sind in der folgenden Tabelle aufgeführt.

<i>Tabelle 14. Übertragungsprotokolle-Kombination von WebSphere MQ -Client- und Serverplattformen</i>		
Übertragungsprotokoll	WebSphere MQ-MQI-Client	WebSphere MQ-Server
TCP/IP	UNIX-Systeme Windows	UNIX-Systeme Windows
LU 6.2	UNIX-Systeme ¹ Windows	UNIX-Systeme ¹ Windows
NetBIOS	Windows	Windows
SPX	Windows	Windows

Tabelle 14. Übertragungsprotokolle-Kombination von WebSphere MQ -Client- und Serverplattformen (Forts.)

Übertragungsprotokoll	WebSphere MQ-MQI-Client	WebSphere MQ-Server
<p>Anmerkung:</p> <p>1. Außer Linux (POWER -Plattform)</p>		

Zugehörige Konzepte

„TCP-Verbindung unter Windows definieren“ auf Seite 91

Definieren Sie eine TCP-Verbindung, indem Sie einen Kanal auf der sendenden Seite konfigurieren, um die Adresse des Ziels anzugeben, und indem Sie ein Empfangsprogramm auf der Empfangsseite ausführen.

„TCP-Verbindung unter UNIX and Linux definieren“ auf Seite 100

Die Kanaldefinition auf der sendenden Seite gibt die Adresse des Ziels an. Der Listener- oder inet-Dämon ist für die Verbindung an der empfangenden Seite konfiguriert.

„Definieren einer LU 6.2 -Verbindung unter Windows“ auf Seite 92

SNA muss so konfiguriert werden, dass ein LU 6.2-Dialog zwischen den beiden Maschinen aufgebaut werden kann.

„LU 6.2 -Verbindung unter UNIX and Linux definieren“ auf Seite 104

SNA muss so konfiguriert werden, dass ein LU 6.2-Dialog zwischen den beiden Maschinen aufgebaut werden kann.

„NetBIOS -Verbindung unter Windows definieren“ auf Seite 94

WebSphere MQ verwendet drei Typen von NetBIOS -Ressourcen, wenn eine NetBIOS -Verbindung zu einem anderen WebSphere MQ -Produkt hergestellt wird: Sitzungen, Befehle und Namen. Jede dieser Ressourcen hat einen Grenzwert, der entweder standardmäßig oder nach Auswahl während der Installation von NetBIOS festgelegt wird.

„SPX-Verbindung unter Windows definieren“ auf Seite 97

Eine SPX-Verbindung findet nur bei einem Client und Server Anwendung, auf denen Windows XP und Windows 2003 Server ausgeführt werden.

Zugehörige Verweise

„TCP/IP-Verbindungslimits“ auf Seite 109

Die Anzahl der ausstehenden Verbindungsanforderungen, die an einem einzelnen TCP/IP-Port in die Warteschlange gestellt werden können, hängt von der Plattform ab. Ein Fehler tritt auf, wenn der Grenzwert erreicht ist.

TCP/IP-Verbindung definieren

Geben Sie den Transporttyp TCP in der Kanaldefinition auf der Clientseite an. Starten Sie ein Listener-Programm auf dem Server.

Sie können eine TCP/IP-Verbindung auf dem Client durch Angabe des Transporttyps TCP in der Kanaldefinition angeben.

Das Empfangen von Kanalprogrammen wird als Antwort auf eine Startanforderung vom sendenden Kanal gestartet. Hierzu muss zunächst ein Listenerprogramm gestartet werden, damit eingehende Netzanforderungen erkannt werden und der zugehörige Kanal gestartet wird. Das Verfahren zum Starten eines Listenerprogramms ist von der Serverplattform abhängig.

Beachten Sie die zugehörigen Abschnitte für Ihre Client- und Serverplattformen.

TCP/IP-Verbindungslimits

Die Anzahl der ausstehenden Verbindungsanforderungen, die an einem einzelnen TCP/IP-Port in die Warteschlange gestellt werden können, hängt von der Plattform ab. Ein Fehler tritt auf, wenn der Grenzwert erreicht ist.

Dieses Verbindungslimit ist nicht mit der maximalen Anzahl der Clients identisch, die Sie an einen IBM WebSphere MQ-Server anhängen können. Sie können mehr Clients mit einem Server verbinden, bis zu der Ebene, die von den Serversystemressourcen bestimmt wird. Die Rückprotokollwerte für Verbindungsanforderungen werden in der folgenden Tabelle angezeigt:

<i>Tabelle 15. Maximale Anzahl ausstehender Verbindungsanforderungen, die an einem TCP/IP-Port in die Warteschlange</i>	
Serverplattform	Maximale Verbindungsanforderungen
AIX	100
HP-UX	20
Linux	100
IBM	255
Solaris	100
Windows-Server	100
Windows -Workstation	100
z/OS	255

Wenn das Verbindungslimit erreicht ist, erhält der Client den Rückkehrcode MQRC_HOST_NOT_AVAILABLE vom MQCONN -Aufruf und einen Fehler AMQ9202 im Clientfehlerprotokoll (/var/mqm/errors/AMQERR0n.LOG auf UNIX and Linux -Systemen oder amqerr0n.log im Unterverzeichnis 'errors' der IBM WebSphere MQ -Clientinstallation unter Windows). Wenn der Client die Anforderung MQCONN wiederholt, kann dies erfolgreich sein.

Um die Anzahl der Verbindungsanforderungen zu erhöhen, die von dieser Einschränkung generiert werden, können Sie mehrere Empfangsprogramme, die jeweils an einem anderen Port empfangsbereit sind, oder mehrere Warteschlangenmanager haben, vermeiden.

NetBIOS -oder SPX-Verbindung definieren

NetBIOS -und SPX-Verbindungen gelten nur für Windows -Systeme.

Eine NetBIOS -Verbindung gilt nur für einen Client und einen Server unter Windows. Informationen hierzu finden Sie im Abschnitt [NetBIOS-Verbindung definieren](#).

Eine SPX-Verbindung gilt nur für einen Client und einen Server unter Windows XP oder Windows 2003 Server. Siehe [SPX-Verbindung definieren](#).

Einen erweiterten transaktionsorientierten Client konfigurieren

In dieser Themensammlung wird beschrieben, wie die erweiterte transaktionsorientierte Funktion für jede Kategorie von Transaktionsmanager konfiguriert wird.

Für jede Plattform bietet der erweiterte transaktionsorientierte Client Unterstützung für die folgenden externen Transaktionsmanager:

XA-kompatible Transaktionsmanager

Der erweiterte transaktionsorientierte Client stellt die Schnittstelle des XA-Ressourcenmanagers zur Unterstützung XA-konformer Transaktionsmanager wie CICS und Tuxedo bereit.

Microsoft Transaction Server (nurWindows -Systeme)

Nur auf Windows -Systemen unterstützt die Schnittstelle des XA-Ressourcenmanagers auch Microsoft Transaction Server (MTS). Die WebSphere MQ MTS-Unterstützung, die mit dem erweiterten transaktionsorientierten Client bereitgestellt wird, stellt die Brücke zwischen MTS und der Schnittstelle des XA-Ressourcenmanagers dar.

WebSphere Application Server

Frühere Versionen von WebSphere MQ unterstützten WebSphere Application Server Version 4 oder Version 5 und mussten bestimmte Konfigurationstasks ausführen, um den erweiterten transaktionsorientierten Client verwenden zu können. WebSphere Application Server Version 6 und höher enthält einen WebSphere MQ -Messaging-Provider, sodass Sie den erweiterten transaktionsorientierten Client nicht verwenden müssen.

Zugehörige Konzepte

„XA-kompatible Transaktionsmanager konfigurieren“ auf Seite 111

Konfigurieren Sie zuerst den Basisclient von WebSphere MQ und anschließend die erweiterte transaktionsorientierte Funktion anhand der Informationen in diesen Abschnitten.

„Microsoft Transaction Server“ auf Seite 120

Es ist keine zusätzliche Konfiguration erforderlich, bevor Sie MTS als Transaktionsmanager verwenden können. Es gibt jedoch einige Punkte zu beachten.

XA-kompatible Transaktionsmanager konfigurieren

Konfigurieren Sie zuerst den Basisclient von WebSphere MQ und anschließend die erweiterte transaktionsorientierte Funktion anhand der Informationen in diesen Abschnitten.

Anmerkung: In diesem Abschnitt wird davon ausgegangen, dass Sie über ein grundlegendes Verständnis für die XA-Schnittstelle verfügen, die von The Open Group in *Distributed Transaction Processing: The XA Specification* veröffentlicht wurde.

Wenn Sie einen erweiterten transaktionsorientierten Client konfigurieren wollen, müssen Sie zuerst den WebSphere MQ -Basisclient konfigurieren, wie unter [IBM WebSphere MQ -Client installieren](#) beschrieben. Mithilfe der Informationen in diesem Abschnitt können Sie anschließend die erweiterte transaktionsorientierte Funktion für einen XA-kompatiblen Transaktionsmanager wie CICS und Tuxedo konfigurieren.

Ein Transaktionsmanager kommuniziert mit einem Warteschlangenmanager als Ressourcenmanager, der denselben MQI-Kanal verwendet, wie er von der Clientanwendung verwendet wird, die mit dem Warteschlangenmanager verbunden ist. Wenn der Transaktionsmanager einen Funktionsaufruf des Ressourcenmanagers (xa_) ausgibt, wird der MQI-Kanal verwendet, um den Aufruf an den Warteschlangenmanager weiterzuleiten und die Ausgabe vom WS-Manager zurückzuerhalten.

Entweder kann der Transaktionsmanager den MQI-Kanal starten, indem er einen xa_open-Aufruf ausgibt, um den Warteschlangenmanager als Ressourcenmanager zu öffnen, oder die Clientanwendung kann den MQI-Kanal starten, indem er einen MQCONN-oder MQCONNX-Aufruf ausgibt.

- Wenn der Transaktionsmanager den MQI-Kanal startet und die Clientanwendung später MQCONN oder MQCONNX in demselben Thread aufruft, wird der MQCONN-oder MQCONNX-Aufruf erfolgreich ausgeführt und eine Verbindungskennung wird an die Anwendung zurückgegeben. Die Anwendung empfängt keinen Beendigungscode MQCC_WARNING mit einem Ursachencode MQRC_ALREADY_CONNECTED.
- Wenn die Clientanwendung den MQI-Kanal startet und der Transaktionsmanager später xa_open im selben Thread aufruft, wird der xa_open-Aufruf unter Verwendung des MQI-Kanals an den Warteschlangenmanager weitergeleitet.

In einer Wiederherstellungssituation nach einem Fehler, wenn keine Clientanwendungen ausgeführt werden, kann der Transaktionsmanager einen dedizierten MQI-Kanal verwenden, um alle unvollständigen Arbeitseinheiten wiederherzustellen, an denen der Warteschlangenmanager zum Zeitpunkt des Fehlers beteiligt war.

Beachten Sie die folgenden Bedingungen, wenn Sie einen erweiterten transaktionsorientierten Client mit einem XA-konformen Transaktionsmanager verwenden:

- In einem einzelnen Thread kann eine Clientanwendung immer nur mit einem Warteschlangenmanager verbunden sein. Diese Einschränkung gilt nur bei Verwendung eines erweiterten transaktionsorientierten Clients. Eine Clientanwendung, die einen WebSphere MQ -Basisclient verwendet, kann in einem einzigen Thread gleichzeitig mit mehreren Warteschlangenmanagern verbunden sein.
- Jeder Thread einer Clientanwendung kann eine Verbindung zu einem anderen WS-Manager herstellen.
- Eine Clientanwendung kann keine gemeinsam genutzten Verbindungskennungen verwenden.

Um die erweiterte transaktionsorientierte Funktion zu konfigurieren, müssen Sie dem Transaktionsmanager die folgenden Informationen für jeden Warteschlangenmanager bereitstellen, der als Ressourcenmanager fungiert:

- Eine xa_open-Zeichenfolge
- Ein Zeiger auf eine XA-Switchstruktur

Wenn der Transaktionsmanager xa_open aufruft, um den Warteschlangenmanager als Ressourcenmanager zu öffnen, übergibt er die xa_open-Zeichenfolge an den erweiterten transaktionsorientierten Client als Argument xa_info für den Aufruf. Der erweiterte transaktionsorientierte Client verwendet die Informationen in der Zeichenfolge xa_open auf die folgenden Arten:

- Gehen Sie wie folgt vor, um einen MQI-Kanal zum Server-WS-Manager zu starten, wenn die Clientanwendung noch nicht gestartet wurde.
- Überprüfen Sie, ob der Warteschlangenmanager, der der Transaktionsmanager als Ressourcenmanager geöffnet wird, mit dem Warteschlangenmanager identisch ist, zu dem die Clientanwendung eine Verbindung herstellt.
- Um die Funktionen ax_reg und ax_unreg des Transaktionsmanagers zu lokalisieren, wenn der Warteschlangenmanager die dynamische Registrierung verwendet

Für das Format einer xa_open-Zeichenfolge und für weitere Informationen darüber, wie die Informationen in der xa_open-Zeichenfolge von einem erweiterten transaktionsorientierten Client verwendet werden, finden Sie weitere Informationen in [„Das Format einer xa_open-Zeichenfolge.“](#) auf Seite 113.

Eine XA-Switchstruktur ermöglicht es dem Transaktionsmanager, die vom erweiterten transaktionsorientierten Client bereitgestellten xa-Funktionen zu lokalisieren und gibt an, ob der Warteschlangenmanager die dynamische Registrierung verwendet. Informationen zu den XA-Switch-Strukturen, die mit einem erweiterten transaktionsorientierten Client bereitgestellt werden, finden Sie in [„Die XA-Switchstrukturen“](#) auf Seite 116.

Informationen zum Konfigurieren der erweiterten transaktionsorientierten Funktion für einen bestimmten Transaktionsmanager und weitere Informationen zur Verwendung des Transaktionsmanagers mit einem erweiterten transaktionsorientierten Client finden Sie in den folgenden Abschnitten:

- [„Erweiterten transaktionsorientierten Client für CICS konfigurieren“](#) auf Seite 118
- [„Konfigurieren eines erweiterten transaktionsorientierten Clients für Tuxedo“](#) auf Seite 119

Zugehörige Konzepte

[„Die Parameter CHANNEL, TRPTYPE, CONNAME und QMNAME der Zeichenfolge 'xa_open'“](#) auf Seite 114
Verwenden Sie diese Informationen, um zu verstehen, wie der erweiterte transaktionsorientierte Client diese Parameter verwendet, um den Warteschlangenmanager zu ermitteln, zu dem eine Verbindung hergestellt werden soll.

[„Zusätzliche Fehlerverarbeitung für xa_open“](#) auf Seite 116

Der Aufruf 'xa_open' schlägt unter bestimmten Umständen fehl.

Zugehörige Tasks

[„Extended Transactional Client mit SSL-Kanälen verwenden“](#) auf Seite 117

Sie können einen SSL-Kanal nicht mit der xa_open-Zeichenfolge einrichten. Führen Sie die folgenden Anweisungen aus, um die Definitionstabelle für den Clientkanal (ccdt) zu verwenden.

Zugehörige Verweise

[„Die Parameter TPM und AXLIB“](#) auf Seite 115

Ein erweiterter transaktionsorientierter Client verwendet die Parameter "TPM" und "AXLIB", um die Funktionen "ax_reg" und "ax_unreg" des Transaktionsmanagers zu lokalisieren. Diese Funktionen werden nur verwendet, wenn der WS-Manager eine dynamische Registrierung verwendet.

[„Wiederherstellung nach einem Fehler in der erweiterten transaktionsorientierten Verarbeitung“](#) auf Seite 116

Nach einem Fehler muss ein Transaktionsmanager in der Lage sein, alle unvollständigen Arbeitseinheiten wiederherzustellen. Dazu muss der Transaktionsmanager in der Lage sein, als Ressourcenmanager einen

beliebigen Warteschlangenmanager zu öffnen, der zum Zeitpunkt des Ausfalls an einer unvollständigen UO- Unit beteiligt war.

Das Format einer xa_open-Zeichenfolge.

Eine xa_open-Zeichenfolge enthält Paare definierter Parameternamen und -werte.

Eine xa_open-Zeichenfolge hat das folgende Format:

```
parm_name1=parm_value1,parm_name2=parm_value2, ...
```

Dabei ist *parm_name* der Name eines Parameters und *parm_value* der Wert eines Parameters. Die Namen der Parameter sind nicht die Groß-/Kleinschreibung, es sei denn, die Werte der Parameter werden von der Groß-/Kleinschreibung abhängig gemacht. Sie können die Parameter in beliebiger Reihenfolge angeben.

Die Namen, die Bedeutungen und die gültigen Werte der Parameter lauten wie folgt:

Name

Bedeutung und gültige Werte

CHANNEL

Der Name eines MQI-Kanals.

Dies ist ein optionaler Parameter. Wenn dieser Parameter angegeben wird, muss auch der Parameter CONNAME angegeben werden.

TRPTYPE

Das Übertragungsprotokoll für den MQI-Kanal. Die folgenden Werte sind gültig:

LU62

SNA LU 6.2

NETBIOS

NetBIOS

SPX

IPX/SPX

TCP

TCP/IP

Dies ist ein optionaler Parameter. Wird dieser Parameter nicht angegeben, wird der Standardwert TCP angenommen. Bei den Werten des Parameters muss die Groß-/Kleinschreibung nicht beachtet werden.

CONNAME

Die Netzadresse des WS-Managers auf dem Serverende des MQI-Kanals. Die gültigen Werte für diesen Parameter hängen vom Wert des Parameters TRPTYPE ab:

LU62

Ein symbolischer Bestimmungsname, der einen CPI-C-Nebeninformationen-Eintrag identifiziert.

Der netzqualifizierte Name einer Partner-LU ist kein gültiger Wert und kein Aliasname der Partner-LU. Dies liegt daran, dass es keine zusätzlichen Parameter gibt, um einen Transaktionsprogrammnamen (TP) und einen Modusnamen anzugeben.

NETBIOS

Ein NetBIOS-Name.

SPX

Eine 4-Byte-Netzadresse, eine 6 -Byteknotenadresse und eine optionale 2-Byte-Socketnummer. Diese Werte müssen in Hexadezimalschreibweise angegeben werden. Ein Punkt muss die Netz- und Knotenadressen voneinander trennen, und die Socket-Nummer, falls angegeben, muss in runde Klammern eingeschlossen werden. Beispiel:

```
0a0b0c0d.804abcde23a1(5e86)
```

Wenn die Socketnummer weggelassen wird, wird der Standardwert 5e86 angenommen.

TCP

Ein Hostname oder eine IP-Adresse, optional gefolgt von einer Port-Nummer in runden Klammern. Wenn die Portnummer nicht angegeben wird, wird der Standardwert 1414 angenommen.

Dies ist ein optionaler Parameter. Wenn dieser Parameter angegeben wird, muss auch der Parameter CHANNEL angegeben werden.

QMNAME

Der Name des Warteschlangenmanagers am Serverende des MQI-Kanals. Der Name darf weder leer noch ein einzelner Stern (*) sein, noch kann der Name mit einem Stern beginnen. Dies bedeutet, dass der Parameter einen bestimmten WS-Manager nach Namen identifizieren muss.

Dies ist ein obligatorischer Parameter.

Wenn eine Clientanwendung mit einem bestimmten WS-Manager verbunden ist, muss jede Transaktionswiederherstellung von demselben Warteschlangenmanager verarbeitet werden.

Wenn die Anwendung eine Verbindung zu einem z/OS -Warteschlangenmanager herstellt, kann die Anwendung entweder den Namen eines bestimmten Warteschlangenmanagers oder den Namen einer Gruppe mit gemeinsamer Warteschlange angeben. Durch Verwendung des Warteschlangenmanager-namens oder QSG-Namens steuert die Anwendung, ob sie in einer Transaktion mit einer Disposition für die QMGR-Einheit oder mit der Disposition GROUP der Wiederherstellungs-Disposition ausgeführt wird. Die Disposition GROUP der Wiederherstellung ermöglicht die Wiederherstellung der Transaktion, die in einem beliebigen Mitglied der Gruppe QSG verarbeitet werden soll. Um GROUP-Arbeitseinheiten mit Wiederherstellung verwenden zu können, muss das WS-Manager-Attribut von **GROUPUR** aktiviert sein.

TPM

Der Transaktionsmanager, der verwendet wird. Gültige Werte sind CICS und TUXEDO.

Ein erweiterter transaktionsorientierter Client verwendet diesen Parameter und den Parameter AXLIB für den gleichen Zweck. Weitere Informationen zu diesen Parametern finden Sie in den [Parametern für TPM und AXLIB](#).

Dies ist ein optionaler Parameter. Bei den Werten des Parameters muss die Groß-/Kleinschreibung nicht beachtet werden.

AXLIB

Der Name der Bibliothek, die die Funktionen "ax_reg" und "ax_unreg" des Transaktionsmanagers enthält.

Dies ist ein optionaler Parameter.

Hier ist ein Beispiel für eine xa_open-Zeichenfolge:

```
channel=MARS.SVR,trptype=tcp,conname=MARS(1415),qmname=MARS,tpm=cics
```

Die Parameter CHANNEL, TRPTYPE, CONNAME und QMNAME der Zeichenfolge 'xa_open'

Verwenden Sie diese Informationen, um zu verstehen, wie der erweiterte transaktionsorientierte Client diese Parameter verwendet, um den Warteschlangenmanager zu ermitteln, zu dem eine Verbindung hergestellt werden soll.

Wenn die Parameter CHANNEL und CONNAME in der Zeichenfolge xa_open angegeben werden, verwendet der erweiterte transaktionsorientierte Client diese Parameter und den Parameter TRPTYPE, um einen MQI-Kanal an den Warteschlangenmanager des Servers zu starten.

Wenn die Parameter CHANNEL und CONNAME nicht in der Zeichenfolge xa_open angegeben werden, verwendet der erweiterte transaktionsorientierte Client den Wert der Umgebungsvariablen MQSERVER, um einen MQI-Kanal zu starten. Wenn die Umgebungsvariable MQSERVER nicht definiert ist, verwendet der erweiterte transaktionsorientierte Client den Eintrag in der Clientkanaldefinition, die durch den Parameter QMNAME angegeben wird.

In jedem dieser Fälle überprüft der erweiterte transaktionsorientierte Client, ob der Wert des Parameters QMNAME der Name des Warteschlangenmanagers am Serverende des MQI-Kanals ist. Ist dies nicht der Fall, schlägt der Aufruf 'xa_open' fehl, und der Transaktionsmanager meldet den Fehler an der Anwendung.

Wenn der Anwendungsclient eine Verbindung zu einem z/OS -Warteschlangenmanager mit V7.0.1 oder höher herstellt, kann für den Parameter QMNAME der Name einer Gruppe mit gemeinsamer Warteschlange angegeben werden. Dies ermöglicht dem Anwendungsclient die Teilnahme an einer Transaktion mit einer Disposition der Gruppe 'GROUP' der Disposition.

Wenn die Anwendung einen QSG-Namen im Parameterfeld QMNAME verwendet und die Eigenschaft GROUPUR auf dem WS-Manager, mit dem sie verbunden ist, inaktiviert ist, schlägt der Aufruf von xa_open fehl.

Wenn die Anwendung eine Verbindung zu einem WS-Manager mit einer früheren Version als Version 7.0.1 herstellt, ist der xa_open-Aufruf erfolgreich, aber die Transaktion hat eine Disposition mit QMGR-Einheit. Stellen Sie sicher, dass Anwendungen, die die Disposition GROUP der Arbeitseinheit mit Wiederherstellung erfordern, nur mit Warteschlangenmanagern in V7.0.1 oder höher verbunden sind.

Wenn die Clientanwendung MQCONN oder MQCONNX in demselben Thread aufruft, den der Transaktionsmanager zum Absetzen des Xa_open-Aufrufs verwendet hat, empfängt die Anwendung eine Verbindungskennung für den MQI-Kanal, der durch den Aufruf 'xa_open' gestartet wurde. Ein zweiter MQI-Kanal wurde nicht gestartet. Der erweiterte transaktionsorientierte Client überprüft, ob der Wert des Parameters QMgrName im MQCONN-oder MQCONNX-Aufruf der Name des Warteschlangenmanagers am Serverende des MQI-Kanals ist. Ist dies nicht der Fall, schlägt der MQCONN-oder MQCONNX-Aufruf mit einem Ursachencode von MQRC_ANOTHER_Q_MGR_CONNECTED fehl. Wenn der Wert des Parameters QMgrName leer oder ein einzelner Stern (*) ist oder mit einem Stern beginnt, schlägt der MQCONN-oder MQCONNX-Aufruf mit dem Ursachencode MQRC_Q_MGR_NAME_ERROR fehl.

Wenn die Clientanwendung bereits einen MQI-Kanal gestartet hat, indem sie MQCONN oder MQCONNX aufruft, bevor der Transaktionsmanager xa_open im selben Thread aufruft, verwendet der Transaktionsmanager stattdessen diesen MQI-Kanal. Ein zweiter MQI-Kanal wurde nicht gestartet. Der erweiterte transaktionsorientierte Client überprüft, ob der Wert des Parameters QMNAME in der Zeichenfolge 'xa_open' der Name des Server-WS-Managers ist. Ist dies nicht der Fall, schlägt der Aufruf 'xa_open' fehl.

Wenn eine Clientanwendung zuerst einen MQI-Kanal startet, kann der Wert des Parameters QMgrName im MQCONN-oder MQCONNX-Aufruf leer oder ein einzelner Stern (*) sein, oder er kann mit einem Stern beginnen. Unter diesen Umständen müssen Sie jedoch sicherstellen, dass der Warteschlangenmanager, zu dem die Anwendung eine Verbindung herstellt, mit dem Warteschlangenmanager identisch ist, den der Transaktionsmanager als Ressourcenmanager öffnen will, wenn er zu einem späteren Zeitpunkt xa_open im selben Thread aufruft. Es treten möglicherweise weniger Probleme auf, wenn der Wert des Parameters QMgrName den Warteschlangenmanager explizit anhand des Namens identifiziert.

Die Parameter TPM und AXLIB

Ein erweiterter transaktionsorientierter Client verwendet die Parameter "TPM" und "AXLIB", um die Funktionen "ax_reg" und "ax_unreg" des Transaktionsmanagers zu lokalisieren. Diese Funktionen werden nur verwendet, wenn der WS-Manager eine dynamische Registrierung verwendet.

Wenn der TPM-Parameter in einer xa_open-Zeichenfolge angegeben wird, der Parameter AXLIB jedoch nicht angegeben wird, nimmt der erweiterte transaktionsorientierte Client einen Wert für den Parameter AXLIB an, der auf dem Wert des Parameters TPM basiert. Informationen zu den angenommenen Werten für den Parameter AXLIB finden Sie im Abschnitt [Tabelle 16](#) auf Seite 115.

<i>Tabelle 16. Angenommen, Werte des Parameters AXLIB</i>		
Wert von TPM	Plattform	Assumierter Wert von AXLIB
CICS	AIX	/usr/lpp/encina/lib/libEncServer.a (EncServer_shr.o)
CICS	HP-UX	/opt/encina/lib/libEncServer.sl
CICS	Solaris	/opt/encina/lib/libEncServer.so

<i>Tabelle 16. Angenommen, Werte des Parameters AXLIB (Forts.)</i>		
Wert von TPM	Plattform	Assumierter Wert von AXLIB
CICS	Windows-Systeme	libEncServer
Tuxedo	AIX	/usr/lpp/tuxedo/lib/libtux.a(libtux.so.60)
Tuxedo	HP-UX	/opt/tuxedo/lib/libtux.sl
Tuxedo	Solaris	/opt/tuxedo/lib/libtux.so.60
Tuxedo	Windows-Systeme	libtux

Wenn der Parameter AXLIB in einer xa_open-Zeichenfolge angegeben wird, verwendet der erweiterte transaktionsorientierte Client seinen Wert, um jeden angenommenen Wert basierend auf dem Wert des TPM-Parameters zu überschreiben. Der Parameter AXLIB kann auch für einen Transaktionsmanager verwendet werden, für den der TPM-Parameter keinen angegebenen Wert hat.

Zusätzliche Fehlerverarbeitung für xa_open

Der Aufruf 'xa_open' schlägt unter bestimmten Umständen fehl.

In den Themen in diesem Abschnitt werden Situationen beschrieben, in denen der Aufruf 'xa_open' fehlschlägt. Es schlägt auch fehl, wenn eine der folgenden Situationen eintritt:

- Es sind Fehler in der xa_open-Zeichenfolge vorhanden.
- Es sind nicht genügend Informationen zum Starten eines MQI-Kanals vorhanden.
- Beim Versuch, einen MQI-Kanal zu starten, ist ein Fehler aufgetreten (z. B. der Server-WS-Manager nicht aktiv).

Wiederherstellung nach einem Fehler in der erweiterten transaktionsorientierten Verarbeitung

Nach einem Fehler muss ein Transaktionsmanager in der Lage sein, alle unvollständigen Arbeitseinheiten wiederherzustellen. Dazu muss der Transaktionsmanager in der Lage sein, als Ressourcenmanager einen beliebigen Warteschlangenmanager zu öffnen, der zum Zeitpunkt des Ausfalls an einer unvollständigen UO- Unit beteiligt war.

Wenn Sie jemals Konfigurationsdaten ändern müssen, müssen Sie daher sicherstellen, dass alle unvollständigen Arbeitseinheiten aufgelöst wurden, bevor Sie die Änderungen vornehmen. Alternativ müssen Sie sicherstellen, dass die Konfigurationsänderungen die Fähigkeit des Transaktionsmanagers nicht beeinträchtigen, die Warteschlangenmanager zu öffnen, die er öffnen muss. Im Folgenden finden Sie Beispiele für solche Konfigurationsänderungen:

- Inhalt einer xa_open-Zeichenfolge ändern
- Den Wert der Umgebungsvariablen MQSERVER ändern
- Einträge in der Definitionstabelle für den Clientkanal ändern (CCDT)
- Kanaldefinition für Serververbindung löschen

Die XA-Switchstrukturen

Es werden zwei XA-Switchstrukturen mit dem erweiterten transaktionsorientierten Client auf jeder Plattform bereitgestellt.

Diese Switchstrukturen sind:

MQRMIXASwitch

Diese Switchstruktur wird von einem Transaktionsmanager verwendet, wenn ein Warteschlangenmanager, der als Ressourcenmanager fungiert, keine dynamische Registrierung verwendet.

MQRMIXASwitchDynamic

Diese Switchstruktur wird von einem Transaktionsmanager verwendet, wenn ein Warteschlangenmanager, der als Ressourcenmanager fungiert, die dynamische Registrierung verwendet.

Diese Switchstrukturen befinden sich in den in [Tabelle 17](#) auf Seite [117](#) dargestellten Bibliotheken.

<i>Tabelle 17. WebSphere MQ -Bibliotheken mit XA-Switchstrukturen</i>	
Plattform	Bibliothek mit den XA-Switchstrukturen
AIX HP-UX Linux Solaris	<code>MQ_INSTALLATION_PATH/lib/libmqxa</code>
Windows-Systeme	<code>MQ_INSTALLATION_PATH\bin\mqxa.dll</code> ¹
<i>MQ_INSTALLATION_PATH</i> steht für das übergeordnete Verzeichnis, in dem WebSphere MQ installiert ist.	

Der Name des WebSphere MQ -Ressourcenmanagers in jeder Switchstruktur lautet MQSeries_XA_RMI, aber viele Warteschlangenmanager können dieselbe Switchstruktur gemeinsam nutzen.

Zugehörige Konzepte

„Dynamische Registrierung und erweiterte transaktionsorientierte Verarbeitung“ auf Seite [117](#)

Die Verwendung der dynamischen Registrierung ist eine Form der Optimierung, da sie die Anzahl der vom Transaktionsmanager ausgegebenen xa-Funktionsaufrufe verringern kann.

Dynamische Registrierung und erweiterte transaktionsorientierte Verarbeitung

Die Verwendung der dynamischen Registrierung ist eine Form der Optimierung, da sie die Anzahl der vom Transaktionsmanager ausgegebenen xa-Funktionsaufrufe verringern kann.

Wenn ein Warteschlangenmanager keine dynamische Registrierung verwendet, bezieht ein Transaktionsmanager den Warteschlangenmanager in jede UOWUOW.Einheit ein. Der Transaktionsmanager führt dies durch Aufrufen von `xa_start`, `xa_end` und `xa_prepare` aus, selbst wenn der Warteschlangenmanager keine Ressourcen hat, die innerhalb der UOI aktualisiert werden.

Wenn ein Warteschlangenmanager die dynamische Registrierung verwendet, beginnt ein Transaktionsmanager mit der Annahme, dass der Warteschlangenmanager nicht an einer Arbeitseinheit beteiligt ist, und ruft nicht `xa_start` auf. Der WS-Manager wird dann nur dann in die Arbeitseinheit einbezogen, wenn seine Ressourcen innerhalb der Synchronisationspunktsteuerung aktualisiert werden. Wenn dies der Fall ist, ruft der erweiterte transaktionsorientierte Client `ax_reg` auf, um die Beteiligung des WS-Managers zu registrieren.

Extended Transactional Client mit SSL-Kanälen verwenden

Sie können einen SSL-Kanal nicht mit der `xa_open`-Zeichenfolge einrichten. Führen Sie die folgenden Anweisungen aus, um die Definitionstabelle für den Clientkanal (`ccdt`) zu verwenden.

Informationen zu diesem Vorgang

Aufgrund der begrenzten Größe der `xa_open`-Zeichenfolge 'xa_info' ist es nicht möglich, alle Informationen, die für die Einrichtung eines SSL-Kanals erforderlich sind, mithilfe der Zeichenfolgemethode "xa_open" zum Herstellen einer Verbindung zu einem Warteschlangenmanager zu übergeben. Daher müssen Sie entweder die Definitionstabelle für den Clientkanal verwenden oder, falls Ihr Transaktionsmanager zulässt, den Kanal mit MQCONNX erstellen, bevor Sie den Aufruf 'xa_open' absetzen.

Führen Sie die folgenden Schritte aus, um die Definitionstabelle für den Clientkanal zu verwenden:

Vorgehensweise

1. Geben Sie eine `xa_open`-Zeichenfolge an, die nur den obligatorischen Parameter `qmname` (WS-Manager-Name) enthält, z. B. `XA_Open_String=qmname=MYQM`.
2. Verwenden Sie einen Warteschlangenmanager, um einen CLNTCONN-Kanal (Clientverbindungskanal) mit den erforderlichen SSL-Parametern zu definieren. Geben Sie den Namen des Warteschlangenma-

nagers in das Attribut QMNAME in der CLNTCONN-Definition ein. Dieser Wert wird mit dem Namen des Befehls qmname in der Zeichenfolge xa_open abgeglichen.

3. Machen Sie die CLNTCONN-Definition für das Clientsystem in einer Clientkanaldefinitionstabelle (CCDT) oder unter Windows in Active Directory verfügbar.
4. Wenn Sie eine CCDT verwenden, identifizieren Sie die CCDT, die die Definition des Kanals CLNTCONN enthält, indem Sie die Umgebungsvariablen MQCHLLIB und MQCHLTAB verwenden. Legen Sie diese Variablen in den Umgebungen fest, die sowohl von der Clientanwendung als auch vom Transaktionsmanager verwendet werden.

Ergebnisse

Dadurch erhält der Transaktionsmanager eine Kanaldefinition für den entsprechenden Warteschlangenmanager mit den SSL-Attributen, die zur korrekten Authentifizierung erforderlich sind, einschließlich SSLCIPH, der CipherSpec.

Erweiterten transaktionsorientierten Client für CICS konfigurieren

Sie konfigurieren einen erweiterten transaktionsorientierten Client zur Verwendung durch CICS, indem Sie einer CICS-Region eine XAD-Ressourcendefinition hinzufügen.

Fügen Sie die XAD-Ressourcendefinition mit dem CICS-Befehl RDO (Resource Definition Online) **cicsadd** hinzu. Die XAD-Ressourcendefinition gibt die folgenden Informationen an:

- Eine xa_open-Zeichenfolge
- Der vollständig qualifizierte -Pfadname einer Switchloaddatei.

Eine Switchloaddatei wird für die Verwendung durch CICS auf jeder der folgenden Plattformen bereitgestellt: AIX, HP-UX, Solaris und Windows. Jede Switchloaddatei enthält eine Funktion, die einen Zeiger auf die XA-Switchstruktur zurückgibt, die für die dynamische Registrierung verwendet wird, MQRMIXASwitch-Dynamic. Den vollständig qualifizierten Pfadnamen der einzelnen Switchloaddateien finden Sie unter [Tabelle 18 auf Seite 118](#).

Tabelle 18. Die Switchloaddateien	
Plattform	Switchloaddatei
AIX HP-UX Linux Solaris	MQ_INSTALLATION_PATH/lib/amqczsc
Windows-Systeme	MQ_INSTALLATION_PATH\bin\mqcc4swi.dll ¹
MQ_INSTALLATION_PATH steht für das übergeordnete Verzeichnis, in dem WebSphere MQ installiert ist.	

Das folgende Beispiel zeigt eine XAD-Ressourcendefinition für Windows-Systeme:

```
cicsadd -c xad -r REGION1 WMQXA \
  ResourceDescription="WebSphere MQ queue manager MARS" \
  XAOpen="channel=MARS.SVR, trptype=tcp, connname=MARS(1415), qmname=MARS, tpm=cics" \
  SwitchLoadFile="C:\Program Files\IBM\WebSphere MQ\bin\mqcc4swi.dll"
```

Weitere Informationen zum Hinzufügen einer XAD-Ressourcendefinition zu einer CICS-Region finden Sie im Handbuch *CICS Administration Reference* und im *CICS Administration Guide* für Ihre Plattform.

Beachten Sie die folgenden Informationen zur Verwendung von CICS mit einem erweiterten transaktionsorientierten Client:

- Sie können einer CICS-Region nur eine XAD-Ressourcendefinition für WebSphere MQ hinzufügen. Dies bedeutet, dass nur ein Warteschlangenmanager einer Region zugeordnet werden kann und alle

CICS -Anwendungen, die in der Region ausgeführt werden, nur mit diesem Warteschlangenmanager verbunden werden können. Wenn Sie CICS -Anwendungen ausführen wollen, die eine Verbindung zu einem anderen Warteschlangenmanager herstellen, müssen Sie die Anwendungen in einer anderen Region ausführen.

- Jeder Anwendungsserver in einer Region ruft 'xa_open' auf, während er initialisiert wird, und startet einen MQI-Kanal zu dem Warteschlangenmanager, der der Region zugeordnet ist. Dies bedeutet, dass der Warteschlangenmanager gestartet werden muss, bevor ein Anwendungsserver gestartet wird, da andernfalls der Aufruf 'xa_open' fehlschlägt. Alle WebSphere MQ MQI-Clientanwendungen, die später vom Anwendungsserver verarbeitet werden, verwenden denselben MQI-Kanal.
- Wenn ein MQI-Kanal gestartet wird und kein Sicherheitsexit am Clientende des Kanals vorhanden ist, lautet die Benutzer-ID, die vom Clientsystem zur Serververbindung fließt, cics. Unter bestimmten Umständen verwendet der Warteschlangenmanager diese Benutzer-ID für Berechtigungsprüfungen, wenn der MCA der Serververbindung anschließend versucht, im Namen einer Clientanwendung auf die Warteschlangenmanagerressourcen zuzugreifen. Wenn diese Benutzer-ID für Berechtigungsprüfungen verwendet wird, müssen Sie sicherstellen, dass sie über die Berechtigung zum Zugriff auf alle Ressourcen verfügt, auf die sie zugreifen müssen.

Informationen darüber, wann der Warteschlangenmanager diese Benutzer-ID für Berechtigungsprüfungen verwendet, finden Sie im Abschnitt [Security](#).

- Die CICS -Taskbeendigungsexits, die für die Verwendung auf WebSphere MQ -Clientsystemen bereitgestellt werden, sind in [Tabelle 19](#) auf Seite 119 aufgelistet. Sie konfigurieren diese Exits auf dieselbe Weise wie die entsprechenden Exits für WebSphere MQ -Serversysteme. Diese Informationen finden Sie daher unter [CICS -Benutzerexits aktivieren](#).

<i>Tabelle 19. CICS-Taskbeendigungsexits</i>		
Plattform	Quelle	Bibliothek
AIX HP-UX Linux Solaris	amqzscgx.c	amqczscg
Windows-Systeme	amqzscgn.c	mqqc1415.dll

Konfigurieren eines erweiterten transaktionsorientierten Clients für Tuxedo

Aktualisieren Sie die UBBCONFIG-Datei und die Ressourcenmanagertabelle, um die XAD-Ressourcendefinition für die Verwendung durch Tuxedo zu konfigurieren.

Führen Sie die folgenden Aktionen aus, um die XAD-Ressourcendefinition für die Verwendung durch Tuxedo zu konfigurieren:

- Verwenden Sie im Abschnitt GROUPS der Tuxedo-UBBCONFIG-Datei für eine Anwendung den Parameter OPENINFO, um eine xa_open-Zeichenfolge anzugeben.

Ein Beispiel hierfür finden Sie in der Beispieldatei UBBCONFIG, die für die Verwendung mit den Tuxedo-Beispielprogrammen bereitgestellt wird. Unter AIX, HP-UX und Solaris lautet der Name der Datei ubbstxcx.cfg und auf Windows -Systemen lautet der Name der Datei ubbstxcn . cfg.

- Im Eintrag für einen WS-Manager in der Tuxedo-Ressourcenmanagertabelle:
 - udataobj/RM (AIX, HP-UX und Solaris)
 - udataobj\rm (Windows -Systeme)

Geben Sie den Namen einer XA-Switchstruktur und den vollständig qualifizierten Pfadnamen der Bibliothek an, die die Struktur enthält. Ein Beispiel für die Vorgehensweise bei den einzelnen Plattformen finden Sie unter [TUXEDO-Beispiele](#) . Tuxedo unterstützt die dynamische Registrierung eines Ressourcenmanagers, so dass Sie entweder MQRMIXASwitch oder MQRMIXASwitchDynamic verwenden können.

Microsoft Transaction Server

Es ist keine zusätzliche Konfiguration erforderlich, bevor Sie MTS als Transaktionsmanager verwenden können. Es gibt jedoch einige Punkte zu beachten.

Beachten Sie die folgenden Informationen zur Verwendung von MTS mit dem erweiterten transaktionsorientierten Client:

- Eine MTS-Anwendung startet immer einen MQI-Kanal, wenn er eine Verbindung zu einem Server-WS-Manager herstellt. MTS verwendet in seiner Rolle als Transaktionsmanager dann denselben MQI-Kanal, um mit dem Warteschlangenmanager zu kommunizieren.
- Nach einem Fehler muss MTS in der Lage sein, alle unvollständigen Arbeitseinheiten wiederherzustellen. Dazu muss MTS in der Lage sein, mit jedem WS-Manager zu kommunizieren, der zum Zeitpunkt des Ausfalls an einer unvollständigen Arbeitseinheit beteiligt war.

Wenn eine MTS-Anwendung eine Verbindung zu einem Server-WS-Manager herstellt und einen MQI-Kanal startet, extrahiert der erweiterte transaktionsorientierte Client genügend Informationen aus den Parametern des MQCONN- oder MQCONNX-Aufrufs, damit der Kanal nach einem Fehler erneut gestartet werden kann, falls dies erforderlich ist. Der erweiterte transaktionsorientierte Client übergibt die Informationen an MTS, und MTS zeichnet die Informationen in seinem Protokoll auf.

Wenn die MTS-Anwendung einen MQCONN-Aufruf ausgibt, ist diese Information lediglich der Name des Warteschlangenmanagers. Wenn die MTS-Anwendung einen MQCONNX-Aufruf absetzt und eine Kanaldefinitionsstruktur (MQCD) bereitstellt, enthält die Information auch den Namen des MQI-Kanals, die Netzadresse des Server-WS-Managers und das Kommunikationsprotokoll für den Kanal.

In einer Wiederherstellungssituation übergibt MTS diese Informationen an den erweiterten transaktionsorientierten Client zurück, und der erweiterte transaktionsorientierte Client verwendet ihn zum erneuten Starten des MQI-Kanals.

Wenn Sie jemals die Konfigurationsdaten ändern müssen, stellen Sie daher sicher, dass alle unvollständigen Arbeitseinheiten aufgelöst wurden, bevor Sie die Änderungen vornehmen. Stellen Sie alternativ sicher, dass die Konfigurationsänderungen die Fähigkeit des erweiterten transaktionsorientierten Clients nicht beeinträchtigen, einen MQI-Kanal unter Verwendung der von MTS erfassten Informationen erneut zu starten. Im Folgenden finden Sie Beispiele für solche Konfigurationsänderungen:

- Den Wert der Umgebungsvariablen MQSERVER ändern
- Einträge in der Definitionstabelle für den Clientkanal ändern (CCDT)
- Kanaldefinition für Serververbindung löschen
- Beachten Sie die folgenden Bedingungen, wenn Sie einen erweiterten transaktionsorientierten Client mit MTS verwenden:
 - In einem einzelnen Thread kann eine Clientanwendung immer nur mit einem Warteschlangenmanager verbunden sein.
 - Jeder Thread einer Clientanwendung kann eine Verbindung zu einem anderen WS-Manager herstellen.
 - Eine Clientanwendung kann keine gemeinsam genutzten Verbindungskennungen verwenden.

Definieren von MQI-Kanälen

Um einen neuen Kanal zu erstellen, müssen Sie **zwei** Kanaldefinitionen erstellen, eine für jedes Ende der Verbindung, die denselben Channel-Namen und kompatible Kanaltypen verwenden. In diesem Fall sind die Kanaltypen *Serververbindung* und *Clientverbindung*.

Benutzerdefinierte Kanäle

Wenn der Server Kanäle nicht automatisch definiert, gibt es zwei Möglichkeiten, die Kanaldefinitionen zu erstellen und der WebSphere MQ -Anwendung auf der WebSphere MQ MQI-Clientmaschine den Zugriff auf den Kanal zu erteilen:

Diese beiden Methoden werden im Detail beschrieben:

1. Erstellen Sie eine Kanaldefinition auf dem WebSphere MQ -Client und die andere auf dem Server.

Dies gilt für jede Kombination von WebSphere MQ MQI-Client- und -Serverplattformen. Verwenden Sie es, wenn Sie auf dem System gestartet werden, oder um Ihre Konfiguration zu testen.

Ausführliche Informationen zu dieser Vorgehensweise finden Sie im Abschnitt [„Serververbindungs- und Clientverbindungsdefinitionen auf verschiedenen Plattformen erstellen“](#) auf Seite 122.

2. Erstellen Sie beide Kanaldefinitionen auf der Servermaschine.

Verwenden Sie diese Methode, wenn Sie mehrere Kanäle und WebSphere MQ MQI-Clientsysteme gleichzeitig einrichten.

Ausführliche Informationen zu dieser Vorgehensweise finden Sie im Abschnitt [„Serververbindungs- und Clientverbindungsdefinitionen auf dem Server erstellen“](#) auf Seite 125.

Automatisch definierte Kanäle

WebSphere MQ -Produkte auf anderen Plattformen als z/OS enthalten eine Funktion, die automatisch eine Kanaldefinition auf dem Server erstellen kann, wenn keine vorhanden ist.

Wenn eine eingehende Verbindungsanforderung von einem Client empfangen wird und auf diesem Warteschlangenmanager keine entsprechende Serververbindungsdefinition gefunden werden kann, erstellt WebSphere MQ automatisch eine Definition und fügt sie dem Warteschlangenmanager hinzu. Die automatische Definition basiert auf der Definition des Standard-Serververbindungskanals SYSTEM.AUTO.SVRCONN. Sie aktivieren die automatische Definition von Serververbindungsdefinitionen, indem Sie das WS-Manager-Objekt mit dem Befehl ALTER QMGR mit dem Parameter CHAD (oder mit dem PCF-Befehl Change Queue Manager mit dem Parameter ChannelAutoDef ändern) aktualisieren.

Weitere Informationen zur automatischen Erstellung von Kanaldefinitionen finden Sie unter [Automatische Definition von Empfänger- und Serververbindungskanälen](#).

Zugehörige Konzepte

[„Automatisch definierte Kanäle“](#) auf Seite 121

WebSphere MQ -Produkte auf anderen Plattformen als z/OS enthalten eine Funktion, die automatisch eine Kanaldefinition auf dem Server erstellen kann, wenn keine vorhanden ist.

[„Benutzerdefinierte Kanäle“](#) auf Seite 121

Wenn der Server Kanäle nicht automatisch definiert, gibt es zwei Möglichkeiten, die Kanaldefinitionen zu erstellen und der WebSphere MQ -Anwendung auf der WebSphere MQ MQI-Clientmaschine den Zugriff auf den Kanal zu erteilen:

[„Kanalsteuerfunktion“](#) auf Seite 58

Die Kanalsteuerfunktion stellt Funktionen zur Verfügung, mit der Sie Kanäle definieren, überwachen und steuern können.

Automatisch definierte Kanäle

WebSphere MQ -Produkte auf anderen Plattformen als z/OS enthalten eine Funktion, die automatisch eine Kanaldefinition auf dem Server erstellen kann, wenn keine vorhanden ist.

Wenn eine eingehende Verbindungsanforderung von einem Client empfangen wird und auf diesem Warteschlangenmanager keine entsprechende Serververbindungsdefinition gefunden werden kann, erstellt WebSphere MQ automatisch eine Definition und fügt sie dem Warteschlangenmanager hinzu. Die automatische Definition basiert auf der Definition des Standard-Serververbindungskanals SYSTEM.AUTO.SVRCONN. Sie aktivieren die automatische Definition von Serververbindungsdefinitionen, indem Sie das WS-Manager-Objekt mit dem Befehl ALTER QMGR mit dem Parameter CHAD (oder mit dem PCF-Befehl Change Queue Manager mit dem Parameter ChannelAutoDef ändern) aktualisieren.

Benutzerdefinierte Kanäle

Wenn der Server Kanäle nicht automatisch definiert, gibt es zwei Möglichkeiten, die Kanaldefinitionen zu erstellen und der WebSphere MQ -Anwendung auf der WebSphere MQ MQI-Clientmaschine den Zugriff auf den Kanal zu erteilen:

Diese beiden Methoden werden im Detail beschrieben:

1. Erstellen Sie eine Kanaldefinition auf dem WebSphere MQ -Client und die andere auf dem Server.

Dies gilt für jede Kombination von WebSphere MQ MQI-Client- und -Serverplattformen. Verwenden Sie es, wenn Sie auf dem System gestartet werden, oder um Ihre Konfiguration zu testen.

Ausführliche Informationen zu dieser Vorgehensweise finden Sie im Abschnitt „Serververbindungs- und Clientverbindungsdefinitionen auf verschiedenen Plattformen erstellen“ auf Seite 122.

2. Erstellen Sie beide Kanaldefinitionen auf der Servermaschine.

Verwenden Sie diese Methode, wenn Sie mehrere Kanäle und WebSphere MQ MQI-Clientensysteme gleichzeitig einrichten.

Ausführliche Informationen zu dieser Vorgehensweise finden Sie im Abschnitt „Serververbindungs- und Clientverbindungsdefinitionen auf dem Server erstellen“ auf Seite 125.

Serververbindungs- und Clientverbindungsdefinitionen auf verschiedenen Plattformen erstellen

Sie können jede Kanaldefinition auf dem Computer erstellen, auf den sie angewendet wird. Dabei bestehen Einschränkungen hinsichtlich der Erstellung von Kanaldefinitionen auf einem Clientcomputer.

Auf allen Plattformen können Sie WebSphere MQ Script-Befehle (MQSC), PCF-Befehle (Programmable Command Format) oder den IBM WebSphere MQ Explorer verwenden, um einen Serververbindungskanal auf der Servermaschine zu definieren.

Da MQSC-Befehle auf einer Maschine, auf der WebSphere MQ nur als WebSphere MQ MQI-Client installiert wurde, nicht verfügbar sind, müssen Sie verschiedene Methoden zum Definieren eines Clientverbindungskanals auf der Clientmaschine verwenden.

Zugehörige Konzepte

„Clientverbindungskanal auf dem IBM WebSphere MQ MQI-Client erstellen“ auf Seite 123

Sie können einen Clientverbindungskanal auf der Client-Workstation mit MQSERVER oder mit der MQCNO-Struktur in einem MQCONNX-Aufruf definieren.

Zugehörige Tasks

„Definieren eines Serververbindungskanals auf dem Server“ auf Seite 122

Starten Sie bei Bedarf MQSC, und definieren Sie dann den Serververbindungskanal.

Definieren eines Serververbindungskanals auf dem Server

Starten Sie bei Bedarf MQSC, und definieren Sie dann den Serververbindungskanal.

Vorgehensweise

1. Optional: Wenn Ihre Serverplattform nicht z/OS ist, erstellen und starten Sie zuerst einen Warteschlangenmanager und starten Sie dann MQSC-Befehle.

- a) Erstellen Sie einen WS-Manager mit dem Namen QM1. Beispiel:

```
crtmqm QM1
```

- b) Starten Sie den Warteschlangenmanager:

```
strmqm QM1
```

- c) Starten Sie MQSC-Befehle:

```
runmqsc QM1
```

2. Definieren Sie einen Kanal mit dem ausgewählten Namen und dem Kanaltyp *server-connection*.

```
DEFINE CHANNEL(CHAN1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
DESCR('Server-connection to Client_1')
```

Diese Kanaldefinition ist dem Warteschlangenmanager zugeordnet, der auf dem Server ausgeführt wird.

3. Verwenden Sie den folgenden Befehl, um den Zugriff der Eingangsverbindung auf den Warteschlangenmanager zu ermöglichen:

```
SET CHLAUTH(CHAN1) TYPE(ADDRESSMAP) ADDRESS('IP address') MCAUSER('userid')
```

- Dabei verwendet SET CHLAUTH den Namen des Kanals, der im vorherigen Schritt definiert wurde.
- Hierbei steht 'IP address' für die IP-Adresse des Clients.
- Dabei ist 'userid' die ID, die Sie dem Kanal für die Zugriffssteuerung für die Zielwarteschlangen bereitstellen möchten. Bei diesem Feld muss die Groß-/Kleinschreibung beachtet werden.

Sie können die eingehende Verbindung mithilfe einer Reihe unterschiedlicher Attribute identifizieren. Das Beispiel verwendet die IP-Adresse. Zu den alternativen Attributen gehören die Clientbenutzer-ID und der SSL- oder TLS-Betreffname. Weitere Informationen finden Sie in [Kanalauthentifizierungsdatensätze](#).

Clientverbindungskanal auf dem IBM WebSphere MQ MQI-Client erstellen

Sie können einen Clientverbindungskanal auf der Client-Workstation mit MQSERVER oder mit der MQCNO-Struktur in einem MQCONN-Aufruf definieren.

Verwenden von MQSERVER

Sie können die Umgebungsvariable MQSERVER verwenden, um eine einfache Definition eines Clientverbindungskanals anzugeben. Es ist einfach in dem Sinne, dass Sie mit dieser Methode nur einige Attribute des Kanals angeben können.

- Geben Sie wie folgt eine einfache Kanaldefinition in Windows an:

```
SET MQSERVER=ChannelName/TransportType/ConnectionName
```

- Geben Sie wie folgt eine einfache Kanaldefinition auf UNIX and Linux -Systemen an:

```
export MQSERVER=ChannelName/TransportType/ConnectionName
```

Dabei gilt:

- ChannelName muss mit dem Namen identisch sein, der auf dem Server definiert ist. Er darf keinen Schrägstrich enthalten.
- TransportType kann je nach IBM WebSphere MQ MQI-Clientplattform einen der folgenden Werte haben:
 - LU62
 - TCP
 - NETBIOS
 - SPX

Anmerkung: Auf UNIX and Linux-Systemen muss die Groß-/Kleinschreibung beachtet werden und der Wert für 'TransportType' muss in Großbuchstaben geschrieben werden. Ein MQCONN -oder MQCONNX -Aufruf gibt 2058 zurück, wenn der TransportType nicht erkannt wird.

- ConnectionName ist der Name des Servers, wie er für das Übertragungsprotokoll definiert ist (TransportType).

Beispiel unter Windows:

```
SET MQSERVER=CHANNEL1/TCP/MCID66499
```

oder auf UNIX and Linux -Systemen:

```
export MQSERVER=CHANNEL1/TCP/'MCID66499'
```

Anmerkung: Informationen zum Ändern der TCP/IP-Anschlussnummer finden Sie in „MQSERVER“ auf Seite 160.

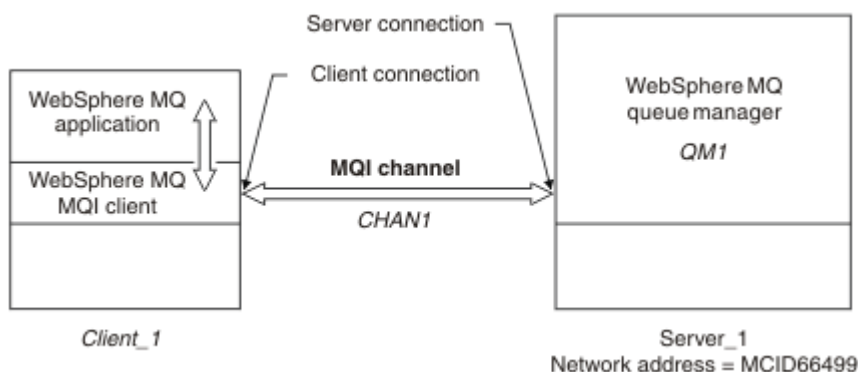


Abbildung 17. Einfache Kanaldefinition

Einige weitere Beispiele für einfache Kanaldefinitionen:

- Unter Windows:

```
SET MQSERVER=CHANNEL1/TCP/9.20.4.56  
SET MQSERVER=CHANNEL1/NETBIOS/BOX643
```

- Auf UNIX and Linux-Systemen:

```
export MQSERVER=CHANNEL1/TCP/'9.20.4.56'  
export MQSERVER=CHANNEL1/LU62/BOX99
```

Hierbei steht BOX99 für den LU 6.2-Verbindungsnamen.

Auf dem IBM WebSphere MQ MQI-Client versuchen alle **MQCONN** -oder **MQCONNX** -Anforderungen, den definierten Kanal zu verwenden, es sei denn, der Kanal wird in einer MQCD-Struktur überschrieben, auf die von der MQCNO-Struktur verwiesen wird, die an **MQCONNX** übergeben wird.

Anmerkung: Weitere Informationen zur Umgebungsvariablen *MQSERVER* finden Sie in „MQSERVER“ auf Seite 160.

MQCNO-Struktur in einem MQCONNX-Aufruf verwenden

Eine IBM WebSphere MQ MQI-Clientanwendung kann die Verbindungsoptionsstruktur MQCNO in einem **MQCONNX** -Aufruf verwenden, um auf eine Kanaldefinitionsstruktur (MQCD) zu verweisen, die die Definition eines Clientverbindungskanals enthält.

Auf diese Weise kann die Clientanwendung die **ChannelName**-, **TransportType**- und **ConnectionName** -Attribute eines Kanals während der Ausführung angeben, so dass die Clientanwendung gleichzeitig eine Verbindung zu mehreren Server-WS-Managern herstellen kann.

Beachten Sie Folgendes: Wenn Sie einen Kanal mit der Umgebungsvariablen *MQSERVER* definieren, können die Attribute **ChannelName**, **TransportType** und **ConnectionName** während der Ausführung nicht angegeben werden.

Eine Clientanwendung kann auch Attribute eines Kanals angeben, z. B. **MaxMsgLength** und **SecurityExit**. Wenn Sie solche Attribute angeben, kann die Clientanwendung Werte für die Attribute angeben, die nicht die Standardwerte sind, und ermöglicht es, Kanalexitprogramme am Clientende eines MQI-Kanals zu nennen.

Wenn ein Kanal Secure Sockets Layer (SSL) oder Transport Layer Security (TLS) verwendet, kann eine Clientanwendung in der MQCD-Struktur auch Informationen bereitstellen, die sich auf SSL oder TLS beziehen. Zusätzliche Informationen zu SSL oder TLS können in der SSL- oder TLS-Konfigurationsoptionsstruktur MQSCO bereitgestellt werden, die auch von der MQCNO-Struktur in einem **MQCONN**-Aufruf referenziert wird.

Weitere Informationen zu den MQCNO-, MQCD- und MQSCO-Strukturen finden Sie in [MQCNO](#), [MQCD](#) und [MQSCO](#).

Anmerkung: Das Beispielprogramm für MQCONN heißt **amqscnxc**. Ein weiteres Beispielprogramm namens **amqssslc** veranschaulicht die Verwendung der MQSCO-Struktur.

Serververbindungs- und Clientverbindungsdefinitionen auf dem Server erstellen

Sie können beide Definitionen auf dem Server erstellen und dann die Clientverbindungsdefinition für den Client verfügbar machen.

Definieren Sie zunächst einen Serververbindungskanal und anschließend einen Clientverbindungskanal. Auf allen Plattformen können Sie WebSphere MQ Script-Befehle (MQSC), PCF-Befehle (Programmable Command Format) oder den IBM WebSphere MQ Explorer verwenden, um einen Serververbindungskanal auf der Servermaschine zu definieren.

Auf dem Server erstellte Clientverbindungskanaldefinitionen werden Clients unter Verwendung einer Definitionstabelle für Clientkanäle (CCDT, Client Channel Definition Table) zur Verfügung gestellt.

Zugehörige Konzepte

„[Definitionstabelle für den Clientkanal](#)“ auf Seite 125

Die Definitionstabelle für den Clientkanal (CCDT) bestimmt die Kanaldefinitionen und Authentifizierungsinformationen, die von Clientanwendungen verwendet werden, um eine Verbindung zum Warteschlangenmanager herzustellen. Auf anderen Plattformen als z/OS wird automatisch eine CCDT erstellt. Anschließend müssen Sie sie für die Clientanwendung zur Verfügung stellen.

Zugehörige Tasks

„[Serververbindungskanal auf dem Server definieren](#)“ auf Seite 127

Erstellen Sie eine Serververbindungskanaldefinition für den Warteschlangenmanager.

„[Definieren des Clientverbindungskanals auf dem Server](#)“ auf Seite 128

Nachdem Sie den Serververbindungskanal definiert haben, definieren Sie nun den entsprechenden Clientverbindungskanal.

„[Zugreifen auf Clientverbindungskanaldefinitionen](#)“ auf Seite 129

Machen Sie die Definitionstabelle für Clientkanal (CCDT) Clientanwendungen zugänglich, indem Sie sie kopieren oder für die gemeinsame Nutzung freigeben, und geben Sie anschließend ihr Verzeichnis und ihren Namen auf dem Clientcomputer an.

Definitionstabelle für den Clientkanal

Die Definitionstabelle für den Clientkanal (CCDT) bestimmt die Kanaldefinitionen und Authentifizierungsinformationen, die von Clientanwendungen verwendet werden, um eine Verbindung zum Warteschlangenmanager herzustellen. Auf anderen Plattformen als z/OS wird automatisch eine CCDT erstellt. Anschließend müssen Sie sie für die Clientanwendung zur Verfügung stellen.

Der Zweck der Clientkanaldefinitionstabelle (CCDT) besteht darin, die Kanaldefinitionen zu ermitteln, die von Clientanwendungen für die Verbindung zum Warteschlangenmanager verwendet werden. Die Kanaldefinition gibt auch die Authentifizierungsinformationen an, die für die Verbindungen gelten.

Die CCDT ist eine Binärdatei. Sie wird von einem WS-Manager generiert. Der WS-Manager liest die CCDT-Datei nicht.

Auf anderen Plattformen als z/OS wird die CCDT beim Erstellen des Warteschlangenmanagers erstellt. Clientverbindungskanäle werden der Tabelle hinzugefügt, wenn Sie den Befehl **DEFINE CHANNEL** verwenden, und ihre Definitionen werden geändert, wenn Sie den Befehl **ALTER CHANNEL** ausgeben.

Sie können die CCDT verwenden, um Clients die Authentifizierungsinformationen zur Verfügung zu stellen, die auf den Widerruf von SSL-Zertifikaten hin überprüft werden sollen. Definieren Sie eine Namensliste, die Authentifizierungsinformationsobjekte enthält, und setzen Sie das WS-Managerattribut **SSLCRLNameList** auf den Namen der Namensliste.

Eine Clientanwendung kann eine CCDT auf verschiedene Arten verwenden. Die CCDT kann auf den Client-Computer kopiert werden. Sie können die CCDT an eine Position kopieren, die von mehreren Clients gemeinsam genutzt wird. Sie können die CCDT als gemeinsam genutzte Datei für den Client zugänglich machen, während sie sich weiterhin auf dem Server befindet.

Wenn Sie die Datei mit FTP kopieren, verwenden Sie die Option `bin`, um den Binärmodus festzulegen; verwenden Sie nicht den Standardmodus ASCII. Whichever Methode, die Sie auswählen, um die CCDT verfügbar zu machen, die Position muss sicher sein, um unberechtigte Änderungen an den Kanälen zu verhindern.

Andere Plattformen als z/OS

Beim Erstellen eines Warteschlangenmanagers wird eine Standard-CCDT mit der Bezeichnung AMQCLCHL.TAB erstellt.

AMQCLCHL.TAB befindet sich standardmäßig im folgenden Verzeichnis auf einem Server:

-   Auf UNIX and Linux-Systemen:

```
/prefix/qmgrs/QUEUENAME/@ipcc
```

Bei dem Namen des Verzeichnisses, auf das `QUEUENAME` verweist, muss die Groß-/Kleinschreibung auf UNIX and Linux -Systemen beachtet werden. Der Verzeichnisname ist möglicherweise nicht mit dem Namen des Warteschlangenmanagers identisch, wenn der Name des Warteschlangenmanagers Sonderzeichen enthält.

-  Unter Windows:

```
MQ_INSTALLATION_PATH\data\qmgrs\QUEUENAME\@ipcc
```

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

Es kann jedoch sein, dass Sie ein anderes Verzeichnis für WS-Manager-Daten verwenden möchten. Sie können den Parameter `-md DataPath` angeben, wenn Sie den Befehl `crtmqm` verwendet haben. Wenn Sie dies tun, befindet sich AMQCLCHL.TAB im `@ipcc`-Verzeichnis des von Ihnen angegebenen *Datenpfads*.

Der Pfad zu der CCDT kann durch Einstellung von `MQCHLLIB` geändert werden. Wenn Sie `MQCHLLIB` festlegen, müssen Sie, wenn Sie mehrere WS-Manager auf demselben Server haben, dieselbe CCDT-Position gemeinsam nutzen.

Die CCDT wird beim Erstellen des Warteschlangenmanagers erstellt. Jeder Eintrag einer CCDT stellt eine Clientverbindung zu einem bestimmten Warteschlangenmanager dar. Ein neuer Eintrag wird hinzugefügt, wenn Sie einen Clientverbindungskanal mit dem Befehl **DEFINE CHANNEL** definieren, und der Eintrag wird aktualisiert, sobald Sie die Clientverbindungskanäle mit dem Befehl **ALTER CHANNEL** ändern.

Informationen zum Angeben der Position der CCDT auf dem Client

Auf einem Clientsystem können Sie die Position der CCDT auf zwei Arten angeben:

- Verwenden Sie die Umgebungsvariablen MQCHLLIB , um das Verzeichnis anzugeben, in dem sich die Tabelle befindet, und MQCHLTAB , um den Dateinamen der Tabelle anzugeben.
- Die Clientkonfigurationsdatei wird verwendet. Verwenden Sie in der Zeilengruppe CHANNELS die Attribute ChannelDefinitionDirectory , um das Verzeichnis anzugeben, in dem sich die Tabelle befindet, und ChannelDefinitionFile , um den Namen der Datei anzugeben.

Wenn die Position sowohl in der Clientkonfigurationsdatei als auch unter Verwendung von Umgebungsvariablen angegeben wird, müssen die Umgebungsvariablen Priorität haben. Sie können diese Funktion verwenden, um eine Standardposition in der Clientkonfigurationsdatei anzugeben und diese bei Bedarf mit Umgebungsvariablen zu überschreiben.

Zugehörige Verweise

„MQCHLLIB“ auf Seite 158

MQCHLLIB gibt den Verzeichnispfad für die Datei an, die die Definitionstabelle für den Clientkanal (CCDT) enthält. Die Datei wird auf dem Server erstellt, kann aber auf die MQI-Client-Workstation WebSphere MQ kopiert werden.

Zugehörige Informationen

[Mit widerrufenden Zertifikaten arbeiten](#)

Migration und Definitionstabellen für Clientkanäle (CCDT)

Das interne Format der Definitionstabelle für Clientkanal kann sich im Allgemeinen mit neuen Release-Levels von IBM WebSphere MQ ändern. Daher kann ein IBM WebSphere MQ MQI-Client eine Definitionstabelle für Clientkanäle nur verwenden, wenn sie von einem Serverwarteschlangenmanager vorbereitet wurde, der denselben Release-Level wie der Client oder einen früheren Release-Level hat.

Ein MQI-Client der Version 7.1 IBM WebSphere MQ kann eine Clientkanaldefinitionstabelle verwenden, die von einem Warteschlangenmanager der Version 6.0 vorbereitet wurde. Ein Client der Version 6.0 kann jedoch keine Definitionstabelle für Clientkanäle verwenden, die von einem Warteschlangenmanager der Version 7.1 vorbereitet wurde.

Clientverbindungskanäle im Active Directory

Auf Windows -Systemen, die Active Directory unterstützen, veröffentlicht IBM WebSphere MQ Clientverbindungskanäle in Active Directory , um eine dynamische Client/Server-Bindung bereitzustellen.

Wenn Clientverbindungskanalobjekte definiert sind, werden sie in eine Clientkanaldefinitionsdatei geschrieben, deren Name standardmäßig AMQCLCHL.TAB lautet. Wenn die Clientverbindungskanäle das TCP/IP-Protokoll verwenden, werden sie vom IBM WebSphere MQ-Server auch in Active Directory veröffentlicht. Wenn der IBM WebSphere MQ-Client festlegt, wie eine Verbindung zum Server hergestellt wird, sucht er mithilfe der folgenden Suchreihenfolge nach einer relevanten Kanalobjektdefinition für den Clientverbindungskanal:

1. MQCD-Datenstruktur MQCONNX
2. Umgebungsvariable MQSERVER
3. Clientkanaldefinitionsdatei
4. Active Directory

Diese Reihenfolge bedeutet, dass alle aktuellen Anwendungen von keiner Änderung betroffen sind. Sie können sich diese Einträge in Active Directory als Datensätze in der Clientkanaldefinitionsdatei vorstellen, und der IBM WebSphere MQ-Client verarbeitet sie auf die gleiche Weise. Um die Unterstützung für die Veröffentlichung von Clientverbindungskanaldefinitionen in Active Directory zu konfigurieren und zu verwalten, verwenden Sie den Befehl setmqscp wie in [setmqscp](#) beschrieben.

Serververbindungskanal auf dem Server definieren

Erstellen Sie eine Serververbindungskanaldefinition für den Warteschlangenmanager.

Vorgehensweise

1. Definieren Sie auf der Servermaschine einen Kanal mit dem ausgewählten Namen und dem Kanaltyp *server-connection*.

Beispiel:

```
DEFINE CHANNEL(CHAN2) CHLTYPE(SVRCONN) TRPTYPE(TCP) +  
DESCR('Server-connection to Client_2')
```

2. Verwenden Sie den folgenden Befehl, um den Zugriff der Eingangsverbindung auf den Warteschlangenmanager zu ermöglichen:

```
SET CHLAUTH(CHAN2) TYPE(ADDRESSMAP) ADDRESS('IP address') MCAUSER('userid')
```

- Dabei verwendet SET CHLAUTH den Namen des Kanals, der im vorherigen Schritt definiert wurde.
- Dabei ist 'IP address' IP-Adresse die IP-Adresse des Clients.
- Dabei ist 'userid' die ID, die Sie dem Kanal für die Zugriffssteuerung für die Zielwarteschlangen bereitstellen möchten. Bei diesem Feld muss die Groß-/Kleinschreibung beachtet werden.

Sie können die eingehende Verbindung mithilfe einer Reihe unterschiedlicher Attribute identifizieren. Das Beispiel verwendet die IP-Adresse. Zu den alternativen Attributen gehören die Clientbenutzer-ID und der SSL- oder TLS-Betreffname. Weitere Informationen finden Sie in [Kanalauthentifizierungsdatensätze](#).

Diese Kanaldefinition ist dem Warteschlangenmanager zugeordnet, der auf dem Server ausgeführt wird.

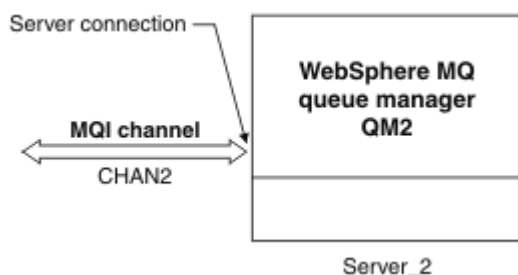


Abbildung 18. Serververbindungskanal definieren

Definieren des Clientverbindungskanals auf dem Server

Nachdem Sie den Serververbindungskanal definiert haben, definieren Sie nun den entsprechenden Clientverbindungskanal.

Vorbereitende Schritte

Definieren Sie den Serververbindungskanal.

Vorgehensweise

1. Definieren Sie einen Kanal mit demselben Namen wie der Serververbindungskanal, aber einen Kanaltyp von *Clientverbindung*. Sie müssen den Verbindungsnamen (CONNNAME) angeben. Bei TCP/IP ist der Verbindungsname die Netzadresse oder der Hostname der Servermaschine. Es empfiehlt sich außerdem, den Namen des Warteschlangenmanagers (QMNAME) anzugeben, zu dem Ihre in der Clientumgebung ausgeführte IBM WebSphere MQ-Anwendung eine Verbindung herstellen soll. Wenn Sie den Namen des Warteschlangenmanagers ändern, können Sie eine Gruppe von Kanälen definieren, um eine Verbindung zu verschiedenen Warteschlangenmanagern herzustellen.

```
DEFINE CHANNEL(CHAN2) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +  
CONNNAME(9.20.4.26) QMNAME(QM2) DESCR('Client-connection to Server_2')
```


2. Verwenden Sie den folgenden Befehl, um den Zugriff der Eingangsverbindung auf den Warteschlangenmanager zu ermöglichen:

```
SET CHLAUTH(CHAN2) TYPE(ADDRESSMAP) ADDRESS('IP-address') MCAUSER('userid')
```

- Dabei verwendet SET CHLAUTH den Namen des Kanals, der im vorherigen Schritt definiert wurde.
- Hierbei steht 'IP address' für die IP-Adresse des Clients.
- Dabei ist 'userid' die ID, die Sie dem Kanal für die Zugriffssteuerung für die Zielwarteschlangen bereitstellen möchten. Bei diesem Feld muss die Groß-/Kleinschreibung beachtet werden.

Sie können die eingehende Verbindung mithilfe einer Reihe unterschiedlicher Attribute identifizieren. Das Beispiel verwendet die IP-Adresse. Zu den alternativen Attributen gehören die Clientbenutzer-ID und der SSL- oder TLS-Betreffname. Weitere Informationen finden Sie in [Kanalauthentifizierungsdatensätze](#).

Ergebnisse

Auf anderen Plattformen als z/OS wird diese Kanaldefinition in einer Datei gespeichert, die als Definitionstabelle für Clientkanal (Client Channel Definition Table, CCDT) bezeichnet wird und dem Warteschlangenmanager zugeordnet ist. Die Definitionstabelle für den Clientkanal kann mehr als eine Kanaldefinition für Clientverbindungen enthalten. Weitere Informationen zur Definitionstabelle für den Clientkanal und zu den entsprechenden Informationen darüber, wie Clientverbindungskanaldefinitionen auf z/OS gespeichert werden, finden Sie in „Definitionstabelle für den Clientkanal“ auf Seite 125.

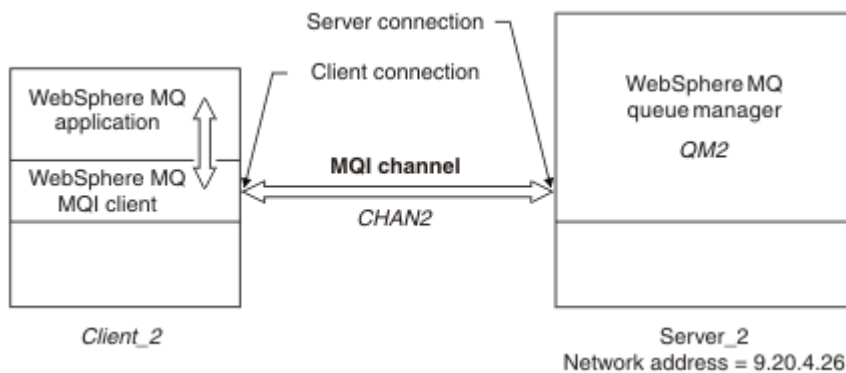


Abbildung 19. Definieren des Clientverbindungskanals

Zugreifen auf Clientverbindungskanaldefinitionen

Machen Sie die Definitionstabelle für Clientkanal (CCDT) Clientanwendungen zugänglich, indem Sie sie kopieren oder für die gemeinsame Nutzung freigeben, und geben Sie anschließend ihr Verzeichnis und ihren Namen auf dem Clientcomputer an.

Vorbereitende Schritte

Die erforderlichen Clientverbindungskanäle wurden definiert.

Unter z/OS haben Sie eine CCDT erstellt. Auf anderen Plattformen wird die CCDT automatisch erstellt und aktualisiert.

Informationen zu diesem Vorgang

Damit eine Clientanwendung die Definitionstabelle für Clientkanal (CCDT) verwenden kann, muss die CCDT für die Anwendung zugänglich gemacht und das Verzeichnis und der Name dieser Tabelle müssen angegeben werden.

Vorgehensweise

1. Sie haben drei Möglichkeiten, die CCDT den Clientanwendungen zugänglich zu machen:
 - a) Optional: Kopieren Sie die CCDT auf den Client-Computer.
 - b) Optional: Kopieren Sie die CCDT in eine Position, die von mehr als einem Client gemeinsam genutzt wird.
 - c) Optional: Lassen Sie die CCDT auf dem Server, lassen Sie sie jedoch vom Client shareable.
Die Position, die Sie für die CCDT auswählen, muss sicher sein, damit nicht autorisierte Änderungen an den Kanälen verhindert werden können.
2. Geben Sie auf dem Client die Position und den Namen der Datei an, die die CCDT auf eine der folgenden drei Arten enthält:

- a) Optional: Verwenden Sie die Zeilengruppe CHANNELS in der Clientkonfigurationsdatei. Weitere Informationen finden Sie unter „Zeilengruppe 'CHANNELS' in der Clientkonfigurationsdatei“ auf [Seite 147](#).
- b) Optional: Verwenden Sie die Umgebungsvariablen MQCHLLIB und MQCHLTAB.

Sie können die Umgebungsvariablen beispielsweise festlegen, indem Sie Folgendes eingeben:

- Auf Systemen mit HP Integrity NonStop Server und UNIX and Linux :

```
export MQCHLLIB=MQ_INSTALLATION_PATH/qmgrs/QUEUEMANAGERNAME/@ipcc
export MQCHLTAB=AMQCLCHL.TAB
```

Dabei steht *MQ_INSTALLATION_PATH* für das übergeordnete Verzeichnis, in dem WebSphere MQ installiert ist.

- c) Optional: Verwenden Sie nur unter Windows den Steuerbefehl **setmqscp** , um die Clientverbindungskanaldefinitionen in Active Directory zu veröffentlichen.

Wenn die Umgebungsvariable MQSERVER gesetzt ist, verwendet ein WebSphere MQ -Client die von MQSERVER angegebene Clientverbindungskanaldefinition anstelle von Definitionen in der Clientkanaldefinitionstabelle.

Kanalexitprogramme für MQI-Kanäle

Für die WebSphere MQ MQI-Clientumgebung auf UNIX-, Linux -und Windows -Systemen sind drei Typen von Kanalexits verfügbar.

Diese sind:

- Sendeexit
- Empfangsexit
- Sicherheitsexit

Diese Exits sind sowohl auf dem Client als auch auf dem Serverende des Kanals verfügbar. Die Exits sind für Ihre Anwendung nicht verfügbar, wenn Sie die Umgebungsvariable MQSERVER verwenden. Kanalexits werden im Abschnitt [Kanalexitprogramme für Messaging-Kanäle](#) erläutert.

Die Sende- und Empfangsexits arbeiten zusammen. Es gibt mehrere Möglichkeiten, wie Sie sie verwenden können:

- Nachricht trennen und erneut assemblieren
- Komprimieren und Dekomprimieren von Daten in einer Nachricht (diese Funktionalität wird als Teil von WebSphere MQ bereitgestellt, Sie können jedoch ein anderes Komprimierungsverfahren verwenden)
- Verschlüsseln und Entschlüsseln von Benutzerdaten (diese Funktion wird als Teil von WebSphere MQ bereitgestellt, aber Sie können ein anderes Verschlüsselungsverfahren verwenden)
- Journaling jeder gesendeten und empfangenen Nachricht

Sie können den Sicherheitsexit verwenden, um sicherzustellen, dass der WebSphere MQ -Client und -Server ordnungsgemäß identifiziert werden, und um den Zugriff zu steuern.

Wenn Sende-oder Empfangsexits auf der Serververbindungsseite der Kanalinstanz MQI-Aufrufe für die Verbindung ausführen müssen, der sie zugeordnet sind, verwenden sie die Verbindungskennung, die im Feld MQCXP Hconn angegeben ist. Sie müssen sich darüber im Klaren sein, dass Sende-und Empfangsexits für Clientverbindungen keine MQI-Aufrufe vornehmen können.

Zugehörige Konzepte

[„Sicherheitsexits auf einer Clientverbindung“ auf Seite 131](#)

Sie können Sicherheitsexitprogramme verwenden, um sicherzustellen, dass der Partner am anderen Ende eines Kanals echt ist. Besondere Hinweise gelten, wenn ein Sicherheitsexit auf eine Clientverbindung angewendet wird.

[Benutzerexits, API-Exits und installierbare WebSphere MQ -Services](#)

Zugehörige Tasks

[Funktionen des Warteschlangenmanagers erweitern](#)

Zugehörige Verweise

[„Pfad zu Exits“ auf Seite 131](#)

Ein Standardpfad für die Position der Kanalexits ist in der Clientkonfigurationsdatei definiert. Kanalexits werden geladen, wenn ein Kanal initialisiert wird.

[„API-Aufruf in einem Sende-oder Empfangsexitprogramm identifizieren“ auf Seite 133](#)

Wenn Sie MQI-Kanäle für Clients verwenden, identifiziert Byte 10 des Agentenpuffers den API-Aufruf, der verwendet wird, wenn ein Sende-oder Empfangsexit aufgerufen wird. Dies ist hilfreich, um zu ermitteln, welche Kanalflüsse Benutzerdaten enthalten und möglicherweise eine Verarbeitung wie die Verschlüsselung oder die digitale Signatur erfordern.

Pfad zu Exits

Ein Standardpfad für die Position der Kanalexits ist in der Clientkonfigurationsdatei definiert. Kanalexits werden geladen, wenn ein Kanal initialisiert wird.

Auf UNIX-, Linux -und Windows-Systemen wird Ihrem System während der Installation des WebSphere MQ MQI-Clients eine Clientkonfigurationsdatei hinzugefügt. In dieser Datei wird unter Verwendung der Zeilengruppe ein Standardpfad für die Position der Kanalexits auf dem Client definiert:

```
ClientExitPath:  
  ExitsDefaultPath=string  
  ExitsDefaultPath64=string
```

Dabei steht *string* für eine Dateiposition in einem Format, das der Plattform entspricht.

Wenn ein Kanal initialisiert wird, wird nach einem MQCONN -oder MQCONNX -Aufruf die Clientkonfigurationsdatei durchsucht. Die Zeilengruppe 'ClientExitPath' wird gelesen, und alle Kanalexits, die in der Kanaldefinition angegeben sind, werden geladen.

Sicherheitsexits auf einer Clientverbindung

Sie können Sicherheitsexitprogramme verwenden, um sicherzustellen, dass der Partner am anderen Ende eines Kanals echt ist. Besondere Hinweise gelten, wenn ein Sicherheitsexit auf eine Clientverbindung angewendet wird.

[Abbildung 20 auf Seite 133](#) veranschaulicht die Verwendung von Sicherheitsexits in einer Clientverbindung unter Verwendung des Objektberechtigungsmanagers von WebSphere MQ zur Authentifizierung eines Benutzers. Entweder 'SecurityParmsPtr' oder 'SecurityParmsOffset' wird in der MQCNO-Struktur auf dem Client definiert, und es sind Sicherheitsexits an beiden Enden des Kanals vorhanden. Nachdem der normale Sicherheitsnachrichtenaustausch beendet wurde und der Kanal für die Ausführung bereit ist, wird die MQCSP-Struktur, auf die über das Feld MQCXP SecurityParms zugegriffen wird, an den Sicherheitsexit auf dem Client übergeben. Der Exittyp wird auf MQXR_SEC_PARMS gesetzt. Der Sicherheitsexit kann die Benutzer-ID und das Kennwort nicht ändern, oder es kann eine oder beide der beiden Änderungen ändern. Die vom Exit zurückgegebenen Daten werden dann an das Serververbindungsende des Kanals gesendet. Die MQCSP-Struktur wird auf dem Serververbindungsende des Kanals wiederhergestellt und an den Sicherheitsexit der Serververbindung übergeben, auf den über das Feld MQCXP SecurityParms

zugegriffen wird. Der Sicherheitsexit empfängt und verarbeitet diese Daten. Bei dieser Verarbeitung wird normalerweise jede Änderung rückgängig gemacht, die an den Feldern für Benutzer-ID und Kennwort im Clientexit vorgenommen wurde, die dann zur Autorisierung der Verbindung des Warteschlangenmanagers verwendet werden. Die resultierende MQCSP-Struktur wird unter Verwendung von SecurityParmsPtr in der MQCNO-Struktur auf dem WS-Manager-System referenziert.

Wenn 'SecurityParmsPtr' oder 'SecurityParmsOffset' in der MQCNO-Struktur definiert sind und ein Sicherheitsexit nur an einem Ende des Kanals vorhanden ist, empfängt und verarbeitet der Sicherheitsexit die MQCSP-Struktur. Aktionen wie die Verschlüsselung sind für einen einzelnen Benutzerexit nicht geeignet, da kein Exit vorhanden ist, um die ergänzende Aktion auszuführen.

Wenn 'SecurityParmsPtr' und 'SecurityParmsOffset' nicht in der MQCNO-Struktur definiert sind und ein Sicherheitsexit an einem oder beiden Enden des Kanals vorhanden ist, werden die Sicherheitsexits oder -exits aufgerufen. Der Sicherheitsexit kann seine eigene MQCSP-Struktur zurückgeben, die über die SicherheitParmsPtr-Datei adressiert wird. Der Sicherheitsexit wird erst wieder aufgerufen, wenn er beendet wird (ExitReason von MQXR_TERM). Der Exit-Writer kann den Speicher, der für den MQCSP in dieser Phase verwendet wird, freilassen.

Wenn eine Serververbindungskanalinstanz mehr als einen Datenaustausch gemeinsam verwendet, ist das Muster der Aufrufe an den Sicherheitsexit auf den zweiten und nachfolgenden Datenaustausch beschränkt.

Für den ersten Datenaustausch ist das Muster so, als ob die Kanalinstanz keine Dialoge gemeinsam verwendet. Für den zweiten und nachfolgenden Datenaustausch wird der Sicherheitsexit niemals mit MQXR_INIT, MQXR_INIT_SEC oder MQXR_SEC_MSG aufgerufen. Es wird mit MQXR_SEC_PARMS aufgerufen.

In einer Kanalinstanz mit gemeinsamen Gesprächen wird MQXR_TERM nur für den letzten Datenaustausch aufgerufen, der ausgeführt wird.

Jeder Dialog hat die Möglichkeit, im MQXR_SEC_PARMS-Aufruf des Exits die MQCD zu ändern; auf dem Serververbindungsende des Kanals kann diese Funktion hilfreich sein, um beispielsweise die Werte MCAUserIdentifier oder LongMCAUserIdPtr zu ändern, bevor die Verbindung zum Warteschlangenmanager hergestellt wird.

Server-connection exit	Client-connection exit
	Invoked with MQXR_INIT Responds with MQXCC_OK
Invoked with MQXR_INIT Responds with MQXCC_OK	
	Invoked with MQXR_INIT_SEC Responds with MQXCC_OK
Invoked with MQXR_INIT_SEC Responds with MQXCC_OK	
	Invoked with MQXR_SEC_PARMS Responds with MQXCC_OK
Invoked with MQXR_SEC_PARMS Responds with MQXCC_OK	
Data transfer begins	
Invoked with MQXR_TERM Responds with MQXCC_OK	Invoked with MQXR_TERM Responds with MQXCC_OK

Abbildung 20. Clientverbindung-Eingeganger Austausch mit Vereinbarung für Clientverbindung unter Verwendung von Sicherheitsparametern

Anmerkung: Sicherheitsexitanwendungen, die vor WebSphere MQ 7.1 erstellt wurden, müssen unter Umständen aktualisiert werden. Weitere Informationen finden Sie im Abschnitt [Kanalsicherheits-Exitprogramme](#).

API-Aufruf in einem Sende-oder Empfangsexitprogramm identifizieren

Wenn Sie MQI-Kanäle für Clients verwenden, identifiziert Byte 10 des Agentenpuffers den API-Aufruf, der verwendet wird, wenn ein Sende-oder Empfangsexit aufgerufen wird. Dies ist hilfreich, um zu ermitteln, welche Kanalflüsse Benutzerdaten enthalten und möglicherweise eine Verarbeitung wie die Verschlüsselung oder die digitale Signatur erfordern.

In der folgenden Tabelle werden die Daten angezeigt, die in Byte 10 des Kanalfusses angezeigt werden, wenn ein API-Aufruf verarbeitet wird.

Anmerkung: Dies sind nicht die einzigen Werte dieses Bytes. Es sind weitere **reservierte** Werte vorhanden.

Tabelle 20. API-Aufrufe identifizieren		
API-Aufruf	Wert von Byte 10 für Anforderung	Wert von Byte 10 für Antwort
MQCONN „1“ auf Seite 134, „2“ auf Seite 134	X'81 '	X' 91 '
MQDISC „1“ auf Seite 134	X'82 '	X' 92 '

Tabelle 20. API-Aufrufe identifizieren (Forts.)

API-Aufruf	Wert von Byte 10 für Anforderung	Wert von Byte 10 für Antwort
MQOPEN „3“ auf Seite 134	X'83 '	X' 93 '
MQCLOSE	X'84 '	X' 94 '
MQGET „4“ auf Seite 134	X'85 '	X' 95 '
MQPUT „4“ auf Seite 134	X'86 '	X' 96 '
MQPUT1-Anforderung „4“ auf Seite 134	X'87 '	X' 97 '
MQSET-Anforderung	X'88 '	X' 98 '
MQINQ-Anforderung	X'89 '	X' 99 '
MQCMIT-Anforderung	X'8A '	X' 9A '
MQBACK-Anforderung	X'8B '	X' 9B '
MQSTAT-Anforderung	X'8D '	X' 9D '
MQSUB-Anforderung	X'8E '	X' 9E '
MQSUBRQ-Anforderung	X'8F '	X' 9F '
xa_start-Anforderung	X'A1 '	X'B1 '
xa_end-Anforderung	X'A2 '	X'B2 '
xa_open-Anforderung	X'A3 '	X'B3 '
xa_close-Anforderung	X'A4 '	X'B4 '
xa_prepare-Anforderung	X'A5 '	X'B5 '
xa_commit-Anforderung	X'A6 '	X'B6 '
xa_rollback-Anforderung	X'A7 '	X'B7 '
Anforderung 'xa_forget'	X'A8 '	X'B8 '
xa_recover, Anforderung	X'A9 '	X'B9 '
Anforderung 'xa_abgeschlossen'	X'AA'	X'BA '

Anmerkungen:

1. Die Verbindung zwischen dem Client und dem Server wird von der Clientanwendung unter Verwendung von MQCONN eingeleitet. Daher gibt es für diesen Befehl vor allem mehrere andere Netzabläufe. Dasselbe gilt für MQDISC, das die Netzverbindung beendet.
2. MQCONNX wird in der gleichen Weise wie MQCONN für die Zwecke der Client/Server-Verbindung behandelt.
3. Wenn eine große Verteilerliste geöffnet wird, kann es mehr als einen Netzfluss pro MQOPEN-Aufruf geben, um alle erforderlichen Daten an den SVRCONN-MCA zu übergeben.
4. Große Nachrichten können die Größe des Übertragungssegments überschreiten. Wenn dies geschieht, kann es viele Netzabläufe geben, die aus einem einzigen API-Aufruf resultieren.

Verbindung eines Clients mit einer Gruppe mit gemeinsamer Warteschlange herstellen

Sie können einen Client mit einer Gruppe mit gemeinsamer Warteschlange verbinden, indem Sie einen MQI-Kanal zwischen einem Client und einem WS-Manager auf einem Server erstellen, der Mitglied einer Gruppe mit gemeinsamer Warteschlange ist.

Eine Gruppe mit gemeinsamer Warteschlange wird von einer Gruppe von Warteschlangenmanagern gebildet, die auf dieselbe Gruppe gemeinsam genutzter Warteschlangen zugreifen können.

Ein Client, der in eine gemeinsam genutzte Warteschlange gestellt wird, kann eine Verbindung zu jedem Mitglied der Gruppe mit gemeinsamer Warteschlange herstellen. Die Vorteile einer Verbindung zu einer Gruppe mit gemeinsamer Warteschlange sind in der Front-End- und Back-End-Verfügbarkeit sowie in der Kapazitätserhöhung möglich. Sie können eine Verbindung zu einem bestimmten WS-Manager oder zu der generischen Schnittstelle herstellen.

Die direkte Verbindung zu einem Warteschlangenmanager in einer Gruppe mit gemeinsamer Warteschlange bietet den Vorteil, dass Sie Nachrichten in eine gemeinsam genutzte Zielwarteschlange stellen können, wodurch die Back-End-Verfügbarkeit erhöht wird.

Wenn Sie eine Verbindung zur generischen Schnittstelle einer Gruppe mit gemeinsamer Warteschlange herstellen, wird eine Sitzung mit einem der WS-Manager in der Gruppe geöffnet. Dies erhöht die Front-End-Verfügbarkeit, da der Client-WS-Manager eine Verbindung zu jedem WS-Manager in der Gruppe herstellen kann. Sie stellen über die generische Schnittstelle eine Verbindung zu der Gruppe her, wenn Sie keine Verbindung zu einem bestimmten Warteschlangenmanager innerhalb der Gruppe mit gemeinsamer Warteschlange herstellen möchten.

Die generische Schnittstelle kann ein WLM/DNS-Gruppenname, ein generischer VTAM -Ressourcenname oder eine andere allgemeine Schnittstelle zur Gruppe mit gemeinsamer Warteschlange sein.

Um eine Verbindung zur generischen Schnittstelle einer Gruppe mit gemeinsamer Warteschlange herzustellen, müssen Sie Kanaldefinitionen erstellen, auf die jeder WS-Manager in der Gruppe zugreifen kann. Dazu müssen auf jedem Warteschlangenmanager in der Gruppe dieselben Definitionen vorhanden sein.

Definieren Sie den SVRCONN-Kanal wie folgt:

```
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
MCAUSER(' ') QSGDISP(GROUP)
```

Kanaldefinitionen auf dem Server werden in einem gemeinsam genutzten DB2 -Repository gespeichert. Jeder WS-Manager in der Gruppe mit gemeinsamer Warteschlange stellt eine lokale Kopie der Definition her und stellt sicher, dass Sie immer eine Verbindung zum richtigen Serververbindungskanal herstellen, wenn Sie einen MQCONN- oder MQCONNX-Aufruf absetzen.

Definieren Sie den CLNTCONN-Kanal wie folgt:

```
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNAME(WLM/DNS groupname) QMNAME(QSG1) +
DESCR('Client-connection to Queue Sharing Group QSG1') QSGDISP(GROUP)
```

Da die generische Schnittstelle der Gruppe mit gemeinsamer Warteschlange im Feld CONNAME im Clientverbindungskanal gespeichert ist, können Sie jetzt eine Verbindung zu jedem WS-Manager in der Gruppe herstellen und sie in gemeinsam genutzte Warteschlangen stellen, deren Eigner diese Gruppe ist.

Zugehörige Konzepte

„Kanaldefinitionen erstellen“ auf Seite 136

Um eine Verbindung zur generischen Schnittstelle einer Gruppe mit gemeinsamer Warteschlange herzustellen, müssen Sie Kanaldefinitionen erstellen, auf die jeder WS-Manager in der Gruppe zugreifen kann. Dazu müssen auf jedem Warteschlangenmanager in der Gruppe dieselben Definitionen vorhanden sein.

Kanaldefinitionen erstellen

Um eine Verbindung zur generischen Schnittstelle einer Gruppe mit gemeinsamer Warteschlange herzustellen, müssen Sie Kanaldefinitionen erstellen, auf die jeder WS-Manager in der Gruppe zugreifen kann. Dazu müssen auf jedem Warteschlangenmanager in der Gruppe dieselben Definitionen vorhanden sein.

Definieren Sie den SVRCONN-Kanal wie folgt:

```
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
MCAUSER(' ') QSGDISP(GROUP)
```

Kanaldefinitionen auf dem Server werden in einem gemeinsam genutzten DB2 -Repository gespeichert. Jeder WS-Manager in der Gruppe mit gemeinsamer Warteschlange stellt eine lokale Kopie der Definition her und stellt sicher, dass Sie immer eine Verbindung zum richtigen Serververbindungskanal herstellen, wenn Sie einen MQCONN-oder MQCONNX-Aufruf absetzen.

Definieren Sie den CLNTCONN-Kanal wie folgt:

```
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNAME(WLM/DNS groupname) QMNAME(QSG1) +
DESCR('Client-connection to Queue Sharing Group QSG1') QSGDISP(GROUP)
```

Da die generische Schnittstelle der Gruppe mit gemeinsamer Warteschlange im Feld CONNAME im Clientverbindungskanal gespeichert ist, können Sie jetzt eine Verbindung zu jedem WS-Manager in der Gruppe herstellen und sie in gemeinsam genutzte Warteschlangen stellen, deren Eigner diese Gruppe ist.

Client mit einer Konfigurationsdatei konfigurieren

Konfigurieren Sie Ihre Clients mit Attributen in einer Textdatei. Diese Attribute können durch Umgebungsvariablen oder auf andere plattformspezifische Methoden überschrieben werden.

Sie konfigurieren IBM WebSphere MQ MQI client mithilfe einer Textdatei, die der auf UNIX and Linux -Plattformen verwendeten Konfigurationsdatei des Warteschlangenmanagers qm.ini ähnelt. Die Datei enthält eine Reihe von Zeilengruppen, von denen jede eine Anzahl Zeilen im Format **attribute-name=wert** enthält.

In dieser Dokumentation wird diese Datei als *WebSphere MQ MQI-Clientkonfigurationsdatei* bezeichnet; ihr Dateiname lautet im Allgemeinen mqclient.ini, Sie können ihr jedoch einen anderen Namen geben. Die Konfigurationsdaten in dieser Datei gelten für alle Plattformen und für Clients, die MQI, IBM WebSphere MQ classes for Java, IBM WebSphere MQ classes for JMS, IBM WebSphere MQ classes for .NET und XMS verwenden.

Obwohl die Attribute in der IBM WebSphere MQ MQI client-Konfigurationsdatei für die meisten IBM WebSphere MQ-Clients gelten, gibt es bestimmte Attribute, die nicht von verwalteten .NET- und XMS .NET-Clients oder von Clients verwendet werden, die entweder die IBM WebSphere MQ classes for Java oder die IBM WebSphere MQ classes for JMS verwenden. Weitere Informationen finden Sie unter [„Welche IBM WebSphere MQ-Clients können die einzelnen Attribute lesen“](#) auf Seite 139.

Die Konfigurationsfunktionen gelten für alle Verbindungen, die eine Clientanwendung zu Warteschlangenmanagern herstellt, nicht für eine bestimmte Verbindung zu einem Warteschlangenmanager. Attribute, die sich auf eine Verbindung zu einem einzelnen Warteschlangenmanager beziehen, können programmgestützt konfiguriert werden, z. B. mithilfe einer MQCD-Struktur oder mithilfe einer Clientkanaldefinitionstabelle (CCDT).

Umgebungsvariablen, die in Releases von IBM WebSphere MQ vor Version 7.0 unterstützt wurden, werden weiterhin unterstützt, und wenn eine solche Umgebungsvariable mit einem entsprechenden Wert in der Clientkonfigurationsdatei übereinstimmt, überschreibt die Umgebungsvariable den Wert der Clientkonfigurationsdatei.

Für eine Clientanwendung, die IBM WebSphere MQ classes for JMS verwendet, können Sie die Clientkonfigurationsdatei auch wie folgt überschreiben:

- Eigenschaften in der JMS-Konfigurationsdatei festlegen

- Java-Systemeigenschaften festlegen, die auch die JMS-Konfigurationsdatei überschreiben

Für den .NET-Client können Sie auch die Clientkonfigurationsdatei und die entsprechenden Umgebungsvariablen mithilfe der .NET-Anwendungskonfigurationsdatei überschreiben.

Beachten Sie, dass Sie nicht mehrere Kanalverbindung mit der Clientkonfigurationsdatei einrichten können.

Beispiel einer Clientkonfigurationsdatei

```

** Module Name: mqclient.ini                **
** Type       : WebSphere MQ MQI client configuration file      **
** Function    : Define the configuration of a client           **
**                                                    **
*****
** Notes      :
** 1) This file defines the configuration of a client           **
**                                                    **
*****

ClientExitPath:
  ExitsDefaultPath=/var/mqm/exits
  ExitsDefaultPath64=/var/mqm/exits64

TCP:
  Library1=DLLName1
  KeepAlive = Yes
  ClntSndBuffSize=32768
  ClntRcvBuffSize=32768
  Connect_Timeout=0

MessageBuffer:
  MaximumSize=-1
  Updatepercentage=-1
  PurgeTime=0

LU62:
  TPName
  Library1=DLLName1
  Library2=DLLName2

PreConnect:
  Module=amqldapi
  Function=myFunc
  Data=ldap://myLDAPServer.com:389/cn=wmq,ou=ibm,ou=com
  Sequence=1

CHANNELS:
  DefRecon=YES
  ServerConnectionParms=SALES.SVRCONN/TCP/hostname.x.com(1414)

```

Zugehörige Verweise

„Position der Clientkonfigurationsdatei“ auf Seite 138

Eine IBM WebSphere MQ MQI-Clientkonfigurationsdatei kann an mehreren Positionen gespeichert werden.

„Zeilengruppe 'CHANNELS' in der Clientkonfigurationsdatei“ auf Seite 147

Verwenden Sie die Zeilengruppe CHANNELS, um Informationen zu Clientkanälen anzugeben.

„ClientExitPath, Zeilengruppe der Clientkonfigurationsdatei“ auf Seite 150

Verwenden Sie die Zeilengruppe "ClientExitPath", um die Standardpositionen von Kanalexits auf dem Client anzugeben.

„LU62-, NETBIOS- und SPX-Zeilengruppen in der Clientkonfigurationsdatei“ auf Seite 150

Verwenden Sie diese Zeilengruppen nur auf Windows-Systemen, um Konfigurationsparameter für die angegebenen Netzprotokolle anzugeben.

„MessageBuffer-Zeilengruppe der Clientkonfigurationsdatei“ auf Seite 151

Verwenden Sie die Zeilengruppe 'MessageBuffer', um Informationen zu Nachrichtenpuffern anzugeben.

„SSL-Zeilengruppe der Clientkonfigurationsdatei“ auf Seite 152

Geben Sie über die SSL-Zeilengruppe Informationen über die Verwendung von SSL oder TLS an.

„TCP-Zeilengruppe der Clientkonfigurationsdatei“ auf Seite 155

Verwenden Sie die TCP-Zeilengruppe, um die Konfigurationsparameter des TCP-Netzprotokolls anzugeben.

„WebSphere MQ -Umgebungsvariablen verwenden“ auf Seite 156

Dieser Abschnitt beschreibt die Umgebungsvariablen, die Sie mit WebSphere MQ MQI-Clientanwendungen verwenden können.

„Konfigurationsdaten des Warteschlangenmanagers ändern“ auf Seite 467

Über die hier beschriebenen Attribute wird die Konfiguration eines einzelnen Warteschlangenmanagers geändert. Sie überschreiben alle Einstellungen für WebSphere MQ.

Position der Clientkonfigurationsdatei

Eine IBM WebSphere MQ MQI-Clientkonfigurationsdatei kann an mehreren Positionen gespeichert werden.

Eine Clientanwendung verwendet den folgenden Suchpfad, um die IBM WebSphere MQ MQI-Clientkonfigurationsdatei zu lokalisieren:

1. Die Position, die durch die Umgebungsvariable MQCLNTCF angegeben wird.

Das Format dieser Umgebungsvariablen ist eine vollständige URL. Dies bedeutet, dass der Dateiname nicht unbedingt `mqclient.ini` sein muss, und erleichtert das Platzieren der Datei in einem mit einem Netz verbundenen Dateisystem.

Dabei ist Folgendes zu beachten:

- C-, .NET- und XMS -Clients unterstützen nur das `file:`-Protokoll; das `file:`-Protokoll wird vorausgesetzt, wenn die URL-Zeichenfolge nicht mit `protocol:` beginnt
- Für Java 1.4.2 JREs, die das Lesen von Umgebungsvariablen nicht unterstützen, kann die MQCLNTCF-Umgebungsvariable mit einer MQCLNTCF-Java-Systemeigenschaft überschrieben werden.

2. Eine Datei mit dem Namen `mqclient.ini` in dem aktuellen Arbeitsverzeichnis der Anwendung.

3. Eine Datei mit dem Namen `mqclient.ini` im IBM WebSphere MQ -Datenverzeichnis für Windows-, UNIX and Linux -Systeme.

Dabei ist Folgendes zu beachten:

- Das IBM WebSphere MQ -Datenverzeichnis ist auf bestimmten Plattformen nicht vorhanden, z. B. IBM i und z/OS, oder in Fällen, in denen der Client mit einem anderen Produkt bereitgestellt wurde.
- Auf UNIX and Linux -Systemen ist dies das Verzeichnis `/var/mqm`
- Auf Windows -Plattformen konfigurieren Sie die Umgebungsvariable `MQ_FILE_PATH` während der Installation so, dass sie auf das Datenverzeichnis verweist. Normalerweise handelt es sich um `C:\Program Files\IBM\WebSphere MQ`
- Um Java 1.4.2 JREs zuzulassen, die das Lesen von Umgebungsvariablen nicht unterstützen, können Sie die Umgebungsvariable `MQ_FILE_PATH` manuell mit einer Java-Systemeigenschaft `MQ_FILE_PATH` überschreiben.

4. Eine Datei mit dem Namen `mqclient.ini` in einem Standardverzeichnis, das für die Plattform geeignet ist und für die Benutzer zugänglich ist:

- Für alle Java-Clients ist dies der Wert der Java-Systemeigenschaft `user.home`.
- Für C-Clients auf UNIX and Linux-Plattformen ist dies der Wert der Umgebungsvariablen `HOME`.
- Für C-Clients unter Windows sind dies die verketteten Werte der Umgebungsvariablen `HOMEDRIVE` und `HOMEPATH`.

Anmerkung: Für den IBM WebSphere MQ-Client für HP Integrity NonStop Server muss sich die Datei `mqclient.ini` im OSS-Dateisystem befinden. Guardian-Anwendungen müssen entweder die Datei `mqclient.ini` in das IBM WebSphere MQ-Datenverzeichnis stellen oder die Umgebungsvariable `MQCLNTCF` auf eine Position im OSS-Dateisystem setzen.

Welche IBM WebSphere MQ-Clients können die einzelnen Attribute lesen

Die meisten Attribute in der IBM WebSphere MQ MQI client-Konfigurationsdatei können vom C-Client und von den nicht verwalteten .NET-Clients verwendet werden. Allerdings gibt es einige Attribute, die nicht von verwalteten .NET- und XMS .NET-Clients gelesen werden, oder von Clients, die entweder IBM WebSphere MQ classes for Java oder IBM WebSphere MQ classes for JMS verwenden.

<i>Tabelle 21. Welche Attribute gelten für jeden Typ von Client</i>						
mqcli-ent.ini, Zeilengruppenname und Attribute	Beschreibung	C und nicht verwaltete .NET	Java	JMS	Verwaltete .NET-Instanz	Verwaltete XMS .NET-Instanz
CHANNELS-Zeilengruppe						
<u>CCSID</u>	Die Nummer des codierten Zeichensatzes, der verwendet werden soll.	Ja	Nein	Nein	Ja	Ja
<u>ChannelDefinitionDirectory</u>	Der Verzeichnispfad zu der Datei, die die Definitionstabelle für den Clientkanal enthält.	Ja	Nein	Nein	Ja	Ja
<u>ChannelDefinitionFile</u>	Der Name der Datei, die die Definitionstabelle für den Clientkanal enthält.	Ja	Nein	Nein	Ja	Ja
<u>ReconDelay</u>	Eine Verwaltungsoption, mit der die Verzögerung für die Verbindungswiederverbindung für Clientprogramme konfiguriert werden kann, die die Verbindung automatisch herstellen können.	Ja	Nein	Ja	Ja	Ja

Tabelle 21. Welche Attribute gelten für jeden Typ von Client (Forts.)

mqcli-ent.ini, Zeilengruppenname und Attribute	Beschreibung	C und nicht verwaltete .NET	Java	JMS	Verwaltete .NET-Instanz	Verwaltete XMS .NET-Instanz
<u>DefRecon</u>	Eine Verwaltungsoption, mit der Clientprogramme automatisch erneuert verbunden werden können, oder um die automatische Neuverbindung eines Clientprogramms zu inaktivieren, das so geschrieben wurde, dass es automatisch wieder verbunden wird.	Ja	Nein	Ja	Ja	Ja
<u>MQReconnectTimeout</u>	Das Zeitlimit in Sekunden, in dem die Verbindung zu einem Client wieder hergestellt werden kann.	Ja	Nein	Nein	Ja	Nein
<u>ServerConnectionParms</u>	Die Position des IBM WebSphere MQ-Servers und das zu verwendende Kommunikationsverfahren.	Ja	Nein	Nein	Ja	Ja

Tabelle 21. Welche Attribute gelten für jeden Typ von Client (Forts.)

mqcli-ent.ini, Zeilengruppenname und Attribute	Beschreibung	C und nicht verwaltete .NET	Java	JMS	Verwaltete .NET-Instanz	Verwaltete XMS .NET-Instanz
Put1DefaultAlwaysSync	Steuert das Verhalten des Funktionsaufrufs MQPUT1 mit der Option MQPMO_RESPONSE_AS_QDEF.	Ja	Ja	Ja	Ja	Ja
Zeilengruppe 'ClientExitPath'						
ExitsDefaultPath	Gibt die Position der 32-Bit-Kanal-exits für Clients an.	Ja	Nein	Nein	Ja	Ja
ExitsDefaultPath64	Gibt die Position der 64-Bit-Kanal-exits für Clients an.	Ja	Nein	Nein	Ja	Ja
JavaExitsClassPath	Die Werte, die dem Klassenpfad hinzugefügt werden sollen, wenn ein Java-Exit ausgeführt wird.	Nein	Ja	Ja	Nein	Nein
Zeilengruppe 'MessageBuffer'						
MaximumSize	Größe (in Kilobyte) des Read-Ahead-Puffers im Bereich von 1 bis 999 999.	Ja	Ja	Ja	Ja	Ja

Tabelle 21. Welche Attribute gelten für jeden Typ von Client (Forts.)

mqcli-ent.ini, Zeilengruppenname und Attribute	Beschreibung	C und nicht verwaltete .NET	Java	JMS	Verwaltete .NET-Instanz	Verwaltete XMS .NET-Instanz
<u>PurgeTime</u>	Intervall (in Sekunden), nach dem Nachrichten, die im Puffer für die Lese-puffer (Read-ahead-Puffer) bleiben, gelöscht	Ja	Ja	Ja	Ja	Ja
<u>UpdatePercentage</u>	Der Prozentwert für die Aktualisierung im Bereich von 1-100, der bei der Berechnung des Schwellen-werts verwendet wird, um zu ermitteln, wann eine Clientanwendung eine neue Anforderung an den Server stellt.	Ja	Ja	Ja	Ja	Ja
SSL-Zeilengruppe						
<u>CDPCheckExtensions</u>	Gibt an, ob SSL-oder TLS-Kanäle in diesem Warteschlangenmanager versuchen, CDP-Server zu überprüfen, die in den Zertifikatserweiterungen des CrIDistributionPoint-Zertifikats benannt sind.	Ja	Nein	Nein	Nein	Nein

Tabelle 21. Welche Attribute gelten für jeden Typ von Client (Forts.)

mqcli-ent.ini, Zeilengruppenname und Attribute	Beschreibung	C und nicht verwaltete .NET	Java	JMS	Verwaltete .NET-Instanz	Verwaltete XMS .NET-Instanz
Certificate-Label	Die Zertifikatsbezeichnung der Kanaldefinition.	Ja	Nein	Nein	Nein	Nein
Certificate-ValPolicy	Bestimmt den Typ der verwendeten Zertifikatsprüfung.	Ja	Nein	Nein	Nein	Nein
ClientRevocationChecks	Legt fest, wie die Überprüfung der Zertifikatswiderufung konfiguriert wird, wenn der Clientverbindungsaufruf einen SSL/TLS-Kanal verwendet.	Ja	Nein	Nein	Nein	Nein
Encryption-PolicySuiteB	Legt fest, ob ein Kanal eine Suite-B-kompatible Verschlüsselung verwendet und welcher Grad der Stärke verwendet werden soll.	Ja	Nein	Nein	Nein	Nein

Tabelle 21. Welche Attribute gelten für jeden Typ von Client (Forts.)

mqcli-ent.ini, Zeilengruppenname und Attribute	Beschreibung	C und nicht verwaltete .NET	Java	JMS	Verwaltete .NET-Instanz	Verwaltete XMS .NET-Instanz
<u>OCSPAu-thentication</u>	Definiert das Verhalten von IBM WebSphere MQ, wenn OCSP aktiviert ist und die OCSP-Widerrufsprüfung nicht in der Lage ist, den Status des Zertifikatswiderrufs zu ermitteln.	Ja	Nein	Nein	Nein	Nein
<u>OCSPCheckExtensions</u>	Steuert, ob IBM WebSphere MQ auf die AuthorityInfoAccess-Zertifikatserweiterungen einwirkt.	Ja	Nein	Nein	Nein	Nein
<u>SSLCryptoHardware</u>	Legt die Parameterzeichenfolge fest, die erforderlich ist, um die auf dem System vorhandene Verschlüsselungshardware PKCS #11 zu konfigurieren.	Ja	Nein	Nein	Nein	Nein


Tabelle 21. Welche Attribute gelten für jeden Typ von Client (Forts.)

mqcli-ent.ini, Zeilengruppenname und Attribute	Beschreibung	C und nicht verwaltete .NET	Java	JMS	Verwaltete .NET-Instanz	Verwaltete XMS .NET-Instanz
SSLFipsRequired	Gibt an, ob nur FIPS-zertifizierte Algorithmen verwendet werden sollen, wenn die Verschlüsselung in IBM WebSphere MQ ausgeführt wird.	Ja	Nein	Nein	Nein	Nein
SSLHTTPProxyName	Die Zeichenfolge ist entweder der Hostname oder die Netzadresse des HTTP-Proxy-Servers, der von GSKit für OCSP-Prüfungen verwendet werden soll.	Ja	Nein	Nein	Nein	Nein
SSLKeyRepository	Die Position des Schlüsselrepositorys, in dem das digitale Zertifikat des Benutzers gespeichert ist, im Stammformat.	Ja	Nein	Nein	Nein	Nein

Tabelle 21. Welche Attribute gelten für jeden Typ von Client (Forts.)

mqcli-ent.ini, Zeilengruppenname und Attribute	Beschreibung	C und nicht verwaltete .NET	Java	JMS	Verwaltete .NET-Instanz	Verwaltete XMS .NET-Instanz
<u>SSLKeyResetCount</u>	Die Anzahl der nicht verschlüsselten Byte, die in einem SSL- oder TLS-Kanal gesendet und empfangen wurden, bevor der geheime Schlüssel neu verhandelt wird.	Ja	Nein	Nein	Nein	Nein
TCP-Zeilengruppe						
<u>ClntRcvBuffSize</u>	Die Größe des TCP/IP-Empfangspuffers (in Byte), der vom Clientende einer Clientverbindungs-Serververbindungskanals verwendet wird.	Ja	Ja	Ja	Ja	Ja
<u>ClntSndBuffSize</u>	Die Größe des TCP/IP-Sendepuffers (in Byte), der vom Clientende einer Clientverbindungs-Serververbindungskanals verwendet wird.	Ja	Ja	Ja	Ja	Ja

Tabelle 21. Welche Attribute gelten für jeden Typ von Client (Forts.)

mqcli-ent.ini, Zeilengruppenname und Attribute	Beschreibung	C und nicht verwaltete .NET	Java	JMS	Verwaltete .NET-Instanz	Verwaltete XMS .NET-Instanz
<u>Connect_Timeout</u>	Die Anzahl der Sekunden, bevor ein Versuch unternommen wird, das Socket-Zeitlimit zu verbinden.	Ja	Ja	Ja	Nein	Nein
<u>IPAddressVersion</u>	Gibt das IP-Protokoll an, das für eine Kanalverbindung verwendet werden soll.	Ja	Nein	Nein	Ja	Ja
<u>KeepAlive</u>	Schaltet die KeepAlive-Funktion ein oder aus.	Ja	Ja	Ja	Ja	Ja
 <u>Library1</u>	Nur unter Windows: der Name der TCP/IP-Sockets-DLL.	Ja	Nein	Nein	Nein	Nein

Für die HP Integrity NonStop Server können Sie die Zeilengruppen TMF und TmfGateway verwenden, um mit dem TMF-Gateway zu kommunizieren.

Zeilengruppe 'CHANNELS' in der Clientkonfigurationsdatei

Verwenden Sie die Zeilengruppe CHANNELS, um Informationen zu Clientkanälen anzugeben.

Die folgenden Attribute können in die Zeilengruppe CHANNELS aufgenommen werden:

CCSID=number

Die Nummer des codierten Zeichensatzes, der verwendet werden soll.

Die CCSID-Nummer entspricht dem Parameter der MQCCSID-Umgebung.

ChannelDefinitionDirectory=Pfad

Der Verzeichnispfad zu der Datei, die die Definitionstabelle für den Clientkanal enthält.

Auf Windows -Systemen ist das Standardverzeichnis das IBM WebSphere MQ -Installationsverzeichnis, in der Regel C:\Program Files\IBM\WebSphere MQ. Auf UNIX and Linux-Systemen ist das Standardverzeichnis /var/mqm.

Der Pfad "ChannelDefinitionDirectory" entspricht dem Umgebungsparameter MQCHLLIB.

ChannelDefinitionFile=Dateiname| AMQCLCHL.TAB

Der Name der Datei, die die Definitionstabelle für den Clientkanal enthält.

Die Definitionstabelle für den Clientkanal ist äquivalent zum Umgebungsparameter MQCHLTAB.

ReconDelay=(Verzögerung[,Zufallsintervall])(Verzögerung[,Zufallsintervall])...

Das Attribut ReconDelay stellt eine Verwaltungsoption zur Verfügung, mit der die Verzögerung der Verbindungswiederherleitung für Clientprogramme konfiguriert werden kann, die die Verbindung automatisch wiederherstellen können. Nachfolgend ist eine Beispielkonfiguration zu finden:

```
ReconDelay=(1000,200)(2000,200)(4000,1000)
```

Das gezeigte Beispiel definiert eine Anfangsverzögerung von einer Sekunde plus einem zufälligen Intervall von bis zu 200 Millisekunden. Die nächste Verzögerung beträgt zwei Sekunden plus ein zufälliger Intervall von bis zu 200 Millisekunden. Alle nachfolgenden Verzögerungen sind vier Sekunden, plus ein zufälliger Zeitraum von bis zu 1000 Millisekunden.

DefRecon=NO|YES|QMGR|DISABLED

Das Attribut DefRecon bietet eine Verwaltungsoption, mit der Clientprogramme automatisch erneut verbunden werden können, oder um die automatische Neuverbindung eines Clientprogramms zu inaktivieren, das so geschrieben wurde, dass es automatisch wieder verbunden wird. Wenn ein Programm eine Option verwendet, wie z. B. MQPMO_LOGICAL_ORDER, können Sie diese Option auswählen, wenn ein Programm mit der Neuverbindung nicht kompatibel ist.

Die Interpretation der Optionen DefRecon hängt davon ab, ob ein MQCNO_RECONNECT_* -Wert auch im Clientprogramm festgelegt ist und welcher Wert festgelegt ist.

Wenn das Clientprogramm eine Verbindung mit MQCONNherstellt oder die Option MQCNO_RECONNECT_AS_DEF mit MQCONNXfestlegt, wird der von DefRecon festgelegte Wert für die Verbindungswiederherstellung wirksam. Wenn im Programm kein Wert für die Verbindungswiederherstellung oder die Option DefRecon festgelegt ist, wird das Clientprogramm nicht automatisch erneut verbunden.

Die automatische Clientverbindungswiederholung wird von IBM WebSphere MQ -Klassen für Java nicht unterstützt.

NO

Sofern nicht von MQCONNXüberschrieben, wird die Clientverbindung nicht automatisch wiederhergestellt.

YES

Wenn nicht durch MQCONNXüberschrieben, stellt der Client die Verbindung automatisch wieder her.

QMGR

Wenn nicht durch MQCONNXüberschrieben, stellt der Client die Verbindung automatisch wieder her, aber nur mit demselben Warteschlangenmanager. Die Option QMGR hat dieselbe Wirkung wie MQCNO_RECONNECT_Q_MGR.

INAKTIVIERT

Die Verbindungswiederholung ist inaktiviert, auch wenn sie vom Clientprogramm mit dem MQI-Aufruf MQCONNX angefordert wird.

Die automatische Neuverbindung des Clients hängt von zwei Werten ab:

- Die in der Anwendung festgelegte Verbindungswiederverbindungsoption
- DefRecon -Wert in der mqclient.ini -Datei

Tabelle 22. Die automatische Verbindungswiederherstellung hängt von den Werten ab, die in der Anwendung und in der Datei mqclient.ini festgelegt sind.

DefRecon-Wert in der mqclient.ini	In der Anwendung festgelegte Verbindungswiederholungsoptionen			
	MQCNO_RECONNECT	MQCNO_RECONNECT_Q_MGR	MQCNO_RECONNECT_AS_DEF	MQCNO_RECONNECT_DISABLED

Tabelle 22. Die automatische Verbindungswiederherstellung hängt von den Werten ab, die in der Anwendung und in der Datei `mqclient.ini` festgelegt sind. (Forts.)

DefRecon- Wert in der mqcli- ent.ini	In der Anwendung festgelegte Verbindungswiederholungsoptionen			
	NO	YES	QMGR	Nein
YES	YES	QMGR	YES	Nein
QMGR	YES	QMGR	QMGR	Nein
INAKTI- VIERT	Nein	Nein	Nein	Nein

MQReconnectTimeout

Das Zeitlimit in Sekunden, in dem die Verbindung zu einem Client wieder hergestellt werden kann. Der Standardwert ist 1800 Sekunden (30 Minuten).

IBM WebSphere MQ classes for XMS .NET-Clients können über die Eigenschaft `XMSC.WMQ_CLIENT_RECONNECT_TIMEOUT` ein Zeitlimit für die Verbindungswiederherstellung angeben. Der Standardwert für diese Eigenschaft ist 1800 Sekunden (30 Minuten).

ServerConnectionParms

Die Parameter `ServerConnectionParms` entsprechen dem Umgebungsparameter `MQSERVER` und geben die Position des IBM WebSphere MQ -Servers und das zu verwendende Übertragungsverfahren an. Das Attribut `ServerConnectionParms` definiert nur einen einfachen Kanal; Sie können mit ihm keinen SSL-Kanal oder einen Kanal mit Kanalexits definieren. Es handelt sich um eine Zeichenfolge im Format `ChannelName/TransportType/ConnectionName, ConnectionName` muss ein vollständig qualifizierter Netzname sein. Der `Kanalname` darf keinen Schrägstrich ("/) enthalten, da mit diesem Zeichen der Kanalname, Transporttyp und Verbindungsname getrennt werden.

Wenn "`ServerConnectionParms`" zum Definieren eines Clientkanals verwendet wird, wird eine maximale Nachrichtenlänge von 100 MB verwendet. Daher ist die maximale Nachrichtengröße, die für den Kanal wirksam ist, der im `SVRCONN`-Kanal auf dem Server angegebene Wert.

Beachten Sie, dass nur eine einzige Clientkanalverbindung hergestellt werden kann. Wenn Sie z. B. zwei Einträge haben:

```
ServerConnectionParms=R1.SVRCONN/TCP/localhost(1963)
ServerConnectionParms=R2.SVRCONN/TCP/localhost(1863)
```

wird nur die zweite verwendet.

Geben Sie `ConnectionName` als durch Kommas getrennte Liste mit Namen für den angegebenen Transporttyp an. Im Allgemeinen ist nur ein Name erforderlich. Sie können mehrere `hostnames` bereitstellen, um mehrere Verbindungen mit den gleichen Eigenschaften zu konfigurieren. Die Verbindungen werden in der Reihenfolge versucht, in der sie in der Verbindungsliste angegeben sind, bis eine Verbindung erfolgreich hergestellt wurde. Wenn keine Verbindung erfolgreich ist, beginnt der Client erneut zu verarbeiten. Verbindungslisten sind eine Alternative zu WS-Manager-Gruppen, um Verbindungen für wiederverbindbare Clients zu konfigurieren.

Put1DefaultAlwaysSync=NEIN|JA

Steuert das Verhalten des Funktionsaufrufs `MQPUT1` mit der Option `MQPMO_RESPONSE_AS_Q_DEF`.

Nein

Wenn `MQPUT1` mit `MQPMO_SYNCPOINT` festgelegt ist, verhält es sich wie `MQPMO_ASYNC_RESPONSE`. Wenn `MQPUT1` mit `MQPMO_NO_SYNCPOINT` festgelegt ist, verhält es sich ebenfalls wie `MQPMO_SYNC_RESPONSE`. Dies ist der Standardwert.

YES

MQPUT1 verhält sich so, als ob MQPMO_SYNC_RESPONSE gesetzt ist, unabhängig davon, ob MQPMO_SYNCPOINT oder MQPMO_NO_SYNCPOINT festgelegt ist.

ClientExitPath, Zeilengruppe der Clientkonfigurationsdatei

Verwenden Sie die Zeilengruppe "ClientExitPath", um die Standardpositionen von Kanalexits auf dem Client anzugeben.

Die folgenden Attribute können in die ClientExitPath-Zeilengruppe aufgenommen werden:

ExitsDefaultPath=Zeichenfolge

Gibt die Position der 32-Bit-Kanalexits für Clients an.

ExitsDefaultPath64=Zeichenfolge

Gibt die Position der 64-Bit-Kanalexits für Clients an.

JavaExitsClassPath=Zeichenfolge

Die Werte, die dem Klassenpfad hinzugefügt werden sollen, wenn ein Java-Exit ausgeführt wird. Dies wird von Exits in einer anderen Sprache ignoriert.

In der JMS-Konfigurationsdatei erhält der ClassPath von JavaExits den Standardnamen com.ibm.mq.cfg. Präfix und dieser vollständige Name werden auch in der Systemeigenschaft WebSphere MQ V7.0 verwendet. In Version 6.0 wurde dieses Attribut mit der Systemeigenschaft com.ibm.mq.exitClasspath angegeben, die in der Readme-Datei zu Version 6.0 dokumentiert ist. Die Verwendung von 'com.ibm.mq.exitClasspath' ist nicht mehr weiter unterstützt. Wenn sowohl 'JavaExitsClassPath' als auch 'exitClasspath' vorhanden sind, wird 'JavaExitsClassPath' berücksichtigt. Wenn nur die Verwendung von exitClasspath vorhanden ist, wird sie in WebSphere MQ V7.0 berücksichtigt.

LU62-, NETBIOS- und SPX-Zeilengruppen in der Clientkonfigurationsdatei

Verwenden Sie diese Zeilengruppen nur auf Windows-Systemen, um Konfigurationsparameter für die angegebenen Netzprotokolle anzugeben.

LU62

Verwenden Sie die Zeilengruppe LU62, um die SNA LU 6.2-Protokollkonfigurationsparameter anzugeben. Die folgenden Attribute können in diese Zeilengruppe aufgenommen werden:

Library1=DL-Name|_WCPIC32

Der Name der APPC-DLL.

Library2=DL-Name|_WCPIC32

Dasselbe wie Library1, wenn der Code in zwei separaten Bibliotheken gespeichert ist.

TP-Name

Der Name des TP-Namens, der auf dem fernen Standort gestartet werden soll.

NETBIOS

Verwenden Sie die NETBIOS-Zeilengruppe, um die Konfigurationsparameter für das NetBIOS-Protokoll anzugeben. Die folgenden Attribute können in diese Zeilengruppe aufgenommen werden:

AdapterNum=Zahl|_0

Die Nummer des LAN-Adapters.

Library1=DL-Name|_NETAPI32

Der Name der NetBIOS-DLL.

LocalName=Name

Der Name, unter dem dieser Computer im LAN bekannt ist.

Dies entspricht dem Umgebungsparameter MQNAME.

NumCmds=Zahl|_1

Gibt an, wie viele Befehle zugeordnet werden sollen.

NumSess=Zahl|1

Gibt an, wie viele Sitzungen zugeordnet werden sollen.

SPX

Verwenden Sie die SPX-Zeilengruppe, um SPX-Protokollkonfigurationsparameter anzugeben. Die folgenden Attribute können in diese Zeilengruppe aufgenommen werden:

BoardNum=Zahl|_0

Die LAN-Adapternummer.

KeepAlive=YES|NO

Schalten Sie die KeepAlive-Funktion ein oder aus.

KeepAlive=YES bewirkt, dass SPX in regelmäßigen Abständen prüft, ob das andere Ende der Verbindung noch verfügbar ist. Ist dies nicht der Fall, wird der Kanal geschlossen.

Library1=DL-Name|_WSOCK32.DLL

Der Name der SPX-DLL.

Library2=DL-Name|_WSOCK32.DLL

Wie Library1, wird verwendet, wenn der Code in zwei separaten Bibliotheken gespeichert ist.

Socket=Zahl|5E86

Die SPX-Socket-Nummer in Hexadezimalschreibweise.

MessageBuffer-Zeilengruppe der Clientkonfigurationsdatei

Verwenden Sie die Zeilengruppe 'MessageBuffer', um Informationen zu Nachrichtenpuffern anzugeben.

Die folgenden Attribute können in der Zeilengruppe 'MessageBuffer' enthalten sein:

MaximumSize=Ganzzahl|_1

Größe (in Kilobyte) des Read-Ahead-Puffers im Bereich von 1 bis 999 999.

Die folgenden Sonderwerte sind vorhanden:

-1

Der Client bestimmt den entsprechenden Wert.

0

Vorauslesen ist für den Client inaktiviert.

PurgeTime=Ganzzahl|_600

Intervall (in Sekunden), nach dem Nachrichten, die im Puffer für die Lesepuffer (Read-ahead-Puffer) bleiben, gelöscht

Wenn die Clientanwendung Nachrichten auf der Basis von MsgId oder CorrelId auswählt, ist es möglich, dass der Read-Ahead-Puffer Nachrichten enthält, die mit einer zuvor angeforderten MsgId oder CorrelId an den Client gesendet wurden. Diese Nachrichten würden dann in den Read-Ahead-Puffer gestellt, bis ein MQGET-Aufruf mit einer entsprechenden MsgId oder CorrelId ausgegeben wird. Sie können Nachrichten aus dem Read-Ahead-Puffer löschen, indem Sie PurgeTime festlegen. Alle Nachrichten, die länger als das Bereinigungsintervall im Read-Ahead-Puffer geblieben sind, werden automatisch gelöscht. Diese Nachrichten wurden bereits aus der Warteschlange auf dem Warteschlangenmanager entfernt, so dass sie verloren gehen, wenn sie nicht durchsucht werden.

Der gültige Bereich liegt im Bereich von 1 bis 999 999 Sekunden oder der Sonderwert 0, d. e. es findet keine Bereinigung statt.

UpdatePercentage=Ganzzahl|_-1

Der Prozentwert für die Aktualisierung im Bereich von 1-100, der bei der Berechnung des Schwellenwerts verwendet wird, um zu ermitteln, wann eine Clientanwendung eine neue Anforderung an den Server stellt. Der Sonderwert -1 gibt an, dass der Client den entsprechenden Wert bestimmt.

Der Client sendet in regelmäßigen Abständen eine Anforderung an den Server, die angibt, wie viele Daten die Clientanwendung verbraucht hat. Eine Anforderung wird gesendet, wenn die Anzahl der

Byte (n), die vom Client über MQGET-Aufrufe abgerufen werden, den Schwellenwert T überschreitet. n wird jedes Mal, wenn eine neue Anforderung an den Server gesendet wird, auf null zurückgesetzt.

Der Schwellenwert T wird wie folgt berechnet:

$$T = \text{Upper} - \text{Lower}$$

Der obere Wert entspricht der im Attribut *MaximumSize* angegebenen Größe des Lesepuffers (in Kilobyte). Der Standardwert ist 100 Kb.

Der untere Bereich ist kleiner als der obere und wird durch das Attribut *UpdatePercentage* angegeben. Dieses Attribut ist eine Zahl im Bereich von 1 bis 100 und hat einen Standardwert von 20. Der untere wird wie folgt berechnet:

$$\text{Lower} = \text{Upper} \times \text{UpdatePercentage} / 100$$

Beispiel 1:

Die Attribute *MaximumSize* und *UpdatePercentage* nehmen die Standardwerte von 100 Kb und 20 Kb an.

Der Client ruft MQGET auf, um eine Nachricht abzurufen, und wiederholt dies wiederholt. Dies wird so lange fortgesetzt, bis MQGET n Byte belegt hat.

Berechnung verwenden

$$T = \text{Upper} - \text{Lower}$$

T ist $(100-20) = 80$ Kb.

Wenn MQGET-Aufrufe also 80 Kb aus einer Warteschlange entfernt haben, stellt der Client automatisch eine neue Anforderung aus.

Beispiel 2:

Die *MaximumSize*-Attribute nimmt ihren Standardwert 100 Kb und der Wert 40 für 'UpdatePercentage' (*UpdatePercentage*) ausgewählt.

Der Client ruft MQGET auf, um eine Nachricht abzurufen, und wiederholt dies wiederholt. Dies wird so lange fortgesetzt, bis MQGET n Byte belegt hat.

Berechnung verwenden

$$T = \text{Upper} - \text{Lower}$$

T ist $(100-40) = 60$ Kb

Wenn MQGET-Aufrufe also 60 Kb aus einer Warteschlange entfernt haben, stellt der Client automatisch eine neue Anforderung ab. Dies ist früher als in BEISPIEL 1, wo die Standardwerte verwendet wurden.

Die Auswahl eines größeren Schwellenwerts T neigt daher dazu, die Häufigkeit zu verringern, mit der Anforderungen vom Client an den Server gesendet werden. Umgekehrt erhöht die Auswahl eines kleineren Schwellenwerts T die Häufigkeit von Anforderungen, die vom Client an den Server gesendet werden.

Die Auswahl eines großen Schwellenwerts T kann jedoch bedeuten, dass die Leistungssteigerung von Read-Ahead-Wert reduziert wird, da die Wahrscheinlichkeit, dass der Read-Ahead-Puffer leer wird, erhöht werden kann. Wenn dies geschieht, muss ein MQGET-Aufruf möglicherweise anhalten und darauf warten, dass Daten vom Server eintreffen.

SSL-Zeilengruppe der Clientkonfigurationsdatei

Geben Sie über die SSL-Zeilengruppe Informationen über die Verwendung von SSL oder TLS an.

Die folgenden Attribute können in die SSL-Zeilengruppe eingeschlossen werden:

CDPCheckExtensions=YES|NO

CDPCheckExtensions gibt an, ob SSL- oder TLS-Kanäle auf diesem Warteschlangenmanager versuchen, CDP-Server zu überprüfen, die in den Zertifikatserweiterungen von CrlDistributionPoint aufgeführt sind.

Für dieses Attribut sind die folgenden Werte möglich:

- YES: SSL- oder TLS-Kanäle versuchen, die CDP-Server zu überprüfen, um festzustellen, ob ein digitales Zertifikat widerrufen wird.
- NO: SSL- oder TLS-Kanäle versuchen nicht, CDP-Server zu überprüfen. Dieser Wert stellt den Standardwert dar.

CertificateLabel = Zeichenfolge

Die Zertifikatsbezeichnung der Kanaldefinition.

Dieses Attribut kann von C- und nicht verwalteten .NET-Clients gelesen werden.

CertificateValPolicy=Zeichenfolge

Bestimmt den Typ der verwendeten Zertifikatsprüfung.

ANY

Verwenden Sie eine beliebige Zertifikatvalidierungsrichtlinie, die von der zugrunde liegenden Secure Sockets Library unterstützt wird. Dies ist die Standardeinstellung.

RFC5280

Verwenden Sie nur die Zertifikatsprüfung, die mit dem Standard RFC 5280 kompatibel ist.

ClientRevocationChecks = REQUIRED | OPTIONAL | DISABLED


Legt fest, wie die Überprüfung der Zertifikatswiderrufung konfiguriert wird, wenn der Clientverbindungsauftrag einen SSL/TLS-Kanal verwendet. Siehe auch **OCSPAuthentication**.

Dieses Attribut kann von C- und nicht verwalteten .NET-Clients gelesen werden.

Für dieses Attribut sind die folgenden Werte möglich:

ERFORDERLICH (Standardwert)

Es wird versucht, die Zertifikatswiderrufkonfiguration aus der CCDT zu laden und die Widerrufsprüfung wie konfiguriert auszuführen. Wenn die CCDT-Datei nicht geöffnet werden kann oder es nicht möglich ist, das Zertifikat zu prüfen (z. B. weil ein OCSP- oder CRL-Server nicht verfügbar ist), schlägt der MQCONN-Aufruf fehl. Es wird keine Widerrufsprüfung durchgeführt, wenn die CCDT keine Widerrufskonfiguration enthält, dies jedoch nicht dazu führt, dass der Kanal fehlschlägt.

 Auf Windows-Systemen können Sie auch Active Directory für die CRL-Widerrufsprüfung verwenden. Sie können Active Directory nicht für die OCSP-Widerrufsprüfung verwenden.

OPTIONAL

Wie für REQUIRED, aber wenn es nicht möglich ist, die Zertifikatswiderrufkonfiguration zu laden, schlägt der Kanal nicht fehl.

INAKTIVIERT

Es wird kein Versuch unternommen, die Zertifikatswiderrufkonfiguration aus der CCDT zu laden, und es wird keine Überprüfung der Zertifikatswiderrufprüfung durchgeführt.

Anmerkung: Wenn Sie MQCONNX anstelle von MQCONN-Aufrufen verwenden, können Sie die Authentifizierungsinformationen (MQAIR) über das MQSCO angeben. Das Standardverhalten mit MQCONNX ist daher nicht zu scheitern, wenn die CCDT-Datei nicht geöffnet werden kann, sondern angenommen wird, dass Sie eine MQAIR-Datei bereitstellen (auch wenn Sie dies nicht tun).

EncryptionPolicySuiteB=Zeichenfolge

Legt fest, ob ein Kanal eine Suite-B-kompatible Verschlüsselung verwendet und welcher Grad der Stärke verwendet werden soll. Folgende Werte sind möglich:

KEINE

Die Suite-B-kompatible Verschlüsselung wird nicht verwendet. Dies ist die Standardeinstellung.

128_BIT,192_BIT

Setzt die Sicherheitsstärke sowohl auf 128-Bit-als auch auf 192-Bit-Stufen.

128_BIT

Setzt die Sicherheitsstärke auf 128-Bit-Ebene.

192_BIT

Setzt die Sicherheitsstärke auf 192-Bit-Ebene.

OCSPAuthentication=OPTIONAL| REQUIRED|WARN

Definiert das Verhalten von WebSphere MQ, wenn OCSP aktiviert ist und die OCSP-Widerrufsprüfung den Zertifikatswiderrufsstatus nicht ermitteln kann. Drei Werte sind möglich:

OPTIONAL

Jedes Zertifikat mit einem Widerrufsstatus, das durch die OCSP-Prüfung nicht festgestellt werden kann, wird akzeptiert, und es wird keine Warnung oder Fehlernachricht generiert. Die SSL- oder TLS-Verbindung wird so fortgesetzt, als wäre keine Widerrufsüberprüfung durchgeführt worden.

ERFORDERLICH

Die OCSP-Prüfung muss für jedes überprüfte SSL- oder TLS-Zertifikat zu einem endgültigen Widerrufsergebnis führen. Jedes SSL- oder TLS-Zertifikat mit einem Widerrufsstatus, der nicht geprüft werden kann, wird mit einer Fehlernachricht zurückgewiesen. Wenn WS-Manager-SSL-Ereignisnachrichten aktiviert sind, wird eine Nachricht MQRC_CHANNEL_SSL_ERROR mit einem 'ReasonQualifier' von MQRC_SSL_HANDSHAKE_ERROR generiert. Die Verbindung wurde aufgehoben.

Dies ist der Standardwert.

WARN

Eine Warnung wird in den Fehlerprotokollen des Warteschlangenmanagers angezeigt, wenn eine OCSP-Widerrufsprüfung nicht in der Lage ist, den Widerrufstatus eines SSL- oder TLS-Zertifikats zu ermitteln. Wenn WS-Manager-SSL-Ereignisnachrichten aktiviert sind, wird eine Nachricht MQRC_CHANNEL_SSL_WARNING mit einem 'ReasonQualifier' von MQRC_SSL_UNKNOWN_REVOCATION generiert. Die Verbindung kann fortgesetzt werden.

OCSPCheckExtensions=YES|NO

Steuert, ob WebSphere MQ auf AuthorityInfoAccess-Zertifikatserweiterungen reagiert. Wenn der Wert auf NO gesetzt ist, ignoriert WebSphere MQ AuthorityInfoAccess-Zertifikatserweiterungen und versucht keine OCSP-Sicherheitsprüfung. Der Standardwert ist YES.

SSLCryptoHardware=Zeichenfolge

Legt die Parameterzeichenfolge fest, die erforderlich ist, um die auf dem System vorhandene Verschlüsselungshardware PKCS #11 zu konfigurieren.

Geben Sie eine Zeichenfolge im folgenden Format an: *GSK_PKCS11=Treiberpfad und -dateiname; Tokenbezeichnung; Tokenkennwort; symmetrische Verschlüsselungseinstellung;*

Beispiel: `GSK_PKCS11=/usr/lib/pkcs11/PKCS11_API.so;tokenlabel;passw0rd;SYMMETRIC_CIPHER_ON`

Der Treiberpfad ist ein absoluter Pfad zur gemeinsam genutzten Bibliothek, die Unterstützung für die PKCS #11-Karte bietet. Der Name der Treiberdatei ist der Name der gemeinsam genutzten Bibliothek. Ein Beispiel für den Wert, der für den PKCS #11-Treiberpfad und den Dateinamen erforderlich ist, ist `/usr/lib/pkcs11/PKCS11_API.so`. Um auf symmetrische Chiffrieroperationen über GSKit zuzugreifen, geben Sie den Parameter für die symmetrische Verschlüsselungseinstellung an. Der Wert dieses Parameters lautet entweder:

SYMMETRIC_CIPHER_OFF

Es werden keine symmetrischen Verschlüsselungsoperationen aufgerufen. Dies ist die Standardeinstellung.

SYMMETRIC_CIPHER_ON

Zugriff auf Operationen zur symmetrischen Verschlüsselung.

Die maximal zulässige Länge beträgt 256 Zeichen. Standardmäßig erfolgt keine Angabe. Wenn Sie eine Zeichenfolge angeben, die sich nicht im richtigen Format befindet, wird ein Fehler generiert.

SSLFipsRequired=YES|NO

Gibt an, ob nur FIPS-zertifizierte Algorithmen verwendet werden sollen, wenn die Verschlüsselung in WebSphere MQ ausgeführt wird. Wenn eine Verschlüsselungshardware konfiguriert ist, werden die vom

Hardwareprodukt bereitgestellten Verschlüsselungsmodul verwendet. Diese können-je nach dem verwendeten Hardwareprodukt-möglicherweise FIPS-zertifiziert sein, die auf eine bestimmte Ebene zertifiziert sind.

SSLHTTPProxyName=Zeichenfolge

Die Zeichenfolge ist entweder der Hostname oder die Netzadresse des HTTP-Proxy-Servers, der von GSKit für OCSP-Prüfungen verwendet werden soll. Auf die Adresse kann optional eine in Klammern gesetzte Portnummer folgen. Wenn Sie die Portnummer nicht angeben, wird der Standard-HTTP-Port 80 verwendet. Auf den Plattformen HP-UX PA-RISC und Sun Solaris SPARC und für 32-Bit-Clients unter AIX kann die Netzadresse nur eine IPv4 -Adresse sein; auf anderen Plattformen kann sie eine IPv4 -oder IPv6 -Adresse sein.

Dieses Attribut kann erforderlich sein, wenn z. B. eine Firewall den Zugriff auf die URL des OCSP-Responder verhindert.

SSLKeyRepository=Pfadname

Die Position des Schlüsselrepositorys, in dem das digitale Zertifikat des Benutzers gespeichert ist, im Stammformat. Das heißt, er enthält den vollständigen Pfad und den Dateinamen ohne Erweiterung.

SSLKeyResetCount=Ganzzahl| 0

Die Anzahl der nicht verschlüsselten Byte, die in einem SSL-oder TLS-Kanal gesendet und empfangen wurden, bevor der geheime Schlüssel neu verhandelt wird.

Der Wert muss im Bereich von 0 bis 999999999 liegen.

Der Standardwert ist 0, was bedeutet, dass geheime Schlüssel nie neu verhandelt werden.

Wenn Sie einen Wert zwischen 1 und 32768 angeben, verwenden SSL- oder TLS-Kanäle 32768 (32KB) als Zählerstand für das Zurücksetzen des geheimen Schlüssels. Dadurch werden überhöhte Schlüsselresets vermieden, die bei kleinen Rücksetzwerten für geheime Schlüssel auftreten würden.

TCP-Zeilengruppe der Clientkonfigurationsdatei

Verwenden Sie die TCP-Zeilengruppe, um die Konfigurationsparameter des TCP-Netzprotokolls anzugeben.

Die folgenden Attribute können in die TCP-Zeilengruppe aufgenommen werden:

ClntRcvBuffSize=Zahl| 32768

Die Größe des TCP/IP-Empfangspuffers (in Byte), der vom Clientende eines Clientverbindungs-Serververbindungskanal verwendet wird. Der Wert null gibt an, dass das Betriebssystem die Puffergrößen verwalten wird, im Gegensatz zu den Puffergrößen, die von WebSphere MQ festgelegt werden.

ClntSndBuffSize=Zahl| 32768

Die Größe des TCP/IP-Sendepuffers (in Byte), der vom Clientende eines Clientverbindungs-Serververbindungskanal verwendet wird. Der Wert null gibt an, dass das Betriebssystem die Puffergrößen verwalten wird, im Gegensatz zu den Puffergrößen, die von WebSphere MQ festgelegt werden.

Connect_Timeout=Anzahl

Die Anzahl der Sekunden, bevor ein Versuch, eine Verbindung zum Socket herzustellen, das Zeitlimit überschreitet. Der Standardwert ist 0, sofern der Kanal nicht mit einer Clientkanalgewichtung ungleich null konfiguriert wurde. In diesem Fall ist der Standardwert 5.

IPAddressVersion=MQIPADDR_IPV4|MQIPADDR_IPV6

Gibt das IP-Protokoll an, das für eine Kanalverbindung verwendet werden soll.

Sie verfügt über die möglichen Zeichenfolgewerte von MQIPADDR_IPV4 oder MQIPADDR_IPV6. Diese Werte haben dieselbe Bedeutung wie IPV4 und IPV6 in **ALTER QMGR IPADDRV**.

KeepAlive=YES|NO

Schalten Sie die KeepAlive-Funktion ein oder aus. KeepAlive=YES bewirkt, dass TCP/IP in regelmäßigen Abständen überprüft, ob das andere Ende der Verbindung noch verfügbar ist. Ist dies nicht der Fall, wird der Kanal geschlossen.

Windows Library1=DL-Name| WSOCK32

(Nur Windows) Der Name der TCP/IP-Sockets-DLL.

Zeilengruppen 'TMF' und 'TMF/Gateway '

Das von IBM WebSphere MQ bereitgestellte TMF/Gateway wird in einer Pathway-Umgebung ausgeführt. Verwenden Sie die TMF- und TMF/Gateway-Zeilengruppen, um die erforderlichen Konfigurationsparameter für den IBM WebSphere MQ-Client für HP Integrity NonStop Server für die Kommunikation mit dem TMF/Gateway anzugeben.

Wenn Sie TMF verwenden wollen, müssen Sie eine TMF-Zeilengruppe und eine TmfGateway-Zeilengruppe für jeden WS-Manager definieren, mit dem Sie kommunizieren. Alle Werte werden aus Ihrer Konfiguration abgeleitet.

TMF-Zeilengruppe

PathMon=Name

Der Name des definierten Pathmon-Prozesses, der die Serverklassen für das TMF/Gateway definiert.

TmfGateway, Zeilengruppe

Die folgenden Attribute können in diese Zeilengruppe aufgenommen werden:

QManager=Name

Der Name des Warteschlangenmanagers.

Server=Name

Der Name der Serverklasse für den TMF/Gateway, der für diesen Warteschlangenmanager konfiguriert ist.

Beispiel

Das folgende Beispiel zeigt eine TMF-Zeilengruppe, die mit zwei TmfGateway-Zeilengruppen für zwei verschiedene WS-Manager auf verschiedenen Servern definiert ist:

```
TMF :
  PathMon=$PSD1P

TmfGateway:
  QManager=MQ5B
  Server=MQ-MQ5B

TmfGateway:
  QManager=MQ5C
  Server=MQ-MQ5C
```

WebSphere MQ -Umgebungsvariablen verwenden

Dieser Abschnitt beschreibt die Umgebungsvariablen, die Sie mit WebSphere MQ MQI-Clientanwendungen verwenden können.

Sie können Umgebungsvariablen auf die folgenden Arten verwenden:

- Über die Variablen im Systemprofil können Sie permanente Änderungen vornehmen.
- Soll eine Änderung lediglich für die betreffende Sitzung vorgenommen werden, führen Sie einen Befehl in der Befehlszeile aus.
- Um einem oder mehreren Variablen einen bestimmten Wert in Abhängigkeit von der ausgeführten Anwendung zu geben, fügen Sie Befehle zu einer Befehlsscriptdatei hinzu, die von der Anwendung verwendet wird.

WebSphere MQ verwendet Standardwerte für die nicht festgelegten Variablen.

Sofern nicht anders angegeben, sind Befehle auf allen WebSphere MQ MQI-Clientplattformen verfügbar.

Für jede Umgebungsvariable kann mit dem plattformspezifischen Befehl die aktuelle Einstellung angezeigt oder der Wert einer Variablen zurückgesetzt werden.

Beispiel:

Wert einer Umgebungsvariablen festlegen oder zurücksetzen		
Aktion	Befehl	
	Windows	UNIX and Linux-Systeme
Entfernt die Variable	SET MQSERVER=	unset MQSERVER
Zeigt die aktuelle Einstellung an.	SET MQSERVER	echo \$MQSERVER
Zeigt alle Umgebungsvariablen für die Sitzung an	set	set

Informationen zu den einzelnen Variablen finden Sie in folgenden Unterabschnitten:

Zugehörige Konzepte

„Client mit einer Konfigurationsdatei konfigurieren“ auf Seite 136

Konfigurieren Sie Ihre Clients mit Attributen in einer Textdatei. Diese Attribute können durch Umgebungsvariablen oder auf andere plattformspezifische Methoden überschrieben werden.

Zugehörige Verweise

[Umgebungsvariablen](#)

MQCCSID

MQCCSID gibt die Nummer des codierten Zeichensatzes an, der verwendet werden soll, und überschreibt den CCSID-Wert, mit dem der Server konfiguriert wurde.

Weitere Informationen hierzu finden Sie im Abschnitt [ID des codierten Zeichensatzes \(CCSID\) des Clients oder des Servers auswählen](#).

Verwenden Sie einen der folgenden Befehle, um diese Variable zu definieren:

- Windows:

```
SET MQCCSID=number
```

- Für UNIX and Linux -Systeme:

```
export MQCCSID=number
```

MQCERTVPOL

MQCERTVPOL gibt die Validierungsrichtlinie für Zertifikate an.

Weitere Informationen zu Zertifikatsprüfrichtlinien in WebSphere MQ finden Sie unter [Zertifikatsprüfrichtlinien in WebSphere MQ](#).

Diese Umgebungsvariable überschreibt die Einstellung *CertificateValPolicy* in der SSL-Zeilengruppe der Client-INI-Datei. Die Variable kann auf einen der folgenden Werte gesetzt werden:

ANY

Verwenden Sie eine beliebige Zertifikatvalidierungsrichtlinie, die von der zugrunde liegenden Secure Sockets Library unterstützt wird

RFC5280

Verwenden Sie nur die Zertifikatsprüfung, die mit dem Standard RFC 5280 kompatibel ist.

Verwenden Sie einen der folgenden Befehle, um diese Variable zu definieren:

- Windows:

```
SET MQCERTVPOL=value
```

- Für UNIX and Linux -Systeme:

```
export MQCERTVPOL=value
```

MQCHLLIB

MQCHLLIB gibt den Verzeichnispfad für die Datei an, die die Definitionstabelle für den Clientkanal (CCDT) enthält. Die Datei wird auf dem Server erstellt, kann aber auf die MQI-Client-Workstation WebSphere MQ kopiert werden.

Wenn MQCHLLIB nicht festgelegt ist, nimmt der Pfad für den Client standardmäßig die folgenden Werte an:

- **Windows** Unter Windows: `MQ_INSTALLATION_PATH`
- **UNIX** **Linux** Für UNIX and Linux-Systeme: `/var/mqm/`

Für die Befehle **crtmqm** und **strmqm** wird standardmäßig einer von zwei Pfadgruppen verwendet. Wenn *datapath* festgelegt ist, wird der Pfad standardmäßig auf eine der ersten festgelegten Werte gesetzt. Wenn *datapath* nicht festgelegt ist, wird der Pfad standardmäßig auf einen der zweiten Gruppe gesetzt.

- **Windows** Für Windows: `datapath\@ipcc`
- **UNIX** **Linux** Für UNIX -und Linux-Systeme: `datapath/@ipcc`

Oder:

- **Windows** Für Windows: `MQ_INSTALLATION_PATH\data\qmgrs\qmgrname\@ipcc`
- **UNIX** **Linux** Für UNIX and Linux-Systeme: `/prefix/qmgrs/qmgrname/@ipcc`

Dabei gilt:

- `MQ_INSTALLATION_PATH` steht für das übergeordnete Verzeichnis, in dem IBM WebSphere MQ installiert ist.
- Falls vorhanden, ist *datapath* der Wert von DataPath , der in der Zeilengruppe des Warteschlangenmanagers definiert ist.
- *prefix* ist der Wert von Prefix, der in der Zeilengruppe des Warteschlangenmanagers definiert ist. Das Präfix lautet normalerweise `/var/mqm` auf UNIX -und Linux -Plattformen.
- *qmgrname* ist der Wert des Directory -Attributs, das in der Zeilengruppe des Warteschlangenmanagers definiert ist. Der Wert kann sich von dem Namen des tatsächlichen Warteschlangenmanagers unterscheiden. Der Wert wurde möglicherweise geändert, um Sonderzeichen zu ersetzen.
- Die Zeilengruppe für den Warteschlangenmanager ist in der Datei `mqs.ini` unter UNIX und Linux und in der Registry unter Windows definiert

Anmerkungen:

1. Wenn diese Option festgelegt ist, überschreibt MQCHLLIB den Pfad, der zum Lokalisieren der CCDT verwendet wird.
2. Umgebungsvariablen, wie z. B. MQCHLLIB, können auf eine plattformspezifische Weise auf einen Prozess oder einen Job oder systemweit zugeordnet werden.
3. Wenn Sie MQCHLLIB systemweit auf einem Server definieren, wird derselbe Pfad zur CCDT-Datei für alle Warteschlangenmanager auf dem Server festgelegt. Wenn Sie die Umgebungsvariable MQCHLLIB nicht festlegen, ist der Pfad für jeden Warteschlangenmanager unterschiedlich. Warteschlangenmanager lesen den Wert von MQCHLLIB im Befehl **crtmqm** oder **strmqm** , wenn er festgelegt ist.
4. Wenn Sie mehrere WS-Manager auf einem einzigen Server erstellen, ist die Unterscheidung wichtig, aus folgendem Grund: Wenn Sie MQCHLLIB systemweit definieren, aktualisiert jeder Warteschlangenmanager die gleiche CCDT-Datei. Die Datei enthält die Clientverbindungsdefinitionen von allen Warteschlangenmanagern auf dem Server. Wenn die gleiche Definition auf mehreren Warteschlangenmanagern vorhanden ist, z. B. `SYSTEM.DEF.CLNTCONN` , enthält die Datei die letzte Definition. Wenn Sie

einen Warteschlangenmanager erstellen und MQCHLLIB festgelegt ist, wird SYSTEM.DEF.CLNTCONN in der CCDT aktualisiert. Die Aktualisierung überschreibt die von einem anderen Warteschlangenmanager erstellte SYSTEM.DEF.CLNTCONN. Wenn Sie die frühere Definition geändert haben, gehen Ihre Änderungen verloren. Aus diesem Grund müssen Sie nach Alternativen suchen, um MQCHLLIB als systemweite Umgebungsvariable auf dem Server zu definieren.

5. Die Option MQSC und PCF NOREPLACE in einer Clientverbindungsdefinition prüft den Inhalt der CCDT-Datei nicht. Eine Clientverbindungskanaldefinition mit demselben Namen, die zuvor erstellt wurde, jedoch nicht von diesem Warteschlangenmanager, wird unabhängig von der Option NOREPLACE ersetzt. Wenn die Definition zuvor von demselben WS-Manager erstellt wurde, wird die Definition nicht ersetzt.
6. Mit dem Befehl `zcpmqobj -t clchl1tab` wird die CCDT-Datei gelöscht und erneut erstellt. Die Datei wird nur mit den Clientverbindungsdefinitionen erneut erstellt, die auf dem Warteschlangenmanager erstellt wurden, für den der Befehl ausgeführt wird.
7. Andere Befehle, die die CCDT aktualisieren, ändern nur die Clientverbindungskanäle, die denselben Kanalnamen haben. Andere Clientverbindungskanäle in der Datei werden nicht geändert.
8. Der Pfad für MQCHLLIB benötigt keine Anführungszeichen.

Beispiele

Verwenden Sie einen der folgenden Befehle, um diese Variable zu definieren:

-  Windows:

```
SET MQCHLLIB=pathname
```

Beispiel:

```
SET MQCHLLIB=C:\wmqtest
```

-   Für UNIX and Linux -Systeme:

```
export MQCHLLIB=pathname
```

MQCHLTAB

MQCHLTAB gibt den Namen der Datei an, die die Clientkanaldefinitionstabelle (ccdt) enthält. Der Standarddateiname ist AMQCLCHL.TAB.

Informationen darüber, wo sich die Clientkanaldefinitionstabelle auf einem Server befindet, finden Sie in „Definitionstabelle für den Clientkanal“ auf Seite 125.

Verwenden Sie einen der folgenden Befehle, um diese Variable zu definieren:

- Unter Windows:

```
SET MQCHLTAB=filename
```

- Auf UNIX and Linux-Systemen:

```
export MQCHLTAB=filename
```

Beispiel:

```
SET MQCHLTAB=ccdf1.tab
```

In der gleichen Weise wie für den Client gibt die Umgebungsvariable MQCHLTAB auf dem Server den Namen der Definitionstabelle für den Clientkanal an.

MQIPADDRV

MQIPADDRV gibt an, welches IP-Protokoll für eine Kanalverbindung verwendet werden soll. Die möglichen Zeichenfolgenwerte sind "MQIPADDR_IPV4" oder "MQIPADDR_IPV6". Diese Werte haben dieselbe Bedeutung wie "IPV4" und "IPV6" in ALTER QMGR IPADDRV. Wenn diese Umgebungsvariable nicht gesetzt ist, wird der Wert "MQIPADDR_IPV4" angenommen.

Verwenden Sie einen der folgenden Befehle, um diese Variable zu definieren:

- Für Windows:

```
SET MQIPADDRV=MQIPADDR_IPV4|MQIPADDR_IPV6
```

- Für UNIX and Linux -Systeme:

```
export MQIPADDRV=MQIPADDR_IPV4|MQIPADDR_IPV6
```

MQNAME

MQNAME gibt den lokalen NetBIOS -Namen an, den die WebSphere MQ -Prozesse verwenden können.

Eine vollständige Beschreibung und die Regeln der Vorrangstellung auf dem Client und dem Server finden Sie in „[NetBIOS -Verbindung unter Windows definieren](#)“ auf Seite 94 .

Verwenden Sie den folgenden Befehl, um diese Variable zu definieren:

```
SET MQNAME=Your_env_Name
```

Beispiel:

```
SET MQNAME=CLIENT1
```

NetBIOS erfordert auf einigen Plattformen für jede Anwendung einen anderen Namen (festgelegt durch MQNAME), wenn mehrere WebSphere MQ -Anwendungen gleichzeitig auf dem WebSphere MQ MQI-Client ausgeführt werden.

MQSERVER

Mit der Umgebungsvariablen "MQSERVER" wird der Minimalkanal definiert. MQSERVER gibt die Position des WebSphere MQ -Servers und die zu verwendende Übertragungsmethode an.

Sie können MQSERVER nicht verwenden, um einen SSL-Kanal oder einen Kanal mit Kanalexits zu definieren. Ausführliche Informationen zum Definieren eines SSL-Kanals finden Sie im Abschnitt [Kanäle mit SSL schützen](#).

ConnectionName muss ein vollständig qualifizierter Netzname sein. Das *ChannelName* kann den Schrägstrich (/) nicht enthalten, da dieses Zeichen zum Trennen des Kanalnamens, des Transporttyps und des Verbindungsnamens verwendet wird. Wenn die Umgebungsvariable MQSERVER zum Definieren eines Clientkanals verwendet wird, wird eine maximale Nachrichtenlänge (MAXMSGL) von 100 MB verwendet. Daher ist die maximale Nachrichtengröße, die für den Kanal wirksam ist, der im SVRCONN-Kanal auf dem Server angegebene Wert.

Verwenden Sie einen der folgenden Befehle, um diese Variable zu definieren:

- Windows:

```
SET MQSERVER=ChannelName/TransportType/ConnectionName
```

- Für UNIX and Linux -Systeme:

```
export MQSERVER='ChannelName/TransportType/ConnectionName'
```


TransportType kann abhängig von Ihrer IBM WebSphere MQ-Clientplattform einen der folgenden Werte haben:

- LU62
- TCP
- NETBIOS
- SPX

ConnectionName kann eine durch Kommas getrennte Liste mit Verbindungsnamen sein. Die Verbindungsnamen in der Liste werden auf ähnliche Weise für mehrere Verbindungen in einer Clientverbindungstabelle verwendet. Die Liste *ConnectionName* kann als Alternative zu WS-Manager-Gruppen verwendet werden, um mehrere Verbindungen für den Client anzugeben, die versucht werden sollen. Wenn Sie einen Warteschlangenmanager mit mehreren Instanzen konfigurieren, können Sie unter Umständen eine *ConnectionName* -Liste verwenden, um verschiedene Warteschlangenmanagerinstanzen anzugeben.

TCP/IP-Standardport

Bei TCP/IP geht WebSphere MQ standardmäßig davon aus, dass der Kanal mit Port 1414 verbunden wird.

Dies kann wie folgt geändert werden:

- Die Anschlussnummer in Klammern als letzter Teil des Verbindungsnamens hinzufügen:
 - Für Windows:

```
SET MQSERVER=ChannelName/TransportType/ConnectionName(PortNumber)
```

- Für UNIX and Linux -Systeme:

```
export MQSERVER='ChannelName/TransportType/ConnectionName(PortNumber)'
```

- Ändern Sie die `mqclient.ini` -Datei, indem Sie die Portnummer zum Protokollnamen hinzufügen, z. B.:

```
TCP:  
port=2001
```

- WebSphere MQ zur Servicedatei hinzufügen, wie im Abschnitt „TCP/IP-Listener verwenden“ auf Seite [101](#) beschrieben.

SPX-Standardsocket

Bei SPX geht WebSphere MQ standardmäßig davon aus, dass der Kanal mit dem Socket 5E86 verbunden wird.

Dies kann wie folgt geändert werden:

- Hinzufügen der Socketnummer in eckigen Klammern als letzter Teil des Verbindungsnamens:

```
SET MQSERVER=ChannelName/TransportType/ConnectionName(SocketNumber)
```

Geben Sie bei SPX-Verbindungen den Verbindungsnamen und den Socket in der Form `network.node(socket)` an. Wenn sich der WebSphere MQ -Client und -Server in demselben Netz befinden, muss das Netz nicht angegeben werden. Wenn Sie den Standardsocket verwenden, muss der Socket nicht angegeben werden.

- Ändern Sie die `qm.ini` -Datei, indem Sie die Portnummer zum Protokollnamen hinzufügen, z. B.:

```
SPX:  
socket=5E87
```

Verwenden von MQSERVER

Wenn Sie die Umgebungsvariable MQSERVER verwenden, um den Kanal zwischen Ihrer WebSphere MQ MQI-Clientmaschine und einer Servermaschine zu definieren, ist dies der einzige Kanal, der Ihrer Anwendung zur Verfügung steht, und es wird nicht auf die Definitionstabelle für Clientkanäle (CCDT) verwiesen.

In dieser Situation bestimmt das Empfangsprogramm, das Sie auf der Servermaschine ausgeführt haben, den Warteschlangenmanager, zu dem Ihre Anwendung eine Verbindung herstellt. Es handelt sich um denselben Warteschlangenmanager, mit dem das Empfangsprogramm verbunden ist.

Wenn die MQCONN-oder MQCONNX-Anforderung einen anderen Warteschlangenmanager als den Listener angibt, mit dem das Empfangsprogramm verbunden ist, oder wenn der MQSERVER-Parameter *TransportType* nicht erkannt wird, schlägt die MQCONN-oder MQCONNX-Anforderung mit dem Rückkehrcode MQRC_Q_MGR_NAME_ERROR fehl.

Auf UNIX and Linux-Systemen könnten Sie MQSERVER wie in einem der folgenden Beispiele definieren:

```
export MQSERVER=CHANNEL1/TCP/'9.20.4.56(2002)'  
export MQSERVER=CHANNEL1/LU62/BOX99
```

Alle MQCONN -oder MQCONNX -Anforderungen versuchen dann, den Kanal zu verwenden, den Sie definiert haben, es sei denn, von der an MQCONNXbereitgestellten MQCNO-Struktur wurde auf eine MQCD-Struktur verwiesen. In diesem Fall hat der durch die MQCD-Struktur angegebene Kanal Vorrang vor allen durch die Umgebungsvariable MQSERVER angegebenen.

Die MQSERVER-Umgebungsvariable hat Vorrang vor jeder Clientkanaldefinition, auf die durch MQCHLLIB und MQCHLTAB verwiesen wird.

MQSERVER wird abgebrochen

Geben Sie Folgendes ein, um MQSERVER abzubrechen und zur Clientkanaldefinitionstabelle zurückzukehren, auf die durch MQCHLLIB und MQCHLTAB verwiesen wird:

- Unter Windows:

```
SET MQSERVER=
```

- Auf UNIX and Linux-Systemen:

```
unset MQSERVER
```

MQSSLCRYP

MQSSLCRYP enthält eine Parameterzeichenfolge, die es Ihnen ermöglicht, die auf dem System vorhandene Verschlüsselungshardware zu konfigurieren. Die gültigen Werte entsprechen den Werten für den Parameter SSLCRYP des Befehls ALTER QMGR.

Verwenden Sie einen der folgenden Befehle, um diese Variable zu definieren:

- Auf Windows -Systemen:

```
SET MQSSLCRYP=string
```

- Auf UNIX and Linux-Systemen:

```
export MQSSLCRYP=string
```

Zugehörige Verweise

Befehl **ALTER QMGR**, Parameter **SSLCRYP**

MQSSLFIPS

MQSSLFIPS gibt an, ob bei der Verschlüsselung in WebSphere MQ nur FIPS-zertifizierte Algorithmen verwendet werden sollen. Die Werte sind mit dem Wert für den Parameter SSLFIPS des Befehls ALTER QMGR identisch.

Die Verwendung von FIPS-zertifizierten Algorithmen ist von der Verwendung von Verschlüsselungshardware betroffen. Weitere Informationen finden Sie im Abschnitt [Angeben, dass nur FIPS-zertifizierte CipherSpecs zur Laufzeit auf dem MQI-Client verwendet werden](#).

Verwenden Sie einen der folgenden Befehle, um diese Variable zu definieren:

- Auf Windows -Systemen:

```
SET MQSSLFIPS=YES|NO
```

- Auf UNIX and Linux-Systemen:

```
export MQSSLFIPS=YES|NO
```

Der Standardwert ist NO.

MQSSLKEYR

MQSSLKEYR gibt die Position des Schlüsselrepositorys an, in dem das digitale Zertifikat gespeichert ist, das dem Benutzer gehört, im Stammformat. Das Stammformat beinhaltet den vollständigen Pfad und den Dateinamen ohne Erweiterung. Ausführliche Informationen finden Sie im Abschnitt zum Parameter SSLKEYR des Befehls ALTER QMGR.

Verwenden Sie einen der folgenden Befehle, um diese Variable zu definieren:

- Auf Windows -Systemen:

```
SET MQSSLKEYR=pathname
```

- Auf UNIX and Linux-Systemen:

```
export MQSSLKEYR=pathname
```

Es gibt keinen Standardwert.

MQSSLPROXY

MQSSLPROXY gibt den Hostnamen und die Portnummer des HTTP-Proxy-Servers an, der von GSKit für OCSP-Prüfungen verwendet werden soll.

Verwenden Sie einen der folgenden Befehle, um diese Variable zu definieren:

- Auf Windows -Systemen:

```
SET MQSSLPROXY=string
```

- Auf UNIX and Linux-Systemen:

```
export MQSSLPROXY="string"
```

Die Zeichenfolge ist entweder der Hostname oder die Netzadresse des HTTP-Proxy-Servers, der von GSKit für OCSP-Prüfungen verwendet werden soll. Auf die Adresse kann optional eine in Klammern gesetzte Portnummer folgen. Wenn Sie die Portnummer nicht angeben, wird der Standard-HTTP-Port 80 verwendet.

Auf UNIX and Linux -Systemen können Sie z. B. einen der folgenden Befehle verwenden:

- ```
export MQSSLPROXY="proxy.example.com(80)"
```

- `export MQSSLPROXY="127.0.0.1"`

## MQSSLRESET

MQSSLRESET stellt die Anzahl der unverschlüsselten Byte dar, die auf einem SSL- oder TLS-Kanal gesendet und empfangen werden, bevor der geheime Schlüssel neu verhandelt wird.

Weitere Informationen zur Neuvereinbarung von geheimen Schlüsseln finden Sie im Abschnitt [Geheime SSL- und TLS-Schlüssel zurücksetzen](#).

Er kann auf eine ganze Zahl im Bereich von 0 bis 999 999 999 gesetzt werden. Der Standardwert ist 0, was bedeutet, dass geheime Schlüssel nie neu verhandelt werden. Wenn Sie einen SSL- oder TLS-Rücksetzungszähler für geheime Schlüssel im Bereich von 1 Byte bis 32 KB angeben, verwenden SSL- oder TLS-Kanäle einen Rücksetzungszähler für geheime Schlüssel von 32 KB. Dieser Rücksetzungszähler für geheime Schlüssel soll übermäßige Schlüsselrücksetzungen vermeiden, die bei kleinen Rücksetzungswerten für geheime SSL- oder TLS-Schlüssel auftreten würden.

Verwenden Sie einen der folgenden Befehle, um diese Variable zu definieren:

- Unter Windows:

```
SET MQSSLRESET=integer
```

- Unter UNIX and Linux:

```
export MQSSLRESET=integer
```

## Eingereihtes Publish/Subscribe steuern

Sie können den Status des eingereichten Publish/Subscribe starten, stoppen und anzeigen. Darüber hinaus können Sie Datenströme hinzufügen und entfernen sowie Warteschlangenmanager aus einer Brokerhierarchie hinzufügen und löschen.

Weitere Informationen zum Steuern von eingereichtem Publish/Subscribe finden Sie in den folgenden Unterabschnitten:

## Publish/Subscribe-Nachrichtenattribute in der Warteschlange festlegen

Sie steuern das Verhalten einiger Publish/Subscribe-Nachrichtenattribute mit Hilfe von WS-Managerattributen. Die anderen Attribute, die Sie in der Zeilengruppe *Broker* der Datei *qm.ini* steuern.

### Informationen zu diesem Vorgang

Sie können die folgenden Publish/Subscribe-Attribute festlegen: Details hierzu finden Sie unter [Parameter des Warteschlangenmanagers](#).

| <i>Tabelle 23. Publish/Subscribe-Konfigurationsparameter</i>   |                    |
|----------------------------------------------------------------|--------------------|
| Beschreibung                                                   | MQSC-Parametername |
| Wiederholungszähler für Befehlsnachricht                       | <b>PSRTCNT</b>     |
| Unzustellbare Befehlseingabennachricht verwerfen               | <b>PSNPMMSG</b>    |
| Verhalten nach unzustellbarer Befehlsantwortnachricht          | <b>PSNPRES</b>     |
| Verarbeiten von Befehlsnachrichten unter Synchronisationspunkt | <b>PSSYNCP</b>     |

Die Zeilengruppe 'Broker' wird verwendet, um die folgenden Konfigurationseinstellungen zu verwalten:

- `PersistentPublishRetry = yes | force`

Wenn Sie `Ja` angeben, schlägt eine Veröffentlichung einer persistenten Nachricht über die Publish/Subscribe-Schnittstelle in der Warteschlange fehl, und es wurde keine negative Antwort angefordert, wird die Veröffentlichungsoperation erneut versucht.

Wenn Sie eine negative Antwortnachricht angefordert haben, wird die negative Antwort gesendet, und es findet keine weitere Wiederholung statt.

Wenn Sie `Erzwingen` angeben, wird die Veröffentlichungsoperation, wenn eine Veröffentlichung einer persistenten Nachricht über die Publish/Subscribe-Schnittstelle in der Warteschlange fehlschlägt, erneut versucht, bis sie erfolgreich verarbeitet wurde. Es wird keine negative Antwort gesendet.

- `NonPersistentPublishRetry = yes | force`

Wenn Sie `Ja` angeben, schlägt eine Veröffentlichung einer nicht persistenten Nachricht über die Publish/Subscribe-Schnittstelle in der Warteschlange fehl, und es wurde keine negative Antwort angefordert, wird die Veröffentlichungsoperation erneut versucht.

Wenn Sie eine negative Antwortnachricht angefordert haben, wird die negative Antwort gesendet, und es findet keine weitere Wiederholung statt.

Wenn Sie `Erzwingen` angegeben haben und eine Veröffentlichung einer nicht persistenten Nachricht über die Publish/Subscribe-Schnittstelle in der Warteschlange fehlschlägt, wird die Veröffentlichungsoperation so lange erneut versucht, bis sie erfolgreich verarbeitet wurde. Es wird keine negative Antwort gesendet.

**Anmerkung:** Wenn Sie diese Funktionalität für nicht persistente Nachrichten aktivieren möchten, müssen Sie außerdem den Wert für `NonPersistentPublishRetry` festlegen, um sicherzustellen, dass das WS-Managerattribut **PSSYNCP**T auf `Yes` gesetzt ist.

Dies kann sich auch auf die Leistung der Verarbeitung nicht persistenter Veröffentlichungen auswirken, da die **MQGET** aus der STREAM-Warteschlange jetzt unter einem Synchronisationspunkt auftritt.

- `PublishBatchSize=Anzahl`

Der Broker verarbeitet normalerweise Veröffentlichungsnachrichten innerhalb eines Synchronisationspunkts. Es kann ineffizient sein, jede Veröffentlichung einzeln festzuschreiben, und unter bestimmten Umständen kann der Broker mehrere Publizierungsnachrichten in einer einzigen UO- Einheit verarbeiten. Dieser Parameter gibt die maximale Anzahl der Publizierungsnachrichten an, die in einer einzigen UO- Unit verarbeitet werden können.

Der Standardwert für `PublishBatchSize` ist 5.

- `PublishBatchInterval=Anzahl`

Der Broker verarbeitet normalerweise Veröffentlichungsnachrichten innerhalb eines Synchronisationspunkts. Es kann ineffizient sein, jede Veröffentlichung einzeln festzuschreiben, und unter bestimmten Umständen kann der Broker mehrere Publizierungsnachrichten in einer einzigen UO- Einheit verarbeiten. Dieser Parameter gibt die maximale Zeit (in Millisekunden) zwischen der ersten Nachricht in einem Stapel und einer nachfolgenden Veröffentlichung an, die in demselben Stapel enthalten ist.

Ein Stapelintervall von 0 zeigt an, dass bis zu `PublishBatchSize` -Nachrichten verarbeitet werden können, sofern die Nachrichten sofort verfügbar sind.

Der Standardwert für `PublishBatchInterval` ist null.

## Vorgehensweise

Verwenden Sie WebSphere MQ Explorer, programmierbare Befehle oder den Befehl `runmqsc`, um die Warteschlangenmanagerattribute zu ändern, die das Verhalten von Publish/Subscribe steuern.

### Beispiel

```
ALTER QMGR PSNPRES (SAFE)
```

## In Warteschlange eingereichtes Publish/Subscribe starten

## Vorbereitende Schritte

Lesen Sie die Beschreibung von [PSMODE](#) , um die drei Modi von Publish/Subscribe zu verstehen:

- COMPAT
- Inaktiviert
- Aktiviert

**Anmerkung:** Wenn Sie eine Migration von Version 6.0 durchgeführt haben, müssen Sie **strmqbrk** verwenden, um den Publish/Subscribe-Brokerstatus von Version 6.0 zu migrieren, wenn Sie mit einem aktualisierten Warteschlangenmanager arbeiten. Dies gilt nicht für z/OS.

## Informationen zu diesem Vorgang

Legen Sie das Attribut QMGR PSMODE fest, um entweder die Publish/Subscribe-Schnittstelle in der Warteschlange (auch als Broker bezeichnet) oder die Publish/Subscribe-Steuerkomponente (auch als Publish/Subscribe der Version 7 bezeichnet) zu starten. Um die Publish/Subscribe-Warteschlange in die Warteschlange zu starten, müssen Sie PSMODE auf ENABLED setzen. Der Standardwert ist ENABLED .

## Vorgehensweise

Verwenden Sie WebSphere MQ Explorer oder den Befehl **runmqsc** , um die Publish/Subscribe-Schnittstelle in der Warteschlange zu aktivieren, wenn die Schnittstelle noch nicht aktiviert ist.

### Beispiel

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

## Nächste Schritte

WebSphere MQ verarbeitet Befehle für eingereichtes Publish/Subscribe und MQI-Aufrufe (Publish/Subscribe Message Queue Interface).

# In Warteschlange eingereichtes Publish/Subscribe stoppen

## Vorbereitende Schritte

Eingereichtes Publish/Subscribe ist veraltet.

Lesen Sie die Beschreibung von [PSMODE](#) , um die drei Modi von Publish/Subscribe zu verstehen:

- COMPAT
- INAKTIVIERT
- ENABLED

## Informationen zu diesem Vorgang

Setzen Sie das Attribut QMGR PSMODE , um die Publish/Subscribe-Schnittstelle in der Warteschlange (auch als Broker bezeichnet) oder die Publish/Subscribe-Steuerkomponente (auch bekannt als Publish/Subscribe der Version 7) oder beides zu stoppen. Um die Publish/Subscribe-Warteschlange in der Warteschlange zu stoppen, müssen Sie PSMODE auf COMPAT setzen. Wenn Sie die Publish/Subscribe-Steuerkomponente vollständig stoppen möchten, setzen Sie PSMODE auf DISABLED .

## Vorgehensweise

Verwenden Sie WebSphere MQ Explorer oder den Befehl **runmqsc** , um die Publish/Subscribe-Schnittstelle in der Warteschlange zu inaktivieren.

## Beispiel

**ALTER QMGR** PSMODE (COMPAT)

## Datenstrom hinzufügen

Sie können Datenströme manuell hinzufügen, sodass sie mit Datenströmen koexistieren, die von Version 6.0 -Warteschlangenmanagern migriert wurden.

### Vorbereitende Schritte

Machen Sie sich mit der Funktionsweise von Publish/Subscribe-Datenströmen vertraut, indem Sie das Thema [Datenströme und Themenlesen](#).

### Informationen zu diesem Vorgang

Verwenden Sie den PCF-Befehl, **runmqscoder** IBM WebSphere MQ Explorer , um diese Schritte auszuführen.

**Anmerkung:** Sie können die Schritte 1 und 2 in beliebiger Reihenfolge ausführen. Führen Sie Schritt 3 nur aus, wenn die Schritte 1 und 2 beide abgeschlossen sind.

### Vorgehensweise

1. Definieren Sie eine lokale Warteschlange mit demselben Namen wie der Version 6.0 -Datenstrom.
2. Definieren Sie ein lokales Thema mit demselben Namen wie der Version 6.0 -Datenstrom.
3. Fügen Sie den Namen der Warteschlange zur Namensliste hinzu: `SYSTEM.QPUBSUB.QUEUE.NAMELIST`
4. Wiederholen Sie diese Schritte für alle Warteschlangenmanager unter Version 7.1 oder höher in der Publish/Subscribe-Hierarchie.

### Wird hinzugefügt 'Sport'

Im Beispiel für die gemeinsame Nutzung des Datenstroms 'Sport' arbeiten Version 6.0 -und Version 7.1 -Warteschlangenmanager in derselben Publish/Subscribe-Hierarchie. Die Version 6.0 -Warteschlangenmanager nutzen gemeinsam einen Datenstrom mit dem Namen 'Sport'. Das Beispiel zeigt, wie eine Warteschlange und ein Thema auf Version 7.1 -Warteschlangenmanagern mit dem Namen 'Sport' mit einer Themenzeichenfolge 'Sport' erstellt werden, die mit dem Datenstrom der Version 6 gemeinsam genutzt wird 'Sport'.

Eine Version 7.1 Veröffentlichungsanwendung, Veröffentlichung im Thema 'Sport' mit der Themenzeichenfolge 'Soccer/Results', erstellt die resultierende Themenzeichenfolge 'Sport/Soccer/Results'. Auf Version 7.1 -Warteschlangenmanagern empfangen Subskribenten des Themas 'Sport' mit der Themenzeichenfolge 'Soccer/Results' die Veröffentlichung.

Auf Version 6.0 -Warteschlangenmanagern empfangen Subskribenten des Datenstroms 'Sport' mit der Themenzeichenfolge 'Soccer/Results' die Veröffentlichung.

```
runmqsc QM1
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM1.
define qlocal('Sport')
 1 : define qlocal('Sport')
AMQ8006: WebSphere MQ queue created.
define topic('Sport') topicstr('Sport')
 2 : define topic('Sport') topicstr('Sport')
AMQ8690: WebSphere MQ topic created.
alter namelist(SYSTEM.QPUBSUB.QUEUE.NAMELIST) NAMES('Sport', 'SYSTEM.BROKER.DEFAULT.STREAM',
'SYSTEM.BROKER.ADMIN.STREAM')
 3 : alter namelist(SYSTEM.QPUBSUB.QUEUE.NAMELIST) NAMES('Sport', 'SYSTEM.BROKER.DE
FAULT.STREAM', 'SYSTEM.BROKER.ADMIN.STREAM')
AMQ8551: WebSphere MQ namelist changed.
```

**Anmerkung:** Sie müssen sowohl die vorhandenen Namen im Namenslistenobjekt als auch die neuen Namen, die Sie hinzufügen, im Befehl **alter namelist** angeben.

## Nächste Schritte

Informationen über den Datenstrom werden an andere Broker in der Hierarchie übergeben.

Wenn es sich bei einem Broker um Version 6.0 handelt, verwalten Sie ihn als Version 6.0 -Broker. Dies bedeutet, dass Sie die Möglichkeit haben, die Datenstromwarteschlange manuell zu erstellen oder den Broker dynamisch zu erstellen, wenn er benötigt wird. Die Warteschlange basiert auf der Modellwarteschlangendefinition `SYSTEM.BROKER.MODEL.STREAM`.

Wenn es sich bei einem Broker um Version 7.1 handelt, müssen Sie jeden Version 7.1 -Warteschlangenmanager manuell in der Hierarchie konfigurieren.

## Datenstrom löschen

Sie können einen Datenstrom aus einem Warteschlangenmanager von IBM WebSphere MQ Version 7.1 oder höher löschen.

### Vorbereitende Schritte

Die Verwendung von eingereichter Publish/Subscribe ist in IBM WebSphere MQ Version 7.1 veraltet.

Bevor Sie einen Datenstrom löschen, müssen Sie sicherstellen, dass keine weiteren Subskriptionen für den Datenstrom vorhanden sind, und alle Anwendungen, die den Datenstrom verwenden, in den Wartemodus versetzen. Wenn Veröffentlichungen weiterhin in einen gelöschten Datenstrom fließen, ist hoher Verwaltungsaufwand notwendig, um den einwandfreien Betriebsstatus des Systems wiederherzustellen.

### Informationen zu diesem Vorgang

Anweisungen zum Löschen des Datenstroms aus allen Version 6.0 -Warteschlangenmanagern, mit denen er verbunden ist, finden Sie unter [Datenstrom löschen \(ps11870\\_.htm in der Dokumentation zu v6.0\)](#).

### Vorgehensweise

1. Suchen Sie nach allen verbundenen Brokern, die diesen Datenstrom hosten.
2. Brechen Sie alle Subskriptionen für den Datenstrom auf allen Brokern ab.
3. Entfernen Sie die Warteschlange (mit dem gleichen Namen wie der Datenstrom) aus der Namensliste `SYSTEM.QPUBSUB.QUEUE.NAMELIST`.
4. Löschen oder bereinigen Sie alle Nachrichten aus der Warteschlange mit demselben Namen wie der Datenstrom.
5. Löschen Sie die Warteschlange mit demselben Namen wie der Datenstrom.
6. Löschen Sie das zugeordnete Themenobjekt.

### Nächste Schritte

1. Wiederholen Sie die Schritte 3 bis 5 für alle anderen verbundenen Version 7.1 -Warteschlangenmanager oder höher, die den Datenstrom als Host für den Datenstrom enthalten.
2. Entfernen Sie den Datenstrom von allen anderen verbundenen Version 6.0 -Warteschlangenmanagern oder früheren Warteschlangenmanagern.

## Einen Subskriptionspunkt hinzufügen

Vorgehensweise zum Hinzufügen eines Subskriptionspunkts, der nicht von IBM WebSphere MQ Event Broker oder IBM WebSphere MQ Message Broker von **migmbbrk** migriert wurde. Erweitern Sie eine vorhandene Publish/Subscribe-Anwendung, die Sie von IBM WebSphere MQ Event Broker oder IBM WebSphere MQ Message Broker migriert haben, mit einem neuen Subskriptionspunkt.



## Vorbereitende Schritte

1. Führen Sie die Migration von IBM WebSphere MQ Event Broker und IBM WebSphere MQ Message Broker Version 6.0 auf IBM WebSphere MQ Version 7.1 durch.
2. Stellen Sie sicher, dass der Subskriptionspunkt nicht bereits in `SYSTEM.QPUBSUB.SUBPOINT.NAMELIST` definiert ist.
3. Überprüfen Sie, ob ein Themenobjekt oder eine Themenzeichenfolge mit dem gleichen Namen wie der Subskriptionspunkt vorhanden ist.

## Informationen zu diesem Vorgang

Vorhandene IBM WebSphere MQ Event Broker-Anwendungen verwenden Subskriptionspunkte. Neue IBM WebSphere MQ Version 7.1 -Anwendungen verwenden keine Subskriptionspunkte, aber sie können mit vorhandenen Anwendungen interagieren, die den Migrationsmechanismus für Subskriptionspunkte verwenden.

Ein Subskriptionspunkt wurde möglicherweise nicht von **migmbbrk** migriert, wenn der Subskriptionspunkt zum Zeitpunkt der Migration nicht verwendet wurde.

Sie können einen Subskriptionspunkt zu vorhandenen eingereichten Publish/Subscribe-Programmen hinzufügen, die von IBM WebSphere MQ Event Broker migriert wurden.

Subskriptionspunkte funktionieren nicht mit eingereichten Publish/Subscribe-Programmen, die `MQRFH1`-Header verwenden, die von IBM WebSphere MQ Version 6.0 oder früher migriert wurden.

Es ist nicht erforderlich, Subskriptionspunkte hinzuzufügen, um integrierte Publish/Subscribe-Anwendungen zu verwenden, die für IBM WebSphere MQ Version 7.1 geschrieben wurden.

## Vorgehensweise

1. Fügen Sie den Namen des Subskriptionspunkts zu `SYSTEM.QPUBSUB.SUBPOINT.NAMELIST` hinzu.
  - Unter `z/OS` ist **NLTYPE** standardmäßig auf `NONE` gesetzt.
  - Wiederholen Sie den Schritt auf jedem Warteschlangenmanager, der in derselben Publish/Subscribe-Topologie verbunden ist.
2. Fügen Sie ein Themenobjekt hinzu, vorzugsweise geben Sie ihm den Namen des Subskriptionspunkts mit einer Themenzeichenfolge an, die mit dem Namen des Subskriptionspunkts übereinstimmt.
  - Wenn sich der Subskriptionspunkt in einem Cluster befindet, fügen Sie das Themenobjekt als Clusterthema auf dem Clusterthemenhost hinzu.
  - Wenn ein Themenobjekt mit der gleichen Themenzeichenfolge wie der Name des Subskriptionspunkts vorhanden ist, verwenden Sie das vorhandene Themenobjekt. Sie müssen die Auswirkungen des Subskriptionspunkts kennen, der ein vorhandenes Thema wiederverwendet. Wenn das vorhandene Thema Teil einer vorhandenen Anwendung ist, müssen Sie die Kollision zwischen zwei identisch benannten Themen auflösen.
  - Wenn ein Themenobjekt mit demselben Namen wie der Subskriptionspunkt vorhanden ist, aber eine andere Themenzeichenfolge enthält, erstellen Sie ein Thema mit einem anderen Namen.
3. Setzen Sie das Attribut **Topic WILDCARD** auf den Wert `BLOCK`.

Wenn Sie Abonnements für `#` oder `*` blockieren, werden Platzhalterzeichen für Abonnements für Subskriptionspunkte isoliert, siehe [Wildcards und Subskriptionspunkte](#).

4. Legen Sie alle Attribute fest, die Sie im Themenobjekt benötigen.

## Beispiel

Das Beispiel zeigt eine **runmqsc**-Befehlsdatei, die zwei Subskriptionspunkte hinzufügt, `USD` und `GBP`.

```
DEFINE TOPIC(USD) TOPICSTR(USD)
DEFINE TOPIC(GBP) TOPICSTR(GBP) WILDCARD(BLOCK)
ALTER NL(SYSTEM.QPUBSUB.SUBPOINT.NAMELIST) NAMES(SYSTEM.BROKER.DEFAULT.SUBPOINT, USD, GBP)
```

### Anmerkung:

1. Schließen Sie den Standardsubskriptionspunkt in die Liste der Subskriptionspunkte ein, die mit dem Befehl **ALTER** hinzugefügt wurden. **ALTER** löscht vorhandene Namen in der Namensliste.
2. Definieren Sie die Themen, bevor Sie die Namensliste ändern. Der Warteschlangenmanager prüft nur die Namensliste, wenn der Warteschlangenmanager gestartet wird und wenn die Namensliste geändert wird.

## Warteschlangenmanager mit einer Brokerhierarchie verbinden

Sie können einen lokalen Warteschlangenmanager mit einem übergeordneten Warteschlangenmanager verbinden, um eine Brokerhierarchie zu ändern.

### Vorbereitende Schritte

1. Modus für eingereichtes Publish/Subscribe aktivieren. Siehe [In Warteschlange eingereichte Publish/Subscribe starten](#).
2. Diese Änderung wird über eine IBM WebSphere MQ -Verbindung an den übergeordneten Warteschlangenmanager weitergegeben. Es gibt zwei Möglichkeiten, die Verbindung herzustellen.
  - Verbinden Sie die Warteschlangenmanager mit einem IBM WebSphere MQ -Cluster (siehe [Warteschlangenmanager zu einem Cluster hinzufügen](#)).
  - Richten Sie eine Punkt-zu-Punkt-Kanalverbindung unter Verwendung einer Übertragungswarteschlange oder eines Warteschlangenmanager-Aliasnamens ein, die denselben Namen wie der übergeordnete Warteschlangenmanager hat. Weitere Informationen zum Einrichten einer Punkt-zu-Punkt-Kanalverbindung finden Sie im Abschnitt [WebSphere MQ Verfahren für verteiltes Messaging](#).

### Informationen zu diesem Vorgang

Verwenden Sie den Befehl `ALTER QMGR PARENT (PARENT_NAME) runmqsc`, um untergeordnete Elemente mit übergeordneten Elementen zu verbinden.

Verteiltes Publish/Subscribe wird mithilfe von WS-Manager-Clustern und Clusterthemendefinitionen implementiert. Für die Interoperabilität mit IBM WebSphere MQ Version 6.0 und WebSphere Message Broker Version 6.1 und WebSphere Event Broker Version 6.1 und früher können Sie auch Warteschlangenmanager von Version 7.1 oder höher mit einer Brokerhierarchie verbinden, solange der Publish/Subscribe-Modus in der Warteschlange aktiviert ist.

### Vorgehensweise

`ALTER QMGR PARENT (ÜBERGEORDNET)`

### Beispiel

Das erste Beispiel zeigt, wie QM2 als untergeordnetes Element von QM1 zugeordnet wird und wie QM2 nach seiner Verbindung abgefragt wird:

```
C:>runmqsc QM2
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM2
alter qmgr parent(QM1)
 1 : alter qmgr parent(QM1)
AMQ8005: WebSphere MQ queue manager changed.
display pubsub all
 2 : display pubsub all
AMQ8723: Display pub/sub status details.
QMNAME(QM2) TYPE(LOCAL)
STATUS(ACTIVE)
AMQ8723: Display pub/sub status details.
QMNAME(QM1) TYPE(PARENT)
STATUS(ACTIVE)
```

Das nächste Beispiel zeigt das Ergebnis der Abfrage von QM1 für die zugehörigen Verbindungen:

```

C:\Documents and Settings\Admin>runmqsc QM1
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM1.
display pubsub all
 2 : display pubsub all
AMQ8723: Display pub/sub status details.
 QMNAME(QM1) TYPE(LOCAL)
 STATUS(ACTIVE)
AMQ8723: Display pub/sub status details.
 QMNAME(QM2) TYPE(CHILD)
 STATUS(ACTIVE)

```

## Nächste Schritte

Sie können Themen auf einem Broker oder Warteschlangenmanager definieren, die für Publisher und Subskribenten auf den verbundenen Warteschlangenmanagern verfügbar sind. Weitere Informationen finden Sie unter [Verwaltungsthema definieren](#) .

### Zugehörige Konzepte

[Datenströme und Themen](#)

[Einführung in das Publish/Subscribe-Messaging für WebSphere MQ](#)

### Zugehörige Verweise

[DISPLAY PUBSUB](#)

## Warteschlangenmanager aus einer Brokerhierarchie trennen

Trennen Sie die Verbindung zwischen einem untergeordneten Warteschlangenmanager und einem übergeordneten Warteschlangenmanager in einer Brokerhierarchie.

### Informationen zu diesem Vorgang

Mit dem Befehl **ALTER QMGR** können Sie die Verbindung zwischen einem Warteschlangenmanager und einer Brokerhierarchie trennen. Sie können einen WS-Manager jederzeit in beliebiger Reihenfolge trennen.

Die entsprechende Anforderung zum Aktualisieren des übergeordneten Elements wird gesendet, wenn die Verbindung zwischen den Warteschlangenmanagern ausgeführt wird.

### Vorgehensweise

```
ALTER QMGR PARENT('')
```

### Beispiel

```

C:\Documents and Settings\Admin>runmqsc QM2
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM2.
 1 : alter qmgr parent('')
AMQ8005: WebSphere MQ queue manager changed.
 2 : display pubsub type(child)
AMQ8147: WebSphere MQ object not found.
display pubsub type(parent)
 3 : display pubsub type(parent)
AMQ8147: WebSphere MQ object not found.

```

## Nächste Schritte

Sie können alle Datenströme, Warteschlangen und manuell definierten Kanäle, die nicht mehr benötigt werden, löschen.

## WS-Manager-Cluster konfigurieren

Über die Links in diesem Abschnitt erfahren Sie, wie Cluster funktionieren, wie eine Clusterkonfiguration entworfen wird und wie ein einfacher Cluster eingerichtet wird.

## Vorbereitende Schritte

Eine Einführung in Clustering-Konzepte finden Sie in den folgenden Themen:

- [Wie funktionieren Cluster?](#)
- [„Vergleich von Clustering und verteilter Steuerung von Warteschlangen“ auf Seite 174](#)
- [„Komponenten eines Clusters“ auf Seite 176](#)

Wenn Sie den WS-Manager-Cluster entwerfen, müssen Sie einige Entscheidungen treffen. Sie müssen zuerst entscheiden, welche WS-Manager im Cluster die vollständigen Repositorys der Clusterinformationen enthalten sollen. Jeder Warteschlangenmanager, den Sie erstellen, kann in einem Cluster arbeiten. Sie können eine beliebige Anzahl von Warteschlangenmanagern für diesen Zweck auswählen, aber die ideale Zahl ist zwei. Informationen zum Auswählen von Warteschlangenmanagern zum Speichern der vollständigen Repositorys finden Sie unter [„Clusterwarteschlangenmanager für die Aufnahme von vollständigen Repositorys auswählen“ auf Seite 192](#).

Weitere Informationen zum Entwerfen des Clusters finden Sie in den folgenden Abschnitten:

- [„Cluster verwalten“ auf Seite 193](#)
- [„Namenskonventionen für Cluster“ auf Seite 194](#)
- [„Überlappende Cluster“ auf Seite 195](#)

## Beispiel

Der kleinste mögliche Cluster enthält nur zwei WS-Manager. In diesem Fall enthalten beide WS-Manager vollständige Repositorys. Sie benötigen nur wenige Definitionen, um den Cluster zu konfigurieren, und dennoch gibt es bei jedem WS-Manager einen hohen Grad an Autonomie.

[Abbildung 21 auf Seite 172](#) zeigt einen Cluster mit dem Namen DEMOCLSTR und zwei Warteschlangenmanagern mit den Namen QM1 und QM2.

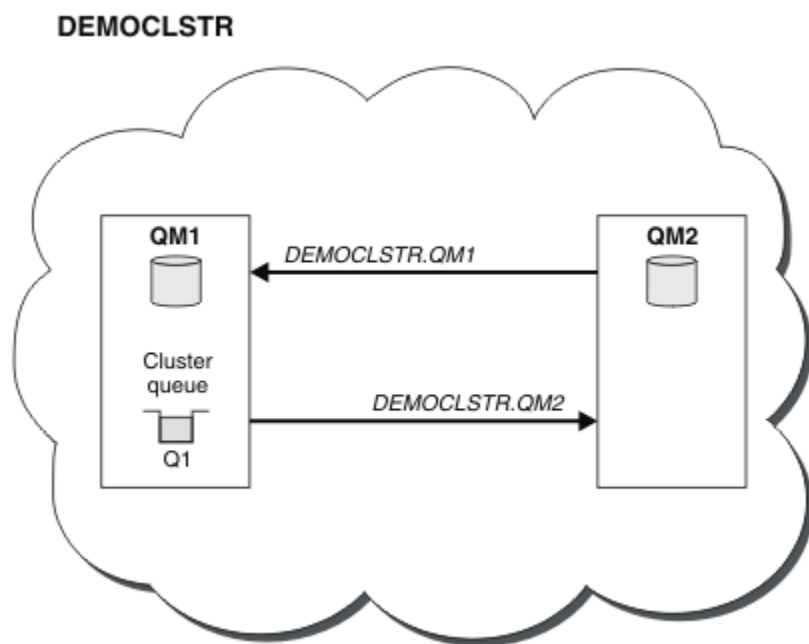


Abbildung 21. Kleiner Cluster mit zwei WS-Managern

- Die Warteschlangenmanager haben lange Namen wie LONDON und NEWYORK. In den erweiterten Tasks und Lastausgleichstasks werden dieselben Namen verwendet. Unter IBM WebSphere MQ für z/OS sind die Namen von Warteschlangenmanagern auf vier Zeichen beschränkt.

- Die Namen der Warteschlangenmanager bedeuten, dass sich jeder Warteschlangenmanager auf einer separaten Maschine befindet. Sie können diese Tasks mit allen Warteschlangenmanagern auf derselben Maschine ausführen.
- Die Tasks verwenden IBM WebSphere MQ -Scriptbefehle, wie sie vom Systemadministrator mit **MQSC** -Befehlen eingegeben würden. Es gibt andere Möglichkeiten zur Eingabe von Befehlen, einschließlich der Verwendung des einfacheren IBM WebSphere MQ Explorer. Die Verwendung von WebSphere MQ -Scriptbefehlen soll veranschaulichen, welche IBM WebSphere MQ -Befehle in den Tasks verwendet werden.

Anweisungen zum Einrichten eines ähnlichen Beispielclusters finden Sie in [„Neuen Cluster einrichten“](#) auf Seite 201.

## Nächste Schritte

Weitere Informationen zum Konfigurieren und Arbeiten mit Clustern finden Sie in den folgenden Abschnitten:

- [„Kommunikation in einem Cluster einrichten“](#) auf Seite 198
- [„IBM WebSphere MQ -Cluster verwalten“](#) auf Seite 200
- [„Nachrichten an und von Clustern weiterleiten“](#) auf Seite 268
- [„Cluster für Workload-Management verwenden“](#) auf Seite 284

Weitere Informationen zur Konfiguration Ihres Clusters finden Sie unter [„Tipps zum Clustering“](#) auf Seite 196.

### Zugehörige Konzepte

[Cluster](#)

## Zugriffssteuerung und mehrere Clusterübertragungswarteschlangen

Wählen Sie zwischen drei Prüfmodi aus, wenn eine Anwendung Nachrichten in ferne Clusterwarteschlangen einreicht. Die Modi werden über Remotezugriff auf die Clusterwarteschlange überprüft, lokal auf SYSTEM. CLUSTER. TRANSMIT. QUEUE geprüft oder auf lokale Profile für die Clusterwarteschlange oder den Cluster-WS-Manager überprüft.

IBM WebSphere MQ gibt Ihnen die Möglichkeit, entweder lokal oder lokal und remote zu überprüfen, ob ein Benutzer berechtigt ist, eine Nachricht in eine ferne Warteschlange einzureihen. Eine typische IBM WebSphere MQ -Anwendung verwendet nur die lokale Überprüfung und stützt sich auf den fernen Warteschlangenmanager, der die Zugriffsprüfungen auf dem lokalen WS-Manager vertraut. Wenn die ferne Prüfung nicht verwendet wird, wird die Nachricht mit der Berechtigung des fernen Nachrichtenkanalprozesses in die Zielwarteschlange gestellt. Um die Fernprüfung verwenden zu können, müssen Sie die Berechtigung 'put' für den empfangenden Kanal auf die Kontextsicherheit setzen.

Die lokalen Prüfungen werden für die Warteschlange, die die Anwendung öffnet, durchgeführt. Bei der verteilten Steuerung von Warteschlangen öffnet die Anwendung normalerweise eine Definition einer fernen Warteschlange und es werden Zugriffsprüfungen für die Definition der fernen Warteschlange durchgeführt. Wenn die Nachricht mit einem vollständigen Routing-Header verbunden wird, werden die Prüfungen für die Übertragungswarteschlange durchgeführt. Wenn eine Anwendung eine Clusterwarteschlange öffnet, die sich nicht im lokalen WS-Manager befindet, gibt es kein lokales Objekt, das überprüft werden kann. Die Zugriffssteuerungsprüfungen werden für die Clusterübertragungswarteschlange SYSTEM. CLUSTER. TRANSMIT. QUEUE durchgeführt. Selbst bei mehreren Clusterübertragungswarteschlangen werden ab Version 7.5 lokale Zugriffssteuerungsprüfungen für ferne Clusterwarteschlangen für SYSTEM. CLUSTER. TRANSMIT. QUEUE durchgeführt.

Die Auswahl der lokalen oder fernen Prüfung ist eine Auswahl zwischen zwei Extremwerte. Die ferne Überprüfung ist in differenzierter Ausführung. Jeder Benutzer muss über ein Zugriffssteuerungsprofil auf jedem WS-Manager im Cluster verfügen, um in eine beliebige Clusterwarteschlange zu stellen. Die lokale Überprüfung ist grob-grainiert. Jeder Benutzer benötigt nur ein Zugriffssteuerungsprofil für die Clusterübertragungswarteschlange auf dem Warteschlangenmanager, mit dem sie verbunden sind. Mit

diesem Profil können sie eine Nachricht in jede Clusterwarteschlange auf einem beliebigen WS-Manager in einem beliebigen Cluster einlegen.

Ab Version 7.1 haben Administratoren eine andere Möglichkeit, die Zugriffssteuerung für Clusterwarteschlangen zu konfigurieren. Mit dem Befehl **setmqaut** können Sie ein Sicherheitsprofil für eine Clusterwarteschlange auf einem beliebigen Warteschlangenmanager im Cluster erstellen. Das Profil wirkt sich darauf aus, wenn Sie eine ferne Clusterwarteschlange lokal öffnen und dabei nur den Namen der Warteschlange angeben. Sie können auch ein Profil für einen fernen WS-Manager einrichten. Wenn Sie dies tun, kann der Warteschlangenmanager das Profil eines Benutzers überprüfen, der eine Clusterwarteschlange öffnet, indem er einen vollständig qualifizierten Namen bereitstellt.

Die neuen Profile funktionieren nur, wenn Sie die Zeilengruppe des Warteschlangenmanagers ändern, **ClusterQueueAccessControl** in RQMName . Der Standardwert ist Xmitq . Sie müssen Profile für alle vorhandenen Clusterwarteschlangen erstellen, die Clusterwarteschlangen verwenden. Wenn Sie die Zeilengruppe in RQMName ändern, ohne Profile zu erstellen, werden die Anwendungen wahrscheinlich fehlschlagen.

**Tipp:** Die Änderungen, die an der Zugriffsprüfung für Clusterwarteschlangen in Version 7.1 vorgenommen wurden, gelten nicht für die ferne Warteschlangensteuerung. Es werden weiterhin Zugriffsprüfungen für lokale Definitionen durchgeführt. Die Änderungen bedeuten, dass Sie denselben Ansatz verfolgen können, um die Zugriffsprüfung für Clusterwarteschlangen und Clusterthemen zu konfigurieren.

### Zugehörige Konzepte

„Clustering: Anwendungsisolation mit mehreren Clusterübertragungswarteschlangen“ auf Seite 306  
Sie können die Nachrichtenflüsse zwischen Warteschlangenmanagern in einem Cluster isolieren. Sie können Nachrichten, die von verschiedenen Clustersenderkanälen transportiert werden, in verschiedene Clusterübertragungswarteschlangen stellen. Sie können den Ansatz in einem einzelnen Cluster oder mit überlappenden Clustern verwenden. Das Thema enthält Beispiele und einige bewährte Verfahren, die Sie bei der Auswahl eines zu verwendenden Ansatzes führen.

## Vergleich von Clustering und verteilter Steuerung von Warteschlangen

Vergleichen Sie die Komponenten, die für die Verbindung von WS-Managern mit verteilter Steuerung von Warteschlangen und Clustering definiert werden müssen.

Wenn Sie keine Cluster verwenden, sind Ihre Warteschlangenmanager unabhängig und kommunizieren mit der verteilten Steuerung von Warteschlangen. Wenn ein Warteschlangenmanager Nachrichten an einen anderen senden muss, müssen Sie Folgendes definieren:

- eine Übertragungswarteschlange
- Ein Kanal zum fernen Warteschlangenmanager

Abbildung 22 auf Seite 174 zeigt die Komponenten, die für die verteilte Steuerung von Warteschlangen erforderlich sind.

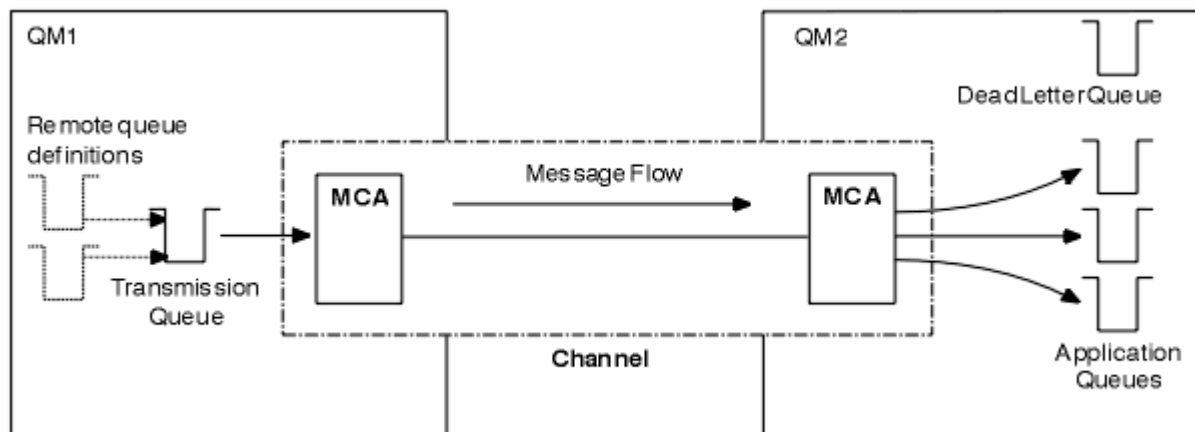


Abbildung 22. Verteilte Steuerung von Warteschlangen

Wenn Sie Warteschlangenmanager in einem Cluster gruppieren, stehen Warteschlangen in einem beliebigen WS-Manager allen anderen Warteschlangenmanagern im Cluster zur Verfügung. Jeder WS-Manager kann ohne explizite Definitionen eine Nachricht an jeden anderen Warteschlangenmanager in demselben Cluster senden. Sie stellen keine Kanaldefinitionen, Definitionen für ferne Warteschlangen oder Übertragungswarteschlangen für die einzelnen Ziele zur Verfügung. Jeder WS-Manager in einem Cluster verfügt über eine einzige Übertragungswarteschlange, von der er Nachrichten an jeden anderen WS-Manager im Cluster übertragen kann. Jeder WS-Manager in einem Cluster muss nur Folgendes definieren:

- Ein Clusterempfängerkanal, auf dem Nachrichten empfangen werden sollen.
- Ein Clustersenderkanal, mit dem er sich selbst einführt und die Informationen zum Cluster

### Definitionen zum Festlegen eines Clusters im Vergleich zu verteilten Warteschlangensteuerung

Sehen Sie sich [Abbildung 23](#) auf Seite 175 an, in dem jeweils vier Warteschlangenmanager mit jeweils zwei Warteschlangen angezeigt werden. Überlegen Sie, wie viele Definitionen benötigt werden, um diese WS-Manager mit Hilfe der verteilten Steuerung von Warteschlangen zu verbinden. Vergleichen Sie die Anzahl der Definitionen, die zum Festlegen des gleichen Netzes wie ein Cluster benötigt werden.

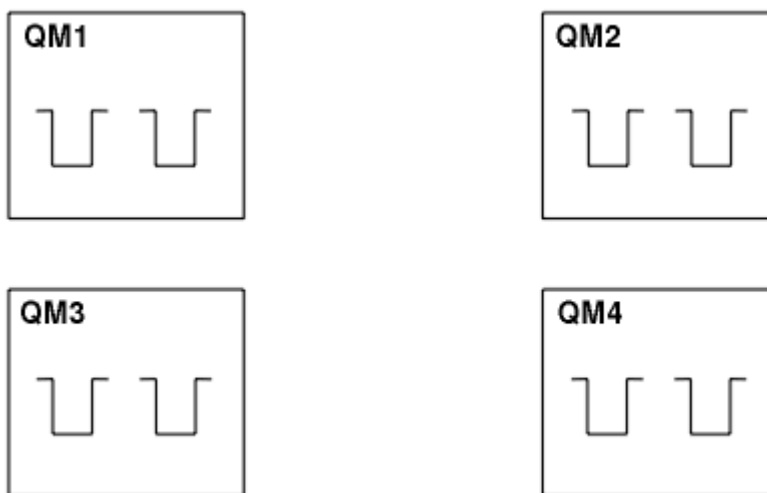


Abbildung 23. Ein Netz mit vier Warteschlangenmanagern

### Definitionen zum Festlegen eines Netzes unter Verwendung der verteilten Steuerung von Warteschlangen

Um das in [Abbildung 22](#) auf Seite 174 unter Verwendung der verteilten Warteschlangensteuerung gezeigte Netz einzurichten, können Sie die folgenden Definitionen verwenden:

| Tabelle 24. Definitionen für verteilte Steuerung von Warteschlangen                                                            |                       |              |
|--------------------------------------------------------------------------------------------------------------------------------|-----------------------|--------------|
| Beschreibung                                                                                                                   | Anzahl pro WS-Manager | Gesamtanzahl |
| Eine Senderkanaldefinition für einen Kanal, auf dem Nachrichten an alle anderen Warteschlangenmanager gesendet werden sollen.  | 3                     | 12           |
| Eine Empfängerkanaldefinition für einen Kanal, auf dem Nachrichten von jedem anderen WS-Manager empfangen werden sollen.       | 3                     | 12           |
| Eine Übertragungswarteschlangendefinition für eine Übertragungswarteschlange zu jedem anderen Warteschlangenmanager            | 3                     | 12           |
| Eine lokale Warteschlangendefinition für jede lokale Warteschlange.                                                            | 2                     | 8            |
| Eine Definition einer fernen Warteschlange für jede ferne Warteschlange, in die dieser WS-Manager Nachrichten einlegen möchte. | 6                     | 24           |



Sie können diese Anzahl von Definitionen unter Verwendung generischer Empfängerkanaldefinitionen reduzieren. Die maximale Anzahl von Definitionen kann bis zu 17 auf jedem Warteschlangenmanager betragen, was insgesamt 68 für dieses Netz ist.

## Definitionen zum Festlegen eines Netzes mit Clustern

Um das in [Abbildung 22 auf Seite 174](#) gezeigte Netz unter Verwendung von Clustern einzurichten, benötigen Sie die folgenden Definitionen:

| <i>Tabelle 25. Definitionen für Clustering</i>                                                                                                   |                              |                     |
|--------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------|---------------------|
| <b>Beschreibung</b>                                                                                                                              | <b>Anzahl pro WS-Manager</b> | <b>Gesamtanzahl</b> |
| Clustersenderkanaldefinition für einen Kanal, auf dem Nachrichten an einen Repository-WS-Manager gesendet werden sollen                          | 1                            | 4                   |
| Eine Clusterempfängerkanaldefinition für einen Kanal, auf dem Nachrichten von anderen Warteschlangenmanagern im Cluster empfangen werden sollen. | 1                            | 4                   |
| Eine lokale Warteschlangendefinition für jede lokale Warteschlange.                                                                              | 2                            | 8                   |

Um diesen Cluster von Warteschlangenmanagern (mit zwei vollständigen Repositories) zu konfigurieren, benötigen Sie auf jedem Warteschlangenmanager vier Definitionen, insgesamt sind insgesamt 16 Definitionen vorhanden. Außerdem müssen Sie die Warteschlangenmanager-Definitionen für zwei der Warteschlangenmanager ändern, damit sie vollständige Warteschlangenmanager-Repository-Warteschlangenmanager für den Cluster bilden.

Es ist nur eine CLUSSDR - und eine CLUSRCVR -Kanaldefinition erforderlich. Wenn der Cluster definiert ist, können Sie Warteschlangenmanager (außer den Repository-WS-Managern) hinzufügen oder entfernen, ohne dass die anderen WS-Manager gestört werden.

Wenn Sie einen Cluster verwenden, wird die Anzahl der Definitionen reduziert, die zum Festlegen eines Netzes mit vielen Warteschlangenmanagern erforderlich sind.

Wenn weniger Definitionen vorhanden sind, besteht die Gefahr eines Fehlers:

- Objektnamen stimmen immer überein, z. B. der Kanalname in einem Sender-Empfänger-Paar.
- Der in einer Kanaldefinition angegebene Übertragungswarteschlangenname stimmt immer mit der korrekten Übertragungswarteschlangendefinition oder dem Namen der Übertragungswarteschlange überein, die in einer Definition einer fernen Warteschlange angegeben ist.
- Eine QREMOTE -Definition verweist immer auf die richtige Warteschlange auf dem fernen Warteschlangenmanager.

Sobald ein Cluster konfiguriert ist, können Sie Clusterwarteschlangen von einem Warteschlangenmanager in einen anderen im Cluster verschieben, ohne dass Systemverwaltungsaufgaben für einen anderen Warteschlangenmanager ausgeführt werden müssen. Es besteht keine Möglichkeit, die Definitionen von Kanal-, Fernwarteschlangen- oder Übertragungswarteschlangen zu löschen oder zu ändern. Sie können neue Warteschlangenmanager zu einem Cluster hinzufügen, ohne dass das vorhandene Netz unterbrochen wird.

## Komponenten eines Clusters

Cluster bestehen aus Warteschlangenmanagern, Clusterrepositories, Clusterkanälen und Clusterwarteschlangen.

In den folgenden Abschnitten finden Sie Informationen zu den einzelnen Clusterkomponenten:

### Zugehörige Konzepte

[Cluster](#)

[„Vergleich von Clustering und verteilter Steuerung von Warteschlangen“ auf Seite 174](#)



Vergleichen Sie die Komponenten, die für die Verbindung von WS-Managern mit verteilter Steuerung von Warteschlangen und Clustering definiert werden müssen.

[„IBM WebSphere MQ -Cluster verwalten“ auf Seite 200](#)

Sie können IBM WebSphere MQ -Cluster erstellen, erweitern und verwalten.

### **Zugehörige Tasks**

[„WS-Manager-Cluster konfigurieren“ auf Seite 171](#)

Über die Links in diesem Abschnitt erfahren Sie, wie Cluster funktionieren, wie eine Clusterkonfiguration entworfen wird und wie ein einfacher Cluster eingerichtet wird.

[„Neuen Cluster einrichten“ auf Seite 201](#)

Führen Sie die folgenden Anweisungen aus, um den Beispielcluster zu konfigurieren. In separaten Anweisungen wird beschrieben, wie der Cluster auf TCP/IP, LU 6.2 und mit einer einzelnen Übertragungswarteschlange oder mehreren Übertragungswarteschlangen eingerichtet wird. Testen Sie den Cluster, indem Sie eine Nachricht von einem WS-Manager an den anderen Warteschlangenmanager senden.

## **Cluster-Repository**

Ein Repository ist eine Zusammenstellung von Informationen über die Warteschlangenmanager, die zu einem Cluster gehören.

Zu den Repositoryinformationen gehören die Namen der Warteschlangenmanager, ihre Standorte, ihre Kanäle, welche Warteschlangen von ihnen betrieben werden und weitere Informationen. Die Informationen werden in Form von Nachrichten in einer Warteschlange mit dem Namen `SYSTEM.CLUSTER.REPOSITORY.QUEUE` gespeichert. Die Warteschlange ist eines der Standardobjekte. Sie wird definiert, wenn Sie einen WebSphere MQ -Warteschlangenmanager erstellen, außer unter WebSphere MQ für z/OS.

Üblicherweise verfügen zwei Warteschlangenmanager in einem Cluster über ein vollständiges Repository. Die übrigen Warteschlangenmanager verfügen alle über ein Teilrepository.

### **Vollständiges Repository und Teilrepository**

Ein Warteschlangenmanager, der über einen vollständigen Satz von Informationen über jeden Warteschlangenmanager im Cluster verfügt, besitzt ein vollständiges Repository. Andere Warteschlangenmanager im Cluster besitzen Teilrepositorys mit einer Untermenge der Informationen in den vollständigen Repositorys.

Ein Teilrepository enthält nur Informationen über die Warteschlangenmanager, mit denen der Warteschlangenmanager Nachrichten austauschen muss. Die Warteschlangenmanager fordern Aktualisierungen zu den von Ihnen benötigten Informationen an, d. h., wenn sich die Informationen ändern, sendet ihnen der Warteschlangenmanager mit dem vollständigen Repository die neuen Informationen. Die meiste Zeit enthält ein Teilrepository alle Informationen, die ein Warteschlangenmanager zur Ausführung innerhalb des Clusters benötigt. Falls zusätzliche Informationen benötigt werden, ruft er diese aus dem vollständigen Repository ab und nimmt eine entsprechende Aktualisierung des Teilrepositorys vor. Für die Anforderung und den Empfang von Aktualisierungen der Repositorys verwenden die Warteschlangenmanager die Warteschlange `SYSTEM.CLUSTER.COMMAND.QUEUE`. Diese Warteschlange ist eines der Standardobjekte.

## **Clusterwarteschlangenmanager**

Ein Clusterwarteschlangenmanager ist ein Warteschlangenmanager, der Mitglied eines Clusters ist.

Ein Warteschlangenmanager kann mehreren Clustern angehören. Jeder Clusterwarteschlangenmanager muss einen Namen haben, der für alle Cluster, denen er angehört, eindeutig ist.

Ein Clusterwarteschlangenmanager kann Warteschlangen betreiben, die er den anderen Warteschlangenmanagern im Cluster zugänglich macht. Ein Clusterwarteschlangenmanager muss keine Warteschlangen hosten oder zugänglich machen. Er kann Nachrichten in den Cluster einspeisen und nur Antworten empfangen, die explizit an ihn gerichtet sind, und nicht an ausgeschriebene Warteschlangen.

In WebSphere MQ für z/OS kann ein Clusterwarteschlangenmanager Mitglied einer Gruppe mit gemeinsamer Warteschlange sein. In diesem Fall teilt er seine Warteschlangendefinitionen mit anderen Warteschlangenmanagern in derselben Gruppe mit gemeinsamer Warteschlange.

Clusterwarteschlangenmanager sind autonom. Sie besitzen die vollständige Kontrolle über die von ihnen definierten Warteschlangen und Kanäle. Ihre Definitionen können nicht von anderen Warteschlangenmanagern geändert werden (im Gegensatz zu Warteschlangenmanagern in derselben Gruppe mit gemeinsamer Warteschlange). Repository-Warteschlangenmanager kontrollieren nicht die Definitionen der anderen Warteschlangenmanager im Cluster. Sie verfügen über einen vollständigen Satz von Definitionen, der bei Bedarf verwendet werden kann. Ein Cluster ist eine Verknüpfung von Warteschlangenmanagern.

Nachdem Sie auf einem Clusterwarteschlangenmanager eine Definition erstellt oder geändert haben, werden die Informationen an den Warteschlangenmanager mit dem vollständigen Repository gesendet. Andere Repositories im Cluster werden später aktualisiert.

## **Warteschlangenmanager mit vollständigem Repository**

Ein Warteschlangenmanager mit vollständigem Repository ist ein Clusterwarteschlangenmanager, der über eine vollständige Darstellung der Ressourcen des Clusters verfügt. Um Verfügbarkeit sicherzustellen, sollten in jedem Cluster mindestens zwei Warteschlangenmanager mit vollständigem Repository eingerichtet werden. Warteschlangenmanager mit vollständigem Repository empfangen Informationen, die von den anderen Warteschlangenmanagern im Cluster gesendet werden, und aktualisieren ihre Repositories. Sie tauschen Nachrichten untereinander aus, um sicherzustellen, dass beide immer auf dem neuesten Stand der Informationen über den Cluster sind.

## **Warteschlangenmanager und Repositories**

Jeder Cluster hat mindestens einen (besser zwei) Warteschlangenmanager mit vollständigem Repository, die alle Informationen über die Warteschlangenmanager, Warteschlangen und Kanäle in einem Cluster enthalten. Diese Repositories enthalten auch Anforderungen von den anderen Warteschlangenmanagern im Cluster zur Aktualisierung der Informationen.

Jeder der anderen Warteschlangenmanager verfügt über ein Teilrepository, das Informationen über die Untergruppe von Warteschlangen und Warteschlangenmanagern enthält, mit denen er kommunizieren muss. Die Warteschlangenmanager erstellen ihre Teilrepositories, indem sie Anfragen stellen, wenn sie zum ersten Mal auf eine andere Warteschlange oder einen anderen Warteschlangenmanager zugreifen müssen. Dabei stellen sie die Anforderung, dass sie alle neuen Informationen erhalten, die die jeweilige Warteschlange oder den jeweiligen Warteschlangenmanager betreffen.

Jeder Warteschlangenmanager speichert seine Repositoryinformationen in Form von Nachrichten in der Warteschlange `SYSTEM.CLUSTER.REPOSITORY.QUEUE`. Die Warteschlangenmanager tauschen Repositoryinformationen in Form von Nachrichten in der Warteschlange `SYSTEM.CLUSTER.COMMAND.QUEUE` aus.

Jeder Warteschlangenmanager, der einem Cluster beiträgt, definiert einen Clustersenderkanal (`CLUSDR`) zu einem der Repositories. Er erfährt sofort, welche anderen Warteschlangenmanager im Cluster über vollständige Repositories verfügen. Von dem Moment an kann der Warteschlangenmanager Informationen aus den Repositories anfordern. Wenn der Warteschlangenmanager Informationen an das ausgewählte Repository sendet, sendet er gleichzeitig Informationen an ein anderes Repository (falls vorhanden).

Ein vollständiges Repository wird aktualisiert, wenn der Warteschlangenmanager mit dem Repository Informationen von einem der Warteschlangenmanager empfängt, die mit ihm verbunden sind. Die neuen Informationen werden auch an ein anderes Repository gesendet, um das Risiko einer Verzögerung zu verringern, falls der Repository-Warteschlangenmanager außer Betrieb ist. Da alle Informationen doppelt gesendet werden, müssen Duplikate in den Repositories gelöscht werden. Jedes Informationselement besitzt eine Folgenummer, die von den Repositories zur Erkennung von Duplikaten genutzt wird. Alle Repositories tauschen Nachrichten untereinander aus, um immer auf demselben Stand zu sein.

## Clusterwarteschlangen

Eine Clusterwarteschlange wird von einem Clusterwarteschlangenmanager anderen Warteschlangenmanagern im Cluster zur Verfügung gestellt.

Definieren Sie eine Clusterwarteschlange als lokale Warteschlange auf dem Clusterwarteschlangenmanager, auf dem sich die Warteschlange befindet. Geben Sie den Namen des Clusters an, zu dem die Warteschlange gehört. Das folgende Beispiel zeigt einen **runmqsc** -Befehl zum Definieren einer Clusterwarteschlange mit der Option CLUSTER :

```
DEFINE QLOCAL(Q1) CLUSTER(SALES)
```

Eine Clusterwarteschlangendefinition wird den anderen Warteschlangenmanagern im Cluster zugänglich gemacht. Die anderen Warteschlangenmanager im Cluster können ohne entsprechende Definition einer fernen Warteschlange Nachrichten in eine Clusterwarteschlange einreihen. Über eine Clusternamensliste kann eine Clusterwarteschlange in mehreren Clustern zugänglich gemacht werden.

Wenn eine Warteschlange zugänglich gemacht wird, können alle Warteschlangenmanager im Cluster Nachrichten in diese Warteschlange einreihen. Um eine Nachricht einzureihen, muss der Warteschlangenmanager anhand der vollständigen Repositorys ermitteln, wo sich die Warteschlange befindet. Anschließend fügt der Warteschlangenmanager der Nachricht einige Routing-Informationen hinzu und stellt sie dann in eine Clusterübertragungswarteschlange.

Eine Clusterwarteschlange kann eine Warteschlange sein, die von Mitgliedern einer Gruppe mit gemeinsamer Warteschlange in IBM WebSphere MQ for z/OS gemeinsam genutzt wird.

## Wird gebunden

Sie können einen Cluster erstellen, in dem mehr als ein Warteschlangenmanager eine Instanz derselben Clusterwarteschlange hostet. Stellen Sie sicher, dass alle Nachrichten in einer Sequenz an die gleiche Instanz der Warteschlange gesendet werden. Sie können eine Reihe von Nachrichten an eine bestimmte Warteschlange binden, indem Sie die Option MQOO\_BIND\_ON\_OPEN im Aufruf MQOPEN verwenden.

## Clusterübertragungswarteschlangen

Abgesehen von z/OS kann ein Warteschlangenmanager Nachrichten für andere Warteschlangenmanager in einem Cluster in mehrere Übertragungswarteschlangen stellen. Es gibt zwei Möglichkeiten, einen Warteschlangenmanager so zu konfigurieren, dass er Nachrichten in mehreren Clusterübertragungswarteschlangen speichern kann. Wenn Sie das Warteschlangenmanagerattribut DEFCLXQ auf CHANNEL setzen, wird für jeden Clustersenderkanal automatisch eine andere Clusterübertragungswarteschlange anhand von SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE erstellt. Wenn Sie die Option CLCHNAME für eine Clusterübertragungswarteschlange so setzen, dass sie mit einem oder auch mehreren Clustersenderkanälen übereinstimmt, kann der Warteschlangenmanager in dieser Übertragungswarteschlange Nachrichten für diese Clustersenderkanäle speichern.



**Achtung:** Wenn Sie dedizierte SYSTEM.CLUSTER.TRANSMIT.QUEUES mit einem Warteschlangenmanager verwenden, für den ein Upgrade von einer früheren Version des Produkts durchgeführt wurde, stellen Sie sicher, dass für SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE die Option SHARE/NOSHARE auf **SHARE** gesetzt ist.

Eine Nachricht für eine Clusterwarteschlange auf einem anderen Warteschlangenmanager wird vor dem Senden in eine Clusterübertragungswarteschlange gestellt. Ein Clustersenderkanal überträgt die Nachrichten von einer Clusterübertragungswarteschlange in Clusterempfängerkanäle auf anderen WS-Managern. Standardmäßig enthält eine systemdefinierte Clusterübertragungswarteschlange alle Nachrichten, die an andere Cluster-WS-Manager übertragen werden sollen. Die Warteschlange wird als SYSTEM.CLUSTER.TRANSMIT.QUEUE bezeichnet. Ein Warteschlangenmanager, der Teil eines Clusters ist, kann Nachrichten in dieser Clusterübertragungswarteschlange an jeden anderen WS-Manager im selben Cluster senden.

Eine Definition für die einzelne SYSTEM.CLUSTER.TRANSMIT.QUEUE -Warteschlange wird standardmäßig auf jedem Warteschlangenmanager erstellt, außer unter z/OS.

Auf allen Plattformen außer auf z/OS können Sie einen Warteschlangenmanager für die Weiterleitung von Nachrichten an andere Warteschlangenmanager im Cluster mithilfe von mehreren Übertragungswarteschlangen konfigurieren. Sie können zusätzliche Clusterübertragungswarteschlangen manuell definieren oder die Warteschlangen vom Warteschlangenmanager automatisch erstellen lassen.

Wenn die Warteschlangen automatisch vom Warteschlangenmanager erstellt werden sollen, ändern Sie das WS-Managerattribut `DEFCLXQ` von `SCTQ` in `CHANNEL`. Das Ergebnis ist, dass der Warteschlangenmanager für jeden zu erstellenden Clustersenderkanal eine einzelne Clusterübertragungswarteschlange erstellt. Die Übertragungswarteschlangen werden als permanente dynamische Warteschlangen aus der Modellwarteschlange `SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE` erstellt. Der Name jeder permanenten dynamischen Warteschlange lautet `SYSTEM.CLUSTER.TRANSMIT.ChannelName`. Der Name des Clustersenderkanals, dem jede permanente dynamische Clusterübertragungswarteschlange zugeordnet ist, wird im Attribut `CLCHNAME` der lokalen Übertragungswarteschlange festgelegt. Nachrichten für ferne Cluster-WS-Manager werden in die permanente dynamische Clusterübertragungswarteschlange für den zugehörigen Clustersenderkanal und nicht in `SYSTEM.CLUSTER.TRANSMIT.QUEUE` gestellt.

Zur manuellen Erstellung der Clusterübertragungswarteschlangen müssen Sie zunächst eine lokale Warteschlange erstellen, wobei Sie für das Attribut `USAGE` den Wert `XMITQ` und für das Attribut `CLCHNAME` einen generischen Kanalnamen festlegen, der in einen oder mehrere Clustersenderkanäle aufgelöst werden kann. Weitere Informationen hierzu finden Sie im Abschnitt [ClusterChannelName](#). Wenn Sie Clusterübertragungswarteschlangen manuell erstellen, haben Sie die Möglichkeit, die Übertragungswarteschlange einem einzelnen Clustersenderkanal oder mehreren Clustersenderkanälen zuzuordnen. Das Attribut `CLCHNAME` ist ein generischer Name. Dies bedeutet, dass Sie mehrere Platzhalterzeichen (" \* ") in den Namen einfügen können.

Mit Ausnahme der anfänglichen Clustersenderkanäle, die Sie manuell erstellen, um einen WS-Manager mit einem vollständigen Repository zu verbinden, werden die Clustersenderkanäle automatisch erstellt. Sie werden automatisch erstellt, wenn eine Nachricht zum Übertragen an einen Cluster-WS-Manager vorhanden ist. Sie werden mit demselben Namen wie der Name des Clusterempfängerkanals erstellt, der Clusternachrichten für diesen bestimmten Cluster auf dem Zielwarteschlangenmanager empfängt.

Wenn Sie einer Namenskonvention für Clusterempfängerkanäle folgen, ist es möglich, einen generischen Wert für `CLCHNAME` zu definieren, der verschiedene Arten von Clusternachrichten in verschiedene Übertragungswarteschlangen filtert. Wenn Sie sich beispielsweise für Clusterempfängerkanäle an die Namenskonvention `ClusterName.QmgrName` halten, werden durch den generischen Namen `ClusterName.*` Nachrichten für verschiedene Cluster auf verschiedene Übertragungswarteschlangen verteilt. Sie müssen die Übertragungswarteschlangen manuell definieren und `CLCHNAME` in jeder Übertragungswarteschlange auf `ClusterName.*` setzen.

Änderungen an der Zuordnung von Clusterübertragungswarteschlangen zu Clustersenderkanälen werden nicht sofort wirksam. Die momentan zugeordnete Übertragungswarteschlange, die ein Clustersenderkanal verarbeitet, enthält möglicherweise Nachrichten, die gerade vom Clustersenderkanal übertragen werden. Nur wenn keine Nachrichten in der aktuell zugeordneten Übertragungswarteschlange von einem Clustersenderkanal verarbeitet werden, kann der WS-Manager die Zuordnung des Clustersenderkanals zu einer anderen Übertragungswarteschlange ändern. Dies kann entweder auftreten, wenn keine Nachrichten in der Übertragungswarteschlange verbleiben, die vom Clustersenderkanal verarbeitet werden sollen, oder wenn die Verarbeitung von Nachrichten ausgesetzt wird und der Clustersenderkanal keine " In-Flight " -Nachrichten enthält. In diesem Fall werden alle nicht verarbeiteten Nachrichten für den Clustersenderkanal in die neu zugeordnete Übertragungswarteschlange übertragen, und die Zuordnung der Clustersenderkanäle wird geändert.

Sie können eine Definition einer fernen Warteschlange erstellen, die in eine Clusterübertragungswarteschlange aufgelöst wird. In der Definition befindet sich der WS-Manager `QMX` in demselben Cluster wie der lokale Warteschlangenmanager, und es gibt keine Übertragungswarteschlange, `QMX`.

```
DEFINE QREMOTE(A) RNAME(B) RQMNAME(QMX)
```

Bei der Auflösung von Warteschlangennamen hat die Clusterübertragungswarteschlange Vorrang vor der Standardübertragungswarteschlange. Eine Nachricht, die an A gesendet wird, wird in der Clusterübertragungswarteschlange gespeichert und dann an die ferne Warteschlange B unter `QMX` gesendet.

WS-Manager können auch mit anderen WS-Managern kommunizieren, die nicht Teil eines Clusters sind. Sie müssen Kanäle und eine Übertragungswarteschlange in derselben Weise wie in einer Umgebung mit verteilter Steuerung von Warteschlangen für den anderen Warteschlangenmanager definieren.

**Anmerkung:** Anwendungen müssen in Warteschlangen schreiben, die in die Clusterübertragungswarteschlange aufgelöst werden, und sie dürfen nicht direkt in die Clusterübertragungswarteschlange schreiben.

## Automatische Definition ferner Warteschlangen

Ein WS-Manager in einem Cluster benötigt keine Definition einer fernen Warteschlange für ferne Warteschlangen im Cluster. Der Clusterwarteschlangenmanager sucht die Position einer fernen Warteschlange aus dem vollständigen Repository. Sie fügt der Nachricht Routing-Informationen hinzu und versetzt sie in die Clusterübertragungswarteschlange. WebSphere MQ erstellt automatisch eine Definition, die einer Definition einer fernen Warteschlange entspricht, sodass die Nachricht gesendet werden kann.

Eine automatisch erstellte Definition einer fernen Warteschlange kann nicht geändert oder gelöscht werden. Wenn Sie jedoch den Befehl `DISPLAY QUEUE runmqsc` mit dem Attribut `CLUSINFO` verwenden, können Sie alle lokalen Warteschlangen auf einem Warteschlangenmanager sowie alle Clusterwarteschlangen anzeigen, einschließlich Clusterwarteschlangen auf fernen Warteschlangenmanagern. Beispiel:

```
DISPLAY QUEUE(*) CLUSINFO
```

### Zugehörige Verweise

[ClusterChannelName \(MQCHAR20\)](#)

## Clusterkanäle

Sie müssen Clusterempfänger- und Clustersenderkanäle für Warteschlangenmanager in Ihrem Cluster definieren. Besondere Hinweise gelten für vollständige Repositories.

Innerhalb von Clustern werden Nachrichten zwischen Clusterwarteschlangenmanagern auf einem speziellen Kanaltyp verteilt, für den Sie Clusterempfängerkanaldefinitionen und Clustersenderkanaldefinitionen benötigen.

## Clustersenderkanal: CLUSSDR

Definieren Sie manuell einen Clustersenderkanal zu einem vollständigen Repository in jedem Warteschlangenmanager im Cluster. Die Clustersenderdefinition ermöglicht es dem Warteschlangenmanager, seinen ersten Kontakt zum Cluster herzustellen. Er benennt den Warteschlangenmanager mit vollständigem Repository, an den der Warteschlangenmanager bevorzugt Clusterinformationen sendet. Der Clustersenderkanal wird verwendet, um das Repository über Änderungen am Status des Warteschlangenmanagers zu benachrichtigen. Beispiel: Eine Warteschlange wird hinzugefügt oder entfernt. Darüber hinaus wird dieser Kanal auch zur Nachrichtenübertragung verwendet.

Die Warteschlangenmanager mit dem vollständigen Repository wiederum verfügen über Clustersenderkanäle, die auf den jeweils anderen Repository-Warteschlangenmanager verweisen. Über diese Kanäle informieren sie sich gegenseitig über Änderungen am Clusterstatus.

Es ist von geringer Bedeutung, auf welches vollständige Repository eine CLUSSDR -Kanaldefinition verweist. Sobald der erste Kontakt hergestellt wurde, werden weitere Cluster-WS-Manager-Objekte nach Bedarf automatisch definiert. Der Warteschlangenmanager kann Clusterinformationen an jedes vollständige Repository und Nachrichten an jeden Warteschlangenmanager senden.

Die CLUSSDR -Definitionen, die auf den Warteschlangenmanagern mit vollständigem Repository erstellt werden, sind besonders. Alle Aktualisierungen, die von den vollständigen Repositories ausgetauscht werden, fließen exklusiv über diese Kanäle. Der Administrator steuert das Netz der vollständigen Repositories explizit. Der Administrator muss einen CLUSSDR -Kanal von jedem Warteschlangenmanager mit vollständigem Repository zu jedem anderen Warteschlangenmanager mit vollständigem Repository im Cluster definieren. Der Administrator muss die CLUSSDR -Definitionen auf Warteschlangenmanagern mit vollständigem Repository manuell erstellen und darf sie nicht automatisch definieren.

Clustersenderkanäle dürfen nur definiert werden, um ein Teilrepository mit einem vollständigen Repository zu verbinden oder um zwei vollständige Repositories miteinander zu verbinden. Die manuelle Konfiguration eines CLUSSDR -Kanals, der ein Teilrepository oder einen Warteschlangenmanager außerhalb des Clusters adressiert, führt zu Fehlermeldungen wie AMQ9427 und AMQ9428, die ausgegeben werden.

Obwohl dies manchmal als temporäre Situation unvermeidlich sein kann (z. B. beim Ändern der Position eines vollständigen Repositories), sollten falsche Definitionen so schnell wie möglich gelöscht werden, damit diese Fehler nicht mehr ausgegeben werden.

## **Clusterempfängerkanal: CLUSRCVR**

Eine Clusterempfängerkanaldefinition definiert das Ende eines Kanals, auf dem ein Clusterwarteschlangenmanager Nachrichten von anderen Warteschlangenmanagern im Cluster empfangen kann.

Ein Clusterempfängerkanal kann auch Informationen zu den Clusterinformationen enthalten, die für das lokale Repository bestimmt sind. Durch die Definition des Clusterempfängerkanals zeigt der Warteschlangenmanager den anderen Clusterwarteschlangenmanagern an, dass er für den Empfang von Nachrichten verfügbar ist. Für jeden Clusterwarteschlangenmanager ist mindestens ein Clusterempfängerkanal erforderlich.

Eine CLUSRCVR -Definition ermöglicht anderen Warteschlangenmanagern die automatische Definition entsprechender CLUSSDR -Kanaldefinitionen.

### **Zugehörige Konzepte**

„Automatische Definition von Clusterkanälen“ auf Seite 182

Ein Warteschlangenmanager muss eine Definition für einen Clustersenderkanal haben, damit er eine Nachricht an ein fernes Ziel senden kann. Nachdem Sie einen Warteschlangenmanager in einen Cluster eingeführt haben, indem Sie seine anfänglichen CLUSSDR -und CLUSRCVR -Definitionen erstellt haben, erstellt WebSphere MQ automatisch Clustersenderkanaldefinitionen, wenn sie benötigt werden. Automatisch definierte Clustersenderkanäle können nicht geändert werden. Sie können ihr Verhalten mithilfe eines Exits für automatische Kanaldefinition ändern.

### **Automatische Definition von Clusterkanälen**

Ein Warteschlangenmanager muss eine Definition für einen Clustersenderkanal haben, damit er eine Nachricht an ein fernes Ziel senden kann. Nachdem Sie einen Warteschlangenmanager in einen Cluster eingeführt haben, indem Sie seine anfänglichen CLUSSDR -und CLUSRCVR -Definitionen erstellt haben, erstellt WebSphere MQ automatisch Clustersenderkanaldefinitionen, wenn sie benötigt werden. Automatisch definierte Clustersenderkanäle können nicht geändert werden. Sie können ihr Verhalten mithilfe eines Exits für automatische Kanaldefinition ändern.

Wenn die Clustersenderseite und die Clusterempfängerseite eines Kanals definiert sind, wird der Kanal gestartet. Ein automatisch definierter Kanal bleibt aktiv, bis er nicht mehr benötigt wird, und wird mithilfe der normalen Unterbrechungsintervallregeln beendet.

Automatisch definierte Clustersenderkanäle erhalten ihre Attribute von der jeweiligen Definition für den entsprechenden Clusterempfängerkanal im Empfangswarteschlangenmanager. Selbst wenn ein manuell definierter Clustersenderkanal vorhanden ist, werden seine Attribute automatisch geändert, um sicherzustellen, dass sie der entsprechenden Clusterempfängerdefinition entsprechen. Angenommen, Sie definieren beispielsweise einen CLUSRCVR ohne Angabe einer Portnummer im Parameter CONNAME und definieren manuell einen CLUSSDR, der eine Portnummer angibt. Wenn der automatisch definierte CLUSSDR den manuell definierten Clustersenderkanal ersetzt, wird die Portnummer (vom CLUSRCVR übernommen) leer. Es wird die Standardportnummer verwendet und der Kanal fällt aus.

Sie können eine automatisch definierte Clustersenderdefinition nicht ändern.

Mit dem Befehl `DISPLAY CHANNEL runmqsc` können keine automatisch definierten Kanäle angezeigt werden. Verwenden Sie den folgenden Befehl, um die automatisch definierten Kanäle anzuzeigen:

```
DISPLAY CLUSQMGR(qMgrName)
```



Verwenden Sie den folgenden Befehl, um den Status des automatisch definierten CLUSSDR -Kanals anzuzeigen, der der erstellten CLUSRCVR -Kanaldefinition entspricht:

```
DISPLAY CHSTATUS(channelname)
```

Sie können den WebSphere MQ -Exit für die automatische Kanaldefinition verwenden, wenn Sie ein Benutzerexitprogramm schreiben möchten, um einen Clustersenderkanal oder Clusterempfängerkanal anzupassen. Sie können den Exit für automatische Kanaldefinition in einer Clusterumgebung für folgende Zwecke verwenden:

- Anpassung von Kommunikationsdefinitionen, d. h. SNA- LU6.2 -Namen
- Add or remove other exits, for example, security exits
- Ändern Sie die Namen von Kanalexits. Sie müssen den Namen eines CLUSSDR -Kanalexits ändern, weil der Name des CLUSSDR -Kanalexits automatisch aus der CLUSRCVR -Kanaldefinition generiert wird. Der automatisch generierte Name ist möglicherweise falsch und fast sicher falsch, wenn sich die beiden Kanäle auf verschiedenen Plattformen befinden. Das Format der Exitnamen ist auf verschiedenen Plattformen unterschiedlich. Beispiel: unter Windows hingegen `SCYEXIT('drive:\path\library(secexit)')`.

Exitnamen auf anderen Plattformen als z/OS haben das allgemeine Format *Pfad/Bibliothek(Funktion)*. Wenn *function* vorhanden ist, werden bis zu acht Zeichen für diese Zeichen verwendet. Andernfalls wird die auf acht Zeichen abgeschnittene *Bibliothek* verwendet. Zum Beispiel:

- `/var/mqm/exits/myExit.so(MsgExit)` konvertiert in MSGEXIT
- `/var/mqm/exits/myExit` konvertiert in MYEXIT
- `/var/mqm/exits/myExit.so(ExitLongName)` konvertiert in EXITLONG

Verwenden Sie das Kanalattribut LOCLADDR, um einen abgehenden (TCP-) Kanal für die Verwendung einer bestimmten IP-Adresse, eines bestimmten Ports oder eines bestimmten Portbereichs zu aktivieren. LOCLADDR ist nützlich, wenn Sie über mehr als eine Netz Karte verfügen und wenn ein Kanal einen bestimmten Kanal für abgehende Kommunikation verwenden soll.

Wenn Sie eine virtuelle IP-Adresse in CLUSSDR-Kanälen angeben möchten, verwenden Sie die IP-Adresse aus dem LOCLADDR-Kanal in einem manuell definierten CLUSSDR. Wenn Sie den Portbereich angeben möchten, verwenden Sie den Portbereich aus dem CLUSRCVR.

Wenn ein Cluster LOCLADDR verwenden muss, um die Kanäle für abgehende Kommunikation an eine bestimmte IP-Adresse zu binden, müssen Sie einen Exit für automatische Kanaldefinition schreiben, um den Wert für LOCLADDR in einem ihrer automatisch definierten CLUSSDR -Kanäle zu erzwingen, und Sie müssen ihn im manuell definierten CLUSSDR -Kanal angeben.

Geben Sie im Feld LOCLADDR des Kanals CLUSRCVR keine IP-Adresse an, es sei denn, alle Warteschlangenmanager befinden sich auf demselben Server. Die IP-Adresse LOCLADDR wird an die automatisch definierten CLUSSDR-Kanäle aller Warteschlangenmanager weitergegeben, die über den Kanal CLUSRCVR verbunden sind.

Geben Sie eine Portnummer oder einen Portbereich in den LOCLADDR eines CLUSRCVR -Kanals ein, wenn alle Warteschlangenmanager in einem Cluster einen bestimmten Port oder Portbereich für ihre gesamte abgehende Kommunikation verwenden sollen

**distributed** Auf verteilten Plattformen kann ein Standardwert als lokale Adresse gesetzt werden, der für alle Senderkanäle übernommen wird, für die keine lokale Adresse definiert ist. Der Standardwert wird definiert, indem die Umgebungsvariable MQ\_LCLADDR vor dem Starten des Warteschlangenmanagers festgelegt wird. Das Format dieses Werts entspricht dem des MQSC-Attributs LOCLADDR.

Automatisch definierte Clustersenderkanaldefinitionen sind keine realen Kanalobjekte. Auf anderen Plattformen als z/OS ist dem OAM (Object Authority Manager, Objektberechtigungsmanager) ihre Existenz nicht bekannt. Wenn Sie versuchen, Start-, Stopp-, Ping-, Zurücksetzungs- oder Auflösungsbefehle auf automatisch definierten Clustersenderkanälen auszugeben, prüft der OAM, ob Sie berechtigt sind, dieselbe Aktion auf dem Clusterempfängerkanal für den Cluster auszuführen.

Wenn der Cluster mit PROPCTL Anwendungsheader wie RFH2 aus Nachrichten entfernen muss, die von einem WebSphere MQ Version 7-Warteschlangenmanager an einen Warteschlangenmanager einer früheren Version von WebSphere MQ gesendet werden, müssen Sie einen Exit für automatische Kanaldefinition schreiben, der PROPCTL auf den Wert NONE setzt. Der Exit ist erforderlich, da Clustersenderkanäle ihre Definition auf den entsprechenden Clusterempfängerkanälen basieren. Da der Clusterempfängerkanal der früheren Version kein Attribut PROPCTL hat, wird das Attribut vom automatischen Clustersenderkanal auf COMPAT gesetzt. Das Attribut wird unabhängig von der Einstellung im manuellen Clustersenderkanal auf COMPAT gesetzt.

### **Zugehörige Verweise**

[Lokale Adresse \(LOCLADDR\)](#)

### **Standardclusterobjekte**

Erstellen Sie die Standardclusterobjekte, wenn Sie WebSphere MQ -Cluster verwenden. Sie sind in der Gruppe der Standardobjekte enthalten, die automatisch erstellt werden, wenn Sie einen WS-Manager definieren.

Sie können die Standardkanaldefinitionen auf dieselbe Weise wie jede andere Kanaldefinition durch die Ausführung von WebSphere MQ-Scriptbefehlen oder PCF-Befehlen ändern.

Ändern Sie außer für SYSTEM . CLUSTER . HISTORY . QUEUE keine Standardwarteschlangendefinitionen.

#### **SYSTEM . CLUSTER . COMMAND . QUEUE**

Jeder Warteschlangenmanager in einem Cluster besitzt eine lokale Warteschlange namens SYSTEM . CLUSTER . COMMAND . QUEUE, über die Nachrichten an das vollständige Repository übertragen werden. Die Nachrichten enthalten neue oder geänderte Informationen über den Warteschlangenmanager selbst oder Anforderungen für Informationen über andere Warteschlangenmanager. SYSTEM . CLUSTER . COMMAND . QUEUE ist normalerweise leer.

#### **SYSTEM . CLUSTER . HISTORY . QUEUE**

Jeder Warteschlangenmanager in einem Cluster besitzt eine lokale Warteschlange namens SYSTEM . CLUSTER . HISTORY . QUEUE. In SYSTEM . CLUSTER . HISTORY . QUEUE wird zu Servicezwecken das Protokoll der Clusterstatusinformationen gespeichert.

In den Standardobjekteinstellungen ist SYSTEM . CLUSTER . HISTORY . QUEUE auf PUT(ENABLED) gesetzt. Um die Erfassung von Protokoll Daten zu unterdrücken, ändern Sie die Einstellung in PUT(DISABLED).

#### **SYSTEM . CLUSTER . REPOSITORY . QUEUE**

Jeder Warteschlangenmanager in einem Cluster besitzt eine lokale Warteschlange namens SYSTEM . CLUSTER . REPOSITORY . QUEUE. In dieser Warteschlange werden alle Informationen über das vollständige Repository gespeichert. Sie ist normalerweise nicht leer.

#### **SYSTEM . CLUSTER . TRANSMIT . QUEUE**

Jeder Warteschlangenmanager verfügt über eine Definition für eine lokale Warteschlange namens SYSTEM . CLUSTER . TRANSMIT . QUEUE. SYSTEM . CLUSTER . TRANSMIT . QUEUE ist die Standardübertragungswarteschlange für alle Nachrichten an alle Warteschlangen und Warteschlangenmanager innerhalb von Clustern. Sie können die Standardübertragungswarteschlange für jeden Clustersenderkanal in SYSTEM . CLUSTER . TRANSMIT . *ChannelName* ändern, indem Sie das Warteschlangenmanagerattribut DEFXMITQ. Die Warteschlange SYSTEM . CLUSTER . TRANSMIT . QUEUE kann nicht gelöscht werden. Es wird auch verwendet, um Berechtigungsprüfungen zu definieren, ob die verwendete Standardübertragungswarteschlange SYSTEM . CLUSTER . TRANSMIT . QUEUE oder SYSTEM . CLUSTER . TRANSMIT . *ChannelName* ist.

#### **SYSTEM . DEF . CLUSRCVR**

In jedem Cluster gibt es eine Standarddefinition für den Clusterempfängerkanal (CLUSRCVR) mit dem Namen SYSTEM . DEF . CLUSRCVR. SYSTEM . DEF . CLUSRCVR liefert Standardwerte für alle Attribute, die Sie bei der Erstellung eines Clusterempfängerkanals für einen Warteschlangenmanager im Cluster nicht angeben.

#### **SYSTEM . DEF . CLUSSDR**

In jedem Cluster gibt es eine Standarddefinition für den Clustersenderkanal (CLUSSDR) mit dem Namen SYSTEM . DEF . CLUSSDR. SYSTEM . DEF . CLUSSDR liefert Standardwerte für alle Attribute, die



Sie bei der Erstellung eines Clustersenderkanals für einen Warteschlangenmanager im Cluster nicht angeben.

## Clusterübertragungswarteschlangen und Clustersenderkanäle

Nachrichten zwischen Clustering-WS-Managern werden in Clusterübertragungswarteschlangen gespeichert und von Clustersenderkanälen weitergeleitet.

Wenn Sie den Clustersenderkanal anzeigen, sehen Sie, dass er einer Übertragungswarteschlange zugeordnet ist. Zu einem beliebigen Zeitpunkt ist ein Clustersenderkanal einer Übertragungswarteschlange zugeordnet. Wenn Sie die Konfiguration des Kanals ändern, kann es beim nächsten Start zu einer anderen Übertragungswarteschlange wechseln. Führen Sie den folgenden MQSC-Befehl aus, um die Übertragungswarteschlangen anzuzeigen, denen Clustersenderkanäle zugeordnet sind:

```
DISPLAY CHSTATUS(*) WHERE(CHLTYPE EQ CLUSSDR)
```

```
AMQ8417: Display Channel Status details.
CHANNEL(TO.QM2) CHLTYPE(CLUSSDR)
CONNAME(9.146.163.190(1416)) CURRENT
QMNAME(QM2) STATUS(STOPPED)
SUBSTATE() XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

Die Übertragungswarteschlange, die im gespeicherten Kanalstatus eines gestoppten Clustersenderkanals angezeigt wird, kann sich ändern, wenn der Kanal erneut gestartet wird. „[Auswahl von Standardübertragungswarteschlangen nach Clustersenderkanälen](#)“ auf Seite 186 beschreibt den Prozess der Auswahl einer Standardübertragungswarteschlange; „[Auswahl von manuell definierten Übertragungswarteschlangen nach Clustersenderkanälen](#)“ auf Seite 186 beschreibt den Prozess der Auswahl einer manuell definierten Übertragungswarteschlange.

Wenn ein Clustersenderkanal gestartet wird, überprüft er seine Zuordnung zu Übertragungswarteschlangen. Wenn die Konfiguration von Übertragungswarteschlangen oder die Standardwerte des Warteschlangenmanagers geändert werden, kann er den Kanal möglicherweise mit einer anderen Übertragungswarteschlange verknüpft werden. Wenn der Kanal aufgrund einer Konfigurationsänderung mit einer anderen Übertragungswarteschlange neu gestartet wird, findet ein Prozess der Übertragung von Nachrichten an die neu zugeordnete Übertragungswarteschlange statt. In „[Funktionsweise des Prozesses zum Wechseln des Clustersenderkanals in eine andere Übertragungswarteschlange](#)“ auf Seite 187 wird der Prozess der Übertragung eines Clustersenderkanals von einer Übertragungswarteschlange in eine andere beschrieben.

Das Verhalten von Clustersenderkanälen unterscheidet sich von Sender- und Serverkanälen. Sie bleiben der gleichen Übertragungswarteschlange zugeordnet, bis das Kanalattribut **XMITQ** geändert wird. Wenn Sie das Attribut für die Übertragungswarteschlange auf einem Sender- oder Serverkanal ändern und ihn erneut starten, werden die Nachrichten nicht von der alten Übertragungswarteschlange in die neue übertragen.

Ein weiterer Unterschied zwischen Clustersenderkanälen und Sender- oder Serverkanälen besteht darin, dass mehrere Clustersenderkanäle eine Clusterübertragungswarteschlange öffnen können, aber nur ein Sender- oder Serverkanal eine normale Übertragungswarteschlange öffnen kann. Bis Version 7.5 -Clusterverbindungen die einzelne Clusterübertragungswarteschlange gemeinsam nutzen, SYSTEM.CLUSTER.TRANSMIT.QUEUE. Ab Version 7.5 haben Sie die Option, dass Clustersenderkanäle keine Übertragungswarteschlangen gemeinsam nutzen. Die Exklusivität wird nicht erzwungen; sie ist ein Ergebnis der Konfiguration. Sie können den Pfad einer Nachricht in einem Cluster so konfigurieren, dass er keine Übertragungswarteschlangen oder Kanäle mit Nachrichten gemeinsam nutzt, die zwischen anderen Anwendungen fließen. Weitere Informationen hierzu finden Sie in den Abschnitten „[Clustering: Planung der Konfiguration von Clusterübertragungswarteschlangen](#)“ auf Seite 310 und „[Cluster und Cluster-Übertragungswarteschlange hinzufügen, um den Datenverkehr der Clusternachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet werden](#)“ auf Seite 224.

## Auswahl von Standardübertragungswarteschlangen nach Clustersenderkanälen

Eine Clusterübertragungswarteschlange ist entweder eine Systemstandardwarteschlange mit einem Namen, der mit `SYSTEM.CLUSTER.TRANSMIT` beginnt, oder eine manuell definierte Warteschlange. Ein Clustersenderkanal wird einer Clusterübertragungswarteschlange auf eine der beiden folgenden Arten zugeordnet: durch den Standardmechanismus der Clusterübertragungswarteschlange oder durch manuelle Konfiguration.

Die standardmäßige Clusterübertragungswarteschlange wird als WS-Manager-Attribut **DEFCLXQ** festgelegt. Der Wert ist entweder `SCTQ` oder `CHANNEL`. Neue und migrierte WS-Manager werden auf `SCTQ` gesetzt. Sie können den Wert in `CHANNEL` ändern.

Wenn `SCTQ` festgelegt ist, ist die Standard-Cluster-Übertragungswarteschlange `SYSTEM.CLUSTER.TRANSMIT.QUEUE`. Jeder Clustersenderkanal kann diese Warteschlange öffnen. Die Clustersenderkanäle, die die Warteschlange öffnen, sind diejenigen, die nicht mit manuell definierten Clusterübertragungswarteschlangen verknüpft sind.

Wenn `CHANNEL` festgelegt ist, kann der Warteschlangenmanager für jeden Clustersenderkanal eine separate permanente dynamische Übertragungswarteschlange erstellen. Jede Warteschlange hat den Namen `SYSTEM.CLUSTER.TRANSMIT.ChannelName` und wird aus der Modellwarteschlange `SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE` erstellt. Jedem Clustersenderkanal, der keiner manuell definierten Clusterübertragungswarteschlange zugeordnet ist, wird eine permanentdynamische Clusterübertragungswarteschlange zugeordnet. Die Warteschlange wird vom Warteschlangenmanager erstellt, wenn er eine separate Clusterübertragungswarteschlange für das Clusterziel benötigt, das von diesem Clustersenderkanal bedient wird, und es ist keine Warteschlange vorhanden.

Einige Clusterziele können von Clustersenderkanälen, die manuell definierten Übertragungswarteschlangen zugeordnet sind, und anderen durch die Standardwarteschlange oder-warteschlangen bedient werden. In der Zuordnung von Clustersenderkanälen mit Übertragungswarteschlangen haben die manuell definierten Übertragungswarteschlangen immer Vorrang vor den Standardübertragungswarteschlangen.

Die Rangfolge der Clusterübertragungswarteschlangen ist in [Abbildung 24](#) auf Seite 186 dargestellt. Der einzige Clustersenderkanal, der keiner manuell definierten Clusterübertragungswarteschlange zugeordnet ist, ist `CS.QM1`. Sie ist keiner manuell definierten Übertragungswarteschlange zugeordnet, da keiner der Kanalnamen im Attribut **CLCHNAME** der Übertragungswarteschlangen mit `CS.QM1` übereinstimmt.

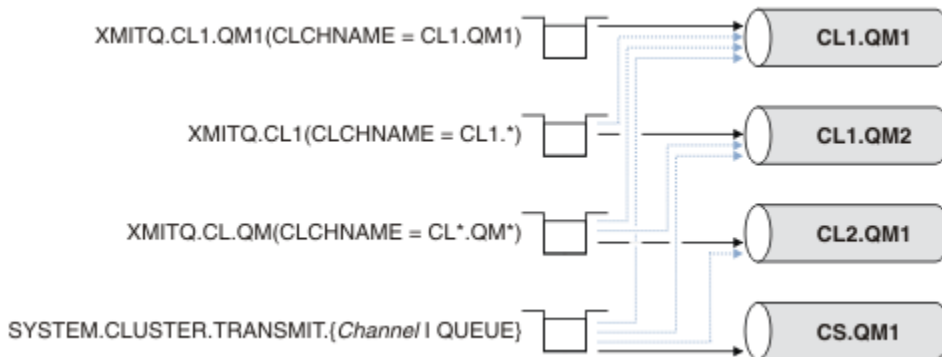


Abbildung 24. Übertragungswarteschlange/Clustersenderkanalvorrangstellung

## Auswahl von manuell definierten Übertragungswarteschlangen nach Clustersenderkanälen

Eine manuell definierte Warteschlange hat das Attribut **USAGE** für die Übertragungswarteschlange, das auf `XMITQ` gesetzt ist, und das Attribut "Clusterkanalname" **CLCHNAME** auf einen bestimmten oder generischen Kanalnamen gesetzt.

Wenn der Name im Warteschlangenattribut von **CLCHNAME** mit einem Clustersenderkanalnamen übereinstimmt, wird der Kanal der Warteschlange zugeordnet. Der Name ist entweder eine exakte Übereinstim-

mung, wenn der Name keine Platzhalterzeichen enthält, oder es ist die beste Übereinstimmung, wenn der Name Platzhalterzeichen enthält.

Wenn **CLCHNAME** -Definitionen in mehreren Übertragungswarteschlangen mit demselben Clustersenderkanal übereinstimmen, werden die Definitionen als überlappend definiert. Um die Mehrdeutigkeit zu beheben, gibt es eine Rangfolge zwischen Übereinstimmungen. Exakte Übereinstimmungen haben immer Vorrang. [Abbildung 24](#) auf Seite 186 zeigt Zuordnungen zwischen Übertragungswarteschlangen und Clustersenderkanälen. Die schwarzen Pfeile zeigen die tatsächlichen Assoziationen und die grauen Pfeile, mögliche Assoziationen an. Die Prioritätsreihenfolge der Übertragungswarteschlangen in [Abbildung 24](#) auf Seite 186 lautet:

#### **XMITQ.CL1.QM1**

Für die Übertragungswarteschlange XMITQ.CL1.QM1 ist das Attribut **CLCHNAME** auf CL1.QM1 festgelegt. Die Definition des Attributs **CLCHNAME**, CL1.QM1, weist keine Platzhalterzeichen auf und hat Vorrang vor allen anderen CLCHNAME -Attributen, die in anderen Übertragungswarteschlangen definiert sind, die mit Platzhalterzeichen übereinstimmen. Der WS-Manager speichert alle Clusternachrichten, die vom CL1.QM1 -Clustersenderkanal in der Übertragungswarteschlange von XMITQ.CL1.QM1 übertragen werden sollen. Die einzige Ausnahme ist, wenn für mehrere Übertragungswarteschlangen das Attribut **CLCHNAME** auf CL1.QM1 gesetzt ist. In diesem Fall speichert der Warteschlangenmanager Nachrichten für den Clustersenderkanal von CL1.QM1 in jeder dieser Warteschlangen. Er wählt eine Warteschlange willkürlich aus, wenn der Kanal gestartet wird. Es kann eine andere Warteschlange auswählen, wenn der Kanal erneut gestartet wird.

#### **XMITQ.CL1**

Für die Übertragungswarteschlange XMITQ.CL1 ist das Attribut **CLCHNAME** auf CL1.\* festgelegt. Die Definition des Attributs **CLCHNAME** (CL1.\*) weist ein abschließendes Platzhalterzeichen auf, das mit dem Namen eines beliebigen Clustersenderkanals übereinstimmt, der mit CL1. beginnt. Der Warteschlangenmanager speichert jede Clusternachricht, die von einem beliebigen Clustersenderkanal übertragen werden soll, dessen Name mit CL1. in der Übertragungswarteschlange XMITQ.CL1 beginnt, es sei denn, es gibt eine Übertragungswarteschlange mit einer spezifischeren Übereinstimmung, wie z. B. die Warteschlange XMITQ.CL1.QM1. Ein abschließendes Platzhalterzeichen macht die Definition weniger spezifisch als eine Definition ohne Platzhalterzeichen und genauer als eine Definition mit mehreren Platzhaltern oder Platzhalterzeichen, auf die weitere abschließende Zeichen folgen.

#### **XMITQ.CL.QM**

XMITQ.CL.QM ist der Name der Übertragungswarteschlange, in der das Attribut **CLCHNAME** auf CL\*.QM\* gesetzt ist. Die Definition von CL\*.QM\* weist zwei Platzhalterzeichen auf, die mit dem Namen eines beliebigen Clustersenderkanals übereinstimmen, der mit CL. beginnt, und enthält oder endet mit QM. Die Übereinstimmung ist weniger spezifisch als eine Übereinstimmung mit einem Platzhalterzeichen.

#### **SYSTEM.CLUSTER.TRANSMIT.channelName | QUEUE**

Wenn für keine Übertragungswarteschlange ein Attribut **CLCHNAME** vorhanden ist, das mit dem Namen des Clustersenderkanals übereinstimmt, den der Warteschlangenmanager verwenden soll, verwendet der Warteschlangenmanager die Standardwarteschlange für Clusterübertragungen. Die Standardclusterübertragungswarteschlange ist entweder die Einzelsystemclusterübertragungswarteschlange SYSTEM.CLUSTER.TRANSMIT.QUEUE oder eine Systemclusterübertragungswarteschlange, die der Warteschlangenmanager für einen bestimmten Clustersenderkanal SYSTEM.CLUSTER.TRANSMIT.channelName erstellt hat. Welche Warteschlange der Standardwert ist, hängt von der Einstellung des Attributs des Warteschlangenmanagers **DEFXMITQ** ab.

**Tipp:** Sofern Sie keinen eindeutigen Bedarf an überlappenden Definitionen haben, vermeiden Sie sie, da sie zu komplizierten Konfigurationen führen können, die schwer zu verstehen sind.

## **Funktionsweise des Prozesses zum Wechseln des Clustersenderkanals in eine andere Übertragungswarteschlange**

Wenn Sie die Zuordnung von Clustersenderkanälen zu Clusterübertragungswarteschlangen ändern möchten, ändern Sie den Parameter **CLCHNAME** einer beliebigen Übertragungswarteschlange oder des Warteschlangenmanagerparameters **DEFCLXQ** jederzeit. Nichts passiert sofort. Änderungen treten nur auf, wenn ein Kanal gestartet wird. Wenn er gestartet wird, prüft er, ob Nachrichten aus derselben Übertra-

gungswarteschlange weitergesendet werden. Drei Arten von Änderungen ändern die Zuordnung eines Clustersenderkanals mit einer Übertragungswarteschlange.

1. Wenn der Parameter **CLCHNAME** der Übertragungswarteschlange neu definiert wird, wird der Clustersenderkanal momentan weniger spezifisch oder leer sein, oder die Clusterübertragungswarteschlange wird gelöscht, wenn der Kanal gestoppt wird.

Eine andere Clusterübertragungswarteschlange könnte jetzt eine bessere Übereinstimmung für den Kanalnamen sein. Oder, wenn keine anderen Übertragungswarteschlangen mit dem Namen des Clustersenderkanals übereinstimmen, muss die Zuordnung auf die Standardübertragungswarteschlange zurückgesetzt werden.

2. Den Parameter **CLCHNAME** einer anderen Clusterübertragungswarteschlange neu definieren oder eine Clusterübertragungswarteschlange hinzufügen.

Der Parameter **CLCHNAME** einer anderen Übertragungswarteschlange könnte jetzt eine bessere Übereinstimmung für den Clustersenderkanal sein als die Übertragungswarteschlange, der der Clustersenderkanal momentan zugeordnet ist. Wenn der Clustersenderkanal momentan einer Standard-Cluster-Übertragungswarteschlange zugeordnet ist, kann er einer manuell definierten Clusterübertragungswarteschlange zugeordnet werden.

3. Wenn der Clustersenderkanal momentan einer Standard-Cluster-Übertragungswarteschlange zugeordnet ist, ändern Sie den Parameter des **DEFCLXQ** -Warteschlangenmanagers.

Wenn sich die Zuordnung eines Clustersenderkanals ändert, wechselt der Kanal, wenn der Kanal gestartet wird, seine Zuordnung zu der neuen Übertragungswarteschlange. Während des Switchs wird sichergestellt, dass keine Nachrichten verloren gehen. Nachrichten werden in die neue Übertragungswarteschlange in der Reihenfolge übertragen, in der der Kanal die Nachrichten an den fernen WS-Manager übertragen würde.

**Hinweis:** Bei der gemeinsamen Weiterleitung von Nachrichten in einem Cluster müssen Sie Nachrichten in Gruppen einordnen, um sicherzustellen, dass Nachrichten, die in der Reihenfolge zugestellt werden müssen, in der Reihenfolge zugestellt werden. In seltenen Fällen können Nachrichten in einem Cluster nicht mehr in Ordnung sein.

Der Switchprozess durchläuft die folgenden transaktionsorientierten Schritte. Wenn der Switch-Prozess unterbrochen wird, wird der aktuelle transaktionsorientierte Schritt wieder aufgenommen, wenn der Kanal erneut gestartet wird.

### **Schritt 1-Prozessnachrichten aus der ursprünglichen Übertragungswarteschlange**

Der Clustersenderkanal ist der neuen Übertragungswarteschlange zugeordnet, die er möglicherweise mit anderen Clustersenderkanälen gemeinsam nutzen kann. Nachrichten für den Clustersenderkanal werden weiterhin in die ursprüngliche Übertragungswarteschlange gestellt. Ein Übergangsschaltprozess überträgt Nachrichten aus der ursprünglichen Übertragungswarteschlange in die neue Übertragungswarteschlange. Der Clustersenderkanal leitet die Nachrichten von der neuen Übertragungswarteschlange an den Clusterempfängerkanal weiter. Der Kanalstatus zeigt den Clustersenderkanal an, der immer noch der alten Übertragungswarteschlange zugeordnet ist.

Der Switch-Prozess überträgt auch weiterhin neu eingekommene Nachrichten. Dieser Schritt wird fortgesetzt, bis die Anzahl der verbleibenden Nachrichten, die durch den Schaltvorgang weitergeleitet werden sollen, den Wert Null erreicht. Wenn die Anzahl der Nachrichten null erreicht, wird die Prozedur in Schritt 2 verschoben.

Während Schritt 1 wird die Plattenaktivität für den Kanal erhöht. Persistente Nachrichten werden von der ersten Übertragungswarteschlange und von der zweiten Übertragungswarteschlange aus festgeschrieben. Diese Plattenaktivität wird zusätzlich zu den Nachrichten, die festgeschrieben werden, wenn sie in die Übertragungswarteschlange gestellt und aus der Übertragungswarteschlange entfernt werden, als Teil der normalen Übertragung der Nachrichten festgeschrieben. Im Idealfall kommen während des Schaltvorganges keine Meldungen ein, so daß der Übergang so schnell wie möglich erfolgen kann. Wenn Nachrichten ankommen, werden sie vom Switch-Prozess verarbeitet.

## Schritt 2-Prozessnachrichten aus der neuen Übertragungswarteschlange

Sobald keine Nachrichten in der ursprünglichen Übertragungswarteschlange für den Clustersenderkanal verbleiben, werden neue Nachrichten direkt in die neue Übertragungswarteschlange gestellt. Der Kanalstatus zeigt den Clustersenderkanal an, der der neuen Übertragungswarteschlange zugeordnet ist. Die folgende Nachricht wird in das Fehlerprotokoll des Warteschlangenmanagers geschrieben:  
"AMQ7341 Die Übertragungswarteschlange für Kanal *ChannelName* ist *QueueName*."

## Mehrere Clusterübertragungswarteschlangen und Clusterübertragungswarteschlangen-Attribute

Es besteht die Möglichkeit, Clusternachrichten an verschiedene Warteschlangenmanager weiterzuleiten, die die Nachrichten in einer einzelnen Clusterübertragungswarteschlange oder in mehreren Warteschlangen speichern. Mit einer Warteschlange verfügen Sie über eine Gruppe von Attributen für Clusterübertragungswarteschlangen, die festgelegt und abgefragt werden können, mit mehreren Warteschlangen über mehrere Gruppen. Bei einigen Attributen ist die Verwendung mehrerer Gruppen ein Vorteil: z. B. die Warteschlangenlänge abfragen, wie viele Nachrichten darauf warten, von einem oder einer Gruppe von Kanälen weitergeleitet zu werden, und nicht von allen Kanälen. Für andere Attribute ist die Verwendung mehrerer Gruppen ein Nachteil: Sie möchten beispielsweise wahrscheinlich nicht dieselben Zugriffsberechtigungen für jede Clusterübertragungswarteschlange konfigurieren. Aus diesem Grund werden Zugriffsberechtigungen immer auf das Profil für SYSTEM.CLUSTER.TRANSMIT.QUEUE und nicht auf Profile für eine bestimmte Clusterübertragungswarteschlange überprüft. Wenn Sie differenziertere Sicherheitsüberprüfungen anwenden möchten, lesen Sie den Abschnitt „Zugriffssteuerung und mehrere Clusterübertragungswarteschlangen“ auf Seite 173.

## Mehrere Clustersenderkanäle und mehrere Übertragungswarteschlangen

Ein WS-Manager speichert eine Nachricht in einer Clusterübertragungswarteschlange, bevor er sie an einen Clustersenderkanal weiterleitet. Er wählt einen Clustersenderkanal aus, der mit dem Ziel für die Nachricht verbunden ist. Es kann eine Auswahl an Clustersenderkanälen geben, die alle mit derselben Zieladresse verbunden sind. Das Ziel kann die gleiche physische Warteschlange sein, die über mehrere Clustersenderkanäle mit einem einzigen Warteschlangenmanager verbunden ist. Die Zieladresse kann auch viele physische Warteschlangen mit demselben Warteschlangennamen sein, die sich auf verschiedenen Warteschlangenmanagern in demselben Cluster befinden. Wenn eine Auswahl von Clustersenderkanälen ist, die mit einem Ziel verbunden sind, wählt der Lastausgleichsalgorithmus einen aus. Die Auswahl hängt von einer Reihe von Faktoren ab; siehe Algorithmus für Clusterauslastungsmanagement.

In Abbildung 25 auf Seite 190 sind CL1.QM1, CL1.QM2 und CS.QM1 alle Kanäle, die zu demselben Ziel führen können. Wenn Sie beispielsweise Q1 in CL1 auf QM1 und QM2 definieren, stellen CL1.QM1 und CL1.QM2 beide Routen zu demselben Ziel, Q1, auf zwei verschiedenen Warteschlangenmanagern bereit. Wenn sich der Kanal CS.QM1 auch in CL1 befindet, ist es auch ein Kanal, den eine Nachricht für Q1 annehmen kann. Die Clusterzugehörigkeit von CS.QM1 kann durch eine Clusternamensliste definiert werden. Aus diesem Grund enthält der Kanalname keinen Clusternamen in seiner Konstruktion. Abhängig von den Parametern für den Lastausgleich und der sendenden Anwendung werden möglicherweise einige Nachrichten für Q1 in jeder der Übertragungswarteschlangen, XMITQ.CL1.QM1, XMITQ.CL1 und SYSTEM.CLUSTER.TRANSMIT.CS.QM1, platziert.

Wenn Sie den Nachrichtenverkehr so trennen möchten, dass Nachrichten für dasselbe Ziel keine Warteschlangen oder Kanäle mit Nachrichten für verschiedene Ziele gemeinsam nutzen, müssen Sie überlegen, wie Sie den Datenverkehr zunächst auf verschiedene Clustersenderkanäle aufteilen und dann die Nachrichten für einen bestimmten Kanal auf eine andere Übertragungswarteschlange aufteilen. Clusterwarteschlangen in demselben Cluster, auf demselben Warteschlangenmanager, verwenden normalerweise dieselben Clusterkanäle. Das Definieren mehrerer Cluster-Übertragungswarteschlangen allein reicht nicht aus, um den Clusternachrichtenverkehr in verschiedene Warteschlangen zu trennen. Wenn Sie Nachrichten nicht für verschiedene Zielwarteschlangen auf unterschiedlichen Kanälen voneinander trennen, verwenden die Nachrichten dieselbe Clusterübertragungswarteschlange.

Eine einfache Möglichkeit, die Kanäle zu trennen, die Nachrichten annehmen, besteht darin, mehrere Cluster zu erstellen. Definieren Sie auf jedem WS-Manager in jedem Cluster nur eine Clusterwarteschlange. Wenn Sie dann einen anderen Clusterempfängerkanal für jede Cluster-/WS-Manager-Kombination

definieren, verwenden die Nachrichten für jede Clusterwarteschlange keinen Clusterkanal mit Nachrichten für andere Clusterwarteschlangen. Wenn Sie separate Übertragungswarteschlangen für die Clusterkanäle definieren, speichert der sendende Warteschlangenmanager Nachrichten für nur eine Clusterwarteschlange in jeder Übertragungswarteschlange. Wenn beispielsweise zwei Clusterwarteschlangen Ressourcen nicht gemeinsam nutzen sollen, können Sie sie entweder in verschiedenen Clustern desselben Warteschlangenmanagers oder auf verschiedenen Warteschlangenmanagern in demselben Cluster platzieren.

Die Auswahl der Clusterübertragungswarteschlange wirkt sich nicht auf den Algorithmus für die Lastverteilung aus. Der Algorithmus für den Lastausgleich wählt den Clustersenderkanal aus, um eine Nachricht weiterzuleiten. Sie stellt die Nachricht in die Übertragungswarteschlange, die von diesem Kanal bedient wird. Wenn der Algorithmus für den Lastausgleich erneut aufgerufen werden soll, z. B. wenn der Kanal gestoppt wird, kann er möglicherweise einen anderen Kanal auswählen, um die Nachricht weiterzuleiten. Wenn er einen anderen Kanal wählt und der neue Kanal Nachrichten aus einer anderen Clusterübertragungswarteschlange weiterleitet, überträgt der Lastausgleichsalgorithmus die Nachricht an die andere Übertragungswarteschlange.

In [Abbildung 25 auf Seite 190](#) werden zwei Clustersenderkanäle, CS.QM1 und CS.QM2, der Standard-systemübertragungswarteschlange zugeordnet. Wenn der Lastausgleichsalgorithmus eine Nachricht in SYSTEM.CLUSTER.TRANSMIT.QUEUE oder einer anderen Clusterübertragungswarteschlange speichert, wird der Name des Clustersenderkanals, der die Nachricht weiterleiten soll, in der Korrelations-ID der Nachricht gespeichert. Jeder Kanal leitet nur die Nachrichten weiter, die mit der Korrelations-ID mit dem Kanalnamen übereinstimmen.

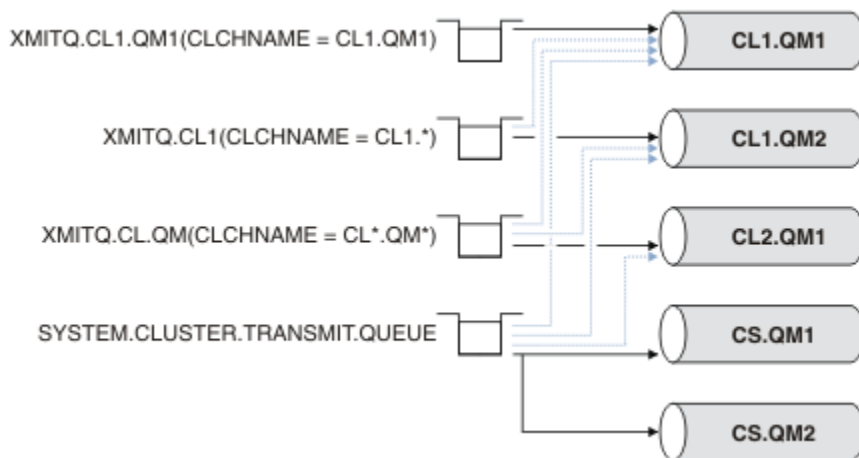


Abbildung 25. Mehrere Clustersenderkanäle

Wenn CS.QM1 stoppt, werden die Nachrichten in der Übertragungswarteschlange für diesen Clustersenderkanal überprüft. Die Nachrichten, die von einem anderen Kanal weitergeleitet werden können, werden durch den Algorithmus für die Lastverteilung erneut verarbeitet. Ihre Korrelations-ID wird auf einen alternativen Clustersenderkanalnamen zurückgesetzt. Wenn es sich bei dem alternativen Clustersenderkanal um CS.QM2 handelt, bleibt die Nachricht in SYSTEM.CLUSTER.TRANSMIT.QUEUE. Wenn der alternative Kanal CL1.QM1 ist, überträgt der Lastausgleichsalgorithmus die Nachricht an XMITQ.CL1.QM1. Wenn der Clustersenderkanal erneut gestartet wird, werden neue Nachrichten und Nachrichten, die nicht für einen anderen Clustersenderkanal markiert wurden, durch den Kanal erneut übertragen.

Sie können die Zuordnung zwischen Übertragungswarteschlangen und Clustersenderkanälen auf einem aktiven System ändern. Sie können einen CLCHNAME -Parameter in einer Übertragungswarteschlange ändern oder den Parameter des Warteschlangenmanagers von **DEFCLXQ** ändern. Wenn ein Kanal, der von der Änderung betroffen ist, erneut gestartet wird, startet er den Übertragungswarteschlangen-Switching-Prozess (siehe „[Funktionsweise des Prozesses zum Wechseln des Clustersenderkanals in eine andere Übertragungswarteschlange](#)“ auf Seite 187).

Der Prozess zum Umschalten der Übertragungswarteschlange beginnt, wenn der Kanal erneut gestartet wird. Der Prozess zur Neuverteilung von Workloads wird gestartet, wenn der Kanal gestoppt wird. Die beiden Prozesse können parallel ausgeführt werden.



Der einfache Fall ist, dass beim Stoppen eines Clustersenderkanals der Prozess des Neuausgleichs nicht dazu führt, dass der Clustersenderkanal geändert wird, um Nachrichten in der Warteschlange weiterzuleiten. Dies ist der Fall, wenn kein anderer Clustersenderkanal die Nachrichten an die richtige Zieladresse weiterleiten kann. Wenn kein alternativer Clustersenderkanal die Nachrichten an ihre Zieladresse weiterleiten soll, bleiben die Nachrichten für denselben Clustersenderkanal markiert, nachdem der Clustersenderkanal gestoppt wurde. Wenn der Kanal ansteht, wenn ein Switch ansteht, werden die Nachrichten von den Vermittlungsprozessen in eine andere Übertragungswarteschlange verschoben, in der sie von demselben Clustersenderkanal verarbeitet werden.

Der komplexere Fall ist, wenn mehr als ein Clustersenderkanal einige Nachrichten an ein und dasselbe Ziel verarbeiten kann. Sie können den Clustersenderkanal stoppen und erneut starten, um den Übertragungswarteschlangenschalter auszulösen. In vielen Fällen hat der Algorithmus für den Lastausgleich bereits Nachrichten aus der ursprünglichen Übertragungswarteschlange in verschiedene Übertragungswarteschlangen verschoben, die von verschiedenen Clustersenderkanälen bedient werden. Nur die Nachrichten, die nicht von einem anderen Clustersenderkanal weitergeleitet werden können, bleiben in die neue Übertragungswarteschlange übertragen. In einigen Fällen, wenn der Kanal schnell erneut gestartet wird, bleiben einige Nachrichten, die durch den Algorithmus für den Lastausgleich übertragen werden könnten, bestehen. In diesem Fall werden einige der verbleibenden Nachrichten durch den Lastausgleichsprozess und einige durch den Prozess zum Umschalten der Übertragungswarteschlange umgeschaltet.

### **Zugehörige Konzepte**

„Clustering: Anwendungsisolierung mit mehreren Clusterübertragungswarteschlangen“ auf Seite 306

Sie können die Nachrichtenflüsse zwischen Warteschlangenmanagern in einem Cluster isolieren. Sie können Nachrichten, die von verschiedenen Clustersenderkanälen transportiert werden, in verschiedene Clusterübertragungswarteschlangen stellen. Sie können den Ansatz in einem einzelnen Cluster oder mit überlappenden Clustern verwenden. Das Thema enthält Beispiele und einige bewährte Verfahren, die Sie bei der Auswahl eines zu verwendenden Ansatzes führen.

„Berechnen der Größe des Protokolls“ auf Seite 436

Schätzen der Größe des Protokollwarteschlangenmanagers.

### **Zugehörige Tasks**

„Clustering: Planung der Konfiguration von Clusterübertragungswarteschlangen“ auf Seite 310

Sie werden durch die Auswahl von Clusterübertragungswarteschlangen geführt. Sie können eine gemeinsame Standardwarteschlange, separate Standardwarteschlangen oder manuell definierte Warteschlangen konfigurieren. Die Konfiguration mehrerer Clusterübertragungswarteschlangen gilt für andere Plattformen als z/OS.

„Erstellen von zwei überlappenden Clustern mit einem Gateway-Warteschlangenmanager“ auf Seite 235  
Befolgen Sie die Anweisungen in der Task, um überlappende Cluster mit einem Gateway-Warteschlangenmanager zu erstellen. Verwenden Sie die Cluster als Ausgangspunkt für die folgenden Beispiele, um Nachrichten in einer Anwendung von Nachrichten an andere Anwendungen in einem Cluster zu isolieren.

„WS-Manager zu einem Cluster hinzufügen: separate Übertragungswarteschlangen“ auf Seite 214

Befolgen Sie diese Anweisungen, um dem erstellten Cluster einen WS-Manager hinzuzufügen. Nachrichten zu Clusterwarteschlangen und Themen werden unter Verwendung mehrerer Clusterübertragungswarteschlangen übertragen.

„Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde“ auf Seite 219

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine zusätzliche Clusterübertragungswarteschlange, um den Nachrichtenverkehr auf einen einzelnen Warteschlangenmanager in einem Cluster zu trennen.

„Cluster und Cluster-Übertragungswarteschlange hinzufügen, um den Datenverkehr der Clusternachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet werden“ auf Seite 224

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Cluster-

nachrichten zu verwenden. Die Lösung verwendet einen zusätzlichen Cluster, um die Nachrichten in einer bestimmten Clusterwarteschlange zu isolieren.

## Clusterwarteschlangenmanager für die Aufnahme von vollständigen Repositories auswählen

In jedem Cluster müssen Sie mindestens eine, vorzugsweise zwei WS-Manager auswählen, um vollständige Repositories zu speichern. Zwei vollständige Repositories sind für alle, aber die außergewöhnlichsten Umstände ausreichend. Wählen Sie, wenn möglich, Warteschlangenmanager aus, die auf stabilen und permanent verbundenen Plattformen gehostet sind, die keine übereinstimmenden Ausfälle haben und die sich geographisch in einer zentralen Position befinden. Beachten Sie außerdem, dass Systeme als vollständige Repository-Hosts dediziert sind und diese Systeme nicht für andere Tasks verwenden.

*Vollständige Repositories* sind WS-Manager, die ein vollständiges Bild des Status des Clusters erhalten. Um diese Informationen gemeinsam nutzen zu können, ist jedes vollständige Repository über CLUSSDR-Kanäle (und die zugehörigen CLUSRCVR-Definitionen) mit jedem anderen vollständigen Repository im Cluster verbunden. Sie müssen diese Kanäle manuell definieren.

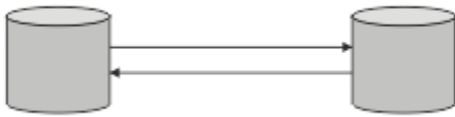


Abbildung 26. Zwei verbundene vollständige Repositories.

Jeder andere WS-Manager im Cluster verwaltet ein Bild dessen, was er derzeit über den Status des Clusters in einem *Teilrepository* weiß. Diese WS-Manager veröffentlichen Informationen über sich selbst und fordern Informationen zu anderen Warteschlangenmanagern unter Verwendung von zwei verfügbaren vollständigen Repositories an. Wenn ein ausgewähltes vollständiges Repository nicht verfügbar ist, wird ein anderes verwendet. Wenn das ausgewählte vollständige Repository wieder verfügbar wird, erfasst es die neuesten neuen und geänderten Informationen von den anderen, so dass sie in Schritt halten. Wenn alle vollständigen Repositories außer Betrieb sind, verwenden die anderen WS-Manager die Informationen, die sie in ihren Teilrepositories haben. Sie beschränken sich jedoch auf die Verwendung der Informationen, die sie haben; neue Informationen und Anforderungen für Aktualisierungen können nicht verarbeitet werden. Wenn die vollständigen Repositories wieder eine Verbindung zum Netz herstellen, werden Nachrichten ausgetauscht, um alle Repositories (sowohl vollständige als auch partielle) auf dem neuesten Stand zu bringen.

Wenn Sie die Zuordnung von vollständigen Repositories planen, müssen Sie die folgenden Aspekte berücksichtigen:

- Die Warteschlangenmanager, die zum Speichern der vollständigen Repositories ausgewählt wurden, müssen zuverlässig und verwaltet sein. Wählen Sie die Warteschlangenmanager aus, die auf einer stabilen und permanent verbundenen Plattform gehostet werden.
- Berücksichtigen Sie die geplanten Ausfälle für die Systeme, die als Host für Ihre vollständigen Repositories verwendet werden, und stellen Sie sicher, dass sie nicht übereinstimmende Ausfälle aufweisen.
- Berücksichtigen Sie die Netzleistung: Wählen Sie die Warteschlangenmanager aus, die sich geographisch in einer zentralen Position befinden oder die dasselbe System wie andere Warteschlangenmanager im Cluster gemeinsam nutzen.
- Überlegen Sie, ob ein Warteschlangenmanager Mitglied von mehreren Clustern ist. Es kann administrativ praktisch sein, denselben WS-Manager für die Verwendung der vollständigen Repositories für mehrere Cluster zu verwenden, vorausgesetzt, dieser Vorteil ist ausgeglichen, wie ausgelastet die Auslastung des Warteschlangenmanagers ist.
- Sie sollten einige Systeme dedizieren, um nur vollständige Repositories zu enthalten, und diese Systeme nicht für andere Tasks zu verwenden. Auf diese Weise wird sichergestellt, dass diese Systeme nur für die Konfiguration des Warteschlangenmanagers gewartet werden müssen und nicht aus dem Service für die Wartung anderer Geschäftsanwendungen entfernt werden. Außerdem stellt sie sicher, dass die



Task zum Verwalten des Repositorys nicht mit Anwendungen für Systemressourcen konkurrenzfähig ist. Dies kann besonders in großen Clustern (beispielsweise Cluster mit mehr als tausend Warteschlangenmanagern) von Vorteil sein, wenn die Auslastung des Clusterstatus durch die vollständigen Repositorys erheblich höher ist.

Es ist möglich, mehr als zwei vollständige Repositorys zu verwenden, wird aber selten empfohlen. Obwohl Objektdefinitionen (d. a. Warteschlangen, Themen und Kanäle) in alle verfügbaren vollständigen Repositorys fließen, werden nur Anforderungen von einem Teilrepository an maximal zwei vollständige Repositorys gestellt. Dies bedeutet, dass, wenn mehr als zwei vollständige Repositorys definiert sind und alle zwei vollständigen Repositorys nicht mehr verfügbar sind, einige Teilrepositorys möglicherweise keine Aktualisierungen empfangen, die sie erwarten würden. Weitere Informationen finden Sie unter [MQ-Cluster: Warum nur zwei vollständige Repositorys?](#)

Eine Situation, in der Sie möglicherweise mehr als zwei vollständige Repositorys definieren können, ist die Migration vorhandener vollständiger Repositorys auf neue Hardware oder neue Warteschlangenmanager. In diesem Fall sollten Sie die Ersatz-Vollrepositorys einführen und bestätigen, dass sie vollständig gefüllt wurden, bevor Sie die vorherigen vollständigen Repositorys entfernen. Wenn Sie ein vollständiges Repository hinzufügen, müssen Sie sich daran erinnern, dass Sie es mit CLUSSDR -Kanälen direkt mit jedem anderen vollständigen Repository verbinden müssen.

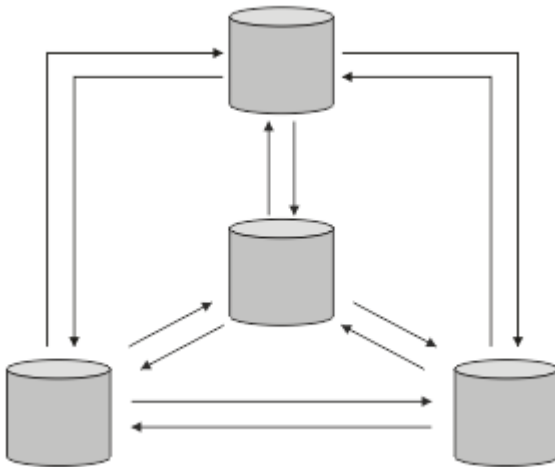


Abbildung 27. Mehr als zwei verbundene vollständige Repositorys

### Zugehörige Informationen

[MQ-Cluster: Warum nur zwei vollständige Repositorys?](#)

[Wie groß kann ein MQ-Cluster sein?](#)

## Cluster verwalten

Wählen Sie aus, welche WS-Manager mit welchem vollständigen Repository verknüpft werden sollen. Berücksichtigen Sie den Leistungseinwirkungseffekt, die Version des Warteschlangenmanagers und die Angabe, ob mehrere CLUSSDR -Kanäle wünschenswert sind.

Nachdem Sie die Warteschlangenmanager ausgewählt haben, um vollständige Repositorys zu speichern, müssen Sie festlegen, welche WS-Manager mit welchem vollständigen Repository verknüpft werden sollen. Die Kanaldefinition CLUSSDR verknüpft einen Warteschlangenmanager mit einem vollständigen Repository, aus dem es Informationen zu den anderen vollständigen Repositorys im Cluster herausfindet. Von da an sendet der Warteschlangenmanager Nachrichten an alle zwei vollständigen Repositorys. Es wird immer versucht, zuerst die Kanaldefinition in CLUSSDR zu verwenden. Sie können auswählen, ob ein Warteschlangenmanager mit einem vollständigen Repository verknüpft werden soll. Wählen Sie bei der Auswahl die Topologie Ihrer Konfiguration und die physische oder geografische Position der Warteschlangenmanager aus.

Da alle Clusterinformationen an zwei vollständige Repositorys gesendet werden, kann es zu Situationen kommen, in denen Sie eine zweite CLUSSDR -Kanaldefinition erstellen möchten. Sie können einen

zweiten CLUSSDR -Kanal in einem Cluster definieren, der über viele vollständige Repositorys verfügt, die sich über einen weiten Bereich erstrecken. Sie können dann steuern, an welche zwei vollständigen Repositorys Ihre Daten gesendet werden.

## Namenskonventionen für Cluster

Sie sollten die Benennung von Warteschlangenmanagern in demselben Cluster unter Verwendung einer Namenskonvention berücksichtigen, die den Cluster angibt, zu dem der Warteschlangenmanager gehört. Verwenden Sie eine ähnliche Namenskonvention für Kanalnamen und erweitern Sie diese, um die Kanalmerkmale zu beschreiben.

### Bewährte Verfahren bei der Benennung von MQ -Clustern

Obwohl Clusternamen bis zu 48 Zeichen umfassen können, sind relativ kurze Clusternamen hilfreich, wenn Namenskonventionen auf andere Objekte angewendet werden. Weitere Informationen finden Sie unter „[Bewährte Verfahren bei der Auswahl von Clusterkanalnamen](#)“ auf Seite 194.

Bei der Auswahl eines Clusternamens ist es normalerweise hilfreich, den 'Zweck' des Clusters (der wahrscheinlich langlebig ist) und nicht den 'Inhalt' darzustellen. Beispiel: 'B2BPROD' oder 'ACTTEST' statt 'QM1\_QM2\_QM3\_CLUS'.

### Bewährte Verfahren bei der Auswahl von Clusterwarteschlangenmanagernamen

Wenn Sie einen neuen Cluster und seine Member völlig neu erstellen, ziehen Sie eine Namenskonvention für die Warteschlangenmanager in Betracht, die ihre Clusternutzung widerspiegelt. Jeder WS-Manager muss einen anderen Namen haben. Sie können jedoch Warteschlangenmanagern in einem Cluster eine Gruppe ähnlicher Namen geben, um logische Gruppierungen zu identifizieren und sich daran zu erinnern (z. B. 'ACTTQM1, ACTTQM2).

Relatively short queue manager names (for example less than 8 characters) help if you choose to use the convention described in the next section, or something similar, for channel names.

### Bewährte Verfahren bei der Auswahl von Clusterkanalnamen

Da Warteschlangenmanager und Cluster Namen mit bis zu 48 Zeichen haben können und ein Kanalname auf 20 Zeichen begrenzt ist, müssen Sie beim ersten Benennen von Objekten darauf achten, dass Sie die Namenskonvention nicht mitten in einem Projekt ändern müssen (siehe vorherigen Abschnitt).

Denken Sie beim Definieren von Kanälen daran, dass automatisch erstellte Clustersenderkanäle auf jedem Warteschlangenmanager im Cluster ihren Namen von dem entsprechenden Clusterempfängerkanal übernehmen, der auf dem empfangenden Warteschlangenmanager im Cluster konfiguriert ist. Diese müssen daher eindeutig sein und *auf fernen Warteschlangenmanagern im Cluster sinnvoll sein*.

Eine allgemeine Methode ist die Verwendung des Namens des Warteschlangenmanagers, dem der Clusternamen vorangestellt ist. Wenn der Clusternamen beispielsweise CLUSTER1 lautet und die Warteschlangenmanager QM1, QM2 sind, lauten die Clusterempfängerkanäle CLUSTER1.QM1, CLUSTER1.QM2.

Sie können diese Konvention erweitern, wenn Kanäle unterschiedliche Prioritäten haben oder unterschiedliche Protokolle verwenden. For example:

- CLUSTER1.QM1.S1
- CLUSTER1.QM1.N3
- CLUSTER1.QM1.T4

In diesem Beispiel könnte S1 der erste SNA-Kanal, N3 der NetBIOS -Kanal mit der Netzpriorität 3 und T4 TCP-IP-Adresse, die ein IPV4 -Netz verwendet, sein.

### Gemeinsame Kanaldefinitionen benennen

Eine einzelne Kanaldefinition kann von mehreren Clustern gemeinsam genutzt werden. In diesem Fall müssen die hier vorgeschlagenen Namenskonventionen geändert werden. Wie in [Kanaldefinitionen](#)

verwalten beschrieben ist es jedoch in der Regel vorzuziehen, für jeden Cluster diskrete Kanäle zu definieren.

### **Ältere Namenskonventionen für Kanäle**

Außerhalb von Clusterumgebungen war es in der Vergangenheit üblich, eine 'FROMQM . TO . TARGETQM' -Namenskonvention zu verwenden, sodass Sie möglicherweise feststellen, dass vorhandene Cluster etwas Ähnliches verwendet haben (z. B. CLUSTER . TO . TARGET). Dies wird nicht als Teil eines neuen Clusterbenennungsschemas empfohlen, weil es die verfügbaren Zeichen weiter reduziert, um 'nützliche' Informationen in Ihrem Kanalnamen zu vermitteln.

## **Überlappende Cluster**

Überlappende Cluster bieten zusätzliche Verwaltungsfunktionen. Verwenden Sie Namenslisten, um die Anzahl der Befehle zu reduzieren, die für die Verwaltung von überlappenden Clustern erforderlich sind.

Sie können Cluster erstellen, die sich überschneiden. Es gibt eine Reihe von Gründen für die Definition von überlappenden Clustern, z. B.:

- Damit andere Organisationen ihre eigene Verwaltung haben können.
- Damit unabhängige Anwendungen separat verwaltet werden können.
- Zum Erstellen von Serviceklassen.

In [Abbildung 28 auf Seite 196](#) ist der Warteschlangenmanager STF2 ein Member der beiden Cluster. Wenn ein Warteschlangenmanager Mitglied mehrerer Cluster ist, können Sie die Namenslisten nutzen, um die Anzahl der benötigten Definitionen zu reduzieren. Namelists enthalten eine Liste mit Namen, z. B. Clusternamen. Sie können eine Namensliste erstellen, die die Cluster benennt. Geben Sie die Namensliste im Befehl ALTER QMGR für STF2 an, damit sie für beide Cluster ein vollständiger WS-Manager-Repository ist.

Wenn Sie mehr als einen Cluster in Ihrem Netzwerk haben, müssen Sie ihnen unterschiedliche Namen geben. Wenn zwei Cluster mit demselben Namen jemals zusammengeführt werden, ist es nicht möglich, sie wieder zu trennen. Es ist auch eine gute Idee, den Clustern und Kanälen unterschiedliche Namen zu geben. Sie werden leichter unterschieden, wenn Sie die Ausgabe von den DISPLAY -Befehlen betrachten. Die Namen von Warteschlangenmanagern müssen innerhalb eines Clusters eindeutig sein, damit sie ordnungsgemäß funktionieren.

### **Serviceklassen definieren**

Stellen Sie sich eine Universität vor, die über einen Warteschlangenmanager für jedes Mitglied des Personals und jeden Schüler verfügt. Nachrichten zwischen Mitarbeitern sind auf Kanälen mit hoher Priorität und hoher Bandbreite zu bereisen. Nachrichten zwischen Studenten werden auf billigeren, langsameren Kanälen zu reisen. Sie können dieses Netz mit Hilfe der traditionellen Methoden zur verteilten Steuerung von Warteschlangen konfigurieren. WebSphere MQ wählt anhand des Namens der Zielwarteschlange und des Namens des Warteschlangenmanagers aus, welche Kanäle verwendet werden sollen.

Um die Mitarbeiter und Studenten eindeutig zu unterscheiden, können Sie ihre Warteschlangenmanager in zwei Clustern gruppieren (siehe [Abbildung 28 auf Seite 196](#)). WebSphere MQ verschiebt Nachrichten in die Besprechungswarteschlange im Mitarbeitercluster nur über Kanäle, die in diesem Cluster definiert sind. Die Nachrichten für die Warteschlange 'gossip' im Cluster 'students' gehen über Kanäle, die in diesem Cluster definiert sind, und empfängt die entsprechende Serviceklasse.

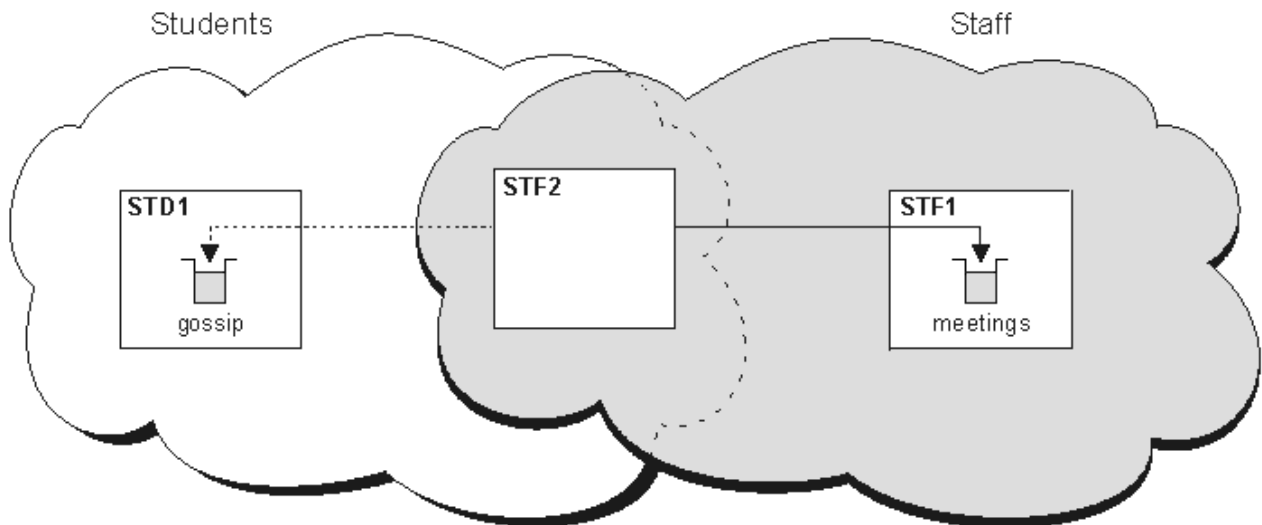


Abbildung 28. Serviceklassen

## Tipps zum Clustering

Möglicherweise müssen Sie einige Änderungen an Ihren Systemen oder Anwendungen vornehmen, bevor Sie Clustering verwenden. Es gibt sowohl Ähnlichkeiten als auch Unterschiede zwischen dem Verhalten der verteilten Steuerung von Warteschlangen.

- Der WebSphere MQ Explorer kann WebSphere MQ für z/OS -Warteschlangenmanager mit älteren Versionen als 6.0 nicht direkt verwalten.
- Sie müssen manuelle Konfigurationsdefinitionen zu Warteschlangenmanagern außerhalb eines Clusters hinzufügen, damit sie auf Clusterwarteschlangen zugreifen können.
- Wenn Sie zwei Cluster mit demselben Namen zusammenführen, können Sie sie nicht erneut trennen. Daher ist es ratsam, allen Clustern einen eindeutigen Namen zu geben.
- Wenn eine Nachricht in einem Warteschlangenmanager ankommt, aber keine Warteschlange vorhanden ist, wird die Nachricht in die Warteschlange für dead-Mail gestellt. Wenn es keine Warteschlange für dead-letter gibt, schlägt der Kanal fehl und versucht erneut, die Warteschlange zu wiederholen. Die Verwendung der Warteschlange "dead-letter" ist mit der Verwendung der verteilten Steuerung von Warteschlangen identisch.
- Die Integrität persistenter Nachrichten wird beibehalten. Nachrichten werden aufgrund der Verwendung von Clustern nicht dupliziert oder gehen verloren.
- Die Verwendung von Clustern reduziert die Systemverwaltung. Cluster machen es einfach, größere Netzwerke mit vielen mehr WS-Managern zu verbinden, als Sie in der Lage wären, die verteilte Steuerung von Warteschlangen zu verwenden. Es besteht die Gefahr, dass Sie übermäßige Netzressourcen in Anspruch nehmen, wenn Sie versuchen, die Kommunikation zwischen jedem WS-Manager in einem Cluster zu aktivieren.
- Wenn Sie den WebSphere MQ Explorer verwenden, der die Warteschlangenmanager in einer Baumstruktur darstellt, kann die Ansicht für große Cluster umständlich sein.
- WebSphere MQ Explorer kann einen Cluster mit Repository-WS-Managern unter WebSphere MQ für z/OS Version 6 oder höher verwalten. Sie müssen kein zusätzliches Repository auf einem separaten System festlegen. Für frühere Versionen von WebSphere MQ unter z/OS kann WebSphere MQ Explorer keinen Cluster mit Repository-Warteschlangenmanagern verwalten. Sie müssen ein zusätzliches Repository auf einem System benennen, das WebSphere MQ Explorer verwalten kann.
- Der Zweck von Verteilerlisten besteht darin, einen einzelnen MQPUT -Befehl zu verwenden, um dieselbe Nachricht an mehrere Ziele zu senden. Verteilerlisten werden unter WebSphere MQ für AIX, IBM i, HP-UX, Solaris, Linux und Windows unterstützt. Sie können Verteilerlisten mit WS-Manager-Clustern verwenden. In einem Cluster werden alle Nachrichten zum Zeitpunkt MQPUT erweitert. Der Vorteil in Bezug auf den Datenaustausch im Netz ist nicht so hoch wie in einer Nicht-Clustering-Umgebung. Der

Vorteil von Verteilerlisten besteht darin, dass die zahlreichen Kanäle und Übertragungswarteschlangen nicht manuell definiert werden müssen.

- Wenn Sie Cluster verwenden möchten, um die Auslastung Ihrer Workload zu überprüfen, untersuchen Sie Ihre Anwendungen. Sie können feststellen, ob Nachrichten von einem bestimmten Warteschlangenmanager oder in einer bestimmten Reihenfolge verarbeitet werden müssen. Solche Anwendungen sollen Nachrichtenaffinitäten haben. Möglicherweise müssen Sie Ihre Anwendungen ändern, bevor Sie sie in komplexen Clustern verwenden können.
- Sie können die Option MQ00\_BIND\_ON\_OPEN in einem MQOPEN verwenden, um das Senden von Nachrichten an ein bestimmtes Ziel zu erzwingen. Wenn der Zielwarteschlangenmanager nicht verfügbar ist, werden die Nachrichten erst zugestellt, wenn der WS-Manager wieder verfügbar ist. Nachrichten werden aufgrund des Risikos der Duplizierung nicht an einen anderen WS-Manager weitergeleitet.
- Wenn ein WS-Manager ein Cluster-Repository hosten soll, müssen Sie seinen Hostnamen oder seine IP-Adresse kennen. Sie müssen diese Informationen im Parameter CONNAME angeben, wenn Sie die Definition CLUSSDR auf anderen Warteschlangenmanagern, die den Cluster verbinden, definieren. Wenn Sie DHCP verwenden, kann sich die IP-Adresse ändern, da DHCP bei jedem Neustart eines Systems eine neue IP-Adresse zuordnen kann. Daher dürfen Sie die IP-Adresse in den CLUSSDR-Definitionen nicht angeben. Selbst wenn alle CLUSSDR-Definitionen den Hostnamen und nicht die IP-Adresse angeben, sind die Definitionen immer noch nicht zuverlässig. DHCP aktualisiert nicht notwendigerweise den DNS-Verzeichniseintrag für den Host mit der neuen Adresse. Wenn Sie WS-Manager als vollständige Repositories auf Systemen benennen müssen, die DHCP verwenden, installieren Sie die Software, die garantiert, dass Ihr DNS-Verzeichnis auf dem neuesten Stand ist.
- Verwenden Sie keine generischen Namen, z. B. VTAM generische Ressourcen oder dynamische Domäne Name Server (DDNS) generische Namen als Verbindungsnamen für Ihre Kanäle. Wenn dies der Fall ist, stellen die Kanäle möglicherweise eine Verbindung zu einem anderen Warteschlangenmanager her als erwartet.
- Sie können eine Nachricht nur aus einer lokalen Clusterwarteschlange abrufen, aber Sie können eine Nachricht in eine beliebige Warteschlange in einem Cluster stellen. Wenn Sie eine Warteschlange zur Verwendung des Befehls MQGET öffnen, öffnet der Warteschlangenmanager die lokale Warteschlange.
- Sie müssen keine Ihrer Anwendungen ändern, wenn Sie einen einfachen WebSphere MQ-Cluster einrichten. Die Anwendung kann die Zielwarteschlange im MQOPEN-Aufruf benennen und muss die Position des Warteschlangenmanagers nicht kennen. Wenn Sie einen Cluster für das Workload-Management einrichten, müssen Sie Ihre Anwendungen überprüfen und sie bei Bedarf ändern.
- Sie können die aktuellen Überwachungs- und Statusdaten für einen Kanal oder eine Warteschlange mit den Befehlen DISPLAY CHSTATUS und DISPLAY QSTATUS **runmqsc** anzeigen. Die Überwachungsdaten können verwendet werden, um die Leistung und den Status des Systems zu messen. Die Überwachung wird über WS-Manager-, Warteschlangen- und Kanalattribute gesteuert. Die Überwachung automatisch definierter Clustersenderkanäle ist mit dem WS-Manager-Attribut MONACLS möglich.

## **Zugehörige Konzepte**

### Cluster

#### Wie funktionieren Cluster?

„Vergleich von Clustering und verteilter Steuerung von Warteschlangen“ auf Seite 174

Vergleichen Sie die Komponenten, die für die Verbindung von WS-Managern mit verteilter Steuerung von Warteschlangen und Clustering definiert werden müssen.

„Komponenten eines Clusters“ auf Seite 176

Cluster bestehen aus Warteschlangenmanagern, Clusterrepositories, Clusterkanälen und Clusterwarteschlangen.

„IBM WebSphere MQ-Cluster verwalten“ auf Seite 200

Sie können IBM WebSphere MQ-Cluster erstellen, erweitern und verwalten.

## **Zugehörige Tasks**

„WS-Manager-Cluster konfigurieren“ auf Seite 171

Über die Links in diesem Abschnitt erfahren Sie, wie Cluster funktionieren, wie eine Clusterkonfiguration entworfen wird und wie ein einfacher Cluster eingerichtet wird.

„Neuen Cluster einrichten“ auf Seite 201

Führen Sie die folgenden Anweisungen aus, um den Beispielcluster zu konfigurieren. In separaten Anweisungen wird beschrieben, wie der Cluster auf TCP/IP, LU 6.2 und mit einer einzelnen Übertragungswarteschlange oder mehreren Übertragungswarteschlangen eingerichtet wird. Testen Sie den Cluster, indem Sie eine Nachricht von einem WS-Manager an den anderen Warteschlangenmanager senden.

## Kommunikation in einem Cluster einrichten

Ein Kanalinitiator wird benötigt, um einen Kommunikationskanal zu starten, wenn eine Nachricht ausgegeben werden soll. Ein Kanallistener wartet, bis das andere Ende eines Kanals gestartet wird, um die Nachricht zu empfangen.

### Vorbereitende Schritte

Wenn Sie die Kommunikation zwischen Warteschlangenmanagern in einem Cluster herstellen möchten, konfigurieren Sie einen Link mit einem der unterstützten Übertragungsprotokolle. Die unterstützten Protokolle sind TCP oder LU 6.2 auf allen Plattformen und NetBIOS oder SPX auf Windows -Systemen. Im Rahmen dieser Konfiguration benötigen Sie außerdem Kanalinitiatoren und Kanallistener, wie Sie es mit der verteilten Steuerung von Warteschlangen tun.

### Informationen zu diesem Vorgang

Für alle Cluster-WS-Manager ist ein Kanalinitiator erforderlich, um die systemdefinierte Initialisierungswarteschlange `SYSTEM.CHANNEL.INITQ` zu überwachen. `SYSTEM.CHANNEL.INITQ` ist die Initialisierungswarteschlange für alle Übertragungswarteschlangen, einschließlich der Clusterübertragungswarteschlange.

Jeder WS-Manager muss über einen Kanallistener verfügen. Ein Kanal-Listener-Programm wartet auf eingehende Netzanforderungen und startet den entsprechenden Empfängerkanal, wenn er benötigt wird. Die Implementierung von Channel-Listenern ist plattformspezifisch, es gibt jedoch einige allgemeine Funktionen. Auf allen WebSphere MQ -Plattformen kann der Listener mit dem Befehl `START LISTENER` gestartet werden. Auf Systemen mit WebSphere MQ für IBM i, Windows und UNIX and Linux können Sie den Listener automatisch gleichzeitig mit dem Warteschlangenmanager starten. Wenn Sie den Listener automatisch starten möchten, setzen Sie das Attribut `CONTROL` des Objekts `LISTENER` auf `QMGR` oder `STARTONLY`.

### Vorgehensweise

1. Starten Sie den Kanalinitiator.

- 

#### IBM WebSphere MQ für Windows-, UNIX and Linux -Systeme

Wenn Sie einen WS-Manager starten, wird automatisch ein Kanalinitiator gestartet, wenn das Warteschlangenmanager-Attribut `SCHINIT` auf `QMGR` gesetzt ist. Andernfalls kann er mit dem Befehl `runmqsc START CHINIT` oder dem Steuerbefehl `runmqchi` gestartet werden.

2. Starten Sie den Kanal-Listener.

- 

#### IBM WebSphere MQ for Windows

Verwenden Sie entweder das von WebSphere MQ bereitgestellte Channel-Listener-Programm oder die vom Betriebssystem bereitgestellten Funktionen.

Verwenden Sie zum Starten des WebSphere MQ -Kanallisteners den Befehl `RUNQLSR`. Beispiel:

```
RUNQLSR -t tcp -p 1414 -m QM1
```

- 

## IBM WebSphere MQ auf UNIX and Linux -Systemen

Verwenden Sie entweder das von WebSphere MQ bereitgestellte Kanalempfangsprogramm oder die vom Betriebssystem bereitgestellten Funktionen, z. B. **inetd** für TCP-Kommunikation.

Verwenden Sie zum Starten des WebSphere MQ -Kanalempfangsprogramms den Befehl **runmqclsr**. Beispiel:

```
runmqclsr -t tcp -p 1414 -m QM1
```

Wenn Sie **inetd** zum Starten von Kanälen verwenden möchten, konfigurieren Sie zwei Dateien:

- a. Bearbeiten Sie die Datei `/etc/services`. Sie müssen als Superuser oder als Root angemeldet sein. Wenn die folgende Zeile nicht in der Datei enthalten ist, fügen Sie sie wie folgt hinzu:

```
MQSeries 1414/tcp # Websphere MQ channel listener
```

Dabei ist 1414 die von IBM WebSphere MQ benötigte Portnummer. Sie können die Portnummer ändern, aber sie muss mit der Port-Nummer übereinstimmen, die auf der Senderseite angegeben wurde.

- b. Bearbeiten Sie die Datei `/etc/inetd.conf`. Wenn die folgende Zeile nicht in dieser Datei enthalten ist, fügen Sie sie wie dargestellt hinzu:

```
MQSeries stream tcp nowait mqm MQ_INSTALLATION_PATH/bin/amqcrista amqcrista
-m queue.manager.name
```

Dabei wird `MQ_INSTALLATION_PATH` durch das übergeordnete Verzeichnis ersetzt, in dem WebSphere MQ installiert ist.

Die Aktualisierungen werden aktiv, nachdem **inetd** die Konfigurationsdateien erneut gelesen hat. Geben Sie die folgenden Befehle von der Rootbenutzer-ID aus:

Unter AIX:

```
refresh -s inetd
```

Unter HP-UX:

```
inetd -c
```

Unter Solaris oder Linux:

- a. Suchen Sie die Prozess-ID von **inetd** mit dem Befehl:

```
ps -ef | grep inetd
```

- b. Führen Sie den entsprechenden Befehl aus:

- Für Solaris 9 und Linux:

```
kill -1 inetd processid
```

- Für Solaris 10 oder höhere Versionen:

```
inetconv
```

## Wie lange werden die Informationen in den Warteschlangenmanager-Repositorys aufbewahrt?

WS-Manager-Repositorys behalten Informationen für 30 Tage bei. Ein automatischer Prozess aktualisiert die Informationen, die gerade verwendet werden, effizient.

Wenn ein Warteschlangenmanager einige Informationen über sich selbst sendet, speichern die vollständigen und partiellen Repository-WS-Manager die Informationen für 30 Tage. Informationen werden bei-

spielsweise gesendet, wenn ein WS-Manager die Erstellung einer neuen Warteschlange bewirbt. Damit diese Informationen nicht auslaufen, werden die Warteschlangenmanager nach 27 Tagen automatisch alle Informationen über sich selbst erneut senden. Wenn ein Teilrepository eine neue Anforderung zum Teil über die 30-Tage-Lebensdauer sendet, bleibt die Ablaufzeit die ursprünglichen 30 Tage.

Wenn Informationen verfallen, wird sie nicht sofort aus dem Repository entfernt. Stattdessen wird sie für eine Karenzzeit von 60 Tagen gehalten. Wenn innerhalb der Karenzzeit keine Aktualisierung empfangen wird, werden die Informationen entfernt. Die Karenzzeit ermöglicht es, dass ein WS-Manager zum Ablaufdatum vorübergehend außer Betrieb war. Wenn ein WS-Manager länger als 90 Tage von einem Cluster getrennt wird, wird er nicht mehr Teil des Clusters. Wenn er jedoch wieder eine Verbindung zum Netz herstellt, wird er wieder Teil des Clusters. Vollständige Repositories verwenden keine Informationen, die abgelaufen sind, um neue Anforderungen von anderen Warteschlangenmanagern zu erfüllen.

Wenn ein Warteschlangenmanager eine Anforderung zum Sichern/Enddatum aus einem vollständigen Repository sendet, dauert die Anforderung ebenfalls 30 Tage. Nach 27 Tagen IBM WebSphere MQ prüft die Anforderung. Wenn sie in den 27 Tagen referenziert wurde, wird sie automatisch aktualisiert. Ist dies nicht der Fall, bleibt sie verfallen und wird vom WS-Manager aktualisiert, wenn sie erneut benötigt wird. Das Abflauen von Anforderungen verhindert eine Ansammlung von Anforderungen für Informationen von ruhenden WS-Managern.

**Anmerkung:** Bei großen Clustern kann es unterbrechend sein, wenn viele Warteschlangenmanager automatisch alle Informationen zu sich selbst erneut senden. Siehe [„Die Neuerung in einem großen Cluster kann die Leistung und Verfügbarkeit des Clusters beeinträchtigen.“](#) auf Seite 331.

### Zugehörige Konzepte

[„Clustering: Best Practices für REFRESH CLUSTER verwenden“](#) auf Seite 331

Sie verwenden den Befehl **REFRESH CLUSTER**, um alle lokal gespeicherten Informationen zu einem Cluster zu löschen und diese Informationen aus den vollständigen Repositories im Cluster erneut zu erstellen. Sie sollten diesen Befehl nicht verwenden, außer in außergewöhnlichen Umständen. Wenn Sie es verwenden müssen, gibt es besondere Hinweise darauf, wie Sie es verwenden. Diese Informationen sind ein Leitfaden, der auf Tests und Feedback von Kunden basiert.

## IBM WebSphere MQ -Cluster verwalten

Sie können IBM WebSphere MQ -Cluster erstellen, erweitern und verwalten.

Details zum Verwalten Ihrer IBM WebSphere MQ -Cluster finden Sie in den folgenden Unterabschnitten:

### Zugehörige Konzepte

[Cluster](#)

[Wie funktionieren Cluster?](#)

[„Vergleich von Clustering und verteilter Steuerung von Warteschlangen“](#) auf Seite 174

Vergleichen Sie die Komponenten, die für die Verbindung von WS-Managern mit verteilter Steuerung von Warteschlangen und Clustering definiert werden müssen.

[„Komponenten eines Clusters“](#) auf Seite 176

Cluster bestehen aus Warteschlangenmanagern, Clusterrepositories, Clusterkanälen und Clusterwarteschlangen.

### Zugehörige Tasks

[„WS-Manager-Cluster konfigurieren“](#) auf Seite 171

Über die Links in diesem Abschnitt erfahren Sie, wie Cluster funktionieren, wie eine Clusterkonfiguration entworfen wird und wie ein einfacher Cluster eingerichtet wird.

[„Neuen Cluster einrichten“](#) auf Seite 201

Führen Sie die folgenden Anweisungen aus, um den Beispielcluster zu konfigurieren. In separaten Anweisungen wird beschrieben, wie der Cluster auf TCP/IP, LU 6.2 und mit einer einzelnen Übertragungswarteschlange oder mehreren Übertragungswarteschlangen eingerichtet wird. Testen Sie den Cluster, indem Sie eine Nachricht von einem WS-Manager an den anderen Warteschlangenmanager senden.



## Neuen Cluster einrichten

Führen Sie die folgenden Anweisungen aus, um den Beispielcluster zu konfigurieren. In separaten Anweisungen wird beschrieben, wie der Cluster auf TCP/IP, LU 6.2 und mit einer einzelnen Übertragungswarteschlange oder mehreren Übertragungswarteschlangen eingerichtet wird. Testen Sie den Cluster, indem Sie eine Nachricht von einem WS-Manager an den anderen Warteschlangenmanager senden.

### Vorbereitende Schritte

- Anstatt diese Anweisungen zu befolgen, können Sie einen der Assistenten verwenden, die mit IBM WebSphere MQ Explorer bereitgestellt werden, um einen Cluster wie den von dieser Task erstellten Cluster zu erstellen. Klicken Sie mit der rechten Maustaste auf den Ordner 'Warteschlangenmanagercluster', klicken Sie anschließend auf **Neu > Warteschlangenmanagercluster** und befolgen Sie die Anweisungen im Assistenten.
- Hintergrundinformationen zu den Schritten, die zum Einrichten eines Clusters ausgeführt werden, finden Sie unter „Clusterwarteschlangen“ auf Seite 179, [Kanäle](#) und [Empfangsprogramme](#).

### Informationen zu diesem Vorgang

Sie richten ein neues IBM WebSphere MQ-Netz für eine Geschäftskette ein. Das Geschäft hat zwei Filialen, eine in London und eine in New York. Die Daten und Anwendungen für die einzelnen Filiale werden von Systemen gehostet, auf denen separate Warteschlangenmanager ausgeführt werden. Die beiden WS-Manager werden als LONDON und NEWYORK bezeichnet. Die Bestandsanwendung wird auf dem System in New York ausgeführt und ist mit dem Warteschlangenmanager NEWYORK verbunden. Die Anwendung wird durch den Eingang von Nachrichten in der INVENTQ-Warteschlange gesteuert, die von NEWYORK bereitgestellt wird. Die beiden WS-Manager LONDON und NEWYORK sollen in einem Cluster mit dem Namen INVENTORY verknüpft werden, damit beide Nachrichten an den INVENTQ stellen können.

Abbildung 29 auf Seite 201 zeigt, wie dieser Cluster aussieht.

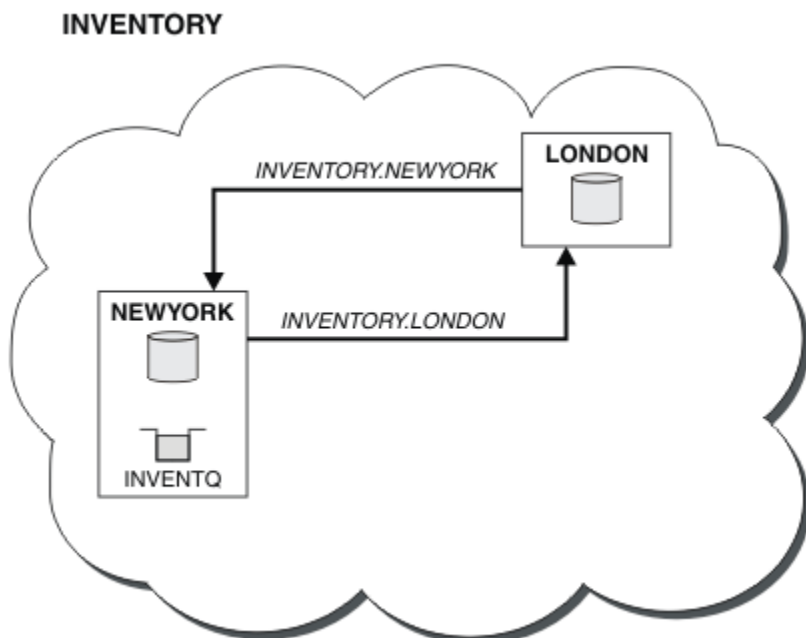


Abbildung 29. Cluster BESTANDSFÜHRUNG mit zwei WS-Managern

Sie können jeden Warteschlangenmanager im Cluster konfigurieren, der nicht unter z/OS ausgeführt wird, um Nachrichten über verschiedene Clusterübertragungswarteschlangen an andere Warteschlangenmanager im Cluster zu senden.

Die Anweisungen zum Festlegen des Clusters variieren je nach Transportprotokoll, Anzahl der Übertragungswarteschlangen oder der Plattform. Sie haben die Wahl zwischen drei Kombinationen. Das Prüfungsverfahren bleibt für alle Kombinationen gleich.

## Prozedur

- [„Cluster mit TCP/IP mit einer einzigen Übertragungswarteschlange pro WS-Manager konfigurieren“ auf Seite 202](#)
- [„Cluster unter TCP/IP mit mehreren Übertragungswarteschlangen pro WS-Manager konfigurieren“ auf Seite 205](#)
- [„Cluster mit LU 6.2 unter z/OS einrichten“ auf Seite 208](#)
- [„Cluster verifizieren“ auf Seite 210](#)

## Ergebnisse

[Abbildung 29 auf Seite 201](#) zeigt den in dieser Task eingerichteten Cluster INVENTORY.

INVENTORY ist eindeutig ein kleiner Cluster. Sie ist jedoch als Proof of Concept nützlich. Das Wichtige an diesem Cluster ist der Umfang, den es für die zukünftige Erweiterung bietet.

### Zugehörige Konzepte

[Cluster](#)

[Wie funktionieren Cluster?](#)

[„Vergleich von Clustering und verteilter Steuerung von Warteschlangen“ auf Seite 174](#)

Vergleichen Sie die Komponenten, die für die Verbindung von WS-Managern mit verteilter Steuerung von Warteschlangen und Clustering definiert werden müssen.

[„Komponenten eines Clusters“ auf Seite 176](#)

Cluster bestehen aus Warteschlangenmanagern, Clusterrepositorys, Clusterkanälen und Clusterwarteschlangen.

[„IBM WebSphere MQ -Cluster verwalten“ auf Seite 200](#)

Sie können IBM WebSphere MQ -Cluster erstellen, erweitern und verwalten.

### Zugehörige Tasks

[„WS-Manager-Cluster konfigurieren“ auf Seite 171](#)

Über die Links in diesem Abschnitt erfahren Sie, wie Cluster funktionieren, wie eine Clusterkonfiguration entworfen wird und wie ein einfacher Cluster eingerichtet wird.

## ***Cluster mit TCP/IP mit einer einzigen Übertragungswarteschlange pro WS-Manager konfigurieren***

### Vorbereitende Schritte

- Unter AIX, HP-UX, IBM i, Linux, Solaris, and Windows muss das Warteschlangenmanagerattribut **DEFCLXQ** als Standardwert **SCTQ** beibehalten werden.

### Informationen zu diesem Vorgang

Führen Sie die folgenden Schritte aus, um einen Cluster unter AIX, HP-UX, IBM i, Linux, Solaris, and Windows mit dem Transportprotokoll TCP/IP einzurichten.

### Vorgehensweise

1. Entscheiden Sie sich für die Organisation des Clusters und dessen Namen.

Sie haben entschieden, die beiden WS-Manager LONDON und NEWYORK in einen Cluster zu verlinken. Ein Cluster mit nur zwei Warteschlangenmanagern bietet nur einen marginalen Vorteil gegenüber einem Netz, das die verteilte Steuerung von Warteschlangen verwenden soll. Es ist ein guter Weg, um

zu beginnen, und es bietet Raum für zukünftige Erweiterungen. Wenn Sie neue Filialen Ihres Geschäfts öffnen, können Sie die neuen WS-Manager problemlos dem Cluster hinzufügen. Durch das Hinzufügen neuer WS-Manager wird das vorhandene Netz nicht zerrüttet; siehe „[WS-Manager zu einem Cluster hinzufügen](#)“ auf Seite 211.

Die einzige Anwendung, die Sie gerade ausführen, ist die Bestandsanwendung. Der Clusternamen lautet INVENTORY.

## 2. Entscheiden Sie, welche WS-Manager vollständige Repositorys enthalten sollen.

In jedem Cluster müssen Sie mindestens einen Warteschlangenmanager (oder vorzugsweise zwei) für die Aufnahme von vollständigen Repositorys benennen. In diesem Beispiel gibt es nur zwei WS-Manager, LONDON und NEWYORK, die beide vollständige Repositorys enthalten.

- a. Sie können die restlichen Schritte in beliebiger Reihenfolge ausführen.
- b. Wenn Sie die Schritte ausführen, werden möglicherweise Warnungen in das WS-Manager-Protokoll geschrieben. Die Nachrichten sind ein Ergebnis fehlender Definitionen, die Sie noch nicht hinzugefügt haben.

Examples of the responses to the commands are shown in a box like this after each step in this task. These examples show the responses returned by WebSphere MQ for AIX. The responses vary on other platforms.

- c. Bevor Sie mit diesen Schritten fortfahren, müssen Sie sicherstellen, dass die WS-Manager gestartet wurden.

## 3. Ändern Sie die WS-Manager-Definitionen, um Repositorydefinitionen hinzuzufügen.

Ändern Sie auf jedem WS-Manager, der ein vollständiges Repository aufnehmen soll, die lokale WS-Manager-Definition mit dem Befehl ALTER QMGR und geben Sie das Attribut REPOS an:

```
ALTER QMGR REPOS(INVENTORY)
```

```
1 : ALTER QMGR REPOS(INVENTORY)
AMQ8005: Websphere MQ queue manager changed.
```

Geben Sie Folgendes ein, wenn Sie Folgendes eingeben:

- a. `runmqsc LONDON`
- b. `ALTER QMGR REPOS(INVENTORY)`

LONDON wird in ein vollständiges Repository geändert.

## 4. Definieren Sie die Empfangsprogramme.

Definieren Sie einen Listener, der Netzanforderungen von anderen Warteschlangenmanagern für jeden WS-Manager im Cluster akzeptiert. Geben Sie auf den LONDON -Warteschlangenmanagern den folgenden Befehl aus:

```
DEFINE LISTENER(LONDON_LS) TRPTYPE(TCP) CONTROL(QMGR)
```

**Anmerkung:** Wenn Sie einen Listener definieren, sollte eine Portnummer definiert werden, wenn Sie IP-Adressen im Feld CONNAME verwenden und die Portnummer nicht der Standardport (1414) ist. Beispiel:

```
DEFINE LISTENER(LONDON_LS) TRPTYPE(TCP) CONTROL(QMGR) PORT(1415)
```

Mit dem Attribut CONTROL wird sichergestellt, dass das Empfangsprogramm gestartet und gestoppt wird, wenn der Warteschlangenmanager ausgeführt wird.

Der Listener wird nicht gestartet, wenn er definiert ist. Daher muss er mit dem folgenden MQSC-Befehl zum ersten Mal manuell gestartet werden:

```
START LISTENER(LONDON_LS)
```

Geben Sie ähnliche Befehle für alle anderen WS-Manager im Cluster ein, und ändern Sie den Namen des Listeners für die einzelnen WS-Manager.

Es gibt mehrere Möglichkeiten, diese Listener zu definieren, wie in [Listener](#) gezeigt.

#### 5. Definieren Sie den CLUSRCVR-Kanal für den WS-Manager von LONDON .

Definieren Sie auf jedem Warteschlangenmanager in einem Cluster einen Clusterempfängerkanal, auf dem der Warteschlangenmanager Nachrichten empfangen kann. CLUSRCVR definiert den Verbindungsnamen des Warteschlangenmanagers. Der Verbindungsname wird in den Repositorys gespeichert, auf die sich andere Warteschlangenmanager beziehen können. Das Schlüsselwort CLUSTER zeigt die Verfügbarkeit des Warteschlangenmanagers an, um Nachrichten von anderen Warteschlangenmanagern im Cluster zu empfangen.

In diesem Beispiel lautet der Kanalname INVENTORY . LONDON, und der Verbindungsname (CONNAME) ist die Netzadresse der Maschine, auf der sich der Warteschlangenmanager befindet, LONDON . CHSTORE . COM. Die Netzadresse kann als alphanumerischer DNS-Hostname oder als IP-Adresse in der Schreibweise mit Trennzeichen IPv4 eingegeben werden. Beispiel: 192 . 0 . 2 . 0 oder IPv6 im Hexadezimalformat; Beispiel: 2001 : DB8 : 0204 : acff : fe97 : 2c34 : fde0 : 3485. Die Portnummer wurde nicht angegeben, daher wird der Standardport (1414) verwendet.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
```

```
1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
AMQ8014: Websphere MQ channel created.
07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
```

#### 6. Definieren Sie den CLUSRCVR-Kanal für den WS-Manager von NEWYORK .

Wenn der Kanal-Listener den Standardport (normalerweise 1414) verwendet und der Cluster keinen WS-Manager in z/OS enthält, können Sie den CONNAME weglassen.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager NEWYORK')
```

#### 7. Definieren Sie den CLUSSDR-Kanal auf dem LONDON -Warteschlangenmanager.

Definieren Sie auf jedem Warteschlangenmanager in einem Cluster einen Clustersenderkanal. Der Warteschlangenmanager sendet Nachrichten an einen der Warteschlangenmanager mit vollständigem Repository im Clustersenderkanal. In diesem Fall gibt es nur zwei WS-Manager, die beide vollständige Repositorys enthalten. Sie müssen jeweils über eine CLUSSDR -Definition verfügen, die auf den CLUSRCVR -Kanal verweist, der auf dem anderen Warteschlangenmanager definiert ist. Die Kanalnamen, die in den CLUSSDR -Definitionen angegeben sind, müssen mit den Kanalnamen in den entsprechenden CLUSRCVR -Definitionen übereinstimmen. Sobald ein Warteschlangenmanager Definitionen für einen Clusterempfängerkanal und einen Clustersenderkanal in demselben Cluster hat, wird der Clustersenderkanal gestartet.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
```

```
1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
AMQ8014: Websphere MQ channel created.
07/09/98 13:00:18 Channel program started.
```

#### 8. Definieren Sie den CLUSSDR-Kanal auf dem NEWYORK -Warteschlangenmanager.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from NEWYORK to repository at LONDON')
```

## 9. Definieren Sie die Clusterwarteschlange INVENTQ.

Definieren Sie die INVENTQ -Warteschlange auf dem NEWYORK -Warteschlangenmanager und geben Sie dabei das Schlüsselwort CLUSTER an.

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

```
1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
AMQ8006: Websphere MQ queue created.
```

Das Schlüsselwort CLUSTER bewirkt, dass die Warteschlange für den Cluster zugänglich gemacht wird. Sobald die Warteschlange definiert ist, wird sie den anderen WS-Managern im Cluster zur Verfügung gestellt. Sie können Nachrichten an sie senden, ohne eine Definition einer fernen Warteschlange für sie vornehmen zu müssen.

Alle Definitionen sind vollständig. Starten Sie auf allen Plattformen auf jedem WS-Manager ein Empfangsprogramm. Das Listenerprogramm wartet auf eingehende Netzanforderungen und startet den Clusterempfängerkanal, wenn er benötigt wird.

## **Cluster unter TCP/IP mit mehreren Übertragungswarteschlangen pro WS-Manager konfigurieren**

### **Informationen zu diesem Vorgang**

Führen Sie die folgenden Schritte aus, um einen Cluster unter AIX, HP-UX, IBM i, Linux, Solaris, and Windows mit dem Transportprotokoll TCP/IP einzurichten. Die Repository-WS-Manager sind so konfiguriert, dass sie eine andere Clusterübertragungswarteschlange verwenden, um Nachrichten an die anderen Warteschlangenmanager und andere Warteschlangenmanager im Cluster zu senden. Wenn Sie dem Cluster WS-Manager hinzufügen, die auch andere Übertragungswarteschlangen verwenden sollen, folgen Sie der Task „WS-Manager zu einem Cluster hinzufügen: separate Übertragungswarteschlangen“ auf Seite 214. Unter z/OS kann ein Warteschlangenmanager nicht für die Verwendung separater Clusterübertragungswarteschlangen konfiguriert werden.

### **Vorgehensweise**

1. Entscheiden Sie sich für die Organisation des Clusters und dessen Namen.

Sie haben entschieden, die beiden WS-Manager LONDON und NEWYORK in einen Cluster zu verlinken. Ein Cluster mit nur zwei Warteschlangenmanagern bietet nur einen marginalen Vorteil gegenüber einem Netz, das die verteilte Steuerung von Warteschlangen verwenden soll. Es ist ein guter Weg, um zu beginnen, und es bietet Raum für zukünftige Erweiterungen. Wenn Sie neue Filialen Ihres Geschäfts öffnen, können Sie die neuen WS-Manager problemlos dem Cluster hinzufügen. Durch das Hinzufügen neuer WS-Manager wird das vorhandene Netz nicht zerrüttet; siehe „WS-Manager zu einem Cluster hinzufügen“ auf Seite 211.

Die einzige Anwendung, die Sie gerade ausführen, ist die Bestandsanwendung. Der Clustername lautet INVENTORY.

2. Entscheiden Sie, welche WS-Manager vollständige Repositorys enthalten sollen.

In jedem Cluster müssen Sie mindestens einen Warteschlangenmanager (oder vorzugsweise zwei) für die Aufnahme von vollständigen Repositorys benennen. In diesem Beispiel gibt es nur zwei WS-Manager, LONDON und NEWYORK, die beide vollständige Repositorys enthalten.

- a. Sie können die restlichen Schritte in beliebiger Reihenfolge ausführen.

- b. Wenn Sie die Schritte ausführen, werden möglicherweise Warnungen in das WS-Manager-Protokoll geschrieben. Die Nachrichten sind ein Ergebnis fehlender Definitionen, die Sie noch nicht hinzugefügt haben.

Examples of the responses to the commands are shown in a box like this after each step in this task. These examples show the responses returned by WebSphere MQ for AIX. The responses vary on other platforms.

- c. Bevor Sie mit diesen Schritten fortfahren, müssen Sie sicherstellen, dass die WS-Manager gestartet wurden.
3. Ändern Sie die WS-Manager-Definitionen, um Repositorydefinitionen hinzuzufügen.

Ändern Sie auf jedem WS-Manager, der ein vollständiges Repository aufnehmen soll, die lokale WS-Manager-Definition mit dem Befehl ALTER QMGR und geben Sie das Attribut REPOS an:

```
ALTER QMGR REPOS(INVENTORY)
```

```
1 : ALTER QMGR REPOS(INVENTORY)
AMQ8005: Websphere MQ queue manager changed.
```

Geben Sie Folgendes ein, wenn Sie Folgendes eingeben:

- a. runmqsc LONDON
- b. ALTER QMGR REPOS(INVENTORY)

LONDON wird in ein vollständiges Repository geändert.

4. Ändern Sie die WS-Manager-Definitionen, um separate Clusterübertragungswarteschlangen für die einzelnen Ziele zu erstellen.

```
ALTER QMGR DEFCLXQ(CHANNEL)
```

Auf jedem WS-Manager, den Sie dem Cluster hinzufügen, entscheiden Sie, ob separate Übertragungswarteschlangen verwendet werden sollen oder nicht. Weitere Informationen finden Sie in den Abschnitten „WS-Manager zu einem Cluster hinzufügen“ auf Seite 211 und „WS-Manager zu einem Cluster hinzufügen: separate Übertragungswarteschlangen“ auf Seite 214.

5. Definieren Sie die Empfangsprogramme.

Definieren Sie einen Listener, der Netzanforderungen von anderen Warteschlangenmanagern für jeden WS-Manager im Cluster akzeptiert. Geben Sie auf den LONDON -Warteschlangenmanagern den folgenden Befehl aus:

```
DEFINE LISTENER(LONDON_LS) TRPTYPE(TCP) CONTROL(QMGR)
```

**Anmerkung:** Wenn Sie einen Listener definieren, sollte eine Portnummer definiert werden, wenn Sie IP-Adressen im Feld CONNAME verwenden und die Portnummer nicht der Standardport (1414) ist. Beispiel:

```
DEFINE LISTENER(LONDON_LS) TRPTYPE(TCP) CONTROL(QMGR) PORT(1415)
```

Mit dem Attribut CONTROL wird sichergestellt, dass das Empfangsprogramm gestartet und gestoppt wird, wenn der Warteschlangenmanager ausgeführt wird.

Der Listener wird nicht gestartet, wenn er definiert ist. Daher muss er mit dem folgenden MQSC-Befehl zum ersten Mal manuell gestartet werden:

```
START LISTENER(LONDON_LS)
```

Geben Sie ähnliche Befehle für alle anderen WS-Manager im Cluster ein, und ändern Sie den Namen des Listeners für die einzelnen WS-Manager.

Es gibt mehrere Möglichkeiten, diese Listener zu definieren, wie in [Listener](#) gezeigt.

6. Definieren Sie den CLUSRCVR-Kanal für den WS-Manager von LONDON .

Definieren Sie auf jedem Warteschlangenmanager in einem Cluster einen Clusterempfängerkanal, auf dem der Warteschlangenmanager Nachrichten empfangen kann. CLUSRCVR definiert den Verbindungsnamen des Warteschlangenmanagers. Der Verbindungsname wird in den Repositorys gespeichert, auf die sich andere Warteschlangenmanager beziehen können. Das Schlüsselwort CLUSTER zeigt die Verfügbarkeit des Warteschlangenmanagers an, um Nachrichten von anderen Warteschlangenmanagern im Cluster zu empfangen.

In diesem Beispiel lautet der Kanalname INVENTORY . LONDON, und der Verbindungsname (CONNAME) ist die Netzadresse der Maschine, auf der sich der Warteschlangenmanager befindet, LONDON . CHSTORE . COM. Die Netzadresse kann als alphanumerischer DNS-Hostname oder als IP-Adresse in der Schreibweise mit Trennzeichen IPv4 eingegeben werden. Beispiel: 192 . 0 . 2 . 0 oder IPv6 im Hexadezimalformat; Beispiel: 2001 : DB8 : 0204 : acff : fe97 : 2c34 : fde0 : 3485. Die Portnummer wurde nicht angegeben, daher wird der Standardport (1414) verwendet.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
```

```
1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
AMQ8014: Websphere MQ channel created.
07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
```

7. Definieren Sie den CLUSRCVR-Kanal für den WS-Manager von NEWYORK .

Wenn der Kanal-Listener den Standardport (normalerweise 1414) verwendet und der Cluster keinen WS-Manager in z/OS enthält, können Sie den CONNAME weglassen.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager NEWYORK')
```

8. Definieren Sie den CLUSSDR-Kanal auf dem LONDON -Warteschlangenmanager.

Definieren Sie auf jedem Warteschlangenmanager in einem Cluster einen Clustersenderkanal. Der Warteschlangenmanager sendet Nachrichten an einen der Warteschlangenmanager mit vollständigem Repository im Clustersenderkanal. In diesem Fall gibt es nur zwei WS-Manager, die beide vollständige Repositorys enthalten. Sie müssen jeweils über eine CLUSSDR -Definition verfügen, die auf den CLUSRCVR -Kanal verweist, der auf dem anderen Warteschlangenmanager definiert ist. Die Kanalnamen, die in den CLUSSDR -Definitionen angegeben sind, müssen mit den Kanalnamen in den entsprechenden CLUSRCVR -Definitionen übereinstimmen. Sobald ein Warteschlangenmanager Definitionen für einen Clusterempfängerkanal und einen Clustersenderkanal in demselben Cluster hat, wird der Clustersenderkanal gestartet.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
```

```
1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
AMQ8014: Websphere MQ channel created.
07/09/98 13:00:18 Channel program started.
```

9. Definieren Sie den CLUSSDR-Kanal auf dem NEWYORK -Warteschlangenmanager.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from NEWYORK to repository at LONDON')
```

10. Definieren Sie die Clusterwarteschlange INVENTQ.



Definieren Sie die INVENTQ -Warteschlange auf dem NEWYORK -Warteschlangenmanager und geben Sie dabei das Schlüsselwort CLUSTER an.

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

```
1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
AMQ8006: Websphere MQ queue created.
```

Das Schlüsselwort CLUSTER bewirkt, dass die Warteschlange für den Cluster zugänglich gemacht wird. Sobald die Warteschlange definiert ist, wird sie den anderen WS-Managern im Cluster zur Verfügung gestellt. Sie können Nachrichten an sie senden, ohne eine Definition einer fernen Warteschlange für sie vornehmen zu müssen.

Alle Definitionen sind vollständig. Starten Sie auf allen Plattformen auf jedem WS-Manager ein Empfangsprogramm. Das Listenerprogramm wartet auf eingehende Netzanforderungen und startet den Clusterempfängerkanal, wenn er benötigt wird.

## Cluster mit LU 6.2 unter z/OS einrichten

### Vorgehensweise

1. Entscheiden Sie sich für die Organisation des Clusters und dessen Namen.

Sie haben entschieden, die beiden WS-Manager LONDON und NEWYORK in einen Cluster zu verlinken. Ein Cluster mit nur zwei Warteschlangenmanagern bietet nur einen marginalen Vorteil gegenüber einem Netz, das die verteilte Steuerung von Warteschlangen verwenden soll. Es ist ein guter Weg, um zu beginnen, und es bietet Raum für zukünftige Erweiterungen. Wenn Sie neue Filialen Ihres Geschäfts öffnen, können Sie die neuen WS-Manager problemlos dem Cluster hinzufügen. Durch das Hinzufügen neuer WS-Manager wird das vorhandene Netz nicht zerrüttet; siehe „WS-Manager zu einem Cluster hinzufügen“ auf Seite 211.

Die einzige Anwendung, die Sie gerade ausführen, ist die Bestandsanwendung. Der Clustername lautet INVENTORY.

2. Entscheiden Sie, welche WS-Manager vollständige Repositorys enthalten sollen.

In jedem Cluster müssen Sie mindestens einen Warteschlangenmanager (oder vorzugsweise zwei) für die Aufnahme von vollständigen Repositorys benennen. In diesem Beispiel gibt es nur zwei WS-Manager, LONDON und NEWYORK, die beide vollständige Repositorys enthalten.

- a. Sie können die restlichen Schritte in beliebiger Reihenfolge ausführen.
- b. Eventuell werden während der Ausführung der Schritte Warnungen auf der z/OS-Systemkonsole ausgegeben. Die Nachrichten sind ein Ergebnis fehlender Definitionen, die Sie noch nicht hinzugefügt haben.
- c. Bevor Sie mit diesen Schritten fortfahren, müssen Sie sicherstellen, dass die WS-Manager gestartet wurden.

3. Ändern Sie die WS-Manager-Definitionen, um Repositorydefinitionen hinzuzufügen.

Ändern Sie auf jedem WS-Manager, der ein vollständiges Repository aufnehmen soll, die lokale WS-Manager-Definition mit dem Befehl ALTER QMGR und geben Sie das Attribut REPOS an:

```
ALTER QMGR REPOS(INVENTORY)
```

```
1 : ALTER QMGR REPOS(INVENTORY)
AMQ8005: Websphere MQ queue manager changed.
```

Geben Sie Folgendes ein, wenn Sie Folgendes eingeben:

- a. runmqsc LONDON



b. ALTER QMGR REPOS(INVENTORY)

LONDON wird in ein vollständiges Repository geändert.

4. Definieren Sie die Empfangsprogramme.

Der Listener wird nicht gestartet, wenn er definiert ist. Daher muss er mit dem folgenden MQSC-Befehl zum ersten Mal manuell gestartet werden:

```
START LISTENER(LONDON_LS)
```

Geben Sie ähnliche Befehle für alle anderen WS-Manager im Cluster ein, und ändern Sie den Namen des Listeners für die einzelnen WS-Manager.

5. Definieren Sie den CLUSRCVR-Kanal für den WS-Manager von LONDON .

Definieren Sie auf jedem Warteschlangenmanager in einem Cluster einen Clusterempfängerkanal, auf dem der Warteschlangenmanager Nachrichten empfangen kann. CLUSRCVR definiert den Verbindungsnamen des Warteschlangenmanagers. Der Verbindungsname wird in den Repositorys gespeichert, auf die sich andere Warteschlangenmanager beziehen können. Das Schlüsselwort CLUSTER zeigt die Verfügbarkeit des Warteschlangenmanagers an, um Nachrichten von anderen Warteschlangenmanagern im Cluster zu empfangen.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager LONDON')
```

```
1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager LONDON')
AMQ8014: Websphere MQ channel created.
07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
```

6. Definieren Sie den CLUSRCVR-Kanal für den WS-Manager von NEWYORK .

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNAME(NEWYORK.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager NEWYORK')
```

7. Definieren Sie den CLUSSDR-Kanal auf dem LONDON -Warteschlangenmanager.

Definieren Sie auf jedem Warteschlangenmanager in einem Cluster einen Clustersenderkanal. Der Warteschlangenmanager sendet Nachrichten an einen der Warteschlangenmanager mit vollständigem Repository im Clustersenderkanal. In diesem Fall gibt es nur zwei WS-Manager, die beide vollständige Repositorys enthalten. Sie müssen jeweils über eine CLUSSDR -Definition verfügen, die auf den CLUSRCVR -Kanal verweist, der auf dem anderen Warteschlangenmanager definiert ist. Die Kanalnamen, die in den CLUSSDR -Definitionen angegeben sind, müssen mit den Kanalnamen in den entsprechenden CLUSRCVR -Definitionen übereinstimmen. Sobald ein Warteschlangenmanager Definitionen für einen Clusterempfängerkanal und einen Clustersenderkanal in demselben Cluster hat, wird der Clustersenderkanal gestartet.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
CONNAME(CPIC) CLUSTER(INVENTORY)
DESCR('LU62 Cluster-sender channel from LONDON to repository at NEWYORK')
```

```
1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
CONNAME(NEWYORK.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-sender channel from LONDON to repository at NEWYORK')
AMQ8014: Websphere MQ channel created.
07/09/98 13:00:18 Channel program started.
```

8. Definieren Sie den CLUSSDR-Kanal auf dem NEWYORK -Warteschlangenmanager.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
CONNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
DESCR('LU62 Cluster-sender channel from NEWYORK to repository at LONDON')
```

## 9. Definieren Sie die Clusterwarteschlange INVENTQ.

Definieren Sie die INVENTQ -Warteschlange auf dem NEWYORK -Warteschlangenmanager und geben Sie dabei das Schlüsselwort CLUSTER an.

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

```
1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
AMQ8006: Websphere MQ queue created.
```

Das Schlüsselwort CLUSTER bewirkt, dass die Warteschlange für den Cluster zugänglich gemacht wird. Sobald die Warteschlange definiert ist, wird sie den anderen WS-Managern im Cluster zur Verfügung gestellt. Sie können Nachrichten an sie senden, ohne eine Definition einer fernen Warteschlange für sie vornehmen zu müssen.

Alle Definitionen sind vollständig. Starten Sie auf allen Plattformen auf jedem WS-Manager ein Empfangsprogramm. Das Listenerprogramm wartet auf eingehende Netzanforderungen und startet den Clusterempfängerkanal, wenn er benötigt wird.

### Cluster verifizieren

## Informationen zu diesem Vorgang

Sie können den Cluster auf eine oder mehrere der folgenden Arten überprüfen:

1. Ausführung von Verwaltungsbefehlen zum Anzeigen von Cluster- und Kanalattributen.
2. Führen Sie die Beispielprogramme aus, um Nachrichten in einer Clusterwarteschlange zu senden und zu empfangen.
3. Schreiben Sie Ihre eigenen Programme, um eine Anforderungsnachricht an eine Clusterwarteschlange zu senden und mit Antwortnachrichten an eine nicht in Gruppen zusammengefasste Antwortwarteschlange zu antworten.

## Vorgehensweise

Geben Sie DISPLAY **runmqsc** -Befehle aus, um den Cluster zu überprüfen.

Die Antworten, die Sie sehen, sollten wie die Antworten in den folgenden Schritten sein.

1. Führen Sie im Warteschlangenmanager NEWYORK den Befehl **DISPLAY CLUSQMGR** aus:

```
dis clusqmgr(*)
```

```
1 : dis clusqmgr(*)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(NEWYORK) CLUSTER(INVENTORY)
CHANNEL(INVENTORY.NEWYORK)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(LONDON) CLUSTER(INVENTORY)
CHANNEL(INVENTORY.LONDON)
```

2. Führen Sie im Warteschlangenmanager NEWYORK den Befehl **DISPLAY CHANNEL STATUS** aus:

```
dis chstatus(*)
```

```

1 : dis chstatus(*)
AMQ8417: Display Channel Status details.
CHANNEL(INVENTORY.NEWYORK) XMITQ()
CONNNAME(192.0.2.0) CURRENT
CHLTYPE(CLUSRCVR) STATUS(RUNNING)
RQMNAME(LONDON)
AMQ8417: Display Channel Status details.
CHANNEL(INVENTORY.LONDON) XMITQ(SYSTEM.CLUSTER.TRANSMIT.INVENTORY.LONDON)
CONNNAME(192.0.2.1) CURRENT
CHLTYPE(CLUSSDR) STATUS(RUNNING)
RQMNAME(LONDON)

```

Senden Sie Nachrichten zwischen den beiden Warteschlangenmanagern mit **amqsput**.

3. Führen Sie unter LONDON den Befehl **amqsput INVENTQ LONDON** aus.

Geben Sie einige Nachrichten ein, gefolgt von einer Leerzeile.

4. Führen Sie unter NEWYORK den Befehl **amqsget INVENTQ NEWYORK** aus.

Die Nachrichten, die Sie in LONDON eingegeben haben, werden angezeigt. Nach 15 Sekunden wird das Programm beendet.

Senden Sie Nachrichten zwischen den beiden WS-Managern, die Ihre eigenen Programme verwenden.

In den folgenden Schritten stellt LONDON eine Nachricht an die INVENTQ in NEWYORK und empfängt eine Antwort in ihrer Warteschlange LONDON\_reply.

5. Geben Sie in LONDON eine Nachricht in die Clusterwarteschlange ein.
  - a) Definieren Sie eine lokale Warteschlange mit dem Namen LONDON\_reply.
  - b) Setzen Sie die Optionen für MQOPEN auf MQOO\_OUTPUT.
  - c) Geben Sie den Aufruf MQOPEN aus, um die Warteschlange INVENTQ zu öffnen.
  - d) Setzen Sie den Namen *ReplyToQ* im Nachrichtendeskriptor auf LONDON\_reply.
  - e) Geben Sie den Aufruf MQPUT aus, um die Nachricht einzureihen.
  - f) Schreiben Sie die Nachricht fest.
6. Empfangen Sie auf NEWYORK die Nachricht in der Clusterwarteschlange und stellen Sie eine Antwort in die Antwortwarteschlange ein.
  - a) Setzen Sie die Optionen für MQOPEN auf MQOO\_BROWSE.
  - b) Geben Sie den Aufruf MQOPEN aus, um die Warteschlange INVENTQ zu öffnen.
  - c) Geben Sie den MQGET-Aufruf aus, um die Nachricht von INVENTQ abzurufen.
  - d) Rufen Sie den Namen *ReplyToQ* aus dem Nachrichtendeskriptor ab.
  - e) Geben Sie den Namen *ReplyToQ* im Feld ObjectName des Objektdeskriptors ein.
  - f) Setzen Sie die Optionen für MQOPEN auf MQOO\_OUTPUT.
  - g) Geben Sie den Aufruf MQOPEN aus, um LONDON\_reply im Warteschlangenmanager LONDON zu öffnen.
  - h) Geben Sie den Aufruf MQPUT aus, um die Nachricht in LONDON\_reply einzureihen.
7. Empfangen Sie auf LONDON die Antwort.
  - a) Setzen Sie die Optionen für MQOPEN auf MQOO\_BROWSE.
  - b) Geben Sie den Aufruf MQOPEN aus, um die Warteschlange LONDON\_reply zu öffnen.
  - c) Geben Sie den Aufruf MQGET aus, um die Nachricht von LONDON\_reply abzurufen.

## WS-Manager zu einem Cluster hinzufügen

Befolgen Sie diese Anweisungen, um dem erstellten Cluster einen WS-Manager hinzuzufügen. Nachrichten zu Clusterwarteschlangen und Themen werden unter Verwendung der einzigen Clusterübertragungswarteschlange SYSTEM.CLUSTER.TRANSMIT.QUEUE übertragen.

## Vorbereitende Schritte

**Anmerkung:** Damit Änderungen an einem Cluster im gesamten Cluster weitergegeben werden können, muss immer mindestens ein vollständiges Repository verfügbar sein. Stellen Sie sicher, dass Ihre Repositories verfügbar sind, bevor Sie diese Task starten.

Szenario:

- Der INVENTORY -Cluster wird wie in „Neuen Cluster einrichten“ auf Seite 201 beschrieben konfiguriert. Es enthält zwei WS-Manager, LONDON und NEWYORK, die beide vollständige Repositories enthalten.
- Der WS-Manager PARIS ist Eigentum der primären Installation. Ist dies nicht der Fall, müssen Sie den Befehl `setmqenv` ausführen, um die Befehlsumgebung für die Installation einzurichten, zu der PARIS gehört.
- Die TCP-Verbindung besteht zwischen allen drei Systemen, und der Warteschlangenmanager wird mit einem TCP-Listener konfiguriert, der unter der Steuerung des Warteschlangenmanagers gestartet wird.

## Informationen zu diesem Vorgang

1. Eine neue Filiale des Filialgeschäftlers wird in Paris eingerichtet und Sie möchten dem Cluster einen WS-Manager mit dem Namen PARIS hinzufügen.
2. Der Warteschlangenmanager PARIS sendet Bestandsaktualisierungen an die Anwendung, die auf dem System in New York ausgeführt wird, indem Nachrichten in die Warteschlange von INVENTQ gestellt werden.

Führen Sie die folgenden Schritte aus, um einen WS-Manager zu einem Cluster hinzuzufügen.

## Vorgehensweise

1. Entscheiden Sie, welches vollständige Repository PARIS auf das erste Element verweist.

Jeder WS-Manager in einem Cluster muss sich auf einen oder einen anderen der vollständigen Repositories beziehen. Es sammelt Informationen über den Cluster aus einem vollständigen Repository und erstellt so ein eigenes Teilrepository. Wählen Sie eines der Repositories als vollständiges Repository aus. Sobald ein neuer WS-Manager dem Cluster hinzugefügt wird, wird er sofort auch über das andere Repository informiert. Informationen zu Änderungen an einem WS-Manager werden direkt an zwei Repositories gesendet. In diesem Beispiel verbinden Sie PARIS mit dem Warteschlangenmanager LONDON aus rein geographischen Gründen.

**Anmerkung:** Führen Sie die verbleibenden Schritte in beliebiger Reihenfolge aus, nachdem der WS-Manager PARIS gestartet wurde.

2. Definieren Sie einen CLUSRCVR -Kanal auf WS-Manager PARIS.

Jeder WS-Manager in einem Cluster muss einen Clusterempfängerkanal definieren, auf dem er Nachrichten empfangen kann. Definieren Sie unter PARIS Folgendes:

```
DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for queue manager PARIS')
```

Der Clusterempfängerkanal wirbt für die Verfügbarkeit des Warteschlangenmanagers, um Nachrichten von anderen Warteschlangenmanagern im Cluster INVENTORY zu empfangen. Es ist nicht erforderlich, Definitionen auf anderen Warteschlangenmanagern für ein sendende Ende an den Clusterempfängerkanal INVENTORY . PARIS zu setzen. Andere Definitionen werden bei Bedarf automatisch erstellt.

3. Definieren Sie einen CLUSSDR -Kanal auf WS-Manager PARIS.

Jeder Warteschlangenmanager in einem Cluster muss einen Clustersenderkanal definieren, über den er Nachrichten an sein erstes vollständiges Repository senden kann.

Erstellen Sie unter PARIS die folgende Definition für einen Kanal mit dem Namen INVENTORY.LONDON zum Warteschlangenmanager mit der Netzadresse LONDON.CHSTORE.COM.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at LONDON')
```

4. Optional: Wenn dieser Warteschlangenmanager wieder in einen Cluster aufgenommen wird, führen Sie einige zusätzliche Schritte aus.

a) Wenn Sie einen Warteschlangenmanager zu einem Cluster hinzufügen, der zuvor aus demselben Cluster entfernt wurde, überprüfen Sie, ob er jetzt als Cluster-Member angezeigt wird. Ist dies nicht der Fall, führen Sie die folgenden zusätzlichen Schritte aus:

i) Geben Sie den Befehl **REFRESH CLUSTER** auf dem Warteschlangenmanager aus, den Sie hinzufügen. Dieser Schritt stoppt die Clusterkanäle und gibt Ihrem lokalen Clustercache eine neue Gruppe von Folgeummern, die sichergestellt werden, dass sie im Rest des Clusters für das Up-to-Datum konfiguriert werden.

```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

**Anmerkung:** Bei großen Clustern kann der Befehl **REFRESH CLUSTER** während seiner Ausführung und danach in 27-Tage-Intervallen, wenn die Clusterobjekte ihre Statusaktualisierungen automatisch an alle interessierten Warteschlangenmanager hochladen, zu Unterbrechungen führen. Nähere Informationen hierzu erhalten Sie im Abschnitt Die Aktualisierung in einem großen Cluster kann sich auf die Leistung und Verfügbarkeit auswirken.

ii) Starten Sie den CLUSSDR-Kanal erneut (z. B. mit dem Befehl START CHANNEL).

iii) Starten Sie den CLUSRCVR-Kanal erneut.

b) Wenn es sich bei dem Cluster um einen Publish/Subscribe-Cluster handelt und der erneut zu verknüpfende Warteschlangenmanager über Subskriptionen verfügt, geben Sie den folgenden Befehl aus, um sicherzustellen, dass die Proxy-Subskriptionen im Cluster ordnungsgemäß synchronisiert sind:

```
REFRESH QMGR TYPE(PROXYSUB)
```

## Ergebnisse

In der folgenden Abbildung ist der Cluster dargestellt, der von dieser Task eingerichtet wurde.

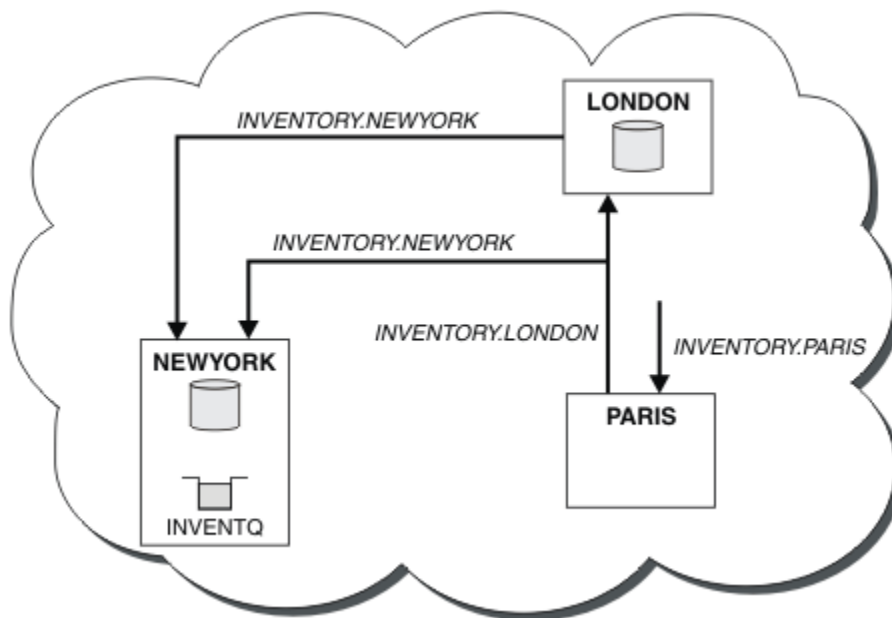


Abbildung 30. Der INVENTORY -Cluster mit drei Warteschlangenmanagern

Wenn Sie nur zwei Definitionen, eine CLUSRCVR -Definition und eine CLUSSDR -Definition, haben, haben wir den WS-Manager PARIS dem Cluster hinzugefügt.

Jetzt lernt der PARIS -Warteschlangenmanager aus dem vollständigen Repository in LONDON, dass die INVENTQ -Warteschlange vom WS-Manager NEWYORK gehostet wird. Wenn eine von dem System in Paris gehostete Anwendung versucht, Nachrichten an den INVENTQ zu stellen, definiert PARIS automatisch einen Clustersenderkanal, um eine Verbindung zum Clusterempfängerkanal INVENTORY . NEWYORK herzustellen. Die Anwendung kann Antworten empfangen, wenn ihr Warteschlangenmanager-Name als Zielwarteschlangenmanager und eine Warteschlange für Antwortantworten angegeben ist.

## WS-Manager zu einem Cluster hinzufügen: separate Übertragungswarteschlangen

Befolgen Sie diese Anweisungen, um dem erstellten Cluster einen WS-Manager hinzuzufügen. Nachrichten zu Clusterwarteschlangen und Themen werden unter Verwendung mehrerer Clusterübertragungswarteschlangen übertragen.

### Vorbereitende Schritte

- Der Warteschlangenmanager ist auf einer anderen Plattform als z/OS definiert.
- Der WS-Manager ist kein Member von Clustern.
- Der Cluster ist vorhanden. Es gibt ein vollständiges Repository, zu dem dieser WS-Manager eine direkte Verbindung herstellen kann und das Repository verfügbar ist. Informationen zu den Schritten zum Erstellen des Clusters finden Sie in „[Neuen Cluster einrichten](#)“ auf Seite 201.

### Informationen zu diesem Vorgang

Diese Task ist eine Alternative zu „[WS-Manager zu einem Cluster hinzufügen](#)“ auf Seite 211, in der Sie einen WS-Manager zu einem Cluster hinzufügen, der Clusternachrichten in eine einzelne Übertragungswarteschlange stellt.

In dieser Task fügen Sie einen Warteschlangenmanager zu einem Cluster hinzu, der automatisch separate Clusterübertragungswarteschlangen für jeden Clustersenderkanal erstellt.

Um die Anzahl der Definitionen von Warteschlangen klein zu halten, wird standardmäßig eine einzelne Übertragungswarteschlange verwendet. Die Verwendung separater Übertragungswarteschlangen ist von

Vorteil, wenn Sie den Datenverkehr überwachen wollen, der für verschiedene Warteschlangenmanager und verschiedene Cluster bestimmt ist. Zudem kann auch die Trennung von Datenverkehr nach Empfängern wünschenswert sein, um diesen zu isolieren oder bestimmte Leistungsziele zu erreichen.

## Vorgehensweise

1. Ändern Sie den Standardtyp der Übertragungswarteschlange des Clusterkanals.

Ändern Sie den WS-Manager PARIS:

```
ALTER QMGR DEFCLXQ(CHANNEL)
```

Jedesmal, wenn der WS-Manager einen Clustersenderkanal erstellt, um eine Nachricht an einen Warteschlangenmanager zu senden, wird eine Clusterübertragungswarteschlange erstellt. Die Übertragungswarteschlange wird nur von diesem Clustersenderkanal verwendet. Die Übertragungswarteschlange ist permanent dynamisch. Er wird aus der Modellwarteschlange SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE mit dem Namen SYSTEM . CLUSTER . TRANSMIT . *ChannelName* erstellt.



**Achtung:** Wenn Sie dedizierte SYSTEM . CLUSTER . TRANSMIT . QUEUES mit einem Warteschlangenmanager verwenden, für den ein Upgrade von einer früheren Version des Produkts durchgeführt wurde, stellen Sie sicher, dass für SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE die Option SHARE/NOSHARE auf **SHARE** gesetzt ist.

2. Entscheiden Sie, welches vollständige Repository PARIS auf das erste Element verweist.

Jeder WS-Manager in einem Cluster muss sich auf einen oder einen anderen der vollständigen Repositories beziehen. Es sammelt Informationen über den Cluster aus einem vollständigen Repository und erstellt so ein eigenes Teilrepository. Wählen Sie eines der Repositories als vollständiges Repository aus. Sobald ein neuer WS-Manager dem Cluster hinzugefügt wird, wird er sofort auch über das andere Repository informiert. Informationen zu Änderungen an einem WS-Manager werden direkt an zwei Repositories gesendet. In diesem Beispiel verbinden Sie PARIS mit dem Warteschlangenmanager LONDON aus rein geographischen Gründen.

**Anmerkung:** Führen Sie die verbleibenden Schritte in beliebiger Reihenfolge aus, nachdem der WS-Manager PARIS gestartet wurde.

3. Definieren Sie einen CLUSRCVR -Kanal auf WS-Manager PARIS.

Jeder WS-Manager in einem Cluster muss einen Clusterempfängerkanal definieren, auf dem er Nachrichten empfangen kann. Definieren Sie unter PARIS Folgendes:

```
DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for queue manager PARIS')
```

Der Clusterempfängerkanal wirbt für die Verfügbarkeit des Warteschlangenmanagers, um Nachrichten von anderen Warteschlangenmanagern im Cluster INVENTORY zu empfangen. Es ist nicht erforderlich, Definitionen auf anderen Warteschlangenmanagern für ein sendende Ende an den Clusterempfängerkanal INVENTORY . PARIS zu setzen. Andere Definitionen werden bei Bedarf automatisch erstellt.

4. Definieren Sie einen CLUSSDR -Kanal auf WS-Manager PARIS.

Jeder Warteschlangenmanager in einem Cluster muss einen Clustersenderkanal definieren, über den er Nachrichten an sein erstes vollständiges Repository senden kann.

Erstellen Sie unter PARIS die folgende Definition für einen Kanal mit dem Namen INVENTORY . LONDON zum Warteschlangenmanager mit der Netzadresse LONDON . CHSTORE . COM.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at LONDON')
```

Der WS-Manager erstellt automatisch die permanente dynamische Clusterübertragungswarteschlange SYSTEM . CLUSTER . TRANSMIT . INVENTORY . LONDON aus der Modellwarteschlange SYSTEM . CLUS-

TER . TRANSMIT . MODEL . QUEUE. Sie setzt das Attribut CLCHNAME der Übertragungswarteschlange auf INVENTORY . LONDON.

## Ergebnisse

In der folgenden Abbildung ist der Cluster dargestellt, der von dieser Task eingerichtet wurde.

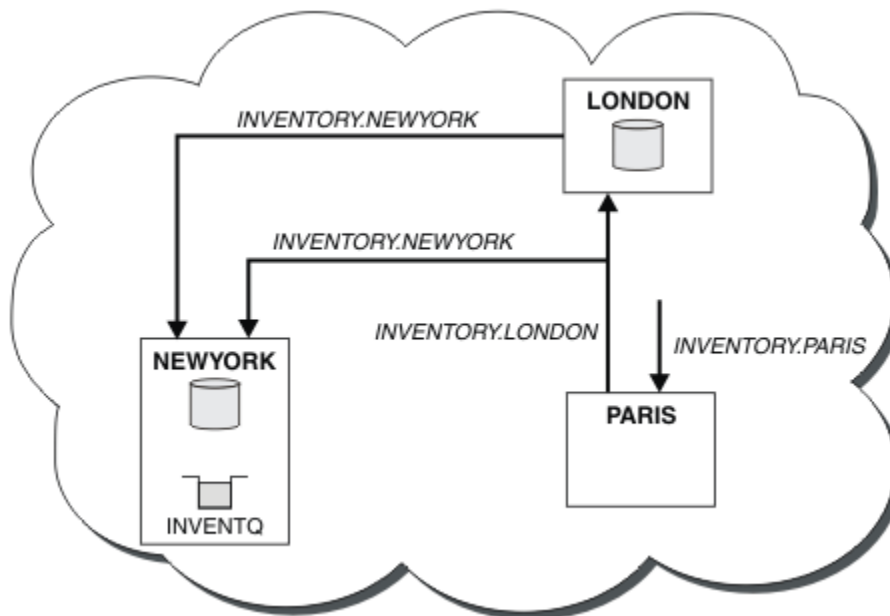


Abbildung 31. Der INVENTORY -Cluster mit drei Warteschlangenmanagern

Wenn Sie nur zwei Definitionen, eine CLUSRCVR -Definition und eine CLUSSDR -Definition, haben, haben wir den WS-Manager PARIS dem Cluster hinzugefügt.

Jetzt lernt der PARIS -Warteschlangenmanager aus dem vollständigen Repository in LONDON, dass die INVENTQ -Warteschlange vom WS-Manager NEWYORK gehostet wird. Wenn eine von dem System in Paris gehostete Anwendung versucht, Nachrichten an den INVENTQ zu stellen, definiert PARIS automatisch einen Clustersenderkanal, um eine Verbindung zum Clusterempfängerkanal INVENTORY . NEWYORK herzustellen. Die Anwendung kann Antworten empfangen, wenn ihr Warteschlangenmanager-Name als Zielwarteschlangenmanager und eine Warteschlange für Antwortantworten angegeben ist.

## Definition einer fernen Warteschlange hinzufügen, um Nachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet wurden

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine ferne Definition einer Clusterwarteschlange und einen separaten Senderkanal und eine separate Übertragungswarteschlange.

## Vorbereitende Schritte

Erstellen Sie die in Abbildung 37 auf Seite 236 in „Erstellen von zwei überlappenden Clustern mit einem Gateway-Warteschlangenmanager“ auf Seite 235 gezeigten überlappenden Cluster, indem Sie die Schritte in dieser Task ausführen.

## Informationen zu diesem Vorgang

Die Lösung verwendet die verteilte Steuerung von Warteschlangen, um die Nachrichten für die Server App -Anwendung von anderen Nachrichtenübertragungen auf dem Gateway-Warteschlangenmanager zu



trennen. Sie müssen eine Definition einer fernen Clusterwarteschlange auf QM1 definieren, um die Nachrichten in eine andere Übertragungswarteschlange und einen anderen Kanal umzuleiten. Die Definition der fernen Warteschlange muss einen Verweis auf die spezifische Übertragungswarteschlange enthalten, in der Nachrichten nur für Q1 auf QM3 gespeichert werden. In [Abbildung 32](#) auf Seite 217 wird der Aliasname der Clusterwarteschlange Q1A durch eine ferne Warteschlangendefinition Q1R sowie eine Übertragungswarteschlange und einen Senderkanal ergänzt.

In dieser Lösung werden alle Antwortnachrichten unter Verwendung der allgemeinen SYSTEM.CLUSTER.TRANSMIT.QUEUE zurückgegeben.

Der Vorteil dieser Lösung ist, dass es leicht ist, den Datenverkehr für mehrere Zielwarteschlangen auf demselben WS-Manager im selben Cluster zu trennen. Der Nachteil der Lösung besteht darin, dass Sie keinen Cluster-Workload-Ausgleich zwischen mehreren Kopien von Q1 auf verschiedenen Warteschlangenmanagern verwenden können. Informationen zum Überwinden dieses Nachteils finden Sie in [„Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde“](#) auf Seite 219. Außerdem müssen Sie den Switch von einer Übertragungswarteschlange in die andere verwalten.

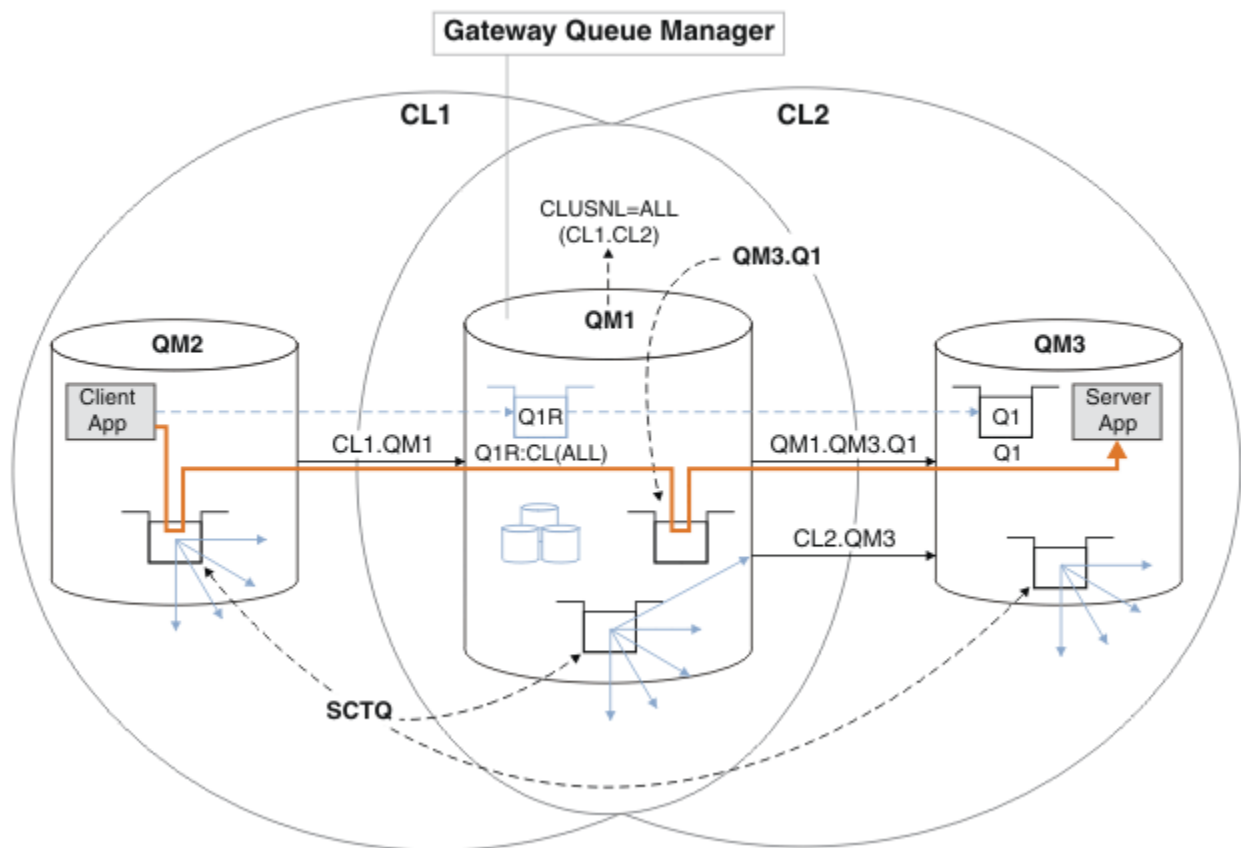


Abbildung 32. Anwendung 'Client-Server' in Hub-und Spoke-Cluster-Architektur unter Verwendung ferner Warteschlangendefinitionen implementiert

## Vorgehensweise

1. Erstellen Sie einen Kanal, um den Nachrichtenverkehr für Q1 vom Gateway-Warteschlangenmanager zu trennen.
  - a) Erstellen Sie einen Senderkanal auf dem Gateway-Warteschlangenmanager, QM1, an den Zielwarteschlangenmanager QM3.

```
DEFINE CHANNEL(QM1.QM3.Q1) CHLTYPE(SDR) CONNAME(QM3HostName(1413)) XMITQ(QM3.Q1) REPLACE
```

- b) Erstellen Sie einen Empfängerkanal auf dem Zielwarteschlangenmanager, QM3.

```
DEFINE CHANNEL(QM1.QM3.Q1) CHLTYPE(RCVR) REPLACE
```

2. Erstellen Sie eine Übertragungswarteschlange auf dem Gateway-Warteschlangenmanager für den Nachrichtenverkehr in Q1.

```
DEFINE QLOCAL(QM3.Q1) USAGE(XMITQ) REPLACE
START CHANNEL(QM1.QM3.Q1)
```

Wenn der Kanal gestartet wird, der der Übertragungswarteschlange zugeordnet ist, wird die Übertragungswarteschlange dem Kanal zugeordnet. Der Kanal wird automatisch gestartet, sobald die Übertragungswarteschlange dem Kanal zugeordnet wurde.

3. Ergänzen Sie die Clusterwarteschlangenaliasdefinition für Q1 auf dem Gateway-Warteschlangenmanager mit einer Definition der fernen Clusterwarteschlange.

```
DEFINE QREMOTE CLUSNL(ALL) RNAME(Q1) RQMNAME(QM3) XMITQ(QM3.Q1) REPLACE
```

## Nächste Schritte

Testen Sie die Konfiguration, indem Sie eine Nachricht an Q1 unter QM3 von QM2 unter Verwendung der Clusterwarteschlangendefinition Q1R auf dem Gateway-WS-Manager QM1 senden.

1. Führen Sie das Beispielprogramm **amqspout** unter QM2 aus, um eine Nachricht einzureihen.

```
C:\IBM\MQ>amqspout Q1R QM2
Sample AMQSPUT0 start
target queue is Q1R
Sample request message from QM2 to Q1 using Q1R
```

```
Sample AMQSPUT0 end
```

2. Führen Sie das Beispielprogramm **amqsget** aus, um die Nachricht von Q1 unter QM3 abzurufen

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1R>
no more messages
Sample AMQSGET0 end
```

## Zugehörige Konzepte

„[Zugriffssteuerung und mehrere Clusterübertragungswarteschlangen](#)“ auf Seite 173

Wählen Sie zwischen drei Prüfmodi aus, wenn eine Anwendung Nachrichten in ferne Clusterwarteschlangen einreicht. Die Modi werden über Remotezugriff auf die Clusterwarteschlange überprüft, lokal auf SYSTEM.CLUSTER.TRANSMIT.QUEUE geprüft oder auf lokale Profile für die Clusterwarteschlange oder den Cluster-WS-Manager überprüft.

„[Clustering: Anwendungsisolierung mit mehreren Clusterübertragungswarteschlangen](#)“ auf Seite 306

Sie können die Nachrichtenflüsse zwischen Warteschlangenmanagern in einem Cluster isolieren. Sie können Nachrichten, die von verschiedenen Clustersenderkanälen transportiert werden, in verschiedene Clusterübertragungswarteschlangen stellen. Sie können den Ansatz in einem einzelnen Cluster oder mit überlappenden Clustern verwenden. Das Thema enthält Beispiele und einige bewährte Verfahren, die Sie bei der Auswahl eines zu verwendenden Ansatzes führen.

## Zugehörige Tasks

„[WS-Manager zu einem Cluster hinzufügen: separate Übertragungswarteschlangen](#)“ auf Seite 214

Befolgen Sie diese Anweisungen, um dem erstellten Cluster einen WS-Manager hinzuzufügen. Nachrichten zu Clusterwarteschlangen und Themen werden unter Verwendung mehrerer Clusterübertragungswarteschlangen übertragen.

„[Erstellen von zwei überlappenden Clustern mit einem Gateway-Warteschlangenmanager](#)“ auf Seite 235

Befolgen Sie die Anweisungen in der Task, um überlappende Cluster mit einem Gateway-Warteschlangenmanager zu erstellen. Verwenden Sie die Cluster als Ausgangspunkt für die folgenden Beispiele, um Nachrichten in einer Anwendung von Nachrichten an andere Anwendungen in einem Cluster zu isolieren.

„Definition einer fernen Warteschlange hinzufügen, um Nachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet wurden“ auf Seite 216

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine ferne Definition einer Clusterwarteschlange und einen separaten Senderkanal und eine separate Übertragungswarteschlange.

„Ändern der Standardeinstellung in separate Clusterübertragungswarteschlangen, um den Nachrichten-  
datenverkehr zu isolieren“ auf Seite 241

Sie können die Standardweise ändern, in der ein WS-Manager Nachrichten für eine Clusterwarteschlange oder ein Topic in einer Übertragungswarteschlange speichert. Wenn Sie den Standardwert ändern, können Sie Clusternachrichten auf einem Gateway-Warteschlangenmanager isolieren.

„Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde“ auf Seite 219

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine zusätzliche Clusterübertragungswarteschlange, um den Nachrichtenverkehr auf einen einzelnen Warteschlangenmanager in einem Cluster zu trennen.

„Cluster und Cluster-Übertragungswarteschlange hinzufügen, um den Datenverkehr der Clusternachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet werden“ auf Seite 224

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet einen zusätzlichen Cluster, um die Nachrichten in einer bestimmten Clusterwarteschlange zu isolieren.

„Clustering: Planung der Konfiguration von Clusterübertragungswarteschlangen“ auf Seite 310

Sie werden durch die Auswahl von Clusterübertragungswarteschlangen geführt. Sie können eine gemeinsame Standardwarteschlange, separate Standardwarteschlangen oder manuell definierte Warteschlangen konfigurieren. Die Konfiguration mehrerer Clusterübertragungswarteschlangen gilt für andere Plattformen als z/OS.

## **Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde**

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine zusätzliche Clusterübertragungswarteschlange, um den Nachrichtenverkehr auf einen einzelnen Warteschlangenmanager in einem Cluster zu trennen.

### **Vorbereitende Schritte**

1. Der Gateway-Warteschlangenmanager muss sich unter Version 7.5 oder höher und auf einer anderen Plattform als z/OS befinden.
2. Erstellen Sie die in Abbildung 37 auf Seite 236 in „Erstellen von zwei überlappenden Clustern mit einem Gateway-Warteschlangenmanager“ auf Seite 235 gezeigten überlappenden Cluster, indem Sie die Schritte in dieser Task ausführen.

### **Informationen zu diesem Vorgang**

Fügen Sie auf dem Gateway-Warteschlangenmanager QM1 eine Übertragungswarteschlange hinzu, und legen Sie das zugehörige Warteschlangenattribut CLCHNAME fest. Setzen Sie CLCHNAME auf den Namen des Clusterempfängerkanals unter QM3; siehe Abbildung 33 auf Seite 221.

Diese Lösung bietet eine Reihe von Vorteilen gegenüber der Lösung, die in „Definition einer fernen Warteschlange hinzufügen, um Nachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet wurden“ auf Seite 216 beschrieben wird:

- Sie erfordert weniger zusätzliche Definitionen.
- Er unterstützt den Lastausgleich zwischen mehreren Kopien der Zielwarteschlange, Q1, auf verschiedenen WS-Managern in demselben Cluster, CL2.
- Der Gateway-Warteschlangenmanager wechselt automatisch in die neue Konfiguration, wenn der Kanal neu gestartet wird, ohne dass Nachrichten zu verlieren sind.
- Der Warteschlangenmanager des Gateways setzt die Nachrichten in derselben Reihenfolge fort, in der er sie empfangen hat. Dies geschieht auch dann, wenn der Wechsel mit Nachrichten für die Warteschlange Q1 bei QM3 noch unter SYSTEM . CLUSTER . TRANSMIT . QUEUE erfolgt.

Die Konfiguration zum Isolieren des Clusternachrichtenverkehrs in Abbildung 33 auf Seite 221 führt nicht zu einer starken Isolation des Datenverkehrs als die Konfiguration, die ferne Warteschlangen in „Definition einer fernen Warteschlange hinzufügen, um Nachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet wurden“ auf Seite 216 verwendet. Wenn der Warteschlangenmanager QM3 in CL2 eine Reihe unterschiedlicher Clusterwarteschlangen und Serveranwendungen hostet, nutzen alle diese Warteschlangen den Clusterkanal CL2 . QM3 und verbinden QM1 mit QM3. Die zusätzlichen Flüsse werden in Abbildung 33 auf Seite 221 durch den grauen Pfeil dargestellt, der den potenziellen Clusternachrichtenverkehr von der SYSTEM . CLUSTER . TRANSMIT . QUEUE zum Clustersenderkanal darstellt CL2 . QM3.

Die Abhilfe besteht darin, den WS-Manager auf die Aufnahme einer Clusterwarteschlange in einem bestimmten Cluster zu beschränken. Wenn der Warteschlangenmanager bereits eine Reihe von Clusterwarteschlangen bereitstellt, müssen Sie diese Einschränkung erfüllen, indem Sie entweder einen anderen Warteschlangenmanager erstellen oder einen anderen Cluster erstellen. Weitere Informationen finden Sie im Abschnitt „Cluster und Cluster-Übertragungswarteschlange hinzufügen, um den Datenverkehr der Clusternachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet werden“ auf Seite 224.

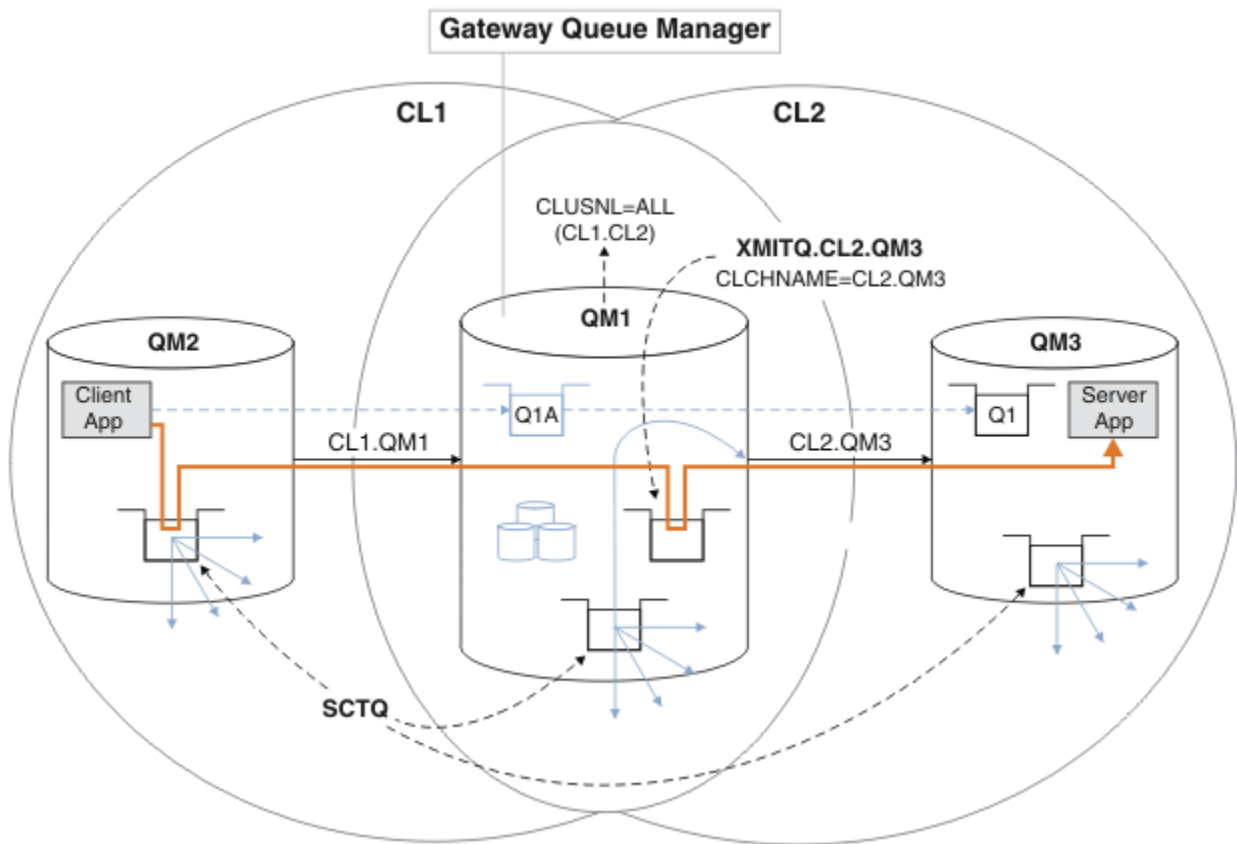


Abbildung 33. Die Client-Server-Anwendung wurde in Hub-und Spoke-Architektur unter Verwendung einer zusätzlichen Clusterübertragungswarteschlange implementiert.

## Vorgehensweise

1. Erstellen Sie eine zusätzliche Clusterübertragungswarteschlange für den Clustersenderkanal CL2.QM3 auf dem Gateway-WS-Manager, QM1.

```
*... on QM1
DEFINE QLOCAL(XMITQ.CL2.QM3) USAGE(XMITQ) CLCHNAME(CL2.QM3)
```

2. Wechseln Sie zur Verwendung der Übertragungswarteschlange XMITQ.CL2.QM3.

- a) Stoppen Sie den Clustersenderkanal CL2.QM3.

```
*... On QM1
STOP CHANNEL(CL2.QM3)
```

Die Antwort ist, dass der Befehl akzeptiert wird:

```
AMQ8019: Stop WebSphere MQ channel accepted.
```

- b) Überprüfen Sie, ob der Kanal CL2.QM3 gestoppt wurde.

Wenn der Kanal nicht gestoppt wird, können Sie den Befehl **STOP CHANNEL** erneut mit der Option **FORCE** ausführen. Ein Beispiel für die Einstellung der Option **FORCE** wäre, wenn der Kanal nicht gestoppt wird, und Sie können den anderen Warteschlangenmanager nicht erneut starten, um den Kanal zu synchronisieren.

```
*... On QM1
start
```

Die Antwort ist eine Zusammenfassung des Kanalstatus.

```
AMQ8417: Display Channel Status details.
CHANNEL (CL2.QM3) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1413)) CURRENT
RQMNAME (QM3) STATUS (STOPPED)
SUBSTATE (MQGET) XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

c) Starten Sie den Kanal CL2.QM3.

```
*... On QM1
START CHANNEL (CL2.QM3)
```

Die Antwort ist, dass der Befehl akzeptiert wird:

```
AMQ8018: Start WebSphere MQ channel accepted.
```

d) Überprüfen Sie, ob der Kanal gestartet wurde

```
*... On QM1
DISPLAY CHSTATUS (CL2.QM3)
```

Die Antwort ist eine Zusammenfassung des Kanalstatus:

```
AMQ8417: Display Channel Status details.
CHANNEL (CL2.QM3) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1413)) CURRENT
RQMNAME (QM3) STATUS (RUNNING)
SUBSTATE (MQGET) XMITQ (XMITQ.CL2.QM3)
```

e) Überprüfen Sie, ob die Übertragungswarteschlange gewechselt wurde.

Überwachen Sie das Fehlerprotokoll des Gateway-Warteschlangenmanagers auf die Nachricht "AMQ7341 Die Übertragungswarteschlange für Kanal CL2.QM3 ist XMITQ.CL2.QM3".

## Nächste Schritte

Testen Sie die separate Übertragungswarteschlange, indem Sie eine Nachricht von QM2 an Q1 auf QM3 unter Verwendung der Warteschlangenaliasdefinition Q1A senden.

1. Führen Sie das Beispielprogramm **amqspout** unter QM2 aus, um eine Nachricht einzureihen.

```
C:\IBM\MQ>amqspout Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A
```

```
Sample AMQSPUT0 end
```

2. Führen Sie das Beispielprogramm **amqsget** aus, um die Nachricht von Q1 unter QM3 abzurufen

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

## Zugehörige Konzepte

[„Zugriffssteuerung und mehrere Clusterübertragungswarteschlangen“ auf Seite 173](#)

Wählen Sie zwischen drei Prüfmodi aus, wenn eine Anwendung Nachrichten in ferne Clusterwarteschlangen einreicht. Die Modi werden über Remotezugriff auf die Clusterwarteschlange überprüft, lokal auf SYSTEM.CLUSTER.TRANSMIT.QUEUE geprüft oder auf lokale Profile für die Clusterwarteschlange oder den Cluster-WS-Manager überprüft.

[„Clustering: Anwendungsisolierung mit mehreren Clusterübertragungswarteschlangen“ auf Seite 306](#)

Sie können die Nachrichtenflüsse zwischen Warteschlangenmanagern in einem Cluster isolieren. Sie können Nachrichten, die von verschiedenen Clustersenderkanälen transportiert werden, in verschiedene Clusterübertragungswarteschlangen stellen. Sie können den Ansatz in einem einzelnen Cluster oder mit überlappenden Clustern verwenden. Das Thema enthält Beispiele und einige bewährte Verfahren, die Sie bei der Auswahl eines zu verwendenden Ansatzes führen.

„Clusterübertragungswarteschlangen und Clustersenderkanäle“ auf Seite 185

Nachrichten zwischen Clustering-WS-Managern werden in Clusterübertragungswarteschlangen gespeichert und von Clustersenderkanälen weitergeleitet.

### **Zugehörige Tasks**

„WS-Manager zu einem Cluster hinzufügen: separate Übertragungswarteschlangen“ auf Seite 214

Befolgen Sie diese Anweisungen, um dem erstellten Cluster einen WS-Manager hinzuzufügen. Nachrichten zu Clusterwarteschlangen und Themen werden unter Verwendung mehrerer Clusterübertragungswarteschlangen übertragen.

„Erstellen von zwei überlappenden Clustern mit einem Gateway-Warteschlangenmanager“ auf Seite 235

Befolgen Sie die Anweisungen in der Task, um überlappende Cluster mit einem Gateway-Warteschlangenmanager zu erstellen. Verwenden Sie die Cluster als Ausgangspunkt für die folgenden Beispiele, um Nachrichten in einer Anwendung von Nachrichten an andere Anwendungen in einem Cluster zu isolieren.

„Definition einer fernen Warteschlange hinzufügen, um Nachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet wurden“ auf Seite 216

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine ferne Definition einer Clusterwarteschlange und einen separaten Senderkanal und eine separate Übertragungswarteschlange.

„Ändern der Standardeinstellung in separate Clusterübertragungswarteschlangen, um den Nachrichtenverkehr zu isolieren“ auf Seite 241

Sie können die Standardweise ändern, in der ein WS-Manager Nachrichten für eine Clusterwarteschlange oder ein Topic in einer Übertragungswarteschlange speichert. Wenn Sie den Standardwert ändern, können Sie Clusternachrichten auf einem Gateway-Warteschlangenmanager isolieren.

„Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde“ auf Seite 219

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine zusätzliche Clusterübertragungswarteschlange, um den Nachrichtenverkehr auf einen einzelnen Warteschlangenmanager in einem Cluster zu trennen.

„Cluster und Cluster-Übertragungswarteschlange hinzufügen, um den Datenverkehr der Clusternachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet werden“ auf Seite 224

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet einen zusätzlichen Cluster, um die Nachrichten in einer bestimmten Clusterwarteschlange zu isolieren.

„Clustering: Planung der Konfiguration von Clusterübertragungswarteschlangen“ auf Seite 310

Sie werden durch die Auswahl von Clusterübertragungswarteschlangen geführt. Sie können eine gemeinsame Standardwarteschlange, separate Standardwarteschlangen oder manuell definierte Warteschlangen konfigurieren. Die Konfiguration mehrerer Clusterübertragungswarteschlangen gilt für andere Plattformen als z/OS.



## Cluster und Cluster-Übertragungswarteschlange hinzufügen, um den Datenverkehr der Clusternachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet werden

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet einen zusätzlichen Cluster, um die Nachrichten in einer bestimmten Clusterwarteschlange zu isolieren.

### Vorbereitende Schritte

Die Schritte in der Task werden geschrieben, um die in [Abbildung 33 auf Seite 221](#) dargestellte Konfiguration zu ändern.

1. Der Gateway-Warteschlangenmanager muss sich unter Version 7.5 oder höher und auf einer anderen Plattform als z/OS befinden.
2. Erstellen Sie die in [Abbildung 37 auf Seite 236](#) in „Erstellen von zwei überlappenden Clustern mit einem Gateway-Warteschlangenmanager“ auf [Seite 235](#) gezeigten überlappenden Cluster, indem Sie die Schritte in dieser Task ausführen.
3. Führen Sie die Schritte unter [Abbildung 33 auf Seite 221](#) in „[Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde](#)“ auf [Seite 219](#) aus, um die Lösung ohne den zusätzlichen Cluster zu erstellen. Verwenden Sie diese Option als Basis für die Schritte in dieser Task.

### Informationen zu diesem Vorgang

Die Lösung zum Eingrenzen des Nachrichtenverkehrs auf eine einzelne Anwendung in „[Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde](#)“ auf [Seite 219](#) funktioniert, wenn die Ziel-Clusterwarteschlange die einzige Clusterwarteschlange auf einem Warteschlangenmanager ist. Wenn dies nicht der Fall ist, haben Sie zwei Möglichkeiten. Verschieben Sie die Warteschlange entweder in einen anderen Warteschlangenmanager oder erstellen Sie einen Cluster, der die Warteschlange aus anderen Clusterwarteschlangen auf dem Warteschlangenmanager isoliert.

Diese Task führt Sie durch die Schritte zum Hinzufügen eines Clusters zum Isolieren der Zielwarteschlange. Der Cluster wird zu diesem Zweck hinzugefügt. In der Praxis ist es die Aufgabe, bestimmte Anwendungen systematisch zu isolieren, wenn Sie Cluster- und Clusterbenennungsschemata entwerfen. Wenn Sie einen Cluster jedes Mal hinzufügen, wenn eine Warteschlange isoliert wird, kann die Isolation möglicherweise mit vielen Clustern enden. In dieser Task ändern Sie die Konfiguration in „[Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde](#)“ auf [Seite 219](#), indem Sie einen Cluster CL3 hinzufügen, um Q1 auf QM3 zu isolieren. Anwendungen werden während der gesamten Änderung weiterhin ausgeführt.

Die neuen und geänderten Definitionen werden in [Abbildung 34 auf Seite 225](#) hervorgehoben. Die Zusammenfassung der Änderungen lautet wie folgt: Erstellen Sie einen Cluster. Dies bedeutet, dass Sie auch ein neues vollständiges Clusterrepository erstellen müssen. Im Beispiel wird QM3 zu einem der vollständigen Repositories für CL3 gemacht. Erstellen Sie Clustersender- und Clusterempfängerkanäle für QM1, um den Gateway-Warteschlangenmanager zum neuen Cluster hinzuzufügen. Ändern Sie die Definition von Q1, um sie in CL3 zu wechseln. Ändern Sie die Clusternamensliste auf dem Gateway-Warteschlangenmanager, und fügen Sie eine Clusterübertragungswarteschlange hinzu, um den neuen Clusterkanal zu verwenden. Schalten Sie schließlich den Warteschlangenalias Q1A in die neue Clusternamensliste ein.

IBM WebSphere MQ kann keine Nachrichten aus der Übertragungswarteschlange XMITQ.CL2.QM3, die Sie in „[Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde](#)“ auf [Seite 219](#) hinzugefügt haben, automatisch in die neue Übertragungswarteschlange XMITQ.CL3.QM3 übertragen. Es kann Nachrichten nur dann automatisch übertragen, wenn beide Übertragungswarteschlangen vom selben Clustersenderkanal bedient werden. Stattdessen beschreibt die Task eine Möglichkeit, den Switch manuell auszuführen.



Dies kann für Sie geeignet sein. Wenn die Übertragung abgeschlossen ist, haben Sie die Möglichkeit, die Standard-Clusterübertragungswarteschlange für andere CL2 -Clusterwarteschlangen unter QM3 zu verwenden. Oder Sie können XMITQ.CL2.QM3 weiterhin verwenden. Wenn Sie sich dafür entscheiden, auf eine Standard-Cluster-Übertragungswarteschlange zurückzusetzen, verwaltet der Gateway-Warteschlangenmanager den Switch automatisch für Sie.

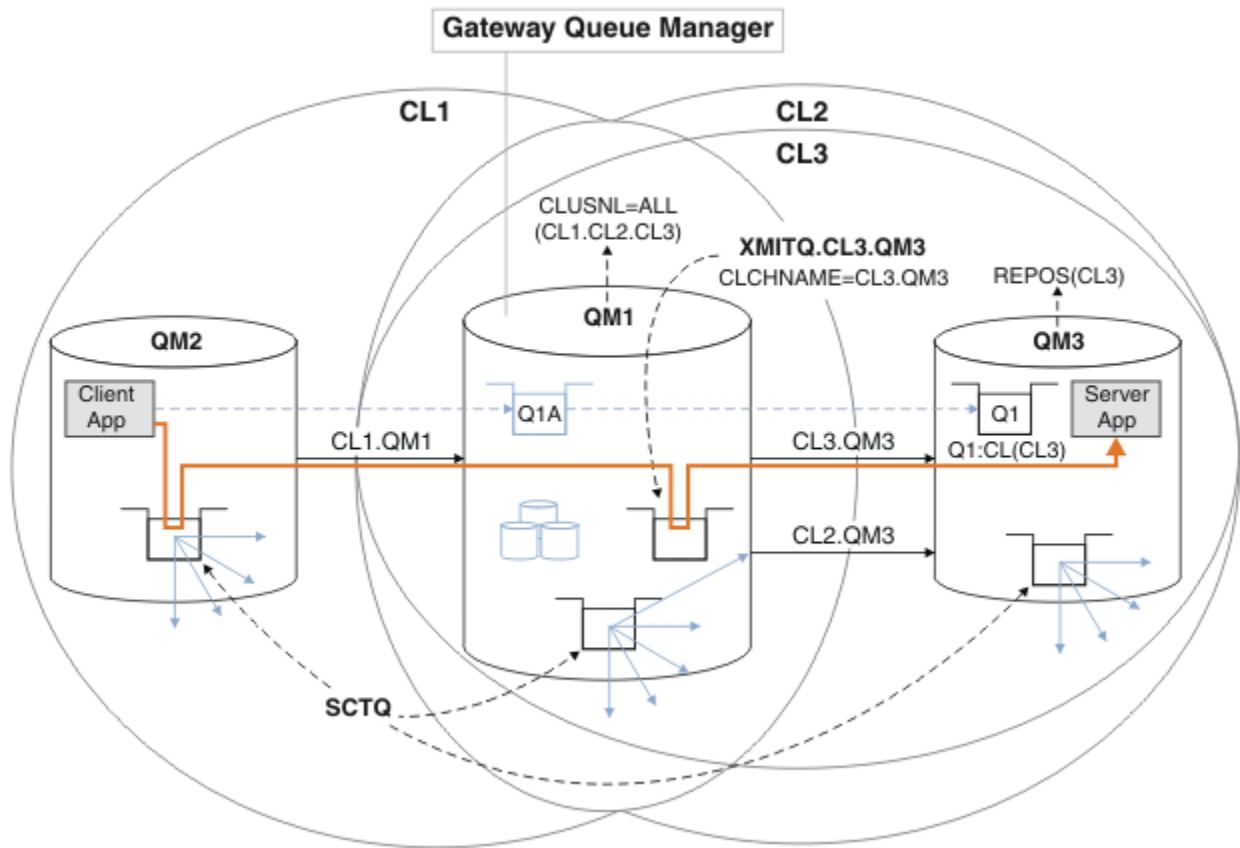


Abbildung 34. Verwenden eines zusätzlichen Clusters zum Trennen des Nachrichtenverkehrs im Gateway-Warteschlangenmanager, der zu einer Anzahl von Clusterwarteschlangen auf demselben Warteschlangenmanager wechselt

## Vorgehensweise

1. Ändern Sie die Warteschlangenmanager QM3 und QM5, um sie sowohl für CL2 als auch für CL3 zu erstellen.

Um einen Warteschlangenmanager zu einem Member mehrerer Cluster zu machen, muss er eine Clusternamensliste verwenden, um die Cluster zu identifizieren, zu der er gehört.

```
*... On QM3 and QM5
DEFINE NAMELIST(CL23) NAMES(CL2, CL3) REPLACE
ALTER QMGR REPOS(' ') REPOSNL(CL23)
```

2. Definieren Sie die Kanäle zwischen den Warteschlangenmanagern QM3 und QM5 für CL3.

```
*... On QM3
DEFINE CHANNEL(CL3.QM5) CHLTYPE(CLUSSDR) CONNAME('localhost(1415)') CLUSTER(CL3) REPLACE
DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSRCVR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE

*... On QM5
DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE
DEFINE CHANNEL(CL3.QM5) CHLTYPE(CLUSRCVR) CONNAME('localhost(1415)') CLUSTER(CL3) REPLACE
```

3. Fügen Sie den Gateway-Warteschlangenmanager zu CL3 hinzu.

Fügen Sie den Gateway-WS-Manager hinzu, indem Sie QM1 als Teilrepository zu CL3 hinzufügen. Erstellen Sie ein Teilrepository, indem Sie Clustersenderkanäle und Clusterempfängerkanäle zu QM1 hinzufügen.

Fügen Sie außerdem CL3 zur Namensliste aller Cluster hinzu, die mit dem Gateway-WS-Manager verbunden sind.

```
*... On QM1
DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE
DEFINE CHANNEL(CL3.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL3) REPLACE
ALTER NAMLIST(ALL) NAMES(CL1, CL2, CL3)
```

4. Fügen Sie eine Clusterübertragungswarteschlange für Nachrichten an CL3 unter QM3 zum Gateway-Warteschlangenmanager QM1 hinzu.

Stoppen Sie zunächst den Clustersenderkanal, der Nachrichten aus der Übertragungswarteschlange überträgt, bis Sie bereit sind, Übertragungswarteschlangen zu wechseln.

```
*... On QM1
DEFINE QLOCAL(XMITQ.CL3.QM3) USAGE(XMITQ) CLCHNAME(CL3.QM3) GET(DISABLED) REPLACE
```

5. Bereinigen Sie Nachrichten aus der vorhandenen Clusterübertragungswarteschlange XMITQ.CL2.QM3.

Diese Unterprozedur dient dazu, die Reihenfolge der Nachrichten in Q1 beizubehalten, damit sie mit der Reihenfolge übereinstimmt, in der sie im Gateway-Warteschlangenmanager angekommen sind. Bei Clustern ist die Nachrichtenreihenfolge nicht vollständig gewährleistet, ist aber wahrscheinlich. Wenn eine garantierte Nachrichtenreihenfolge erforderlich ist, müssen Anwendungen die Reihenfolge der Nachrichten definieren. Weitere Informationen finden Sie in [Die Reihenfolge, in der Nachrichten aus einer Warteschlange abgerufen werden](#).

- a) Ändern Sie die Zielwarteschlange Q1 in QM3 von CL2 in CL3.

```
*... On QM3
ALTER QLOCAL(Q1) CLUSTER(CL3)
```

- b) Überwachen Sie XMITQ.CL3.QM3, bis Nachrichten gestartet werden, die an ihn gesendet werden.

Die Zustellung von Nachrichten an XMITQ.CL3.QM3 wird gestartet, wenn der Wechsel von Q1 in CL3 an den Gateway-Warteschlangenmanager weitergegeben wird.

```
*... On QM1
DISPLAY QUEUE(XMITQ.CL3.QM3) CURDEPTH
```

- c) Überwachen Sie XMITQ.CL2.QM3, bis keine Nachrichten vorhanden sind, die auf die Zustellung an Q1 auf QM3 warten.

**Anmerkung:** XMITQ.CL2.QM3 speichert möglicherweise Nachrichten für andere Warteschlangen in QM3, die Mitglieder von CL2 sind. In diesem Fall wird die Tiefe möglicherweise nicht auf null gehen.

```
*... On QM1
DISPLAY QUEUE(XMITQ.CL2.QM3) CURDEPTH
```

- d) Abrufen aus der neuen Clusterübertragungswarteschlange aktivieren, XMITQ.CL3.QM3

```
*... On QM1
ALTER QLOCAL(XMITQ.CL3.QM3) GET(ENABLED)
```

6. Entfernen Sie die alte Clusterübertragungswarteschlange XMITQ.CL2.QM3, falls sie nicht mehr benötigt wird.

Nachrichten für Clusterwarteschlangen in CL2 auf QM3 werden auf die Verwendung der Standard-Cluster-Übertragungswarteschlange auf dem Gateway-WS-Manager QM1 zurückgesetzt. Die Standard-Cluster-Übertragungswarteschlange ist entweder SYSTEM.CLUSTER.TRANSMIT.QUEUE oder SYS-

TEM.CLUSTER.TRANSMIT.CL2.QM3. Welche davon abhängt, ob der Wert des Warteschlangenmanagerattributs **DEFCLXQ** in QM1 SCTQ oder CHANNEL ist. Der Warteschlangenmanager überträgt Nachrichten automatisch von XMITQ.CL2.QM3, wenn der Clustersenderkanal CL2.QM3 als Nächstes gestartet wird.

- a) Ändern Sie die Übertragungswarteschlange XMITQ.CL2.QM3 aus einer Cluster-Übertragungswarteschlange, um eine normale Übertragungswarteschlange zu sein.

Dadurch wird die Zuordnung der Übertragungswarteschlange zu beliebigen Clustersenderkanälen unterbrochen. In der Antwort überträgt IBM WebSphere MQ Nachrichten automatisch von XMITQ.CL2.QM3 an die Standardübertragungswarteschlange des Clusters, wenn der Clustersenderkanal als Nächstes gestartet wird. Bis dahin werden die Nachrichten für CL2 unter QM3 weiterhin auf XMITQ.CL2.QM3 gesetzt.

```
*... On QM1
ALTER QLOCAL(XMITQ.CL2.QM3) CLCHNAME('')
```

- b) Stoppen Sie den Clustersenderkanal CL2.QM3.

Wenn Sie den Clustersenderkanal stoppen und erneut starten, wird die Übertragung von Nachrichten von XMITQ.CL2.QM3 in die Standardübertragungswarteschlange des Clusters eingeleitet. In der Regel müssen Sie den Kanal manuell stoppen und starten, um die Übertragung zu starten. Die Übertragung wird automatisch gestartet, wenn der Kanal nach dem Abschließen des Unterbrechungsintervalls erneut gestartet wird.

```
*... On QM1
STOP CHANNEL(CL2.QM3)
```

Die Antwort ist, dass der Befehl akzeptiert wird:

AMQ8019: Stop WebSphere MQ channel accepted.

- c) Überprüfen Sie, ob der Kanal CL2.QM3 gestoppt wurde.

Wenn der Kanal nicht gestoppt wird, können Sie den Befehl **STOP CHANNEL** erneut mit der Option **FORCE** ausführen. Ein Beispiel für die Einstellung der Option **FORCE** wäre, wenn der Kanal nicht gestoppt wird, und Sie können den anderen Warteschlangenmanager nicht erneut starten, um den Kanal zu synchronisieren.

```
*... On QM1
DISPLAY CHSTATUS(CL2.QM3)
```

Die Antwort ist eine Zusammenfassung des Kanalstatus.

AMQ8417: Display Channel Status details.

|                          |                      |
|--------------------------|----------------------|
| CHANNEL(CL2.QM3)         | CHLTYPE(CLUSSDR)     |
| CONNAME(127.0.0.1(1413)) | CURRENT              |
| RQMNAME(QM3)             | STATUS(STOPPED)      |
| SUBSTATE(MQGET)          | XMITQ(XMITQ.CL2.QM3) |

- d) Starten Sie den Kanal CL2.QM3.

```
*... On QM1
START CHANNEL(CL2.QM3)
```

Die Antwort ist, dass der Befehl akzeptiert wird:

AMQ8018: Start WebSphere MQ channel accepted.

- e) Überprüfen Sie, ob der Kanal gestartet wurde

```
*... On QM1
DISPLAY CHSTATUS(CL2.QM3)
```

Die Antwort ist eine Zusammenfassung des Kanalstatus:

```
AMQ8417: Display Channel Status details.
CHANNEL (CL2.QM3) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1413)) CURRENT
QMNAME (QM3) STATUS (RUNNING)
SUBSTATE (MQGET) XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE | CL2.QM3)
```

- f) Überwachen Sie das Fehlerprotokoll des Gateway-Warteschlangenmanagers auf die Nachricht "AMQ7341 Die Übertragungswarteschlange für Kanal CL2.QM3 ist SYSTEM.CLUSTER.TRANSMIT.QUEUE | CL2.WSM3".
- g) Löschen Sie die Clusterübertragungswarteschlange XMITQ.CL2.QM3.

```
*... On QM1
DELETE QLOCAL (XMITQ.CL2.QM3)
```

## Nächste Schritte

Testen Sie die separate Clusterwarteschlange, indem Sie eine Nachricht von QM2 an Q1 auf QM3 unter Verwendung der Warteschlangenaliasdefinition Q1A senden.

1. Führen Sie das Beispielprogramm **amqspout** unter QM2 aus, um eine Nachricht einzureihen.

```
C:\IBM\MQ>amqspout Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A
```

```
Sample AMQSPUT0 end
```

2. Führen Sie das Beispielprogramm **amqsget** aus, um die Nachricht von Q1 unter QM3 abzurufen

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

## Zugehörige Konzepte

[„Zugriffsteuerung und mehrere Clusterübertragungswarteschlangen“](#) auf Seite 173

Wählen Sie zwischen drei Prüfmodi aus, wenn eine Anwendung Nachrichten in ferne Clusterwarteschlangen einreicht. Die Modi werden über Remotezugriff auf die Clusterwarteschlange überprüft, lokal auf SYSTEM.CLUSTER.TRANSMIT.QUEUE geprüft oder auf lokale Profile für die Clusterwarteschlange oder den Cluster-WS-Manager überprüft.

[„Clustering: Anwendungsisolation mit mehreren Clusterübertragungswarteschlangen“](#) auf Seite 306

Sie können die Nachrichtenflüsse zwischen Warteschlangenmanagern in einem Cluster isolieren. Sie können Nachrichten, die von verschiedenen Clustersenderkanälen transportiert werden, in verschiedene Clusterübertragungswarteschlangen stellen. Sie können den Ansatz in einem einzelnen Cluster oder mit überlappenden Clustern verwenden. Das Thema enthält Beispiele und einige bewährte Verfahren, die Sie bei der Auswahl eines zu verwendenden Ansatzes führen.

[„Clusterübertragungswarteschlangen und Clustersenderkanäle“](#) auf Seite 185

Nachrichten zwischen Clustering-WS-Managern werden in Clusterübertragungswarteschlangen gespeichert und von Clustersenderkanälen weitergeleitet.

## Zugehörige Tasks

[„WS-Manager zu einem Cluster hinzufügen: separate Übertragungswarteschlangen“](#) auf Seite 214

Befolgen Sie diese Anweisungen, um dem erstellten Cluster einen WS-Manager hinzuzufügen. Nachrichten zu Clusterwarteschlangen und Themen werden unter Verwendung mehrerer Clusterübertragungswarteschlangen übertragen.

[„Erstellen von zwei überlappenden Clustern mit einem Gateway-Warteschlangenmanager“](#) auf Seite 235

Befolgen Sie die Anweisungen in der Task, um überlappende Cluster mit einem Gateway-Warteschlangenmanager zu erstellen. Verwenden Sie die Cluster als Ausgangspunkt für die folgenden Beispiele, um Nachrichten in einer Anwendung von Nachrichten an andere Anwendungen in einem Cluster zu isolieren.

„Definition einer fernen Warteschlange hinzufügen, um Nachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet wurden“ auf Seite 216

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine ferne Definition einer Clusterwarteschlange und einen separaten Senderkanal und eine separate Übertragungswarteschlange.

„Ändern der Standardeinstellung in separate Clusterübertragungswarteschlangen, um den Nachrichtenverkehr zu isolieren“ auf Seite 241

Sie können die Standardweise ändern, in der ein WS-Manager Nachrichten für eine Clusterwarteschlange oder ein Topic in einer Übertragungswarteschlange speichert. Wenn Sie den Standardwert ändern, können Sie Clusternachrichten auf einem Gateway-Warteschlangenmanager isolieren.

„Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde“ auf Seite 219

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine zusätzliche Clusterübertragungswarteschlange, um den Nachrichtenverkehr auf einen einzelnen Warteschlangenmanager in einem Cluster zu trennen.

„Cluster und Cluster-Übertragungswarteschlange hinzufügen, um den Datenverkehr der Clusternachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet werden“ auf Seite 224

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet einen zusätzlichen Cluster, um die Nachrichten in einer bestimmten Clusterwarteschlange zu isolieren.

„Clustering: Planung der Konfiguration von Clusterübertragungswarteschlangen“ auf Seite 310

Sie werden durch die Auswahl von Clusterübertragungswarteschlangen geführt. Sie können eine gemeinsame Standardwarteschlange, separate Standardwarteschlangen oder manuell definierte Warteschlangen konfigurieren. Die Konfiguration mehrerer Clusterübertragungswarteschlangen gilt für andere Plattformen als z/OS.

## Mit DHCP einem Cluster einen WS-Manager hinzufügen

Fügen Sie einen Warteschlangenmanager zu einem Cluster unter Verwendung von DHCP hinzu. Die Task zeigt das Auslassen von CONNAME -Wert in einer CLUSRCVR -Definition an.

### Vorbereitende Schritte

**Anmerkung:** Damit Änderungen an einem Cluster im gesamten Cluster weitergegeben werden können, muss immer mindestens ein vollständiges Repository verfügbar sein. Stellen Sie sicher, dass Ihre Repositories verfügbar sind, bevor Sie diese Task starten.

In der Übung werden zwei spezielle Funktionen veranschaulicht:

- Die Möglichkeit, den Wert CONNAME in einer CLUSRCVR -Definition zu übergehen.
- Die Möglichkeit, +QMNAME+ in einer CLUSSDR -Definition zu verwenden.

Unter z/OS wird keine Funktion bereitgestellt.

Szenario:

- Der INVENTORY-Cluster wurde wie in „Neuen Cluster einrichten“ auf Seite 201 beschrieben eingerichtet. Es enthält zwei WS-Manager, LONDON und NEWYORK, die beide vollständige Repositories enthalten.

- Eine neue Filiale des Filialgeschäftlers wird in Paris eingerichtet und Sie möchten dem Cluster einen WS-Manager mit dem Namen PARIS hinzufügen.
- Warteschlangenmanager PARIS sendet Bestandsaktualisierungen an die Anwendung, die auf dem System in New York ausgeführt wird, indem Nachrichten in die Warteschlange INVENTQ gestellt werden.
- Die Netzkonnektivität besteht zwischen allen drei Systemen.
- Das Netzprotokoll ist TCP.
- Das WS-Manager-System von PARIS verwendet DHCP, d. es bedeutet, dass sich die IP-Adressen beim Systemneustart ändern können.
- Die Kanäle zwischen den Systemen PARIS und LONDON werden gemäß einer definierten Namenskonvention benannt. Die Konvention verwendet den WS-Manager-Namen des vollständigen Repository-WS-Managers auf LONDON.
- Administratoren des PARIS -Warteschlangenmanagers verfügen über keine Informationen zum Namen des Warteschlangenmanagers im LONDON -Repository. Der Name des WS-Managers auf dem LONDON -Repository unterliegt Änderungen.

## Informationen zu diesem Vorgang

Führen Sie die folgenden Schritte aus, um einen WS-Manager mit DHCP einem Cluster hinzuzufügen.

## Vorgehensweise

1. Entscheiden Sie, welches vollständige Repository PARIS auf das erste Element verweist.

Jeder WS-Manager in einem Cluster muss sich auf einen oder einen anderen der vollständigen Repositories beziehen. Es sammelt Informationen über den Cluster aus einem vollständigen Repository und erstellt so ein eigenes Teilrepository. Wählen Sie eines der Repositories als vollständiges Repository aus. Sobald ein neuer WS-Manager dem Cluster hinzugefügt wird, wird er sofort auch über das andere Repository informiert. Informationen zu Änderungen an einem WS-Manager werden direkt an zwei Repositories gesendet. In diesem Beispiel wird PARIS aus geographischen Gründen mit dem WS-Manager LONDON verknüpft.

**Anmerkung:** Führen Sie die verbleibenden Schritte in beliebiger Reihenfolge aus, nachdem der WS-Manager PARIS gestartet wurde.

2. Definieren Sie einen CLUSRCVR-Kanal auf WS-Manager PARIS.

Jeder WS-Manager in einem Cluster muss einen Clusterempfängerkanal definieren, auf dem er Nachrichten empfangen kann. Definieren Sie unter PARIS Folgendes:

```
DEFINE CHANNEL (INVENTORY.PARIS) CHLTYPE (CLUSRCVR)
TRPTYPE (TCP) CLUSTER (INVENTORY)
DESCR ('Cluster-receiver channel for queue manager PARIS')
```

Der Clusterempfängerkanal wirbt für die Verfügbarkeit des Warteschlangenmanagers, um Nachrichten von anderen Warteschlangenmanagern im Cluster INVENTORY zu empfangen. Sie müssen den CONNAME nicht auf dem Clusterempfängerkanal angeben. Sie können IBM WebSphere MQ anfordern, um den Verbindungsnamen aus dem System zu ermitteln, indem Sie entweder CONNAME weglassen oder CONNAME ( ' ' ) angeben. IBM WebSphere MQ generiert den Wert CONNAME unter Verwendung der aktuellen IP-Adresse des Systems (siehe [CONNAME](#)). Es ist nicht erforderlich, Definitionen auf anderen Warteschlangenmanagern für ein sendende Ende an den Clusterempfängerkanal INVENTORY . PARIS zu setzen. Andere Definitionen werden bei Bedarf automatisch erstellt.

3. Definieren Sie einen CLUSSDR-Kanal auf WS-Manager PARIS.

Jeder WS-Manager in einem Cluster muss einen Clustersenderkanal definieren, auf dem er Nachrichten an sein erstes vollständiges Repository senden kann. Geben Sie unter PARIS die folgende Definition für einen Kanal mit dem Namen INVENTORY . +QMNAME+ an den Warteschlangenmanager mit der Netzadresse LONDON . CHSTORE . COM an.

```
DEFINE CHANNEL (INVENTORY.+QMNAME+) CHLTYPE (CLUSSDR) TRPTYPE (TCP)
```

```
CONNNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at LONDON')
```

4. Optional: Wenn dieser Warteschlangenmanager wieder in einen Cluster aufgenommen wird, führen Sie einige zusätzliche Schritte aus.

a) Wenn Sie einen Warteschlangenmanager zu einem Cluster hinzufügen, der zuvor aus demselben Cluster entfernt wurde, überprüfen Sie, ob er jetzt als Cluster-Member angezeigt wird. Ist dies nicht der Fall, führen Sie die folgenden zusätzlichen Schritte aus:

i) Geben Sie den Befehl **REFRESH CLUSTER** auf dem Warteschlangenmanager aus, den Sie hinzufügen. Dieser Schritt stoppt die Clusterkanäle und gibt Ihrem lokalen Clustercache eine neue Gruppe von Folgenummern, die sichergestellt werden, dass sie im Rest des Clusters für das Up-to-Datum konfiguriert werden.

```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

**Anmerkung:** Bei großen Clustern kann der Befehl **REFRESH CLUSTER** während seiner Ausführung und danach in 27-Tage-Intervallen, wenn die Clusterobjekte ihre Statusaktualisierungen automatisch an alle interessierten Warteschlangenmanager hochladen, zu Unterbrechungen führen. Nähere Informationen hierzu erhalten Sie im Abschnitt [Die Aktualisierung in einem großen Cluster kann sich auf die Leistung und Verfügbarkeit auswirken](#).

ii) Starten Sie den CLUSSDR-Kanal erneut (z. B. mit dem Befehl [START CHANNEL](#)).

iii) Starten Sie den CLUSRCVR-Kanal erneut.

b) Wenn es sich bei dem Cluster um einen Publish/Subscribe-Cluster handelt und der erneut zu verknüpfende Warteschlangenmanager über Subskriptionen verfügt, geben Sie den folgenden Befehl aus, um sicherzustellen, dass die Proxy-Subskriptionen im Cluster ordnungsgemäß synchronisiert sind:

```
REFRESH QMGR TYPE(PROXYSUB)
```

## Ergebnisse

Der Cluster, der von dieser Task eingerichtet wird, entspricht dem von [„WS-Manager zu einem Cluster hinzufügen“](#) auf Seite 211:

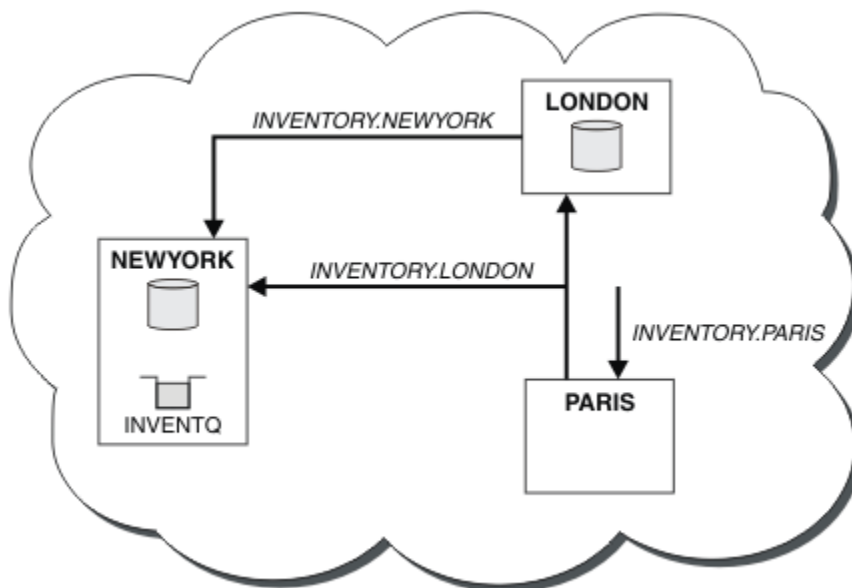


Abbildung 35. Cluster BESTANDSFÜHRUNG mit drei Warteschlangenmanagern



Wenn Sie nur zwei Definitionen, eine CLUSRCVR -Definition und eine CLUSSDR -Definition angeben, haben wir den WS-Manager PARIS dem Cluster hinzugefügt.

Auf dem PARIS -Warteschlangenmanager wird der CLUSSDR mit der Zeichenfolge +QMNAME+ gestartet. Auf dem LONDON -System löst IBM WebSphere MQ +QMNAME+ in den Namen des Warteschlangenmanagers (LONDON) auf. IBM WebSphere MQ gleicht dann die Definition für einen Kanal namens INVENTORY . LONDON mit der entsprechenden CLUSRCVR -Definition ab.

WebSphere MQ sendet den aufgelösten Kanalnamen an den PARIS -Warteschlangenmanager zurück. In PARIS wird die Kanaldefinition CLUSSDR für den Kanal mit dem Namen INVENTORY . +QMNAME+ durch eine intern generierte CLUSSDR -Definition für INVENTORY . LONDON ersetzt. Diese Definition enthält den aufgelösten Kanalnamen, ist aber ansonsten mit der von Ihnen vorgenommenen Definition von +QMNAME+ identisch. Die Cluster-Repositorys werden auch mit der Kanaldefinition mit dem neu aufgelösten Kanalnamen auf dem neuesten Stand gebracht.

#### **Anmerkung:**

1. Der Kanal, der mit dem Namen +QMNAME+ erstellt wurde, wird sofort inaktiv. Es wird nie verwendet, um Daten zu übertragen.
2. Kanalexits sehen möglicherweise die Änderung des Kanalnamens zwischen einem Aufruf und dem nächsten.

Jetzt lernt der PARIS -Warteschlangenmanager aus dem Repository in LONDON, dass die INVENTQ -Warteschlange vom WS-Manager NEWYORK gehostet wird. Wenn eine Anwendung, die vom System in Paris gehostet wird, versucht, Nachrichten in den INVENTQ, PARIS einzureihen, definiert automatisch einen Clustersenderkanal, um eine Verbindung zum Clusterempfängerkanal INVENTORY . NEWYORKherzustellen. Die Anwendung kann Antworten empfangen, wenn ihr Warteschlangenmanager-Name als Zielwarteschlangenmanager und eine Warteschlange für Antwortantworten angegeben ist.

#### **Zugehörige Verweise**

[CHANNEL DEFINE CHANNEL](#)

## **Hinzufügen eines Warteschlangenmanagers, der eine Warteschlange enthält**

Fügen Sie einen weiteren WS-Manager zum Cluster hinzu, um eine weitere INVENTQ -Warteschlange zu hosten. Anforderungen werden abwechselnd an die Warteschlangen in den einzelnen Warteschlangenmanagern gesendet. Es müssen keine Änderungen an dem vorhandenen INVENTQ -Host vorgenommen werden.

### **Vorbereitende Schritte**

**Anmerkung:** Damit Änderungen an einem Cluster im gesamten Cluster weitergegeben werden können, muss immer mindestens ein vollständiges Repository verfügbar sein. Stellen Sie sicher, dass Ihre Repositorys verfügbar sind, bevor Sie diese Task starten.

Szenario:

- Der INVENTORY -Cluster wurde wie in „WS-Manager zu einem Cluster hinzufügen“ auf Seite 211 beschrieben konfiguriert. Es enthält drei Warteschlangenmanager; LONDON und NEWYORK enthalten beide vollständige Repositorys, PARIS enthält ein Teilrepository. Die Bestandsanwendung wird auf dem System in New York ausgeführt und ist mit dem NEWYORK -Warteschlangenmanager verbunden. Die Anwendung wird durch den Eingang von Nachrichten in der INVENTQ -Warteschlange gesteuert.
- In Toronto wird ein neues Geschäft aufgebaut. Um zusätzliche Kapazitäten bereitzustellen, möchten Sie die Inventaranwendung auf dem System in Toronto sowie in New York ausführen.
- Die Netzkonnektivität besteht zwischen allen vier Systemen.
- Das Netzprotokoll ist TCP.

**Anmerkung:** Der WS-Manager TORONTO enthält nur ein Teilrepository. Wenn Sie einen WS-Manager mit vollem Repository zu einem Cluster hinzufügen möchten, lesen Sie den Abschnitt „[Vollrepositorys in einen anderen WS-Manager verschieben](#)“ auf Seite 249.



## Informationen zu diesem Vorgang

Führen Sie die folgenden Schritte aus, um einen WS-Manager hinzuzufügen, der eine Warteschlange enthält.

### Vorgehensweise

1. Entscheiden Sie, welches vollständige Repository TORONTO auf das erste Element verweist.

Jeder WS-Manager in einem Cluster muss sich auf einen oder einen anderen der vollständigen Repositories beziehen. Es sammelt Informationen über den Cluster aus einem vollständigen Repository und erstellt so ein eigenes Teilrepository. Es ist nicht besonders wichtig, welches Repository Sie auswählen. In diesem Beispiel wählen Sie NEWYORK aus. Sobald der neue WS-Manager dem Cluster beigetreten ist, kommuniziert er mit beiden Repositories.

2. Definieren Sie den Kanal CLUSRCVR .

Jeder WS-Manager in einem Cluster muss einen Clusterempfängerkanal definieren, auf dem er Nachrichten empfangen kann. Definieren Sie unter TORONTO einen CLUSRCVR -Kanal:

```
DEFINE CHANNEL(INVENTORY.TORONTO) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNNAME(TORONTO.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for TORONTO')
```

Der TORONTO -Warteschlangenmanager macht seine Verfügbarkeit für den Empfang von Nachrichten von anderen WS-Managern im INVENTORY -Cluster mit Hilfe seines Clusterempfängerkanals bekannt.

3. Definieren Sie einen CLUSSDR -Kanal auf WS-Manager TORONTO.

Jeder Warteschlangenmanager in einem Cluster muss einen Clustersenderkanal definieren, auf dem er Nachrichten an sein erstes vollständiges Repository senden kann. Wählen Sie in diesem Fall NEWYORK aus. TORONTO benötigt die folgende Definition:

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from TORONTO to repository at NEWYORK')
```

4. Optional: Wenn dieser Warteschlangenmanager wieder in einen Cluster aufgenommen wird, führen Sie einige zusätzliche Schritte aus.

- a) Wenn Sie einen Warteschlangenmanager zu einem Cluster hinzufügen, der zuvor aus demselben Cluster entfernt wurde, überprüfen Sie, ob er jetzt als Cluster-Member angezeigt wird. Ist dies nicht der Fall, führen Sie die folgenden zusätzlichen Schritte aus:

- i) Geben Sie den Befehl **REFRESH CLUSTER** auf dem Warteschlangenmanager aus, den Sie hinzufügen. Dieser Schritt stoppt die Clusterkanäle und gibt Ihrem lokalen Clustercache eine neue Gruppe von Folge Nummern, die sichergestellt werden, dass sie im Rest des Clusters für das Up-to-Datum konfiguriert werden.

```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

**Anmerkung:** Bei großen Clustern kann der Befehl **REFRESH CLUSTER** während seiner Ausführung und danach in 27-Tage-Intervallen, wenn die Clusterobjekte ihre Statusaktualisierungen automatisch an alle interessierten Warteschlangenmanager hochladen, zu Unterbrechungen führen. Nähere Informationen hierzu erhalten Sie im Abschnitt [Die Aktualisierung in einem großen Cluster kann sich auf die Leistung und Verfügbarkeit auswirken](#).

- ii) Starten Sie den CLUSSDR-Kanal erneut (z. B. mit dem Befehl [START CHANNEL](#) ).
- iii) Starten Sie den CLUSRCVR-Kanal erneut.

- b) Wenn es sich bei dem Cluster um einen Publish/Subscribe-Cluster handelt und der erneut zu verknüpfende Warteschlangenmanager über Subskriptionen verfügt, geben Sie den folgenden Befehl aus, um sicherzustellen, dass die Proxy-Subskriptionen im Cluster ordnungsgemäß synchronisiert sind:

```
REFRESH QMGR TYPE(PROXYSUB)
```

5. Überprüfen Sie die Bestandsanwendung auf Nachrichtenaffinitäten.

Bevor Sie fortfahren, stellen Sie sicher, dass die Inventaranwendung keine Abhängigkeiten von der Reihenfolge der Verarbeitung von Nachrichten hat und die Anwendung auf dem System in Toronto installiert.

6. Definieren Sie die Clusterwarteschlange INVENTQ.

Die INVENTQ -Warteschlange, die bereits vom NEWYORK -Warteschlangenmanager gehostet wird, befindet sich ebenfalls in TORONTO. Definieren Sie sie auf dem TORONTO -Warteschlangenmanager wie folgt:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

## Ergebnisse

In [Abbildung 36 auf Seite 234](#) wird der INVENTORY -Cluster angezeigt, der von dieser Task eingerichtet wurde.

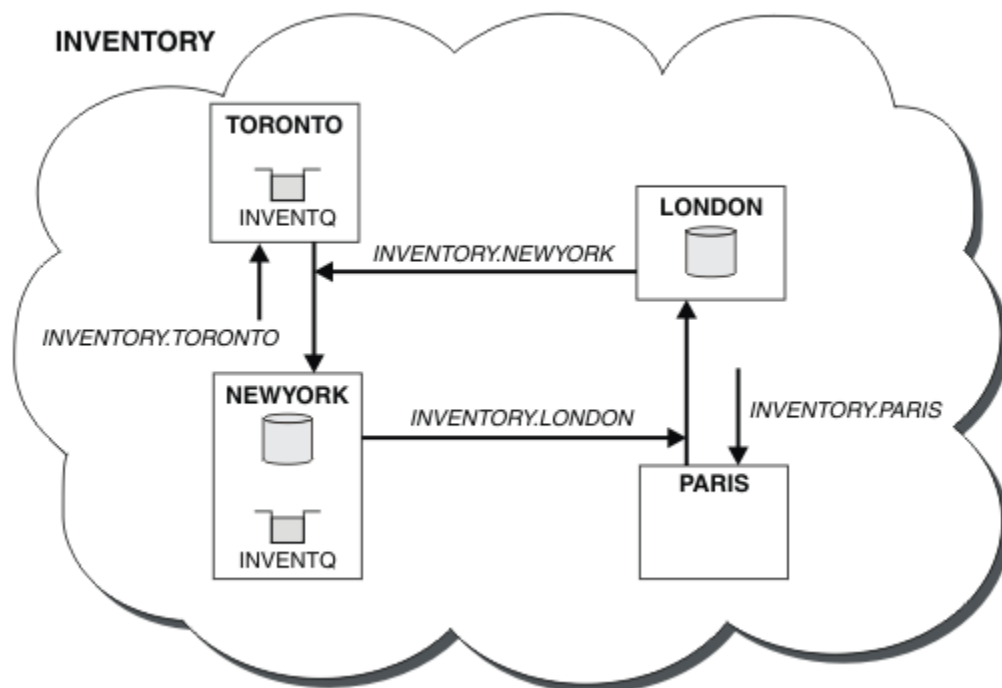


Abbildung 36. Der INVENTORY -Cluster mit vier Warteschlangenmanagern

Die INVENTQ -Warteschlange und die Inventaranwendung werden jetzt auf zwei Warteschlangenmanagern im Cluster gehostet. Dies erhöht die Verfügbarkeit, beschleunigt den Durchsatz von Nachrichten und ermöglicht die Verteilung der Auslastung zwischen den beiden Warteschlangenmanagern. Nachrichten, die entweder von TORONTO oder NEWYORK an INVENTQ gestellt werden, werden, wenn möglich, von der Instanz auf dem lokalen WS-Manager bearbeitet. Nachrichten, die von LONDON oder PARIS gestellt werden, werden abwechselnd an TORONTO oder NEWYORK weitergeleitet, so dass die Auslastung ausgeglichen ist.

Diese Änderung am Cluster wurde ausgeführt, ohne dass die Definitionen in den Warteschlangenmanagern NEWYORK, LONDON und PARIS geändert werden müssen. Die vollständigen Repositories in diesen Warteschlangenmanagern werden automatisch mit den Informationen aktualisiert, die sie benötigen, um Nachrichten an INVENTQ in TORONTO senden zu können. Die Inventaranwendung funktioniert weiterhin, wenn einer der NEWYORK oder der WS-Manager von TORONTO nicht mehr verfügbar ist und die Kapazität ausreicht. Die Bestandsanwendung muss in der Lage sein, ordnungsgemäß zu arbeiten, wenn sie an beiden Standorten gehostet wird.

Wie Sie im Ergebnis dieser Task sehen können, können Sie dieselbe Anwendung in mehr als einem Warteschlangenmanager ausführen. Sie können das Clustering gleichmäßig auf die Verteilungsauslastung verteilen.

Eine Anwendung ist möglicherweise nicht in der Lage, Datensätze an beiden Standorten zu verarbeiten. Angenommen, Sie möchten eine Abfrage zum Kunden-Account hinzufügen und die Anwendung, die in LONDON und NEWYORK ausgeführt wird, hinzufügen. Ein Accountdatensatz kann nur an einem Ort gehalten werden. Sie können die Verteilung von Anforderungen mit Hilfe eines Datenpartitionierungsverfahrens steuern. Sie können die Verteilung der Datensätze aufteilen. Sie können die Hälfte der Datensätze anordnen, z. B. für die Kontonummern 00000-49999, die in LONDON gehalten werden sollen. Die andere Hälfte im Bereich von 50000 bis 99999 wird in NEWYORK gehalten. Anschließend können Sie ein Exitprogramm für die Clusterauslastung schreiben, um das Accountfeld in allen Nachrichten zu untersuchen und die Nachrichten an den entsprechenden Warteschlangenmanager weiterzuleiten.

## **Nächste Schritte**

Nachdem Sie nun alle Definitionen abgeschlossen haben, können Sie den Kanalinitiator unter IBM WebSphere MQ for z/OS noch nicht starten. Starten Sie auf allen Plattformen ein Listenerprogramm auf dem Warteschlangenmanager TORONTO. Das Listenerprogramm wartet auf eingehende Netzanforderungen und startet den Clusterempfängerkanal, wenn er benötigt wird.

## **Erstellen von zwei überlappenden Clustern mit einem Gateway-Warteschlangenmanager**

Befolgen Sie die Anweisungen in der Task, um überlappende Cluster mit einem Gateway-Warteschlangenmanager zu erstellen. Verwenden Sie die Cluster als Ausgangspunkt für die folgenden Beispiele, um Nachrichten in einer Anwendung von Nachrichten an andere Anwendungen in einem Cluster zu isolieren.

## **Informationen zu diesem Vorgang**

In [Abbildung 37 auf Seite 236](#) wird die Beispiel-Clusterkonfiguration gezeigt, die zur Veranschaulichung des Eingrenzungs von Datenverkehr auf Clusternachrichten verwendet wird. Das Beispiel wird in [„Clustering: Anwendungsisolierung mit mehreren Clusterübertragungswarteschlangen“ auf Seite 306](#) beschrieben.

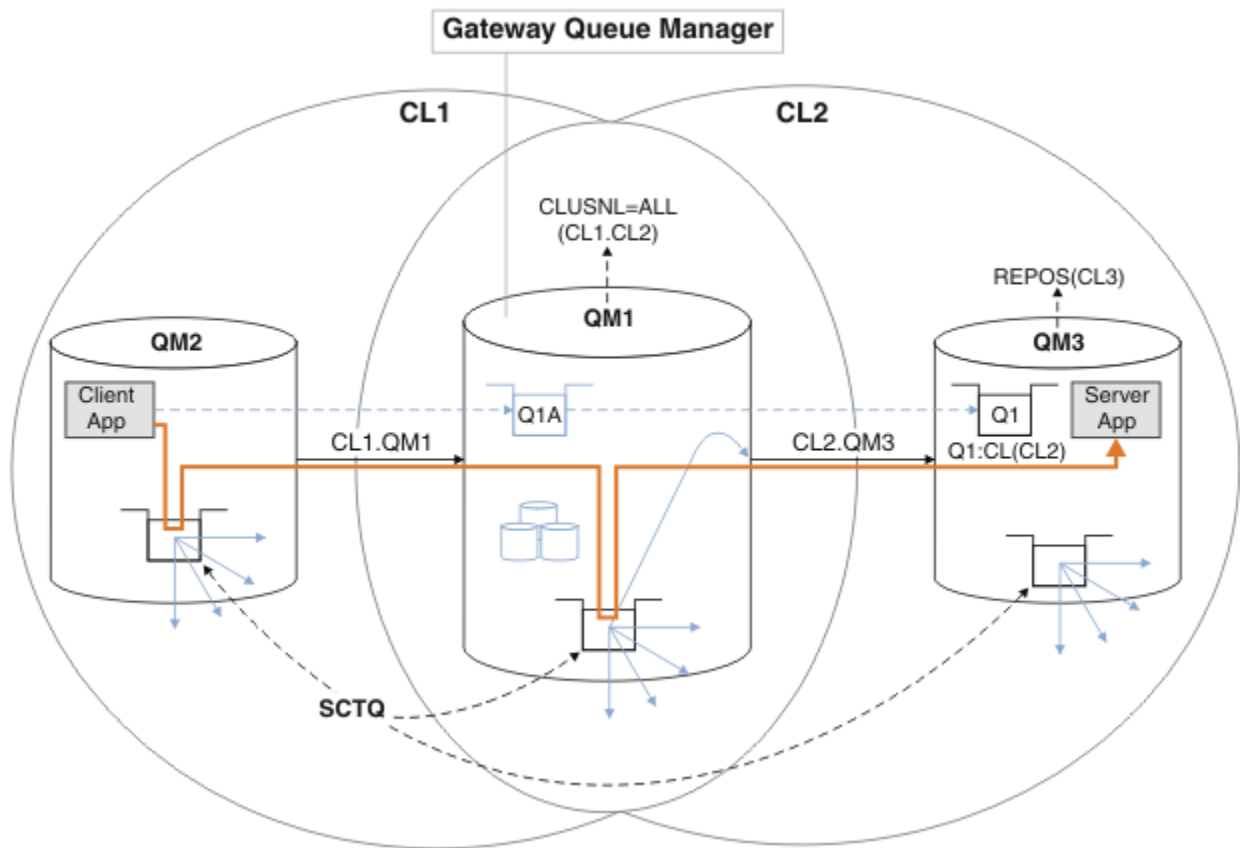


Abbildung 37. In Hub- und Spoke-Architektur mit IBM WebSphere MQ-Clustern implementierte Client/Server-Anwendung

Um die Anzahl der Schritte zum Konstruieren des Beispiels so gering wie möglich zu halten, wird die Konfiguration einfach gehalten und nicht realistisch gehalten. Das Beispiel könnte die Integration von zwei Clustern darstellen, die von zwei separaten Organisationen erstellt wurden. Ein realistischeres Szenario finden Sie unter „Clustering: Planung der Konfiguration von Clusterübertragungswarteschlangen“ auf Seite 310.

Führen Sie die Schritte aus, um die Cluster zu erstellen. Die Cluster werden in den folgenden Beispielen verwendet, um den Nachrichtenverkehr von der Clientanwendung auf die Serveranwendung zu isolieren.

Die Anweisungen fügen ein paar zusätzliche Warteschlangenmanager hinzu, so dass jeder Cluster über zwei Repositorys verfügt. Der Gateway-Warteschlangenmanager wird aus Leistungsgründen nicht als Repository verwendet.

## Vorgehensweise

1. Erstellen und starten Sie die Warteschlangenmanager QM1, QM2, QM3, QM4, QM5.

```
crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE QMn
strmqm QmgrName
```

**Anmerkung:** QM4 und QM5 sind die vollständigen Sicherungsrepositorys für die Cluster.

2. Definieren und starten Sie die Empfangsprogramme für jeden der Warteschlangenmanager.

```
*... On QMn
DEFINE LISTENER(TCP141n) TRPTYPE(TCP) IPADDR(hostname) PORT(141n) CONTROL(QMGR) REPLACE
START LISTENER(TCP141n)
```

3. Erstellen Sie eine Clusternamensliste für alle Cluster.

```
*... On QM1
DEFINE NAMELIST(ALL) NAMES(CL1, CL2) REPLACE
```

4. Erstellen Sie QM2 und QM4 vollständige Repositorys für CL1, QM3 und QM5 vollständige Repositorys für CL2.

a) Für CL1:

```
*... On QM2 and QM4
ALTER QMGR REPOS(CL1) DEFCLXQ(SCTQ)
```

b) Für CL2:

```
*... On QM3 and QM5
ALTER QMGR REPOS(CL2) DEFCLXQ(SCTQ)
```

5. Fügen Sie die Kanäle "Clustersender" und "Clusterempfänger" für jeden Warteschlangenmanager und Cluster hinzu.

Führen Sie die folgenden Befehle in QM2, QM3, QM4 und QM5 aus, wobei *c*, *n* und *m* die in [Tabelle 26](#) auf Seite 237 für jeden Warteschlangenmanager angezeigten Werte übernehmen:

*Tabelle 26. Parameterwerte für die Erstellung von Clustern 1 und 2*

| Warteschlangenmanager | Cluster<br><i>c</i> | Anderes Repository<br><i>n</i> | Dieses Repository<br><i>m</i> |
|-----------------------|---------------------|--------------------------------|-------------------------------|
| QM2                   | 1                   | 4                              | 2                             |
| QM4                   | 1                   | 2                              | 4                             |
| QM3                   | 2                   | 5                              | 3                             |
| QM5                   | 2                   | 3                              | 5                             |

```
*... On QMm
DEFINE CHANNEL(CLc.QMn) CHLTYPE(CLUSSDR) CONNAME('localhost(141n)') CLUSTER(CLc) REPLACE
DEFINE CHANNEL(CLc.QMm) CHLTYPE(CLUSRCVR) CONNAME('localhost(141m)') CLUSTER(CLc) REPLACE
```

6. Fügen Sie den Gateway-Warteschlangenmanager QM1 zu jedem Cluster hinzu.

```
*... On QM1
DEFINE CHANNEL(CL1.QM2) CHLTYPE(CLUSSDR) CONNAME('localhost(1412)') CLUSTER(CL1) REPLACE
DEFINE CHANNEL(CL1.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL1) REPLACE
DEFINE CHANNEL(CL2.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL2) REPLACE
DEFINE CHANNEL(CL2.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL2) REPLACE
```

7. Fügen Sie die lokale Warteschlange Q1 dem Warteschlangenmanager QM3 im Cluster CL2 hinzu.

```
*... On QM3
DEFINE QLOCAL(Q1) CLUSTER(CL2) REPLACE
```

8. Fügen Sie den Cluster-WS-Manager-Aliasnamen Q1A zum Gateway-Warteschlangenmanager hinzu.

```
*... On QM1
DEFINE QALIAS(Q1A) CLUSNL(ALL) TARGET(Q1) TARGTYPE(Queue) DEFBIND(NOTFIXED) REPLACE
```

**Anmerkung:** Anwendungen, die den Aliasnamen des Warteschlangenmanagers auf einem anderen Warteschlangenmanager verwenden, aber QM1, müssen DEFBIND(NOTFIXED) angeben, wenn sie die Aliaswarteschlange öffnen. **DEFBIND** gibt an, ob die Routing-Informationen im Nachrichtenheader festgelegt werden, wenn die Warteschlange von der Anwendung geöffnet wird. Wenn sie auf den Standardwert ÖFFNEN gesetzt ist, werden Nachrichten an Q1@QM1 weitergeleitet. Q1@QM1 ist nicht vorhanden, sodass Nachrichten von anderen Warteschlangenmanagern in einer Warteschlange für nicht zustellbare Nachrichten enden. Wenn Sie das Warteschlangenattribut auf DEFBIND(NOTFIXED) setzen, verhalten sich Anwendungen wie **amqspu**t, die standardmäßig die Warteschlangeneinstellung **DEFBIND** verwenden, ordnungsgemäß.

9. Fügen Sie die Clusterwarteschlangenmanageraliasdefinitionen für alle Clusterwarteschlangenmanager zum Gateway-Warteschlangenmanager QM1 hinzu.

```
*... On QM1
DEFINE QREMOTE(QM2) RNAME(' ') RQMNAME(QM2) CLUSNL(ALL) REPLACE
DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSNL(ALL) REPLACE
```

**Tipp:** Die Warteschlangenmanager-Aliasnamendefinitionen auf dem Gateway-Warteschlangenmanager übertragen Nachrichten, die auf einen Warteschlangenmanager in einem anderen Cluster verweisen; siehe Clusterwarteschlangenmanager-Aliasnamen.

## Nächste Schritte

1. Testen Sie die Definition des Warteschlangenaliasnamens, indem Sie unter QM3 unter Verwendung der Warteschlangenaliasdefinition Q1A eine Nachricht von QM2 an Q1 senden.

- a. Führen Sie das Beispielprogramm **amqsput** unter QM2 aus, um eine Nachricht einzureihen.

```
C:\IBM\MQ>amqsput Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A
```

```
Sample AMQSPUT0 end
```

- b. Führen Sie das Beispielprogramm **amqsget** aus, um die Nachricht von Q1 unter QM3 abzurufen

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

2. Testen Sie die Aliasdefinitionen des Warteschlangenmanagers, indem Sie eine Anforderungsnachricht senden und eine Antwortnachricht in einer temporären Antwortwarteschlange empfangen.

Das Diagramm zeigt den Pfad, den die Antwortnachricht zurück zu einer temporären dynamischen Warteschlange mit dem Namen RQ nimmt. Die Serveranwendung, die mit QM3 verbunden ist, öffnet die Antwortwarteschlange unter Verwendung des Warteschlangenmanagernamens QM2. Der Warteschlangenmanagername QM2 ist unter QM1 als Aliasname eines Clusterwarteschlangenmanagers definiert. QM3 leitet die Antwortnachricht an QM1 weiter. QM1 leitet die Nachricht an QM2 weiter.

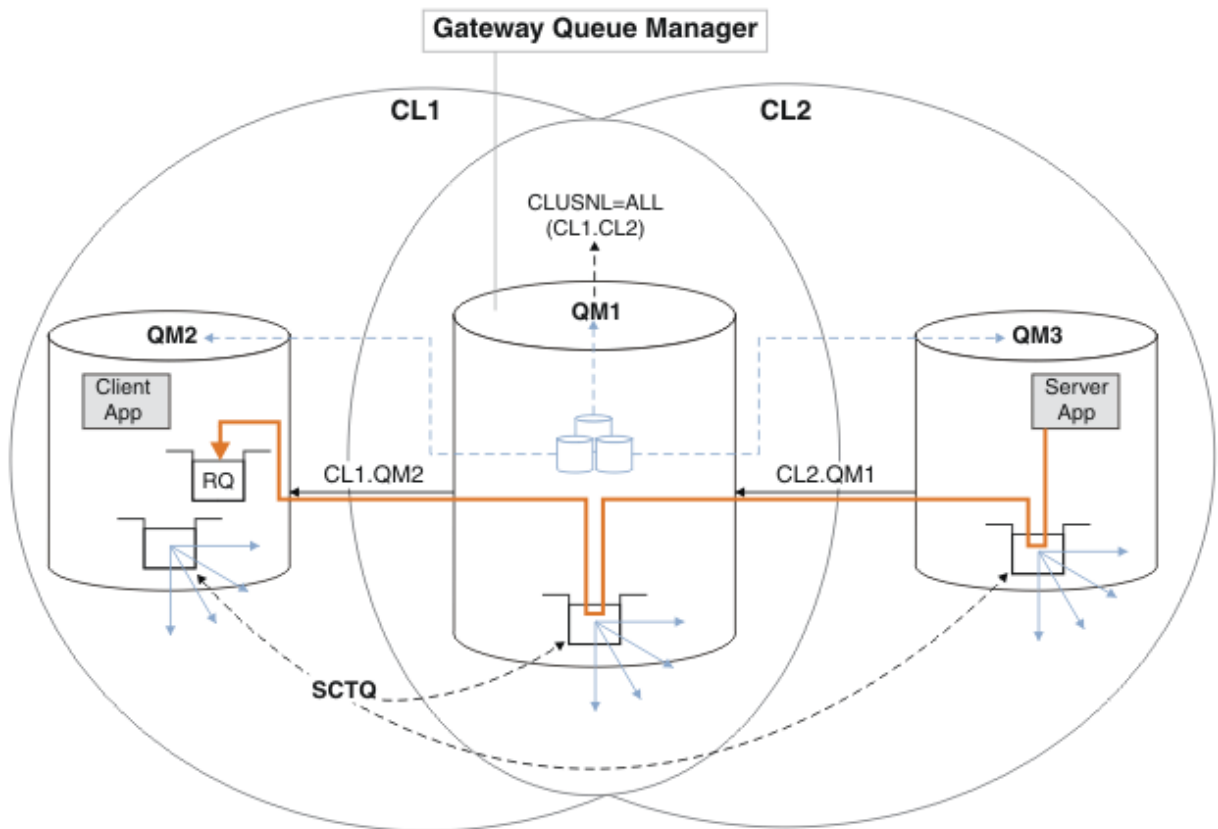


Abbildung 38. Verwenden eines Warteschlangenmanager-Aliasnamens, um die Antwortnachricht an einen anderen Cluster zurückzugeben

Die Art und Weise, wie das Routing funktioniert, ist wie folgt. Jeder Warteschlangenmanager in jedem Cluster verfügt über eine Warteschlangenmanager-Aliasnamensdefinition unter QM1. Die Aliasnamen werden in allen Clustern gruppiert. Die grauen gestrichelten Pfeile von jedem der Aliasnamen zu einem Warteschlangenmanager zeigen, dass jeder Warteschlangenmanager-Aliasname in mindestens einem der Cluster in einen echten Warteschlangenmanager aufgelöst wird. In diesem Fall wird der Aliasname QM2 sowohl in Cluster CL1 als auch in CL2 zusammengefasst und in den realen Warteschlangenmanager QM2 in CL1 aufgelöst. Die Serveranwendung erstellt die Antwortnachricht unter Verwendung des Namens der Empfangswarteschlange für Antworten RQ und des Namens des Antwortwarteschlangenmanagers QM2. Die Nachricht wird an QM1 weitergeleitet, da die Warteschlangenmanager-Aliasdefinition QM2 in QM1 im Cluster CL2 definiert ist und der Warteschlangenmanager QM2 nicht im Cluster CL2 ist. Da die Nachricht nicht an den Zielwarteschlangenmanager gesendet werden kann, wird sie an den Warteschlangenmanager gesendet, der die Aliasdefinition enthält.

QM1 stellt die Nachricht in die Clusterübertragungswarteschlange unter QM1 zur Übertragung an QM2. QM1 leitet die Nachricht an QM2 weiter, weil die Warteschlangenmanager-Aliasdefinition unter QM1 für QM2 QM2 als realen Zielwarteschlangenmanager definiert. Die Definition ist nicht kreisförmig, da die Aliasdefinitionen nur auf reale Definitionen verweisen können. Der Aliasname kann nicht auf sich selbst verweisen. Die reale Definition wird von QM1 aufgelöst, da sich sowohl QM1 als auch QM2 in demselben Cluster befinden: CL1. QM1 ermittelt die Verbindungsinformationen für QM2 aus dem Repository für CL1 und leitet die Nachricht an QM2 weiter. Damit die Nachricht von QM1 weitergeleitet wird, muss die Serveranwendung die Antwortwarteschlange mit der Option DEFBIND auf MQBND\_BIND\_NOT\_FIXED geöffnet haben. Wenn die Serveranwendung die Antwortwarteschlange mit der Option MQBND\_BIND\_ON\_OPEN geöffnet hat, wird die Nachricht nicht weitergeleitet und in eine Warteschlange für nicht zustellbare Nachrichten eingereiht.

- a. Erstellen Sie eine Clusteranforderungswarteschlange mit einem Auslöser unter QM3.

\*... On QM3

```
DEFINE QLOCAL(QR) CLUSTER(CL2) TRIGGER INITQ(SYSTEM.DEFAULT.INITIATION.QUEUE) PRO
CESS(ECHO) REPLACE
```

- b. Erstellen Sie die Clusterwarteschlangenaliasdefinition QR auf dem Gateway-Warteschlangenmanager QM1.

```
*... On QM1
DEFINE QALIAS(QRA) CLUSNL(ALL) TARGET(QR) TARGTYPE(Queue) DEFBIND(NOTFIXED) REPLACE
```

- c. Erstellen Sie eine Prozessdefinition, um das Beispielloch-Programm **amqsech** unter QM3 zu starten.

```
*... On QM3
DEFINE PROCESS(ECHO) APPLICID(AMQSECH) REPLACE
```

- d. Erstellen Sie unter QM2 eine Modellwarteschlange für das Beispielprogramm **amqsreq**, um die temporäre dynamische Antwortwarteschlange zu erstellen.

```
*... On QM2
DEFINE QMODEL(SYSTEM.SAMPLE.REPLY) REPLACE
```

- e. Testen Sie die Definition des WS-Manager-Aliasnamens, indem Sie eine Anforderung von QM2 an QR unter QM3 unter Verwendung der Warteschlangenaliasdefinition QRA senden.

- i) Führen Sie das Auslösemonitorprogramm unter QM3 aus.

```
runmqtrm -m QM3
```

Die Ausgabe ist

```
C:\IBM\MQ>runmqtrm -m QM3
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
01/02/2012 16:17:15: WebSphere MQ trigger monitor started.
```

```

01/02/2012 16:17:15: Waiting for a trigger message
```

- ii) Führen Sie das Beispielprogramm **amqsreq** in QM2 aus, um eine Anforderung zu stellen und warten Sie auf eine Antwort.

```
C:\IBM\MQ>amqsreq QRA QM2
Sample AMQSREQ0 start
server queue is QRA
replies to 4F2961C802290020
A request message from QM2 to QR on QM3

response <A request message from QM2 to QR on QM3>
no more replies
Sample AMQSREQ0 end
```

### Zugehörige Konzepte

„Zugriffssteuerung und mehrere Clusterübertragungswarteschlangen“ auf Seite 173

Wählen Sie zwischen drei Prüfmodi aus, wenn eine Anwendung Nachrichten in ferne Clusterwarteschlangen einreicht. Die Modi werden über Remotezugriff auf die Clusterwarteschlange überprüft, lokal auf SYSTEM.CLUSTER.TRANSMIT.QUEUE geprüft oder auf lokale Profile für die Clusterwarteschlange oder den Cluster-WS-Manager überprüft.

„Clustering: Anwendungsisolierung mit mehreren Clusterübertragungswarteschlangen“ auf Seite 306

Sie können die Nachrichtenflüsse zwischen Warteschlangenmanagern in einem Cluster isolieren. Sie können Nachrichten, die von verschiedenen Clustersenderkanälen transportiert werden, in verschiedene Clusterübertragungswarteschlangen stellen. Sie können den Ansatz in einem einzelnen Cluster oder mit überlappenden Clustern verwenden. Das Thema enthält Beispiele und einige bewährte Verfahren, die Sie bei der Auswahl eines zu verwendenden Ansatzes führen.



## Zugehörige Tasks

„WS-Manager zu einem Cluster hinzufügen: separate Übertragungswarteschlangen“ auf Seite 214  
Befolgen Sie diese Anweisungen, um dem erstellten Cluster einen WS-Manager hinzuzufügen. Nachrichten zu Clusterwarteschlangen und Themen werden unter Verwendung mehrerer Clusterübertragungswarteschlangen übertragen.

„Erstellen von zwei überlappenden Clustern mit einem Gateway-Warteschlangenmanager“ auf Seite 235  
Befolgen Sie die Anweisungen in der Task, um überlappende Cluster mit einem Gateway-Warteschlangenmanager zu erstellen. Verwenden Sie die Cluster als Ausgangspunkt für die folgenden Beispiele, um Nachrichten in einer Anwendung von Nachrichten an andere Anwendungen in einem Cluster zu isolieren.

„Definition einer fernen Warteschlange hinzufügen, um Nachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet wurden“ auf Seite 216

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine ferne Definition einer Clusterwarteschlange und einen separaten Senderkanal und eine separate Übertragungswarteschlange.

„Ändern der Standardeinstellung in separate Clusterübertragungswarteschlangen, um den Nachrichten-  
datenverkehr zu isolieren“ auf Seite 241

Sie können die Standardweise ändern, in der ein WS-Manager Nachrichten für eine Clusterwarteschlange oder ein Topic in einer Übertragungswarteschlange speichert. Wenn Sie den Standardwert ändern, können Sie Clusternachrichten auf einem Gateway-Warteschlangenmanager isolieren.

„Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von  
einem Gateway-Warteschlangenmanager gesendet wurde“ auf Seite 219

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine zusätzliche Clusterübertragungswarteschlange, um den Nachrichtenverkehr auf einen einzelnen Warteschlangenmanager in einem Cluster zu trennen.

„Cluster und Cluster-Übertragungswarteschlange hinzufügen, um den Datenverkehr der Clusternachrichten  
zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet werden“ auf Seite 224

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet einen zusätzlichen Cluster, um die Nachrichten in einer bestimmten Clusterwarteschlange zu isolieren.

„Clustering: Planung der Konfiguration von Clusterübertragungswarteschlangen“ auf Seite 310

Sie werden durch die Auswahl von Clusterübertragungswarteschlangen geführt. Sie können eine gemeinsame Standardwarteschlange, separate Standardwarteschlangen oder manuell definierte Warteschlangen konfigurieren. Die Konfiguration mehrerer Clusterübertragungswarteschlangen gilt für andere Plattformen als z/OS.

## Ändern der Standardeinstellung in separate Clusterübertragungswarteschlangen, um den Nachrichtendatenverkehr zu isolieren

Sie können die Standardweise ändern, in der ein WS-Manager Nachrichten für eine Clusterwarteschlange oder ein Topic in einer Übertragungswarteschlange speichert. Wenn Sie den Standardwert ändern, können Sie Clusternachrichten auf einem Gateway-Warteschlangenmanager isolieren.

## Vorbereitende Schritte

1. Der Gateway-Warteschlangenmanager muss sich unter Version 7.5 oder höher und auf einer anderen Plattform als z/OS befinden.
2. Erstellen Sie die in Abbildung 37 auf Seite 236 in „Erstellen von zwei überlappenden Clustern mit einem Gateway-Warteschlangenmanager“ auf Seite 235 gezeigten überlappenden Cluster, indem Sie die Schritte in dieser Task ausführen.

## Informationen zu diesem Vorgang

Wenn Sie die Architektur mit mehreren Clusterwarteschlangen implementieren wollen, muss Ihr Gateway-Warteschlangenmanager Version 7.5 oder höher entsprechen. Alle für die Verwendung mehrerer Clusterübertragungswarteschlangen verwendeten Warteschlangen müssen den Standardwarteschlangentyp der Clusterübertragung im Gateway-Warteschlangenmanager ändern. Ändern Sie den Wert des Warteschlangenmanagerattributs **DEFCLXQ** unter QM1 von SCTQ in CHANNEL (siehe Abbildung 39 auf Seite 242). Das Diagramm zeigt einen Nachrichtenfluss. Für Datenflüsse zu anderen Warteschlangenmanagern oder zu anderen Clustern erstellt der Warteschlangenmanager zusätzliche permanente dynamische Clusterübertragungswarteschlangen. Jeder Clustersenderkanal überträgt Nachrichten aus einer anderen Clusterübertragungswarteschlange.

Die Änderung wird nicht sofort wirksam, es sei denn, Sie verbinden den Gateway-WS-Manager zum ersten Mal mit Clustern. Die Task enthält Schritte für den typischen Fall, dass eine Änderung an einer vorhandenen Konfiguration verwaltet wird. Informationen zum Festlegen eines Warteschlangenmanagers für die Verwendung separater Clusterübertragungswarteschlangen beim ersten Joins eines Clusters finden Sie in „WS-Manager zu einem Cluster hinzufügen: separate Übertragungswarteschlangen“ auf Seite 214.

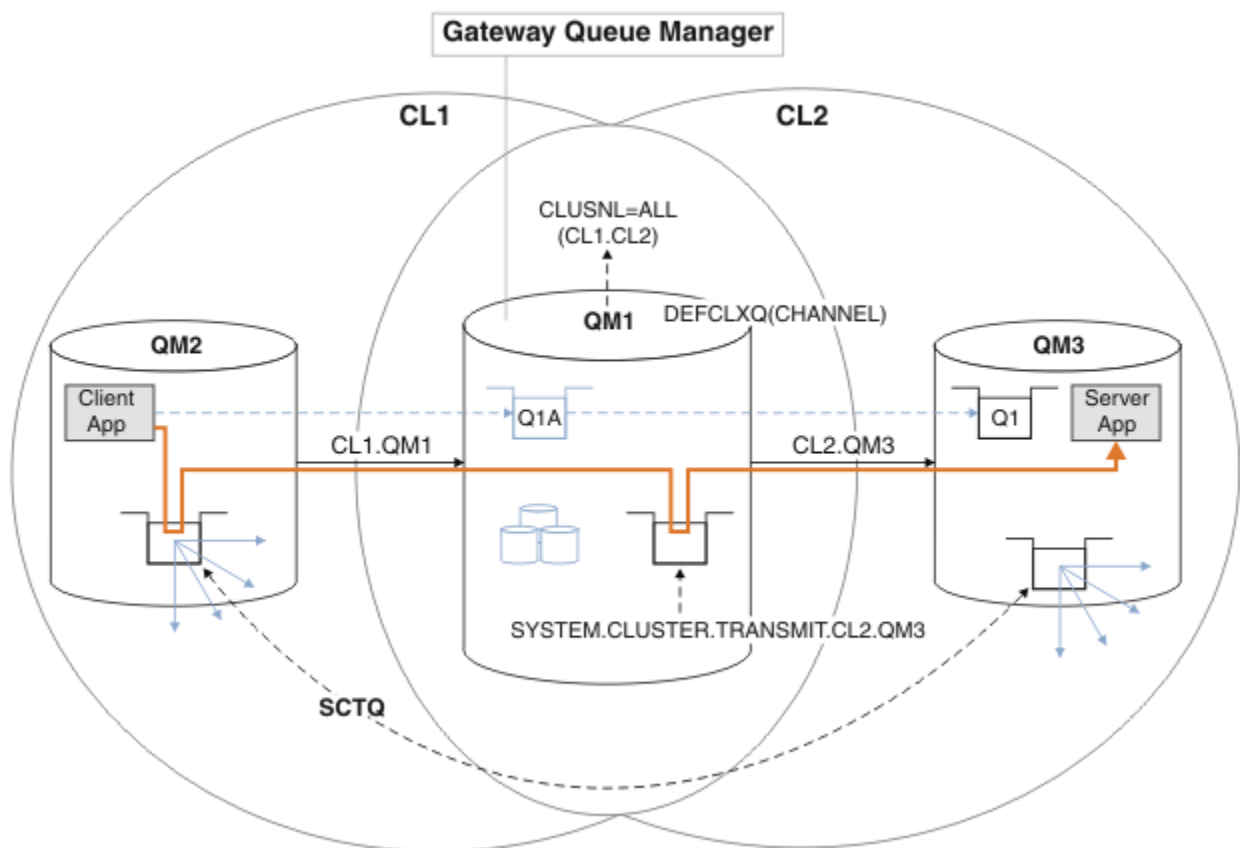


Abbildung 39. Die Client-Server-Anwendung, die in Hub-und-Spoke-Architektur implementiert ist, mit separaten Clusterübertragungswarteschlangen auf dem Gateway-Warteschlangenmanager.

## Vorgehensweise

1. Ändern Sie den Gateway-WS-Manager, um separate Clusterübertragungswarteschlangen zu verwenden.

```
*... On QM1
ALTER QMGR DEFCLXQ(CHANNEL)
```

2. Wechseln Sie zu den separaten Clusterübertragungswarteschlangen.

Jeder Clustersenderkanal, der keine Switches ausführt, um separate Clusterübertragungswarteschlangen zu verwenden, wenn er als Nächstes gestartet wird.

Um die aktiven Kanäle umzuschalten, müssen Sie entweder den Warteschlangenmanager erneut starten oder die folgenden Schritte ausführen:

- a) Listen Sie die Clustersenderkanäle auf, die mit SYSTEM.CLUSTER.TRANSMIT.QUEUE ausgeführt werden.

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')
```

Die Antwort ist eine Liste der Kanalstatusberichte:

```
AMQ8417: Display Channel Status details.
CHANNEL (CL1.QM2) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1412)) CURRENT
RQMNAME (QM2) STATUS (RUNNING)
SUBSTATE (MQGET) XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL (CL2.QM3) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1413)) CURRENT
RQMNAME (QM3) STATUS (RUNNING)
SUBSTATE (MQGET) XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL (CL2.QM5) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1415)) CURRENT
RQMNAME (QM5) STATUS (RUNNING)
SUBSTATE (MQGET) XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL (CL1.QM4) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1414)) CURRENT
RQMNAME (QM4) STATUS (RUNNING)
SUBSTATE (MQGET) XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

- b) Kanäle stoppen, die ausgeführt werden

Führen Sie für jeden Kanal in der Liste den folgenden Befehl aus:

```
*... On QM1
STOP CHANNEL(ChannelName)
```

Dabei ist *ChannelName* jeder CL1.QM2, CL1.QM4, CL1.QM3, CL1.QM5.

Die Antwort ist, dass der Befehl akzeptiert wird:

```
AMQ8019: Stop WebSphere MQ channel accepted.
```

- c) Überwachen, welche Kanäle gestoppt sind

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')
```

Die Antwort ist eine Liste der Kanäle, die noch aktiv sind, und Kanäle, die gestoppt wurden:

```
AMQ8417: Display Channel Status details.
CHANNEL (CL1.QM2) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1412)) CURRENT
RQMNAME (QM2) STATUS (STOPPED)
SUBSTATE () XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL (CL2.QM3) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1413)) CURRENT
RQMNAME (QM3) STATUS (STOPPED)
SUBSTATE () XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
```

```

CHANNEL (CL2.QM5) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1415)) CURRENT
RQMNAME (QM5) STATUS (STOPPED)
SUBSTATE () XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL (CL1.QM4) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1414)) CURRENT
RQMNAME (QM4) STATUS (STOPPED)
SUBSTATE () XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)

```

d) Starten Sie jeden gestoppten Kanal.

Führen Sie diesen Schritt für alle Kanäle aus, die ausgeführt wurden. Wenn ein Kanal nicht gestoppt wird, kann der Befehl **STOP CHANNEL** mit der Option FORCE erneut ausgeführt werden. Ein Beispiel für die Einstellung der Option FORCE wäre, wenn der Kanal nicht gestoppt wird, und Sie können den anderen Warteschlangenmanager nicht erneut starten, um den Kanal zu synchronisieren.

```
*... On QM1
START CHANNEL (CL2.QM5)
```

Die Antwort ist, dass der Befehl akzeptiert wird:

```
AMQ8018: Start WebSphere MQ channel accepted.
```

e) Überwachen Sie die Übertragungswarteschlangen, die umgeschaltet werden.

Überwachen Sie das Fehlerprotokoll des Gateway-Warteschlangenmanagers auf die Nachricht "AMQ7341 Die Übertragungswarteschlange für Kanal CL2.QM3 ist SYSTEM.CLUSTER.TRANSMIT.QUEUE | CL2.WSM3".

f) Vergewissern Sie sich, dass SYSTEM.CLUSTER.TRANSMIT.QUEUE nicht mehr verwendet wird.

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE (XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')
DISPLAY QUEUE (SYSTEM.CLUSTER.TRANSMIT.QUEUE) CURDEPTH
```

Die Antwort ist eine Liste der Kanalstatusberichte und die Tiefe von SYSTEM.CLUSTER.TRANSMIT.QUEUE:

```
AMQ8420: Channel Status not found.
AMQ8409: Display Queue details.
QUEUE (SYSTEM.CLUSTER.TRANSMIT.QUEUE) TYPE (QLOCAL)
CURDEPTH (0)
```

g) Kanäle überwachen, die gestartet werden

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE (XMITQ LK 'SYSTEM.CLUSTER.TRANSMIT.*')
```

Die Antwort ist eine Liste der Kanäle, die in diesem Fall bereits mit den neuen Standard-Cluster-Übertragungswarteschlangen ausgeführt werden:

```
AMQ8417: Display Channel Status details.
CHANNEL (CL1.QM2) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1412)) CURRENT
RQMNAME (QM2) STATUS (RUNNING)
SUBSTATE (MQGET)
XMITQ (SYSTEM.CLUSTER.TRANSMIT.CL1.QM2)
AMQ8417: Display Channel Status details.
CHANNEL (CL2.QM3) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1413)) CURRENT
RQMNAME (QM3) STATUS (RUNNING)
SUBSTATE (MQGET)
XMITQ (SYSTEM.CLUSTER.TRANSMIT.CL2.QM3)
```

```

AMQ8417: Display Channel Status details.
CHANNEL (CL2.QM5) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1415)) CURRENT
RQMNAME (QM5) STATUS (RUNNING)
SUBSTATE (MQGET)
XMITQ (SYSTEM.CLUSTER.TRANSMIT.CL2.QM5)
AMQ8417: Display Channel Status details.
CHANNEL (CL1.QM4) CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1414)) CURRENT
RQMNAME (QM4) STATUS (RUNNING)
SUBSTATE (MQGET)
XMITQ (SYSTEM.CLUSTER.TRANSMIT.CL1.QM4)

```

## Nächste Schritte

1. Testen Sie die automatisch definierte Clusterübertragungswarteschlange, indem Sie eine Nachricht von QM2 an Q1 unter QM3 senden und den Warteschlangennamen mit der Warteschlangenaliasdefinition Q1A auflösen.

a. Führen Sie das Beispielprogramm **amqspout** unter QM2 aus, um eine Nachricht einzureihen.

```

C:\IBM\MQ>amqspout Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A

Sample AMQSPUT0 end

```

b. Führen Sie das Beispielprogramm **amqsget** aus, um die Nachricht von Q1 unter QM3 abzurufen

```

C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end

```

2. Überlegen Sie, ob die Sicherheit durch die Konfiguration der Sicherheit für die Clusterwarteschlangen auf den Warteschlangenmanagern, in denen Nachrichten für die Clusterwarteschlangen stammen, neu konfiguriert werden soll.

## Zugehörige Konzepte

„Zugriffssteuerung und mehrere Clusterübertragungswarteschlangen“ auf Seite 173

Wählen Sie zwischen drei Prüfmodi aus, wenn eine Anwendung Nachrichten in ferne Clusterwarteschlangen einreicht. Die Modi werden über Remotezugriff auf die Clusterwarteschlange überprüft, lokal auf SYSTEM.CLUSTER.TRANSMIT.QUEUE geprüft oder auf lokale Profile für die Clusterwarteschlange oder den Cluster-WS-Manager überprüft.

„Clustering: Anwendungsisolation mit mehreren Clusterübertragungswarteschlangen“ auf Seite 306

Sie können die Nachrichtenflüsse zwischen Warteschlangenmanagern in einem Cluster isolieren. Sie können Nachrichten, die von verschiedenen Clustersenderkanälen transportiert werden, in verschiedene Clusterübertragungswarteschlangen stellen. Sie können den Ansatz in einem einzelnen Cluster oder mit überlappenden Clustern verwenden. Das Thema enthält Beispiele und einige bewährte Verfahren, die Sie bei der Auswahl eines zu verwendenden Ansatzes führen.

## Zugehörige Tasks

„WS-Manager zu einem Cluster hinzufügen: separate Übertragungswarteschlangen“ auf Seite 214

Befolgen Sie diese Anweisungen, um dem erstellten Cluster einen WS-Manager hinzuzufügen. Nachrichten zu Clusterwarteschlangen und Themen werden unter Verwendung mehrerer Clusterübertragungswarteschlangen übertragen.

„Erstellen von zwei überlappenden Clustern mit einem Gateway-Warteschlangenmanager“ auf Seite 235

Befolgen Sie die Anweisungen in der Task, um überlappende Cluster mit einem Gateway-Warteschlangenmanager zu erstellen. Verwenden Sie die Cluster als Ausgangspunkt für die folgenden Beispiele, um Nachrichten in einer Anwendung von Nachrichten an andere Anwendungen in einem Cluster zu isolieren.

„Definition einer fernen Warteschlange hinzufügen, um Nachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet wurden“ auf Seite 216

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine ferne Definition einer Clusterwarteschlange und einen separaten Senderkanal und eine separate Übertragungswarteschlange.

„Ändern der Standardeinstellung in separate Clusterübertragungswarteschlangen, um den Nachrichtenverkehr zu isolieren“ auf Seite 241

Sie können die Standardweise ändern, in der ein WS-Manager Nachrichten für eine Clusterwarteschlange oder ein Topic in einer Übertragungswarteschlange speichert. Wenn Sie den Standardwert ändern, können Sie Clusternachrichten auf einem Gateway-Warteschlangenmanager isolieren.

„Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde“ auf Seite 219

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine zusätzliche Clusterübertragungswarteschlange, um den Nachrichtenverkehr auf einen einzelnen Warteschlangenmanager in einem Cluster zu trennen.

„Cluster und Cluster-Übertragungswarteschlange hinzufügen, um den Datenverkehr der Clusternachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet werden“ auf Seite 224

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet einen zusätzlichen Cluster, um die Nachrichten in einer bestimmten Clusterwarteschlange zu isolieren.

„Clustering: Planung der Konfiguration von Clusterübertragungswarteschlangen“ auf Seite 310

Sie werden durch die Auswahl von Clusterübertragungswarteschlangen geführt. Sie können eine gemeinsame Standardwarteschlange, separate Standardwarteschlangen oder manuell definierte Warteschlangen konfigurieren. Die Konfiguration mehrerer Clusterübertragungswarteschlangen gilt für andere Plattformen als z/OS.

## Clusterwarteschlange aus einem WS-Manager entfernen

Inaktivieren Sie die INVENTQ -Warteschlange in Toronto. Senden Sie alle Bestandsnachrichten an New York und löschen Sie die Warteschlange INVENTQ in Toronto, wenn sie leer ist.

### Vorbereitende Schritte

**Anmerkung:** Damit Änderungen an einem Cluster im gesamten Cluster weitergegeben werden können, muss immer mindestens ein vollständiges Repository verfügbar sein. Stellen Sie sicher, dass Ihre Repositories verfügbar sind, bevor Sie diese Task starten.

Szenario:

- Der INVENTORY -Cluster wurde wie in „Hinzufügen eines Warteschlangenmanagers, der eine Warteschlange enthält“ auf Seite 232 beschrieben konfiguriert. Sie enthält vier WS-Manager. LONDON und NEWYORK enthalten beide vollständige Repositories. PARIS und TORONTO enthalten Teilrepositories. Die Inventaranwendung wird auf den Systemen in New York und Toronto ausgeführt und wird durch das Eintreffen von Nachrichten in der INVENTQ -Warteschlange gesteuert.
- Aufgrund der geringeren Auslastung möchten Sie die Inventaranwendung in Toronto nicht mehr ausführen. Sie möchten die INVENTQ -Warteschlange, die vom Warteschlangenmanager TORONTO gehostet wird, inaktivieren und TORONTO -Feednachrichten in die INVENTQ -Warteschlange in NEWYORK stellen.

- Die Netzkonnektivität besteht zwischen allen vier Systemen.
- Das Netzprotokoll ist TCP.

## Informationen zu diesem Vorgang

Führen Sie die folgenden Schritte aus, um eine Clusterwarteschlange zu entfernen.

### Vorgehensweise

1. Geben Sie an, dass die Warteschlange nicht mehr verfügbar ist.

Wenn Sie eine Warteschlange aus einem Cluster entfernen möchten, entfernen Sie den Clusternamen aus der lokalen Warteschlangendefinition. Ändern Sie die INVENTQ unter TORONTO so, dass sie für den Rest des Clusters nicht zugänglich ist:

```
ALTER QLOCAL(INVENTQ) CLUSTER('')
```

2. Überprüfen Sie, ob die Warteschlange nicht mehr verfügbar ist.

Überprüfen Sie in einem vollständigen WS-Manager-Repository entweder LONDON oder NEWYORK, ob die Warteschlange nicht mehr vom Warteschlangenmanager TORONTO per Hosting bereitgestellt wird, indem Sie den folgenden Befehl ausgeben:

```
DIS QCLUSTER (INVENTQ)
```

TORONTO ist in den Ergebnissen nicht aufgeführt, wenn der Befehl ALTER erfolgreich ausgeführt wurde.

3. Inaktivieren Sie die Warteschlange.

Inaktivieren Sie die INVENTQ -Warteschlange in TORONTO , so dass keine weiteren Nachrichten in diese Warteschlange geschrieben werden können:

```
ALTER QLOCAL(INVENTQ) PUT(DISABLED)
```

Jetzt werden Nachrichten, die mit MQ00\_BIND\_ON\_OPEN in diese Warteschlange übertragen werden, in die Warteschlange für nicht zustellbare Nachrichten eingereiht. Sie müssen alle Anwendungen stoppen, um Nachrichten explizit in die Warteschlange dieses Warteschlangenmanagers zu stellen.

4. Überwachen Sie die Warteschlange, bis sie leer ist.

Überwachen Sie die Warteschlange mit dem Befehl DISPLAY QUEUE , indem Sie die Attribute IP-PROCS, OP-PROCS und CURDEPTH angeben, oder verwenden Sie den Befehl **WRKMQMSTS** unter IBM i. Wenn die Anzahl der Ein- und Ausgabeprozesse und die aktuelle Tiefe der Warteschlangensare alle null sind, ist die Warteschlange leer.

5. Überwachen Sie den Kanal, um sicherzustellen, dass keine unbestätigten Nachrichten vorhanden sind.

Um sicherzustellen, dass keine unbestätigten Nachrichten auf dem Kanal INVENTORY.TORONTO vorhanden sind, überwachen Sie den Clustersenderkanal INVENTORY.TORONTO auf jedem der anderen Warteschlangenmanager. Setzen Sie den Befehl DISPLAY CHSTATUS ab, indem Sie den Parameter INDOUBT von jedem WS-Manager angeben:

```
DISPLAY CHSTATUS(INVENTORY.TORONTO) INDOUBT
```

Wenn unbestätigte Nachrichten vorhanden sind, müssen Sie diese beheben, bevor Sie fortfahren. Sie können z. B. versuchen, den Befehl RESOLVE channel auszugeben oder den Kanal zu stoppen und erneut zu starten.

6. Löschen Sie die lokale Warteschlange.

Wenn Sie sicher sind, dass keine Nachrichten mehr an die Inventaranwendung unter TORONTO zugestellt werden sollen, können Sie die Warteschlange löschen:

```
DELETE QLOCAL(INVENTQ)
```

7. Sie können die Inventaranwendung jetzt aus dem System in Toronto entfernen.

Durch das Entfernen der Anwendung wird die Duplizierung vermieden und Speicherplatz auf dem System eingespart.

## Ergebnisse

Der Cluster, der mit dieser Task konfiguriert wird, ist wie der von der vorherigen Task eingerichtete Cluster. Der Unterschied ist, dass die INVENTQ -Warteschlange nicht mehr auf WS-Manager TORONTO verfügbar ist.

Wenn Sie die Warteschlange in Schritt 1 außer Betrieb genommen haben, hat der Warteschlangenmanager TORONTO eine Nachricht an die beiden vollständigen Warteschlangenmanager-Repositorys gesendet. Sie hat sie über die Änderung des Status benachrichtigt. Die vollständigen WS-Manager-Repository-WS-Manager übergeben diese Informationen an andere Warteschlangenmanager im Cluster, die Aktualisierungen an den Informationen zu INVENTQ angefordert haben.

Wenn ein Warteschlangenmanager eine Nachricht in die INVENTQ -Warteschlange einreicht, zeigt das aktualisierte Teilrepository an, dass die Warteschlange von INVENTQ nur auf dem NEWYORK -Warteschlangenmanager verfügbar ist. Die Nachricht wird an den WS-Manager von NEWYORK gesendet.

## Nächste Schritte

In dieser Task war nur eine Warteschlange zum Entfernen vorhanden, und es wurde nur ein Cluster entfernt, aus dem sie entfernt werden konnte.

Angenommen, es gibt viele Warteschlangen, die sich auf eine Namensliste beziehen, die viele Clusternamen enthält. Der Warteschlangenmanager TORONTO kann beispielsweise nicht nur die INVENTQ, sondern auch die PAYROLLQ, SALESQ und PURCHASESQ hosten. TORONTO macht diese Warteschlangen in allen entsprechenden Clustern verfügbar ( INVENTORY, PAYROLL, SALES und PURCHASES ). Definieren Sie eine Namensliste der Clusternamen auf dem WS-Manager von TORONTO :

```
DEFINE NAMELIST(TOROLIST)
 DESCR('List of clusters TORONTO is in')
 NAMES(INVENTORY, PAYROLL, SALES, PURCHASES)
```

Fügen Sie die Namensliste zu jeder Warteschlangendefinition hinzu:

```
DEFINE QLOCAL(INVENTQ) CLUSNL(TOROLIST)
DEFINE QLOCAL(PAYROLLQ) CLUSNL(TOROLIST)
DEFINE QLOCAL(SALESQ) CLUSNL(TOROLIST)
DEFINE QLOCAL(PURCHASESQ) CLUSNL(TOROLIST)
```

Nehmen Sie nun an, dass Sie alle diese Warteschlangen aus dem SALES -Cluster entfernen möchten, da die Operation SALES von der Operation PURCHASES übernommen werden soll. Sie müssen lediglich die TOROLIST -Namensliste ändern, um den Namen des SALES -Clusters aus dem Cluster zu entfernen.

Wenn Sie eine einzelne Warteschlange aus einem der Cluster in der Namensliste entfernen möchten, erstellen Sie eine Namensliste, die die verbleibende Liste der Clusternamen enthält. Ändern Sie anschließend die Warteschlangendefinition so, dass sie die neue Namensliste verwendet. Gehen Sie wie folgt vor, um den PAYROLLQ aus dem INVENTORY -Cluster

1. Erstellen Sie eine Namensliste:

```
DEFINE NAMELIST(TOROSHORTLIST)
 DESCR('List of clusters TORONTO is in other than INVENTORY')
 NAMES(PAYROLL, SALES, PURCHASES)
```



## 2. Ändern Sie die PAYROLLQ -Warteschlangendefinition:

```
ALTER QLOCAL(PAYROLLQ) CLUSNL(TOROSHORTLIST)
```

## Vollrepositorys in einen anderen WS-Manager verschieben

Versetzen Sie ein vollständiges Repository von einem WS-Manager in einen anderen, und erstellen Sie das neue Repository aus den Informationen, die im zweiten Repository enthalten sind.

### Vorbereitende Schritte

**Anmerkung:** Damit Änderungen an einem Cluster im gesamten Cluster weitergegeben werden können, muss immer mindestens ein vollständiges Repository verfügbar sein. Stellen Sie sicher, dass Ihre Repositories verfügbar sind, bevor Sie diese Task starten.

Szenario:

- Der INVENTORY -Cluster wurde wie in „[WS-Manager zu einem Cluster hinzufügen](#)“ auf Seite 211 beschrieben konfiguriert.
- Aus geschäftlichen Gründen möchten Sie jetzt das vollständige Repository aus dem WS-Manager LONDON entfernen und durch ein vollständiges Repository im Warteschlangenmanager PARIS ersetzen. Der WS-Manager von NEWYORK muss weiterhin ein vollständiges Repository enthalten.

### Informationen zu diesem Vorgang

Führen Sie die folgenden Schritte aus, um ein vollständiges Repository in einen anderen WS-Manager zu verschieben.

### Vorgehensweise

1. Ändern Sie PARIS , um es zu einem vollständigen WS-Manager-Repository zu machen.

Geben Sie unter PARIS den folgenden Befehl aus:

```
ALTER QMGR REPOS(INVENTORY)
```

2. Fügen Sie einen CLUSSDR -Kanal unter PARIS hinzu.

PARIS hat derzeit einen Clustersenderkanal, der auf LONDON zeigt. LONDON ist nicht mehr zum Speichern eines vollständigen Repositories für den Cluster mehr erforderlich. PARIS muss über einen neuen Clustersenderkanal verfügen, der auf NEWYORK verweist, wo das andere vollständige Repository jetzt angehalten ist.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at NEWYORK')
```

3. Definieren Sie einen CLUSSDR -Kanal in NEWYORK , der auf PARIS verweist.

Derzeit verfügt NEWYORK über einen Clustersenderkanal, der auf LONDON zeigt. Nun, da das andere vollständige Repository in PARIS verschoben wurde, müssen Sie einen neuen Clustersenderkanal in NEWYORK hinzufügen, der auf PARIS verweist.

```
DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from NEWYORK to repository at PARIS')
```

Wenn Sie den Clustersenderkanal zu PARIS hinzufügen, erfährt PARIS von NEWYORK über den Cluster. Es baut sein eigenes vollständiges Repository mit Hilfe der Informationen aus NEWYORK auf.

4. Überprüfen Sie, ob der Warteschlangenmanager PARIS jetzt über ein vollständiges Repository verfügt.

Überprüfen Sie, ob der WS-Manager PARIS ein eigenes vollständiges Repository aus dem vollständigen Repository auf dem Warteschlangenmanager NEWYORK erstellt hat. Setzen Sie die folgenden Befehle ab:

```
DIS QCLUSTER(*) CLUSTER (INVENTORY)
DIS CLUSQMR(*) CLUSTER (INVENTORY)
```

Überprüfen Sie, ob diese Befehle Details zu denselben Ressourcen in diesem Cluster wie in NEWYORK enthalten.

**Anmerkung:** Wenn der WS-Manager NEWYORK nicht verfügbar ist, kann dieses Informationsgebäude nicht vollständig ausgeführt werden. Bewegen Sie sich nicht zum nächsten Schritt, bis die Task abgeschlossen ist.

#### 5. WS-Managerdefinition in LONDON ändern

Ändern Sie schließlich den Warteschlangenmanager in LONDON , so dass er kein vollständiges Repository mehr für den Cluster enthält. Geben Sie unter LONDON den folgenden Befehl aus:

```
ALTER QMGR REPOS(' ')
```

Der WS-Manager erhält keine Clusterinformationen mehr. Nach 30 Tagen laufen die Informationen ab, die in ihrem vollständigen Repository gespeichert sind. Der WS-Manager LONDON erstellt jetzt ein eigenes Teilrepository.

#### 6. Entfernen oder ändern Sie alle ausstehenden Definitionen.

Wenn Sie sicher sind, dass die neue Anordnung Ihres Clusters wie erwartet funktioniert, entfernen oder ändern Sie manuell definierte CLUSSDR-Definitionen, die nicht mehr korrekt sind.

- Auf dem PARIS -Warteschlangenmanager müssen Sie den Clustersenderkanal auf LONDON stoppen und löschen und anschließend den Befehl start channel absetzen, damit der Cluster die automatischen Kanäle erneut verwenden kann:

```
STOP CHANNEL(INVENTORY.LONDON)
DELETE CHANNEL(INVENTORY.LONDON)
START CHANNEL(INVENTORY.LONDON)
```

- Auf dem NEWYORK -Warteschlangenmanager müssen Sie den Clustersenderkanal auf LONDON stoppen und löschen und anschließend den Befehl start channel absetzen, damit der Cluster die automatischen Kanäle erneut verwenden kann:

```
STOP CHANNEL(INVENTORY.LONDON)
DELETE CHANNEL(INVENTORY.LONDON)
START CHANNEL(INVENTORY.LONDON)
```

- Ersetzen Sie alle anderen Clustersenderkanäle im Cluster, die auf LONDON verweisen, durch Kanäle, die auf NEWYORK oder PARIS verweisen. In diesem kleinen Beispiel gibt es keine anderen. Um zu überprüfen, ob weitere vergessene Warteschlangenmanager vorhanden sind, geben Sie den Befehl DISPLAY CHANNEL von jedem Warteschlangenmanager aus und geben Sie dabei TYPE(CLUSSDR) an. Beispiel:

```
DISPLAY CHANNEL(*) TYPE(CLUSSDR)
```

Es ist wichtig, dass Sie diese Task so schnell wie möglich ausführen, nachdem Sie das vollständige Repository von LONDON in PARIS verschoben haben. In der Zeit, bevor Sie diese Task ausführen, können Warteschlangenmanager, die über manuell definierte CLUSSDR-Kanäle mit dem Namen INVENTORY.LONDON verfügen, Anforderungen für Informationen senden, die diesen Kanal verwenden.

Nachdem LONDON nicht mehr ein vollständiges Repository ist, wird es, wenn es solche Anforderungen empfängt, Fehlernachrichten in das Fehlerprotokoll des Warteschlangenmanagers schreiben. Die folgenden Beispiele zeigen, welche Fehlernachrichten in LONDON angezeigt werden:

- AMQ9428: Unexpected publication of a cluster queue object received
- AMQ9432: Query received by a non-repository queue manager

Der Warteschlangenmanager LONDON antwortet nicht auf die Anforderungen für Informationen, da er kein vollständiges Repository mehr ist. Die Warteschlangenmanager, die Informationen von LONDON anfordern, müssen sich auf NEWYORK für Clusterinformationen verlassen, bis ihre manuell definierten CLUSSDR-Definitionen korrigiert werden, um auf PARIS zu verweisen. Diese Situation darf auf lange Sicht nicht als gültige Konfiguration toleriert werden.

## Ergebnisse

In [Abbildung 40 auf Seite 251](#) wird der Cluster angezeigt, der von dieser Task eingerichtet wurde.

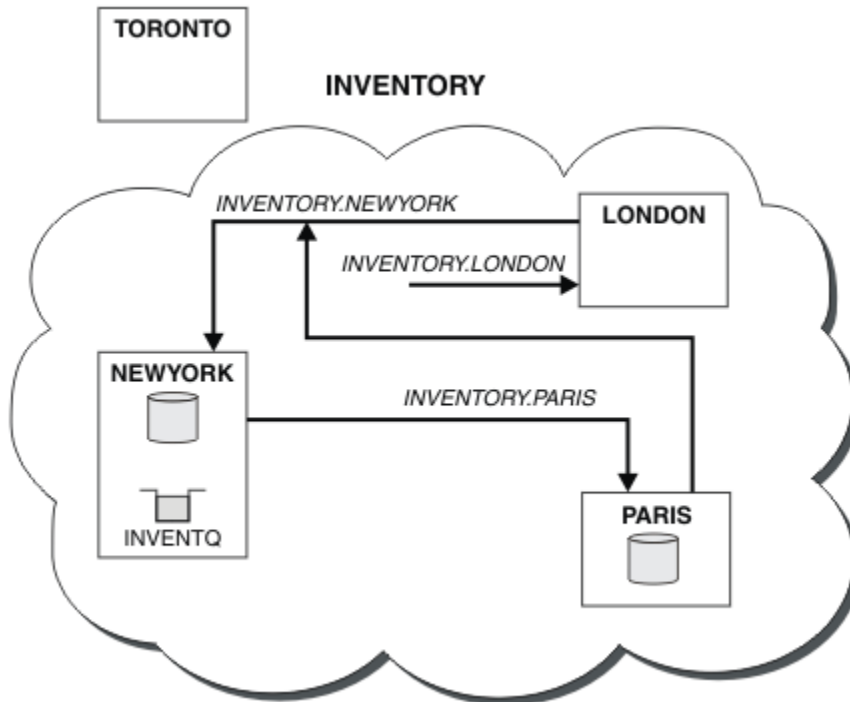


Abbildung 40. Der INVENTORY -Cluster mit dem vollständigen Repository wurde in PARIS verschoben.

## Konvertieren eines vorhandenen Netzes in einen Cluster

Konvertieren Sie ein existierendes verteiltes Warteschlangennetz in einen Cluster und fügen Sie einen zusätzlichen Warteschlangenmanager hinzu, um die Kapazität zu erhöhen.

### Vorbereitende Schritte

In [„Neuen Cluster einrichten“ auf Seite 201](#) über [„Vollrepositorys in einen anderen WS-Manager verschieben“ auf Seite 249](#) haben Sie einen neuen Cluster erstellt und erweitert. Die nächsten beiden Tasks untersuchen einen anderen Ansatz: die Konvertierung eines vorhandenen Netzes von Warteschlangenmanagern in einen Cluster.

**Anmerkung:** Damit Änderungen an einem Cluster im gesamten Cluster weitergegeben werden können, muss immer mindestens ein vollständiges Repository verfügbar sein. Stellen Sie sicher, dass Ihre Repositories verfügbar sind, bevor Sie diese Task starten.

Szenario:

- Es ist bereits ein IBM WebSphere MQ -Netz vorhanden, das die landesweiten Filialen eines Kettengeschäfts verbindet. Es verfügt über eine Hub-und-Speichenstruktur: Alle Warteschlangenmanager sind mit einem zentralen Warteschlangenmanager verbunden. Der zentrale WS-Manager befindet sich auf dem System, auf dem die Bestandsanwendung ausgeführt wird. Die Anwendung wird durch das Eintreffen von Nachrichten in die Warteschlange von INVENTQ gesteuert, für die jeder Warteschlangenmanager über eine Definition einer fernen Warteschlange verfügt.

Dieses Netz ist in [Abbildung 41](#) auf Seite 252 dargestellt.

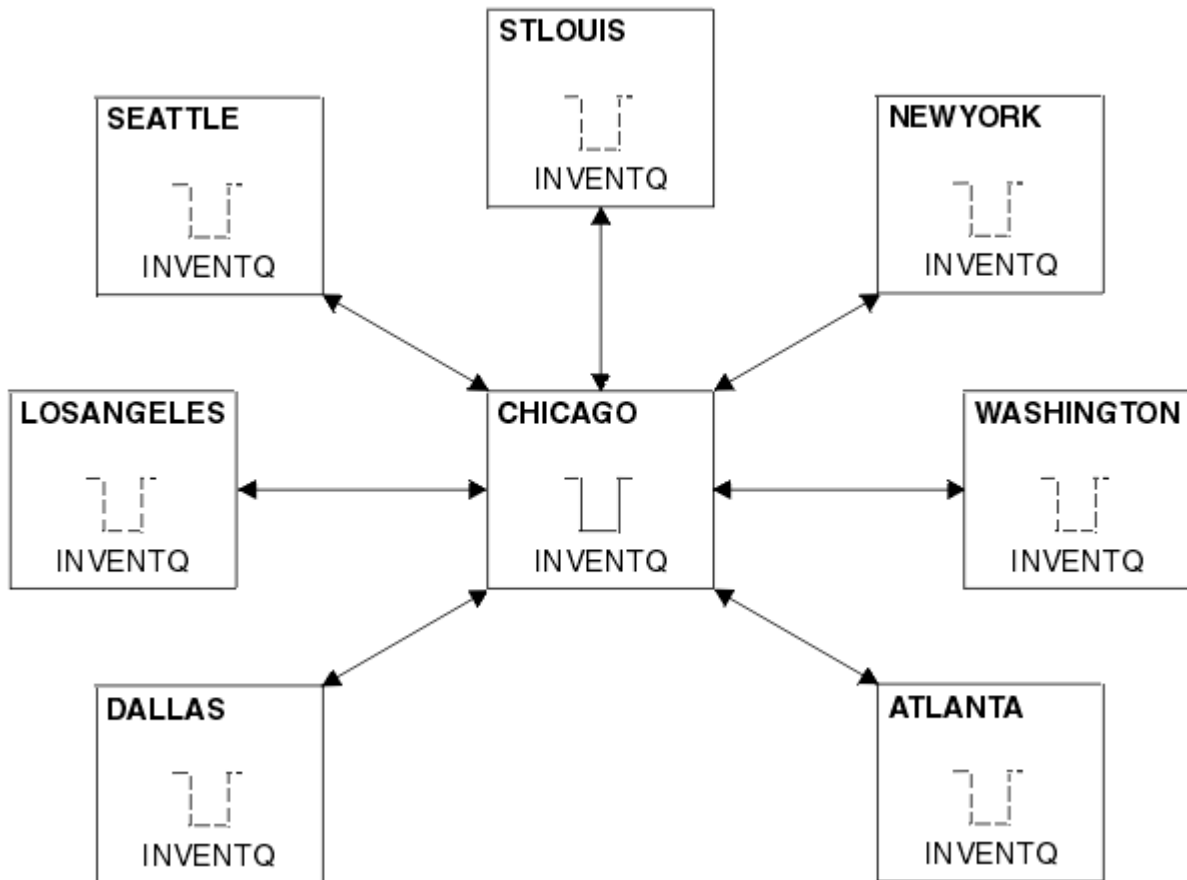


Abbildung 41. Ein Hub-und-Spoke-Netz

- Um die Verwaltung zu vereinfachen, werden Sie dieses Netz in einen Cluster konvertieren und einen anderen Warteschlangenmanager am zentralen Standort erstellen, um die Workload gemeinsam zu nutzen.

Der Clustername lautet CHNSTORE.

**Anmerkung:** Der Clustername CHNSTORE wurde ausgewählt, damit Clusterempfängerkanalnamen mit Namen im Format *cluster-name.queue-manager* erstellt werden können, die die maximale Länge von 20 Zeichen nicht überschreiten, z. B. CHNSTORE.WASHINGTON.

- Beide zentralen Warteschlangenmanager sollen vollständige Repositorys als Host für die Bestandsanwendung verwenden.
- Die Inventaranwendung soll durch das Eintreffen von Nachrichten in die Warteschlange von INVENTQ gesteuert werden, die von einem der zentralen WS-Manager gehostet wird.
- Die Bestandsanwendung ist die einzige Anwendung, die parallel ausgeführt wird und von mehr als einem Warteschlangenmanager zugänglich ist. Alle anderen Anwendungen werden weiterhin wie zuvor ausgeführt.
- Alle Filialen verfügen über Netzkonnektivität zu den beiden zentralen WS-Managern.
- Das Netzprotokoll ist TCP.

### Informationen zu diesem Vorgang

Führen Sie die folgenden Schritte aus, um ein vorhandenes Netz in einen Cluster zu konvertieren.

#### Vorgehensweise

1. Überprüfen Sie die Bestandsanwendung auf Nachrichtenaffinitäten.

Bevor Sie fortfahren, stellen Sie sicher, dass die Anwendung Nachrichtenaffinitäten verarbeiten kann. Nachrichtenaffinitäten sind die Beziehungen zwischen Dialognachrichten, die zwischen zwei Anwendungen ausgetauscht werden, wobei die Nachrichten von einem bestimmten WS-Manager oder in einer bestimmten Reihenfolge verarbeitet werden müssen. Weitere Informationen zu Nachrichtenaffinitäten finden Sie in „Nachrichtenaffinitäten bearbeiten“ auf Seite 298.

2. Ändern Sie die beiden zentralen WS-Manager, um sie vollständig in Repository-WS-Managern zu erstellen.

Die beiden WS-Manager CHICAGO und CHICAGO2 befinden sich im Hub dieses Netzes. Sie haben entschieden, alle Aktivitäten, die dem Filialcluster zugeordnet sind, auf diese beiden WS-Manager zu konzentrieren. Neben der Bestandsanwendung und den Definitionen für die INVENTQ -Warteschlange sollen diese WS-Manager die beiden vollständigen Repositories für den Cluster hosten. Geben Sie an jedem der beiden Warteschlangenmanager den folgenden Befehl aus:

```
ALTER QMGR REPOS(CHNSTORE)
```

3. Definieren Sie einen CLUSRCVR -Kanal auf jedem Warteschlangenmanager.

Definieren Sie in jedem WS-Manager im Cluster einen Clusterempfängerkanal und einen Clustersenderkanal. Es spielt keine Rolle, welchen Kanal Sie zuerst definieren.

Erstellen Sie eine CLUSRCVR -Definition, um jeden Warteschlangenmanager, seine Netzadresse und andere Informationen für den Cluster zugänglich zu machen. Beispiel für WS-Manager ATLANTA:

```
DEFINE CHANNEL(CHNSTORE.ATLANTA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNNAME(ATLANTA.CHSTORE.COM) CLUSTER(CHNSTORE)
DESCR('Cluster-receiver channel')
```

4. Definieren Sie einen CLUSSDR -Kanal auf jedem WS-Manager.

Erstellen Sie in jedem WS-Manager eine CLUSSDR -Definition, um diesen Warteschlangenmanager mit einem oder einem der vollständigen WS-Manager-Repositories zu verbinden. Sie können z. B. ATLANTA mit CHICAGO2 verknüpfen:

```
DEFINE CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNNAME(CHICAGO2.CHSTORE.COM) CLUSTER(CHNSTORE)
DESCR('Cluster-sender channel to repository queue manager')
```

5. Installieren Sie die Bestandsanwendung unter CHICAGO2.

Sie haben bereits die Bestandsanwendung auf WS-Manager CHICAGO. Jetzt müssen Sie eine Kopie dieser Anwendung auf WS-Manager CHICAGO2 erstellen.

6. Definieren Sie die INVENTQ -Warteschlange auf den zentralen WS-Managern.

Ändern Sie unter CHICAGO die lokale Warteschlangendefinition für die Warteschlange INVENTQ , um die Warteschlange für den Cluster verfügbar zu machen. Geben Sie den folgenden Befehl aus:

```
ALTER QLOCAL(INVENTQ) CLUSTER(CHNSTORE)
```

Geben Sie unter CHICAGO2 eine Definition für die gleiche Warteschlange an:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(CHNSTORE)
```

Unter z/OS können Sie die Option MAKEDEF der Funktion COMMAND von **CSQUTIL** verwenden, um auf CHICAGO2 eine exakte Kopie von INVENTQ auf CHICAGO zu erstellen.

Wenn Sie diese Definitionen erstellen, wird eine Nachricht an die vollständigen Repositories in CHICAGO und CHICAGO2 gesendet, und die darin enthaltenen Informationen werden aktualisiert. Der Warteschlangenmanager ermittelt aus den vollständigen Repositories, wenn er eine Nachricht an den INVENTQ einreicht, dass es eine Auswahl an Zieladressen für die Nachrichten gibt.

7. Überprüfen Sie, ob die Clusteränderungen weitergegeben wurden.

Überprüfen Sie, ob die Definitionen, die Sie im vorherigen Schritt erstellt haben, durch den Cluster weitergegeben wurden. Geben Sie den folgenden Befehl in einem vollständigen WS-Manager-Repository aus:

## Hinzufügen eines neuen, miteinander verbundenen Clusters

Fügen Sie einen neuen Cluster hinzu, der einige WS-Manager mit einem vorhandenen Cluster gemeinsam nutzt.

### Vorbereitende Schritte

#### Anmerkung:

1. Damit Änderungen an einem Cluster im gesamten Cluster weitergegeben werden können, muss immer mindestens ein vollständiges Repository verfügbar sein. Stellen Sie sicher, dass Ihre Repositories verfügbar sind, bevor Sie diese Task starten.
2. Bevor Sie mit dieser Task beginnen, müssen Sie nach Warteschlangennamen suchen und die Auswirkungen kennen. Möglicherweise müssen Sie eine Warteschlange umbenennen oder Warteschlangennamen einrichten, bevor Sie fortfahren können.

Szenario:

- Ein WebSphere MQ -Cluster wurde wie in [„Konvertieren eines vorhandenen Netzes in einen Cluster“](#) auf Seite 251 beschrieben eingerichtet.
- Es soll ein neuer Cluster mit dem Namen MAILORDER implementiert werden. Dieser Cluster umfasst vier der Warteschlangenmanager im CHNSTORE -Cluster: CHICAGO, CHICAGO2, SEATTLE und ATLANTA sowie zwei weitere Warteschlangenmanager: HARTFORD und OMAHA. Die Anwendung MAILORDER wird auf dem System in Omaha ausgeführt, das mit dem Warteschlangenmanager OMAHA verbunden ist. Sie wird von den anderen Warteschlangenmanagern im Cluster gesteuert, die Nachrichten in die Warteschlange MORDERQ einreihen.
- Die vollständigen Repositories für den MAILORDER -Cluster werden auf den beiden Warteschlangenmanagern CHICAGO und CHICAGO2 verwaltet.
- Das Netzprotokoll ist TCP.

### Informationen zu diesem Vorgang

Führen Sie die folgenden Schritte aus, um einen neuen, miteinander verbundenen Cluster hinzuzufügen.

### Vorgehensweise

1. Erstellen Sie eine Namensliste der Clusternamen.

Die Warteschlangenmanager mit vollständigem Repository unter CHICAGO und CHICAGO2 werden nun die vollständigen Repositories für beide Cluster CHNSTORE und MAILORDER enthalten. Erstellen Sie zunächst eine Namensliste, die die Namen der Cluster enthält. Definieren Sie die Namensliste unter CHICAGO und CHICAGO2 wie folgt:

```
DEFINE NAMELIST(CHAINMAIL)
 DESCR('List of cluster names')
 NAMES(CHNSTORE, MAILORDER)
```

2. Ändern Sie die beiden WS-Manager-Definitionen.

Ändern Sie nun die beiden WS-Manager-Definitionen in CHICAGO und CHICAGO2. Derzeit zeigen diese Definitionen, dass die Warteschlangenmanager vollständige Repositories für den Cluster CHNSTORE enthalten. Ändern Sie diese Definition, um anzuzeigen, dass die Warteschlangenmanager vollständige Repositories für alle in der CHAINMAIL -Namensliste aufgelisteten Cluster enthalten. Ändern Sie die CHICAGO -und CHICAGO2 -Warteschlangenmanagerdefinitionen:

```
ALTER QMGR REPOS(' ') REPOSNL(CHAINMAIL)
```

3. Ändern Sie die CLUSRCVR -Kanäle unter CHICAGO und CHICAGO2.

Die Kanaldefinitionen von CLUSRCVR in CHICAGO und CHICAG02 zeigen, dass die Kanäle im Cluster CHNSTORE verfügbar sind. Sie müssen die Clusterempfängerdefinition ändern, um anzuzeigen, dass die Kanäle für alle Cluster verfügbar sind, die in der Namensliste von CHAINMAIL aufgelistet sind. Ändern Sie die Clusterempfängerdefinition in CHICAGO:

```
ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSRCVR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

Geben Sie in CHICAG02den folgenden Befehl ein:

```
ALTER CHANNEL(CHNSTORE.CHICAG02) CHLTYPE(CLUSRCVR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

4. Ändern Sie die CLUSSDR -Kanäle unter CHICAGO und CHICAG02.

Ändern Sie die beiden CLUSSDR -Kanaldefinitionen, um die Namensliste hinzuzufügen. Geben Sie in CHICAGO den folgenden Befehl ein:

```
ALTER CHANNEL(CHNSTORE.CHICAG02) CHLTYPE(CLUSSDR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

Geben Sie in CHICAG02den folgenden Befehl ein:

```
ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSSDR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

5. Erstellen Sie eine Namensliste unter SEATTLE und ATLANTA.

Da SEATTLE und ATLANTA zu mehreren Clustern gehören, müssen Sie eine Namensliste mit den Namen der Cluster erstellen. Definieren Sie die Namensliste unter SEATTLE und ATLANTA wie folgt:

```
DEFINE NAMELIST(CHAINMAIL)
DESCR('List of cluster names')
NAMES(CHNSTORE, MAILORDER)
```

6. Ändern Sie die CLUSRCVR -Kanäle unter SEATTLE und ATLANTA.

Die Kanaldefinitionen von CLUSRCVR in SEATTLE und ATLANTA zeigen, dass die Kanäle im Cluster CHNSTORE verfügbar sind. Ändern Sie die Clusterempfangskanaldefinitionen, um zu zeigen, dass die Kanäle für alle Cluster verfügbar sind, die in der CHAINMAIL -Namensliste aufgeführt sind. Geben Sie in SEATTLEden folgenden Befehl ein:

```
ALTER CHANNEL(CHNSTORE.SEATTLE) CHLTYPE(CLUSRCVR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

Geben Sie in ATLANTAden folgenden Befehl ein:

```
ALTER CHANNEL(CHNSTORE.ATLANTA) CHLTYPE(CLUSRCVR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

7. Ändern Sie die CLUSSDR -Kanäle unter SEATTLE und ATLANTA.

Ändern Sie die beiden CLUSSDR -Kanaldefinitionen, um die Namensliste hinzuzufügen. Geben Sie in SEATTLE den folgenden Befehl ein:

```
ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSSDR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

Geben Sie in ATLANTAden folgenden Befehl ein:

```
ALTER CHANNEL(CHNSTORE.CHICAG02) CHLTYPE(CLUSSDR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

8. Definieren Sie CLUSRCVR -und CLUSSDR -Kanäle unter HARTFORD und OMAHA .

Definieren Sie auf den beiden neuen Warteschlangenmanagern HARTFORD und OMAHA Cluster-Empfänger- und Clustersenderkanäle. Es spielt keine Rolle, in welcher Reihenfolge Sie die Definitionen machen. Geben Sie in HARTFORD Folgendes ein:

```
DEFINE CHANNEL(MAILORDER.HARTFORD) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(HARTFORD.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-receiver channel for HARTFORD')

DEFINE CHANNEL(MAILORDER.CHICAGO) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(CHICAGO.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-sender channel from HARTFORD to repository at CHICAGO')
```

Geben Sie in OMAHA Folgendes ein:

```
DEFINE CHANNEL(MAILORDER.OMAHA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(OMAHA.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-receiver channel for OMAHA')

DEFINE CHANNEL(MAILORDER.CHICAGO) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(CHICAGO.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-sender channel from OMAHA to repository at CHICAGO')
```

9. Definieren Sie die Warteschlange MORDERQ unter OMAHA .

Der letzte Schritt für diese Task ist die Definition der Warteschlange MORDERQ auf dem WS-Manager OMAHA . Geben Sie in OMAHA Folgendes ein:

```
DEFINE QLOCAL(MORDERQ) CLUSTER(MAILORDER)
```

10. Überprüfen Sie, ob die Clusteränderungen weitergegeben wurden.

Überprüfen Sie, ob die Definitionen, die Sie mit den vorherigen Schritten erstellt haben, durch den Cluster weitergegeben wurden. Geben Sie die folgenden Befehle in einem Repository-WS-Manager aus:

```
DIS QCLUSTER (MORDERQ)
DIS CLUSQMGR
```

11.

## Ergebnisse

Der Cluster, der von dieser Task konfiguriert wird, wird in [Abbildung 42 auf Seite 257](#) angezeigt.

Jetzt haben wir zwei überlappende Cluster. Die vollständigen Repositorys für beide Cluster befinden sich in CHICAGO und CHICAGO2. Die Anwendung für die Mail-Bestellung, die auf OMAHA ausgeführt wird, ist unabhängig von der Bestandsanwendung, die unter CHICAGO ausgeführt wird. Einige der Warteschlangenmanager, die sich im CHNSTORE -Cluster befinden, befinden sich jedoch auch im MAILORDER -Cluster, sodass sie Nachrichten an beide Anwendungen senden können. Bevor Sie diese Task zur Überlappung von zwei Clustern durchführen können, müssen Sie die Möglichkeit von Warteschlangennamenskollisionen kennen.

Angenommen, in NEWYORK im Cluster CHNSTORE und in OMAHA im Cluster MAILORDER gibt es eine Warteschlange namens ACCOUNTQ . Wenn Sie sich mit den Clustern überschneiden und dann eine Anwendung unter SEATTLE eine Nachricht in die Warteschlange einreicht ACCOUNTQ , kann die Nachricht an jede Instanz von ACCOUNTQ gesendet werden.



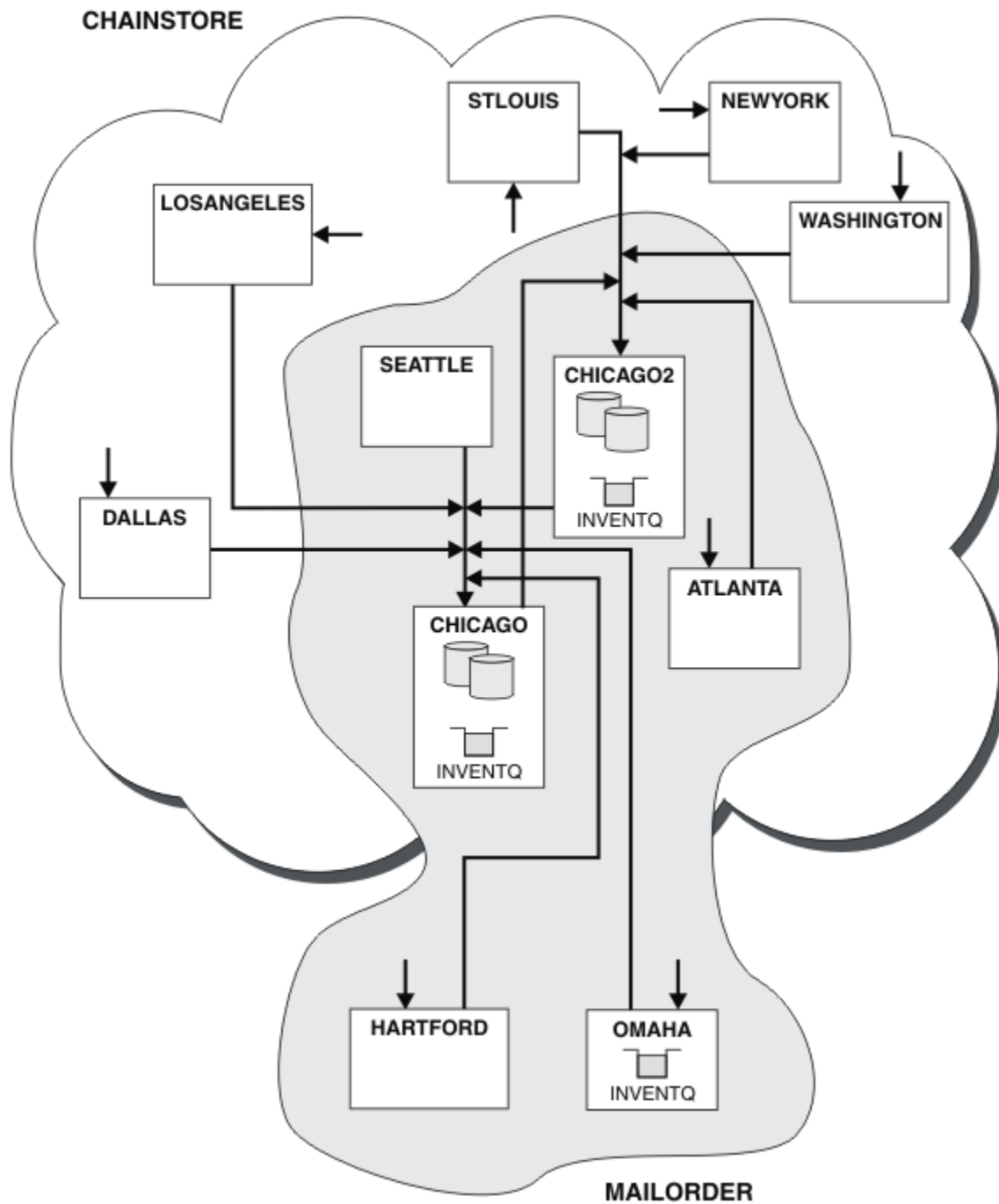


Abbildung 42. Interverbundene Cluster

### Nächste Schritte

Angenommen, Sie möchten den MAILORDER-Cluster mit dem CHNSTORE-Cluster zusammenführen, um einen großen Cluster mit dem Namen CHNSTORE zu bilden.

Gehen Sie wie folgt vor, um den MAILORDER -Cluster mit dem CHNSTORE -Cluster zusammenzuführen, sodass CHICAGO und CHICAGO2 die vollständigen Repositorys enthalten:

- Ändern Sie die Warteschlangenmanagerdefinitionen für CHICAGO und CHICAGO2, entfernen Sie das Attribut REPOSNL, das die Namensliste angibt (CHAINMAIL), und ersetzen Sie es durch ein REPOS-Attribut, das den Clusternamen angibt (CHNSTORE). Beispiel:

```
ALTER QMGR(CHICAGO) REPOSNL(' ') REPOS(CHNSTORE)
```

- Ändern Sie auf jedem WS-Manager im MAILORDER -Cluster alle Kanaldefinitionen und Warteschlangendefinitionen, um den Wert des Attributs CLUSTER von MAILORDER in CHNSTORE zu ändern. Geben Sie beispielsweise bei HARTFORD Folgendes ein:

```
ALTER CHANNEL(MAILORDER.HARTFORD) CLUSTER(CHNSTORE)
```

Geben Sie bei OMAHA Folgendes ein:

```
ALTER QLOCAL(MORDERQ) CLUSTER(CHNSTORE)
```

- Ändern Sie alle Definitionen, die die Clusternamensliste angeben CHAINMAIL, d. h. die CLUSRCVR -und CLUSSDR -Kanaldefinitionen unter CHICAGO, CHICAGO2, SEATTLE und ATLANTA, um stattdessen den Cluster anzugeben CHNSTORE.

In diesem Beispiel sehen Sie den Vorteil der Verwendung von Namenslisten. Anstatt die Warteschlangenmanagerdefinitionen für CHICAGO und CHICAGO2 zu ändern, können Sie den Wert der Namensliste CHAINMAIL ändern. Ebenso können Sie das erforderliche Ergebnis erzielen, indem Sie die Namensliste ändern, anstatt die CLUSRCVR -und CLUSSDR -Kanaldefinitionen in CHICAGO, CHICAGO2, SEATTLE und ATLANTA zu ändern.

## Clusternetzwerk entfernen

Entfernen Sie einen Cluster aus einem Netz und stellen Sie die verteilte Warteschlangenkonfiguration wieder her.

### Vorbereitende Schritte

**Anmerkung:** Damit Änderungen an einem Cluster im gesamten Cluster weitergegeben werden können, muss immer mindestens ein vollständiges Repository verfügbar sein. Stellen Sie sicher, dass Ihre Repositories verfügbar sind, bevor Sie diese Task starten.

Szenario:

- Ein IBM WebSphere MQ-Cluster wurde wie in [„Konvertieren eines vorhandenen Netzes in einen Cluster“](#) auf Seite 251 beschrieben konfiguriert.
- Dieser Cluster soll jetzt aus dem System entfernt werden. Das Netz von Warteschlangenmanagern soll so funktionieren, wie es vor der Implementierung des Clusters ausgeführt wurde.

### Informationen zu diesem Vorgang

Führen Sie die folgenden Schritte aus, um ein Clusternetz zu entfernen.

### Vorgehensweise

1. Entfernen Sie Clusterwarteschlangen aus dem CHNSTORE -Cluster.

Ändern Sie unter CHICAGO und CHICAGO2 die Definition der lokalen Warteschlange für die Warteschlange INVENTQ, um die Warteschlange aus dem Cluster zu entfernen. Geben Sie den folgenden Befehl aus:

```
ALTER QLOCAL(INVENTQ) CLUSTER(' ')
```

Wenn Sie die Warteschlange ändern, werden die Informationen in den vollständigen Repositories aktualisiert und im gesamten Cluster repliziert. Aktive Anwendungen, die MQ00\_BIND\_NOT\_FIXED verwenden, und Anwendungen, die MQ00\_BIND\_AS\_Q\_DEF verwenden, bei denen die Warteschlange

mit DEFBIND (NOTFIXED) definiert wurde, schlagen beim nächsten versuchten Aufruf MQPUT oder MQPUT1 fehl. Der Ursachencode MQRC\_UNKOWN\_OBJECT\_NAME wird zurückgegeben.

Sie müssen Schritt 1 nicht zuerst ausführen, aber wenn Sie dies nicht tun, führen Sie es stattdessen nach Schritt 4 aus.

2. Stoppen Sie alle Anwendungen, die Zugriff auf die Clusterwarteschlange haben.

Stoppen Sie alle Anwendungen, die Zugriff auf Clusterwarteschlangen haben. Wenn Sie dies nicht tun, bleiben einige Clusterinformationen möglicherweise auf dem lokalen Warteschlangenmanager, wenn Sie den Cluster in Schritt 5 aktualisieren. Diese Informationen werden entfernt, wenn alle Anwendungen gestoppt wurden und die Clusterkanäle getrennt wurden.

3. Entfernen Sie das Repository-Attribut aus den vollständigen WS-Managern des Repositorys.

Ändern Sie unter CHICAGO und CHICAGO2 die Warteschlangenmanagerdefinitionen, um das Repository-Attribut zu entfernen. Geben Sie dazu den folgenden Befehl aus:

```
ALTER QMGR REPOS(' ')
```

Die WS-Manager informieren die anderen WS-Manager im Cluster, dass sie nicht mehr die vollständigen Repositorys enthalten. Wenn die anderen WS-Manager diese Informationen empfangen, wird eine Nachricht angezeigt, die angibt, dass das vollständige Repository beendet wurde. Außerdem wird mindestens eine Nachricht angezeigt, die angibt, dass keine Repositorys mehr für den Cluster CHNSTORE verfügbar sind.

4. Entfernen Sie die Clusterkanäle.

Entfernen Sie auf CHICAGO die Clusterkanäle:

```
ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR) CLUSTER(' ')\nALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSRCVR) CLUSTER(' ')
```

**Anmerkung:** Es ist wichtig, zuerst den Befehl CLUSSDR und dann den Befehl CLUSRCVR auszugeben. Geben Sie zuerst den Befehl CLUSRCVR und dann den Befehl CLUSSDR nicht aus. Dadurch werden unbestätigte Kanäle erstellt, die über den Status STOPPED verfügen. Anschließend müssen Sie einen START CHANNEL -Befehl absetzen, um die gestoppten Kanäle wiederherzustellen, z. B. START CHANNEL(CHNSTORE.CHICAGO).

Es werden Nachrichten angezeigt, die angeben, dass keine Repositorys für den Cluster CHNSTORE vorhanden sind.

Wenn Sie die Clusterwarteschlangen nicht wie in Schritt 1 beschrieben entfernt haben, tun Sie dies jetzt.

5. Stoppen Sie die Clusterkanäle.

Stoppen Sie auf CHICAGO die Clusterkanäle mit den folgenden Befehlen:

```
STOP CHANNEL(CHNSTORE.CHICAGO2)\nSTOP CHANNEL(CHNSTORE.CHICAGO)
```

6. Wiederholen Sie die Schritte 4 und 5 für die einzelnen WS-Manager im Cluster.
7. Stoppen Sie die Clusterkanäle, und entfernen Sie anschließend alle Definitionen für die Clusterkanäle und Clusterwarteschlangen von jedem Warteschlangenmanager.
8. Optional: Löschen Sie die zwischengespeicherten Clusterinformationen, die vom WS-Manager gehalten werden.

Obwohl die WS-Manager nicht mehr Mitglieder des Clusters sind, behalten sie jeweils eine zwischengespeicherte Kopie von Informationen zum Cluster. Wenn Sie diese Daten entfernen möchten, lesen Sie die Task „WS-Manager in den Status vor dem Cluster zurückschreiben“ auf Seite 264.

9. Ferne Warteschlangendefinitionen für INVENTQ ersetzen

Damit das Netz weiterhin funktionieren kann, ersetzen Sie die Definition der fernen Warteschlange für den INVENTQ in jedem WS-Manager.

10. Aufkreichen Sie den Cluster.

Löschen Sie keine Warteschlangen-oder Kanaldefinitionen, die nicht mehr benötigt werden.

## Entfernen eines Warteschlangenmanagers aus einem Cluster

Entfernen Sie einen Warteschlangenmanager aus einem Cluster in Szenarios, in denen der WS-Manager normalerweise mit mindestens einem vollständigen Repository im Cluster kommunizieren kann.

### Vorbereitende Schritte

Diese Methode ist die bewährte Methode für Szenarios, in denen mindestens ein vollständiges Repository verfügbar ist und von dem Warteschlangenmanager, der entfernt wird, kontaktiert werden kann. Diese Methode beinhaltet den geringsten manuellen Eingriff und ermöglicht dem WS-Manager die Aushandlung eines kontrollierten Rückzugs aus dem Cluster. Wenn der Warteschlangenmanager, der entfernt wird, keine Verbindung zu einem vollständigen Repository aufnehmen kann, finden Sie weitere Informationen in „Entfernen eines Warteschlangenmanagers aus einem Cluster: Alternative Methode“ auf Seite 262.

Bevor Sie den WS-Manager aus dem Cluster entfernen, müssen Sie sicherstellen, dass der WS-Manager keine Ressourcen mehr bereitstellt, die vom Cluster benötigt werden:

- Wenn der WS-Manager ein vollständiges Repository enthält, führen Sie die Schritte 1 bis 4 unter „Vollrepositorys in einen anderen WS-Manager verschieben“ auf Seite 249 aus.
- Wenn der Warteschlangenmanager Clusterwarteschlangen enthält, führen Sie die Schritte 1 bis 7 von „Clusterwarteschlange aus einem WS-Manager entfernen“ auf Seite 246 aus.
- Wenn der Warteschlangenmanager Clusterthemen enthält, löschen Sie die Themen (z. B. mit dem Befehl `DELETE TOPIC`) oder verschieben Sie sie auf andere Hosts.

**Anmerkung:** Wenn Sie einen Warteschlangenmanager aus einem Cluster entfernen und der WS-Manager weiterhin ein Clusterthema enthält, versucht der Warteschlangenmanager möglicherweise weiterhin, Veröffentlichungen an die Warteschlangenmanager zu übergeben, die im Cluster bleiben, bis das Thema gelöscht wird.

### Informationen zu diesem Vorgang

Diese Beispieltask entfernt den WS-Manager LONDON aus dem INVENTORY -Cluster. Der INVENTORY -Cluster wird wie in „WS-Manager zu einem Cluster hinzufügen“ auf Seite 211 beschrieben eingerichtet und wie in „Clusterwarteschlange aus einem WS-Manager entfernen“ auf Seite 246 beschrieben geändert.

Der Prozess zum Entfernen eines Warteschlangenmanagers aus einem Cluster ist komplizierter als der Prozess zum Hinzufügen eines Warteschlangenmanagers.

Wenn ein Warteschlangenmanager einem Cluster beiträgt, verfügen die vorhandenen Mitglieder des Clusters über keine Kenntnisse des neuen Warteschlangenmanagers und haben daher keine Interaktionen mit dem Cluster. Es müssen neue Sender- und Empfängerkanäle auf dem Verbindungswarteschlangenmanager erstellt werden, damit sie eine Verbindung zu einem vollständigen Repository herstellen können.

Wenn ein Warteschlangenmanager aus einem Cluster entfernt wird, ist es wahrscheinlich, dass Anwendungen, die mit dem Warteschlangenmanager verbunden sind, Objekte verwenden, wie z. B. Warteschlangen, die an anderer Stelle im Cluster enthalten sind. Auch Anwendungen, die mit anderen Warteschlangenmanagern im Cluster verbunden sind, verwenden möglicherweise Objekte, die auf dem Zielwarteschlangenmanager gehostet sind. Als Ergebnis dieser Anwendungen kann der aktuelle WS-Manager zusätzliche Senderkanäle erstellen, um die Kommunikation mit anderen Cluster-Mitgliedern als dem vollständigen Repository herzustellen, das für die Teilnahme am Cluster verwendet wurde. Jeder WS-Manager im Cluster verfügt über eine zwischengespeicherte Kopie von Daten, die andere Cluster-Mitglieder beschreiben. Dies kann die Entfernung einschließen, die gerade entfernt wird.

### Vorgehensweise

1. Ändern Sie die manuell definierten Clusterempfängerkanäle, um sie aus dem Cluster zu entfernen, auf WS-Manager LONDON:

```
ALTER CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) CLUSTER(' ')
```

2. Ändern Sie die manuell definierten Clustersenderkanäle, um sie aus dem Cluster zu entfernen, auf WS-Manager LONDON:

```
ALTER CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSSDR) CLUSTER(' ')
```

Die anderen WS-Manager im Cluster lernen, dass dieser Warteschlangenmanager und seine Clusterressourcen nicht mehr Teil des Clusters sind.

3. Überwachen Sie die Clustersendewarteschlange auf dem Warteschlangenmanager LONDON, bis keine Nachrichten vorhanden sind, die darauf warten, in ein vollständiges Repository im Cluster fließen zu können.

```
DISPLAY CHSTATUS(INVENTORY.LONDON) XQMSGSA
```

Wenn Nachrichten in der Übertragungswarteschlange verbleiben, stellen Sie fest, warum sie nicht an die vollständigen Repositories von PARIS und NEWYORK gesendet werden, bevor Sie fortfahren.

## Ergebnisse

Der WS-Manager LONDON ist nicht mehr Teil des Clusters. Er kann jedoch weiterhin als unabhängiger Warteschlangenmanager eingesetzt werden.

## Nächste Schritte

Das Ergebnis dieser Änderungen kann bestätigt werden, indem der folgende Befehl für die verbleibenden Member des Clusters ausgegeben wird:

```
DISPLAY CLUSQMgr(LONDON)
```

Der Warteschlangenmanager wird so lange angezeigt, bis die automatisch definierten Clustersenderkanäle gestoppt wurden. Sie können warten, bis dies geschehen ist, oder Sie können mit dem folgenden Befehl fortfahren, um aktive Instanzen zu überwachen:

```
DISPLAY CHANNEL(INVENTORY.LONDON)
```

Wenn Sie sicher sind, dass diesem WS-Manager keine Nachrichten mehr zugestellt werden, können Sie die Clustersenderkanäle auf LONDON stoppen, indem Sie den folgenden Befehl für die verbleibenden Member des Clusters ausgeben:

```
STOP CHANNEL(INVENTORY.LONDON) STATUS(INACTIVE)
```

Nachdem die Änderungen im gesamten Cluster weitergegeben wurden und keine weiteren Nachrichten an diesen Warteschlangenmanager gesendet werden, stoppen Sie den CLUSRCVR -Kanal unter LONDON und löschen Sie ihn nicht mehr:

```
STOP CHANNEL(INVENTORY.LONDON)
DELETE CHANNEL(INVENTORY.LONDON)
```

Der entfernte WS-Manager kann zu einem späteren Zeitpunkt wieder in den Cluster aufgenommen werden, wie in „WS-Manager zu einem Cluster hinzufügen“ auf Seite 211 beschrieben. Der entfernte Warteschlangenmanager speichert weiterhin die Kenntnisse der verbleibenden Mitglieder des Clusters für bis zu 90 Tage ein. Wenn Sie nicht warten möchten, bis dieser Cache abläuft, kann er zwangsweise entfernt werden, wie in [„WS-Manager in den Status vor dem Cluster zurückschreiben“](#) auf Seite 264 beschrieben.

## **Entfernen eines Warteschlangenmanagers aus einem Cluster: Alternative Methode**

Entfernen Sie einen Warteschlangenmanager aus einem Cluster in Szenarios, in denen der WS-Manager aufgrund eines signifikanten System-oder Konfigurationsproblems nicht mit einem vollständigen Repository im Cluster kommunizieren kann.

### **Vorbereitende Schritte**

Diese alternative Methode zum Entfernen eines Warteschlangenmanagers aus einem Cluster wird manuell gestoppt und löscht alle Clusterkanäle, die den entfernten WS-Manager mit dem Cluster verbinden, und entfernt den Warteschlangenmanager zwangsweise aus dem Cluster. Diese Methode wird in Szenarios verwendet, in denen der Warteschlangenmanager, der entfernt wird, nicht mit einem der vollständigen Repositories kommunizieren kann. Dies kann (z. B.) sein, weil der Warteschlangenmanager nicht mehr funktioniert hat oder weil zwischen dem WS-Manager und dem Cluster ein verlängerter Kommunikationsfehler aufgetreten ist. Verwenden Sie andernfalls die gängigste Methode: [„Entfernen eines Warteschlangenmanagers aus einem Cluster“](#) auf Seite 260.

Bevor Sie den WS-Manager aus dem Cluster entfernen, müssen Sie sicherstellen, dass der WS-Manager keine Ressourcen mehr bereitstellt, die vom Cluster benötigt werden:

- Wenn der WS-Manager ein vollständiges Repository enthält, führen Sie die Schritte 1 bis 4 unter [„Vollrepositorys in einen anderen WS-Manager verschieben“](#) auf Seite 249 aus.
- Wenn der Warteschlangenmanager Clusterwarteschlangen enthält, führen Sie die Schritte 1 bis 7 von [„Clusterwarteschlange aus einem WS-Manager entfernen“](#) auf Seite 246 aus.
- Wenn der Warteschlangenmanager Clusterthemen enthält, löschen Sie die Themen (z. B. mit dem Befehl `DELETE TOPIC`) oder verschieben Sie sie auf andere Hosts.

**Anmerkung:** Wenn Sie einen Warteschlangenmanager aus einem Cluster entfernen und der WS-Manager weiterhin ein Clusterthema enthält, versucht der Warteschlangenmanager möglicherweise weiterhin, Veröffentlichungen an die Warteschlangenmanager zu übergeben, die im Cluster bleiben, bis das Thema gelöscht wird.

### **Informationen zu diesem Vorgang**

Diese Beispieltask entfernt den WS-Manager LONDON aus dem INVENTORY -Cluster. Der INVENTORY -Cluster wird wie in [„WS-Manager zu einem Cluster hinzufügen“](#) auf Seite 211 beschrieben eingerichtet und wie in [„Clusterwarteschlange aus einem WS-Manager entfernen“](#) auf Seite 246 beschrieben geändert.

Der Prozess zum Entfernen eines Warteschlangenmanagers aus einem Cluster ist komplizierter als der Prozess zum Hinzufügen eines Warteschlangenmanagers.

Wenn ein Warteschlangenmanager einem Cluster beiträgt, verfügen die vorhandenen Mitglieder des Clusters über keine Kenntnisse des neuen Warteschlangenmanagers und haben daher keine Interaktionen mit dem Cluster. Es müssen neue Sender- und Empfängerkanäle auf dem Verbindungswarteschlangenmanager erstellt werden, damit sie eine Verbindung zu einem vollständigen Repository herstellen können.

Wenn ein Warteschlangenmanager aus einem Cluster entfernt wird, ist es wahrscheinlich, dass Anwendungen, die mit dem Warteschlangenmanager verbunden sind, Objekte verwenden, wie z. B. Warteschlangen, die an anderer Stelle im Cluster enthalten sind. Auch Anwendungen, die mit anderen Warteschlangenmanagern im Cluster verbunden sind, verwenden möglicherweise Objekte, die auf dem Zielwarteschlangenmanager gehostet sind. Als Ergebnis dieser Anwendungen kann der aktuelle WS-Manager zusätzliche Senderkanäle erstellen, um die Kommunikation mit anderen Cluster-Mitgliedern als dem vollständigen Repository herzustellen, das für die Teilnahme am Cluster verwendet wurde. Jeder WS-Manager im Cluster verfügt über eine zwischengespeicherte Kopie von Daten, die andere Cluster-Mitglieder beschreiben. Dies kann die Entfernung einschließen, die gerade entfernt wird.

Diese Prozedur kann in einem Notfall angemessen sein, wenn es nicht möglich ist, auf den WS-Manager zu warten, um den Cluster ordnungsgemäß zu verlassen.

## Vorgehensweise

1. Stoppen Sie alle Kanäle, die für die Kommunikation mit anderen Warteschlangenmanagern im Cluster verwendet werden. Verwenden Sie `MODE (FORCE)`, um den `CLUSRCVR` -Kanal auf `WS-Manager LONDON` zu stoppen. Andernfalls müssen Sie möglicherweise auf den Kanal des Sende-`WS-Managers` warten, um den Kanal zu stoppen:

```
STOP CHANNEL (INVENTORY.LONDON) MODE (FORCE)
STOP CHANNEL (INVENTORY.TORONTO)
STOP CHANNEL (INVENTORY.PARIS)
STOP CHANNEL (INVENTORY.NEWYORK)
```

2. Überwachen Sie den Kanalstatus auf dem Warteschlangenmanager `LONDON`, bis die Kanäle gestoppt werden:

```
DISPLAY CHSTATUS (INVENTORY.LONDON)
DISPLAY CHSTATUS (INVENTORY.TORONTO)
DISPLAY CHSTATUS (INVENTORY.PARIS)
DISPLAY CHSTATUS (INVENTORY.NEWYORK)
```

Nach dem Stoppen der Kanäle werden keine weiteren Anwendungsnachrichten an oder von den anderen `WS-Managern` in dem Cluster gesendet.

3. Löschen Sie die manuell definierten Clusterkanäle auf dem `WS-Manager LONDON`:

```
DELETE CHANNEL (INVENTORY.NEWYORK)
DELETE CHANNEL (INVENTORY.TORONTO)
```

4. Die verbleibenden `WS-Manager` im Cluster behalten weiterhin die Kenntnisse des entfernten Warteschlangenmanagers und können weiterhin Nachrichten an diese Warteschlangenmanager senden. Um die Kenntnisse aus den verbleibenden `WS-Managern` zu löschen, setzen Sie den entfernten `WS-Manager` aus dem Cluster in einem der vollständigen Repositories zurück:

```
RESET CLUSTER (INVENTORY) ACTION (FORCEREMOVE) QMNAME (LONDON) QUEUES (YES)
```

Wenn ein anderer Warteschlangenmanager im Cluster vorhanden ist, der denselben Namen wie der entfernte Warteschlangenmanager hat, geben Sie **QMID** des entfernten Warteschlangenmanagers an.

## Ergebnisse

Der `WS-Manager LONDON` ist nicht mehr Teil des Clusters. Er kann jedoch weiterhin als unabhängiger Warteschlangenmanager eingesetzt werden.

## Nächste Schritte

Das Ergebnis dieser Änderungen kann bestätigt werden, indem der folgende Befehl für die verbleibenden Member des Clusters ausgegeben wird:

```
DISPLAY CLUSQMGR (LONDON)
```

Der Warteschlangenmanager wird so lange angezeigt, bis die automatisch definierten Clustersenderkanäle gestoppt wurden. Sie können warten, bis dies geschehen ist, oder Sie können mit dem folgenden Befehl fortfahren, um aktive Instanzen zu überwachen:

```
DISPLAY CHANNEL (INVENTORY.LONDON)
```

Nachdem die Änderungen im gesamten Cluster weitergegeben wurden und keine weiteren Nachrichten an diesen Warteschlangenmanager gesendet werden, löschen Sie den `CLUSRCVR` -Kanal unter `LONDON`:

```
DELETE CHANNEL (INVENTORY.LONDON)
```

Der entfernte WS-Manager kann zu einem späteren Zeitpunkt wieder in den Cluster aufgenommen werden, wie in „[WS-Manager zu einem Cluster hinzufügen](#)“ auf Seite 211 beschrieben. Der entfernte Warteschlangenmanager speichert weiterhin die Kenntnisse der verbleibenden Mitglieder des Clusters für bis zu 90 Tage ein. Wenn Sie nicht warten möchten, bis dieser Cache abläuft, kann er zwangsweise entfernt werden, wie in „[WS-Manager in den Status vor dem Cluster zurückschreiben](#)“ auf Seite 264 beschrieben.

## WS-Manager in den Status vor dem Cluster zurückschreiben

Wenn ein WS-Manager aus einem Cluster entfernt wird, behält er die Kenntnisse der verbleibenden Cluster-Member bei. Dieses Wissen verfällt schließlich und wird automatisch gelöscht. Wenn Sie es jedoch sofort löschen möchten, können Sie die Schritte in diesem Thema verwenden.

### Vorbereitende Schritte

Es wird davon ausgegangen, dass der Warteschlangenmanager aus dem Cluster entfernt wurde und keine Arbeit mehr im Cluster mehr ausführt. Beispielsweise empfangen die Warteschlangen keine Nachrichten mehr aus dem Cluster, und es warten keine Anwendungen auf Nachrichten, die in diese Warteschlangen eintreffen.

**Wichtig:** Wenn Sie einen WS-Manager aus einem Cluster entfernen und ihn mit REPOS (YES) aktualisieren, können Sie ihn nicht wieder hinzufügen, indem Sie einfach das Clusterattribut CLUSRCVR ändern. Nachdem Sie das CLUSTER-Attribut des CLUSRCVR in einen nicht leeren Wert geändert haben (d. h. den Clusternamen), müssen Sie zusätzlich den Befehl zum Aktualisieren des Clusters mit REPOS (NO) absetzen. Dann werden die internen Folgeummern im CLUSRCVR auf den aktuellen Stand gebracht. Anschließend kann sich der Warteschlangenmanager erfolgreich wieder in die vollständigen Repositorys und die übrigen Clustermitglieder einführen. (Beachten Sie, dass die REPOS (NO) -Version des Befehls ausgeführt werden muss, nachdem der CLUSRCVR-Kanal den richtigen Clusternamen erhalten hat.)

Diese Einschränkung gilt nur für IBM WebSphere MQ Version 7.5 .

### Informationen zu diesem Vorgang

Wenn ein Warteschlangenmanager aus einem Cluster entfernt wird, behält er die Kenntnisse der verbleibenden Cluster-Member für bis zu 90 Tage bei. Dies kann Systemvorteile haben, insbesondere, wenn der WS-Manager schnell wieder in den Cluster aufgenommen wird. Wenn dieses Wissen schließlich abläuft, wird es automatisch gelöscht. Es gibt jedoch Gründe, warum Sie diese Informationen lieber manuell löschen möchten. Beispiel:

- Möglicherweise möchten Sie bestätigen, dass Sie jede Anwendung in diesem WS-Manager gestoppt haben, die zuvor die Clusterressourcen verwendet hat. Solange die Kenntnisse der verbleibenden Cluster-Member nicht mehr vorhanden sind, schreibt jede solche Anwendung weiterhin in eine Übertragungswarteschlange. Nachdem die Clusterkenntnisse gelöscht wurden, generiert das System eine Fehlernachricht, wenn eine solche Anwendung versucht, Clusterressourcen zu verwenden.
- Wenn Sie Statusinformationen für den Warteschlangenmanager anzeigen, können Sie es vorziehen, Informationen über verbleibende Cluster-Member nicht zu verlaufen.

In dieser Task wird der INVENTORY -Cluster als Beispiel verwendet. Der Warteschlangenmanager LONDON wurde wie in „[Entfernen eines Warteschlangenmanagers aus einem Cluster](#)“ auf Seite 260 beschrieben aus dem INVENTORY -Cluster entfernt. Wenn Sie die Kenntnisse der verbleibenden Member des Clusters löschen möchten, geben Sie die folgenden Befehle im LONDON -Warteschlangenmanager aus.

### Vorgehensweise

1. Entfernen Sie alle Speicher der anderen WS-Manager im Cluster aus diesem Warteschlangenmanager:

```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

2. Überwachen Sie den Warteschlangenmanager, bis alle Clusterressourcen nicht mehr verfügbar sind:



```
DISPLAY CLUSQMGR(*) CLUSTER(INVENTORY)
DISPLAY QCLUSTER(*) CLUSTER(INVENTORY)
DISPLAY TOPIC(*) CLUSTER(INVENTORY)
```

## Zugehörige Konzepte

### Cluster

„Vergleich von Clustering und verteilter Steuerung von Warteschlangen“ auf Seite 174

Vergleichen Sie die Komponenten, die für die Verbindung von WS-Managern mit verteilter Steuerung von Warteschlangen und Clustering definiert werden müssen.

„Komponenten eines Clusters“ auf Seite 176

Cluster bestehen aus Warteschlangenmanagern, Clusterrepositorys, Clusterkanälen und Clusterwarteschlangen.

„IBM WebSphere MQ -Cluster verwalten“ auf Seite 200

Sie können IBM WebSphere MQ -Cluster erstellen, erweitern und verwalten.

## Verwalten eines Warteschlangenmanagers

Setzen Sie einen WS-Manager aus einem Cluster aus und setzen Sie sie wieder fort, um die Wartung durchzuführen.

## Informationen zu diesem Vorgang

Von Zeit zu Zeit müssen Sie möglicherweise eine Wartung für einen Warteschlangenmanager ausführen, der Teil eines Clusters ist. Sie müssen beispielsweise möglicherweise Sicherungen der Daten in den zugehörigen Warteschlangen erstellen oder Fixes auf die Software anwenden. Wenn der Warteschlangenmanager alle Warteschlangen enthält, müssen seine Aktivitäten ausgesetzt werden. Wenn die Wartung abgeschlossen ist, können die Aktivitäten wieder aufgenommen werden.

## Vorgehensweise

1. Setzen Sie einen Warteschlangenmanager aus, indem Sie den Befehl `SUSPEND QMGR runmqsc` ausgeben:

```
SUSPEND QMGR CLUSTER(SALES)
```

Der Befehl `SUSPEND runmqsc` benachrichtigt die Warteschlangenmanager im SALES -Cluster darüber, dass dieser Warteschlangenmanager ausgesetzt wurde.

Der Befehl `SUSPEND QMGR` dient nur dazu, andere Warteschlangenmanager zu informieren, um zu vermeiden, dass Nachrichten an diesen WS-Manager gesendet werden, wenn möglich. Dies bedeutet nicht, dass der WS-Manager inaktiviert ist. Einige Nachrichten, die von diesem WS-Manager bearbeitet werden müssen, werden immer noch an sie gesendet, z. B. wenn dieser Warteschlangenmanager der einzige Host einer Clusterwarteschlange ist.

Solange der WS-Manager ausgesetzt ist, vermeiden die Routinen der Workloadverwaltung, Nachrichten an diese zu senden. Nachrichten, die von diesem WS-Manager bearbeitet werden müssen, enthalten Nachrichten, die vom lokalen Warteschlangenmanager gesendet werden.

WebSphere MQ verwendet einen Lastausgleichsalgorithmus, um festzustellen, welche Ziele geeignet sind, anstatt den lokalen Warteschlangenmanager nach Möglichkeit auszuwählen.

- a) Verwenden Sie die Option `FORCE` im `SUSPEND QMGR` -Befehl, um die Aussetzung eines Warteschlangenmanagers zu erzwingen:

```
SUSPEND QMGR CLUSTER(SALES) MODE(FORCE)
```

`MODE (FORCE)` stoppt zwangsweise alle eingehenden Kanäle von anderen WS-Managern im Cluster. Wenn Sie `MODE (FORCE)` nicht angeben, gilt der Standardwert `MODE (QUIESCE)` .

2. Führen Sie die erforderlichen Wartungsaufgaben aus.

3. Setzen Sie den Befehl `RESUME QMGR runmqsc` ab, um den Warteschlangenmanager wiederaufzunehmen:

```
RESUME QMGR CLUSTER(SALES)
```

### Ergebnisse

Der Befehl `RESUME runmqsc` benachrichtigt die vollständigen Repositorys, dass der Warteschlangenmanager wieder verfügbar ist. Die vollständigen WS-Manager-Repositorys verbreiten diese Informationen an andere Warteschlangenmanager, die Aktualisierungen an Informationen zu diesem Warteschlangenmanager angefordert haben.

## Verwalten der Clusterübertragungswarteschlange

Stellen Sie sicher, dass die Clusterübertragungswarteschlangen zur Verfügung stehen. Sie sind von wesentlicher Bedeutung für die Leistung von Clustern.

### Vorbereitende Schritte

- Stellen Sie sicher, dass die Clusterübertragungswarteschlange nicht voll ist.
- Achten Sie darauf, dass Sie keinen `ALTER runmqsc` -Befehl absetzen, um ihn entweder zu inaktivieren oder versehentlich zu inaktivieren.
- Stellen Sie sicher, dass das Medium, in dem die Clusterübertragungswarteschlange in

## Cluster-WS-Manager wird neu freigegeben

Mit dem Befehl `REFRESH CLUSTER` können Sie automatisch definierte Kanäle und automatisch definierte Clusterobjekte aus dem lokalen Repository entfernen. Es gehen keine Nachrichten verloren.

### Vorbereitende Schritte

Möglicherweise werden Sie von Ihrem IBM Support Center zur Verwendung des Befehls aufgefordert. Verwenden Sie den Befehl nicht ohne sorgfältige Prüfung. Bei großen Clustern kann die Verwendung des Befehls `REFRESH CLUSTER` beispielsweise zu einer Unterbrechung des Clusters führen, während er in Bearbeitung ist, und danach in 27-Tage-Intervallen, wenn die Clusterobjekte Statusaktualisierungen automatisch an alle interessierten Warteschlangenmanager senden. Siehe [„Clustering: Best Practices für REFRESH CLUSTER verwenden“](#) auf Seite 331.

## Informationen zu diesem Vorgang

Ein Warteschlangenmanager kann einen neuen Start in einem Cluster vornehmen. Unter normalen Umständen ist es nicht erforderlich, den Befehl `REFRESH CLUSTER` zu verwenden.

### Vorgehensweise

Geben Sie den Befehl `REFRESH CLUSTER MQSC` von einem Warteschlangenmanager aus, um automatisch definierte Clusterwarteschlangenmanager und Warteschlangenobjekte aus dem lokalen Repository zu entfernen.

Mit dem Befehl werden nur Objekte entfernt, die sich auf andere Warteschlangenmanager beziehen. Es werden keine Objekte entfernt, die sich auf den lokalen Warteschlangenmanager beziehen. Mit dem Befehl werden auch automatisch definierte Kanäle entfernt. Dadurch werden Kanäle entfernt, die keine Nachrichten in der Clusterübertragungswarteschlange enthalten und nicht an einen vollständigen WS-Manager-Repository angeschlossen sind.

## Ergebnisse

Mit dem Befehl `REFRESH CLUSTER` kann ein WS-Manager in Bezug auf seinen vollständigen Repository-Inhalt wirksam kalt gestartet werden. IBM WebSphere MQ stellt sicher, dass keine Daten aus den Warteschlangen verloren gehen.

### Zugehörige Konzepte

„Clustering: Best Practices für `REFRESH CLUSTER` verwenden“ auf Seite 331

Sie verwenden den Befehl **`REFRESH CLUSTER`**, um alle lokal gespeicherten Informationen zu einem Cluster zu löschen und diese Informationen aus den vollständigen Repositories im Cluster erneut zu erstellen. Sie sollten diesen Befehl nicht verwenden, außer in außergewöhnlichen Umständen. Wenn Sie es verwenden müssen, gibt es besondere Hinweise darauf, wie Sie es verwenden. Diese Informationen sind ein Leitfaden, der auf Tests und Feedback von Kunden basiert.

## Warteschlangenmanager wiederherstellen

Verwenden Sie den Befehl `REFRESH CLUSTER runmqsc`, um die Clusterinformationen zu einem Warteschlangenmanager auf den neuesten Stand zu bringen. Führen Sie diese Prozedur aus, nachdem Sie einen Warteschlangenmanager aus einer Zeitpunktsicherung wiederhergestellt haben.

### Vorbereitende Schritte

Sie haben einen Clusterwarteschlangenmanager aus einer zeitpunktbasierter Sicherung wiederhergestellt.

### Informationen zu diesem Vorgang

Um einen Warteschlangenmanager in einem Cluster wiederherzustellen, stellen Sie den Warteschlangenmanager wieder her und aktualisieren Sie anschließend die Clusterinformationen mit dem Befehl `REFRESH CLUSTER runmqsc`.

**Anmerkung:** Bei großen Clustern kann der Befehl **`REFRESH CLUSTER`** während seiner Ausführung und danach in 27-Tage-Intervallen, wenn die Clusterobjekte ihre Statusaktualisierungen automatisch an alle interessierten Warteschlangenmanager hochladen, zu Unterbrechungen führen. Nähere Informationen hierzu erhalten Sie im Abschnitt Die Aktualisierung in einem großen Cluster kann sich auf die Leistung und Verfügbarkeit auswirken.

### Vorgehensweise

Geben Sie den Befehl `REFRESH CLUSTER` auf dem wiederhergestellten Warteschlangenmanager für alle Cluster aus, an denen der Warteschlangenmanager beteiligt ist.

### Nächste Schritte

Es ist nicht erforderlich, den Befehl `REFRESH CLUSTER` auf einem anderen WS-Manager auszugeben.

### Zugehörige Konzepte

„Clustering: Best Practices für `REFRESH CLUSTER` verwenden“ auf Seite 331

Sie verwenden den Befehl **`REFRESH CLUSTER`**, um alle lokal gespeicherten Informationen zu einem Cluster zu löschen und diese Informationen aus den vollständigen Repositories im Cluster erneut zu erstellen. Sie sollten diesen Befehl nicht verwenden, außer in außergewöhnlichen Umständen. Wenn Sie es verwenden müssen, gibt es besondere Hinweise darauf, wie Sie es verwenden. Diese Informationen sind ein Leitfaden, der auf Tests und Feedback von Kunden basiert.

## Clusterkanäle für Verfügbarkeit konfigurieren

Befolgen Sie die bewährten Konfigurationsverfahren, um Clusterkanäle störungsfrei zu halten, wenn unterbrochenen Netzstopps vorhanden sind.

## Vorbereitende Schritte

Cluster entlasten Sie von der Notwendigkeit, Kanäle zu definieren, aber Sie müssen sie trotzdem verwalten. Dieselbe Kanaltechnologie wird für die Kommunikation zwischen Warteschlangenmanagern in einem Cluster verwendet, wie es in der verteilten Steuerung von Warteschlangen verwendet wird. Um Informationen über Clusterkanäle zu erhalten, müssen Sie mit den folgenden Themen vertraut sein:

- Funktionsweise von Kanälen
- Wie Sie ihren Status finden
- Kanalexits verwenden

## Informationen zu diesem Vorgang

Es kann sein, dass Sie die folgenden Punkte besonders berücksichtigen:

## Vorgehensweise

Beachten Sie bei der Konfiguration von Clusterkanälen die folgenden Punkte:

- Wählen Sie Werte für HBINT oder KAINTE auf Clustersenderkanälen und Clusterempfängerkanälen aus, die das Netz nicht mit vielen Überwachungssignalen belasten oder die Nachrichtenflüsse beibehalten. Ein Intervall von weniger als 10 Sekunden führt zu falschen Fehlern, wenn sich Ihr Netz manchmal verlangsamt und Verzögerungen in dieser Länge einführt.
- Legen Sie den Wert für BATCHHB fest, um das Fenster zu verkleinern, um eine Nachricht mit einem Fehler zu verursachen, da dies in einem fehlgeschlagenen Kanal unbestätigt ist. Ein unbestätigter Stapel auf einem fehlgeschlagenen Kanal ist wahrscheinlicher, wenn die Stapelverarbeitung länger zur Füllung angegeben wird. Wenn der Nachrichtenverkehr entlang des Kanals sporadisch mit langen Zeiträumen zwischen den Bursts von Nachrichten ist, ist eine fehlgeschlagene Stapelverarbeitung wahrscheinlicher.
- Es tritt ein Problem auf, wenn die Clustersenderseite eines Kanals fehlschlägt und dann versucht wird, einen Neustart zu starten, bevor das Überwachungssignal oder die "keep alive" den Fehler erkannt hat. Der Kanalsenderneustart wird zurückgewiesen, wenn das Ende des Clusterempfängers des Kanals aktiv geblieben ist. Um den Fehler zu vermeiden, müssen Sie den Clusterempfängerkanal beenden und erneut starten, wenn ein Clustersenderkanal versucht, einen Neustart durchzuführen.

### Auf anderen Plattformen als z/OS

Steuern Sie das Problem der Clusterempfängerseite des verbleibenden aktiven Kanals mithilfe der Attribute `AdoptNewMCA`, `AdoptNewMCATimeout` und `AdoptNewMCACheck` in der Datei `qm.ini` oder in der Windows NT -Registry.

## Nachrichten an und von Clustern weiterleiten

Verwenden Sie Warteschlangenaliasnamen, WS-Manager-Aliasnamen und Definitionen ferner Warteschlangen, um Cluster mit externen Warteschlangenmanagern und anderen Clustern zu verbinden.

Weitere Informationen zum Weiterleiten von Nachrichten an und von Clustern finden Sie in den folgenden Unterabschnitten:

### Zugehörige Konzepte

#### Cluster

#### Wie funktionieren Cluster?

„Vergleich von Clustering und verteilter Steuerung von Warteschlangen“ auf Seite 174

Vergleichen Sie die Komponenten, die für die Verbindung von WS-Managern mit verteilter Steuerung von Warteschlangen und Clustering definiert werden müssen.

„Komponenten eines Clusters“ auf Seite 176

Cluster bestehen aus Warteschlangenmanagern, Clusterrepositorys, Clusterkanälen und Clusterwarteschlangen.

„IBM WebSphere MQ -Cluster verwalten“ auf Seite 200

Sie können IBM WebSphere MQ -Cluster erstellen, erweitern und verwalten.

### „WS-Manager-Aliasnamen und-Cluster“ auf Seite 280

Verwenden Sie WS-Manager-Aliasnamen, um den Namen von Warteschlangenmanagern zu verdecken, wenn Nachrichten an einen Cluster gesendet oder aus einem Cluster gesendet werden, und die Nachrichten, die an einen Cluster gesendet werden, in Lastausgleichsnachrichten gesendet werden.

### „Warteschlangenaliasnamen und -cluster“ auf Seite 283

Verwenden Sie Warteschlangenaliasnamen, um den Namen einer Clusterwarteschlange zu verdecken, eine Warteschlange zu einem Cluster zu machen, andere Attribute zu übernehmen oder andere Zugriffsteuerungen zu übernehmen.

### „Aliasnamen für Antwortwarteschlangen und Cluster“ auf Seite 283

Eine Aliasdefinition für die Warteschlange für Antwortwarteschlangen wird verwendet, um alternative Namen für Antwortinformationen anzugeben. Definitionen von Warteschlangen für Antwortwarteschlangen können mit Clustern verwendet werden, die in einer verteilten Warteschlangenumgebung identisch sind.

## **Zugehörige Tasks**

### „WS-Manager-Cluster konfigurieren“ auf Seite 171

Über die Links in diesem Abschnitt erfahren Sie, wie Cluster funktionieren, wie eine Clusterkonfiguration entworfen wird und wie ein einfacher Cluster eingerichtet wird.

### „Neuen Cluster einrichten“ auf Seite 201

Führen Sie die folgenden Anweisungen aus, um den Beispielcluster zu konfigurieren. In separaten Anweisungen wird beschrieben, wie der Cluster auf TCP/IP, LU 6.2 und mit einer einzelnen Übertragungswarteschlange oder mehreren Übertragungswarteschlangen eingerichtet wird. Testen Sie den Cluster, indem Sie eine Nachricht von einem WS-Manager an den anderen Warteschlangenmanager senden.

## **Anforderung/Antwort in einem Cluster konfigurieren**

Konfigurieren Sie einen Anforderungs-/Antwortnachrichtenpfad von einem WS-Manager außerhalb eines Clusters. Verdecken Sie die inneren Details des Clusters, indem Sie einen Gateway-WS-Manager als Kommunikationspfad zu und vom Cluster verwenden.

## **Vorbereitende Schritte**

Abbildung 43 auf Seite 270 zeigt einen Warteschlangenmanager mit dem Namen QM3, der sich außerhalb des Clusters mit dem Namen DEMO befindet. QM3 kann ein Warteschlangenmanager auf einem WebSphere MQ -Produkt sein, das keine Cluster unterstützt. QM3 enthält eine Warteschlange namens Q3, die wie folgt definiert ist:

```
DEFINE QLOCAL(Q3)
```

Im Cluster befinden sich zwei WS-Manager, die QM1 und QM2 genannt werden. QM2 enthält eine Clusterwarteschlange mit dem Namen Q2, die wie folgt definiert ist:

```
DEFINE QLOCAL(Q2) CLUSTER(DEMO)
```

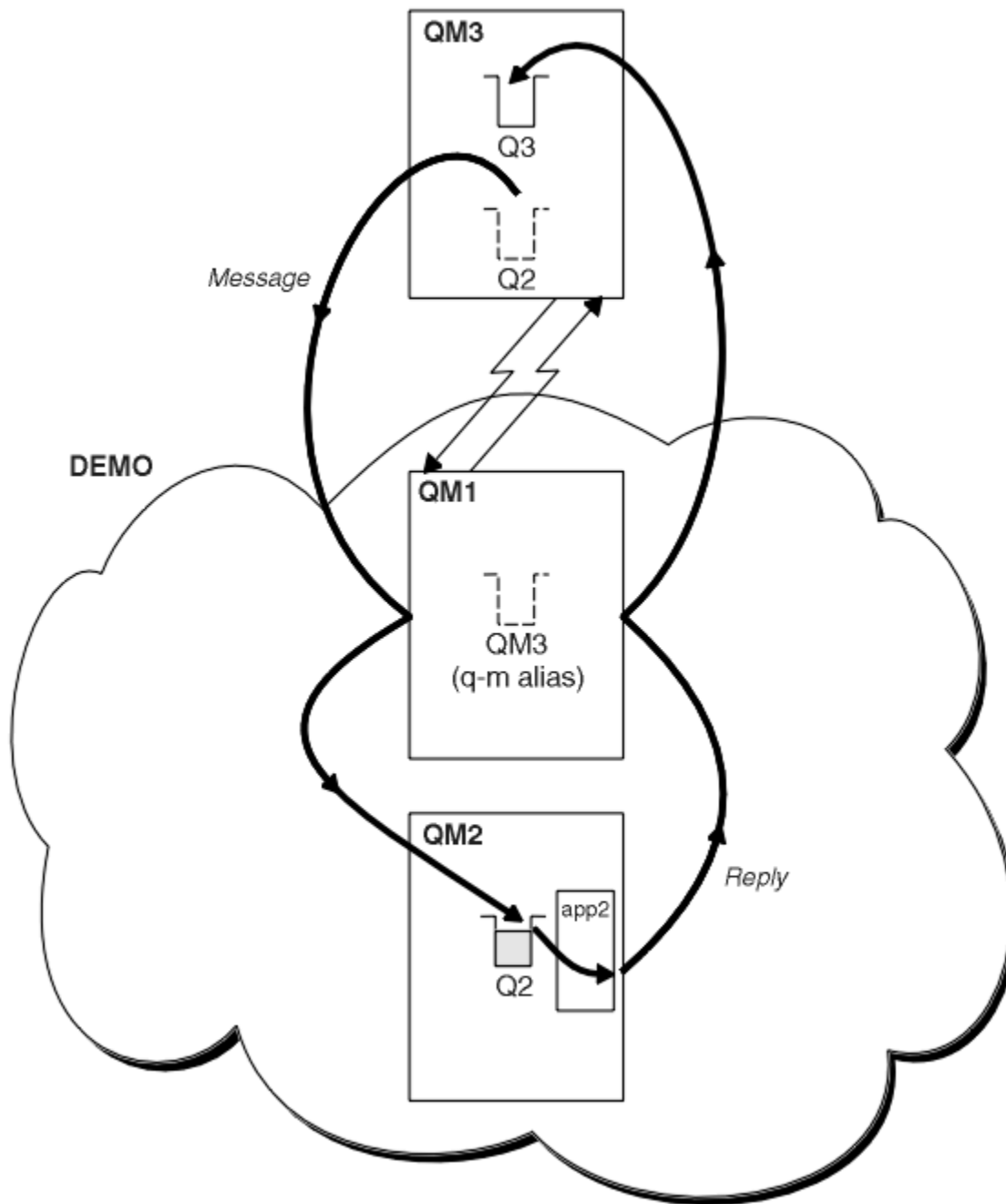


Abbildung 43. Aus einem WS-Manager außerhalb des Clusters einschalten

### Informationen zu diesem Vorgang

Befolgen Sie die Anweisungen in der Prozedur, um den Pfad für die Anforderungs- und Antwortnachrichten zu konfigurieren.

### Vorgehensweise

1. Senden Sie die Anforderungsnachricht an den Cluster.

Berücksichtigen Sie, wie der Warteschlangenmanager, der sich außerhalb des Clusters befindet, eine Nachricht in die Warteschlange Q2 von QM2 eingibt, die sich innerhalb des Clusters befindet. Ein WS-Manager außerhalb des Clusters muss eine QREMOTE -Definition für jede Warteschlange in dem Cluster haben, in den Nachrichten einreicht.

- a) Definieren Sie eine ferne Warteschlange für Q2 unter QM3.

```
DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(QM2) XMITQ(QM1)
```

Da QM3 nicht Teil eines Clusters ist, muss die Kommunikation mit verteilten Warteschlangenverfahren kommunizieren. Daher muss es auch über einen Senderkanal und eine Übertragungswarteschlange zu QM1 verfügen. QM1 benötigt einen entsprechenden Empfängerkanal. Die Kanäle und Übertragungswarteschlangen werden in [Abbildung 43](#) auf Seite 270 nicht explizit angezeigt.

Im Beispiel gibt eine Anwendung in QM3 einen Aufruf MQPUT aus, um eine Nachricht in Q2 einzureihen. Die Definition QREMOTE bewirkt, dass die Nachricht über den Senderkanal, der Nachrichten aus der QM1-Übertragungswarteschlange abrufen, an Q2 an QM2 weitergeleitet wird.

## 2. Empfangen Sie die Antwortnachricht aus dem Cluster.

Verwenden Sie einen WS-Manager-Aliasnamen, um einen Rückgabepfad für Antworten auf einen Warteschlangenmanager außerhalb des Clusters zu erstellen. Das Gateway QM1 macht einen Warteschlangenmanager-Aliasnamen für den Warteschlangenmanager zugänglich, der sich außerhalb des Clusters befindet: QM3. Sie macht QM3 für die Warteschlangenmanager im Cluster zugänglich, indem sie das Clusterattribut einer Warteschlangenmanager-Aliasdefinition für QM3 hinzufügt. Eine WS-Manager-Aliasdefinition ist wie eine ferne Warteschlangendefinition, aber mit einem leeren RNAME.

### a) Definieren Sie einen WS-Manager-Aliasnamen für QM3 auf QM1.

```
DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO)
```

Wir müssen die Auswahl des Namens für die Übertragungswarteschlange berücksichtigen, die zum Weiterleiten von Antworten von QM1 an QM3 verwendet wird. Implizit in der QREMOTE-Definition ist durch die Auslassung des Attributs XMITQ der Name der Übertragungswarteschlange QM3. QM3 ist jedoch derselbe Name, der für den Rest des Clusters unter Verwendung des Warteschlangenmanager-Aliasnamens zugänglich gemacht werden soll. WebSphere MQ ermöglicht es Ihnen nicht, sowohl der Übertragungswarteschlange als auch dem Aliasnamen des Warteschlangenmanagers denselben Namen zu geben. Eine Lösung besteht darin, eine Übertragungswarteschlange zu erstellen, mit der Nachrichten an QM3 mit einem anderen Namen an den WS-Manager-Aliasnamen weitergeleitet werden können.

### b) Geben Sie den Namen der Übertragungswarteschlange in der QREMOTE-Definition an.

```
DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO) XMITQ(QM3.XMIT)
```

Der neue Aliasname des Warteschlangenmanagers verbindet die neue Übertragungswarteschlange QM3.XMIT mit dem Aliasnamen des QM3-Warteschlangenmanagers. Es handelt sich um eine einfache und richtige Lösung, die aber nicht völlig zufriedenstellend ist. Sie hat die Namenskonvention für Übertragungswarteschlangen verletzt, die sie mit dem Namen des Zielwarteschlangenmanagers erhalten. Gibt es alternative Lösungen, die die Namenskonvention der Übertragungswarteschlange beibehalten?

Das Problem tritt auf, weil der Anforderer standardmäßig die Übergabe von QM3 als Antwort-WS-Manager-Namen in der Anforderungsnachricht, die von QM3 gesendet wird, als Antwort anfordert. Der Server unter QM2 verwendet den Namen des QM3-Antwortwarteschlangenmanagers, um QM3 in seinen Antworten zu adressieren. Die Lösung, die QM1 erforderlich ist, um QM3 als WS-Manager-Aliasnamen zugänglich zu machen, um Antwortnachrichten zurückzugeben, und verhindert, dass QM1 QM3 als Namen der Übertragungswarteschlange verwendet.

Statt standardmäßig QM3 als Antwort-WS-Managernamen zu verwenden, müssen Anwendungen in QM3 einen WS-Manager-Aliasnamen für Antwortnachrichten an QM1 übergeben. Der Gateway-Warteschlangenmanager QM1 macht den Warteschlangenmanager-Aliasnamen für Antworten an QM3 und nicht QM3 selbst zugänglich, wodurch der Konflikt mit dem Namen der Übertragungswarteschlange vermieden wird.

### c) Definieren Sie einen WS-Manager-Aliasnamen für QM3 auf QM1.

```
DEFINE QREMOTE(QM3.ALIAS) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO)
```

Es sind zwei Änderungen an den Konfigurationsbefehlen erforderlich.

- i) Die QREMOTE unter QM1 macht jetzt den Aliasnamen des Warteschlangenmanagers QM3 . ALIAS für den Rest des Cluster zugänglich und stellt die Verbindung zum Namen des realen Warteschlangenmanagers QM3her. QM3 ist wiederum der Name der Übertragungswarteschlange, an die Antwortwarteschlangen zurück an QM3 gesendet werden
- ii) Die Clientanwendung muss QM3 . ALIAS als Namen für den Antwortwarteschlangenmanager bereitstellen, wenn er die Anforderungsnachricht erstellt. Sie können QM3 . ALIAS für die Clientanwendung auf eine von zwei Arten zur Verfügung stellen.
  - Code QM3 . ALIAS im Feld für den Namen des Antwortwarteschlangenmanagers, das von MQPUT in MQMDDerstellt wird. Wenn Sie eine dynamische Warteschlange für Antworten verwenden, müssen Sie diese Vorgehensweise auf diese Weise ausführen.
  - Verwenden Sie bei der Bereitstellung des Namens der Warteschlange für die Antwortwarteschlange einen Antwortwarteschlangenalias ( Q3 . ALIAS) anstelle einer Warteschlange für Antwortwarteschlangen.

```
DEFINE QREMOTE(Q3.ALIAS) RNAME(Q3) RQMNAME(QM3.ALIAS)
```

## Nächste Schritte

**Anmerkung:** Sie können die Verwendung von Aliasnamen für Empfangswarteschlangen für Antworten mit **AMQSREQ00** nicht veranschaulichen. Sie öffnet die Warteschlange für Antwortwarteschlangen unter Verwendung des Warteschlangennamens, der in Parameter 3 angegeben ist, oder die Standardmodellwarteschlange für SYSTEM . SAMPLE . REPLY . Sie müssen das Beispiel ändern, indem Sie einen weiteren Parameter angeben, der den Aliasnamen der Empfangswarteschlange für Antworten enthält, um den Aliasnamen des Antwortwarteschlangenmanagers für MQPUTzu benennen.

### Zugehörige Tasks

„Namen eines Cluster-Ziel-WS-Managers ausblenden“ auf Seite 272

Verlegen Sie eine Nachricht an eine Clusterwarteschlange, die in einem beliebigen WS-Manager in einem Cluster definiert ist, ohne den Warteschlangenmanager zu benennen.

### **Namen eines Cluster-Ziel-WS-Managers ausblenden**

Verlegen Sie eine Nachricht an eine Clusterwarteschlange, die in einem beliebigen WS-Manager in einem Cluster definiert ist, ohne den Warteschlangenmanager zu benennen.

## Vorbereitende Schritte

- Vermeiden Sie es, die Namen von Warteschlangenmanagern zu enthüllen, die sich innerhalb des Clusters an Warteschlangenmanager befinden, die sich außerhalb des Clusters befinden.
  - Durch das Auflösen von Referenzen auf den Warteschlangenmanager, der als Host für eine Warteschlange im Cluster fungiert, wird die Flexibilität für den Lastausgleich aufgehoben.
  - Außerdem ist es für Sie schwierig, einen Warteschlangenmanager zu ändern, der als Host für eine Warteschlange im Cluster fungiert.
  - Alternativ können Sie RQMNAME durch einen WS-Manager-Aliasnamen ersetzen, der vom Clusteradministrator bereitgestellt wird.
  - „Namen eines Cluster-Ziel-WS-Managers ausblenden“ auf Seite 272 beschreibt die Verwendung eines WS-Manager-Aliasnamens, um einen WS-Manager außerhalb eines Clusters von der Verwaltung von Warteschlangenmanagern in einem Cluster zu entkoppeln.
- Der vorgeschlagene Weg zum Namen von Übertragungswarteschlangen besteht jedoch darin, ihnen den Namen des Zielwarteschlangenmanagers zu geben. Der Name der Übertragungswarteschlange zeigt den Namen eines Warteschlangenmanagers im Cluster an. Sie müssen auswählen, welche Regel folgen soll. Sie können die Übertragungswarteschlange entweder mit dem Namen des WS-Managers oder mit dem Clusternamen benennen:



### **Benennen Sie die Übertragungswarteschlange mit dem Namen des Gateway-Warteschlangenmanagers.**

Die Offenlegung des Namens des Gateway-WS-Managers an WS-Manager außerhalb eines Clusters ist eine sinnvolle Ausnahme von der Regel, die Namen von Clusterwarteschlangenmanagern zu verdecken.

### **Benennen Sie die Übertragungswarteschlange mit dem Namen des Clusters.**

Wenn Sie die Konvention für die Benennung von Übertragungswarteschlangen mit dem Namen des Zielwarteschlangenmanagers nicht befolgen, verwenden Sie den Clusternamen.

## **Informationen zu diesem Vorgang**

Ändern Sie die Task „[Anforderung/Antwort in einem Cluster konfigurieren](#)“ auf Seite 269, um den Namen des Zielwarteschlangenmanagers innerhalb des Clusters zu verdecken.

## **Vorgehensweise**

Definieren Sie im Beispiel [Abbildung 44](#) auf Seite 274 einen Warteschlangenmanager-Aliasnamen auf dem Gateway-Warteschlangenmanager QM1 mit dem Namen DEMO:

```
DEFINE QREMOTE(DEMO) RNAME(' ') RQMNAME(' ')
```

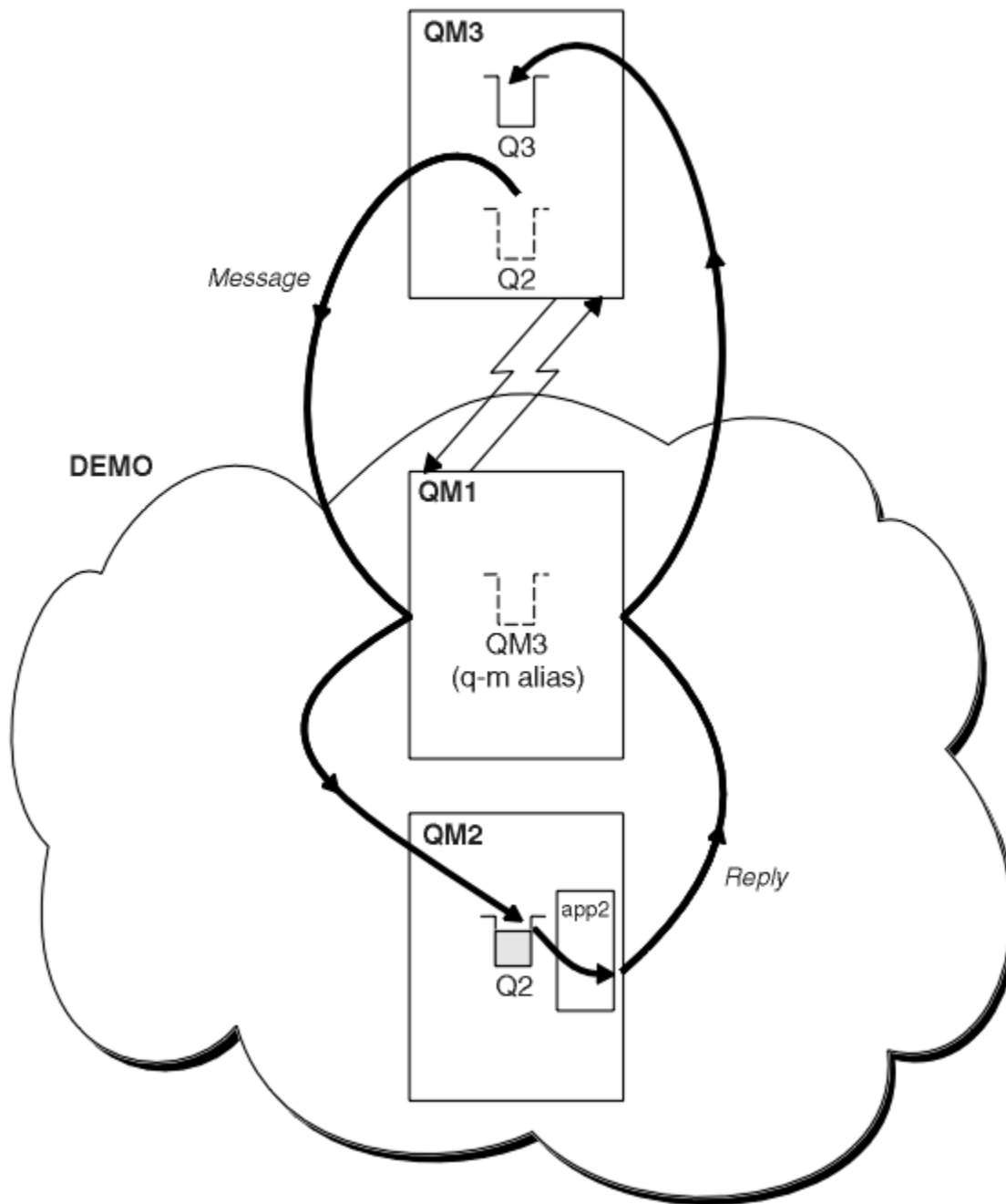


Abbildung 44. Aus einem WS-Manager außerhalb des Clusters einschalten

Die QREMOTE -Definition unter QM1 macht den Aliasnamen des Warteschlangenmanagers DEMO dem Gateway-Warteschlangenmanager bekannt. QM3 kann der Warteschlangenmanager außerhalb des Clusters den Warteschlangenmanager-Aliasnamen DEMO verwenden, um Nachrichten an Clusterwarteschlangen unter DEMO zu senden, anstatt einen tatsächlichen Warteschlangenmanagernamen verwenden zu müssen.

Wenn Sie die Konvention verwenden, den Clusternamen zu verwenden, um die Übertragungswarteschlange zu benennen, die eine Verbindung zu einem Cluster herstellt, wird die Definition der fernen Warteschlange für Q2 wie folgt:

```
DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(DEMO)
```

## Ergebnisse

Nachrichten, die für Q2 auf DEMO bestimmt sind, werden in die Übertragungswarteschlange von DEMO gestellt. Aus der Übertragungswarteschlange werden sie vom senderkanal an den Gateway-Warteschlangenmanager QM1 übertragen. Der Gateway-Warteschlangenmanager leitet die Nachrichten an jeden Warteschlangenmanager im Cluster weiter, in dem sich die Clusterwarteschlange Q2 befindet.

## Request/Antwort von einem Cluster konfigurieren

Konfigurieren Sie einen Anforderungs-/Antwortnachrichtenpfad von einem Cluster zu einem WS-Manager außerhalb des Clusters. Verdecken Sie die Details dazu, wie ein Warteschlangenmanager innerhalb des Clusters über einen Gateway-Warteschlangenmanager außerhalb des Clusters kommuniziert.

## Vorbereitende Schritte

Abbildung 45 auf Seite 276 zeigt einen Warteschlangenmanager ( QM2) innerhalb des Clusters DEMO. Er sendet eine Anforderung an eine Warteschlange Q3, die sich auf dem WS-Manager außerhalb des Clusters befindet. Die Antworten werden an Q2 im QM2 innerhalb des Clusters zurückgegeben.

Für die Kommunikation mit dem WS-Manager außerhalb des Clusters agieren mindestens ein Warteschlangenmanager im Cluster als Gateway. Ein Gateway-WS-Manager hat einen Kommunikationspfad zu den Warteschlangenmanagern außerhalb des Clusters. In dem Beispiel ist QM1 das Gateway.

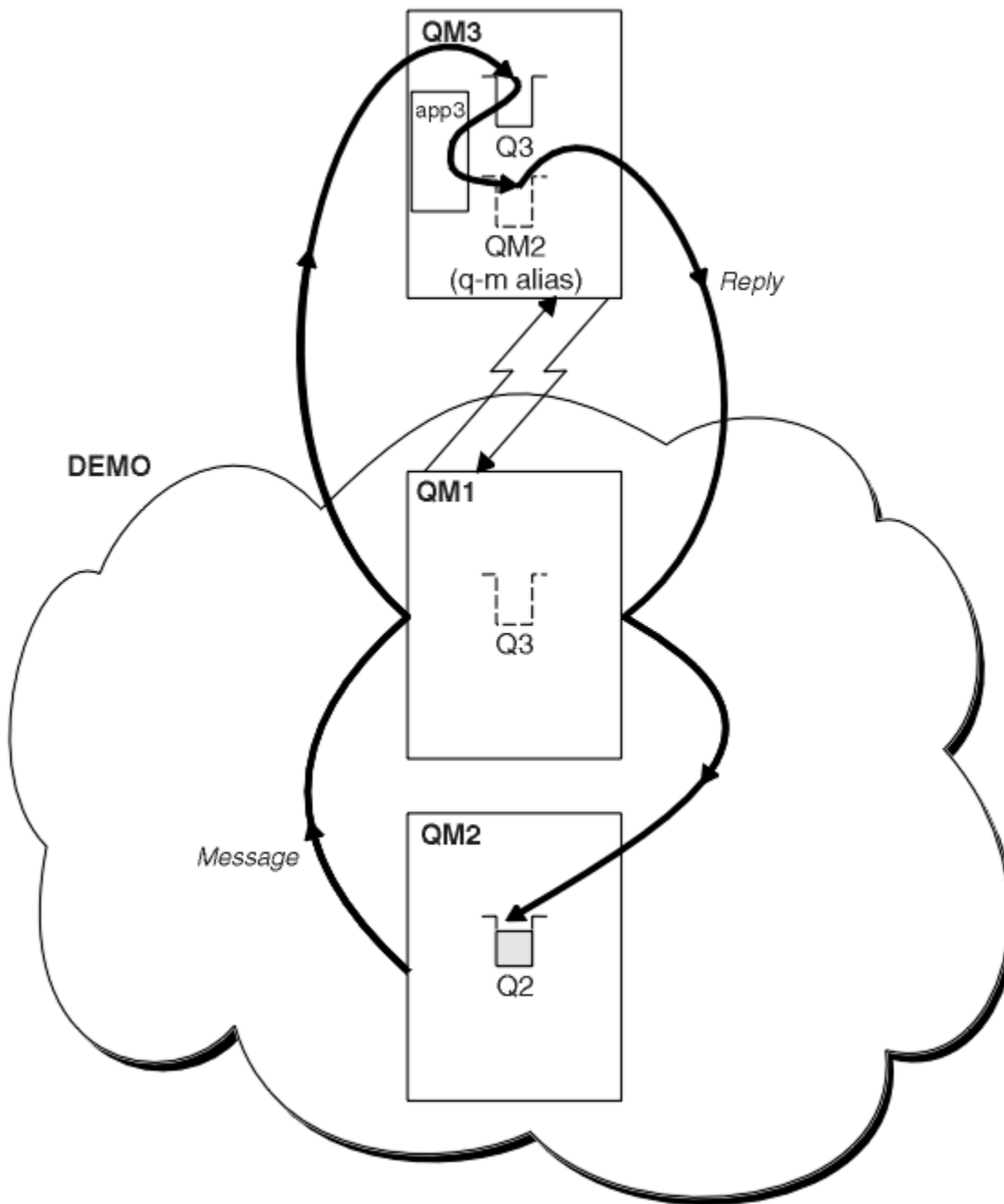


Abbildung 45. In einen WS-Manager außerhalb des Clusters einschalten

### Informationen zu diesem Vorgang

Befolgen Sie die Anweisungen zum Festlegen des Pfads für die Anforderungs- und Antwortnachrichten.

### Vorgehensweise

1. Senden Sie die Anforderungsnachricht aus dem Cluster.

Überlegen Sie, wie der Warteschlangenmanager QM2, der sich im Cluster befindet, eine Nachricht in die Warteschlange Q3 von QM3 einreicht, die sich außerhalb des Clusters befindet.

- a) Erstellen Sie eine QREMOTE -Definition in QM1, die die ferne Warteschlange Q3 für den Cluster zugänglich macht.

```
DEFINE QREMOTE(Q3) RNAME(Q3) RQMNAME(QM3) CLUSTER(DEMO)
```

Sie verfügt außerdem über einen Senderkanal und eine Übertragungswarteschlange zum Warteschlangenmanager, der sich außerhalb des Clusters befindet. QM3 verfügt über einen entsprechenden Empfängerkanal. Die Kanäle werden in [Abbildung 45 auf Seite 276](#) nicht angezeigt.

Eine Anwendung unter QM2 gibt einen MQPUT -Aufruf aus, der die Zielwarteschlange und die Warteschlange angibt, an die Antworten gesendet werden sollen. Die Zielwarteschlange ist Q3 und die Warteschlange für Antwortantworten ist Q2.

Die Nachricht wird an QM1 gesendet, die die Definition der fernen Warteschlange verwendet, um den Warteschlangennamen in Q3 auf QM3 aufzulösen.

2. Empfangen Sie die Antwortnachricht vom WS-Manager außerhalb des Clusters.

Ein WS-Manager außerhalb des Clusters muss für jeden Warteschlangenmanager im Cluster, an den er eine Nachricht sendet, über einen Warteschlangenmanager-Aliasnamen verfügen. Der Aliasname des WS-Managers muss auch den Namen der Übertragungswarteschlange für den Gateway-Warteschlangenmanager angeben. In diesem Beispiel benötigt QM3 eine WS-Manager-Aliasdefinition für QM2:

- a) Erstellen Sie einen WS-Manager-Aliasnamen QM2 unter QM3.

```
DEFINE QREMOTE(QM2) RNAME(' ') RQMNAME(QM2) XMITQ(QM1)
```

QM3 benötigt außerdem eine Sende-Channel-Warteschlange und eine Übertragungswarteschlange für QM1 und QM1 benötigt einen entsprechenden Empfängerkanal.

Die Anwendung **app3** unter QM3 kann dann Antworten an QM2 senden, indem sie einen MQPUT -Aufruf ausgibt und den Warteschlangennamen Q2 und den Namen des Warteschlangenmanagers QM2 angibt.

## Nächste Schritte

Sie können mehr als eine Route aus einem Cluster definieren.

## Lastausgleich von außerhalb eines Clusters konfigurieren

Konfigurieren Sie einen Nachrichtenpfad von einem WS-Manager außerhalb eines Clusters in eine beliebige Kopie einer Clusterwarteschlange. Das Ergebnis ist eine Auslastungsabgleichsanforderungen von außerhalb des Clusters an die einzelnen Instanzen einer Clusterwarteschlange.

## Vorbereitende Schritte

Konfigurieren Sie das Beispiel, wie in [Abbildung 43 auf Seite 270](#) in „Anforderung/Antwort in einem Cluster konfigurieren“ auf [Seite 269](#) dargestellt.

## Informationen zu diesem Vorgang

In diesem Szenario sendet der Warteschlangenmanager außerhalb des Clusters ( QM3 in [Abbildung 46 auf Seite 278](#)) Anforderungen an die Warteschlange Q2. Q2 wird auf zwei Warteschlangenmanagern im Cluster DEMO gehostet, um den Lastausgleich zu verwenden. Eine Warteschlange mit dem Namen Q2 ist auf den Warteschlangenmanagern QM2 und QM4 definiert, aber nicht auf dem Gateway-Warteschlangenmanager QM1. Die Anforderungen von QM3, dem Warteschlangenmanager außerhalb des Clusters, werden an eine der Instanzen von Q2 gesendet.

QM3 ist nicht Teil eines Clusters und kommuniziert mit verteilten Warteschlangenverfahren. Er muss über einen Senderkanal und eine Übertragungswarteschlange zu QM1 verfügen. QM1 benötigt einen entsprechenden Empfängerkanal. Die Kanäle und Übertragungswarteschlangen werden in [Abbildung 46 auf Seite 278](#) nicht explizit angezeigt.

Die Prozedur erweitert das Beispiel in [Abbildung 43 auf Seite 270](#) in „Anforderung/Antwort in einem Cluster konfigurieren“ auf [Seite 269](#).

## Vorgehensweise

1. Definieren Sie eine lokale Warteschlange mit dem Namen Q2 in den einzelnen QM2 und QM4.

```
DEFINE QLOCAL(Q2) CLUSTER(DEMO) DEFBIND(NOTFIXED)
```

- Erstellen Sie eine QREMOTE -Definition für Q2 auf QM3.

```
DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(Q3) XMITQ(QM1)
```

Erstellen Sie eine QREMOTE -Definition für jede Warteschlange in dem Cluster, in der QM3 Nachrichten einreicht.

- Erstellen Sie unter QM3 einen Warteschlangenmanager-Aliasnamen Q3 .

```
DEFINE QREMOTE(Q3) RNAME(' ') RQMNAME(' ') CLUSTER(DEMO) DEFBIND(NOTFIXED)
```

Q3 ist kein echter WS-Manager-Name. Dies ist der Name einer WS-Manager-Aliasdefinition im Cluster, die den Aliasnamen des Warteschlangenmanagers Q3 mit Leerzeichen, ' ', entspricht.

- QM1, der Gateway-Warteschlangenmanager, enthält keine speziellen Definitionen.

## Ergebnisse

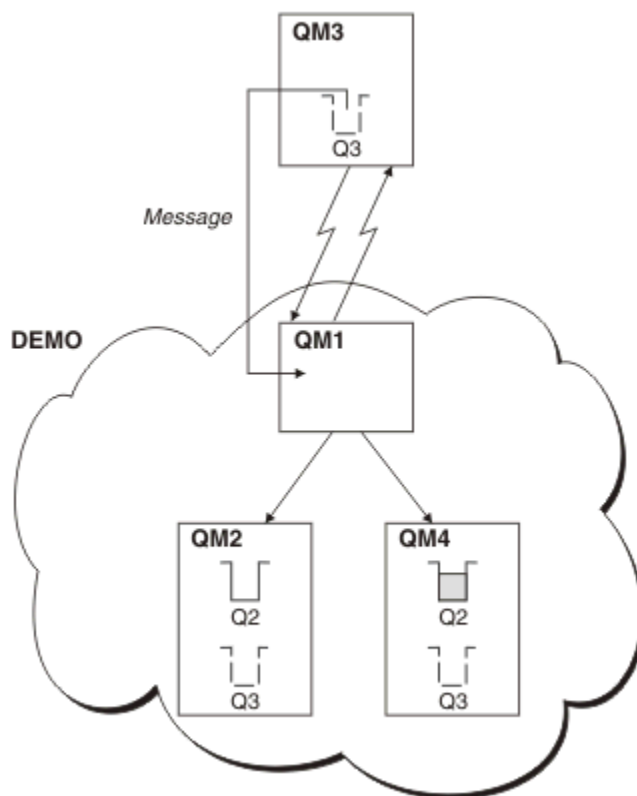


Abbildung 46. Aus einem WS-Manager außerhalb des Clusters einschalten

Wenn eine Anwendung unter QM3 einen MQPUT -Aufruf ausgibt, um eine Nachricht in Q2 einzureihen, führt die QREMOTE -Definition dazu, dass die Nachricht über den Gateway-Warteschlangenmanager QM1 weitergeleitet wird. QM1 verwendet den Lastenausgleich, um Nachrichten, die an Q2 gerichtet sind, zwischen den Warteschlangen mit dem Namen Q2 auf den beiden Warteschlangenmanagern QM2 und QM4 zu verteilen, die Aliasnamen für Clusterwarteschlangenmanager für Q3 haben.

## Nachrichtenpfade zwischen Clustern konfigurieren

Verbinden Sie Cluster unter Verwendung eines Gateway-Warteschlangenmanagers. Stellen Sie Warteschlangen oder Warteschlangenmanager für alle Cluster sichtbar, indem Sie Clusterwarteschlangen- oder Cluster-WS-Manager-Aliasnamen auf dem Gateway-Warteschlangenmanager definieren.

## Informationen zu diesem Vorgang

Anstatt alle Warteschlangenmanager in einem großen Cluster zu gruppieren, können Sie viele kleinere Cluster haben. Jeder Cluster verfügt über einen oder mehrere Warteschlangenmanager, die als Brücke fungieren. Dies hat den Vorteil, dass Sie die Sichtbarkeit von Warteschlangen- und WS-Managernamen in den Clustern einschränken können. Siehe „Überlappende Cluster“ auf Seite 195. Verwenden Sie Aliasnamen, um die Namen von Warteschlangen und Warteschlangenmanagern zu ändern, um Namensunverträglichkeiten zu vermeiden oder um die lokalen Namenskonventionen einzuhalten.

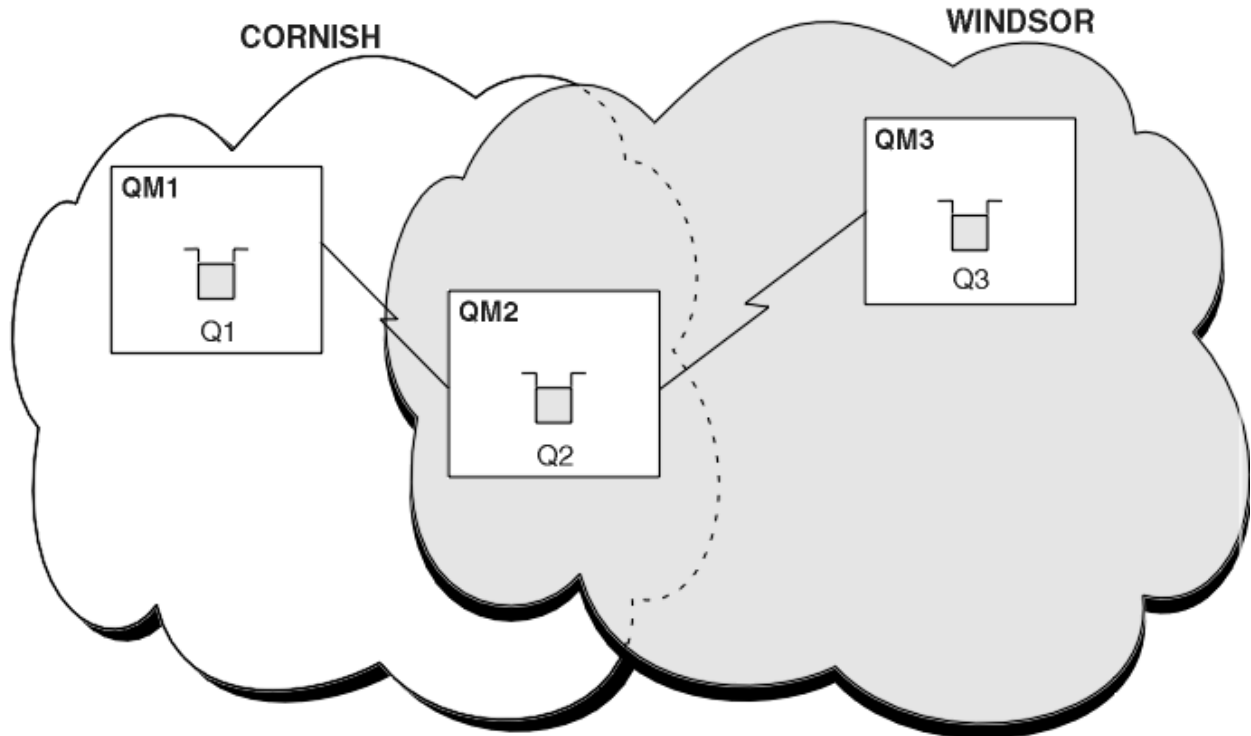


Abbildung 47. Clusterübergreifende Überbrückung

Abbildung 47 auf Seite 279 zeigt zwei Cluster mit einer Brücke zwischen ihnen. Es könnte mehr als eine Brücke vorhanden sein.

Konfigurieren Sie die Cluster mit der folgenden Prozedur:

### Vorgehensweise

1. Definieren Sie eine Clusterwarteschlange Q1 auf QM1.

```
DEFINE QLOCAL(Q1) CLUSTER(CORNISH)
```

2. Definieren Sie eine Clusterwarteschlange Q3 auf QM3.

```
DEFINE QLOCAL(Q3) CLUSTER(WINDSOR)
```

3. Erstellen Sie eine Namensliste mit dem Namen CORNISHWINDSOR unter QM2, die die Namen beider Cluster enthält.

```
DEFINE NAMELIST(CORNISHWINDSOR) DESCR('CornishWindsor namelist')
 NAMES(CORNISH, WINDSOR)
```

4. Definieren Sie eine Clusterwarteschlange, Q2 auf QM2

```
DEFINE QLOCAL(Q2) CLUSNL(CORNISHWINDSOR)
```

## Nächste Schritte

QM2 ist ein Member der beiden Cluster und ist die Brücke zwischen den beiden Clustern. Für jede Warteschlange, die Sie über die Brücke hinweg sichtbar machen möchten, benötigen Sie eine QALIAS-Definition auf der Brücke. In [Abbildung 47 auf Seite 279](#) unter QM2 benötigen Sie beispielsweise Folgendes:

```
DEFINE QALIAS(MYQ3) TARGET(Q3) CLUSTER(CORNISH) DEFBIND(NOTFIXED)
```

Mithilfe des Warteschlangenalias kann eine Anwendung, die mit einem Warteschlangenmanager in CORNISH verbunden ist (z. B. QM1), eine Nachricht an Q3 senden. Er bezieht sich auf Q3 als MYQ3. Die Nachricht wird an Q3 unter QM3 weitergeleitet.

Wenn Sie eine Warteschlange öffnen, müssen Sie DEFBIND entweder auf NOTFIXED oder auf QDEF setzen. Wenn DEFBIND als Standardwert (OPEN) belassen wird, löst der Warteschlangenmanager die Aliasdefinition in den Brückenwarteschlangenmanager auf, in dem sie enthalten ist. Die Brücke leitet die Nachricht nicht weiter.

Für jeden Warteschlangenmanager, den Sie sichtbar machen möchten, benötigen Sie eine WS-Manager-Aliasnamendefinition. Unter QM2 benötigen Sie z. B.:

```
DEFINE QREMOTE(QM1) RNAME(' ') RQMNAME(QM1) CLUSTER(WINDSOR)
```

Eine Anwendung, die mit einem beliebigen WS-Manager in WINDSOR verbunden ist, z. B. QM3, kann eine Nachricht in jede Warteschlange unter QM1 stellen, indem sie QM1 explizit im Aufruf MQOPEN benennt.

## WS-Manager-Aliasnamen und-Cluster

Verwenden Sie WS-Manager-Aliasnamen, um den Namen von Warteschlangenmanagern zu verdecken, wenn Nachrichten an einen Cluster gesendet oder aus einem Cluster gesendet werden, und die Nachrichten, die an einen Cluster gesendet werden, in Lastausgleichsnachrichten gesendet werden.

WS-Manager-Aliasnamen, die unter Verwendung einer Definition einer fernen Warteschlange mit einem leeren RNAME erstellt werden, haben fünf Verwendungen:

### Namen des WS-Managers beim Senden von Nachrichten neu zuordnen

Ein Warteschlangenmanager-Aliasname kann verwendet werden, um den in einem MQOPEN -Aufruf angegebenen Warteschlangenmanagernamen einem anderen Warteschlangenmanager neu zuzuordnen. Es kann sich um einen Cluster-WS-Manager handeln. Beispielsweise kann ein Warteschlangenmanager die Definition des WS-Manager-Aliasnamens haben:

```
DEFINE QREMOTE(YORK) RNAME(' ') RQMNAME(CLUSQM)
```

YORK kann als Aliasname für den WS-Manager CLUSQM verwendet werden. Wenn eine Anwendung auf dem Warteschlangenmanager, die diese Definition vorgenommen hat, eine Nachricht an den Warteschlangenmanager YORK stellt, löst der lokale WS-Manager den Namen in CLUSQM auf. Wenn der lokale WS-Manager nicht CLUSQM genannt wird, wird die Nachricht in die Clusterübertragungswarteschlange gestellt, die in CLUSQM verschoben werden soll. Außerdem ändert er den Übertragungsheader, um CLUSQM anstelle von YORK zu verwenden.

**Anmerkung:** Die Definition gilt nur für den Warteschlangenmanager, der die Definition des Warteschlangenmanagers vornimmt. Um den Aliasnamen für den gesamten Cluster zugänglich zu machen, müssen Sie das Attribut CLUSTER zur Definition der fernen Warteschlange hinzufügen. Anschließend werden Nachrichten von anderen Warteschlangenmanagern, die für YORK bestimmt waren, an CLUSQM gesendet.

### Ändern oder Angeben der Übertragungswarteschlange beim Senden von Nachrichten

Das Aliasing kann verwendet werden, um einen Cluster in ein Nicht-Clustersystem zu verknüpfen. Beispielsweise könnten Warteschlangenmanager im Cluster ITALY mit dem Warteschlangenmanager PALERMO kommunizieren, der sich außerhalb des Clusters befindet. Für die Kommunikation muss



einer der WS-Manager im Cluster als Gateway fungieren. Geben Sie im Gateway-WS-Manager den folgenden Befehl aus:

```
DEFINE QREMOTE(ROME) RNAME(' ') RQMNAME(PALERMO) XMITQ(X) CLUSTER(ITALY)
```

Der Befehl ist eine WS-Manager-Aliasnamendefinition. Sie definiert und macht ROME als Warteschlangenmanager zugänglich, über den Nachrichten von jedem Warteschlangenmanager im Cluster ITALY mehrere Hops ausführen können, um ihr Ziel bei PALERMO zu erreichen. Nachrichten, die in eine Warteschlange gestellt werden, die mit dem Namen des Warteschlangenmanagers, der auf ROME gesetzt ist, geöffnet wurde, werden mit der Aliasdefinition des Warteschlangenmanagers an den Gateway-Warteschlangenmanager gesendet. Dort werden die Nachrichten in die Übertragungswarteschlange X eingereicht und von Nicht-Clusterkanälen in den Warteschlangenmanager PALERMO verschoben.

Die Auswahl des Namens ROME in diesem Beispiel ist nicht signifikant. Die Werte für QREMOTE und RQMNAME können identisch sein.

### **Bestimmung des Ziels beim Empfangen von Nachrichten**

Wenn ein WS-Manager eine Nachricht empfängt, extrahiert er den Namen der Zielwarteschlange und des Warteschlangenmanagers aus dem Übertragungsheader. Sie sucht nach einer WS-Manager-Aliasnamendefinition mit demselben Namen wie der Warteschlangenmanager im Übertragungsheader. Wenn er einen findet, ersetzt er den RQMNAME aus der WS-Manager-Aliasnamendefinition für den Namen des Warteschlangenmanagers im Übertragungsheader.

Es gibt zwei Gründe für die Verwendung eines WS-Manager-Aliasnamens auf diese Weise:

- Nachrichten an einen anderen WS-Manager zu leiten
- Um den Namen des Warteschlangenmanagers zu ändern, der mit dem lokalen WS-Manager identisch sein soll

### **WS-Manager-Aliasnamen in einem Gateway-WS-Manager verwenden, um Nachrichten zwischen Warteschlangenmanagern in verschiedenen Clustern weiterzuleiten.**

Eine Anwendung kann mithilfe eines WS-Manager-Aliasnamens eine Nachricht an eine Warteschlange in einem anderen Cluster senden. Die Warteschlange muss keine Clusterwarteschlange sein. Die Warteschlange ist in einem Cluster definiert. Die Anwendung ist mit einem WS-Manager in einem anderen Cluster verbunden. Ein Gateway-WS-Manager verbindet die beiden Cluster. Wenn die Warteschlange nicht als Cluster-Cluster definiert ist, muss die Anwendung die Warteschlange unter Verwendung des Warteschlangennamens und des Aliasnamens eines Cluster-WS-Managers öffnen, damit die korrekte Weiterleitung erfolgt. Ein Beispiel für eine Konfiguration finden Sie in „[Erstellen von zwei überlappenden Clustern mit einem Gateway-Warteschlangenmanager](#)“ auf Seite 235, von der aus der in [Abbildung 1](#) dargestellte Antwortnachrichtenfluss ausgeführt wird.

Das Diagramm zeigt den Pfad, den die Antwortnachricht zurück zu einer temporären dynamischen Warteschlange mit dem Namen RQ nimmt. Die Serveranwendung, die mit QM3 verbunden ist, öffnet die Antwortwarteschlange unter Verwendung des Warteschlangenmanagernamens QM2. Der Warteschlangenmanagername QM2 ist unter QM1 als Aliasname eines Clusterwarteschlangenmanagers definiert. QM3 leitet die Antwortnachricht an QM1 weiter. QM1 leitet die Nachricht an QM2 weiter.



## Die Verwendung eines Warteschlangenmanagers als Gateway in den Cluster für die Lastverteilung von Nachrichten von außerhalb des Clusters aus.

Sie definieren eine Warteschlange mit dem Namen EDINBURGH in mehr als einem Warteschlangenmanager im Cluster. Sie möchten, dass der Clustering-Mechanismus die Auslastung für Nachrichten, die von außerhalb des Clusters in diese Warteschlange einreisen, ausgleichen soll.

Ein Warteschlangenmanager von außerhalb des Clusters benötigt eine Übertragungswarteschlange und einen Senderkanal zu einem WS-Manager im Cluster. Diese Warteschlange wird als Gateway-WS-Manager bezeichnet. Um den standardmäßigen Lastausgleichsmechanismus nutzen zu können, muss eine der folgenden Regeln gelten:

- Der Gateway-Warteschlangenmanager darf keine Instanz der EDINBURGH -Warteschlange enthalten.
- Der Gateway-WS-Manager gibt CLWLUSEQ (ANY) auf ALTER QMGR an.

Ein Beispiel für einen Lastausgleich von außerhalb eines Clusters finden Sie in [„Lastausgleich von außerhalb eines Clusters konfigurieren“](#) auf Seite 277.

## Aliasnamen für Antwortwarteschlangen und Cluster

Eine Aliasdefinition für die Warteschlange für Antwortwarteschlangen wird verwendet, um alternative Namen für Antwortinformationen anzugeben. Definitionen von Warteschlangen für Antwortwarteschlangen können mit Clustern verwendet werden, die in einer verteilten Warteschlangenumgebung identisch sind.

Beispiel:

- Eine Anwendung im Warteschlangenmanager VENICE sendet über den Aufruf MQPUT eine Nachricht an den Warteschlangenmanager PISA . Die Anwendung stellt die folgenden Antworten auf die Antwortwarteschlange im Nachrichtendeskriptor bereit:

```
ReplyToQ=' QUEUE '
ReplyToQMGr=' '
```

- Damit die Antworten, die an QUEUE gesendet werden, unter PISA auf OTHERQ empfangen werden können, erstellen Sie eine Definition der fernen Warteschlange in VENICE , die als Aliasname für die Antwortwarteschlange verwendet wird. Der Aliasname ist nur auf dem System wirksam, auf dem es erstellt wurde.

```
DEFINE QREMOTE(Queue) RNAME(OTHERQ) RQMNAME(PISA)
```

RQMNAME und QREMOTE können dieselben Namen angeben, auch wenn RQMNAME selbst ein Cluster-WS-Manager ist.

## Warteschlangenaliasnamen und -cluster

Verwenden Sie Warteschlangenaliasnamen, um den Namen einer Clusterwarteschlange zu verdecken, eine Warteschlange zu einem Cluster zu machen, andere Attribute zu übernehmen oder andere Zugriffsteuerungen zu übernehmen.

Eine QALIAS -Definition wird verwendet, um einen Aliasnamen zu erstellen, über den eine Warteschlange bekannt sein soll. Sie können einen Aliasnamen aus einer Reihe von Gründen erstellen:

- Sie möchten mit der Verwendung einer anderen Warteschlange beginnen, aber Sie möchten Ihre Anwendungen nicht ändern.
- Sie möchten nicht, dass Anwendungen den tatsächlichen Namen der Warteschlange, in die sie Nachrichten einreihen, kennen.
- Es kann eine Namenskonvention vorhanden sein, die sich von der Namenskonvention unterscheidet, in der die Warteschlange definiert ist.
- Ihre Anwendungen sind möglicherweise nicht berechtigt, auf die Warteschlange durch ihren tatsächlichen Namen zuzugreifen, sondern nur durch ihren Aliasnamen.

Erstellen Sie mit dem Befehl `DEFINE QALIAS` eine `QALIAS` -Definition auf einem Warteschlangenmanager. Führen Sie z. B. den folgenden Befehl aus:

```
DEFINE QALIAS(PUBLIC) TARGET(LOCAL) CLUSTER(C)
```

Der Befehl wirbt für eine Warteschlange mit dem Namen `PUBLIC` für die Warteschlangenmanager im Cluster `C`. `PUBLIC` ist ein Aliasname, der in die Warteschlange `LOCAL` aufgelöst wird. Nachrichten, die an `PUBLIC` gesendet werden, werden an die Warteschlange `LOCAL` weitergeleitet.

Sie können auch eine Warteschlangenaliasdefinition verwenden, um einen Warteschlangennamen in eine Clusterwarteschlange aufzulösen. Führen Sie z. B. den folgenden Befehl aus:

```
DEFINE QALIAS(PRIVATE) TARGET(PUBLIC)
```

Mit dem Befehl kann ein Warteschlangenmanager den Namen `PRIVATE` verwenden, um auf eine Warteschlange zuzugreifen, die an anderer Stelle im Cluster durch den Namen `PUBLIC` angezeigt wird. Da diese Definition das Attribut `CLUSTER` nicht enthält, gilt sie nur für den Warteschlangenmanager, der sie macht.

## Cluster für Workload-Management verwenden

Indem Sie mehrere Instanzen einer Warteschlange auf verschiedenen Warteschlangenmanagern in einem Cluster definieren, können Sie die Arbeit der Wartung der Warteschlange auf mehrere Server verteilen. Es gibt mehrere Faktoren, die verhindern können, dass Nachrichten in einem anderen Warteschlangenmanager in den Fall eines Fehlers erneut gestellt werden.

Neben der Konfiguration von Clustern zur Reduzierung der Systemverwaltung können Sie Cluster erstellen, in denen mehr als ein Warteschlangenmanager eine Instanz derselben Warteschlange hostet.

Sie können Ihren Cluster so organisieren, dass die Warteschlangenmanager in ihm klonen einander sind. Jeder Warteschlangenmanager ist in der Lage, dieselben Anwendungen auszuführen und lokale Definitionen derselben Warteschlangen zu verwenden. Sie können die Workload zwischen Ihren Warteschlangenmanagern verteilen, indem Sie mehrere Instanzen einer Anwendung verwenden. Jede Instanz der Anwendung empfängt Nachrichten und wird unabhängig voneinander ausgeführt.

Die Verwendung von Clustern bietet folgende Vorteile:

- Höhere Verfügbarkeit Ihrer Warteschlangen und Anwendungen
- Schnellerer Durchsatz von Nachrichten
- Gleichmäßige Verteilung der Arbeitslast in Ihrem Netz

Jeder Warteschlangenmanager, der eine Instanz einer bestimmten Warteschlange enthält, kann Nachrichten verarbeiten, die für diese Warteschlange bestimmt sind. Anwendungen benennen beim Senden von Nachrichten keinen Warteschlangenmanager. Ein Workload-Management-Algorithmus legt fest, welcher Warteschlangenmanager die Nachricht verarbeitet.

Weitere Informationen zu Clusterkonfigurationen für das Workload-Management finden Sie in den folgenden Unterabschnitten:

### Zugehörige Konzepte

#### Cluster

##### Wie funktionieren Cluster?

„Vergleich von Clustering und verteilter Steuerung von Warteschlangen“ auf Seite 174

Vergleichen Sie die Komponenten, die für die Verbindung von WS-Managern mit verteilter Steuerung von Warteschlangen und Clustering definiert werden müssen.

„Komponenten eines Clusters“ auf Seite 176

Cluster bestehen aus Warteschlangenmanagern, Clusterrepositorys, Clusterkanälen und Clusterwarteschlangen.

„IBM WebSphere MQ -Cluster verwalten“ auf Seite 200

Sie können IBM WebSphere MQ -Cluster erstellen, erweitern und verwalten.

„Nachrichten an und von Clustern weiterleiten“ auf Seite 268

Verwenden Sie Warteschlangenaliasnamen, WS-Manager-Aliasnamen und Definitionen ferner Warteschlangen, um Cluster mit externen Warteschlangenmanagern und anderen Clustern zu verbinden.

### **Zugehörige Tasks**

[„WS-Manager-Cluster konfigurieren“ auf Seite 171](#)

Über die Links in diesem Abschnitt erfahren Sie, wie Cluster funktionieren, wie eine Clusterkonfiguration entworfen wird und wie ein einfacher Cluster eingerichtet wird.

[„Neuen Cluster einrichten“ auf Seite 201](#)

Führen Sie die folgenden Anweisungen aus, um den Beispielcluster zu konfigurieren. In separaten Anweisungen wird beschrieben, wie der Cluster auf TCP/IP, LU 6.2 und mit einer einzelnen Übertragungswarteschlange oder mehreren Übertragungswarteschlangen eingerichtet wird. Testen Sie den Cluster, indem Sie eine Nachricht von einem WS-Manager an den anderen Warteschlangenmanager senden.

[Exits für Clusterauslastung schreiben und kompilieren](#)

## **Beispiel für einen Cluster mit mehr als einer Instanz einer Warteschlange**

In diesem Beispiel für einen Cluster mit mehr als einer Instanz einer Warteschlange werden Nachrichten an verschiedene Instanzen der Warteschlange weitergeleitet. Sie können eine Nachricht zu einer bestimmten Instanz der Warteschlange erzwingen, und Sie können auswählen, dass eine Nachrichtenfolge an einen der Warteschlangenmanager gesendet werden soll.

Abbildung 49 auf Seite 286 zeigt einen Cluster, in dem mehr als eine Definition für die Warteschlange Q3 vorhanden ist. Wenn eine Anwendung von QM1 eine Nachricht in Q3 einreicht, weiß sie nicht unbedingt, welche Instanz von Q3 ihre Nachricht verarbeiten wird. Wenn eine Anwendung unter QM2 oder QM4, ausgeführt wird, wenn lokale Instanzen von Q3 vorhanden sind, wird die lokale Instanz von Q3 standardmäßig geöffnet. Durch die Festlegung des Warteschlangenattributs CLWLUSEQ kann die lokale Instanz der Warteschlange wie eine ferne Instanz der Warteschlange behandelt werden.

Die Option MQOPEN DefBind steuert, ob der Zielwarteschlangenmanager ausgewählt wird, wenn der Aufruf MQOPEN ausgegeben wird oder wenn die Nachricht aus der Übertragungswarteschlange übertragen wird.

Wenn Sie DefBind auf MQBND\_BIND\_NOT\_FIXED setzen, kann die Nachricht an eine Instanz der Warteschlange gesendet werden, die verfügbar ist, wenn die Nachricht übertragen wird. Dadurch werden die folgenden Probleme vermieden:

- Die Zielwarteschlange ist nicht verfügbar, wenn die Nachricht auf dem Zielwarteschlangenmanager ankommt.
- Der Status der Warteschlange wurde geändert.
- Die Nachricht wurde mit einem Aliasnamen der Clusterwarteschlange verwendet, und es ist keine Instanz der Zielwarteschlange auf dem Warteschlangenmanager vorhanden, auf dem die Instanz des Aliasnamens der Clusterwarteschlange definiert ist.

Wenn diese Probleme bei der Übertragungszeit erkannt werden, wird eine andere verfügbare Instanz der Zielwarteschlange gesucht, und die Nachricht wird erneut weitergeleitet. Wenn keine Instanzen der Warteschlange verfügbar sind, wird die Nachricht in die Warteschlange für dead-letter gestellt.

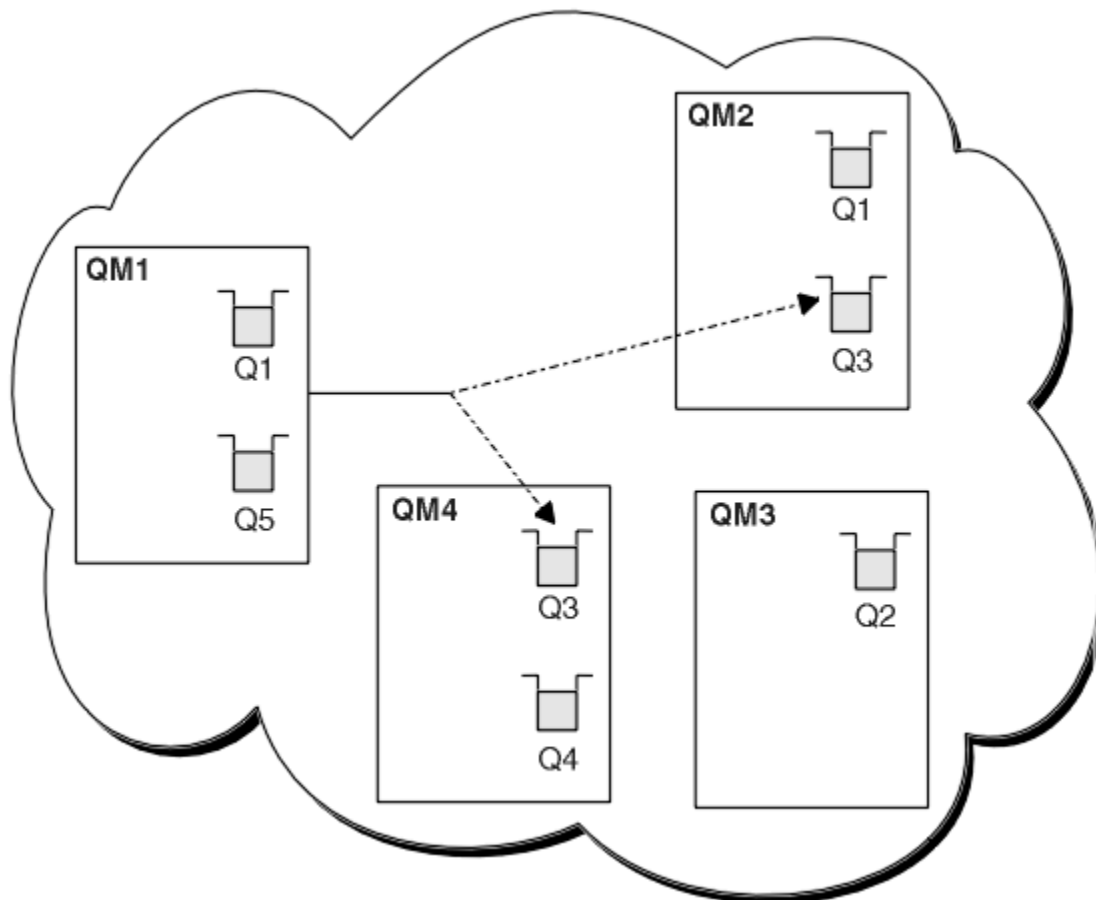


Abbildung 49. Ein Cluster mit mehreren Instanzen derselben Warteschlange

Ein Faktor, der verhindern kann, dass Nachrichten weitergeleitet werden, ist, wenn Nachrichten einem festen Warteschlangenmanager oder Kanal mit `MQBND_BIND_ON_OPEN` zugeordnet wurden. Nachrichten, die an `MQOPEN` gebunden sind, werden nie einem anderen Kanal zugeordnet. Beachten Sie auch, dass die Nachrichtenzuordnung nur dann stattfindet, wenn ein Clusterkanal tatsächlich fehlschlägt. Die Neuordnung tritt nicht auf, wenn der Kanal bereits ausgefallen ist.

Das System versucht, eine Nachricht erneut zu senden, wenn der Zielwarteschlangenmanager nicht mehr in Betrieb ist. In diesem Bereich wirkt sich dies nicht auf die Integrität der Nachricht aus, indem die Gefahr besteht, dass die Nachricht verloren geht oder ein Duplikat erstellt wird. Wenn ein WS-Manager fehlschlägt und eine Nachricht im Zweifel bleibt, wird diese Nachricht nicht weitergeleitet.

## Hinzufügen eines Warteschlangenmanagers, der eine lokale Warteschlange enthält

Führen Sie die folgenden Anweisungen aus, um eine Instanz von `INVENTQ` hinzuzufügen, um zusätzliche Kapazität für die Ausführung des Inventaranwendungs-Systems in Paris und New York bereitzustellen.

### Vorbereitende Schritte

**Anmerkung:** Damit Änderungen an einem Cluster im gesamten Cluster weitergegeben werden können, muss immer mindestens ein vollständiges Repository verfügbar sein. Stellen Sie sicher, dass Ihre Repositories verfügbar sind, bevor Sie diese Task starten.

Szenario:

- Der `INVENTORY` -Cluster wurde wie im Abschnitt [Neuen WS-Manager zu einem Cluster hinzufügen](#) beschrieben konfiguriert. Es enthält drei Warteschlangenmanager; `LONDON` und `NEWYORK` enthalten beide vollständige Repositories, `PARIS` enthält ein Teilrepository. Die Bestandsanwendung wird auf dem

System in New York ausgeführt und ist mit dem NEWYORK -Warteschlangenmanager verbunden. Die Anwendung wird durch den Eingang von Nachrichten in der INVENTQ -Warteschlange gesteuert.

- Wir möchten eine Instanz von INVENTQ hinzufügen, um zusätzliche Kapazität für die Ausführung des Inventaranwendungssystems in Paris und New York bereitzustellen.

## Informationen zu diesem Vorgang

Führen Sie die folgenden Schritte aus, um einen Warteschlangenmanager hinzuzufügen, der eine lokale Warteschlange enthält.

## Vorgehensweise

1. Ändern Sie den PARIS -Warteschlangenmanager.

Für die Anwendung in Paris, um die INVENTQ in Paris und die in New York zu verwenden, müssen wir den Warteschlangenmanager informieren. Geben Sie unter PARIS den folgenden Befehl ein:

```
ALTER QMGR CLWLUSEQ(ANY)
```

2. Überprüfen Sie die Bestandsanwendung auf Nachrichtenaffinitäten.

Bevor Sie fortfahren, stellen Sie sicher, dass die Inventaranwendung keine Abhängigkeiten von der Reihenfolge der Verarbeitung von Nachrichten hat. Weitere Informationen finden Sie in [„Nachrichtenaffinitäten bearbeiten“](#) auf Seite 298.

3. Installieren Sie die Inventaranwendung auf dem System in Paris.

4. Definieren Sie die Clusterwarteschlange INVENTQ.

Die INVENTQ -Warteschlange, die bereits vom NEWYORK -Warteschlangenmanager gehostet wird, wird auch von PARIS gehostet. Definieren Sie sie auf dem PARIS -Warteschlangenmanager wie folgt:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

Nachdem Sie alle Definitionen abgeschlossen haben (falls noch nicht geschehen), starten Sie den Kanalinitiator unter WebSphere MQ für z/OS. Starten Sie auf allen Plattformen ein Listenerprogramm auf dem Warteschlangenmanager PARIS. Der Listener ist für eingehende Netzanforderungen empfängsbereit und startet den Clusterempfängerkanal, wenn er benötigt wird.

## Ergebnisse

In [Abbildung 50 auf Seite 288](#) wird der Cluster angezeigt, der von dieser Task eingerichtet wurde.



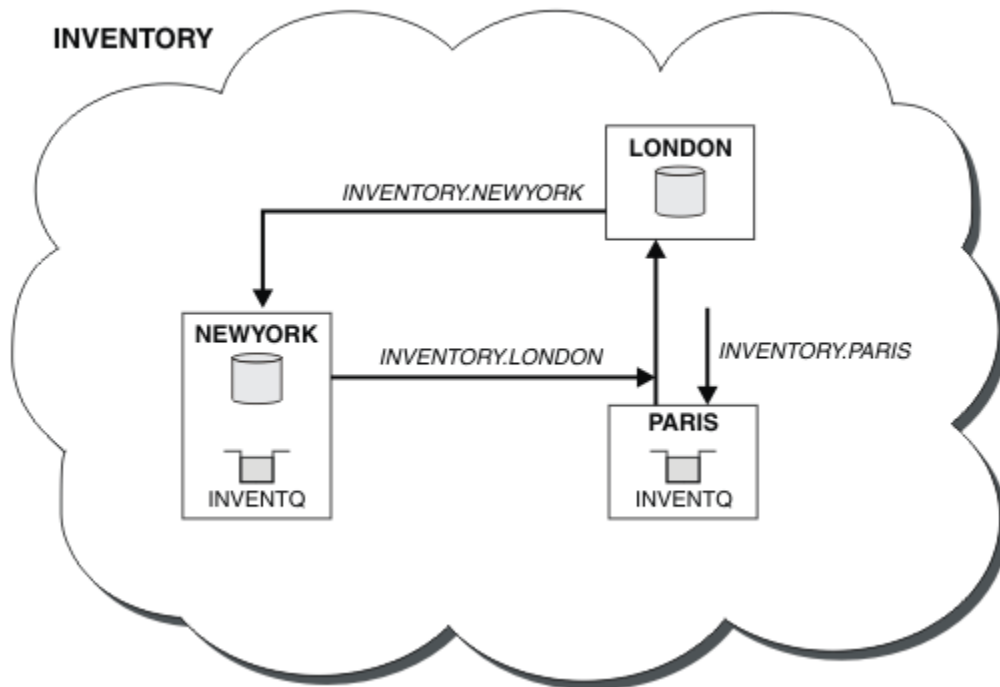


Abbildung 50. Der INVENTORY -Cluster mit drei Warteschlangenmanagern

Die Änderungen an diesem Cluster wurden ohne Änderung der Warteschlangenmanager NEWYORK oder LONDON vorgenommen. Die vollständigen Repositories in diesen Warteschlangenmanagern werden automatisch mit den Informationen aktualisiert, die sie benötigen, um Nachrichten an INVENTQ in PARIS senden zu können.

## Nächste Schritte

Die INVENTQ -Warteschlange und die Inventaranwendung werden jetzt auf zwei Warteschlangenmanagern im Cluster gehostet. Dies erhöht die Verfügbarkeit, beschleunigt den Durchsatz von Nachrichten und ermöglicht die Verteilung der Auslastung zwischen den beiden Warteschlangenmanagern. Nachrichten, die von einem der WS-Manager LONDON, NEWYORK, PARIS an INVENTQ gesendet werden, werden abwechselnd an PARIS oder NEWYORK weitergeleitet, so dass die Auslastung ausgeglichen ist.

## Zwei Netze in einem Cluster verwenden

Führen Sie die folgenden Anweisungen aus, um einen neuen Speicher in TOKYO hinzuzufügen, in dem sich zwei verschiedene Netze befinden. Beide müssen für die Kommunikation mit dem WS-Manager in Tokio verfügbar sein.

## Vorbereitende Schritte

**Anmerkung:** Damit Änderungen an einem Cluster im gesamten Cluster weitergegeben werden können, muss immer mindestens ein vollständiges Repository verfügbar sein. Stellen Sie sicher, dass Ihre Repositories verfügbar sind, bevor Sie diese Task starten.

Szenario:

- Der INVENTORY -Cluster wurde wie im Abschnitt "WS-Manager zu einem Cluster hinzufügen" beschrieben konfiguriert. Es enthält drei Warteschlangenmanager; LONDON und NEWYORK enthalten beide vollständige Repositories, PARIS enthält ein Teilrepository. Die Bestandsanwendung wird auf dem System in New York ausgeführt und ist mit dem NEWYORK -Warteschlangenmanager verbunden. Die Anwendung wird durch den Eingang von Nachrichten in der INVENTQ -Warteschlange gesteuert.



- Ein neues Geschäft wird in TOKYO hinzugefügt, wo es zwei verschiedene Netzwerke gibt. Beide müssen für die Kommunikation mit dem WS-Manager in Tokio verfügbar sein.

## Informationen zu diesem Vorgang

Führen Sie die folgenden Schritte aus, um zwei Netze in einem Cluster zu verwenden.

### Vorgehensweise

1. Entscheiden Sie, welches vollständige Repository TOKYO auf das erste Element verweist.

Jeder Warteschlangenmanager in einem Cluster muss sich auf einen oder einen der vollständigen Repositories beziehen, um Informationen zum Cluster zu erfassen. Es baut sein eigenes Teilrepository auf. Es ist nicht besonders wichtig, welches Repository Sie auswählen. In diesem Beispiel wird NEWYORK ausgewählt. Sobald der neue WS-Manager dem Cluster beigetreten ist, kommuniziert er mit beiden Repositories.

2. Definieren Sie die CLUSRCVR -Kanäle.

Jeder Warteschlangenmanager in einem Cluster muss einen Clusterempfänger definieren, auf dem er Nachrichten empfangen kann. Dieser WS-Manager muss in jedem Netz kommunizieren können.

```
DEFINE CHANNEL(INVENTORY.TOKYO.NETB) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CON-
NAME('TOKYO.NETB.CMSTORE.COM') CLUSTER(INVENTORY) DESCR('Cluster-receiver
channel using network B for TOKYO')
```

```
DEFINE CHANNEL(INVENTORY.TOKYO.NETA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CON-
NAME('TOKYO.NETA.CMSTORE.COM') CLUSTER(INVENTORY) DESCR('Cluster-receiver
channel using network A for TOKYO')
```

3. Definieren Sie einen CLUSSDR -Kanal auf dem Warteschlangenmanager TOKYO..

Jeder Warteschlangenmanager in einem Cluster muss einen Clustersenderkanal definieren, auf dem er Nachrichten an sein erstes vollständiges Repository senden kann. In diesem Fall haben wir NEWYORK ausgewählt, daher benötigt TOKYO die folgende Definition:

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP) CONNAME(NEW-
YORK.CHSTORE.COM) CLUSTER(INVENTORY) DESCR('Cluster-sender channel from TO-
KYO to repository at NEWYORK')
```

Nachdem Sie alle Definitionen abgeschlossen haben, starten Sie den Kanalinitiator unter WebSphere MQ for z/OS, sofern dies noch nicht geschehen ist. Starten Sie auf allen Plattformen ein Listenerprogramm auf dem Warteschlangenmanager PARIS. Das Empfangsprogramm ist empfangsbereit für eingehende Netzanforderungen und startet den Clusterempfängerkanal, wenn er benötigt wird.

### Ergebnisse

In [Abbildung 51 auf Seite 290](#) wird der Cluster angezeigt, der von dieser Task eingerichtet wurde.

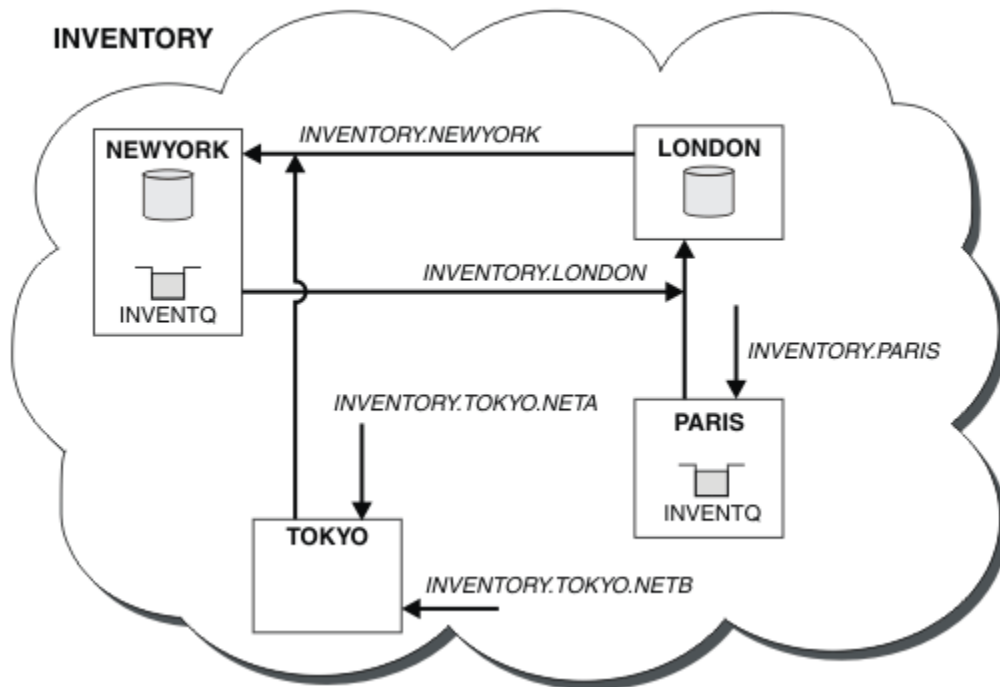


Abbildung 51. Der INVENTORY -Cluster mit vier Warteschlangenmanagern

Wenn Sie nur drei Definitionen vornehmen, haben wir den WS-Manager TOKYO dem Cluster mit zwei verschiedenen Netzrouten hinzugefügt.

### Zugehörige Tasks

„WS-Manager zu einem Cluster hinzufügen“ auf Seite 211

Befolgen Sie diese Anweisungen, um dem erstellten Cluster einen WS-Manager hinzuzufügen. Nachrichten zu Clusterwarteschlangen und Themen werden unter Verwendung der einzigen Clusterübertragungs-warteschlange `SYSTEM.CLUSTER.TRANSMIT.QUEUE` übertragen.

### Primäres und sekundäres Netz in einem Cluster verwenden

Führen Sie die folgenden Anweisungen aus, um ein Netz zum primären Netz zu machen, und ein anderes Netz das Ausweichnetz zu erstellen. Verwenden Sie das Ausweichnetz, wenn ein Problem mit dem primären Netz besteht.

### Vorbereitende Schritte

**Anmerkung:** Damit Änderungen an einem Cluster im gesamten Cluster weitergegeben werden können, muss immer mindestens ein vollständiges Repository verfügbar sein. Stellen Sie sicher, dass Ihre Repositories verfügbar sind, bevor Sie diese Task starten.

Szenario:

- Der INVENTORY -Cluster wurde wie in „Zwei Netze in einem Cluster verwenden“ auf Seite 288 beschrieben konfiguriert. Sie enthält vier WS-Manager; LONDON und NEWYORK enthalten vollständige Repositories; PARIS und TOKYO enthalten Teilrepositories. Die Bestandsanwendung wird auf dem System in New York ausgeführt und ist mit dem Warteschlangenmanager NEWYORK verbunden. Der WS-Manager von TOKYO verfügt über zwei verschiedene Netze, auf denen er kommunizieren kann.
- Sie möchten eines der Netze das primäre Netz und ein anderes der Netze das Ausweichnetz bilden. Sie planen, das Ausweichnetz zu verwenden, wenn ein Problem mit dem primären Netz besteht.

## Informationen zu diesem Vorgang

Verwenden Sie das Attribut NETPRTY , um ein primäres und ein sekundäres Netz in einem Cluster zu konfigurieren.

### Vorgehensweise

Ändern Sie die vorhandenen CLUSRCVR -Kanäle in TOKYO.

Verwenden Sie die folgenden Befehle, um anzugeben, dass das Netz ein Kanal der primäre Kanal und der Kanal des Netzes B der sekundäre Kanal ist:

- a) ALTER CHANNEL(INVENTORY.TOKYO.NETA) CHLTYPE(CLUSRCVR) NETPRTY(2) DESCR('Main cluster-receiver channel for TOKYO')
- b) ALTER CHANNEL(INVENTORY.TOKYO.NETB) CHLTYPE(CLUSRCVR) NETPRTY(1) DESCR('Backup cluster-receiver channel for TOKYO')

### Nächste Schritte

Wenn Sie den Kanal mit unterschiedlichen Netzprioritäten konfigurieren, haben Sie jetzt den Cluster definiert, dass Sie über ein primäres Netz und ein sekundäres Netz verfügen. Die Warteschlangenmanager im Cluster, die diese Kanäle verwenden, verwenden automatisch das primäre Netz, wenn es verfügbar ist. Die WS-Manager-Funktionsübernahme (Failover) für die Verwendung des sekundären Netzes, wenn das primäre Netz nicht verfügbar ist.

## Warteschlange hinzufügen, die als Sicherung dienen soll

Folgen Sie diesen Anweisungen, um eine Sicherung in Chicago für das Inventarsystem bereitzustellen, das jetzt in New York ausgeführt wird. Das Chicagoer System wird nur verwendet, wenn es ein Problem mit dem New Yorker System gibt.

### Vorbereitende Schritte

**Anmerkung:** Damit Änderungen an einem Cluster im gesamten Cluster weitergegeben werden können, muss immer mindestens ein vollständiges Repository verfügbar sein. Stellen Sie sicher, dass Ihre Repositories verfügbar sind, bevor Sie diese Task starten.

Szenario:

- Der INVENTORY -Cluster wurde wie in „WS-Manager zu einem Cluster hinzufügen“ auf Seite 211 beschrieben konfiguriert. Es enthält drei Warteschlangenmanager; LONDON und NEWYORK enthalten beide vollständige Repositories, PARIS enthält ein Teilrepository. Die Bestandsanwendung wird auf dem System in New York ausgeführt und ist mit dem NEWYORK -Warteschlangenmanager verbunden. Die Anwendung wird durch den Eingang von Nachrichten in der INVENTQ -Warteschlange gesteuert.
- Es wird ein neues Geschäft in Chicago eingerichtet, um eine Sicherung für das Inventarsystem bereitzustellen, das jetzt in New York läuft. Das Chicago-System wird nur verwendet, wenn es ein Problem mit dem New Yorker System gibt.

## Informationen zu diesem Vorgang

Führen Sie die folgenden Schritte aus, um eine Warteschlange hinzuzufügen, die als Sicherung dienen soll.

### Vorgehensweise

1. Entscheiden Sie, welches vollständige Repository CHICAGO auf das erste Element verweist.

Jeder Warteschlangenmanager in einem Cluster muss sich auf einen oder einen der vollständigen Repositories beziehen, um Informationen zum Cluster zu erfassen. Es baut sein eigenes Teilrepository auf. Es ist nicht besonders wichtig, welches Repository Sie für einen bestimmten Warteschlangenma-

nager auswählen. In diesem Beispiel wird NEWYORK ausgewählt. Sobald der neue WS-Manager dem Cluster beigetreten ist, kommuniziert er mit beiden Repositories.

2. Definieren Sie den Kanal CLUSRCVR .

Jeder Warteschlangenmanager in einem Cluster muss einen Clusterempfänger definieren, auf dem er Nachrichten empfangen kann. Definieren Sie unter CHICAGO Folgendes:

```
DEFINE CHANNEL(INVENTORY.CHICAGO) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CONNAME(CHICAGO.CMSTORE.COM) CLUSTER(INVENTORY) DESCR('Cluster-receiver channel for CHICAGO')
```

3. Definieren Sie einen CLUSSDR -Kanal auf WS-Manager CHICAGO.

Jeder Warteschlangenmanager in einem Cluster muss einen Clustersenderkanal definieren, auf dem er Nachrichten an sein erstes vollständiges Repository senden kann. In diesem Fall haben wir NEWYORK ausgewählt, daher benötigt CHICAGO die folgende Definition:

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP) CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY) DESCR('Cluster-sender channel from CHICAGO to repository at NEWYORK')
```

4. Ändern Sie die vorhandene Clusterwarteschlange INVENTQ.

Das INVENTQ , das bereits vom NEWYORK -Warteschlangenmanager gehostet wird, ist die Hauptinstanz der Warteschlange.

```
ALTER QLOCAL(INVENTQ) CLWLPRTY(2)
```

5. Überprüfen Sie die Bestandsanwendung auf Nachrichtenaffinitäten.

Bevor Sie fortfahren, stellen Sie sicher, dass die Inventaranwendung keine Abhängigkeiten von der Reihenfolge der Verarbeitung von Nachrichten hat.

6. Installieren Sie die Bestandsanwendung auf dem System in CHICAGO.

7. Definieren Sie die Sicherungs-Cluster-Warteschlange INVENTQ

Der INVENTQ , der bereits vom NEWYORK -Warteschlangenmanager gehostet wird, wird auch als Sicherung von CHICAGO gehostet. Definieren Sie sie auf dem CHICAGO -Warteschlangenmanager wie folgt:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY) CLWLPRTY(1)
```

Nachdem Sie alle Definitionen abgeschlossen haben, starten Sie den Kanalinitiator unter WebSphere MQ for z/OS, sofern dies noch nicht geschehen ist. Starten Sie auf allen Plattformen ein Listenerprogramm auf dem Warteschlangenmanager CHICAGO. Das Empfangsprogramm ist empfangsbereit für eingehende Netzanforderungen und startet den Clusterempfängerkanal, wenn er benötigt wird.

## Ergebnisse

In [Abbildung 52 auf Seite 293](#) wird der Cluster angezeigt, der von dieser Task eingerichtet wurde.

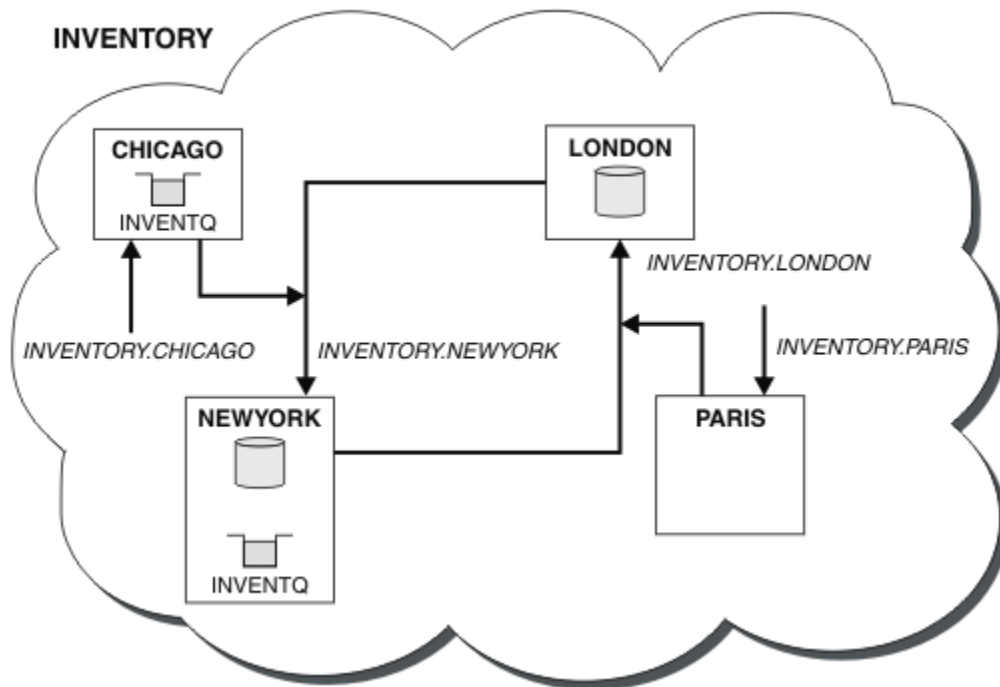


Abbildung 52. Cluster BESTANDSFÜHRUNG mit vier Warteschlangenmanagern

Die INVENTQ -Warteschlange und die Inventaranwendung werden jetzt auf zwei Warteschlangenmanagern im Cluster gehostet. Der WS-Manager von CHICAGO ist eine Sicherung. Nachrichten, die an INVENTQ gesendet werden, werden an NEWYORK weitergeleitet, es sei denn, sie sind nicht verfügbar, wenn sie statt CHICAGO gesendet werden.

**Anmerkung:**

Die Verfügbarkeit eines fernen Warteschlangenmanagers basiert auf dem Status des Kanals für diesen Warteschlangenmanager. Wenn Kanäle gestartet werden, ändert sich ihr Status mehrmals, wobei einige Status für den Algorithmus für das Clusterauslastungsmanagement weniger günstig sind. In der Praxis bedeutet dies, dass Ziele mit einer niedrigeren Priorität (Sicherung) ausgewählt werden können, während die Kanäle zu übergeordneten (primären) Zielen gestartet werden.

Wenn Sie sicherstellen müssen, dass keine Nachrichten an ein Sicherungsziel gesendet werden, verwenden Sie CLWLPRTY nicht. Sie könnten getrennte Warteschlangen verwenden oder CLWLRANK mit manuellem Umschalten vom primären zum Sicherungsziel.

**Einschränkung der Anzahl verwendeter Kanäle**

Befolgen Sie diese Anweisungen, um die Anzahl der aktiven Kanäle zu beschränken, die jeder Server ausführt, wenn eine Preisprüfung auf verschiedenen Warteschlangenmanagern installiert ist.

**Vorbereitende Schritte**

**Anmerkung:** Damit Änderungen an einem Cluster im gesamten Cluster weitergegeben werden können, muss immer mindestens ein vollständiges Repository verfügbar sein. Stellen Sie sicher, dass Ihre Repositories verfügbar sind, bevor Sie diese Task starten.

Szenario:

- Es soll eine Preisprüfung auf verschiedenen Warteschlangenmanagern installiert werden. Um die Anzahl der Kanäle, die für eine niedrige Zahl verwendet werden, zu halten, ist die Anzahl der aktiven Kanäle, die jeder Server ausführt, eingeschränkt. Die Anwendung wird durch den Eingang von Nachrichten in der PRICEQ -Warteschlange gesteuert.

- Vier Server-WS-Manager hosten die Anwendung "Preisprüfung". Zwei Abfragenwarteschlangenmanager senden Nachrichten an den PRICEQ, um einen Preis abzufragen. Zwei weitere WS-Manager werden als vollständige Repositorys konfiguriert.

## Informationen zu diesem Vorgang

Führen Sie die folgenden Schritte aus, um die Anzahl der verwendeten Kanäle zu beschränken.

## Vorgehensweise

1. Wählen Sie zwei vollständige Repositorys aus.

Wählen Sie zwei Warteschlangenmanager aus, die die vollständigen Repositorys für Ihren Preiscluster-Cluster sein sollen. Sie werden als REPOS1 und REPOS2 bezeichnet.

Geben Sie den folgenden Befehl ein:

```
ALTER QMGR REPOS (PRICECHECK)
```

2. Definieren Sie einen CLUSRCVR -Kanal auf jedem Warteschlangenmanager.

Definieren Sie in jedem WS-Manager im Cluster einen Clusterempfängerkanal und einen Clustersenderkanal. Es spielt keine Rolle, die zuerst definiert wird.

```
DEFINE CHANNEL (PRICECHECK.SERVE1) CHLTYPE (CLUSRCVR) TRPTYPE (TCP) CONNAME (SERVER1.COM) CLUSTER (PRICECHECK) DESCR ('Cluster-receiver channel')
```

3. Definieren Sie auf jedem WS-Manager einen CLUSSDR -Kanal.

Erstellen Sie in jedem WS-Manager eine CLUSSDR -Definition, um diesen Warteschlangenmanager mit einem oder einem der vollständigen WS-Manager-Repositorys zu verbinden.

```
DEFINE CHANNEL (PRICECHECK.REPOS1) CHLTYPE (CLUSSDR) TRPTYPE (TCP) CONNAME (REPOS1.COM) CLUSTER (PRICECHECK) DESCR ('Cluster-sender channel to repository queue manager')
```

4. Installieren Sie die Anwendung "Preisprüfung".
5. Definieren Sie die PRICEQ -Warteschlange auf allen WS-Managern des Servers.

Geben Sie den folgenden Befehl für jede ein:

```
DEFINE QLOCAL (PRICEQ) CLUSTER (PRICECHECK)
```

6. Anzahl der Kanäle, die von Abfragen verwendet werden, strikt

Auf den WS-Managern der Abfrage beschränken wir die Anzahl der verwendeten aktiven Kanäle, indem Sie die folgenden Befehle für jede der folgenden Befehle ausgeben:

```
ALTER QMGR CLWLMRUC (2)
```

7. Falls noch nicht geschehen, starten Sie den Kanalinitiator unter WebSphere MQ für z/OS. Starten Sie auf allen Plattformen ein Empfangsprogramm.

Das Empfangsprogramm ist empfangsbereit für eingehende Netzanforderungen und startet den Clusterempfängerkanal, wenn er benötigt wird.

## Ergebnisse

In [Abbildung 53 auf Seite 295](#) wird der Cluster angezeigt, der von dieser Task eingerichtet wurde.

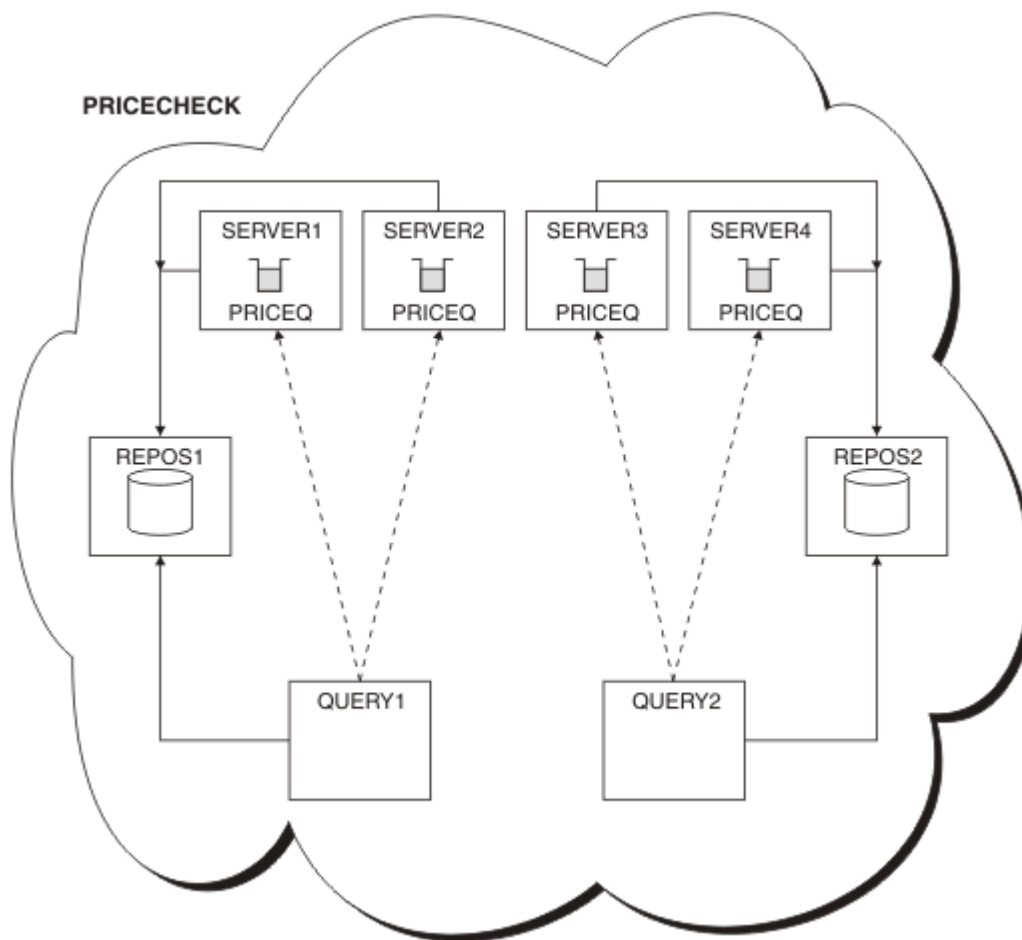


Abbildung 53. Der PRICECHECK -Cluster mit vier Server-WS-Managern, zwei Repositories und zwei Abfragerwarteschlangenmanagern

Obwohl es vier Instanzen der PRICEQ -Warteschlange im PRICECHECK -Cluster gibt, verwendet jeder Abfragerwarteschlangenmanager nur zwei von zwei davon. Der WS-Manager von QUERY1 verfügt beispielsweise nur über aktive Kanäle zu den WS-Managern SERVER1 und SERVER2 . Wenn SERVER1 nicht mehr verfügbar ist, beginnt der QUERY1 -Warteschlangenmanager anschließend mit der Verwendung eines anderen Warteschlangenmanagers, z. B. SERVER3.

### Nächste Schritte

Obwohl es vier Instanzen der PRICEQ -Warteschlange im PRICECHECK -Cluster gibt, verwendet jeder Abfragerwarteschlangenmanager nur zwei von zwei davon. Der WS-Manager von QUERY1 verfügt beispielsweise nur über aktive Kanäle zu den WS-Managern SERVER1 und SERVER2 . Wenn SERVER1 nicht mehr verfügbar ist, beginnt der QUERY1 -Warteschlangenmanager anschließend mit der Verwendung eines anderen Warteschlangenmanagers, z. B. SERVER3.

### Hinzufügen eines leistungsfähigeren Warteschlangenmanagers, der eine Warteschlange enthält

Befolgen Sie diese Anweisungen, um zusätzliche Kapazität bereitzustellen, indem Sie das Bestandssystem in Los Angeles sowie New York ausführen, wo Los Angeles die doppelte Anzahl von Nachrichten als New York verarbeiten kann.

## Vorbereitende Schritte

**Anmerkung:** Damit Änderungen an einem Cluster im gesamten Cluster weitergegeben werden können, muss immer mindestens ein vollständiges Repository verfügbar sein. Stellen Sie sicher, dass Ihre Repositories verfügbar sind, bevor Sie diese Task starten.

Szenario:

- Der INVENTORY-Cluster wurde wie in „WS-Manager zu einem Cluster hinzufügen“ auf Seite 211 beschrieben eingerichtet. Es enthält drei WS-Manager: LONDON und NEWYORK enthalten vollständige Repositories, PARIS enthält ein Teilrepository und stellt Nachrichten aus INVENTQ. Die Bestandsanwendung wird auf dem System in New York ausgeführt, das mit dem NEWYORK -Warteschlangenmanager verbunden ist. Die Anwendung wird durch den Eingang von Nachrichten in der INVENTQ -Warteschlange gesteuert.
- Es wird ein neues Geschäft in Los Angeles eingerichtet. Um zusätzliche Kapazitäten zu schaffen, wollen Sie das Inventarsystem in Los Angeles und New York betreiben. Der neue WS-Manager kann doppelt so viele Nachrichten verarbeiten wie New York.

## Informationen zu diesem Vorgang

Führen Sie die folgenden Schritte aus, um einen leistungsfähigeren Warteschlangenmanager hinzuzufügen, der als Host für eine Warteschlange dient.

## Vorgehensweise

1. Entscheiden Sie, welches vollständige Repository LOSANGELES auf das erste Element verweist.
2. Jeder Warteschlangenmanager in einem Cluster muss sich auf einen oder einen der vollständigen Repositories beziehen, um Informationen zum Cluster zu erfassen. Es baut sein eigenes Teilrepository auf. Es ist nicht besonders wichtig, welches Repository Sie auswählen. In diesem Beispiel wird NEWYORK ausgewählt. Sobald der neue WS-Manager dem Cluster beigetreten ist, kommuniziert er mit beiden Repositories.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from LOSANGELES to repository at NEWYORK')
```

3. Definieren Sie den Kanal CLUSRCVR auf WS-Manager LOSANGELES.

Jeder WS-Manager in einem Cluster muss einen Clusterempfängerkanal definieren, auf dem er Nachrichten empfangen kann. Definieren Sie unter LOSANGELES Folgendes:

```
DEFINE CHANNEL(INVENTORY.LOSANGELES) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LOSANGELES.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for queue manager LOSANGELES')
CLWLWGHT(2)
```

Der Clusterempfängerkanal wirbt für die Verfügbarkeit des Warteschlangenmanagers, um Nachrichten von anderen Warteschlangenmanagern im Cluster INVENTORY zu empfangen. Wenn Sie CLWLWGHT auf zwei setzen, wird sichergestellt, dass der WS-Manager von Los Angeles doppelt so viele Bestandsnachrichten wie New York erhält (wenn der Kanal für NEWYORK auf einen Wert gesetzt ist).

4. Ändern Sie den Kanal CLUSRCVR auf dem Warteschlangenmanager NEWYORK.

Stellen Sie sicher, dass der WS-Manager von Los Angeles doppelt so viele Inventarnachrichten wie New York erhält. Ändern Sie die Definition des Clusterempfängerkanals.

```
ALTER CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) CLWLWGHT(1)
```

5. Überprüfen Sie die Bestandsanwendung auf Nachrichtenaffinitäten.

Bevor Sie fortfahren, stellen Sie sicher, dass die Inventaranwendung keine Abhängigkeiten von der Reihenfolge der Verarbeitung von Nachrichten hat.

6. Installieren Sie die Inventaranwendung auf dem System in Los Angeles.
7. Definieren Sie die Clusterwarteschlange INVENTQ.



Die INVENTQ -Warteschlange, die bereits vom NEWYORK -Warteschlangenmanager gehostet wird, befindet sich ebenfalls in LOSANGELES. Definieren Sie sie auf dem LOSANGELES -Warteschlangenmanager wie folgt:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

Nachdem Sie alle Definitionen abgeschlossen haben, starten Sie den Kanalinitiator unter WebSphere MQ for z/OS, sofern dies noch nicht geschehen ist. Starten Sie auf allen Plattformen ein Listenempfangsprogramm auf dem Warteschlangenmanager LOSANGELES. Das Empfangsprogramm ist empfangsbereit für eingehende Netzanforderungen und startet den Clusterempfängerkanal, wenn er benötigt wird.

## Ergebnisse

In „Hinzufügen eines leistungsfähigeren Warteschlangenmanagers, der eine Warteschlange enthält“ auf Seite 295 wird der Cluster angezeigt, der von dieser Task eingerichtet wurde.

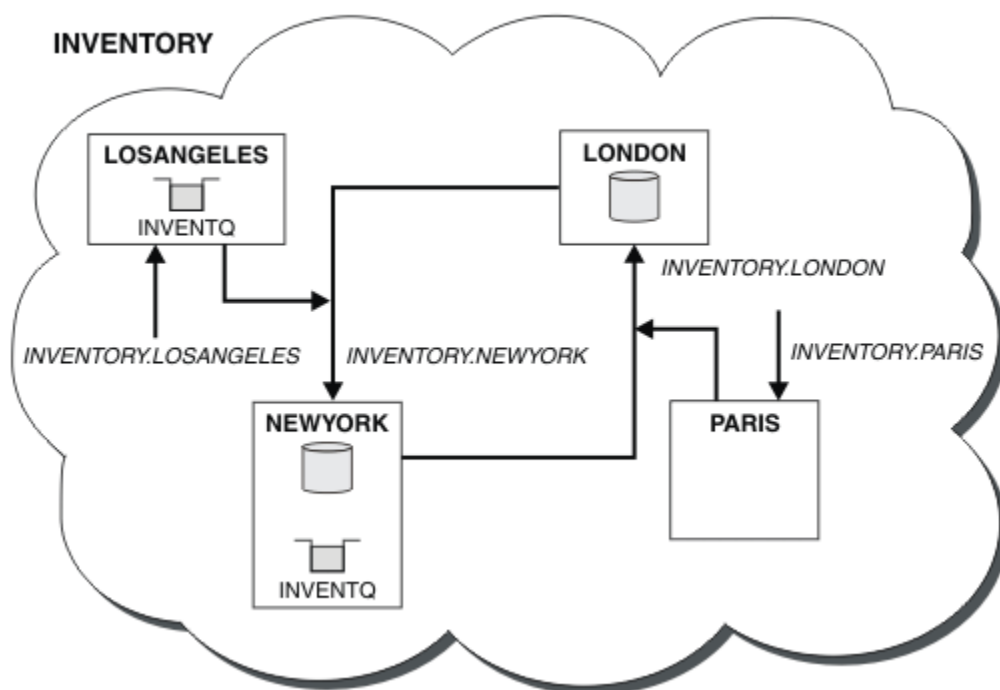


Abbildung 54. Der INVENTORY -Cluster mit vier Warteschlangenmanagern

Diese Änderung am Cluster wurde ausgeführt, ohne dass die Warteschlangenmanager LONDON und PARIS geändert werden müssen. Die Repositories in diesen Warteschlangenmanagern werden automatisch mit den Informationen aktualisiert, die sie benötigen, um Nachrichten an INVENTQ in LOSANGELES senden zu können.

## Nächste Schritte

Die INVENTQ -Warteschlange und die Inventaranwendung werden auf zwei Warteschlangenmanagern im Cluster gehostet. Die Konfiguration erhöht die Verfügbarkeit, beschleunigt den Durchsatz von Nachrichten und ermöglicht die Verteilung der Auslastung zwischen den beiden Warteschlangenmanagern. Nachrichten, die entweder von LOSANGELES oder NEWYORK an INVENTQ gestellt werden, werden, wenn möglich, von der Instanz auf dem lokalen WS-Manager bearbeitet. Nachrichten, die von LONDON oder PARIS gestellt werden, werden an LOSANGELES oder NEWYORK weitergeleitet, wobei doppelt so viele Nachrichten an LOSANGELES gesendet werden.

## Anwendungsprogrammierung und Cluster

Sie müssen keine Programmieränderungen vornehmen, um die Vorteile mehrerer Instanzen derselben Warteschlange nutzen zu können. Einige Programme funktionieren jedoch nicht korrekt, es sei denn, eine Folge von Nachrichten wird an dieselbe Instanz einer Warteschlange gesendet.

Anwendungen können eine Warteschlange mit dem Aufruf MQOPEN öffnen. Anwendungen verwenden den Aufruf MQPUT, um Nachrichten in eine geöffnete Warteschlange einzureihen. Anwendungen können eine einzelne Nachricht mit dem Aufruf MQPUT1 in eine Warteschlange stellen, die noch nicht geöffnet ist.

Wenn Sie Cluster konfigurieren, die über mehrere Instanzen derselben Warteschlange verfügen, gibt es keine spezifischen Hinweise zur Anwendungsprogrammierung. Wenn Sie jedoch die Aspekte des Workloadmanagements im Clustering nutzen möchten, müssen Sie möglicherweise Ihre Anwendungen ändern. Wenn Sie ein Netz einrichten, in dem mehrere Definitionen derselben Warteschlange vorhanden sind, überprüfen Sie Ihre Anwendungen auf Nachrichtenaffinitäten.

Angenommen, Sie haben zwei Anwendungen, die sich auf eine Reihe von Nachrichten stützen, die zwischen ihnen in Form von Fragen und Antworten fließen. Sie möchten wahrscheinlich, dass Antworten auf denselben WS-Manager zurückgehen, der eine Frage gesendet hat. Es ist wichtig, dass die Workload-Management-Routine die Nachrichten nicht an einen WS-Manager sendet, der eine Kopie der Antwortwarteschlange enthält.

Es können Anwendungen vorhanden sein, für die Nachrichten in der Reihenfolge verarbeitet werden müssen (z. B. eine Datenbankanwendung, die Stapel von Nachrichten sendet, die in der Sequenz abgerufen werden müssen). Die Verwendung von segmentierten Nachrichten kann auch zu einem Affinitätsproblem führen.

### Lokale oder ferne Version der Zielwarteschlange öffnen

Achten Sie darauf, wie der Warteschlangenmanager auswählt, ob eine lokale oder eine ferne Version der Zielwarteschlange verwendet wird.

1. Der Warteschlangenmanager öffnet die lokale Version der Zielwarteschlange, um Nachrichten zu lesen, oder um die Attribute der Warteschlange festzulegen.
2. Der Warteschlangenmanager öffnet eine beliebige Instanz der Zielwarteschlange, in die Nachrichten geschrieben werden sollen, wenn mindestens eine der folgenden Bedingungen zutrifft:
  - Es ist keine lokale Version der Zielwarteschlange vorhanden.
  - Der Warteschlangenmanager gibt CLWLUSEQ(ANY) unter ALTER QMGRan.
  - Die Warteschlange auf dem WS-Manager gibt CLWLUSEQ(ANY) an.

### Nachrichtenaffinitäten bearbeiten

Nachrichtenaffinitäten sind selten Teil eines guten Programmierdesigns. Sie müssen Nachrichtenaffinitäten entfernen, um das Clustering vollständig zu verwenden. Wenn Sie keine Nachrichtenaffinitäten entfernen können, können Sie erzwingen, dass zugehörige Nachrichten mit demselben Kanal und mit demselben Warteschlangenmanager zugestellt werden.

Wenn Sie Anwendungen mit Nachrichtenaffinitäten haben, entfernen Sie die Affinitäten, bevor Sie mit der Verwendung von Clustern beginnen.

Durch das Entfernen von Nachrichtenaffinitäten wird die Verfügbarkeit von Anwendungen verbessert. Eine Anwendung sendet einen Stapel von Nachrichten, die Nachrichtenaffinitäten an einen Warteschlangenmanager senden. Der Warteschlangenmanager schlägt fehl, nachdem er nur einen Teil der Stapelverarbeitung empfangen hat. Der sendende Warteschlangenmanager muss warten, bis er die unvollständige Nachrichtenstapelverarbeitung wiederhergestellt und verarbeitet hat, bevor er weitere Nachrichten senden kann.

Durch das Entfernen von Nachrichtenaffinitäten wird auch die Skalierbarkeit von Anwendungen verbessert. Ein Stapel von Nachrichten mit Affinitäten kann Ressourcen auf dem Zielwarteschlangenmanager sperren, während er auf nachfolgende Nachrichten wartet. Diese Ressourcen bleiben möglicherweise lange Zeit gesperrt, so dass andere Anwendungen ihre Arbeit nicht ausführen können.

Darüber hinaus verhindern Nachrichtenaffinitäten, dass die Cluster-Workload-Management-Routinen die beste Auswahl an WS-Managern treffen.

Wenn Sie Affinitäten entfernen möchten, sollten Sie die folgenden Möglichkeiten berücksichtigen:

- Statusinformationen in den Nachrichten eintragen
- Verwaltung von Statusinformation in einem für alle Warteschlangenmanager zugänglichen nicht flüchtigen Speicher, z. B. in einer Db2-Datenbank
- Replizieren von schreibgeschützten Daten, so dass auf mehrere WS-Manager zugegriffen werden kann

Wenn es nicht sinnvoll ist, Ihre Anwendungen so zu ändern, dass Nachrichtenaffinitäten entfernt werden, gibt es eine Reihe möglicher Lösungen für das Problem.

### **Ein bestimmtes Ziel im MQOPEN -Aufruf benennen**

Geben Sie den Namen der fernen Warteschlange und den Namen des Warteschlangenmanagers in jedem MQOPEN -Aufruf an. Alle Nachrichten, die unter Verwendung dieser Objektkennung in die Warteschlange eingereiht werden, gehen an denselben Warteschlangenmanager, bei dem es sich möglicherweise um den lokalen Warteschlangenmanager handelt.

Die Angabe des Namens der fernen Warteschlange und des Namens des Warteschlangenmanagers in jedem MQOPEN -Aufruf hat Nachteile:

- Es wird kein Lastausgleich durchgeführt. Sie profitieren nicht von den Vorteilen des Lastausgleichs in der Clusterauslastung.
- Wenn der Zielwarteschlangenmanager fern ist und mehr als ein Kanal zu ihm vorhanden ist, können die Nachrichten unterschiedliche Routen verwenden, und die Nachrichtenfolge wird immer noch nicht beibehalten.
- Wenn Ihr Warteschlangenmanager über eine Definition für eine Übertragungswarteschlange mit demselben Namen wie der Zielwarteschlangenmanager verfügt, werden Nachrichten in diese Übertragungswarteschlange und nicht in die Clusterübertragungswarteschlange übertragen.

### **Geben Sie den Namen des WS-Managers im Feld für den Antwortwarteschlangenmanager zurück.**

Ermöglichen Sie dem WS-Manager, der die erste Nachricht in einem Stapel empfängt, seinen Namen in seiner Antwort zurück. Dies erfolgt über das Feld `ReplyToQMgr` des Nachrichtendeskriptors. Der WS-Manager am sendenden Ende kann dann den Namen des Antwortwarteschlangenmanagers extrahieren und in allen nachfolgenden Nachrichten angeben.

Die Verwendung der `ReplyToQMgr` -Informationen aus der Antwort hat Nachteile:

- Der anfordernde Warteschlangenmanager muss auf eine Antwort auf seine erste Nachricht warten.
- Sie müssen zusätzlichen Code schreiben, um die Informationen zu `ReplyToQMgr` zu suchen und zu verwenden, bevor Sie nachfolgende Nachrichten senden.
- Wenn mehr als eine Route zum WS-Manager vorhanden ist, wird die Reihenfolge der Nachrichten möglicherweise nicht beibehalten.

### **Legen Sie die Option MQ00\_BIND\_ON\_OPEN im Aufruf MQOPEN fest.**

Erzwingen Sie mit der Option `MQ00_BIND_ON_OPEN` für den Aufruf von `MQOPEN`, dass alle Nachrichten an dasselbe Ziel zugestellt werden sollen. Bei Verwendung von Nachrichtengruppen mit Clustern muss entweder `MQ00_BIND_ON_OPEN` oder `MQ00_BIND_ON_GROUP` angegeben werden, um sicherzustellen, dass alle Nachrichten in der Gruppe an demselben Ziel verarbeitet werden.

Wenn Sie eine Warteschlange öffnen und `MQ00_BIND_ON_OPEN` angeben, werden alle Nachrichten, die an diese Warteschlange gesendet werden, gezwungen, an dieselbe Instanz der Warteschlange zu senden. `MQ00_BIND_ON_OPEN` bindet alle Nachrichten an denselben Warteschlangenmanager und auch an denselben Leitweg. Wenn es beispielsweise eine IP-Route und eine NetBIOS-Route zum selben Ziel gibt, wird

eine dieser Routen ausgewählt, wenn die Warteschlange geöffnet wird, und diese Auswahl wird für alle Nachrichten berücksichtigt, die mit der Objektkennung in dieselbe Warteschlange gestellt werden.

Durch Angabe von `MQ00_BIND_ON_OPEN` erzwingen Sie, dass alle Nachrichten an dasselbe Ziel weitergeleitet werden. Daher werden Anwendungen mit Nachrichtenaffinitäten nicht unterbrochen. Wenn die Zieladresse nicht verfügbar ist, bleiben die Nachrichten in der Übertragungswarteschlange, bis sie wieder verfügbar wird.

`MQ00_BIND_ON_OPEN` gilt auch, wenn der Name des Warteschlangenmanagers beim Öffnen einer Warteschlange im Objektdeskriptor angegeben wird. Es kann mehr als eine Route zum angegebenen WS-Manager geben. Es kann z. B. mehrere Netzpfade geben, oder ein anderer WS-Manager hat möglicherweise einen Aliasnamen definiert. Wenn Sie `MQ00_BIND_ON_OPEN` angeben, wird eine Route ausgewählt, wenn die Warteschlange geöffnet wird.

**Anmerkung:** Dies ist das empfohlene Verfahren. Es funktioniert jedoch nicht in einer Multi-Hop-Konfiguration, in der ein WS-Manager einen Aliasnamen für eine Clusterwarteschlange ausweist. Es hilft auch nicht in Situationen, in denen Anwendungen unterschiedliche Warteschlangen in demselben Warteschlangenmanager für verschiedene Nachrichtengruppen verwenden.

Eine Alternative zur Angabe von `MQ00_BIND_ON_OPEN` im Aufruf `MQOPEN` ist die Änderung Ihrer Warteschlangendefinitionen. Geben Sie in Ihren Warteschlangendefinitionen `DEFBIND (OPEN)` an und lassen Sie die Option `DefBind` im Aufruf `MQOPEN` standardmäßig auf `MQ00_BIND_AS_Q_DEF` zu.

### **Legen Sie die Option `MQ00_BIND_ON_GROUP` im Aufruf `MQOPEN` fest.**

Erzwingen Sie mithilfe der Option `MQ00_BIND_ON_GROUP` für den Aufruf von `MQOPEN`, dass alle Nachrichten in einer Gruppe an dasselbe Ziel zugestellt werden sollen. Bei Verwendung von Nachrichtengruppen mit Clustern muss entweder `MQ00_BIND_ON_OPEN` oder `MQ00_BIND_ON_GROUP` angegeben werden, um sicherzustellen, dass alle Nachrichten in der Gruppe an demselben Ziel verarbeitet werden.

Wenn Sie eine Warteschlange öffnen und `MQ00_BIND_ON_GROUP` angeben, erzwingen Sie alle Nachrichten in einer Gruppe, die an diese Warteschlange gesendet werden, um an dieselbe Instanz der Warteschlange gesendet zu werden. `MQ00_BIND_ON_GROUP` bindet alle Nachrichten in einer Gruppe an denselben Warteschlangenmanager und auch an denselben Leitweg. Wenn es beispielsweise eine IP-Route und eine NetBIOS-Route zum selben Ziel gibt, wird eine dieser Routen ausgewählt, wenn die Warteschlange geöffnet wird und diese Auswahl für alle Nachrichten in einer Gruppe berücksichtigt wird, die mit Hilfe der Objektkennung in dieselbe Warteschlange gestellt wird.

Wenn Sie `MQ00_BIND_ON_GROUP` angeben, erzwingen Sie alle Nachrichten in einer Gruppe, die an dasselbe Ziel weitergeleitet werden sollen. Daher werden Anwendungen mit Nachrichtenaffinitäten nicht unterbrochen. Wenn die Zieladresse nicht verfügbar ist, bleiben die Nachrichten in der Übertragungswarteschlange, bis sie wieder verfügbar wird.

`MQ00_BIND_ON_GROUP` gilt auch, wenn der Name des Warteschlangenmanagers beim Öffnen einer Warteschlange im Objektdeskriptor angegeben wird. Es kann mehr als eine Route zum angegebenen WS-Manager geben. Es kann z. B. mehrere Netzpfade geben, oder ein anderer WS-Manager hat möglicherweise einen Aliasnamen definiert. Wenn Sie `MQ00_BIND_ON_GROUP` angeben, wird eine Route ausgewählt, wenn die Warteschlange geöffnet wird.

Damit `MQ00_BIND_ON_GROUP` wirksam ist, müssen Sie die PUT-Option `MQPMO_LOGICAL_ORDER` in `MQPUT` einfügen. Sie können **GroupId** im `MQMD` der Nachricht auf `MQGI_NONE` setzen und Sie müssen folgende Nachrichtenflags in das `MQMD`-Feld **MsgFlags** der Nachrichten einfügen:

- Letzte Nachricht in der Gruppe: `MQMF_LAST_MSG_IN_GROUP`
- Alle anderen Nachrichten in der Gruppe: `MQMF_MSG_IN_GROUP`

Wenn `MQ00_BIND_ON_GROUP` angegeben ist und die Nachrichten nicht gruppiert sind, ist das Verhalten äquivalent zu `MQ00_BIND_NOT_FIXED`.

**Anmerkung:** Dies ist das empfohlene Verfahren, mit dem sichergestellt wird, dass Nachrichten in einer Gruppe an dasselbe Ziel gesendet werden. Sie funktioniert jedoch nicht in einer Multi-Hop-Konfiguration, in der ein Warteschlangenmanager einen Aliasnamen für eine Clusterwarteschlange bewirbt.

Eine Alternative zur Angabe von MQ00\_BIND\_ON\_GROUP im Aufruf MQOPEN ist die Änderung Ihrer Warteschlangendefinitionen. Geben Sie in Ihren Warteschlangendefinitionen DEFBIND (GROUP) an und lassen Sie die Option DefBind im Aufruf MQOPEN standardmäßig auf MQ00\_BIND\_AS\_Q\_DEF zu.

## **Angepasstes Exitprogramm für Clusterauslastung schreiben**

Anstatt Ihre Anwendungen zu ändern, können Sie das Problem der Nachrichtenaffinitäten umgehen, indem Sie ein Exitprogramm für Clusterauslastung schreiben. Das Schreiben eines Exitprogramms zur Clusterauslastung ist nicht einfach und wird nicht empfohlen. Das Programm muss so gestaltet sein, dass es die Affinität erkennt, indem es den Inhalt von Nachrichten überprüft. Nachdem die Affinität erkannt wurde, muss das Programm das Dienstprogramm für die Workload-Management erzwingen, um alle zugehörigen Nachrichten an denselben Warteschlangenmanager weiterzuleiten.

## **Clustering: Bewährte Verfahren**

Cluster stellen einen Mechanismus für die Verbindung von Warteschlangenmanagern bereit. Die in diesem Abschnitt beschriebenen bewährten Verfahren basieren auf Tests und Feedback von Kunden.

Für eine erfolgreiche Clusterkonfiguration sind eine gute Planung und umfassende Kenntnisse der Grundlagen von IBM WebSphere MQ (z. B. ein gutes Anwendungsmanagement und ein durchdachter Netzentwurf) erforderlich. Stellen Sie sicher, dass Sie mit den Informationen in den folgenden Abschnitten vertraut sind, bevor Sie fortfahren.

### **Zugehörige Konzepte**

[Clustering](#)

[Konzepte der übergreifenden Kommunikation](#)

[Wie funktionieren Cluster?](#)

## **Clustering: Besondere Hinweise zu überlappenden Clustern**

In diesem Abschnitt finden Sie Anleitungen für die Planung und Verwaltung von IBM WebSphere MQ-Clustern. Diese Informationen sind ein Leitfaden, der auf Tests und Feedback von Kunden basiert.

### **Clustereigentumsrecht**

Machen Sie sich mit überlappenden Clustern vertraut, bevor Sie die folgenden Informationen lesen. Die erforderlichen Informationen finden Sie unter [„Überlappende Cluster“](#) auf Seite 195 und [„Nachrichtenpfade zwischen Clustern konfigurieren“](#) auf Seite 278.

Wenn Sie ein System konfigurieren und verwalten, das sich aus überlappenden Clustern zusammensetzt, ist es am besten, die folgenden Schritte zu befolgen:

- Obwohl IBM WebSphere MQ-Cluster wie oben beschrieben nur 'lose verbunden' sind, ist es sinnvoll, einen Cluster als eine einzige Verwaltungseinheit zu betrachten. Dieses Konzept wird verwendet, da die Interaktion zwischen Definitionen auf einzelnen Warteschlangenmanagern für das reibungslose Funktionieren des Clusters von entscheidender Bedeutung ist. Beispiel: Bei der Verwendung von auslastungsabhängigen Clusterwarteschlangen ist es wichtig, dass ein einzelner Administrator oder ein Team die vollständige Gruppe möglicher Destinations für Nachrichten versteht, die von den im Cluster verteilten Definitionen abhängig sind. Weitere triviale, Cluster-Sender-/Empfänger-Kanal-Paare müssen in der gesamten Konfiguration kompatibel sein.
- In Anbetracht dieses früheren Konzeptes, bei dem mehrere Cluster (die von separaten Teams/Einzelpersonen verwaltet werden sollen), ist es wichtig, klare Richtlinien für die Verwaltung der Gateway-Warteschlangenmanager zu haben.
- Es ist sinnvoll, überlappende Cluster als einen einzigen Namensbereich zu behandeln: Kanalnamen und WS-Manager-Namen müssen in einem einzigen Cluster eindeutig sein. Die Verwaltung ist viel einfacher, wenn sie in der gesamten Topologie eindeutig ist. Es empfiehlt sich, eine geeignete Namenskonvention zu verwenden. Mögliche Konventionen werden im Abschnitt [„Namenskonventionen für Cluster“](#) auf Seite 194 beschrieben.

- Manchmal ist eine Zusammenarbeit zwischen Verwaltung und Systemmanagement unumgänglich: zum Beispiel die Zusammenarbeit zwischen Organisationen, die eigene Cluster besitzen, die sich überschneiden müssen. Ein klares Verständnis davon, wer welche besitzt, und durchsetzbare Regeln/Konventionen unterstützt das Clustering beim Zusammenfassen von Clustern.

## Überlappende Cluster: Gateways

Im Allgemeinen ist ein einzelner Cluster einfacher zu verwalten als mehrere Cluster. Daher ist die Erstellung einer großen Anzahl von kleinen Clustern (eine für jede Anwendung zum Beispiel) etwas zu vermeiden, die im Allgemeinen.

Sie können jedoch überlappende Cluster implementieren, um Serviceklassen zur Verfügung zu stellen. For example:

- Wenn Sie konzentrische Cluster haben, bei denen der kleinere für Publish/Subscribe vorgesehen ist. Weitere Informationen hierzu finden Sie im Abschnitt [Vorgehensweise bei der Größe von Systemen](#).
- Wenn einige Warteschlangenmanager von verschiedenen Teams verwaltet werden sollen. Weitere Informationen finden Sie im vorherigen Abschnitt „Clustereigentumsrecht“ auf Seite 301.
- Wenn es aus organisatorischer oder geographischer Sicht sinnvoll ist.
- Wenn funktional entsprechende Cluster mit Namensauflösung arbeiten, z. B. bei der Implementierung von SSL oder TLS in einem vorhandenen Cluster.

Es gibt keine Sicherheitsleistung von überlappenden Clustern; die Möglichkeit, dass Cluster, die von zwei verschiedenen Teams verwaltet werden, sich überschneiden, verbindet sich effektiv mit den Teams und der Topologie. Beliebiger:

- Der Name, der in einem solchen Cluster zugänglich gemacht wird, ist für den anderen Cluster zugänglich.
- Name, der in einem Cluster bekannt gemacht wird, kann in der anderen ausgeschrieben werden, um auswählbare Nachrichten abzuzeichnen.
- Nicht beworbenes Objekt in einem WS-Manager neben dem Gateway kann aus allen Clustern aufgelöst werden, in denen das Gateway ein Mitglied ist.

Der Namespace ist die Union der beiden Cluster und muss als einzelner Namensbereich behandelt werden. Daher wird das Eigentumsrecht an einem überlappenden Cluster von allen Administratoren beider Cluster gemeinsam genutzt.

Wenn ein System mehrere Cluster enthält, kann es erforderlich sein, Nachrichten von Warteschlangenmanagern in einem Cluster an Warteschlangen von Warteschlangenmanagern in einem anderen Cluster weiterzuleiten. In dieser Situation müssen die mehreren Cluster in irgendeiner Weise miteinander verbunden sein: Ein gutes Muster, das zu folgen ist, ist die Verwendung von Gateway-WS-Managern zwischen Clustern. Diese Anordnung verhindert die Erstellung eines schwer zu verwaltenden Netzes aus Punkt-zu-Punkt-Kanälen und stellt einen guten Platz bereit, um solche Probleme als Sicherheitsrichtlinien zu verwalten. Es gibt zwei verschiedene Möglichkeiten, diese Anordnung zu erreichen:

1. Platzieren Sie einen (oder mehrere) Warteschlangenmanager in beiden Clustern unter Verwendung einer zweiten Clusterempfängerdefinition. Diese Anordnung umfasst weniger Verwaltungsdefinitionen, bedeutet aber, wie bereits erwähnt, bedeutet, dass das Eigentumsrecht an einem überlappenden Cluster von allen Administratoren beider Cluster gemeinsam genutzt wird.
2. Pair einen WS-Manager in Cluster mit einem Warteschlangenmanager in Cluster teo unter Verwendung traditioneller Punkt-zu-Punkt-Kanäle.

In einem dieser Fälle können verschiedene Tools verwendet werden, um den Datenverkehr entsprechend weiterzuleiten. Insbesondere können Aliasnamen von Warteschlangen oder Warteschlangenmanagern verwendet werden, um in den anderen Cluster zu steuern. Ein Alias eines Warteschlangenmanagers mit leerer **QMNAME**-Eigenschaft sorgt für einen erneuten Lastausgleich, wo er gewünscht ist.

### Zugehörige Konzepte

[„Namenskonventionen für Cluster“ auf Seite 194](#)



Sie sollten die Benennung von Warteschlangenmanagern in demselben Cluster unter Verwendung einer Namenskonvention berücksichtigen, die den Cluster angibt, zu dem der Warteschlangenmanager gehört. Verwenden Sie eine ähnliche Namenskonvention für Kanalnamen und erweitern Sie diese, um die Kanalmerkmale zu beschreiben.

## Clustering: Aspekte des Topologiedesigns

Dieser Abschnitt enthält eine Anleitung zur Planung und Verwaltung von IBM WebSphere MQ-Clustern. Diese Informationen sind ein Leitfaden, der auf Tests und Feedback von Kunden basiert.

Wenn Sie darüber nachdenken, wo Benutzeranwendungen und interne Verwaltungsprozesse im Voraus lokalisiert werden sollen, können viele Probleme entweder vermieden oder zu einem späteren Zeitpunkt minimiert werden. Dieses Thema enthält Informationen zu Designentscheidungen, die die Leistung verbessern können, und vereinfachen die Verwaltungstasks als Clusterskalierungen.

- [„Leistung der Clustering-Infrastruktur“ auf Seite 303](#)
- [„Vollständige Repositorys“ auf Seite 304](#)
- [„Sollen Anwendungen Warteschlangen in vollständigen Repositorys verwenden?“ auf Seite 305](#)
- [„Kanaldefinitionen verwalten“ auf Seite 305](#)
- [„Lastausgleich über mehrere Kanäle“ auf Seite 306](#)

## Leistung der Clustering-Infrastruktur

Wenn eine Anwendung versucht, eine Warteschlange auf einem Warteschlangenmanager in einem Cluster zu öffnen, registriert der Warteschlangenmanager sein Interesse an den vollständigen Repositorys für diese Warteschlange, so dass er lernen kann, wo sich die Warteschlange im Cluster befindet. Alle Aktualisierungen an der Warteschlangenposition oder -konfiguration werden automatisch von den vollständigen Repositorys an den interessierten WS-Manager gesendet. Diese Registrierung von Interesse wird intern als Subskription bezeichnet (diese Subskriptionen sind nicht mit den IBM WebSphere MQ-Subskriptionen identisch, die für Publish/Subscribe-Messaging in IBM WebSphere MQ verwendet werden).

Alle Informationen zu einem Cluster durchlaufen jedes vollständige Repository. Vollständige Repositorys werden daher immer in einem Cluster für den Verwaltungsnachrichtenverkehr verwendet. Die hohe Auslastung der Systemressourcen bei der Verwaltung dieser Subskriptionen und die Übertragung dieser Nachrichten und die daraus resultierenden Konfigurationsnachrichten können zu einer erheblichen Auslastung der Clustering-Infrastruktur führen. Es gibt eine Reihe von Faktoren, die zu berücksichtigen sind, wenn sichergestellt wird, dass diese Last wo immer möglich verstanden und minimiert wird:

- Je mehr einzelne WS-Manager eine Clusterwarteschlange verwenden, umso mehr Subskriptionen sind im System vorhanden, wodurch der Verwaltungsaufwand bei Änderungen größer ist und interessierte Subskribenten benachrichtigt werden müssen, insbesondere auf den vollständigen WS-Managern des Repositorys. Eine Möglichkeit, den unnötigen Datenverkehr und die Auslastung des gesamten Repositorys zu minimieren, besteht darin, ähnliche Anwendungen (d. a. die Anwendungen, die mit denselben Warteschlangen arbeiten) mit einer kleineren Anzahl von Warteschlangenmanagern zu verbinden.
- Neben der Anzahl der Subskriptionen im System, die sich auf die Leistung auswirken, kann sich die Änderungsrate in der Konfiguration von Clusterobjekten auf die Leistung auswirken, z. B. die häufige Änderung einer Clusterwarteschlangenkonfiguration.
- Wenn ein Warteschlangenmanager Mitglied mehrerer Cluster ist (d. B. er Teil eines überlappenden Clustersystems ist), wird jedes Interesse, das in einer Warteschlange erstellt wird, zu einer Subskription für jeden Cluster, zu dem er gehört, auch dann, wenn dieselben Warteschlangenmanager die vollständigen Repositorys für mehr als einen der Cluster sind. Diese Anordnung erhöht die Belastung des Systems und ist ein Grund, zu überlegen, ob mehrere überlappende Cluster erforderlich sind, und nicht nur ein einzelner Cluster.
- Anwendungsnachrichtenverkehr (d. h. Nachrichten, die von IBM WebSphere MQ-Anwendungen an die Clusterwarteschlangen gesendet werden), gehen nicht über die vollständigen Repositorys, um die Zielwarteschlangenmanager zu erreichen. Dieser Nachrichtenverkehr wird direkt zwischen dem Warteschlangenmanager, in dem die Nachricht in den Cluster eintritt, und dem Warteschlangenmanager, in dem die Clusterwarteschlange vorhanden ist, gesendet. Daher ist es nicht erforderlich, hohe Aufwand-

mengen an Anwendungsnachrichtenverkehr in Bezug auf die vollständigen WS-Manager-Repositorys zu berücksichtigen, es sei denn, die vollständigen WS-Manager-WS-Manager sind einer der beiden genannten WS-Manager. Aus diesem Grund wird empfohlen, vollständige WS-Manager-Repositorys nicht für Anwendungsnachrichtenverkehr in Clustern zu verwenden, in denen die Clustering-Infrastrukturbelastung von Bedeutung ist.

## Vollständige Repositorys

Ein Repository ist eine Zusammenstellung von Informationen über die Warteschlangenmanager, die zu einem Cluster gehören. Ein Warteschlangenmanager, der über einen vollständigen Satz von Informationen über jeden Warteschlangenmanager im Cluster verfügt, besitzt ein vollständiges Repository. Weitere Informationen zu vollständigen Repositorys und Teilrepositorys finden Sie unter [„Cluster-Repository“](#) auf Seite 177.

Vollständige Repositorys müssen auf Servern eingesetzt werden, die zuverlässig und so hoch wie möglich verfügbar sind und Single Points of Failure vermieden werden müssen. Das Clusterdesign muss immer über zwei vollständige Repositorys verfügen. Wenn ein vollständiges Repository nicht vorhanden ist, kann der Cluster trotzdem ausgeführt werden.

Details zu Aktualisierungen für Clusterressourcen, die von einem Warteschlangenmanager in einem Cluster erstellt werden, z. B. Clusterwarteschlangen, werden von diesem Warteschlangenmanager an die meisten in diesem Cluster an zwei vollständige Repositorys gesendet (oder zu einem Cluster, wenn nur ein vollständiger Warteschlangenmanager im Cluster vorhanden ist). Diese vollständigen Repositorys enthalten die Informationen und geben sie an alle WS-Manager im Cluster weiter, die ein Interesse daran haben (d. a. sie abonnieren). Um sicherzustellen, dass jedes Member des Clusters über eine Up-to-Data-Ansicht der Clusterressourcen verfügt, muss jeder WS-Manager in der Lage sein, mit mindestens einem vollständigen WS-Manager-Repository gleichzeitig zu kommunizieren.

Wenn ein Warteschlangenmanager aus irgendeinem Grund nicht mit vollständigen Repositorys kommunizieren kann, kann er im Cluster abhängig von seiner bereits zwischengespeicherten Informationsstufe für einen Zeitraum weiter funktionieren, aber es sind keine neuen Aktualisierungen oder der Zugriff auf zuvor nicht verwendete Clusterressourcen verfügbar.

Aus diesem Grund müssen Sie das Ziel haben, die beiden vollständigen Repositorys immer verfügbar zu halten. Diese Anordnung bedeutet jedoch nicht, dass extreme Maßnahmen ergriffen werden müssen, da der Cluster für eine kurze Zeit ohne vollständiges Repository ausreichend funktioniert.

Es gibt einen weiteren Grund, dass ein Cluster über zwei vollständige WS-Manager-Repository-WS-Manager verfügen muss, die nicht die Verfügbarkeit von Clusterinformationen sind: Dieser Grund besteht darin, sicherzustellen, dass die Clusterinformationen, die im vollständigen Repository-Cache gespeichert sind, an zwei Stellen zu Wiederherstellungszwecken vorhanden sind. Wenn nur ein vollständiges Repository vorhanden ist und seine Informationen zum Cluster verloren gehen, ist ein manueller Eingriff auf alle WS-Manager im Cluster erforderlich, damit der Cluster wieder funktionieren kann. Wenn jedoch zwei vollständige Repositorys vorhanden sind, weil die Informationen immer in zwei vollständigen Repositorys veröffentlicht und abonniert werden, kann das Repository für fehlgeschlagene vollständige Repositorys mit dem Minimum an Aufwand wiederhergestellt werden.

- Es ist möglich, die Verwaltung von WS-Managern mit vollem Repository in einem zwei vollständigen Repository-Cluster-Design auszuführen, ohne die Benutzer des Clusters zu beeinträchtigen: Der Cluster funktioniert weiterhin mit nur einem Repository, so dass die Repositorys nach unten gebracht, die Wartung angewendet und ein weiteres Mal wieder gesichert werden kann. Selbst wenn ein Ausfall im zweiten vollständigen Repository vorhanden ist, wird die Ausführung von Anwendungen für mindestens drei Tage nicht beeinträchtigt.
- Es sei denn, es gibt einen guten Grund für die Verwendung eines dritten Repositorys, wie z. B. die Verwendung eines geographisch lokalen vollständigen Repositorys aus geographischen Gründen, die Verwendung der beiden Repository-Designs. Die Verwendung von drei vollständigen Repositorys bedeutet, dass Sie nie wissen, welche beiden derzeit verwendet werden, und es kann zu Verwaltungsproblemen kommen, die durch die Interaktionen zwischen mehreren Workload-Management-Parametern verursacht werden. Es wird nicht empfohlen, mehr als zwei vollständige Repositorys zu verwenden.



- Wenn Sie nach wie vor eine bessere Verfügbarkeit benötigen, sollten Sie die vollständigen WS-Manager-Repositorys als Multi-Instanz-Warteschlangenmanager oder plattformspezifische Hochverfügbarkeitsunterstützung verwenden, um die Verfügbarkeit zu verbessern.
- Sie müssen alle vollständigen WS-Manager-Repository-Warteschlangenmanager vollständig mit manuell definierten Clustersenderkanälen verbinden. Es ist besonders darauf zu achten, dass der Cluster aus einem vertretbaren Grund mehr als zwei vollständige Repositorys hat. In dieser Situation ist es oft möglich, einen oder mehrere Kanäle zu verpassen und dafür nicht sofort erkennbar zu sein. Wenn keine vollständige Verbindung auftritt, treten häufig Probleme bei der Diagnose von Problemen auf. Sie sind schwer zu diagnostizieren, da einige vollständige Repositorys nicht alle Repositorydaten enthalten und daher in Abhängigkeit von den vollständigen Repositorys, zu denen sie eine Verbindung herstellen, zu Warteschlangenmanagern im Cluster mit unterschiedlichen Sichten des Clusters führt.

## Sollen Anwendungen Warteschlangen in vollständigen Repositorys verwenden?

Ein vollständiges Repository ist in der meisten Hinsicht genau wie jeder andere Warteschlangenmanager, und es ist daher möglich, Anwendungswarteschlangen im vollständigen Repository zu hosten und Anwendungen direkt mit diesen WS-Managern zu verbinden. Sollen Anwendungen Warteschlangen in vollständigen Repositorys verwenden?

Die allgemein akzeptierte Antwort ist "Nein". Obwohl diese Konfiguration möglich ist, bevorzugen viele Kunden, diese Warteschlangenmanager für die Verwaltung des gesamten Repository-Cluster-Caches zu halten. Punkte, die bei der Entscheidung für eine der beiden Optionen zu berücksichtigen sind, werden hier beschrieben, aber letztendlich muss die Clusterarchitektur den besonderen Anforderungen der Umgebung gerecht werden.

- **Upgrades:** Um neue Clusterfunktionen in neuen Releases von IBM WebSphere MQ nutzen zu können, müssen normalerweise zuerst ein Upgrade der Warteschlangenmanager mit vollständigem Repository in diesem Cluster durchführen. Wenn eine Anwendung im Cluster neue Funktionen verwenden möchte, kann es nützlich sein, die vollständigen Repositorys (und eine Teilmenge von Teilrepositorys) zu aktualisieren, ohne eine Reihe von kolokationsfähigen Anwendungen testen zu müssen.
- **Wartung:** Auf ähnliche Weise, wenn Sie eine dringende Wartung auf die vollständigen Repositorys anwenden müssen, können sie mit dem Befehl **REFRESH** erneut gestartet oder aktualisiert werden, ohne Anwendungen zu berühren.
- **Leistung:** Wenn Cluster wachsen und die Anforderungen an die vollständige Repository-Cluster-Cache-Wartung größer werden, verringert sich die Gefahr, dass die Anwendungsleistung durch Konkurrenzsituationen getrennt wird, die die Anwendungsleistung beeinträchtigen, da die Systemressourcen in Konflikt stehen.
- **Hardwarevoraussetzungen:** In der Regel müssen vollständige Repositorys nicht leistungsfähig sein, z. B. ist ein einfacher UNIX-Server mit einer guten Erwartung an die Verfügbarkeit ausreichend. Alternativ dazu muss bei sehr großen oder sich ständig verändernden Clustern die Leistung des gesamten Repository-Computers berücksichtigt werden.
- **Softwarevoraussetzungen:** Anforderungen sind in der Regel der Hauptgrund für die Auswahl von Anwendungs-Warteschlangen in einem vollständigen Repository. In einem kleinen Cluster kann die Kollokation eine Voraussetzung für weniger WS-Manager/Server über alle bedeuten.

## Kanaldefinitionen verwalten

Selbst in einem einzigen Cluster können mehrere Kanaldefinitionen vorhanden sein, die mehrere Routen zwischen zwei Warteschlangenmanagern enthalten.

Manchmal sind parallele Kanäle in einem einzigen Cluster von Vorteil. Jedoch sollte diese Designentscheidung gründlich durchdacht werden. Abgesehen von der Komplexität dieses Designs werden die Kanäle unter Umständen nicht sinnvoll ausgelastet, was im Sinne der Leistungsoptimierung nachteilig wäre. Diese Situation tritt auf, da beim Testen normalerweise viele Nachrichten mit einer konstanten Rate gesendet werden, so dass die parallelen Kanäle vollständig verwendet werden. Bei real-world-Bedingungen eines nicht konstanten Nachrichtenstroms bewirkt der Lastausgleichsalgorithmus jedoch, dass die Leistung sinkt, wenn der Nachrichtenfluss von Kanal zu Kanal umgeschaltet wird.

Wenn ein Warteschlangenmanager mehreren Clustern angehört, besteht die Möglichkeit, eine einzelne Kanaldefinition mit einer Clusternamensliste zu verwenden, anstatt einen separaten CLUSRCVR -Kanal für jeden Cluster zu definieren. Diese Konfiguration kann jedoch später zu Verwaltungsproblemen führen, zum Beispiel, wenn SSL nur auf einen Cluster, nicht aber auf einen zweiten Cluster angewendet werden soll. Es ist daher vorzuziehen, separate Definitionen zu erstellen, und die in „Namenskonventionen für Cluster“ auf Seite 194 vorgeschlagene Namenskonvention unterstützt dies.

## Lastausgleich über mehrere Kanäle

Diese Informationen sind als ein fortgeschrittenes Verständnis des Subjektes gedacht. Eine grundlegende Erläuterung dieses Themas (das verstanden werden muss, bevor die Informationen hier verwendet werden) finden Sie in „Cluster für Workload-Management verwenden“ auf Seite 284, Lastausgleich und Algorithmus für das Cluster-Workload-Management.

Der Algorithmus für die Clusterauslastungsverwaltung stellt eine große Gruppe von Tools zur Verfügung, aber sie dürfen nicht alle zusammen verwendet werden, ohne dass die Funktionsweise und die Interaktion vollständig verstanden werden müssen. Es ist möglicherweise nicht sofort ersichtlich, wie wichtig die Kanäle für den Lastausgleich sind: Der Algorithmus für die Verarbeitung des Workload-Managements verhält sich so, als ob mehrere Cluster-Kanäle zu einem Warteschlangenmanager, der Eigner einer Clusterwarteschlange ist, als mehrere Instanzen dieser Warteschlange behandelt werden. Dieser Prozess wird im folgenden Beispiel ausführlicher erläutert:

1. Es gibt zwei Warteschlangenmanager, die eine Warteschlange in einem Cluster hosten: QM1 und QM2.
2. Es gibt fünf Clusterempfängerkanäle für QM1.
3. Es gibt nur einen Clusterempfängerkanal für QM2.
4. Wenn **MQPUT** oder **MQOPEN** on QM3 eine Instanz auswählt, ist es fünf Mal wahrscheinlicher, dass der Algorithmus die Nachricht an QM1 sendet als an QM2.
5. Die Situation in Schritt 4 tritt auf, weil der Algorithmus sechs Optionen zur Auswahl vorsieht (5 + 1) und Round-Robin über alle fünf Kanäle auf QM1 und den einzelnen Kanal auf QM2.

Eine weitere kluge Funktion besteht darin, dass IBM WebSphere MQ selbst beim Einreihen von Nachrichten in eine Clusterwarteschlange, für die auf dem lokalen Warteschlangenmanager zufällig eine Instanz konfiguriert ist, den Status des lokalen Clusterempfängerkanals verwendet, um zu entscheiden, ob Nachrichten in die lokale Instanz der Warteschlange oder in ferne Instanzen der Warteschlange gestellt werden sollen. In diesem Szenario gilt Folgendes:

1. Beim Einreihen von Nachrichten sieht der Algorithmus für die Auslastungsverwaltung nicht die einzelnen Clusterwarteschlangen an, sondern die Clusterkanäle, die diese Ziele erreichen können.
2. Um die lokalen Zieladressen zu erreichen, werden die lokalen Empfängerkanäle in diese Liste aufgenommen (obwohl sie nicht zum Senden der Nachricht verwendet werden).
3. Wenn ein lokaler Empfängerkanal gestoppt wird, bevorzugt der Workload-Management-Algorithmus eine alternative Instanz bevorzugt, wenn die CLUSRCVR nicht gestoppt ist. Wenn mehrere lokale CLUSRCVR-Instanzen für das Ziel vorhanden sind und mindestens eine Instanz nicht gestoppt ist, bleibt die lokale Instanz berechtigt.

## Clustering: Anwendungsisolation mit mehreren Clusterübertragungswarteschlangen

Sie können die Nachrichtenflüsse zwischen Warteschlangenmanagern in einem Cluster isolieren. Sie können Nachrichten, die von verschiedenen Clustersenderkanälen transportiert werden, in verschiedene Clusterübertragungswarteschlangen stellen. Sie können den Ansatz in einem einzelnen Cluster oder mit überlappenden Clustern verwenden. Das Thema enthält Beispiele und einige bewährte Verfahren, die Sie bei der Auswahl eines zu verwendenden Ansatzes führen.

Wenn Sie eine Anwendung implementieren, haben Sie eine Auswahl darüber, welche IBM WebSphere MQ -Ressourcen sie gemeinsam mit anderen Anwendungen nutzt und welche Ressourcen sie nicht gemeinsam nutzt. Es gibt eine Reihe von Typen von Ressourcen, die gemeinsam genutzt werden können, wobei die Haupttypen der Server selbst, der Warteschlangenmanager, die Kanäle und die Warteschlangen

sind. Sie können Anwendungen mit weniger gemeinsam genutzten Ressourcen konfigurieren. Sie können separate Warteschlangen, Kanäle, Warteschlangenmanager oder sogar Server für einzelne Anwendungen zuordnen. Wenn Sie dies tun, wird die Gesamtsystemkonfiguration größer und komplexer. Die Verwendung von IBM WebSphere MQ-Clustern reduziert die Komplexität der Verwaltung von mehr Servern, Warteschlangenmanagern, Warteschlangen und Kanälen, führt aber eine weitere gemeinsam genutzte Ressource ein, die Cluster-Übertragungswarteschlange `SYSTEM.CLUSTER.TRANSMIT.QUEUE`.

Abbildung 55 auf Seite 308 ist ein Ausschnitt aus einer großen IBM WebSphere MQ -Implementierung, der die Bedeutung der gemeinsamen Nutzung von `SYSTEM.CLUSTER.TRANSMIT.QUEUE` veranschaulicht. Im Diagramm ist die Anwendung `Client App` mit dem Warteschlangenmanager `QM2` im Cluster `CL1` verbunden. Eine Nachricht von `Client App` wird von der Anwendung `Server App` verarbeitet. Die Nachricht wird von `Server App` aus der Clusterwarteschlange `Q1` auf dem Warteschlangenmanager `QM3` in `CLUSTER2` abgerufen. Da sich die Client- und Serveranwendungen nicht in demselben Cluster befinden, wird die Nachricht vom Gateway-Warteschlangenmanager `QM1` übertragen.

Der normale Weg zur Konfiguration eines Cluster-Gateways besteht darin, den Gateway-Warteschlangenmanager zu einem Mitglied aller Cluster zu machen. Auf dem Gateway-WS-Manager werden Clusteraliasnamen für Clusterwarteschlangen in allen Clustern definiert. Die Aliasnamen für Clusterwarteschlangen sind in allen Clustern verfügbar. Nachrichten, die an die Clusterwarteschlangenaliasnamen gestellt werden, werden über den Gateway-Warteschlangenmanager an ihr korrektes Ziel weitergeleitet. Der Gateway-Warteschlangenmanager reiht Nachrichten, die an die Aliase der Clusterwarteschlangen gesendet werden, in die allgemeine `SYSTEM.CLUSTER.TRANSMIT.QUEUE` unter `QM1` ein.

Die Hub-und Spoke-Architektur erfordert alle Nachrichten zwischen Clustern, die über den Gateway-Warteschlangenmanager übergeben werden. Das Ergebnis ist, dass alle Nachrichten durch die Warteschlange des einzelnen Clusters auf `QM1`, `SYSTEM.CLUSTER.TRANSMIT.QUEUE` fließen.

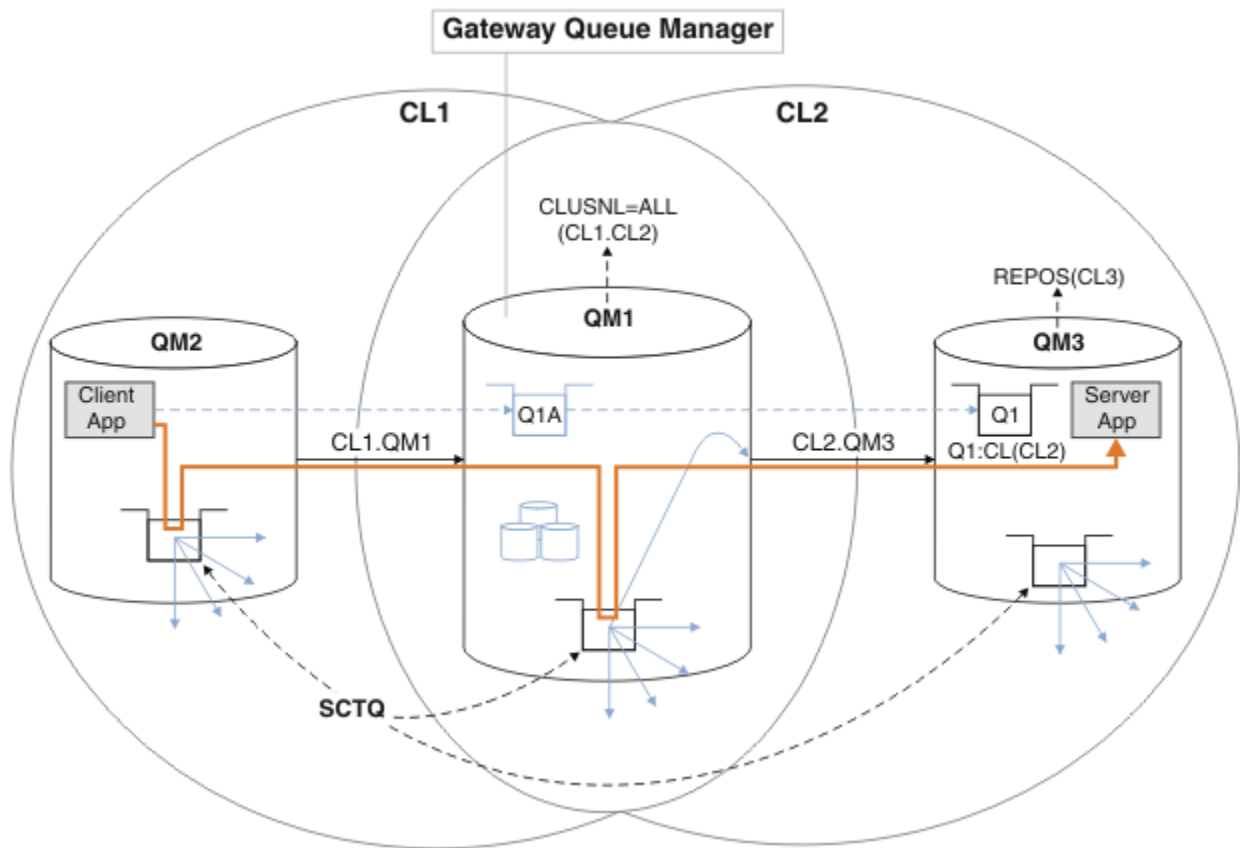
Aus einer Leistungsperspektive ist eine einzelne Warteschlange kein Problem. Eine allgemeine Übertragungswarteschlange stellt in der Regel keinen Leistungsengpass dar. Der Nachrichtendurchsatz auf dem Gateway wird weitgehend durch die Leistung der Kanäle bestimmt, die eine Verbindung zu ihm herstellen. Der Durchsatz wird in der Regel nicht von der Anzahl der Warteschlangen oder der Anzahl der Nachrichten in den Warteschlangen, die die Kanäle verwenden, beeinflusst.

Aus einigen anderen Perspektiven hat die Verwendung einer einzelnen Übertragungswarteschlange für mehrere Anwendungen Nachteile:

- Sie können den Fluss von Nachrichten nicht in ein Ziel vom Nachrichtenfluss zu einem anderen Ziel eingrenzen. Sie können die Speicherung von Nachrichten nicht trennen, bevor sie weitergeleitet werden, selbst wenn sich die Ziele in verschiedenen Clustern auf verschiedenen Warteschlangenmanagern befinden.

Wenn eine Cluster-Destination nicht mehr verfügbar ist, werden Nachrichten für diese Zieladresse in der einzelnen Übertragungswarteschlange, und schließlich füllen sie die Nachrichten aus. Sobald die Übertragungswarteschlange voll ist, stoppt sie Nachrichten, die in die Übertragungswarteschlange für ein beliebtes Clusterziel gestellt werden.

- Es ist nicht einfach, die Übertragung von Nachrichten an verschiedene Cluster-Destinations zu überwachen. Alle Nachrichten befinden sich in der einzelnen Übertragungswarteschlange. Wenn Sie die Länge der Übertragungswarteschlange anzeigen, wird wenig darüber informiert, ob Nachrichten an alle Zieladressen übertragen werden.



**Anmerkung:** Die Pfeile in [Abbildung 55](#) auf Seite 308 und die folgenden Abbildungen sind von unterschiedlichen Typen. Feste Pfeile stellen Nachrichtenflüsse dar. Bei den Beschriftungen auf festen Pfeilen handelt es sich um Nachrichtenkanalnamen. Die grauen ausgefüllten Pfeile sind potenzielle Nachrichtenflüsse von SYSTEM . CLUSTER . TRANSMIT . QUEUE auf Clustersenderkanälen. Schwarze gestrichelte Linien verbinden Beschriftungen mit ihren Zielen. Graue gestrichelte Pfeile sind Referenzen, z. B. von einem MQOPEN Aufruf von Client App an die Clusteraliaswarteschlangendefinition Q1A.

*Abbildung 55. In Hub- und Spoke-Architektur mit IBM WebSphere MQ-Clustern implementierte Client/Server-Anwendung*

In [Abbildung 55](#) auf Seite 308 öffnen Clients von Server App die Warteschlange Q1A. Nachrichten werden unter QM2 in SYSTEM . CLUSTER . TRANSMIT . QUEUE eingereicht, unter QM1 in SYSTEM . CLUSTER . TRANSMIT . QUEUE übertragen und anschließend unter QM3 in Q1 übertragen, wo sie von der Anwendung Server App empfangen werden.

Die Nachricht von Client App wird über die Clusterübertragungswarteschlangen des Systems durch QM2 und QM1 übergeben. In [Abbildung 55](#) auf Seite 308 besteht das Ziel darin, den Nachrichtenfluss auf dem Gateway-Warteschlangenmanager von der Clientanwendung zu isolieren, damit seine Nachrichten nicht in SYSTEM . CLUSTER . TRANSMIT . QUEUE gespeichert werden. Sie können Flüsse auf einem der anderen Clusterwarteschlangenmanager isolieren. Sie können auch Flüsse in die andere Richtung isolieren, zurück an den Client. Um die Beschreibungen der Lösungen kurz zu halten, betrachten die Beschreibungen nur einen einzigen Nachrichtenfluss von der Clientanwendung.

## Lösungen für die Isolierung des Clusternachrichtenverkehrs auf einem Cluster-Gateway-Warteschlangenmanager

Eine Möglichkeit, das Problem zu lösen, besteht darin, WS-Manager-Aliasnamen oder ferne Warteschlangendefinitionen zu verwenden, um eine Brücke zwischen Clustern zu schlagen. Erstellen Sie eine Definition einer fernen Clusterwarteschlange, eine Übertragungswarteschlange und einen Kanal, um jeden Nachrichtenfluss auf dem Gateway-Warteschlangenmanager zu trennen; siehe [„Definition einer fernen](#)

Warteschlange hinzufügen, um Nachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet wurden auf Seite 216.

Ab Version 7.5 sind Cluster-WS-Manager nicht auf eine einzelne Cluster-Übertragungswarteschlange beschränkt. Sie haben zwei Möglichkeiten:

1. Definieren Sie zusätzliche Clusterübertragungswarteschlangen manuell und definieren Sie, welche Clustersenderkanäle Nachrichten aus jeder Übertragungswarteschlange übertragen (siehe „Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde“ auf Seite 219).
2. Ermöglichen Sie dem WS-Manager, zusätzliche Clusterübertragungswarteschlangen automatisch zu erstellen und zu verwalten. Sie definiert für jeden Clustersenderkanal eine andere Clusterübertragungswarteschlange (siehe „Ändern der Standardeinstellung in separate Clusterübertragungswarteschlangen, um den Nachrichtendatenverkehr zu isolieren“ auf Seite 241 ).

Sie können manuell definierte Clusterübertragungswarteschlangen für einige Clustersenderkanäle kombinieren, wobei der Warteschlangenmanager den Rest verwaltet. Die Kombination von Übertragungswarteschlangen ist der Ansatz in „Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde“ auf Seite 219. In dieser Lösung verwenden die meisten Nachrichten zwischen Clustern SYSTEM . CLUSTER . TRANSMIT . QUEUE allgemein. Eine Anwendung ist kritisch, und alle ihre Nachrichtenflüsse werden von anderen Nachrichtenflüssen isoliert, indem eine manuell definierte Clusterübertragungswarteschlange verwendet wird.

Die Konfiguration in „Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde“ auf Seite 219 ist begrenzt. Der Nachrichtendatenverkehr, der in eine Clusterwarteschlange auf demselben WS-Manager in demselben Cluster wie eine andere Clusterwarteschlange läuft, wird nicht getrennt. Sie können den Nachrichtenverkehr auf einzelne Warteschlangen verteilen, indem Sie die Definitionen ferner Warteschlangen verwenden, die Teil der verteilten Steuerung von Warteschlangen sind. Bei Clustern, die mehrere Clusterübertragungswarteschlangen verwenden, können Sie den Nachrichtenverkehr trennen, der zu verschiedenen Clustersenderkanälen führt. Mehrere Clusterwarteschlangen im selben Cluster teilen sich auf demselben Warteschlangenmanager einen Clustersenderkanal gemeinsam. Nachrichten für diese Warteschlangen werden in derselben Übertragungswarteschlange gespeichert, bevor sie vom Gateway-WS-Manager weitergeleitet werden. In der Konfiguration in „Cluster und Cluster-Übertragungswarteschlange hinzufügen, um den Datenverkehr der Clusternachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet werden“ auf Seite 224 wird die Einschränkung seitlich abgestuft, indem ein weiterer Cluster hinzugefügt und der Warteschlangenmanager und die Clusterwarteschlange zu einem Mitglied des neuen Clusters gemacht werden. Der neue Warteschlangenmanager kann der einzige WS-Manager im Cluster sein. Sie können dem Cluster weitere Warteschlangenmanager hinzufügen und denselben Cluster verwenden, um Clusterwarteschlangen auf diesen Warteschlangenmanagern zu isolieren.

### **Zugehörige Konzepte**

„Zugriffssteuerung und mehrere Clusterübertragungswarteschlangen“ auf Seite 173

Wählen Sie zwischen drei Prüfmodi aus, wenn eine Anwendung Nachrichten in ferne Clusterwarteschlangen einreicht. Die Modi werden über Remotezugriff auf die Clusterwarteschlange überprüft, lokal auf SYSTEM . CLUSTER . TRANSMIT . QUEUE geprüft oder auf lokale Profile für die Clusterwarteschlange oder den Cluster-WS-Manager überprüft.

„Clustering: Besondere Hinweise zu überlappenden Clustern“ auf Seite 301

In diesem Abschnitt finden Sie Anleitungen für die Planung und Verwaltung von IBM WebSphere MQ -Clustern. Diese Informationen sind ein Leitfaden, der auf Tests und Feedback von Kunden basiert.

„Clusterübertragungswarteschlangen und Clustersenderkanäle“ auf Seite 185

Nachrichten zwischen Clustering-WS-Managern werden in Clusterübertragungswarteschlangen gespeichert und von Clustersenderkanälen weitergeleitet.

„Überlappende Cluster“ auf Seite 195

Überlappende Cluster bieten zusätzliche Verwaltungsfunktionen. Verwenden Sie Namenslisten, um die Anzahl der Befehle zu reduzieren, die für die Verwaltung von überlappenden Clustern erforderlich sind.

## Zugehörige Tasks

Berechtigung zum Einreihen von Nachrichten in ferne Clusterwarteschlangen berechtigen

„Definition einer fernen Warteschlange hinzufügen, um Nachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet wurden“ auf Seite 216

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine ferne Definition einer Clusterwarteschlange und einen separaten Senderkanal und eine separate Übertragungswarteschlange.

„Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde“ auf Seite 219

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine zusätzliche Clusterübertragungswarteschlange, um den Nachrichtenverkehr auf einen einzelnen Warteschlangenmanager in einem Cluster zu trennen.

„Cluster und Cluster-Übertragungswarteschlange hinzufügen, um den Datenverkehr der Clusternachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet werden“ auf Seite 224

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet einen zusätzlichen Cluster, um die Nachrichten in einer bestimmten Clusterwarteschlange zu isolieren.

„Ändern der Standardeinstellung in separate Clusterübertragungswarteschlangen, um den Nachrichten-datenverkehr zu isolieren“ auf Seite 241

Sie können die Standardweise ändern, in der ein WS-Manager Nachrichten für eine Clusterwarteschlange oder ein Topic in einer Übertragungswarteschlange speichert. Wenn Sie den Standardwert ändern, können Sie Clusternachrichten auf einem Gateway-Warteschlangenmanager isolieren.

„Erstellen von zwei überlappenden Clustern mit einem Gateway-Warteschlangenmanager“ auf Seite 235

Befolgen Sie die Anweisungen in der Task, um überlappende Cluster mit einem Gateway-Warteschlangenmanager zu erstellen. Verwenden Sie die Cluster als Ausgangspunkt für die folgenden Beispiele, um Nachrichten in einer Anwendung von Nachrichten an andere Anwendungen in einem Cluster zu isolieren.

„Nachrichtenpfade zwischen Clustern konfigurieren“ auf Seite 278

Verbinden Sie Cluster unter Verwendung eines Gateway-Warteschlangenmanagers. Stellen Sie Warteschlangen oder Warteschlangenmanager für alle Cluster sichtbar, indem Sie Clusterwarteschlangen- oder Cluster-WS-Manager-Aliasnamen auf dem Gateway-Warteschlangenmanager definieren.

„Clustering: Planung der Konfiguration von Clusterübertragungswarteschlangen“ auf Seite 310

Sie werden durch die Auswahl von Clusterübertragungswarteschlangen geführt. Sie können eine gemeinsame Standardwarteschlange, separate Standardwarteschlangen oder manuell definierte Warteschlangen konfigurieren. Die Konfiguration mehrerer Clusterübertragungswarteschlangen gilt für andere Plattformen als z/OS.

## Zugehörige Verweise

„Sicherheit“ auf Seite 467

Verwenden Sie die Zeilengruppe `Security` in der Datei `qm.ini`, um Optionen für den OAM (Object Authority Manager) anzugeben.

`setmqaut`

## Clustering: Planung der Konfiguration von Clusterübertragungswarteschlangen

Sie werden durch die Auswahl von Clusterübertragungswarteschlangen geführt. Sie können eine gemeinsame Standardwarteschlange, separate Standardwarteschlangen oder manuell definierte Warteschlangen konfigurieren. Die Konfiguration mehrerer Clusterübertragungswarteschlangen gilt für andere Plattformen als z/OS.

## Vorbereitende Schritte

Lesen Sie den Abschnitt „[Art der zu verwendenden Clusterübertragungswarteschlange auswählen](#)“ auf Seite 314.

### Informationen zu diesem Vorgang

Wenn Sie planen, wie ein Warteschlangenmanager für die Auswahl einer Clusterübertragungswarteschlange konfiguriert werden soll, können Sie einige Optionen auswählen.

1. Was ist die Standard-Cluster-Übertragungswarteschlange für die Übertragung von Clusternachrichten?
  - a. Eine gemeinsame Clusterübertragungswarteschlange `SYSTEM.CLUSTER.TRANSMIT.QUEUE`.
  - b. Trennen Sie die Clusterübertragungswarteschlangen voneinander. Der WS-Manager verwaltet die separaten Clusterübertragungswarteschlangen. Sie erstellt sie als permanent-dynamische Warteschlangen aus der Modellwarteschlange `SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE`. Er erstellt für jeden Clustersenderkanal, den er verwendet, eine Clusterübertragungswarteschlange.
2. Für die Clusterübertragungswarteschlangen, die Sie manuell erstellen möchten, haben Sie die folgenden beiden Möglichkeiten:
  - a. Definieren Sie eine separate Übertragungswarteschlange für jeden Clustersenderkanal, den Sie manuell konfigurieren möchten. Geben Sie in diesem Fall das Warteschlangenattribut **CLCHNAME** der Übertragungswarteschlange auf den Namen eines Clustersenderkanals an. Wählen Sie den Clustersenderkanal aus, der Nachrichten aus dieser Übertragungswarteschlange übertragen soll.
  - b. Kombinieren Sie den Nachrichtenverkehr für eine Gruppe von Clustersenderkanälen in derselben Clusterübertragungswarteschlange (siehe [Abbildung 56](#) auf Seite 312). In diesem Fall müssen Sie das Warteschlangenattribut **CLCHNAME** jeder allgemeinen Übertragungswarteschlange auf einen generischen Clustersenderkanalnamen setzen. Ein generischer Clustersenderkanalname ist ein Filter zum Gruppieren von Namen von Clustersenderkanälen. Beispiel: `SALES.*` gruppennamen für alle Clustersenderkanäle, die Namen haben, die mit `SALES.` beginnen. Sie können mehrere Platzhalterzeichen an einer beliebigen Position in der Filterzeichenfolge platzieren. Das Platzhalterzeichen ist ein Stern (`" *`). Er steht für eine Zahl von null bis zu einer beliebigen Anzahl von Zeichen.



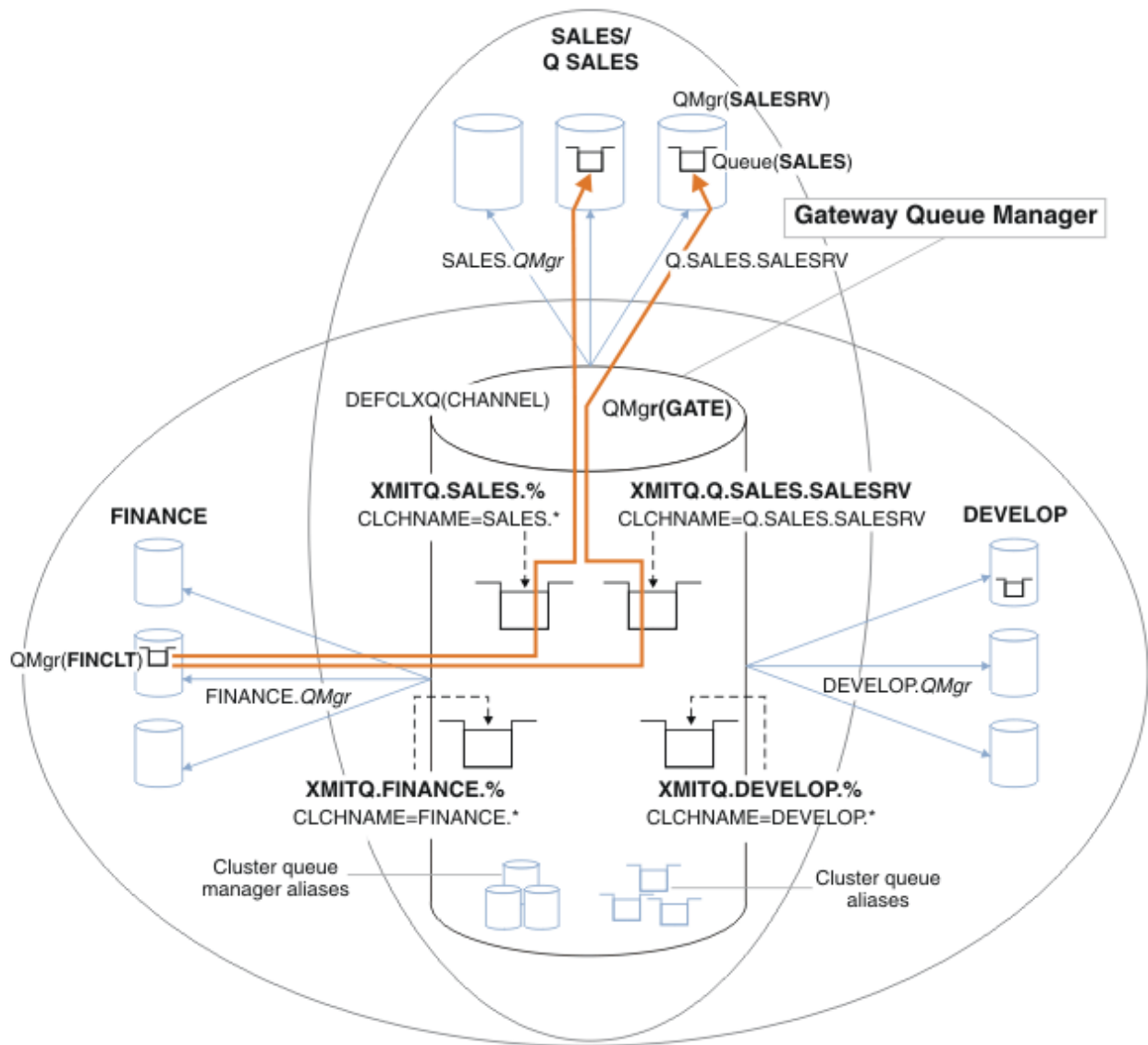


Abbildung 56. Beispiel für eigene Übertragungswarteschlangen für die verschiedenen IBM WebSphere MQ-Abteilungscluster

## Vorgehensweise

1. Wählen Sie den Typ der Standardübertragungswarteschlange für Cluster aus, die verwendet werden soll .
  - Wählen Sie eine einzelne Clusterübertragungswarteschlange oder separate Warteschlangen für jede Clusterverbindung aus.

Übernehmen Sie die Standardeinstellung oder führen Sie den Befehl **MQSC** aus:

```
ALTER QMGR DEFCLXQ(CHANNEL)
```

2. Isolieren Sie alle Nachrichtenflüsse, die keine Cluster-Übertragungswarteschlange mit anderen Flows gemeinsam nutzen dürfen .
  - Siehe „Clustering: Beispielkonfiguration mehrerer Clusterübertragungswarteschlangen“ auf Seite 316. In dem Beispiel ist die SALES-Warteschlange, die isoliert werden muss, ein Member des SALES-Clusters unter SALESRV. Wenn Sie die SALES -Warteschlange eingrenzen möchten, erstellen Sie einen neuen Cluster Q . SALES, erstellen Sie den SALESRV -Warteschlangenmanager zu einem Member und ändern Sie die SALES -Warteschlange, die zu Q . SALES gehören soll.



- WS-Manager, die Nachrichten an SALES senden, müssen ebenfalls Mitglieder des neuen Clusters sein. Wenn Sie einen Clusterwarteschlangenaliasnamen und einen Gateway-Warteschlangenmanager verwenden, wie im Beispiel in vielen Fällen, können Sie die Änderungen begrenzen, um den Gateway-Warteschlangenmanager zu einem Mitglied des neuen Clusters zu machen.
- Durch die Trennung der Flüsse vom Gateway zum Ziel werden jedoch keine Flüsse in das Gateway vom Quellenwarteschlangenmanager getrennt. Aber es erweist sich manchmal als ausreichend, um die Abläufe vom Gateway zu trennen und nicht zum Gateway zu fließen. Wenn dies nicht ausreichend ist, fügen Sie den Quellenwarteschlangenmanager in den neuen Cluster ein. Wenn Nachrichten über das Gateway übertragen werden sollen, versetzen Sie den Clusteralias in den neuen Cluster, und senden Sie weiterhin Nachrichten an den Clusteralias auf dem Gateway und nicht direkt an den Ziel-WS-Manager.

Führen Sie die folgenden Schritte aus, um Nachrichtenflüsse zu isolieren:

- a) Konfigurieren Sie die Ziele der Flüsse so, dass jede Zielwarteschlange die einzige Warteschlange in einem bestimmten Cluster ist, auf diesem Warteschlangenmanager .
  - b) Erstellen Sie die Cluster-Sender- und Clusterempfängerkanäle für alle neuen Cluster, die Sie nach einer systematischen Namenskonvention erstellt haben .
    - Siehe [„Clustering: Besondere Hinweise zu überlappenden Clustern“](#) auf Seite 301.
  - c) Definieren Sie eine Clusterübertragungswarteschlange für jedes isolierte Ziel auf jedem Warteschlangenmanager, der Nachrichten an die Zielwarteschlange sendet .
    - Als Konvention bei der Benennung von Clusterübertragungswarteschlangen wird dem Wert des Attributs für Clusterkanalnamen (CLCHNAME) das Präfix XMITQ . vorangestellt.
3. Erstellen Sie Clusterübertragungswarteschlangen, um die Governance- oder Überwachungsvoraussetzungen zu erfüllen .
- Typische Governance- und Überwachungsanforderungen führen zu einer Übertragungswarteschlange pro Cluster oder einer Übertragungswarteschlange pro Warteschlangenmanager. Wenn Sie die Namenskonvention für Clusterkanäle (*ClusterName.QueueManagerName*) befolgen, ist es einfach, generische Kanalnamen zu erstellen, die einen Cluster von Warteschlangenmanagern oder alle Cluster auswählen, zu denen ein Warteschlangenmanager gehört. (Siehe [„Clustering: Beispielkonfiguration mehrerer Clusterübertragungswarteschlangen“](#) auf Seite 316.)
  - Erweitern Sie die Namenskonvention für Clusterübertragungswarteschlangen, um generische Kanalnamen zu verwenden, indem Sie das Sternsymbol durch ein Prozentzeichen ersetzen. Zum Beispiel:

```
DEFINE QLOCAL(XMITQ.SALES.%) USAGE(XMITQ) CLCHNAME(SALES.*)
```

## Zugehörige Konzepte

[„Clusterübertragungswarteschlangen und Clustersenderkanäle“](#) auf Seite 185

Nachrichten zwischen Clustering-WS-Managern werden in Clusterübertragungswarteschlangen gespeichert und von Clustersenderkanälen weitergeleitet.

[„Clustering: Anwendungsisolierung mit mehreren Clusterübertragungswarteschlangen“](#) auf Seite 306

Sie können die Nachrichtenflüsse zwischen Warteschlangenmanagern in einem Cluster isolieren. Sie können Nachrichten, die von verschiedenen Clustersenderkanälen transportiert werden, in verschiedene Clusterübertragungswarteschlangen stellen. Sie können den Ansatz in einem einzelnen Cluster oder mit überlappenden Clustern verwenden. Das Thema enthält Beispiele und einige bewährte Verfahren, die Sie bei der Auswahl eines zu verwendenden Ansatzes führen.

[„Zugriffsteuerung und mehrere Clusterübertragungswarteschlangen“](#) auf Seite 173

Wählen Sie zwischen drei Prüfmodi aus, wenn eine Anwendung Nachrichten in ferne Clusterwarteschlangen einreicht. Die Modi werden über Remotezugriff auf die Clusterwarteschlange überprüft, lokal auf SYSTEM.CLUSTER.TRANSMIT.QUEUE geprüft oder auf lokale Profile für die Clusterwarteschlange oder den Cluster-WS-Manager überprüft.

[„Clustering: Besondere Hinweise zu überlappenden Clustern“](#) auf Seite 301

In diesem Abschnitt finden Sie Anleitungen für die Planung und Verwaltung von IBM WebSphere MQ-Clustern. Diese Informationen sind ein Leitfaden, der auf Tests und Feedback von Kunden basiert.

### „Überlappende Cluster“ auf Seite 195

Überlappende Cluster bieten zusätzliche Verwaltungsfunktionen. Verwenden Sie Namenslisten, um die Anzahl der Befehle zu reduzieren, die für die Verwaltung von überlappenden Clustern erforderlich sind.

### **Zugehörige Tasks**

„Definition einer fernen Warteschlange hinzufügen, um Nachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet wurden“ auf Seite 216

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine ferne Definition einer Clusterwarteschlange und einen separaten Senderkanal und eine separate Übertragungswarteschlange.

„Cluster-Übertragungswarteschlange zum Isolieren des Clusternachrichtenverkehrs hinzufügen, der von einem Gateway-Warteschlangenmanager gesendet wurde“ auf Seite 219

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet eine zusätzliche Clusterübertragungswarteschlange, um den Nachrichtenverkehr auf einen einzelnen Warteschlangenmanager in einem Cluster zu trennen.

„Cluster und Cluster-Übertragungswarteschlange hinzufügen, um den Datenverkehr der Clusternachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet werden“ auf Seite 224

Ändern Sie die Konfiguration von überlappenden Clustern, die einen Gateway-Warteschlangenmanager verwenden. Nachdem die Änderungsnachrichten von dem Gateway-Warteschlangenmanager an eine Anwendung übertragen wurden, ohne dieselbe Übertragungswarteschlange oder Kanäle wie andere Clusternachrichten zu verwenden. Die Lösung verwendet einen zusätzlichen Cluster, um die Nachrichten in einer bestimmten Clusterwarteschlange zu isolieren.

„Ändern der Standardeinstellung in separate Clusterübertragungswarteschlangen, um den Nachrichten-datenverkehr zu isolieren“ auf Seite 241

Sie können die Standardweise ändern, in der ein WS-Manager Nachrichten für eine Clusterwarteschlange oder ein Topic in einer Übertragungswarteschlange speichert. Wenn Sie den Standardwert ändern, können Sie Clusternachrichten auf einem Gateway-Warteschlangenmanager isolieren.

„Erstellen von zwei überlappenden Clustern mit einem Gateway-Warteschlangenmanager“ auf Seite 235

Befolgen Sie die Anweisungen in der Task, um überlappende Cluster mit einem Gateway-Warteschlangenmanager zu erstellen. Verwenden Sie die Cluster als Ausgangspunkt für die folgenden Beispiele, um Nachrichten in einer Anwendung von Nachrichten an andere Anwendungen in einem Cluster zu isolieren.

„Nachrichtenpfade zwischen Clustern konfigurieren“ auf Seite 278

Verbinden Sie Cluster unter Verwendung eines Gateway-Warteschlangenmanagers. Stellen Sie Warteschlangen oder Warteschlangenmanager für alle Cluster sichtbar, indem Sie Clusterwarteschlangen- oder Cluster-WS-Manager-Aliasnamen auf dem Gateway-Warteschlangenmanager definieren.

### **Art der zu verwendenden Clusterübertragungswarteschlange auswählen**

Vorgehensweise zur Auswahl zwischen verschiedenen Konfigurationsoptionen für die Clusterübertragungswarteschlange.

Ab Version 7.5 können Sie auswählen, welche Clusterübertragungswarteschlange einem Clustersenderkanal zugeordnet werden soll.

1. Sie können alle Clustersenderkanäle der einzigen standardmäßigen Clustersendewarteschlange `SYSTEM.CLUSTER.TRANSMIT.QUEUE` zuordnen. Diese Option ist die Standardeinstellung und die einzige Option für Warteschlangenmanager, die die Version Version 7.1 oder früher ausführen.
2. Sie können alle Clustersenderkanäle so festlegen, dass sie automatisch einer separaten Clusterübertragungswarteschlange zugeordnet werden. Die Warteschlangen werden vom Warteschlangenmanager aus der Modellwarteschlange `SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE` mit dem Namen `SYSTEM.CLUSTER.TRANSMIT.ChannelName` erstellt. Kanäle verwenden ihre eindeutig benannte Clusterübertragungswarteschlange, wenn das Warteschlangenmanagerattribut **DEFCLXQ** auf `CHANNEL` gesetzt ist.



**Achtung:** Wenn Sie dedizierte `SYSTEM.CLUSTER.TRANSMIT.QUEUES` mit einem Warteschlangenmanager verwenden, für den ein Upgrade von einer früheren Version des Produkts durchgeführt wurde, stellen Sie sicher, dass für `SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE` die Option `SHARE/NOSHARE` auf **SHARE** gesetzt ist.

3. Sie können bestimmte Clustersenderkanäle festlegen, die von einer einzelnen Clusterübertragungswarteschlange bedient werden sollen. Wählen Sie diese Option aus, indem Sie eine Übertragungswarteschlange erstellen und ihr Attribut **CLCHNAME** auf den Namen des Clustersenderkanals einstellen.
4. Sie können Gruppen von Clustersenderkanälen auswählen, die von einer einzelnen Clusterübertragungswarteschlange bedient werden sollen. Wählen Sie diese Option aus, indem Sie eine Übertragungswarteschlange erstellen und ihr Attribut **CLCHNAME** auf einen generischen Kanalnamen wie `ClusterName.*` einstellen. Wenn Sie Clusterkanäle nach den Namenskonventionen in „Clustering: Besondere Hinweise zu überlappenden Clustern“ auf Seite 301 benennen, wählt dieser Name alle Clusterkanäle aus, die mit Warteschlangenmanagern im Cluster `ClusterName` verbunden sind.

Sie können eine der Standardoptionen für die Clusterübertragungswarteschlange für einige Clustersenderkanäle mit einer beliebigen Anzahl spezifischer und generischer Cluster-Übertragungswarteschlangen-Konfigurationen kombinieren.

## Bewährte Verfahren

In den meisten Fällen ist bei vorhandenen IBM WebSphere MQ-Installationen die Standardkonfiguration die beste Wahl. Ein Clusterwarteschlangenmanager speichert Clusternachrichten in einer einzelnen Clusterübertragungswarteschlange, `SYSTEM.CLUSTER.TRANSMIT.QUEUE`. Sie haben die Möglichkeit, den Standardwert zum Speichern von Nachrichten für verschiedene Warteschlangenmanager und verschiedene Cluster in separaten Übertragungswarteschlangen oder für die Definition eigener Übertragungswarteschlangen zu ändern.

In den meisten Fällen ist bei neuen IBM WebSphere MQ-Installationen die Standardkonfiguration auch die beste Wahl. Der Prozess der Umschaltung von der Standardkonfiguration auf die alternative Standardkonfiguration mit einer Übertragungswarteschlange für jeden Clustersenderkanal erfolgt automatisch. Die Umschaltung erfolgt ebenfalls automatisch. Die Auswahl der einen oder der anderen ist nicht kritisch, Sie können sie umkehren.

Der Grund für die Auswahl einer anderen Konfiguration ist eher mit Governance und Management zu tun, als mit Funktionalität oder Leistung. Bei einigen Ausnahmebedingungen wird das Verhalten des WS-Managers nicht von der Konfiguration mehrerer Clusterübertragungswarteschlangen unterstützt. Sie führt zu mehr Warteschlangen und erfordert, dass Sie die Überwachungs- und Managementprozeduren, die Sie bereits konfiguriert haben, ändern müssen, die sich auf die einzelne Übertragungswarteschlange beziehen. Aus diesem Grund ist der Rest der Standardkonfiguration die beste Wahl, es sei denn, Sie verfügen über starke Governance- oder Verwaltungsgründe für eine andere Auswahl.

Die Ausnahmen beziehen sich beide darauf, was passiert, wenn die Anzahl der in `SYSTEM.CLUSTER.TRANSMIT.QUEUE` gespeicherten Nachrichten zunimmt. Wenn Sie alle Schritte zum Trennen der Nachrichten für ein Ziel von den Nachrichten für ein anderes Ziel verwenden, sollten Kanal- und Zustellungsprobleme mit einem Ziel die Zustellung an ein anderes Ziel nicht beeinträchtigen. Die Anzahl der in `SYSTEM.CLUSTER.TRANSMIT.QUEUE` gespeicherten Nachrichten kann sich jedoch erhöhen, da Nachrichten nicht schnell genug an ein Ziel zugestellt werden. Die Anzahl der Nachrichten in `SYSTEM.CLUSTER.TRANSMIT.QUEUE` für ein Ziel kann sich auf die Zustellung von Nachrichten an andere Ziele auswirken.

Um Probleme zu vermeiden, die sich aus der Befüllung einer einzelnen Übertragungswarteschlange ergeben, sollten Sie genügend Kapazität in Ihrer Konfiguration aufbauen. Wenn dann ein Ziel fehlschlägt und ein Nachrichtenrückstand zu erstellen beginnt, haben Sie die Zeit, das Problem zu beheben.

Wenn Nachrichten über einen Hub-Warteschlangenmanager, z. B. ein Cluster-Gateway, weitergeleitet werden, nutzen sie eine gemeinsame Übertragungswarteschlange, `SYSTEM.CLUSTER.TRANSMIT.QUEUE`. Wenn die Anzahl der in `SYSTEM.CLUSTER.TRANSMIT.QUEUE` auf dem Gateway-Warteschlangenmanager gespeicherten Nachrichten die maximale Länge erreicht, beginnt der Warteschlangenmanager, neue Nachrichten für die Übertragungswarteschlange zurückzuweisen, bis die Länge abnimmt.

Die Überlastung wirkt sich auf Nachrichten für alle Ziele aus, die über das Gateway weitergeleitet werden. Nachrichten sichern die Übertragungswarteschlangen von anderen Warteschlangenmanagern, die Nachrichten an das Gateway senden. Das Problem manifestiert sich in Nachrichten, die in die Fehlerprotokolle des Warteschlangenmanagers geschrieben werden, den Nachrichtendurchsatz und die abgelaufenen Zeiten zwischen dem Senden einer Nachricht und dem Zeitpunkt, zu dem eine Nachricht an ihrem Zielort ankommt, länger verstrichen sind.

Die Auswirkungen von Überlastung auf eine einzelne Übertragungswarteschlange können sichtbar werden, auch bevor sie voll sind. Wenn Sie einen Mischnachrichtenverkehr mit einigen großen nicht persistenten Nachrichten und einigen kleinen Nachrichten haben, wird die Zeit, in der kleine Nachrichten gesendet werden, mit der Größe der Übertragungswarteschlange erhöht. Die Verzögerung ist darauf zurückzuführen, dass große nicht persistente Nachrichten auf Platte geschrieben werden, die normalerweise nicht auf Platte geschrieben werden. Wenn Sie zeitkritische Nachrichtenflüsse haben, die eine Clusterübertragungswarteschlange mit anderen gemischten Nachrichtenflüssen gemeinsam nutzen, kann es sinnvoll sein, einen speziellen Nachrichtenpfad zu konfigurieren, um ihn von anderen Nachrichtenflüssen zu isolieren. Weitere Informationen finden Sie im Abschnitt [„Cluster und Cluster-Übertragungswarteschlange hinzufügen, um den Datenverkehr der Clusternachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet werden“](#) auf Seite 224 .

Die anderen Gründe für die Konfiguration getrennter Clusterübertragungswarteschlangen sind die Erfüllung der Governance-Anforderungen oder die Vereinfachung der Überwachung von Nachrichten, die an verschiedene Clusterziele gesendet werden. Möglicherweise müssen Sie beispielsweise nachweisen, dass Nachrichten für ein Ziel nie eine Übertragungswarteschlange mit Nachrichten für ein anderes Ziel gemeinsam nutzen.

Ändern Sie das Warteschlangenmanagerattribut **DEFCLXQ**, das die Standardclusterübertragungswarteschlange steuert, um für jeden Clustersenderkanal unterschiedliche Clusterübertragungswarteschlangen zu erstellen. Mehrere Ziele können einen Clustersenderkanal gemeinsam nutzen, sodass Sie Ihre Cluster so planen müssen, dass sie dieses Ziel vollständig erfüllen. Wenden Sie die Methode [„Cluster und Cluster-Übertragungswarteschlange hinzufügen, um den Datenverkehr der Clusternachrichten zu isolieren, die von einem Gateway-Warteschlangenmanager gesendet werden“](#) auf Seite 224 systematisch auf alle Clusterwarteschlangen an. Das Ergebnis, das Sie anstreben, ist es, dass keine Cluster-Destination einen Clustersenderkanal mit einem anderen Clusterziel gemeinsam nutzen kann. Dies hat zur Folge, dass keine Nachricht für eine Cluster-Destination ihre Clusterübertragungswarteschlange mit einer Nachricht für ein anderes Ziel gemeinsam nutzt.

Wenn Sie eine separate Clusterübertragungswarteschlange für einen bestimmten Nachrichtenfluss erstellen, ist es einfach, den Nachrichtenfluss zu diesem Ziel zu überwachen. Wenn Sie eine neue Clusterübertragungswarteschlange verwenden möchten, definieren Sie die Warteschlange, ordnen Sie sie einem Clustersenderkanal zu, und stoppen Sie den Kanal und starten Sie ihn. Die Änderung muss nicht permanent sein. Sie können einen Nachrichtenfluss für eine Weile isolieren, die Übertragungswarteschlange überwachen und anschließend die Standardübertragungswarteschlange erneut verwenden.

### **Zugehörige Tasks**

#### **Clustering: Beispielkonfiguration mehrerer Clusterübertragungswarteschlangen**

In dieser Task wenden Sie die Schritte zum Planen mehrerer Clusterübertragungswarteschlangen auf drei sich überlappende Cluster an. Die Anforderungen bestehen darin, Nachrichten von allen anderen Nachrichtenflüssen in eine Clusterwarteschlange zu trennen und Nachrichten für verschiedene Cluster in verschiedenen Clusterübertragungswarteschlangen zu speichern.

#### **Clustering: Clusterübertragungswarteschlangen wechseln**

Planen Sie, wie die Änderungen an den Clusterübertragungswarteschlangen eines vorhandenen Produktionswarteschlangenmanagers in Kraft gebracht werden.

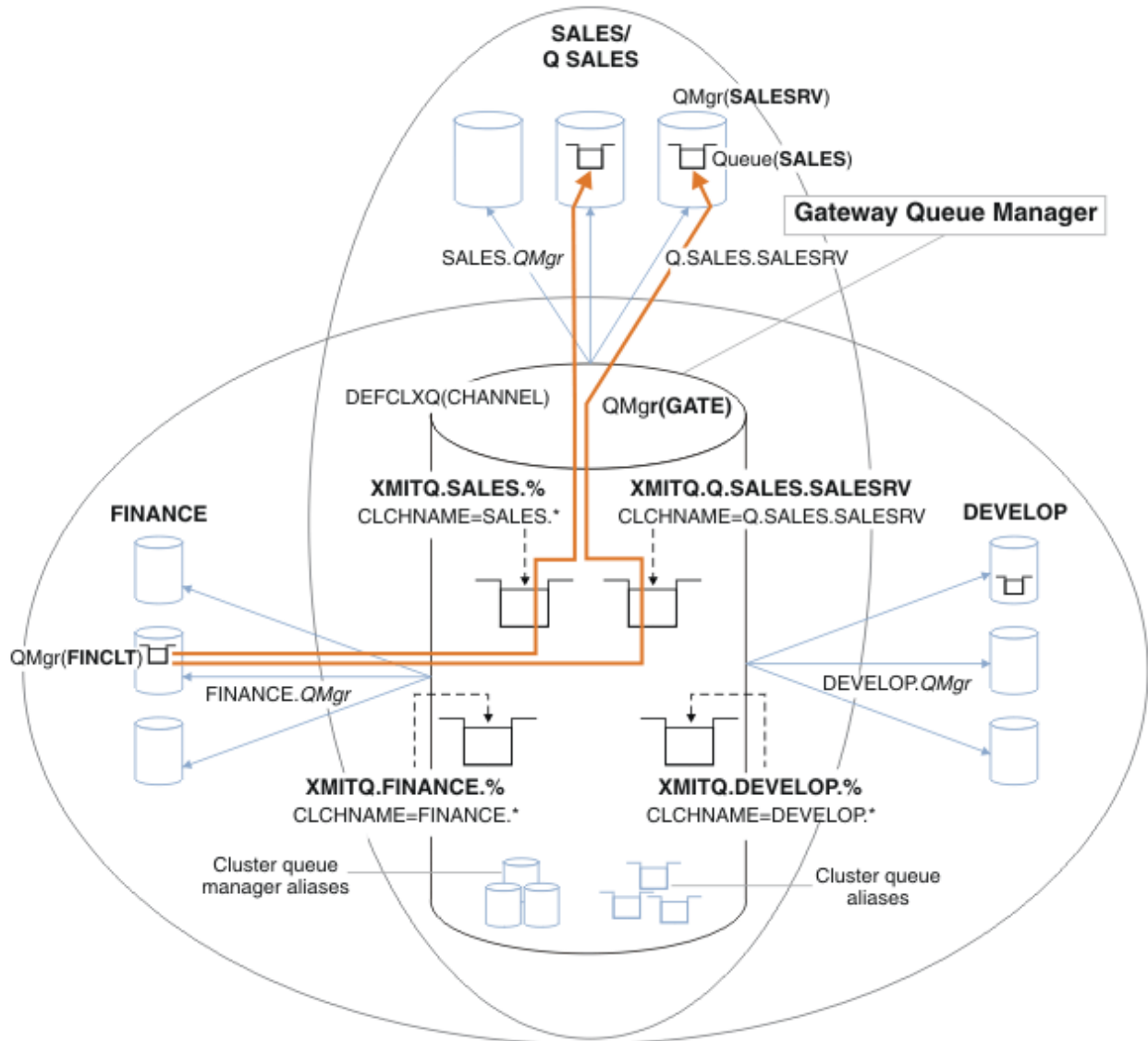
### **Clustering: Beispielkonfiguration mehrerer Clusterübertragungswarteschlangen**

In dieser Task wenden Sie die Schritte zum Planen mehrerer Clusterübertragungswarteschlangen auf drei sich überlappende Cluster an. Die Anforderungen bestehen darin, Nachrichten von allen anderen Nachrichtenflüssen in eine Clusterwarteschlange zu trennen und Nachrichten für verschiedene Cluster in verschiedenen Clusterübertragungswarteschlangen zu speichern.

## Informationen zu diesem Vorgang

Die Schritte in dieser Übung zeigen, wie die Prozedur in „Clustering: Planung der Konfiguration von Clusterübertragungswarteschlangen“ auf Seite 310 angewendet wird und die in [Abbildung 57](#) auf Seite 317 gezeigte Konfiguration erreicht wird. Es ist ein Beispiel für drei sich überlappende Cluster mit einem Gateway-WS-Manager, der mit separaten Clusterübertragungswarteschlangen konfiguriert ist. Die MQSC-Befehle zum Definieren der Cluster werden in „[Beispielcluster erstellen](#)“ auf Seite 319 beschrieben.

Für das Beispiel gibt es zwei Anforderungen. Eine davon ist die Trennung des Nachrichtenflusses vom Gateway-Warteschlangenmanager zu der Verkaufsanwendung, die die Verkäufe protokolliert. Der zweite Punkt ist die Abfrage, wie viele Nachrichten zu einem beliebigen Zeitpunkt darauf warten, an verschiedene Abteilbereiche gesendet zu werden. Die Cluster SALES, FINANCE und DEVELOP sind bereits definiert. Clusternachrichten werden derzeit von SYSTEM.CLUSTER.TRANSMIT.QUEUE weitergeleitet.



*Abbildung 57. Beispiel für eigene Übertragungswarteschlangen für die verschiedenen IBM WebSphere MQ-Abteilungscluster*

Die Schritte zum Ändern der Cluster sind wie folgt: Informationen zu den Definitionen finden Sie unter [Änderungen zum Isolieren der Vertriebswarteschlange in einem neuen Cluster und Trennen der Gateway-Clusterübertragungswarteschlangen](#).



## Vorgehensweise

1. Der erste Konfigurationsschritt ist in "Wählen Sie den Typ der Standardübertragungswarteschlange für Cluster aus, die verwendet werden soll".

Die Entscheidung besteht darin, separate Standardclusterübertragungswarteschlangen zu erstellen, indem der folgende **MQSC** -Befehl auf dem GATE -Warteschlangenmanager ausgeführt wird.

```
ALTER QMGR DEFCLXQ(CHANNEL)
```

Es gibt keinen starken Grund für die Auswahl dieser Standardeinstellung, da die Clusterübertragungswarteschlangen manuell definiert werden sollen. Die Auswahl weist einen schwachen Diagnosewert auf. Wenn eine manuelle Definition falsch ausgeführt wird und eine Nachricht eine Standard-Clusterübertragungswarteschlange abfließt, wird sie in der Erstellung einer permanentdynamischen Clusterübertragungswarteschlange angezeigt.

2. Der zweite Konfigurationsschritt befindet sich in "Isolieren Sie alle Nachrichtenflüsse, die keine Clusterübertragungswarteschlange mit anderen Flows gemeinsam nutzen dürfen".

In diesem Fall erfordert die Verkaufsanwendung, die Nachrichten aus der Warteschlange SALES unter SALESRV empfängt, die Isolation. Es ist nur die Isolation von Nachrichten vom Gateway-WS-Manager erforderlich. Die drei Unterschritte sind:

- a) "Konfigurieren Sie die Ziele der Flüsse so, dass jede Zielwarteschlange die einzige Warteschlange in einem bestimmten Cluster ist, auf diesem Warteschlangenmanager".

Das Beispiel erfordert, dass der WS-Manager SALESRV einem neuen Cluster innerhalb der Verkaufsabteilung hinzugefügt wird. Wenn Sie über wenige Warteschlangen verfügen, die isoliert werden müssen, können Sie sich entscheiden, einen bestimmten Cluster für die SALES -Warteschlange zu erstellen. A possible naming convention for the cluster name is to name such clusters, Q. *QueueName*, for example Q. SALES. Ein alternativer Ansatz, der praktischer ist, wenn Sie eine große Anzahl von Warteschlangen zu isolieren haben, besteht darin, Cluster von isolierten Warteschlangen zu erstellen, in denen und wann dies erforderlich ist. Die Clusternamen können QUEUES. *n* lauten.

In dem Beispiel wird der neue Cluster als Q. SALES bezeichnet. Informationen zum Hinzufügen des neuen Clusters finden Sie in den Definitionen unter Änderungen zum Isolieren der Vertriebswarteschlange in einem neuen Cluster und Trennen der Gateway-Clusterübertragungswarteschlangen. Die Zusammenfassung der Definitionsänderungen lautet wie folgt:

- i) Fügen Sie Q. SALES der Namensliste der Cluster auf den Repository-WS-Managern hinzu. Die Namensliste wird im Parameter **REPOSNL** des Warteschlangenmanagers referenziert.
- ii) Fügen Sie Q. SALES der Namensliste der Cluster auf dem Gateway-Warteschlangenmanager hinzu. Die Namensliste wird in allen Aliasnamendefinitionen der Clusterwarteschlange und des Cluster-WS-Managers auf dem Gateway-WS-Manager bezeichnet.
- iii) Erstellen Sie eine Namensliste auf dem Warteschlangenmanager SALESRV für beide Cluster, zu denen er gehört, und ändern Sie die Clusterzugehörigkeit der SALES -Warteschlange:

```
DEFINE NAMLIST(CLUSTERS) NAMES(SALES, Q.SALES) REPLACE
ALTER QLOCAL(SALES) CLUSTER(' ') CLUSNL(SALESRV.CLUSTERS)
```

Die SALES -Warteschlange ist nur für den Übergang Member der beiden Cluster. Sobald die neue Konfiguration ausgeführt wird, entfernen Sie die SALES -Warteschlange aus dem SALES -Cluster. Weitere Informationen finden Sie im Abschnitt Abbildung 58 auf Seite 322.

- b) "Erstellen Sie die Cluster-Sender- und Clusterempfängerkanäle für alle neuen Cluster, die Sie nach einer systematischen Namenskonvention erstellt haben".
  - i) Fügen Sie den Clusterempfängerkanal Q. SALES. *RepositoryQMgr* zu jedem der Repository-Warteschlangenmanager hinzu
  - ii) Fügen Sie den Clustersenderkanal Q. SALES. *OtherRepositoryQMgr* zu jedem der Repository-Warteschlangenmanager hinzu, um eine Verbindung zum anderen Repository-Manager herzustellen. Starten Sie diese Kanäle.

- iii) Fügen Sie die Clusterempfängerkanäle Q . SALES . SALESRV und Q . SALES . GATE einem der aktiven Repository-Warteschlangenmanager hinzu.
- iv) Fügen Sie die Clustersenderkanäle Q . SALES . SALESRV und Q . SALES . GATE zu den Warteschlangenmanagern SALESRV und GATE hinzu. Verbinden Sie den Clustersenderkanal mit dem Repository-WS-Manager, auf dem Sie die Clusterempfängerkanäle erstellt haben.
- c) " Definieren Sie eine Clusterübertragungswarteschlange für jedes isolierte Ziel auf jedem Warteschlangenmanager, der Nachrichten an die Zielwarteschlange sendet ".

Definieren Sie auf dem Gateway-Warteschlangenmanager die Clusterübertragungswarteschlange XMITQ . Q . SALES . SALESRV für den Q . SALES . SALESRV-Clustersenderkanal:

```
DEFINE QLOCAL(XMITQ.Q.SALES.SALESRV) USAGE(XMITQ) CLCHNAME(Q.SALES.SALESRV) REPLACE
```

3. Der dritte Konfigurationsschritt ist in " Erstellen Sie Clusterübertragungswarteschlangen, um die Governance-oder Überwachungsvoraussetzungen zu erfüllen " enthalten.

Definieren Sie auf dem Gateway-WS-Manager die Clusterübertragungswarteschlangen:

```
DEFINE QLOCAL(XMITQ.SALES) USAGE(XMITQ) CLCHNAME(SALES.*) REPLACE
DEFINE QLOCAL(XMITQ.DEVELOP) USAGE(XMITQ) CLCHNAME(DEVELOP.*) REPLACE
DEFINE QLOCAL(XMITQ.FINANCE) USAGE(XMITQ) CLCHNAME(FINANCE.*) REPLACE
```

## Nächste Schritte

Wechseln Sie in die neue Konfiguration auf dem Gateway-Warteschlangenmanager.

Der Switch wird ausgelöst, indem die neuen Kanäle gestartet werden, und die Kanäle, die jetzt verschiedenen Übertragungswarteschlangen zugeordnet sind, erneut gestartet werden. Alternativ können Sie den Gateway-WS-Manager stoppen und starten.

1. Stoppen Sie die folgenden Kanäle auf dem Gateway-WS-Manager:

```
SALES.Qmgr
DEVELOP.Qmgr
FINANCE.Qmgr
```

2. Starten Sie die folgenden Kanäle auf dem Gateway-WS-Manager:

```
SALES.Qmgr
DEVELOP.Qmgr
FINANCE.Qmgr
Q.SALES.SAVESRV
```

Entfernen Sie nach Abschluss des Switch die Warteschlange SALES aus dem SALES -Cluster (siehe Abbildung 58 auf Seite 322).

### Zugehörige Konzepte

Art der zu verwendenden Clusterübertragungswarteschlange auswählen

Vorgehensweise zur Auswahl zwischen verschiedenen Konfigurationsoptionen für die Clusterübertragungswarteschlange.

### Zugehörige Tasks

Clustering: Clusterübertragungswarteschlangen wechseln

Planen Sie, wie die Änderungen an den Clusterübertragungswarteschlangen eines vorhandenen Produktionswarteschlangenmanagers in Kraft gebracht werden.

#### *Beispielcluster erstellen*

Die Definitionen und Anweisungen zum Erstellen des Beispielclusters und zum Isolieren der SALES -Warteschlange und separate Nachrichten auf dem Gateway-WS-Manager.

## Informationen zu diesem Vorgang

Die vollständigen MQSC -Befehle zum Erstellen der FINANCE-, SALES- und Q.SALES -Cluster werden in Definitionen für die Basiscluster, Änderungen zum Isolieren der Vertriebswarteschlange in einem neuen Cluster und zum Trennen der Gateway-Clusterübertragungswarteschlangen und Entfernen der Vertriebswarteschlange auf dem Warteschlangenmanager SALESRV aus dem Vertriebscluster bereitgestellt. Der DEVELOP -Cluster wird in den Definitionen nicht angegeben, um die Definitionen kürzer zu halten.

## Vorgehensweise

1. Erstellen Sie die SALES - und FINANCE -Cluster und den Gateway-Warteschlangenmanager.

a) Erstellen Sie die Warteschlangenmanager.

Führen Sie den Befehl `crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE QmgrName` für jeden Warteschlangenmanagernamen in [Tabelle 27](#) auf Seite 320 aus.

| Tabelle 27. Namen und Portnummern des Warteschlangenmanagers |                                 |            |
|--------------------------------------------------------------|---------------------------------|------------|
| Beschreibung                                                 | Name des Warteschlangenmanagers | Portnummer |
| Finanzrepository                                             | FINR1                           | 1414       |
| Finanzrepository                                             | FINR2                           | 1415       |
| Finanzclient                                                 | FINCLT                          | 1418       |
| Verkaufsrepository                                           | SALER1                          | 1416       |
| Verkaufsrepository                                           | SALER2                          | 1417       |
| Verkaufsserver                                               | SALESRV                         | 1419       |
| Gateway                                                      | GATE                            | 1420       |

b) Alle WS-Manager starten

Führen Sie den Befehl `strmqm QmgrName` für jeden der WS-Manager-Namen in [Tabelle 27](#) auf Seite 320 aus.

c) Erstellen Sie die Definitionen für jeden der Warteschlangenmanager.

Führen Sie den Befehl `runmqsc QmgrName <filename` aus, wobei die Dateien in [Definitionen für die Basiscluster](#) aufgelistet sind und der Dateiname mit dem Namen des Warteschlangenmanagers übereinstimmt.

### Definitionen für die Basiscluster

#### finr1.txt

```
DEFINE LISTENER(1414) TRPTYPE(TCP) IPADDR(localhost) CONTROL(QMGR) PORT(1414) REPLACE
START LISTENER(1414)
ALTER QMGR REPOS(FINANCE)
DEFINE CHANNEL(FINANCE.FINR2) CHLTYPE(CLUSSDR) CONNAME('localhost(1415)') CLUSTER(FINANCE) REPLACE
DEFINE CHANNEL(FINANCE.FINR1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1414)') CLUSTER(FINANCE) REPLACE
```

#### finr2.txt

```
DEFINE LISTENER(1415) TRPTYPE(TCP) IPADDR(localhost) CONTROL(QMGR) PORT(1415) REPLACE
START LISTENER(1415)
ALTER QMGR REPOS(FINANCE)
DEFINE CHANNEL(FINANCE.FINR1) CHLTYPE(CLUSSDR) CONNAME('localhost(1414)') CLUSTER(FINANCE) REPLACE
DEFINE CHANNEL(FINANCE.FINR2) CHLTYPE(CLUSRCVR) CONNAME('localhost(1415)') CLUSTER(FINANCE) REPLACE
```



## finclt.txt

```
DEFINE LISTENER(1418) TRPTYPE(TCP) IPADDR(localhost) CONTROL(QMGR) PORT(1418) REPLACE
START LISTENER(1418)
DEFINE CHANNEL(FINANCE.FINR1) CHLTYPE(CLUSSDR) CONNAME('localhost(1414)') CLUS□
TER(FINANCE) REPLACE
DEFINE CHANNEL(FINANCE.FINCLT) CHLTYPE(CLUSRCVR) CONNAME('localhost(1418)') CLUS□
TER(FINANCE) REPLACE
DEFINE QMODEL(SYSTEM.SAMPLE.REPLY) REPLACE
```

## saler1.txt

```
DEFINE LISTENER(1416) TRPTYPE(TCP) IPADDR(localhost) CONTROL(QMGR) PORT(1416) REPLACE
START LISTENER(1416)
ALTER QMGR REPOS(SALES)
DEFINE CHANNEL(SALES.SALER2) CHLTYPE(CLUSSDR) CONNAME('localhost(1417)') CLUSTER(SA□
LES) REPLACE
DEFINE CHANNEL(SALES.SALER1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1416)') CLUSTER(SA□
LES) REPLACE
```

## saler2.txt

```
DEFINE LISTENER(1417) TRPTYPE(TCP) IPADDR(localhost) CONTROL(QMGR) PORT(1417) REPLACE
START LISTENER(1417)
ALTER QMGR REPOS(SALES)
DEFINE CHANNEL(SALES.SALER1) CHLTYPE(CLUSSDR) CONNAME('localhost(1416)') CLUSTER(SA□
LES) REPLACE
DEFINE CHANNEL(SALES.SALER2) CHLTYPE(CLUSRCVR) CONNAME('localhost(1417)') CLUSTER(SA□
LES) REPLACE
```

## salesrv.txt

```
DEFINE LISTENER(1419) TRPTYPE(TCP) IPADDR(localhost) CONTROL(QMGR) PORT(1419) REPLACE
START LISTENER(1419)
DEFINE CHANNEL(SALES.SALER1) CHLTYPE(CLUSSDR) CONNAME('localhost(1416)') CLUSTER(SA□
LES) REPLACE
DEFINE CHANNEL(SALES.SALESRV) CHLTYPE(CLUSRCVR) CONNAME('localhost(1419)') CLUSTER(SA□
LES) REPLACE
DEFINE QLOCAL(SALES) CLUSTER(SALES) TRIGGER INITQ(SYSTEM.DEFAULT.INITIATION.QUEUE)
PROCESS(ECHO) REPLACE
DEFINE PROCESS(ECHO) APPLICID(AMQSECH) REPLACE
```

## gate.txt

```
DEFINE LISTENER(1420) TRPTYPE(TCP) IPADDR(LOCALHOST) CONTROL(QMGR) PORT(1420) REPLACE
START LISTENER(1420)
DEFINE NAMLIST(ALL) NAMES(SALES, FINANCE)
DEFINE CHANNEL(FINANCE.FINR1) CHLTYPE(CLUSSDR) CONNAME('LOCALHOST(1414)') CLUSTER(FI□
NANCE) REPLACE
DEFINE CHANNEL(FINANCE.GATE) CHLTYPE(CLUSRCVR) CONNAME('LOCALHOST(1420)') CLUSTER(FI□
NANCE) REPLACE
DEFINE CHANNEL(SALES.SALER1) CHLTYPE(CLUSSDR) CONNAME('LOCALHOST(1416)') CLUSTER(SA□
LES) REPLACE
DEFINE CHANNEL(SALES.GATE) CHLTYPE(CLUSRCVR) CONNAME('LOCALHOST(1420)') CLUSTER(SA□
LES) REPLACE
DEFINE QALIAS(A.SALES) CLUSNL(ALL) TARGET(SALES) TARGTYPE(QUEUE) DEFBIND(NOTFIXED) RE□
PLACE
DEFINE QREMOTE(FINCLT) RNAME(' ') RQMNAME(FINCLT) CLUSNL(ALL) REPLACE
DEFINE QREMOTE(SALESRV) RNAME(' ') RQMNAME(SALESRV) CLUSNL(ALL) REPLACE
```

2. Testen Sie die Konfiguration, indem Sie das Beispielanforderungsprogramm ausführen.

a) Starten Sie das Auslösemonitorprogramm auf dem SALESRV -Warteschlangenmanager.

Öffnen Sie unter Windows ein Befehlsfenster, und führen Sie den Befehl `runmqtrm -m SALESRV` aus.

b) Führen Sie das Beispielanforderungsprogramm aus, und senden Sie eine Anforderung.

Öffnen Sie unter Windows ein Befehlsfenster und führen Sie den Befehl `amqsreq A.SALES FINCLT` aus.

Die Anforderungsnachricht wird zurückgemeldet, und nach 15 Sekunden wird das Beispielprogramm beendet.

- Erstellen Sie die Definitionen, um die SALES -Warteschlange im Q.SALES -Cluster zu isolieren, und trennen Sie die Clusternachrichten für den Cluster SALES und FINANCE auf dem Gateway-WS-Manager.

Führen Sie den Befehl `runmqsc QmgrName < filename` aus, wobei die Dateien in der folgenden Liste aufgeführt sind und der Dateiname fast mit dem Namen des Warteschlangenmanagers übereinstimmt.

### Änderungen zum Isolieren der Verkaufswarteschlange in einem neuen Cluster und Trennen der Gateway-Cluster-Übertragungswarteschlangen

#### chgsaler1.txt

```
DEFINE NAMELIST(CLUSTERS) NAMES(SALES, Q.SALES)
ALTER QMGR REPOS(' ') REPOSNL(CLUSTERS)
DEFINE CHANNEL(Q.SALES.SALER2) CHLTYPE(CLUSSDR) CONNAME('localhost(1417)') CLUSTER(Q.SALES) REPLACE
DEFINE CHANNEL(Q.SALES.SALER1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1416)') CLUSTER(Q.SALES) REPLACE
```

#### chgsaler2.txt

```
DEFINE NAMELIST(CLUSTERS) NAMES(SALES, Q.SALES)
ALTER QMGR REPOS(' ') REPOSNL(CLUSTERS)
DEFINE CHANNEL(Q.SALES.SALER1) CHLTYPE(CLUSSDR) CONNAME('localhost(1416)') CLUSTER(Q.SALES) REPLACE
DEFINE CHANNEL(Q.SALES.SALER2) CHLTYPE(CLUSRCVR) CONNAME('localhost(1417)') CLUSTER(Q.SALES) REPLACE
```

#### chgsalesrv.txt

```
DEFINE NAMELIST (CLUSTERS) NAMES(SALES, Q.SALES)
DEFINE CHANNEL(Q.SALES.SALER1) CHLTYPE(CLUSSDR) CONNAME('localhost(1416)') CLUSTER(Q.SALES) REPLACE
DEFINE CHANNEL(Q.SALES.SAVESRV) CHLTYPE(CLUSRCVR) CONNAME('localhost(1419)') CLUSTER(Q.SALES) REPLACE
ALTER QLOCAL (SALES) CLUSTER(' ') CLUSNL(CLUSTERS)
```

#### chggate.txt

```
ALTER NAMELIST(ALL) NAMES(SALES, FINANCE, Q.SALES)
ALTER QMGR DEFCLXQ(CHANNEL)
DEFINE CHANNEL(Q.SALES.SALER1) CHLTYPE(CLUSSDR) CONNAME('localhost(1416)') CLUSTER(Q.SALES) REPLACE
DEFINE CHANNEL(Q.SALES.GATE) CHLTYPE(CLUSRCVR) CONNAME('localhost(1420)') CLUSTER(Q.SALES) REPLACE
DEFINE QLOCAL (XMITQ.Q.SALES.SALESRV) USAGE(XMITQ) CLCHNAME(Q.SALES.SALESRV) REPLACE
DEFINE QLOCAL (XMITQ.SALES) USAGE(XMITQ) CLCHNAME(SALES.*) REPLACE
DEFINE QLOCAL (XMITQ.FINANCE) USAGE(XMITQ) CLCHNAME(FINANCE.*) REPLACE
```

- Entfernen Sie die Warteschlange SALES aus dem SALES-Cluster.

Führen Sie den Befehl **MQSC** in [Abbildung 58](#) auf Seite 322 aus:

```
ALTER QLOCAL(SALES) CLUSTER('Q.SALES') CLUSNL(' ')
```

*Abbildung 58. Entfernen Sie die Verkaufswarteschlange auf dem WS-Manager SALESRV aus dem Vertriebscluster.*

- Schalten Sie die Kanäle in die neuen Übertragungswarteschlangen ein.

Die Anforderung besteht darin, alle Kanäle zu stoppen und zu starten, die der WS-Manager von GATE verwendet. Stoppen und starten Sie den WS-Manager mit der geringsten Anzahl an Befehlen.

```
endmqm -i GATE
startmqm GATE
```

## Nächste Schritte

- Das Beispielanforderungsprogramm erneut ausführen, um die neue Konfiguration zu überprüfen. Siehe Schritt „2“ auf Seite 321

2. Überwachen Sie die Nachrichten, die durch alle Clusterübertragungswarteschlangen des GATE -Warteschlangenmanagers fließen:
  - a. Ändern Sie die Definition der einzelnen Clusterübertragungswarteschlangen, um die Warteschlangenüberwachung zu aktivieren.

```
ALTER QLOCAL(SYSTEM.CLUSTER.TRANSMIT.
name) STATQ(ON)
```

- b. Überprüfen Sie, ob die Statistiküberwachung des Warteschlangenmanagers OFF ist, um die Ausgabe zu minimieren, und setzen Sie das Überwachungsintervall auf einen niedrigeren Wert, um mehrere Tests bequem auszuführen.

```
ALTER QMGR STATINT(60) STATCHL(OFF) STATQ(OFF) STATMQI(OFF) STATACLS(OFF)
```

- c. Starten Sie den WS-Manager von GATE erneut.
  - d. Führen Sie das Beispielanforderungsprogramm ein paar Mal aus, um zu überprüfen, ob eine gleiche Anzahl von Nachrichten über SYSTEM.CLUSTER.TRANSMIT.Q.SALES.SALESRV und SYSTEM.CLUSTER.TRANSMIT.QUEUE fließt. Fordert den Fluss von Anforderungen durch SYSTEM.CLUSTER.TRANSMIT.Q.SALES.SALESRV an und antwortet über SYSTEM.CLUSTER.TRANSMIT.QUEUE.

```
amqsmon -m GATE -t statistics
```

- e. Die Ergebnisse in einigen Intervallen lauten wie folgt:

```
C:\Documents and Settings\Admin>amqsmon -m GATE -t statistics
MonitoringType: QueueStatistics
QueueManager: 'GATE'
IntervalStartDate: '2012-02-27'
IntervalStartTime: '14.59.20'
IntervalEndDate: '2012-02-27'
IntervalEndTime: '15.00.20'
CommandLevel: 700
ObjectCount: 2
QueueStatistics: 0
 QueueName: 'SYSTEM.CLUSTER.TRANSMIT.QUEUE'
 CreateDate: '2012-02-24'
 CreateTime: '15.58.15'
 ...
 Put1Count: [0, 0]
 Put1FailCount: 0
 PutBytes: [435, 0]
 GetCount: [1, 0]
 GetBytes: [435, 0]
 ...
QueueStatistics: 1
 QueueName: 'SYSTEM.CLUSTER.TRANSMIT.Q.SALES.SAVESRV'
 CreateDate: '2012-02-24'
 CreateTime: '16.37.43'
 ...
 PutCount: [1, 0]
 PutFailCount: 0
 Put1Count: [0, 0]
 Put1FailCount: 0
 PutBytes: [435, 0]
 GetCount: [1, 0]
 GetBytes: [435, 0]
```

```

...
MonitoringType: QueueStatistics
QueueManager: 'GATE'
IntervalStartDate: '2012-02-27'
IntervalStartTime: '15.00.20'
IntervalEndDate: '2012-02-27'
IntervalEndTime: '15.01.20'
CommandLevel: 700
ObjectCount: 2
QueueStatistics: 0
 QueueName: 'SYSTEM.CLUSTER.TRANSMIT.QUEUE'
 CreateDate: '2012-02-24'
 CreateTime: '15.58.15'
 ...
 PutCount: [2, 0]
 PutFailCount: 0
 Put1Count: [0, 0]
 Put1FailCount: 0
 PutBytes: [863, 0]
 GetCount: [2, 0]
 GetBytes: [863, 0]
 ...
QueueStatistics: 1
 QueueName: 'SYSTEM.CLUSTER.TRANSMIT.Q.SALES.SAVESRV'
 CreateDate: '2012-02-24'
 CreateTime: '16.37.43'
 ...
 PutCount: [2, 0]
 PutFailCount: 0
 Put1Count: [0, 0]
 Put1FailCount: 0
 PutBytes: [863, 0]
 GetCount: [2, 0]
 GetBytes: [863, 0]
 ...
2 Records Processed.

```

Eine Anforderungs- und Antwortnachricht wurde im ersten Intervall und zwei in der zweiten Nachricht gesendet. Sie können ableiten, dass die Anforderungsnachrichten in SYSTEM.CLUSTER.TRANSMIT.Q.SALES.SAVESRV und die Antwortnachrichten in SYSTEM.CLUSTER.TRANSMIT.QUEUE gestellt wurden.

### ***Clustering: Clusterübertragungswarteschlangen wechseln***

Planen Sie, wie die Änderungen an den Clusterübertragungswarteschlangen eines vorhandenen Produktionswarteschlangenmanagers in Kraft gebracht werden.

### **Vorbereitende Schritte**

Wenn Sie die Anzahl der Nachrichten reduzieren, die der Switching-Prozess in die neue Übertragungswarteschlange übertragen muss, wird der Wechsel schneller abgeschlossen. Lesen Sie „[Funktionsweise des Prozesses zum Wechseln des Clustersenderkanals in eine andere Übertragungswarteschlange](#)“ auf Seite 187, um die Gründe für das Leeren der Übertragungswarteschlange zu finden, bevor Sie fortfahren.

### **Informationen zu diesem Vorgang**

Sie haben die Wahl zwischen zwei Möglichkeiten, die Änderungen an Clusterübertragungswarteschlangen wirksam zu machen.

1. Lassen Sie den WS-Manager die Änderungen automatisch vornehmen. Dies ist die Standardeinstellung. Der Warteschlangenmanager wechselt die Clustersenderkanäle mit anstehender Übertragungswarteschlange, wenn ein Clustersenderkanal als Nächstes gestartet wird.
2. Nehmen Sie die Änderungen manuell vor. Sie können die Änderungen an einem Clustersenderkanal vornehmen, wenn er gestoppt wird. Sie können sie von einer Clusterübertragungswarteschlange in eine andere übertragen, bevor der Clustersenderkanal gestartet wird.

Welche Faktoren berücksichtigen Sie bei der Entscheidung, welche der beiden Optionen Sie auswählen können, und wie verwalten Sie den Switch?

## Prozedur

- Option 1: Lassen Sie den Warteschlangenmanager die Änderungen automatisch vornehmen (siehe [„Aktive Clustersenderkanäle zu einer anderen Gruppe von Clusterübertragungswarteschlangen wechseln“](#) auf Seite 326).

Wählen Sie diese Option aus, wenn der WS-Manager den Switch für Sie herstellen soll.

Eine alternative Möglichkeit, diese Option zu beschreiben, besteht darin, dass der Warteschlangenmanager einen Clustersenderkanal umschaltet, ohne dass der Kanal gestoppt werden muss. Sie haben die Möglichkeit, den Kanal zu zwingen, den Kanal zu stoppen und dann den Kanal zu starten, damit der Schalter früher passiert. Die Umschaltung wird gestartet, wenn der Kanal gestartet wird und er wird ausgeführt, während der Kanal aktiv ist, was sich von Option 2 unterscheidet. In Option 2 erfolgt die Umschaltung, wenn der Kanal gestoppt wird.

Wenn Sie diese Option auswählen, indem Sie den Switch automatisch passieren lassen, wird der Umschaltvorgang gestartet, wenn ein Clustersenderkanal gestartet wird. Wenn der Kanal nicht gestoppt wird, wird er gestartet, wenn er inaktiv wird, wenn eine Nachricht zum Verarbeiten vorhanden ist. Wenn der Kanal gestoppt ist, starten Sie ihn mit dem Befehl `START CHANNEL`.

Der Switchprozess wird beendet, sobald keine Nachrichten mehr für den Clustersenderkanal in der Übertragungswarteschlange übrig sind, die der Kanal bedient hat. Sobald dies der Fall ist, werden neu eingetroffene Nachrichten für den Clustersenderkanal direkt in der neuen Übertragungswarteschlange gespeichert. Bis dahin werden Nachrichten in der alten Übertragungswarteschlange gespeichert, und der Switching-Prozess überträgt Nachrichten aus der alten Übertragungswarteschlange in die neue Übertragungswarteschlange. Der Clustersenderkanal leitet Nachrichten aus der neuen Clusterübertragungswarteschlange während des gesamten Switching-Prozesses weiter. Wenn der Switchprozess abgeschlossen ist, hängt vom Status des Systems ab. Wenn Sie die Änderungen in einem Wartungsfenster vornehmen, müssen Sie vorher prüfen, ob der Switching-Prozess in der Zeit abgeschlossen ist. Ob die Zeit vollständig abgeschlossen wird, hängt davon ab, ob die Anzahl der Nachrichten, die auf die Übertragung aus der alten Übertragungswarteschlange warten, null erreicht.

Der Vorteil der ersten Methode ist, dass sie automatisch ist. Ein Nachteil besteht darin, dass Sie sicher sein müssen, dass Sie das System steuern können, um den Switchprozess im Wartungsfenster zu beenden, wenn die Konfigurationsänderungen auf ein Verwaltungsfenster beschränkt sind. Wenn Sie sich nicht sicher sein können, ist Option 2 möglicherweise eine bessere Wahl.

- Option 2: Nehmen Sie die Änderungen manuell vor ([„Gestoppten Clustersenderkanal in eine andere Clusterübertragungswarteschlange wechseln“](#) auf Seite 327).

Wählen Sie diese Option aus, wenn Sie den gesamten Switching-Prozess manuell steuern möchten oder ob Sie einen gestoppten oder inaktiven Kanal umschalten wollen. Es ist eine gute Wahl, wenn Sie einige Clustersenderkanäle umschalten und während eines Wartungsfensters den Switch ausführen möchten.

Eine alternative Beschreibung dieser Option ist die Angabe, dass Sie den Clustersenderkanal umschalten, während der Clustersenderkanal gestoppt ist.

Wenn Sie diese Option auswählen, haben Sie die vollständige Kontrolle über den Zeitpunkt, an dem der Switch ausgeführt wird.

Sie können sicher sein, dass Sie den Switching-Prozess in einem festen Zeitraum in einem Wartungsfenster abschließen können. Der Zeitpunkt, zu dem der Switch ausgeführt wird, hängt davon

ab, wie viele Nachrichten von einer Übertragungswarteschlange an die andere übertragen werden müssen. Wenn Nachrichten weiterhin ankommen, kann es zu einer Zeit dauern, bis der Prozess alle Nachrichten übertragen hat.

Sie haben die Möglichkeit, den Kanal zu wechseln, ohne Nachrichten aus der alten Übertragungswarteschlange zu übertragen. Der Switch ist "instant".

Wenn Sie den Clustersenderkanal erneut starten, beginnt er mit der Verarbeitung von Nachrichten in der Übertragungswarteschlange, die Sie neu zugeordnet haben.

Der Vorteil der zweiten Methode besteht darin, dass Sie die Kontrolle über den Schaltvorgang haben. Der Nachteil besteht darin, dass Sie die zu vermittelnden Clustersenderkanäle identifizieren müssen, die erforderlichen Befehle ausführen und alle unbestäubten Kanäle auflösen müssen, die möglicherweise verhindern, dass der Clustersenderkanal gestoppt wird.

## Zugehörige Konzepte

Art der zu verwendenden [Clusterübertragungswarteschlange auswählen](#)

Vorgehensweise zur Auswahl zwischen verschiedenen Konfigurationsoptionen für die Clusterübertragungswarteschlange.

## Zugehörige Tasks

[Clustering: Beispielkonfiguration mehrerer Clusterübertragungswarteschlangen](#)

In dieser Task wenden Sie die Schritte zum Planen mehrerer Clusterübertragungswarteschlangen auf drei sich überlappende Cluster an. Die Anforderungen bestehen darin, Nachrichten von allen anderen Nachrichtenflüssen in eine Clusterwarteschlange zu trennen und Nachrichten für verschiedene Cluster in verschiedenen Clusterübertragungswarteschlangen zu speichern.

[„Aktive Clustersenderkanäle zu einer anderen Gruppe von Clusterübertragungswarteschlangen wechseln“ auf Seite 326](#)

Mit dieser Task erhalten Sie drei Optionen zum Wechseln der aktiven Clustersenderkanäle. Eine Möglichkeit besteht darin, dass der WS-Manager den Switch automatisch macht, was die Ausführung von Anwendungen nicht beeinträchtigt. Die anderen Optionen sind zum manuellen Stoppen und Starten von Kanälen oder zum erneuten Starten des Warteschlangenmanagers.

[„Gestoppten Clustersenderkanal in eine andere Clusterübertragungswarteschlange wechseln“ auf Seite 327](#)

## Zugehörige Informationen

[„Funktionsweise des Prozesses zum Wechseln des Clustersenderkanals in eine andere Übertragungswarteschlange“ auf Seite 187](#)

*Aktive Clustersenderkanäle zu einer anderen Gruppe von Clusterübertragungswarteschlangen wechseln*

Mit dieser Task erhalten Sie drei Optionen zum Wechseln der aktiven Clustersenderkanäle. Eine Möglichkeit besteht darin, dass der WS-Manager den Switch automatisch macht, was die Ausführung von Anwendungen nicht beeinträchtigt. Die anderen Optionen sind zum manuellen Stoppen und Starten von Kanälen oder zum erneuten Starten des Warteschlangenmanagers.

## Vorbereitende Schritte

Ändern Sie die Konfiguration der Clusterübertragungswarteschlange. Sie können das WS-Manager-Attribut **DEFCLXQ** ändern oder das Attribut **CLCHNAME** von Übertragungswarteschlangen hinzufügen oder ändern.

Wenn Sie die Anzahl der Nachrichten reduzieren, die der Switching-Prozess in die neue Übertragungswarteschlange übertragen muss, wird der Wechsel schneller abgeschlossen. Lesen Sie [„Funktionsweise des Prozesses zum Wechseln des Clustersenderkanals in eine andere Übertragungswarteschlange“ auf Seite 187](#), um die Gründe für das Leeren der Übertragungswarteschlange zu finden, bevor Sie fortfahren.

## Informationen zu diesem Vorgang

Verwenden Sie die Schritte in der Task als Basis für die Bearbeitung eines eigenen Plans für die Änderung der Konfiguration der Clusterübertragungswarteschlange.

## Vorgehensweise

### 1. Optional: Aktualisieren Sie den aktuellen Kanalstatus

Erstellen Sie einen Datensatz mit dem Status der aktuellen und gespeicherten Kanäle, die Clusterübertragungswarteschlangen bedienen. Mit den folgenden Befehlen wird der Status angezeigt, der den Systemclusterübertragungswarteschlangen zugeordnet ist. Fügen Sie eigene Befehle hinzu, um den Status anzuzeigen, der den von Ihnen definierten Clusterübertragungswarteschlangen zugeordnet ist. Verwenden Sie eine Konvention wie `XMITQ.ChannelName`, um Clusterübertragungswarteschlangen zu benennen, die Sie definieren, damit der Kanalstatus für diese Übertragungswarteschlangen einfach angezeigt werden kann.

```
DISPLAY CHSTATUS(*) WHERE(XMITQ LK 'SYSTEM.CLUSTER.TRANSMIT.*')
DISPLAY CHSTATUS(*) SAVED WHERE(XMITQ LK 'SYSTEM.CLUSTER.TRANSMIT.*')
```

### 2. Übertragungswarteschlangen wechseln.

- Tun Sie nichts. Der Warteschlangenmanager wechselt beim Neustart die Clustersenderkanäle, wenn er nach dem Stoppen oder Inaktiv erneut gestartet wird.

Wählen Sie diese Option aus, wenn Sie keine Regeln oder Bedenken zum Ändern einer Warteschlangenmanagerkonfiguration haben. Aktive Anwendungen sind von den Änderungen nicht betroffen.

- Starten Sie den Warteschlangenmanager erneut. Alle Clustersenderkanäle werden bei Bedarf automatisch gestoppt und erneut gestartet.

Wählen Sie diese Option aus, um alle Änderungen sofort einzuleiten. Aktive Anwendungen werden durch den Warteschlangenmanager unterbrochen, da er heruntergefahren und erneut gestartet wird.

- Stoppen Sie einzelne Clustersenderkanäle, und starten Sie sie erneut.

Wählen Sie diese Option aus, um ein paar Kanäle sofort zu ändern. Beim Ausführen von Anwendungen wird eine kurze Verzögerung bei der Nachrichtenübertragung zwischen dem Stoppen und dem erneuten Starten des Nachrichtenkanals angezeigt. Der Clustersenderkanal bleibt aktiv, außer während der Zeit, in der Sie ihn gestoppt haben. Während der Vermittlung werden Nachrichten an die alte Übertragungswarteschlange zugestellt, durch den Vermittlungsvorgang in die neue Übertragungswarteschlange übertragen und vom Clustersenderkanal aus der neuen Übertragungswarteschlange weitergeleitet.

### 3. Optional: Kanäle überwachen, während sie umschalten

Zeigen Sie den Kanalstatus und die Übertragungswarteschlangentiefe während des Switchs an. Im folgenden Beispiel wird der Status der Übertragungswarteschlangen des Systemclusters angezeigt.

```
DISPLAY CHSTATUS(*) WHERE(XMITQ LK 'SYSTEM.CLUSTER.TRANSMIT.*')
DISPLAY CHSTATUS(*) SAVED WHERE(XMITQ LK 'SYSTEM.CLUSTER.TRANSMIT.*')
DISPLAY QUEUE('SYSTEM.CLUSTER.TRANSMIT.*') CURDEPTH
```

- 4. Optional: Überwachen Sie die Nachrichten " AMQ7341 Die Übertragungswarteschlange für den Kanal *ChannelName* , die von der Warteschlange *QueueName* in *QueueName* umgeschaltet wurde ", die in das Fehlerprotokoll des Warteschlangenmanagers geschrieben werden.

*Gestoppten Clustersenderkanal in eine andere Clusterübertragungswarteschlange wechseln*

## Vorbereitende Schritte

Sie können einige Konfigurationsänderungen vornehmen und sie jetzt wirksam werden lassen, ohne die betroffenen Clustersenderkanäle zu starten. Alternativ können Sie die Konfigurationsänderungen vornehmen, die Sie als einen der Schritte in der Task benötigen.

Wenn Sie die Anzahl der Nachrichten reduzieren, die der Switching-Prozess in die neue Übertragungswarteschlange übertragen muss, wird der Wechsel schneller abgeschlossen. Lesen Sie „[Funktionsweise des](#)

Prozesses zum Wechseln des Clustersenderkanals in eine andere Übertragungswarteschlange” auf Seite 187 , um die Gründe für das Leeren der Übertragungswarteschlange zu finden, bevor Sie fortfahren.

## Informationen zu diesem Vorgang

Diese Task schaltet die Übertragungswarteschlangen, die von gestoppten oder inaktiven Clustersenderkanälen bereitgestellt werden. Sie können diese Task ausführen, da ein Clustersenderkanal gestoppt ist und Sie die Übertragungswarteschlange sofort umschalten möchten. Dies ist beispielsweise der Fall, wenn ein Clustersenderkanal nicht gestartet wird oder ein anderes Konfigurationsproblem vorliegt. Um das Problem zu beheben, müssen Sie einen Clustersenderkanal erstellen und die Übertragungswarteschlange für den alten Clustersenderkanal mit dem neuen Clustersenderkanal verknüpfen, den Sie definiert haben.

Ein wahrscheinlicher Fall ist, dass Sie die Steuerung steuern wollen, wenn die Neukonfiguration von Clusterübertragungswarteschlangen ausgeführt wird. Um die Rekonfiguration vollständig zu steuern, stoppen Sie die Kanäle, ändern die Konfiguration und wechseln dann die Übertragungswarteschlangen.

## Vorgehensweise

1. Stoppen Sie die Kanäle, die Sie wechseln möchten.

- a) Stoppen Sie alle aktiven oder inaktiven Kanäle, die Sie wechseln möchten. Wenn Sie einen inaktiven Clustersenderkanal stoppen, wird dieser verhindert, während Sie Konfigurationsänderungen vornehmen.

```
STOP CHANNEL(ChannelName) MODE(QUIESCSE) STATUS(STOPPED)
```

2. Optional: Nehmen Sie die Konfigurationsänderungen vor.

Weitere Informationen finden Sie im Abschnitt [„Clustering: Beispielkonfiguration mehrerer Clusterübertragungswarteschlangen”](#) auf Seite 316.

3. Schalten Sie die Clustersenderkanäle in die neuen Clusterübertragungswarteschlangen um.

```
runswchl -m QmgrName -c ChannelName
```

Der Befehl **runswchl** überträgt alle Nachrichten in der alten Übertragungswarteschlange an die neue Übertragungswarteschlange. Wenn die Anzahl der Nachrichten in der alten Übertragungswarteschlange für diesen Kanal den Wert null erreicht, wird der Switch abgeschlossen. Der Befehl ist synchron. Der Befehl schreibt während des Umschaltvorgangs Statusnachrichten in das Fenster.

Während der Übertragungsphase werden vorhandene und neue Nachrichten, die für den Clustersenderkanal bestimmt sind, in die neue Übertragungswarteschlange übertragen.

Da der Clustersenderkanal gestoppt ist, werden die Nachrichten in der neuen Übertragungswarteschlange erstellt. Vergleichen Sie den gestoppten Clustersenderkanal mit dem Schritt „2” auf Seite 327 in [„Aktive Clustersenderkanäle zu einer anderen Gruppe von Clusterübertragungswarteschlangen wechseln”](#) auf Seite 326. In diesem Schritt wird der Clustersenderkanal ausgeführt, sodass Nachrichten nicht notwendigerweise in der neuen Übertragungswarteschlange erstellt werden müssen.

4. Optional: Kanäle überwachen, während sie umschalten

Zeigen Sie in einem anderen Befehlsfenster die Länge der Übertragungswarteschlange während des Switchs an. Im folgenden Beispiel wird der Status der Übertragungswarteschlangen des Systemclusters angezeigt.

```
DISPLAY QUEUE('SYSTEM.CLUSTER.TRANSMIT.*') CURDEPTH
```

5. Optional: Überwachen Sie die Nachrichten " AMQ7341 Die Übertragungswarteschlange für den Kanal *ChannelName* , die von der Warteschlange *QueueName* in *QueueName* umgeschaltet wurde ", die in das Fehlerprotokoll des Warteschlangenmanagers geschrieben werden.

6. Starten Sie die Clustersenderkanäle erneut, die Sie gestoppt haben.



Die Kanäle werden nicht automatisch gestartet, wenn Sie sie gestoppt haben, indem Sie sie in den Status STOPPED stellen.

```
START CHANNEL(ChannelName)
```

### Zugehörige Verweise

[runswchl](#)

[GELÖST-CHANNEL](#)

[STOP CHANNEL](#)

## Clustering: Best Practices für Migration und Änderung

Dieser Abschnitt enthält eine Anleitung zur Planung und Verwaltung von IBM WebSphere MQ-Clustern. Diese Informationen sind ein Leitfaden, der auf Tests und Feedback von Kunden basiert.

1. [„Objekte in einem Cluster verschieben“](#) auf Seite 329 (Bewährte Verfahren für das Verschieben von Objekten innerhalb eines Clusters ohne Installation von Fixpacks oder neuen Versionen von IBM WebSphere MQ).
2. [„Upgrades und Wartungsinstallationen“](#) auf Seite 330 (Bewährte Verfahren für die Aufrechterhaltung einer betriebsweisen Clusterarchitektur und die Ausführung von Wartungs- oder Upgrades und Tests der neuen Architektur).

## Objekte in einem Cluster verschieben

### Anwendungen und ihre Warteschlangen

Wenn Sie eine Warteschlangeninstanz, die auf einem WS-Manager gehostet wird, in einem anderen Warteschlangenmanager verschieben müssen, können Sie mit den Parametern für die Lastverteilung arbeiten, um einen reibungslosen Übergang zu gewährleisten.

Erstellen Sie eine Instanz der Warteschlange, in der sie neu gehostet werden soll, verwenden Sie jedoch die Einstellungen für die Lastverteilung im Cluster, um das Senden von Nachrichten an die ursprüngliche Instanz fortzusetzen, bis Ihre Anwendung bereit ist, zu wechseln. Dies wird mit den folgenden Schritten erreicht:

1. Setzen Sie die Eigenschaft **CLWL RANK** der vorhandenen Warteschlange auf einen hohen Wert (z. B. fünf).
2. Erstellen Sie die neue Instanz der Warteschlange, und setzen Sie die Eigenschaft **CLWL RANK** auf null.
3. Führen Sie eine beliebige weitere Konfiguration des neuen Systems aus, z. B. die Implementierung und das Starten von Anwendungen für die neue Instanz der Warteschlange.
4. Setzen Sie die Eigenschaft **CLWL RANK** der neuen Warteschlangeninstanz auf höher als die ursprüngliche Instanz, z. B. 9.
5. Zulassen, dass die ursprüngliche Warteschlangeninstanz alle in der Warteschlange befindlichen Nachrichten im System verarbeitet und dann die Warteschlange löscht.

### Verschieben ganzer WS-Manager

Wenn sich der WS-Manager auf demselben Host befindet, die IP-Adresse jedoch geändert wird, lautet der Prozess wie folgt:

- DNS, wenn es korrekt verwendet wird, kann die Vereinfachung des Prozesses vereinfachen. Informationen zur Verwendung von DNS durch Festlegen des Kanalattributs [Verbindungsname \(CONNNAME\)](#) finden Sie unter [ALTER CHANNEL](#).
- Wenn Sie ein vollständiges Repository verschieben, müssen Sie sicherstellen, dass Sie mindestens ein anderes vollständiges Repository haben, das problemlos ausgeführt wird (z. B. keine Probleme mit dem Kanalstatus), bevor Sie Änderungen vornehmen.
- Setzen Sie den Warteschlangenmanager mit dem Befehl [SUSPEND QMGR](#) aus, um die Datenverkehrsaufbauung zu vermeiden.

- Ändern Sie die IP-Adresse des Computers. Wenn Ihre CLUSRCVR-Kanaldefinition im Feld CONNAME eine IP-Adresse verwendet, ändern Sie diesen IP-Adresseneintrag. Der DNS-Cache muss möglicherweise durchgebürstet werden, um sicherzustellen, dass Aktualisierungen überall verfügbar sind.
- Wenn der Warteschlangenmanager die Verbindung zu den vollständigen Repositories wiederherstellt, lösen sich automatisch die Kanalautodeinitionen selbst auf.
- Wenn der Warteschlangenmanager ein vollständiges Repository und die Änderungen an der IP-Adresse befindet, muss sichergestellt werden, dass die Teilpartien so bald wie möglich umgestellt werden, um manuell definierte CLUSSDR-Kanäle an die neue Position zu verweisen. Solange dieser Schalter nicht ausgeführt wird, können diese WS-Manager möglicherweise nur die verbleibenden (unveränderten) vollständigen Repositories in Verbindung setzen, und es werden Warnungen angezeigt, die sich auf die falsche Kanaldefinition richten.
- Den WS-Manager mit dem Befehl `RESUME QMGR` wiederaufnehmen.

Wenn der Warteschlangenmanager auf einen neuen Host verschoben werden muss, ist es möglich, die Daten des Warteschlangenmanagers zu kopieren und aus einer Sicherung zurückzuschreiben. Dieser Prozess wird jedoch nicht empfohlen, es sei denn, es gibt keine anderen Optionen; es kann jedoch besser sein, einen Warteschlangenmanager auf einer neuen Maschine zu erstellen und Warteschlangen und Anwendungen zu replizieren, wie im vorherigen Abschnitt beschrieben. Diese Situation bietet einen reibungslosen Rollover/Rollback-Mechanismus.

Wenn Sie entschlossen sind, einen vollständigen Warteschlangenmanager unter Verwendung der Sicherung zu verschieben, befolgen Sie die folgenden bewährten Verfahren:

- Behandeln Sie den gesamten Prozess als Restore des Warteschlangenmanagers von der Sicherung, wobei Sie alle Prozesse anwenden, die Sie in der Regel für die Systemwiederherstellung verwenden, die für Ihre Betriebssystemumgebung geeignet sind.
- Verwenden Sie den Befehl `REFRESH CLUSTER` nach der Migration, um alle lokal gespeicherten Clusterinformationen (einschließlich aller unbestätigten, automatisch definierten Kanäle) zu löschen und die erneute Erstellung zu erzwingen.

**Anmerkung:** Bei großen Clustern kann der Befehl `REFRESH CLUSTER` während seiner Ausführung und danach in 27-Tage-Intervallen, wenn die Clusterobjekte ihre Statusaktualisierungen automatisch an alle interessierten Warteschlangenmanager hochladen, zu Unterbrechungen führen. Nähere Informationen hierzu erhalten Sie im Abschnitt [Die Aktualisierung in einem großen Cluster kann sich auf die Leistung und Verfügbarkeit auswirken](#).

Wenn Sie einen Warteschlangenmanager erstellen und die Konfiguration von einem vorhandenen WS-Manager im Cluster replizieren (wie in diesem Abschnitt beschrieben), behandeln Sie die beiden verschiedenen Warteschlangenmanager niemals so, wie sie eigentlich identisch sind. Geben Sie insbesondere keinen neuen Warteschlangenmanager denselben Warteschlangenmanagernamen und die gleiche IP-Adresse an. Der Versuch, auf diese Weise schnell zu einem Ersatz-Warteschlangenmanager zu wechseln, ist eine häufige Ursache für Probleme in IBM WebSphere MQ-Clustern. Der Cache erwartet, dass Aktualisierungen empfangen werden, einschließlich des Attributs `QMID`, und der Status kann beschädigt sein.

Wenn zwei verschiedene Warteschlangenmanager versehentlich mit demselben Namen erstellt werden, empfiehlt es sich, den Befehl `RESET CLUSTER QMID` zu verwenden, um den fehlerhaften Eintrag aus dem Cluster auszuwerfen.

## Upgrades und Wartungsinstallationen

Vermeiden Sie ein Big-Bang-Szenario (z. B. Stoppen aller Cluster- und Warteschlangenmanageraktivitäten, Anwenden aller Upgrades und Wartungsarbeiten auf alle Warteschlangenmanager und anschließendes Neustarten des gesamten Systems). Cluster sind so konzipiert, dass sie auch mehrere Warteschlangenmanagerversionen unterstützen. Ein gut geplantes, gestaffeltes Wartungskonzept wird daher empfohlen.

Haben Sie einen Backup-Plan:

- Haben Sie unter z/OS PTFs für Rückwärtsmigrationen angewendet?
- Haben Sie Sicherungen erstellt?
- Vermeiden Sie sofort die Verwendung der neuen Clusterfunktionalität: Warten Sie, bis Sie sicher sind, dass alle WS-Manager auf die neue Version aufgerüstet sind, und sind sicher, dass Sie keinen Rollback rückgängig machen werden. Die Verwendung einer neuen Clusterfunktion in einem Cluster, in dem einige WS-Manager noch auf einer früheren Version stehen, kann zu undefiniertem Verhalten führen. Wenn ein Warteschlangenmanager beispielsweise bei der Umstellung von IBM WebSphere MQ Version 6.0 auf IBM WebSphere MQ Version 7.1 ein Clusterthema definiert, werden IBM WebSphere MQ Version 6.0 -Warteschlangenmanager die Definition nicht verstehen oder nicht in der Lage sein, zu diesem Thema zu veröffentlichen.

Migrieren Sie zuerst die vollständigen Repositorys. Obwohl sie Informationen weiterleiten können, die sie nicht verstehen, können sie sie nicht fortbestehen, so dass es nicht der empfohlene Ansatz ist, es sei denn, es ist absolut notwendig. Weitere Informationen finden Sie im Abschnitt [Migration des WS-Managers-Clusters](#).

### Zugehörige Konzepte

„Clustering: Best Practices für REFRESH CLUSTER verwenden“ auf Seite 331

Sie verwenden den Befehl **REFRESH CLUSTER**, um alle lokal gespeicherten Informationen zu einem Cluster zu löschen und diese Informationen aus den vollständigen Repositorys im Cluster erneut zu erstellen. Sie sollten diesen Befehl nicht verwenden, außer in außergewöhnlichen Umständen. Wenn Sie es verwenden müssen, gibt es besondere Hinweise darauf, wie Sie es verwenden. Diese Informationen sind ein Leitfaden, der auf Tests und Feedback von Kunden basiert.

### Clustering: Best Practices für REFRESH CLUSTER verwenden

Sie verwenden den Befehl **REFRESH CLUSTER**, um alle lokal gespeicherten Informationen zu einem Cluster zu löschen und diese Informationen aus den vollständigen Repositorys im Cluster erneut zu erstellen. Sie sollten diesen Befehl nicht verwenden, außer in außergewöhnlichen Umständen. Wenn Sie es verwenden müssen, gibt es besondere Hinweise darauf, wie Sie es verwenden. Diese Informationen sind ein Leitfaden, der auf Tests und Feedback von Kunden basiert.

### Führen Sie nur den Befehl REFRESH CLUSTER aus, wenn Sie dies wirklich tun müssen.

Die IBM WebSphere MQ-Clustertechnologie stellt sicher, dass jede Änderung an der Clusterkonfiguration, z. B. die Änderung an einer Clusterwarteschlange, automatisch allen Mitgliedern des Clusters bekannt gegeben wird, die diese Information benötigen. Es ist nicht notwendig, weitere administrative Schritte zu unternehmen, um diese Weitergabe von Informationen zu erreichen.

Wenn solche Informationen nicht zu den Warteschlangenmanagern im Cluster gelangen, in denen sie erforderlich sind, z. B. eine Clusterwarteschlange, die von einem anderen Warteschlangenmanager im Cluster nicht bekannt ist, wenn eine Anwendung versucht, sie zum ersten Mal zu öffnen, bedeutet dies ein Problem in der Clusterinfrastruktur. Es ist beispielsweise möglich, dass ein Kanal nicht zwischen einem WS-Manager und einem vollständigen WS-Manager-Repository gestartet werden kann. Daher müssen alle Situationen, in denen Inkonsistenzen beobachtet werden, untersucht werden. Lösen Sie die Situation nach Möglichkeit ohne Verwendung des Befehls **REFRESH CLUSTER** auf.

In seltenen Fällen, die an anderer Stelle in dieser Produktdokumentation oder auf Anforderung des IBM Support dokumentiert sind, können Sie den Befehl **REFRESH CLUSTER** verwenden, um alle lokal gespeicherten Informationen zu einem Cluster zu löschen und diese Informationen aus den vollständigen Repositorys im Cluster erneut zu erstellen.

### Die Neuerung in einem großen Cluster kann die Leistung und Verfügbarkeit des Clusters beeinträchtigen.

Die Verwendung des Befehls **REFRESH CLUSTER** kann während der Ausführung des Clusters zu Unterbrechungen führen, z. B. durch eine plötzliche Zunahme der Arbeit für die vollständigen Repositorys, wenn sie die erneute Weitergabe von Clusterressourcen des Warteschlangenmanagers verarbeiten. Wenn

Sie in einem großen Cluster (d. h. viele Hunderte von Warteschlangenmanagern) aktualisieren, sollten Sie die Verwendung des Befehls in der täglichen Arbeit vermeiden, wenn möglich, und alternative Methoden verwenden, um spezifische Inkonsistenzen zu korrigieren. Wenn beispielsweise eine Clusterwarteschlange im Cluster nicht ordnungsgemäß weitergegeben wird, wird die Warteschlangenkonfiguration im gesamten Cluster von einem ersten Untersuchungsverfahren aktualisiert, das die Clusterwarteschlangendefinition aktualisiert, z. B. die Beschreibung der Clusterwarteschlange, die die Warteschlangenkonfiguration ändert. Dieser Prozess kann dazu beitragen, das Problem zu identifizieren und möglicherweise eine temporäre Inkonsistenz zu beheben.

Wenn alternative Methoden nicht verwendet werden können und Sie **REFRESH CLUSTER** in einem großen Cluster ausführen müssen, sollten Sie dies zu Zeiten geringer Systemauslastung oder während eines Wartungsfensters tun, um Auswirkungen auf Benutzerworkloads zu vermeiden. Sie sollten auch vermeiden, einen großen Cluster in einem einzigen Stapel zu aktualisieren und stattdessen die Aktivität wie in „Leistungs- und Verfügbarkeitsprobleme vermeiden, wenn Clusterobjekte automatische Aktualisierungen senden“ auf Seite 332 erläutert zu stagnieren.

## **Leistungs- und Verfügbarkeitsprobleme vermeiden, wenn Clusterobjekte automatische Aktualisierungen senden**

Nachdem ein neues Clusterobjekt in einem Warteschlangenmanager definiert ist, wird eine Aktualisierung für dieses Objekt alle 27 Tage ab der Definitionierungszeit generiert und an alle vollständigen Repositories im Cluster und an alle anderen interessierten Warteschlangenmanager gesendet. Wenn Sie den Befehl **REFRESH CLUSTER** an einen Warteschlangenmanager ausgeben, setzen Sie die Systemzeit für diese automatische Aktualisierung für alle Objekte zurück, die lokal im angegebenen Cluster definiert sind.

Wenn Sie einen großen Cluster (d. a. viele Hunderte von Warteschlangenmanagern) in einem einzigen Stapel oder unter anderen Umständen aktualisieren, z. B. ein System aus der Konfigurationssicherung erneut erstellen, werden alle diese Warteschlangenmanager nach 27 Tagen alle ihre Objektdefinitionen erneut für die vollständigen Repositories bekannt machen. Dies könnte wiederum dazu führen, dass das System erheblich langsamer oder gar nicht mehr verfügbar ist, bis alle Aktualisierungen abgeschlossen sind. Wenn Sie also mehrere Warteschlangenmanager in einem großen Cluster aktualisieren oder erneut erstellen müssen, sollten Sie die Umstellung der einzelnen Warteschlangenmanager auf mehrere Stunden oder Tage verteilen, um Beeinträchtigungen der Systemleistung durch nachfolgende automatische Aktualisierungen zu vermeiden.

## **Die Systemclusterprotokollwarteschlange**

Wenn eine **REFRESH CLUSTER** ausgeführt wird, erstellt der WS-Manager eine Momentaufnahme des Clusterstatus vor der Aktualisierung und speichert sie in `SYSTEM.CLUSTER.HISTORY.QUEUE (SCHQ)`, wenn sie auf dem Warteschlangenmanager definiert ist. Diese Momentaufnahme dient nur zu IBM -Servicezwecken, wenn es zu späteren Problemen mit dem System kommt. Der SCHQ wird standardmäßig auf verteilten WS-Managern beim Start definiert. Für die z/OS -Migration muss die SCHQ manuell definiert werden. Die Nachrichten in der SCHQ laufen nach drei Monaten ab.

### **Zugehörige Konzepte**

Anwendungsprobleme bei der Ausführung von REFRESH CLUSTER

Hinweise zu REFRESH CLUSTER für Publish/Subscribe-Cluster

### **Zugehörige Verweise**

MQSC-Befehlsreferenz: REFRESH CLUSTER

## **Clustering: Verfügbarkeit, mehrere Instanzen und Notfallwiederherstellung**

Dieser Abschnitt enthält eine Anleitung zur Planung und Verwaltung von IBM WebSphere MQ-Clustern. Diese Informationen sind ein Leitfaden, der auf Tests und Feedback von Kunden basiert.

IBM WebSphere MQ Das Clustering selbst ist keine Hochverfügbarkeitslösung, aber unter bestimmten Umständen kann es verwendet werden, um die Verfügbarkeit von Services mithilfe von IBM WebSphere MQ zu verbessern, indem beispielsweise mehrere Instanzen einer Warteschlange auf verschiedenen Warteschlangenmanagern vorhanden sind. Dieser Abschnitt enthält Anleitungen um sicherzustellen, dass die

IBM WebSphere MQ-Infrastruktur eine möglichst hohe Verfügbarkeit aufweist, damit sie in einer solchen Architektur verwendet werden kann.

### **Verfügbarkeit von Clusterressourcen**

Der Grund für die übliche Empfehlung, zwei vollständige Repositorys zu verwalten, besteht darin, dass der Verlust von einem nicht kritisch für die reibungslose Ausführung des Clusters ist. Selbst wenn beide nicht mehr verfügbar sind, gibt es eine 60-Tage-Karenzzeit für vorhandene Kenntnisse, die von Teilrepositorys gehalten werden, obwohl neue oder nicht zuvor aufgerufene Ressourcen (z. B. Warteschlangen) in diesem Ereignis nicht verfügbar sind.

### **Verwenden von Clustern zur Verbesserung der Anwendungsverfügbarkeit**

Ein Cluster kann Ihnen bei der Entwicklung hoch verfügbarer Anwendungen (z. B. einer Serveranwendung für Anforderungen/Antworttyp) helfen, indem Sie mehrere Instanzen der Warteschlange und der Anwendung verwenden. Falls erforderlich, können Prioritätsattribute die Anwendung "live" bevorzugen, es sei denn, ein WS-Manager oder Kanal ist beispielsweise nicht verfügbar. Dies ist leistungsfähig, um schnell umschalten zu können, um die Verarbeitung neuer Nachrichten fortzusetzen, wenn ein Problem auftritt.

Nachrichten, die einem bestimmten Warteschlangenmanager in einem Cluster zugestellt wurden, werden jedoch nur in dieser Warteschlangeninstanz gehalten und stehen erst dann zur Verarbeitung zur Verfügung, wenn dieser WS-Manager wiederhergestellt wird. Aus diesem Grund sollten Sie für die hohe Verfügbarkeit von Daten möglicherweise andere Technologien, wie z. B. Warteschlangenmanager mit mehreren Instanzen, berücksichtigen.

### **Warteschlangenmanager mit mehreren Instanzen**

Hoch verfügbare Software (durch mehrere Instanzen) ist die beste integrierte Lösung zur Aufrechterhaltung der Verfügbarkeit vorhandener Nachrichten. Weitere Informationen finden Sie unter [„WebSphere MQ mit Hochverfügbarkeitskonfigurationen verwenden“](#) auf Seite 343, [„Erstellen eines Warteschlangenmanagers mit mehreren Instanzen“](#) auf Seite 374 und im folgenden Abschnitt. Jeder WS-Manager in einem Cluster kann mit dieser Technik hoch verfügbar gemacht werden, solange alle WS-Manager im Cluster mindestens IBM WebSphere MQ Version 7.0.1 aktiv sind. Wenn sich die WS-Manager im Cluster auf früheren Ebenen befinden, verlieren sie möglicherweise die Verbindung zu den Multi-Instanz-WS-Managern, wenn sie zu einem sekundären IP-System umschlagen.

Wie bereits in diesem Artikel beschrieben, solange zwei vollständige Repositorys konfiguriert sind, sind sie fast von ihrer Natur hoch verfügbar. Falls IBM WebSphere MQ mit hoher Verfügbarkeit benötigt wird, können Multi-Instanz-Warteschlangenmanager für vollständige Repositorys verwendet werden. Es gibt keinen Grund, diese Methoden zu verwenden, und in der Tat für temporäre Ausfälle können diese Methoden während des Failover zusätzliche Leistungskosten verursachen. Wenn Sie die Software HA verwenden, anstatt zwei vollständige Repositorys auszuführen, wird davon abgeraten, weil z. B. bei einem einzelnen Kanalausfall nicht notwendigerweise ein Fehler fehlschlagen würde, aber es kann vorkommen, dass Teilrepositorys nicht in der Lage sind, Clusterressourcen abzufragen.

### **Wiederherstellung nach einem Katastrophenfall**

Eine Disaster-Recovery, z. B. die Wiederherstellung nach einer Beschädigung der Platten, auf denen die Daten eines Warteschlangenmanagers gespeichert waren, ist zwar schwierig, wird von IBM WebSphere MQ aber unterstützt, obwohl der Vorgang nicht automatisch ausgeführt werden kann. Die einzige "wahre" Disaster-Recovery-Option in IBM WebSphere MQ (abgesehen von den durch das Betriebssystem bereitgestellten oder sonstigen grundlegenden Replikationstechnologien) ist die Wiederherstellung aus einer Sicherung. Es gibt einige Cluster-spezifische Punkte, die in diesen Situationen zu berücksichtigen sind:

- Gehen Sie beim Testen von Szenarios zur Notfallwiederherstellung sorgfältig vor. Wenn Sie beispielsweise den Betrieb von Sicherungswarteschlangenmanagern testen, müssen Sie darauf achten, dass sie in demselben Netz online sind, da es möglich ist, versehentlich den Live-Cluster zu verbinden und die 'Stealing' -Nachrichten zu starten, indem sie dieselben benannten Warteschlangen wie in den Live-Cluster-WS-Managern enthalten.
- Der Test zur Wiederherstellung nach einem Katastrophenfall darf nicht in einen aktiven Live-Cluster eingreifen. Zu den Techniken zur Vermeidung von Kollisionen gehören:
  - Vollständige Netztrennung oder Trennung auf Firewall-Ebene.

- Es wird kein Live-SSL-Zertifikat für das Notfallwiederherstellungssystem ausgegeben, es sei denn, ein Notfallwiederherstellungsszenario tritt tatsächlich ein.
- Wenn Sie eine Sicherung eines Warteschlangenmanagers im Cluster wiederherstellen, ist es möglich, dass die Sicherung nicht mehr mit dem Rest des Clusters synchronisiert ist. Der Befehl **REFRESH CLUSTER** kann Aktualisierungen auflösen und mit dem Cluster synchronisieren, aber der Befehl **REFRESH CLUSTER** muss als letzte Möglichkeit verwendet werden. Siehe „Clustering: Best Practices für REFRESH CLUSTER verwenden“ auf Seite 331. Bevor Sie den Befehl verwenden, überprüfen Sie in der unternehmensinternen Dokumentation und der IBM WebSphere MQ-Dokumentation, ob irgendein einfacher Schritt versäumt wurde.
- Wie bei jeder Wiederherstellung müssen die Anwendungen mit der Wiedergabe und dem Verlust von Daten umgehen. Es muss entschieden werden, ob die Warteschlangen in einem bekannten Status gelöscht werden sollen oder ob genügend Informationen vorhanden sind, um die Replays zu verwalten.

## Clustering: Überwachung

In diesem Abschnitt finden Sie Anleitungen für die Planung und Verwaltung von IBM WebSphere MQ-Clustern. Diese Informationen sind ein Leitfaden, der auf Tests und Feedback von Kunden basiert.

### Anwendungsnachrichten im Cluster überwachen

In der Regel durchlaufen alle Clusternachrichten, die den Warteschlangenmanager verlassen, den SYSTEM.CLUSTER.TRANSMIT.QUEUE, unabhängig davon, welcher Clustersenderkanal zur Übertragung der Nachricht verwendet wird. Jeder Kanal enthält Nachrichten, die für diesen Kanal parallel zu allen anderen Clustersenderkanälen ausgerichtet sind. Ein wachsender Aufbau von Nachrichten in dieser Warteschlange kann auf ein Problem mit einem oder mehreren Kanälen hinweisen und muss untersucht werden:

- Die Tiefe der Warteschlange muss entsprechend für das Clusterdesign überwacht werden.
- Der folgende Befehl gibt alle Kanäle zurück, die mehr als eine Nachricht haben, die auf die Übertragungswarteschlange wartet:

```
DIS CHSTATUS(*) WHERE(XQMSGSA GT 1)
```

Bei allen Clusternachrichten in einer einzigen Warteschlange ist es nicht immer leicht zu sehen, welcher Kanal Probleme hat, wenn er beginnt, sich zu füllen. Die Verwendung dieses Befehls ist eine einfache Möglichkeit, zu sehen, welcher Kanal verantwortlich ist.

Sie können einen Cluster-WS-Manager so konfigurieren, dass er mehrere Übertragungswarteschlangen hat. Wenn Sie das WS-Managerattribut DEFCLXQ in CHANNEL ändern, wird jedem Clustersenderkanal eine andere Clusterübertragungswarteschlange zugeordnet. Alternativ können Sie separate Übertragungswarteschlangen manuell konfigurieren. Führen Sie den folgenden Befehl aus, um alle Cluster-Übertragungswarteschlangen anzuzeigen, die Clustersenderkanälen zugeordnet sind:

```
DISPLAY CLUSQMGR (qmgrName) XMITQ
```

Definieren Sie die Clusterübertragungswarteschlangen so, dass sie dem Muster des festen Stamms des Warteschlangennamens auf der linken Seite folgen. Anschließend können Sie die Länge aller Clusterübertragungswarteschlangen abfragen, die vom Befehl **DISPLAY CLUSMGR** zurückgegeben wurden, indem Sie einen generischen Warteschlangennamen verwenden:

```
DISPLAY QUEUE (qname*) CURDEPTH
```

### Überwachen von Steuernachrichten im Cluster

Die Warteschlange SYSTEM.CLUSTER.COMMAND.QUEUE wird für die Verarbeitung aller Cluster-Steuernachrichten für einen Warteschlangenmanager verwendet, die entweder vom lokalen Warteschlangenmanager generiert oder von anderen WS-Managern im Cluster an diesen Warteschlangenmanager gesendet werden. Wenn ein WS-Manager seinen Clusterstatus korrekt aufrechterhält, neigt diese Warteschlange



zu null. Es gibt Situationen, in denen die Tiefe der Nachrichten in dieser Warteschlange vorübergehend zunehmen kann:

- Wenn viele Nachrichten in der Warteschlange vorhanden sind, wird das Churn im Clusterstatus angezeigt.
- Wenn Sie wichtige Änderungen vornehmen, können Sie die Warteschlange zwischen diesen Änderungen einlagern. Wenn z. B. Repositories verschoben werden, können Sie die Warteschlange vor dem Verschieben des zweiten Repositorys auf null setzen.

Während ein Nachrichtenrückstand in dieser Warteschlange vorhanden ist, werden Aktualisierungen für den Clusterstatus oder die clusterbezogenen Befehle nicht verarbeitet. Wenn Nachrichten über einen längeren Zeitraum nicht aus dieser Warteschlange entfernt werden, ist eine weitere Untersuchung erforderlich, die zunächst durch die Überprüfung der Fehlerprotokolle des Warteschlangenmanagers ausgeführt wird. Dadurch kann möglicherweise der Prozess ermittelt werden, durch den diese Situation verursacht wird.

Der `SYSTEM.CLUSTER.REPOSITORY.QUEUE` enthält die Informationen zum Cluster-Repository-Cache als Anzahl von Nachrichten. Es ist üblich, dass Nachrichten immer in dieser Warteschlange vorhanden sind, und mehr für größere Cluster. Daher ist die Tiefe der Nachrichten in dieser Warteschlange kein Problem für die Problemmeldung.

## Protokolle überwachen

Probleme, die im Cluster auftreten, zeigen möglicherweise keine externen Symptome an Anwendungen für viele Tage (und sogar Monate) an, nachdem das Problem aufgrund des Zwischenspeicherns von Informationen und der verteilten Clusterbildung ursprünglich aufgetreten ist. Das ursprüngliche Problem wird jedoch häufig in den IBM WebSphere MQ -Fehlerprotokollen gemeldet. Aus diesem Grund ist es wichtig, diese Protokolle für alle Nachrichten, die sich auf das Clustering beziehen, aktiv zu überwachen. Diese Nachrichten müssen gelesen und verstanden werden, wobei gegebenenfalls Maßnahmen ergriffen werden müssen.

Beispiel: Eine Unterbrechung der Kommunikation mit einem WS-Manager in einem Cluster kann dazu führen, dass bestimmte Clusterressourcen gelöscht werden, da Cluster die Clusterressourcen regelmäßig erneut überprüfen, indem sie die Informationen erneut veröffentlichen. Eine Warnung vor einem potenziell auftretenden Ereignis wird von der Nachricht `AMQ9465` gemeldet. Diese Nachricht weist darauf hin, dass das Problem untersucht werden muss.

## Besondere Hinweise zum Lastausgleich

Wenn die Clusterauslastung zwischen zwei oder mehr Instanzen einer Warteschlange verteilt wird, müssen konsumierende Anwendungen die Nachrichten in jeder der Instanzen verarbeiten. Wenn eine oder mehrere der konsumierenden Anwendungen die Verarbeitung von Nachrichten beenden oder stoppt, ist es möglich, dass das Clustering weiterhin Nachrichten an diese Instanzen der Warteschlange sendet. In dieser Situation werden diese Nachrichten erst verarbeitet, wenn die Anwendungen wieder ordnungsgemäß funktionieren. Aus diesem Grund ist die Überwachung der Anwendungen ein wichtiger Bestandteil der Lösung, und es müssen Maßnahmen ergriffen werden, um die Nachrichten in dieser Situation erneut zu leiten. Ein Beispiel für einen Mechanismus zur Automatisierung dieser Überwachung finden Sie im folgenden Beispiel: [Das Beispielprogramm zur Überwachung von Clusterwarteschlangen \(AMQSCLM\)](#).

## Verfügbarkeit, Wiederherstellung und Neustart

---

Sie können die Verfügbarkeit Ihrer Anwendungen erhöhen, indem Sie dafür sorgen, dass Warteschlangen auch bei Ausfall eines Warteschlangenmanagers verfügbar bleiben und Nachrichten nach einem Server- oder Speicherausfall wiederhergestellt werden.

Sie können die Verfügbarkeit von Clientanwendungen erhöhen, indem Sie über die Clientverbindungswiederholung dafür sorgen, dass ein Client nach dem Ausfall eines Warteschlangenmanagers automatisch auf einen anderen Warteschlangenmanager in einer Gruppe von Warteschlangenmanagern oder auf die neue aktive Instanz eines Multi-Instanz-Warteschlangenmanagers umgeschaltet wird. Die automatische Wiederherstellung einer Clientverbindung wird von WebSphere MQ Classes for Java nicht unterstützt.

Unter Windows implementieren UNIX, Linux und IBM i-Plattformen Serveranwendungen auf einem Warteschlangenmanager mit mehreren Instanzen, der für die Ausführung als einzelner Warteschlangenmanager auf mehreren Servern konfiguriert ist. Wenn der Server, auf dem die aktive Instanz ausgeführt wird, fehlschlägt, wird die Ausführung automatisch auf eine Standby-Instanz desselben Warteschlangenmanagers auf einem anderen Server umgeschaltet. Wenn Sie Serveranwendungen so konfigurieren, dass sie als WS-Manager-Services ausgeführt werden, werden sie erneut gestartet, wenn eine Standby-Instanz zur aktiven WS-Manager-Instanz wird.

Sie können WebSphere MQ als Teil einer plattformspezifischen Clusterlösung wie Microsoft Cluster Server, oder PowerHA für AIX (früher HACMP unter AIX) und andere UNIX and Linux -Clusterlösungen konfigurieren.

Eine andere Möglichkeit, die Verfügbarkeit Ihrer Serveranwendungen zu erhöhen, besteht in ihrer Bereitstellung auf mehreren Computern in einem Warteschlangenmanagercluster.

Ein Messaging-System stellt sicher, dass Nachrichten, die in das System eingegeben werden, an ihr Ziel zugestellt werden. WebSphere MQ kann die Route einer Nachricht verfolgen, wenn sie mit dem Befehl **dspmqrte** von einem Warteschlangenmanager zu einem anderen verschoben wird. Wenn ein System ausfällt, können Nachrichten je nach Art des Fehlers und der Art und Weise, wie ein System konfiguriert wird, auf verschiedene Weise wiederhergestellt werden.

WebSphere MQ stellt sicher, dass Nachrichten nicht verloren gehen, indem Wiederherstellungsprotokolle der Aktivitäten der Warteschlangenmanager verwaltet werden, die den Empfang, die Übertragung und die Zustellung von Nachrichten bearbeiten. Sie verwendet diese Protokolle für drei Arten der Wiederherstellung:

1. *Wiederherstellung erneut starten*, wenn Sie WebSphere MQ auf geplante Weise stoppen.
2. *Fehlerbehebung*, wenn ein Fehler WebSphere MQ stoppt.
3. *Datenträgerwiederherstellung*, um beschädigte Objekte wiederherzustellen.

In allen Fällen stellt die Wiederherstellung den Warteschlangenmanager in den Status zurück, in dem er sich befand, als der Warteschlangenmanager gestoppt wurde, mit der Ausnahme, dass alle inflight-Transaktionen rückgängig gemacht werden. Aus den Warteschlangen werden alle Aktualisierungen entfernt, die zu dem Zeitpunkt, zu dem der Warteschlangenmanager gestoppt wurde, in den Status 'In-Flight' waren. Bei der Wiederherstellung werden alle persistenten Nachrichten zurückgespeichert. Nicht persistente Nachrichten können während des Prozesses verloren gehen.

## Automatische Clientverbindungswiederholung

Sie können Ihre Clientanwendungen automatisch erneut verbinden, ohne zusätzlichen Code schreiben zu müssen, indem Sie eine Reihe von Komponenten konfigurieren.

Die automatische Verbindungswiederholung von Clients erfolgt *integriert*. Die Verbindung wird automatisch an einem beliebigen Punkt im Clientanwendungsprogramm wiederhergestellt und es werden alle Kennungen für geöffnete Objekte wiederhergestellt.

Im Gegensatz dazu erfordert die manuelle Verbindungswiederholung, dass die Clientanwendung eine Verbindung mit MQCONN oder MQCONNX erneut erstellt und Objekte erneut öffnet. Die automatische Clientverbindungswiederholung ist für viele, nicht jedoch für alle Clientanwendungen geeignet.

In [Tabelle 28 auf Seite 337](#) ist das früheste Release der IBM WebSphere MQ-Clientunterstützung aufgeführt, das auf einer Client-Workstation installiert werden muss. Sie müssen für eine Anwendung ein Upgrade für Client-Workstations auf eine dieser Stufen durchführen, um die automatische Clientverbindung zu verwenden. In [Tabelle 29 auf Seite 337](#) sind weitere Voraussetzungen für die Aktivierung der automatischen Clientverbindungswiederholung aufgeführt.

Mit dem Programmzugriff auf Verbindungswiederverbindungsoptionen kann eine Clientanwendung Verbindungsoptionen festlegen. Wenn eine Clientanwendung Zugriff auf Optionen zur Verbindungswiederholung hat, kann sie (außer bei JMS- und XMS-Clients) außerdem einen Ereignishandler erstellen, um Verbindungswiederholungseignisse zu bearbeiten.

Eine vorhandene Clientanwendung kann ohne erneute Kompilierung und Verlinkung von der Unterstützung für die erneute Verbindung profitieren können:



- Konfigurieren Sie bei einer JMS-fremden Anwendung die Umgebungsvariable DefRecon in der Datei mqclient.ini, um Optionen zur Verbindungswiederholung festzulegen. Verwenden Sie eine CCDT, um eine Verbindung zu einem WS-Manager herzustellen. Wenn der Client eine Verbindung zu einem Multi-Instanz-Warteschlangenmanager herstellen soll, stellen Sie die Netzadressen der aktiven und der Standby-WS-Manager-Instanzen in der CCDT bereit.
- Stellen Sie bei einem JMS-Client die Optionen zur Verbindungswiederholung in der Konfiguration der Verbindungsfactory ein. Wenn Sie den WebSphere MQ -Ressourcenadapter oder einen JMS-Client verwenden, der in eine Java EE -Umgebung integriert ist, ist die automatische Clientverbindungswiederholung möglicherweise nicht verfügbar. In einigen verwalteten Umgebungen gibt es Einschränkungen. Weitere Informationen finden Sie unter [Automatische Clientverbindungswiederholung in Java SE- und Java EE -Umgebungen verwenden](#).

**Anmerkung:** Die automatische Clientverbindungswiederholung wird von WebSphere MQ Classes for Java nicht unterstützt.

*Tabelle 28. Unterstützte Clients*

| Clientschnittstelle | Client                                                                                 | Programmzugriff auf Verbindungswiederholungsoptionen | Reconnection-Unterstützung |
|---------------------|----------------------------------------------------------------------------------------|------------------------------------------------------|----------------------------|
| Messaging-APIs      | C, C++, COBOL, nicht verwaltetes Visual Basic, XMS (nicht verwaltetes XMS auf Windows) | 7.0.1                                                | 7.0.1                      |
|                     | JMS (JSE und Java EE -Client-Container und verwaltete Container)                       | 7.0.1.3                                              | 7.0.1.3                    |
|                     | WebSphere MQ Classes for Java                                                          | Nicht unterstützt                                    | Nicht unterstützt          |
|                     | Verwaltete XMS- und verwaltete .NET-Clients: C#, Visual Basic                          | 7.1                                                  | 7.1                        |
| Andere APIs         | Windows Communication Foundation (nicht verwaltet <sup>1</sup> )                       | Nicht unterstützt                                    | 7.0.1                      |
|                     | Windows Communication Foundation (verwaltet <sup>1</sup> )                             | Nicht unterstützt                                    | Nicht unterstützt          |
|                     | Achse 1                                                                                | Nicht unterstützt                                    | Nicht unterstützt          |
|                     | Achse 2                                                                                | Nicht unterstützt                                    | 7.0.1.3                    |
|                     | HTTP (Web 2.0)                                                                         | Nicht unterstützt                                    | 7.0.1.3                    |

1. Definieren Sie den verwalteten oder nicht verwalteten Modus in der WCF-Bindungskonfiguration.

Die automatische Verbindungswiederverbindung hat die folgenden Konfigurationsanforderungen:

*Tabelle 29. Konfigurationsanforderungen für die automatische Verbindungswiederverbindung*

| Komponente                      | Voraussetzung                                  | Auswirkung der Anforderung nicht erfüllen |
|---------------------------------|------------------------------------------------|-------------------------------------------|
| WebSphere MQ-Clientinstallation | Siehe <a href="#">Tabelle 28 auf Seite 337</a> | MQRC_OPTIONS_ERROR                        |
| WebSphere MQ-Serverinstallation | Stufe 7.0.1                                    | MQRC_OPTIONS_ERROR                        |

Tabelle 29. Konfigurationsanforderungen für die automatische Verbindungswiederverbindung (Forts.)

| Komponente         | Voraussetzung                                                                                                                                                                                                                                                                              | Auswirkung der Anforderung nicht erfüllen                                                                         |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| Kanal              | SHARECNV > 0                                                                                                                                                                                                                                                                               | MQRC_ENVIRONMENT_ERROR                                                                                            |
| Anwendungsumgebung | Muss mit Threads versehen sein                                                                                                                                                                                                                                                             | MQRC_ENVIRONMENT_ERROR                                                                                            |
| MQI                | Einer von: <ul style="list-style-type: none"> <li>MQCONN, wobei MQCNO Optionen auf MQCNO_RECONNECT oder MQCNO_RECONNECT_Q_MGR gesetzt ist.</li> <li>Defrecon=YES   QMGRinmqclient.ini</li> <li>In JMS die Eigenschaft CLIENT-RECONNECTOPTIONS der Verbindungsfactory festlegen.</li> </ul> | MQCC_FAILED, wenn eine Verbindung unterbrochen wird oder der Warteschlangenmanager beendet wird oder fehlschlägt. |

In Abbildung 59 auf Seite 338 sind die wichtigsten Interaktionen zwischen Komponenten dargestellt, die bei der Clientverbindungswiederholung stattfinden.

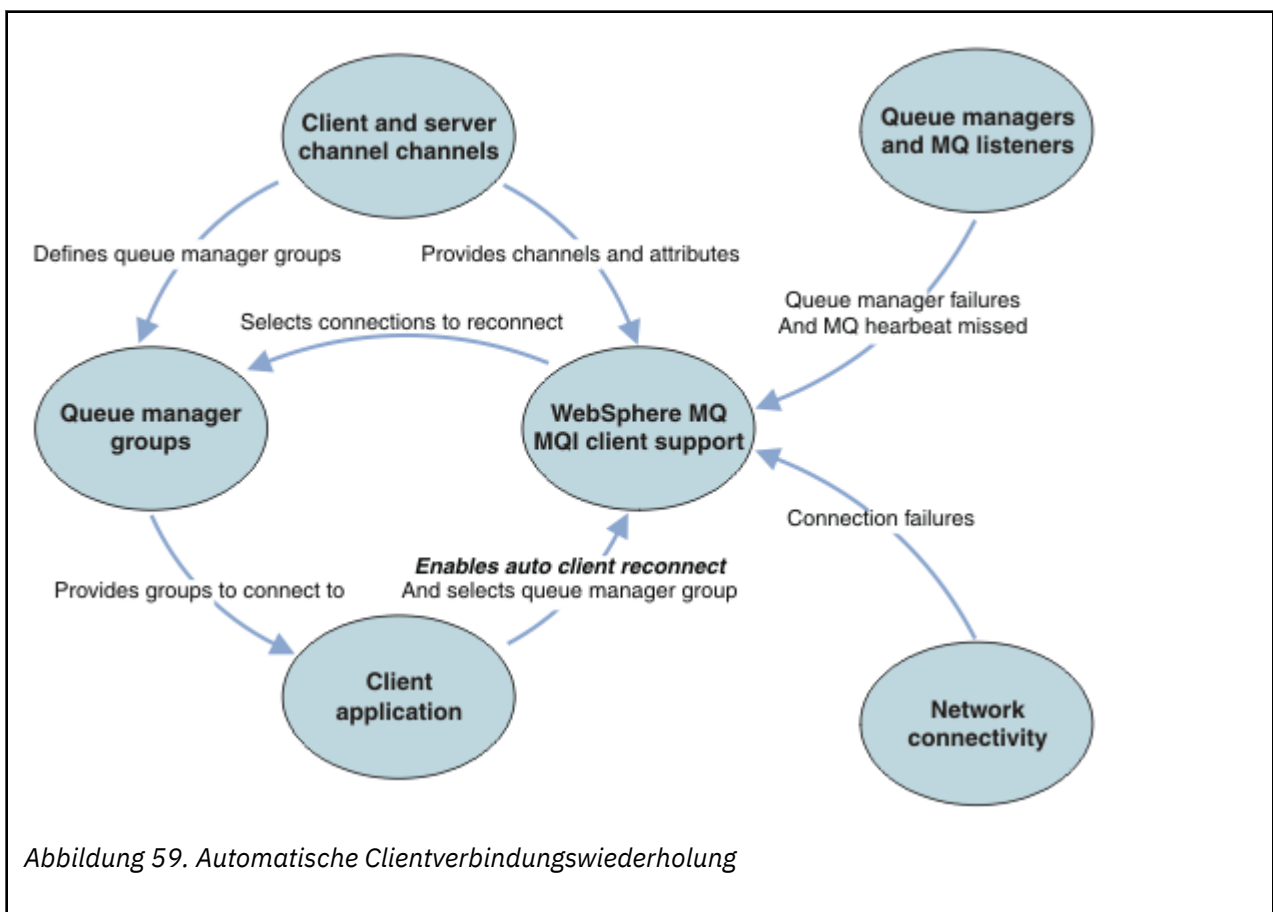


Abbildung 59. Automatische Clientverbindungswiederholung

## Clientanwendung

Die Clientanwendung ist ein IBM WebSphere MQ MQI-Client.

- Standardmäßig werden die Clients nicht automatisch erneut verbunden. Aktivieren Sie die automatische Clientverbindungswiederholung, indem Sie die Option MQCONNX MQCNO MQCNO\_RECONNECT oder MQCNO\_RECONNECT\_Q\_MGR festlegen.
- Viele Anwendungen sind so geschrieben, dass sie in der Lage sind, die automatische Verbindungswiederherstellung ohne zusätzliche Codierung zu nutzen. Aktivieren Sie die automatische Neuverbindung für vorhandene Programme, ohne Änderungen an der Codierung vorzunehmen, indem Sie das Attribut DefRecon in der Zeilengruppe für Kanäle der Konfigurationsdatei mqclient.ini festlegen.
- Verwenden Sie eine der folgenden drei Optionen:
  1. Ändern Sie das Programm so, dass die Logik nicht durch eine erneute Verbindung beeinträchtigt wird. Sie müssen beispielsweise MQI-Aufrufe innerhalb des Synchronisationspunkts absetzen und die zurückliegenden Transaktionen erneut übergeben.
  2. Fügen Sie einen Ereignishandler hinzu, um die Verbindungswiederherstellung zu ermitteln, und stellen Sie den Status der Clientanwendung wieder her, wenn die Verbindung wiederhergestellt wird.
  3. Aktivieren Sie die automatische Verbindungswiederholung nicht: Trennen Sie stattdessen den Client und geben Sie einen neuen MQCONN -oder MQCONNX MQI-Aufruf aus, um eine weitere Warteschlangenmanagerinstanz zu finden, die in derselben Warteschlangenmanagergruppe ausgeführt wird.

Weitere Informationen zu diesen drei Optionen finden Sie im Abschnitt „Anwendungswiederherstellung“ auf Seite 424.

- Die erneute Verbindung zu einem Warteschlangenmanager mit demselben Namen garantiert nicht, dass Sie mit derselben Instanz eines Warteschlangenmanagers erneut verbunden sind.

Verwenden Sie die MQCNO-Option MQCNO\_RECONNECT\_Q\_MGR, um die Verbindung zu einer Instanz desselben Warteschlangenmanagers wiederherzustellen.

- Ein Client kann einen Ereignishandler registrieren, so dass er den Status der Verbindungswiederherstellung erhalten kann. Die an den Ereignishandler übergebene MQHCONN kann nicht verwendet werden. Es werden die folgenden Ursachencodes bereitgestellt:

#### **MQRC\_RECONNECTING**

Die Verbindung ist fehlgeschlagen, und das System versucht, die Verbindung herzustellen. Sie empfangen mehrere MQRC\_RECONNECTING -Ereignisse, wenn mehrere Versuche zur Verbindungswiederherstellung unternommen werden.

#### **MQRC\_RECONNECTED**

Die Verbindungswiederherstellung wurde hergestellt und alle Handles wurden erfolgreich neu aufgebaut.

#### **MQRC\_RECONNECT\_FAILED**

Die Verbindungswiederherstellung war nicht erfolgreich.

#### **MQRC\_RECONNECT\_QMID\_MISMATCH**

Eine wiederverbindbare Verbindung hat MQCNO\_RECONNECT\_Q\_MGR angegeben und die Verbindung hat versucht, die Verbindung mit einem anderen Warteschlangenmanager wiederherzustellen.

#### **MQRC\_RECONNECT\_Q\_MGR\_REQD**

Eine Option, wie z. B. MQMO\_MATCH\_MSG\_TOKEN in einem MQGET -Aufruf, wurde im Clientprogramm angegeben, das eine erneute Verbindung zu demselben Warteschlangenmanager erfordert.

- Ein wiederverbindbarer Client ist in der Lage, die Verbindung automatisch nur *nach* der Verbindung wieder herzustellen. Das heißt, der MQCONNX -Aufruf selbst wird nicht wiederholt, wenn er fehlschlägt. Wenn Sie beispielsweise den Rückkehrcode 2543 - MQRC\_STANDBY\_Q\_MGR von MQCONNX empfangen, geben Sie den Aufruf nach einer kurzen Verzögerung erneut aus.

#### **MQRC\_RECONNECT\_INCOMPATIBLE**

Dieser Ursachencode wird zurückgegeben, wenn die Anwendung versucht, MQPMO\_LOGICAL\_ORDER (mit MQPUT und MQPUT1) oder MQGMO\_LOGICAL\_ORDER (mit MQGET) zu verwenden, wenn Optionen für die Verbindungswiederherstellung festgelegt sind. Der Grund für die Rückgabe des Ursachencodes ist es, sicherzustellen, dass Anwendungen in solchen Fällen nie wieder eine Verbindung herstellen.

## **MQRC\_CALL\_INTERRUPTED**

Dieser Ursachencode wird zurückgegeben, wenn die Verbindungsunterbrechungen während der Ausführung des Commit-Aufrufs unterbrochen werden und der Client die Verbindung wiederherstellt. Ein MQPUT-Aufruf einer persistenten Nachricht außerhalb des Synchronisationspunkts führt ebenfalls zu demselben Ursachencode, der an die Anwendung zurückgegeben wird.

## **Warteschlangenmanager mit mehreren Instanzen**

Vereinfachen Sie durch Verwenden der automatischen Verbindungswiederholung den Neustart von WebSphere MQ-Clientanwendungen, nachdem ein Mehrinstanz-Warteschlangenmanager seine Standby-Instanz aktiviert hat.

Die Standby-Instanz eines Warteschlangenmanagers mit mehreren Instanzen befindet sich in der Regel an einer anderen Netzadresse für die aktive Instanz. Geben Sie die Netzadressen der Instanzen in der Definitionstabelle für die Clientverbindung an (CCDT). Geben Sie entweder eine Liste mit Netzadressen für den Parameter **CONNAME** an oder definieren Sie mehrere Zeilen für den WS-Manager in der CCDT.

Für gewöhnlich wird die Verbindung von WebSphere MQ-Clients zu einem beliebigen Warteschlangenmanager in der Warteschlangenmanagergruppe wiederhergestellt. Manchmal soll ein WebSphere MQ-Client jedoch eine Verbindung zum selben Warteschlangenmanager wiederherstellen. Es kann eine Affinität zu einem Warteschlangenmanager haben. Sie können verhindern, dass ein Client die Verbindung zu einem anderen WS-Manager erneut herstellen kann. Legen Sie die MQCNO -Option MQCNO\_RECONNECT\_Q\_MGR fest. Der WebSphere MQ-Client schlägt fehl, wenn er eine Verbindung zu einem anderen Warteschlangenmanager wiederherstellt. Wenn Sie die Option MQCNO festlegen, schließen MQCNO\_RECONNECT\_Q\_MGR keine anderen Warteschlangenmanager in derselben Warteschlangenmanagergruppe ein. Der Client gibt einen Fehler zurück, wenn der Warteschlangenmanager, zu dem er die Verbindung herstellt, nicht mit dem Warteschlangenmanager identisch ist, mit dem er verbunden ist.

## **WS-Manager-Gruppen**

Sie können auswählen, ob die Clientanwendung immer eine Verbindung zu einem WS-Manager desselben Namens, desselben Warteschlangenmanagers oder einer Gruppe von Warteschlangenmanagern herstellt, die mit demselben QMNAME -Wert in der Clientverbindungstabelle definiert sind.

- Das Attribut *Name* des Warteschlangenmanagers, QMNAME, in der Clientkanaldefinition ist der Name der Warteschlangenmanagergruppe.
- Wenn Sie in Ihrer Clientanwendung den Wert des Parameters MQCONN oder MQCONNX QmgrName auf einen Warteschlangenmanagernamen setzen, stellt der Client nur eine Verbindung zu Warteschlangenmanagern mit diesem Namen her. Wenn Sie dem WS-Manager-Namen einen Stern (\*) voranstellen, stellt der Client eine Verbindung zu jedem WS-Manager in der Warteschlangenmanagergruppe mit demselben QMNAME -Wert her. Eine vollständige Erläuterung finden Sie im Abschnitt WS-Manager-Gruppen in der CCDT .

## **Gruppen mit gemeinsamer Warteschlange**

Die automatische Clientverbindungswiederholung mit z/OS-Gruppen mit gemeinsamer Warteschlange verwendet für die Verbindungswiederholung dieselben Mechanismen wie jede andere Umgebung. Der Client stellt die Verbindung zu derselben Auswahl von Warteschlangenmanagern wieder her, wie für die ursprüngliche Verbindung konfiguriert ist. Wenn beispielsweise eine Definitionstabelle für Clientkanal verwendet wird, sollte der Administrator sicherstellen, dass alle Einträge in der Tabelle in dieselbe z/OS-Gruppe mit gemeinsamer Warteschlange aufgelöst werden.

## **Client-und Serverkanaldefinitionen**

Client-und Serverkanaldefinitionen definieren die Gruppen von Warteschlangenmanagern, mit der eine Clientanwendung erneut eine Verbindung herstellen kann. Die Definitionen steuern die Auswahl und die Ablaufsteuerung von Verbindungswiederanschlüssen sowie andere Faktoren, wie z. B. die Sicherheit; siehe zugehörige Themen. Die relevantesten Kanalattribute, die für die Verbindungswiederverbindung zu berücksichtigen sind, werden in zwei Gruppen aufgelistet:

## **Clientverbindungsattribute**

### **Verbindungsaffinität (AFFINITY)AFFINITY**

Verbindungsaffinität.

### **Clientkanalgewichtung (CLNTWGHT)CLNTWGHT**

Gewichtung des Clientkanals.

### **Verbindungsname (CONNAME)CONNAME**

Verbindungsinformationen.

### **Intervall der Überwachungssignale (HBINT)HBINT**

Intervall der Überwachungssignale. Legen Sie das Intervall der Überwachungssignale auf dem Serververbindungskanal fest.

### **Keepalive-Intervall (KAINT)KAINT**

Keepalive-Intervall. Legen Sie das Keepalive-Intervall auf dem Serververbindungskanal fest.

Der Parameter KAINTE gilt nur für z/OS.

### **Name des Warteschlangenmanagers (QMNAME)QMNAME**

Warteschlangenmanagername.

## **Serververbindungsattribute**

### **Intervall der Überwachungssignale (HBINT)HBINT**

Intervall der Überwachungssignale. Legen Sie das Intervall der Überwachungssignale auf dem Clientverbindungskanal fest.

### **Keepalive-Intervall (KAINT)KAINT**

Keepalive-Intervall. Legen Sie das Keepalive-Intervall auf dem Clientverbindungskanal fest.

Der Parameter KAINTE gilt nur für z/OS.

KAINTE ist ein Überwachungssignal auf Netzebene und HBINT ist ein WebSphere MQ-Überwachungssignal zwischen Client und Warteschlangenmanager. Die Einstellung dieser Überwachungssignale in kürzerer Zeit dient zwei Zwecken:

1. Durch die Simulation der Aktivität auf der Verbindung wird die Netzschichtsoftware, die für das Schließen inaktiver Verbindungen verantwortlich ist, weniger wahrscheinlich die Verbindung beenden.
2. Wenn die Verbindung beendet wird, wird die Verzögerung vor dem Erkennen der unterbrochenen Verbindung verkürzt.

Das standardmäßige TCP/IP-Keepalive-Intervall beträgt zwei Stunden. Ziehen Sie in Betracht, die Attribute KAINTE und HBINT auf eine kürzere Zeit zu setzen. Gehen Sie nicht davon aus, dass das normale Verhalten eines Netzes den Anforderungen der automatischen Verbindungswiederherstellung entspricht. Einige Firewalls können z. B. eine inaktive TCP/IP-Verbindung nach weniger als 10 Minuten beenden.

## **Netz- konnektivität**

Es werden ausschließlich Netzausfälle, die vom Netz an den WebSphere MQ-Client übergeben werden, von der automatischen Verbindungswiederholung des Clients gehandhabt.

- Automatisch vom Transport durchgeführte Verbindungsabgleich sind für IBM WebSphere MQ nicht sichtbar.
- Durch das Festlegen von HBINT können Netzausfälle bearbeitet werden, die für WebSphere MQ unsichtbar sind.

## **Warteschlangenmanager und WebSphere MQ-Listener**

Die Clientwiederherstellung wird durch Serverfehler, Ausfall des Warteschlangenmanagers, Netzkonnektivitätsfehler und durch einen Administrator ausgelöst, der zu einer anderen WS-Manager-Instanz umschaltet.

- Wenn Sie einen Warteschlangenmanager mit mehreren Instanzen verwenden, tritt eine zusätzliche Ursache für die Clientwiederherstellung auf, wenn Sie die Steuerung von der aktiven WS-Manager-Instanz zu einer Standby-Instanz wechseln.

- Das Beenden eines Warteschlangenmanagers mit dem Standardbefehl **endmqm** löst keine automatische Clientverbindungswiederholung aus. Fügen Sie die Option **-r** im Befehl **endmqm** hinzu, um die automatische Clientverbindungswiederholung anzufordern, oder die Option **-s**, um nach dem Beenden zu einer Standby-Warteschlangenmanager-Instanz zu wechseln.

## Unterstützung der automatischen WebSphere MQ-Clientverbindungswiederholung

Wenn Sie die Unterstützung der automatischen Clientverbindungswiederholung im WebSphere MQ-Client verwenden, stellt die Clientanwendung die Verbindung automatisch wieder her und setzt die Verarbeitung fort, ohne einen MQCONN- oder MQCONNX-MQI-Aufruf für die Wiederherstellung der Verbindung zum Warteschlangenmanager auszugeben.

- Die automatische Neuverbindung des Clients wird durch eines der folgenden Vorkommen ausgelöst:
  - Warteschlangenmanager-Fehler
  - Beenden eines Warteschlangenmanagers und Angeben der Option **-r**(Verbindung wiederherstellen) im Befehl **endmqm**
- Die MQCONNX MQCNO-Optionen steuern, ob Sie die automatische Clientverbindungswiederholung aktiviert haben. Eine Beschreibung dieser Optionen finden Sie im Thema [Optionen zur Verbindungswiederholung](#).
- Automatische Clientverbindungsfehler MQI-Aufrufe im Namen Ihrer Anwendung, um die Verbindungskennung und die Kennungen für andere offene Objekte wiederherzustellen, so dass Ihr Programm die normale Verarbeitung wieder aufnehmen kann, nachdem es alle MQI-Fehler verarbeitet hat, die sich aus der unterbrochenen Verbindung ergeben haben. Weitere Informationen finden Sie in [„Wiederherstellung eines automatisch verbundenen Clients“](#) auf Seite 426.
- Wenn Sie ein Kanalexitprogramm für die Verbindung geschrieben haben, empfängt der Exit diese zusätzlichen MQI-Aufrufe.
- Sie können einen Umverbindungsereignishandler registrieren, der ausgelöst wird, wenn die Verbindungswiederverbindung beginnt und wann er beendet wird.

Obwohl die Verbindungswiederholung eigentlich höchstens eine Minute dauert, kann die Wiederherstellung einer Verbindung mehr Zeit beanspruchen, weil ein Warteschlangenmanager möglicherweise zahlreiche Ressourcen verwalten muss. In dieser Zeit hält die Clientanwendung möglicherweise Sperren, die nicht zu WebSphere MQ-Ressourcen gehören. Es gibt einen Zeitlimitwert, der konfiguriert werden kann, um die Zeit zu begrenzen, die ein Client auf die erneute Verbindung wartet. Der Wert (in Sekunden) wird in der Datei `mqclient.ini` festgelegt.

```
Channels:
 MQReconnectTimeout = 1800
```

Nach Ablauf des Zeitlimits werden keine Umverbindungsversuche durchgeführt. Wenn das System feststellt, dass das Zeitlimit abgelaufen ist, gibt es einen MQRC\_RECONNECT\_FAILED -Fehler zurück.

## Überwachung der Konsolnachricht

Eine Reihe von Informationsnachrichten, die vom Warteschlangenmanager oder vom Kanalinitiator ausgegeben werden, ist besonders wichtig. Diese Nachrichten weisen an sich nicht auf ein Problem hin, können aber bei der Verfolgung nützlich sein. Sie weisen auf ein potenzielles Problem hin, das möglicherweise behoben werden muss.

Das Vorhandensein dieser Nachricht kann auch darauf hinweisen, dass eine Benutzeranwendung eine große Anzahl von Nachrichten in die Seitengruppe einreicht, was ein Symptom eines größeren Problems sein kann:

- Ein Problem mit der Benutzeranwendung, die PUT-Nachrichten, wie z. B. eine unkontrollierte Schleife.
- Eine Benutzeranwendung, die die Nachrichten aus der Warteschlange abrufen, funktioniert nicht mehr.



## WebSphere MQ mit Hochverfügbarkeitskonfigurationen verwenden

Wenn Sie Ihre WebSphere MQ -Warteschlangenmanager in einer Hochverfügbarkeitskonfiguration betreiben möchten, können Sie Ihre Warteschlangenmanager so einrichten, dass sie entweder mit einem Hochverfügbarkeitsmanager wie PowerHA für AIX (früher HACMP) oder mit Microsoft Cluster Service (MSCS) arbeiten. oder mit WebSphere MQ -Warteschlangenmanagern mit mehreren Instanzen.

Sie müssen über die folgenden Konfigurationsdefinitionen informiert sein:

### Cluster aus Warteschlangenmanagern

Gruppen von mindestens zwei Warteschlangenmanagern auf einem oder mehreren Computern, die eine automatische Verbindung bereitstellen, und die gemeinsame Nutzung von Warteschlangen für Lastausgleich und Redundanz ermöglichen.

### HA-Cluster

HA-Cluster sind Gruppen von zwei oder mehr Computern und Ressourcen wie Platten und Netzen, die miteinander verbunden und so konfiguriert sind, dass ein High Availability Manager, wie z. B. HACMP (UNIX) oder MSCS (Windows), bei einem Ausfall einen *Failover* ausführt. Die Übernahme überträgt die Statusdaten von Anwendungen vom fehlerhaften Computer auf einen anderen Computer im Cluster und leitet dort ihre Operation erneut ein. Dies bietet eine hohe Verfügbarkeit von Services, die innerhalb des HA-Clusters ausgeführt werden. Die Beziehung zwischen IBM WebSphere MQ-Clustern und HA-Clustern wird in [„Beziehung zwischen HA-Clustern und WS-Manager-Clustern“](#) auf Seite 344 beschrieben.

### Warteschlangenmanager mit mehreren Instanzen

Instanzen desselben Warteschlangenmanagers, die auf zwei oder mehr Computern konfiguriert sind. Wenn Sie mehrere Instanzen starten, wird eine Instanz zur aktiven Instanz, und die anderen Instanzen werden zu Standardinstanzen. Wenn die aktive Instanz ausfällt, übernimmt eine Standby-Instanz, die auf einem anderen Computer ausgeführt wird, automatisch die Ausführung. Sie können Warteschlangenmanager mit mehreren Instanzen verwenden, um Ihre eigenen hoch verfügbaren Messaging-Systeme auf der Basis von WebSphere MQ zu konfigurieren, ohne dass eine Clustertechnologie wie HACMP oder MSCS erforderlich ist. HA-Cluster und Warteschlangenmanager mit mehreren Instanzen sind alternative Methoden zum Hochverfügbarkeitsmanagement von Warteschlangenmanagern. Kombinieren Sie sie nicht, indem Sie einen WS-Manager mit mehreren Instanzen in einem HA-Cluster einreihen.

## Unterschiede zwischen Multi-Instanz-WS-Managern und HA-Clustern

Multi-Instanz-WS-Manager und HA-Cluster sind alternative Methoden, um hohe Verfügbarkeit für Ihre Warteschlangenmanager zu erreichen. Hier sind einige Punkte, die die Unterschiede zwischen den beiden Ansätzen hervorheben.

Multi-Instanz-Warteschlangenmanager enthalten die folgenden Funktionen:

- In WebSphere MQ integrierte Basisunterstützung für Funktionsübernahme
- Schnellere Übernahme als HA-Cluster
- Einfache Konfiguration und Bedienung
- Integration mit WebSphere MQ Explorer

Die Einschränkungen bei Warteschlangenmanagern mit mehreren Instanzen umfassen:

- Hochverfügbarer, leistungsstark vernetzter Netzspeicher erforderlich
- Komplexere Netzkonfiguration, da die IP-Adresse des WS-Managers bei einem Ausfall geändert wird

HA-Cluster enthalten die folgenden Funktionen:

- Die Fähigkeit, mehrere Ressourcen, wie z. B. einen Anwendungsserver oder eine Datenbank, zu koordinieren.
- Flexibler Konfigurationsoptionen, einschließlich Cluster mit mehr als zwei Knoten
- Funktionsübernahme mehrfach ohne Bedienereingriff möglich
- Übernahme der IP-Adresse des WS-Managers als Teil der Funktionsübernahme

Zu den Einschränkungen bei HA-Clustern gehören:

- Weitere Produktkäufe und -qualifikationen sind erforderlich
- Disks, die zwischen den Knoten des Clusters umgeschaltet werden können, sind erforderlich.
- Konfiguration von HA-Clustern ist relativ komplex
- Failover ist historisch eher langsam, aber die neuesten HA-Cluster-Produkte verbessern diese
- Unnötige Failover können auftreten, wenn in den Scripts, die zum Überwachen von Ressourcen wie Warteschlangenmanagern verwendet werden, Mängel vorhanden sind.

## Beziehung zwischen HA-Clustern und WS-Manager-Clustern

WS-Manager-Cluster reduzieren die Verwaltung und stellen den Lastausgleich von Nachrichten über Instanzen von WS-Manager-Clusterwarteschlangen hinweg bereit. Sie bieten auch eine höhere Verfügbarkeit als ein einzelner Warteschlangenmanager, da die Messaging-Anwendungen nach einem Ausfall eines Warteschlangenmanagers immer noch auf überlebenslange Instanzen einer WS-Manager-Clusterwarteschlange zugreifen können. Allerdings bieten WS-Manager-Cluster allein keine automatische Erkennung des Warteschlangenmanagerfehlers und die automatische Auslösung von WS-Manager-Neustarts oder Failover. HA-Cluster stellen diese Funktionen bereit. Die beiden Typen von Clustern können gemeinsam genutzt werden, um gute Wirkung zu erreichen.

## WebSphere MQ mit einem Hochverfügbarkeitscluster unter UNIX and Linux verwenden

Sie können WebSphere MQ mit einem Hochverfügbarkeitscluster auf UNIX and Linux -Plattformen verwenden, z. B. PowerHA für AIX (früher HACMP), Veritas Cluster Server, HP Serviceguard oder einen Red Hat Enterprise Linux -Cluster mit Red Hat Cluster Suite.

Vor WebSphere MQ Version 7.0.1 wurde SupportPac MC91 bereitgestellt, um Sie bei der Konfiguration von HA-Clustern zu unterstützen. WebSphere MQ Version 7.0.1 hat einen höheren Grad an Steuerung bereitgestellt als frühere Versionen, in denen Warteschlangenmanager ihre Daten speichern. Dies vereinfacht die Konfiguration von Warteschlangenmanagern in einem HA-Cluster. Die meisten der mit SupportPac MC91 bereitgestellten Scripts sind nicht mehr erforderlich, und das SupportPac wird zurückgezogen.

Dieser Abschnitt enthält eine Einführung in „HA-Clusterkonfigurationen“ auf Seite 344, die Beziehung von HA-Clustern zu Warteschlangenmanagerclustern, „WebSphere MQ -Clients“ auf Seite 345 und „WebSphere MQ in einem HA-Cluster“ auf Seite 345 und führt Sie durch die Schritte und stellt Beispielscripts bereit, die Sie anpassen können, um Warteschlangenmanager mit einem HA-Cluster zu konfigurieren.

Weitere Informationen zu den in diesem Abschnitt beschriebenen Konfigurationsschritten finden Sie in der HA-Clusterdokumentation zu Ihrer Umgebung.

## HA-Clusterkonfigurationen

In diesem Abschnitt wird der Begriff *Knoten* verwendet, um auf die Entität zu verweisen, auf der ein Betriebssystem und die HA-Software ausgeführt werden; "Computer", "System" oder "Maschine" oder "Partition" oder "Blade" können in dieser Verwendung als Synonyme betrachtet werden. Sie können WebSphere MQ verwenden, um Standby- oder Übernahmekonfigurationen einzurichten, einschließlich der gegenseitigen Übernahme, wenn alle Clusterknoten WebSphere MQ -Workload ausführen.

Eine *Standby*-Konfiguration ist die grundlegendste HA-Clusterkonfiguration, in der ein Knoten funktioniert, während der andere Knoten nur als Standby fungiert. Der Standby-Knoten führt keine Arbeit aus und wird als inaktiv bezeichnet. Diese Konfiguration wird auch als *Cold-Standby* bezeichnet. Eine solche Konfiguration erfordert einen hohen Grad an Redundanz der Hardware. Um Hardware einzusparen, ist es möglich, diese Konfiguration so zu erweitern, dass sie mehrere Workerknoten mit einem einzigen Standby-Knoten hat. Der Punkt in diesem Punkt ist, dass der Standby-Knoten die Arbeit eines anderen Workerknotens übernehmen kann. Diese Konfiguration wird immer noch als Standby-Konfiguration und manchmal auch als "N + 1" -Konfiguration bezeichnet.



Eine *Übernahmekonfiguration* ist eine erweiterte Konfiguration, bei der alle Knoten einige Arbeiten ausführen und kritische Arbeiten im Falle eines Knotenausfalls übernommen werden können.

Eine *einseitige Übernahme* ist eine Konfiguration, bei der ein Standby-Knoten einige zusätzliche, unkritische und unbewegliche Arbeit ausführt. Diese Konfiguration gleicht einer Standby-Konfiguration, aber mit (unkritischen) Arbeiten, die vom Standby-Knoten ausgeführt werden.

Eine *gegenseitige Übernahme* -Konfiguration ist eine Konfiguration, in der alle Knoten hochverfügbare (bewegliche) Arbeiten ausführen. Dieser Typ der HA-Clusterkonfiguration wird auch manchmal als "Aktiv/Aktiv" bezeichnet, um anzuzeigen, dass alle Knoten die kritische Auslastung aktiv verarbeiten.

Bei der erweiterten Bereitschaftskonfiguration oder bei einer der beiden Übernahmekonfigurationen ist es wichtig, die Spitzenlast zu berücksichtigen, die auf einem Knoten platziert werden kann, der die Arbeit anderer Knoten übernehmen kann. Ein solcher Knoten muss über eine ausreichende Kapazität verfügen, um ein akzeptables Leistungsniveau zu gewährleisten.

## Beziehung zwischen HA-Clustern und WS-Manager-Clustern

WS-Manager-Cluster reduzieren die Verwaltung und stellen den Lastausgleich von Nachrichten über Instanzen von WS-Manager-Clusterwarteschlangen hinweg bereit. Sie bieten auch eine höhere Verfügbarkeit als ein einzelner Warteschlangenmanager, da die Messaging-Anwendungen nach einem Ausfall eines Warteschlangenmanagers immer noch auf überlebenslange Instanzen einer WS-Manager-Clusterwarteschlange zugreifen können. Allerdings bieten WS-Manager-Cluster allein keine automatische Erkennung des Warteschlangenmanagerfehlers und die automatische Auslösung von WS-Manager-Neustarts oder Failover. HA-Cluster stellen diese Funktionen bereit. Die beiden Typen von Clustern können gemeinsam genutzt werden, um gute Wirkung zu erreichen.

## WebSphere MQ -Clients

WebSphere MQ -Clients, die mit einem Warteschlangenmanager kommunizieren, der möglicherweise einem Neustart oder einer Übernahme unterliegt, müssen geschrieben werden, um eine unterbrochene Verbindung zu tolerieren, und müssen wiederholt versuchen, die Verbindung wiederherzustellen. WebSphere In MQ Version 7 wurden Features für die Verarbeitung der Definitionstabelle für Clientkanäle (Client Channel Definition Table, CCDT) eingeführt, die Sie bei der Verbindungsverfügbarkeit und beim Lastausgleich unterstützen. Diese sind jedoch nicht direkt relevant, wenn Sie mit einem Failover-System arbeiten.

Der Extended Transactional Client (ETC), der es einem WebSphere MQ -Client ermöglicht, an zweiphasigen Transaktionen teilzunehmen, muss immer eine Verbindung zu demselben Warteschlangenmanager herstellen. Der ETC kann keine Verfahren wie eine IP-Lastausgleichsfunktion verwenden, um aus einer Liste von Warteschlangenmanagern auszuwählen. Wenn Sie ein HA-Produkt verwenden, behält ein Warteschlangenmanager seine Identität (Name und Adresse) bei, je nachdem, auf welchem Knoten er ausgeführt wird, sodass die ETC-Funktion mit Warteschlangenmanagern verwendet werden kann, die unter HA-Steuerung stehen.

## WebSphere MQ in einem HA-Cluster

Alle HA-Cluster haben das Konzept einer Funktionseinheit. Dies ist eine Gruppe von Definitionen, die alle Ressourcen enthält, aus denen der hoch verfügbare Service besteht. Die Einheit der Funktionsübernahme schließt den Service selbst sowie alle anderen Ressourcen ein, von denen er abhängig ist.

HA-Lösungen verwenden unterschiedliche Bedingungen für eine Funktionseinheit:

- Unter PowerHA für AIX wird die Funktionseinheit als *Ressourcengruppe* bezeichnet.
- Auf Veritas Cluster Server wird sie als *Servicegruppe* bezeichnet.
- Auf Serviceguard wird sie als *Paket* bezeichnet.

In diesem Abschnitt wird der Begriff *Ressourcengruppe* verwendet, um eine Funktionseinheit zu definieren.

Die kleinste Funktionseinheit für WebSphere MQ ist ein Warteschlangenmanager. In der Regel enthält die Ressourcengruppe, die den Warteschlangenmanager enthält, auch gemeinsam genutzte Platten in einer Datenträgergruppe oder Plattengruppe, die ausschließlich für die Verwendung durch die Ressourcengruppe reserviert ist, sowie die IP-Adresse, die für die Verbindung zum Warteschlangenmanager verwendet wird. Es ist auch möglich, andere WebSphere MQ -Ressourcen, wie z. B. einen Listener oder einen Auslösemonitor, als separate Ressourcen oder unter der Steuerung des Warteschlangenmanagers selbst in dieselbe Ressourcengruppe einzuschließen.

Ein Warteschlangenmanager, der in einem HA-Cluster verwendet werden soll, muss über seine Daten und Protokolle auf Platten verfügen, die zwischen den Knoten im Cluster gemeinsam genutzt werden. Der HA-Cluster stellt sicher, dass nur ein Knoten im Cluster zu einem Zeitpunkt auf die Platten schreiben kann. Der HA-Cluster kann ein Überwachungsscript verwenden, um den Status des Warteschlangenmanagers zu überwachen.

Es ist möglich, sowohl für die Daten als auch für die Protokolle, die sich auf den Warteschlangenmanager beziehen, eine einzige gemeinsam genutzte Platte zu verwenden. Es ist jedoch üblich, separate gemeinsam genutzte Dateisysteme zu verwenden, so dass sie unabhängig voneinander dimensioniert und optimiert werden können.

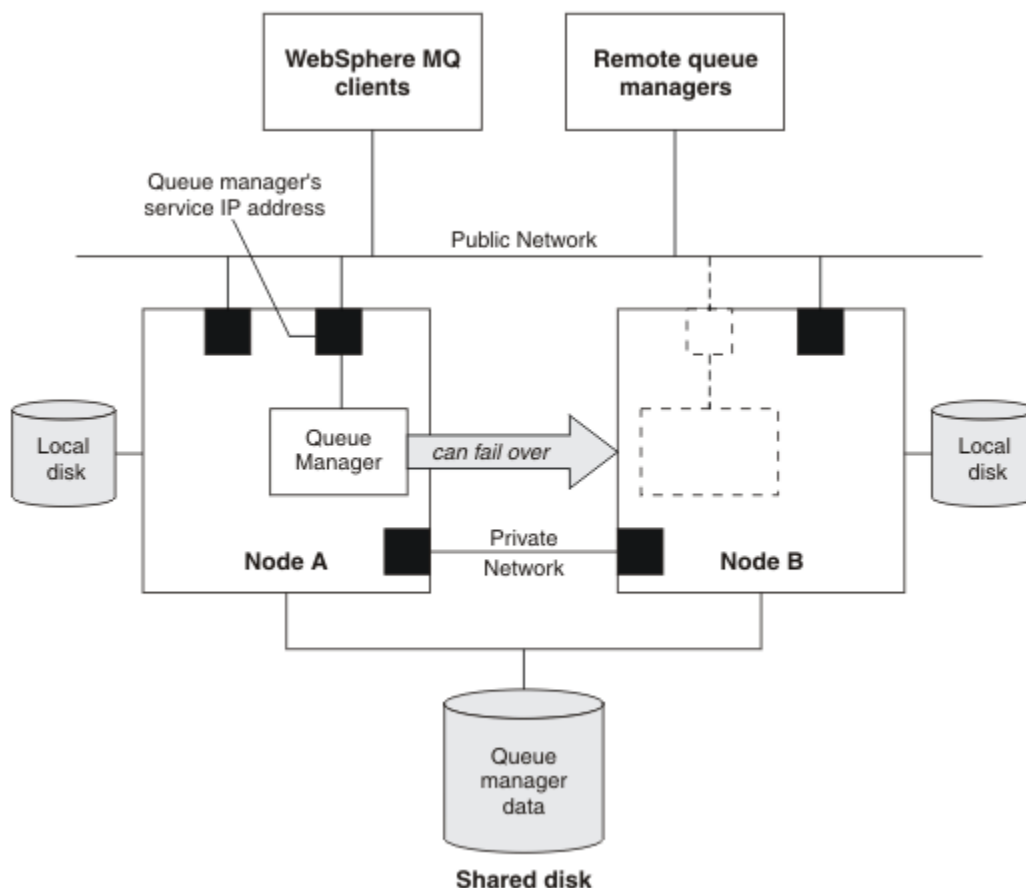


Abbildung 60. HV-Cluster

In Abbildung 1 ist ein HA-Cluster mit zwei Knoten dargestellt. Der HA-Cluster verwaltet die Verfügbarkeit eines Warteschlangenmanagers, der in einer Ressourcengruppe definiert wurde. Hierbei handelt es sich um eine Aktiv/Passiv-oder eine Cold-Standby-Konfiguration, da momentan nur ein Knoten, Knoten A, einen Warteschlangenmanager ausführt. Der WS-Manager wurde mit seinen Daten und Protokolldateien auf einer gemeinsam genutzten Platte erstellt. Der WS-Manager verfügt über eine Service-IP-Adresse, die auch vom HA-Cluster verwaltet wird. Der Warteschlangenmanager ist von der gemeinsam genutzten Platte und der zugehörigen Service-IP-Adresse abhängig. Wenn der HA-Cluster den WS-Manager von Knoten A zu Knoten B nicht mehr absetzt, verschiebt er zunächst die abhängigen Ressourcen des WS-Managers auf den Knoten B und startet dann den Warteschlangenmanager.

Wenn der HA-Cluster mehr als einen Warteschlangenmanager enthält, kann Ihre HA-Clusterkonfiguration dazu führen, dass zwei oder mehr Warteschlangenmanager nach einem Failover auf demselben Knoten ausgeführt werden. Jedem WS-Manager im HA-Cluster muss eine eigene Portnummer zugeordnet werden, die er für den jeweils aktiven Clusterknoten verwendet, der zu einem bestimmten Zeitpunkt aktiv ist.

Im Allgemeinen wird der HA-Cluster als Rootbenutzer ausgeführt. WebSphere MQ wird als Benutzer 'mqm' ausgeführt. Die Verwaltung von WebSphere MQ wird Mitgliedern der Gruppe 'mqm' erteilt. Stellen Sie sicher, dass der Benutzer und die Gruppe mqm auf allen HA-Clusterknoten vorhanden sind. Die Benutzer-ID und die Gruppen-ID müssen im gesamten Cluster konsistent sein. Die Verwaltung von WebSphere MQ durch den Benutzer 'root' ist nicht zulässig. Scripts, die Scripts starten, stoppen oder überwachen, müssen zum Benutzer 'mqm' wechseln.

**Anmerkung:** WebSphere MQ muss ordnungsgemäß auf allen Knoten installiert sein. Die ausführbaren Produktdateien können nicht gemeinsam genutzt werden.

### ***Gemeinsam genutzte Platten konfigurieren***

Ein WebSphere MQ -Warteschlangenmanager in einem HA-Cluster erfordert Datendateien und Protokoll-dateien, die gemeinsam benannte ferne Dateisysteme auf einer gemeinsam genutzten Platte sind.

Führen Sie die folgenden Schritte aus, um die gemeinsam genutzten Platten zu konfigurieren:

1. Entscheiden Sie die Namen der Mountpunkte für die Dateisysteme des Warteschlangenmanagers. Beispiel: /MQHA/qmgrname/data für die Datendateien des Warteschlangenmanagers und /MQHA/qmgrname/log für seine Protokolldateien.
2. Erstellen Sie eine Datenträgergruppe (oder eine Plattengruppe), die die Daten- und Protokolldateien des Warteschlangenmanagers enthält. Diese Datenträgergruppe wird vom HA-Cluster (HA-Cluster) in derselben Ressourcengruppe wie der Warteschlangenmanager verwaltet.
3. Erstellen Sie die Dateisysteme für die Daten und Protokolldateien des Warteschlangenmanagers in der Datenträgergruppe.
4. Erstellen Sie für jeden Knoten wiederum die Mountpunkte für die Dateisysteme, und stellen Sie sicher, dass die Dateisysteme angehängt werden können. Der mqm-Benutzer muss Eigner der Mountpunkte sein.

Abbildung 1 zeigt ein mögliches Layout für einen WS-Manager in einem HA-Cluster. Die Daten- und Protokollverzeichnisse des Warteschlangenmanagers befinden sich auf der gemeinsam genutzten Platte, die unter /MQHA/QM1 angehängt ist. Diese Platte wird bei einem Failover zwischen den Knoten des HA-Clusters umgeschaltet, so dass die Daten überall dort verfügbar sind, wo der WS-Manager erneut gestartet wird. Die Datei `mq.s.ini` enthält eine Zeilengruppe für den Warteschlangenmanager QM1. Die Zeilengruppe 'Log' in der Datei `qm.ini` hat einen Wert für 'LogPath'.

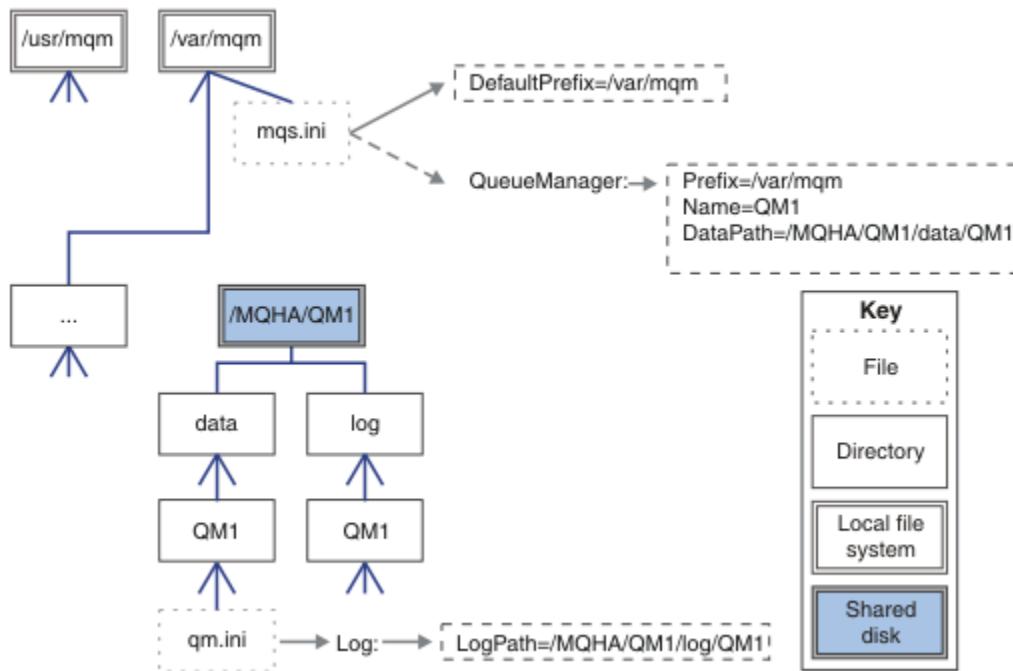


Abbildung 61. Gemeinsam genutzte data- und log-Verzeichnisse

### Warteschlangenmanager für die Verwendung in einem Cluster mit hoher Verfügbarkeit (HA-Cluster) erstellen

Der erste Schritt bei der Verwendung eines Warteschlangenmanagers in einem Hochverfügbarkeitscluster ist die Erstellung des Warteschlangenmanagers auf einem der Knoten.

Wenn Sie einen Warteschlangenmanager für die Verwendung in einem HA-Cluster erstellen möchten, wählen Sie einen der Knoten im Cluster aus, auf dem der Warteschlangenmanager erstellt werden soll. Führen Sie auf diesem Knoten die folgenden Schritte aus:

1. Hängen Sie die Dateisysteme des Warteschlangenmanagers auf dem Knoten an.
2. Erstellen Sie den WS-Manager mit dem Befehl `crtmqm`. Beispiel:
 

```
crtmqm -md /MQHA/qmgrname/data -ld /MQHA/qmgrname/log qmgrname
```
3. Starten Sie den Warteschlangenmanager manuell mit dem Befehl `strmqm`.
4. Führen Sie eine beliebige Erstkonfiguration des Warteschlangenmanagers aus, z. B. die Erstellung von Warteschlangen und Kanälen und die Einstellung des Warteschlangenmanagers, um einen Listener automatisch zu starten, wenn der Warteschlangenmanager gestartet wird.
5. Stoppen Sie den WS-Manager mit dem Befehl `endmqm`.
6. Verwenden Sie den Befehl `dspmqinf`, um den Befehl `addmqinf` anzuzeigen, den Sie in einer späteren Task verwenden können, die im Abschnitt „Warteschlangenmanager-Konfigurationsinformationen zu anderen Knoten in einem Cluster mit hoher Verfügbarkeit (HA-Cluster) hinzufügen“ auf Seite 349 dokumentiert ist:

```
dspmqinf -o command qmgrname
```

Dabei ist `qmgrname` der Name des Warteschlangenmanagers.

7. Der angezeigte Befehl `addmqinf` wird ähnlich wie im folgenden Beispiel aussehen:

```
addmqinf -sQueueManager -vName=qmgrname -vDirectory= \
-vPrefix=/var/mqm -vDataPath=/MQHA/qmgrname/data/qmgrname
```

Notieren Sie den angezeigten Befehl sorgfältig.

8. Entladen Sie die Dateisysteme des Warteschlangenmanagers.

Sie sind jetzt bereit, die in „Warteschlangenmanager-Konfigurationsinformationen zu anderen Knoten in einem Cluster mit hoher Verfügbarkeit (HA-Cluster) hinzufügen“ auf Seite 349 beschriebenen Schritte auszuführen.

### **Warteschlangenmanager-Konfigurationsinformationen zu anderen Knoten in einem Cluster mit hoher Verfügbarkeit (HA-Cluster) hinzufügen**

Sie müssen die Warteschlangenmanager-Konfiguration zu den anderen Knoten in dem HA-Cluster hinzufügen.

Bevor Sie diese Task ausführen können, müssen Sie die Schritte in „Warteschlangenmanager für die Verwendung in einem Cluster mit hoher Verfügbarkeit (HA-Cluster) erstellen“ auf Seite 348 ausgeführt haben.

Wenn Sie die Konfigurationsinformationen für den Warteschlangenmanager zu den einzelnen anderen Knoten im HA-Cluster hinzufügen möchten, führen Sie die folgenden Schritte auf jedem einzelnen dieser Knoten aus:

1. Hängen Sie die WS-Manager-Dateisysteme an.
2. Fügen Sie die Warteschlangenmanager-Konfigurationsinformationen zu dem Knoten hinzu, indem Sie entweder `/var/mqm/mqs.ini` direkt bearbeiten oder den Befehl **addmqinf** ausgeben, der über den Befehl **dspmqinf** in den Schritten 6 und 7 im Abschnitt „Warteschlangenmanager für die Verwendung in einem Cluster mit hoher Verfügbarkeit (HA-Cluster) erstellen“ auf Seite 348 angezeigt wurde.
3. Starten und stoppen Sie den WS-Manager, um die Konfiguration zu überprüfen.

Die Befehle, die zum Starten und Stoppen des Warteschlangenmanagers verwendet werden, müssen von derselben IBM WebSphere MQ-Installation wie der Befehl **addmqinf** ausgegeben werden. Wenn Sie den Warteschlangenmanager von einer anderen Installation aus starten und stoppen möchten, müssen Sie zuerst die dem Warteschlangenmanager zugeordnete Installation mit dem Befehl **setmqm** festlegen. Weitere Informationen finden Sie in [setmqm](#).

4. Entpacken Sie die WS-Manager-Dateisysteme.

### **Warteschlangenmanager starten, der von einem HA-Cluster (Cluster mit hoher Verfügbarkeit) gesteuert wird**

Der WS-Manager wird im HA-Cluster als Ressource dargestellt. Der HA-Cluster muss in der Lage sein, den Warteschlangenmanager zu starten und zu stoppen. In den meisten Fällen können Sie ein Shell-Script zum Starten des Warteschlangenmanagers verwenden. Sie müssen diese Scripts an derselben Position auf allen Knoten im Cluster verfügbar machen, indem Sie entweder ein Netzdateisystem verwenden oder sie auf jede der lokalen Platten kopieren.

**Anmerkung:** Bevor Sie einen fehlgeschlagenen Warteschlangenmanager erneut starten, müssen Sie die Verbindung zu den Anwendungen von dieser Instanz des Warteschlangenmanagers trennen. Ist dies nicht der Fall, wird der WS-Manager möglicherweise nicht ordnungsgemäß erneut gestartet.

Beispiele für geeignete Shell-Scripts sind hier angegeben. Sie können diese an Ihre Anforderungen anpassen und sie zum Starten des Warteschlangenmanagers unter der Steuerung Ihres HA-Clusters verwenden.

Das folgende Shell-Script ist ein Beispiel dafür, wie Sie vom HA-Cluster-Benutzer zum mqm-Benutzer wechseln können, damit der Warteschlangenmanager erfolgreich gestartet werden kann:

```
#!/bin/ksh
A simple wrapper script to switch to the mqm user.
su mqm -c name_of_your_script $*
```

Das folgende Shell-Script ist ein Beispiel dafür, wie ein Warteschlangenmanager gestartet wird, ohne Annahmen über den aktuellen Status des Warteschlangenmanagers zu treffen. Beachten Sie, dass es

eine extrem abrupte Methode verwendet, alle Prozesse zu beenden, die zum Warteschlangenmanager gehören:

```
#!/bin/ksh
#
This script robustly starts the queue manager.
#
The script must be run by the mqm user.

The only argument is the queue manager name. Save it as QM variable
QM=$1

if [-z "$QM"]
then
 echo "ERROR! No queue manager name supplied"
 exit 1
fi

End any queue manager processes which might be running.

srchstr="(|-m)$QM *.*$"
for process in amqzmuc0 amqzma0 amqfcxba amqfqpab amqpcsea amqzlaa0 \
 amqzlsa0 runmqchi runmqlsr amqcrsta amqrrmfa amqrmppa \
 amqzfuma amqzdmaa amqzmuf0 amqzmuir0 amqzmgr0
do
 ps -ef | tr "\t" " " | grep $process | grep -v grep | \
 egrep "$srchstr" | awk '{print $2}' | \
 xargs kill -9 > /dev/null 2>&1
done

It is now safe to start the queue manager.
The stirmqm command does not use the -x flag.
stirmqm ${QM}
```

Sie können das Script ändern, um andere zugehörige Programme zu starten.

### ***Warteschlangenmanager stoppen, der von einem HA-Cluster (Cluster mit hoher Verfügbarkeit) gesteuert wird***

In den meisten Fällen können Sie ein Shell-Script verwenden, um einen Warteschlangenmanager zu stoppen. Beispiele für geeignete Shell-Scripts sind hier angegeben. Sie können diese an Ihre Anforderungen anpassen und sie verwenden, um den Warteschlangenmanager unter der Steuerung Ihres HA-Clusters zu stoppen.

Das folgende Script ist ein Beispiel für das sofortige Stoppen, ohne dass Annahmen über den aktuellen Status des Warteschlangenmanagers gemacht werden. Das Script muss vom mqm-Benutzer ausgeführt werden. Es kann daher erforderlich sein, dieses Script in ein Shell-Script einzuschließen, um den Benutzer vom HA-Clusterbenutzer zu mqm zu wechseln (ein Beispiel-Shell-Script wird in „[Warteschlangenmanager starten, der von einem HA-Cluster \(Cluster mit hoher Verfügbarkeit\) gesteuert wird](#)“ auf Seite 349 bereitgestellt):

```
#!/bin/ksh
#
The script ends the QM by using two phases, initially trying an immediate
end with a time-out and escalating to a forced stop of remaining
processes.
#
The script must be run by the mqm user.
#
There are two arguments: the queue manager name and a timeout value.
QM=$1
TIMEOUT=$2

if [-z "$QM"]
then
 echo "ERROR! No queue manager name supplied"
 exit 1
fi

if [-z "$TIMEOUT"]
then
 echo "ERROR! No timeout specified"
 exit 1
fi
```

```

for severity in immediate brutal
do
End the queue manager in the background to avoid
it blocking indefinitely. Run the TIMEOUT timer
at the same time to interrupt the attempt, and try a
more forceful version. If the brutal version fails,
nothing more can be done here.

echo "Attempting ${severity} end of queue manager '${QM}'"
case $severity in

immediate)
Minimum severity of endmqm is immediate which severs connections.
HA cluster should not be delayed by clients
endmqm -i ${QM} &
;;

brutal)
This is a forced means of stopping queue manager processes.

srchstr="(|-m)$QM *.*$"
for process in amqzmuc0 amqzma0 amqfcxba amqfqpub amqpcsea amqzlaa0 \
amqzlsa0 runmqchi runmqlsr amqcrsta amqrrmfa amqrmppa \
amqzfuma amqzdmaa amqzmuf0 amqzmur0 amqzmgr0
do
ps -ef | tr "\t" " " | grep $process | grep -v grep | \
egrep "$srchstr" | awk '{print $2}' | \
xargs kill -9 > /dev/null 2>&1
done

esac

TIMED_OUT=yes
SECONDS=0
while (($SECONDS < ${TIMEOUT}))
do
TIMED_OUT=yes
i=0
while [$i -lt 5]
do
Check for execution controller termination
srchstr="(|-m)$QM *.*$"
cnt=`ps -ef | tr "\t" " " | grep amqzma0 | grep -v grep | \
egrep "$srchstr" | awk '{print $2}' | wc -l`
i=`expr $i + 1`
sleep 1
if [$cnt -eq 0]
then
TIMED_OUT=no
break
fi
done

if [${TIMED_OUT} = "no"]
then
break
fi

echo "Waiting for ${severity} end of queue manager '${QM}'"
sleep 1
done # timeout loop

if [${TIMED_OUT} = "yes"]
then
continue # to next level of urgency
else
break # queue manager is ended, job is done
fi

done # next phase

```

## ***Warteschlangenmanager überwachen***

Es ist üblich, einen Weg für den Hochverfügbarkeitscluster bereitzustellen, um den Status des Warteschlangenmanagers in regelmäßigen Abständen zu überwachen. In den meisten Fällen können Sie hierfür ein Shell-Script verwenden. Beispiele für geeignete Shell-Scripts sind hier angegeben. Sie können diese Scripts an Ihre Anforderungen anpassen und sie verwenden, um zusätzliche Überwachungsprüfungen speziell für Ihre Umgebung zu erstellen.

Ab WebSphere MQ Version 7.1 können auf einem System mehrere Installationen von WebSphere MQ koexistieren. Weitere Informationen zu mehreren Installationen finden Sie unter [Mehrere Installationen](#). Wenn Sie das Überwachungsscript für mehrere Installationen verwenden möchten, einschließlich Installationen der Version 7.1 oder höher, müssen Sie möglicherweise einige zusätzliche Schritte ausführen. Wenn Sie über eine primäre Installation verfügen oder das Script mit Versionen vor Version 7.1 verwenden, müssen Sie `MQ_INSTALLATION_PATH` nicht angeben, um das Script zu verwenden. Andernfalls stellen die folgenden Schritte sicher, dass der `MQ_INSTALLATION_PATH` ordnungsgemäß identifiziert wird:

1. Verwenden Sie den Befehl `crtmqenv` aus einer Installation von Version 7.1, um den richtigen `MQ_INSTALLATION_PATH` für einen Warteschlangenmanager anzugeben:

```
crtmqenv -m qmname
```

Dieser Befehl gibt den richtigen `MQ_INSTALLATION_PATH`-Wert für den mit `qmname` angegebenen Warteschlangenmanager zurück.

2. Führen Sie das Überwachungsscript mit den entsprechenden `qmname` - und `MQ_INSTALLATION_PATH` -Parametern aus.

**Anmerkung:** PowerHA für AIX bietet keine Möglichkeit, dem Überwachungsprogramm für den Warteschlangenmanager einen Parameter bereitzustellen. Sie müssen für jeden WS-Manager ein separates Überwachungsprogramm erstellen, das den Namen des WS-Managers einbindet. Es folgt ein Beispiel für ein Script, das unter AIX verwendet wird, um den Namen des Warteschlangenmanagers einzubinden:

```
#!/bin/ksh
su mqm -c name_of_monitoring_script qmname MQ_INSTALLATION_PATH
```

Dabei ist `MQ_INSTALLATION_PATH` ein optionaler Parameter, der den Pfad zu der Installation von IBM WebSphere MQ angibt, dem der Warteschlangenmanager `qmname` zugeordnet ist.

Das folgende Script ist nicht robust gegenüber der Möglichkeit, dass `runmqsc` blockiert wird. In der Regel behandeln HA-Cluster ein blockiertes Überwachungsscript als Fehler und sind selbst robust gegenüber dieser Möglichkeit.

Das Script toleriert jedoch, dass sich der WS-Manager im Startstatus befindet. Dies liegt daran, dass der HA-Cluster die Überwachung des WS-Managers so schnell wie möglich gestartet hat. Einige HA-Cluster unterscheiden zwischen einer Startphase und einer aktiven Phase für Ressourcen, aber es ist notwendig, die Dauer der Startphase zu konfigurieren. Da die Zeit, die zum Starten eines Warteschlangenmanagers benötigt wird, von der Anzahl der zu ergreifenden Arbeiten abhängt, ist es schwierig, eine maximale Zeit für den Start eines Warteschlangenmanagers auszuwählen. Wenn Sie einen zu niedrigen Wert auswählen, geht der HA-Cluster fälschlicherweise davon aus, dass der WS-Manager nicht gestartet wurde, wenn er nicht vollständig ausgeführt wurde. Dies könnte zu einer endlosen Folge von Failover führen.

Dieses Script muss vom `mqm`-Benutzer ausgeführt werden. Es kann daher erforderlich sein, dieses Script in ein Shell-Script einzuschließen, um den Benutzer vom HA-Clusterbenutzer zu `mqm` zu wechseln (ein Beispiel-Shell-Script wird in „[Warteschlangenmanager starten, der von einem HA-Cluster \(Cluster mit hoher Verfügbarkeit\) gesteuert wird](#)“ auf Seite 349 bereitgestellt):

```
#!/bin/ksh
#
This script tests the operation of the queue manager.
#
An exit code is generated by the runmqsc command:
0 => Either the queue manager is starting or the queue manager is running and responds.
Either is OK.
>0 => The queue manager is not responding and not starting.
#
This script must be run by the mqm user.
QM=$1
MQ_INSTALLATION_PATH=$2

if [-z "$QM"]
then
 echo "ERROR! No queue manager name supplied"
 exit 1
fi
```



```

if [-z "$MQ_INSTALLATION_PATH"]
then
No path specified, assume system primary install or MQ level < 7.1.0.0
echo "INFO: Using shell default value for MQ_INSTALLATION_PATH"
else
echo "INFO: Prefixing shell PATH variable with $MQ_INSTALLATION_PATH/bin"
PATH=$MQ_INSTALLATION_PATH/bin:$PATH
fi

Test the operation of the queue manager. Result is 0 on success, non-zero on error.
echo "ping qmgr" | runmqsc ${QM} > /dev/null 2>&1
pingresult=$?

if [$pingresult -eq 0]
then # ping succeeded

 echo "Queue manager '${QM}' is responsive"
 result=0

else # ping failed

 # Don't condemn the queue manager immediately, it might be starting.
 srchstr="(|m)$QM *.*$"
 cnt=`ps -ef | tr "\t" " " | grep strmqm | grep "$srchstr" | grep -v grep \
 | awk '{print $2}' | wc -l`
 if [$cnt -gt 0]
 then
 # It appears that the queue manager is still starting up, tolerate
 echo "Queue manager '${QM}' is starting"
 result=0
 else
 # There is no sign of the queue manager starting
 echo "Queue manager '${QM}' is not responsive"
 result=$pingresult
 fi

fi

exit $result

```

## ***Warteschlangenmanager unter die Steuerung des Clusters mit hoher Verfügbarkeit (HA-Clusters) bringen***

Sie müssen den Warteschlangenmanager unter der Steuerung des HA-Clusters mit der IP-Adresse und den gemeinsam genutzten Platten des WS-Managers konfigurieren.

Führen Sie die folgenden Schritte aus, um eine Ressourcengruppe so zu definieren, dass sie den Warteschlangenmanager und alle zugehörigen Ressourcen enthält:

1. Erstellen Sie die Ressourcengruppe, die den Warteschlangenmanager, den Datenträger oder die Plattengruppe des Warteschlangenmanagers und die IP-Adresse des WS-Managers enthält. Die IP-Adresse ist eine virtuelle IP-Adresse, nicht die IP-Adresse des Computers.
2. Stellen Sie sicher, dass der HA-Cluster die Ressourcen zwischen den Clusterknoten korrekt umschaltet und bereit ist, den Warteschlangenmanager zu steuern.

## ***Warteschlangenmanager aus einem Cluster mit hoher Verfügbarkeit (HA-Cluster) löschen***

Möglicherweise möchten Sie einen Warteschlangenmanager von einem Knoten entfernen, der nicht mehr für die Ausführung des Warteschlangenmanagers erforderlich ist.

Führen Sie die folgenden Schritte aus, um den Warteschlangenmanager aus einem Knoten in einem HA-Cluster zu entfernen:

1. Entfernen Sie den Knoten aus dem HA-Cluster, so dass der HA-Cluster nicht mehr versuchen wird, den Warteschlangenmanager auf diesem Knoten zu aktivieren.
2. Verwenden Sie den folgenden **rmvmqinf** -Befehl, um die Konfigurationsdaten des Warteschlangenmanagers zu entfernen:

```
rmvmqinf qmgrname
```

Verwenden Sie den Befehl **dlmqm**, um den Warteschlangenmanager vollständig zu löschen. Beachten Sie jedoch, dass die Daten und Protokolldateien des Warteschlangenmanagers dabei vollständig gelöscht werden. Nachdem Sie den WS-Manager gelöscht haben, können Sie mit dem Befehl **rmvmqinf** verbleibende Konfigurationsdaten von den anderen Knoten entfernen.

## Microsoft Cluster Service (MSCS) unterstützen

Einführung und Konfiguration von MSCS für die Unterstützung der Funktionsübernahme von virtuellen Servern.

### Diese Informationen gelten nur für WebSphere MQ für Windows .

Mit Microsoft Cluster Service (MSCS) können Sie Server zu einem *Cluster* verbinden, was eine höhere Verfügbarkeit von Daten und Anwendungen ermöglicht und die Verwaltung des Systems vereinfacht. MSCS kann Server- oder Anwendungsfehler automatisch erkennen und wiederherstellen.

MSCS unterstützt die *Funktionsübernahme* von *virtuellen Servern*, die Anwendungen, Websites, Druckwarteschlangen oder Dateifreigaben entsprechen (z. B. ihre Plattenspindeln, Dateien und IP-Adressen).

*Failover* ist der Prozess, mit dem MSCS einen Fehler in einer Anwendung auf einem Computer im Cluster feststellt und die unterbrechungsgesteuerte Anwendung ordnungsgemäß herunterfährt, die Statusdaten an den anderen Computer überträgt und die Anwendung dort erneut initialisiert.

In diesem Abschnitt werden MSCS-Cluster eingeführt und die Konfiguration der MSCS-Unterstützung in den folgenden Abschnitten beschrieben:

- [„Einführung in MSCS-Cluster“](#) auf Seite 354
- [„IBM WebSphere MQ für MSCS-Clustering einrichten“](#) auf Seite 355

Anschließend erfahren Sie in den folgenden Abschnitten, wie WebSphere MQ für MSCS-Clustering konfiguriert wird:

- [„Warteschlangenmanager für die Verwendung mit MSCS erstellen“](#) auf Seite 357
- [„Warteschlangenmanager in MSCS-Speicher versetzen“](#) auf Seite 358
- [„WS-Manager unter MSCS-Steuerung einschalten“](#) auf Seite 360
- [„Entfernen eines Warteschlangenmanagers aus MSCS-Steuerung“](#) auf Seite 367

Anschließend erhalten Sie einige nützliche Hinweise zur Verwendung von MSCS mit WebSphere MQ und ausführliche Informationen zu den MSCS-Unterstützungsprogrammen von WebSphere MQ in den folgenden Abschnitten:

- [„Hinweise und Tipps zur Verwendung von MSCS“](#) auf Seite 368
- [„IBM WebSphere MQ-Unterstützungsprogramme für MSCS“](#) auf Seite 371

### Einführung in MSCS-Cluster

MSCS-Cluster sind Gruppen aus zwei oder mehr Computern, die miteinander verbunden und so konfiguriert sind, dass MSCS bei einem Ausfall eine *Funktionsübernahme* durchführt, die Statusdaten von Anwendungen vom fehlerhaften Computer auf einen anderen Computer im Cluster überträgt und ihre Operation dort erneut einleitet.

[„WebSphere MQ mit Hochverfügbarkeitskonfigurationen verwenden“](#) auf Seite 343 enthält einen Vergleich zwischen MSCS-Clustern, Multi-Instanz-Warteschlangenmanagern und WebSphere MQ -Clustern.

In diesem Abschnitt und seinen untergeordneten Themen bedeutet der Begriff *Cluster*, wenn er von sich selbst verwendet wird, **immer** einen MSCS-Cluster. Dies unterscheidet sich von einem WebSphere MQ -Cluster, der an anderer Stelle in diesem Handbuch beschrieben wird.

Ein Cluster mit zwei Maschinen besteht aus zwei Computern (z. B. A und B), die gemeinsam mit einem Netz für den Clientzugriff über eine *virtuelle IP-Adresse* verbunden sind. Sie können auch durch ein oder mehrere private Netze miteinander verbunden sein. A und B verwenden mindestens eine Platte für die Serveranwendungen, die jeweils verwendet werden sollen. Es gibt auch eine weitere gemeinsam genutzte Platte, die ein redundantes Array unabhängiger Platten (*RAID*) der Stufe 1 für die exklusive Nutzung

von MSCS sein muss; dies wird als *Quorumplatte* bezeichnet. MSCS überwacht beide Computer, um zu überprüfen, ob die Hardware und die Software ordnungsgemäß ausgeführt werden.

In einer einfachen Konfiguration wie dieser sind auf beiden Computern alle Anwendungen installiert, aber nur Computer A läuft mit Live-Anwendungen; Computer B läuft einfach nur laufen und warten. Wenn Computer A einen beliebigen Bereich von Problemen feststellt, beendet MSCS die unterbrechungsfreie Anwendung ordnungsgemäß, überträgt die zugehörigen Statusdaten auf den anderen Computer und leitet die Anwendung dort erneut ein. Dies wird als *Funktionsübernahme* bezeichnet. Anwendungen können *clusterbewusst* ausgeführt werden, damit sie vollständig mit MSCS und Failover zusammenarbeiten.

Abbildung 62 auf Seite 355 zeigt eine typische Konfiguration für einen Cluster mit zwei Computern.

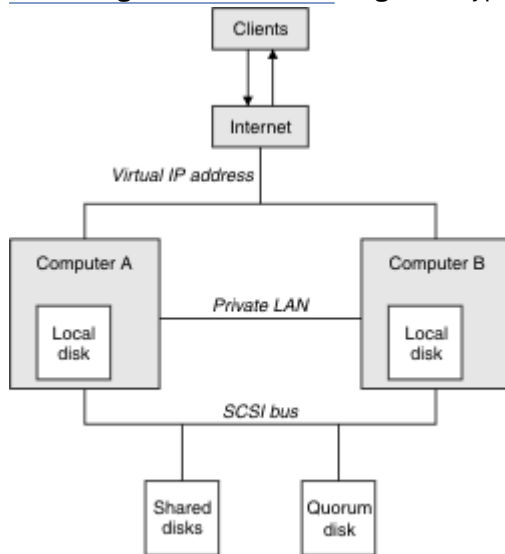


Abbildung 62. MSCS-Cluster mit zwei Computern

Jeder Computer kann unter der Steuerung von MSCS auf die gemeinsam genutzte Platte zugreifen, aber nur jeweils einen Computer. Im Falle einer Funktionsübernahme schaltet MSCS den Zugriff auf den anderen Computer um. Die gemeinsam genutzte Platte selbst ist in der Regel eine RAID-Platte, muss aber nicht vorhanden sein.

Jeder Computer ist mit dem externen Netz für den Clientzugriff verbunden, und jeder Computer verfügt über eine IP-Adresse. Ein externer Client, der mit diesem Cluster kommuniziert, kennt jedoch nur eine *virtuelle IP-Adresse* und MSCS leitet den IP-Datenverkehr im Cluster entsprechend weiter.

MSCS führt außerdem eine eigene Kommunikation zwischen den beiden Computern aus, entweder über eine oder mehrere private Verbindungen oder über das öffentliche Netz, z. B. um ihre Status mit dem Überwachungssignal zu überwachen und ihre Datenbanken zu synchronisieren.

### **IBM WebSphere MQ für MSCS-Clustering einrichten**

Zur Konfiguration von IBM WebSphere MQ für Clustering machen Sie den Warteschlangenmanager zur Einheit für die Funktionsübernahme für MSCS. Sie definieren einen Warteschlangenmanager als eine Ressource für MSCS, der sie dann überwachen kann, und sie an einen anderen Computer im Cluster übertragen kann, wenn ein Problem auftritt.

Um Ihr System hierfür einzurichten, installieren Sie zunächst IBM WebSphere MQ auf jedem Computer im Cluster.

Da der Warteschlangenmanager dem IBM WebSphere MQ-Installationsnamen zugeordnet ist, sollte der IBM WebSphere MQ-Installationsname auf allen Computern im Cluster identisch sein. Weitere Informationen finden Sie im Abschnitt [Installieren und deinstallieren](#).

Die WS-Manager selbst müssen nur auf dem Computer, auf dem sie erstellt werden, vorhanden sein. Im Falle einer Funktionsübernahme leitet der MSCS die Warteschlangenmanager auf dem anderen Computer ein. Die WS-Manager müssen jedoch ihre Protokoll- und Datendateien auf einer gemeinsam genutzten Clusterplatte und nicht auf einem lokalen Laufwerk haben. Wenn bereits ein Warteschlangenmanager auf

einem lokalen Laufwerk installiert ist, können Sie ihn mit einem Tool migrieren, das mit IBM WebSphere MQ bereitgestellt wird. Weitere Informationen hierzu finden Sie im Abschnitt [„Warteschlangenmanager in MSCS-Speicher versetzen“](#) auf Seite 358. Wenn Sie neue Warteschlangenmanager für die Verwendung mit MSCS erstellen möchten, lesen Sie den Abschnitt [„Warteschlangenmanager für die Verwendung mit MSCS erstellen“](#) auf Seite 357.

Verwenden Sie nach der Installation und Migration den MSCS-Clusteradministrator, um MSCS auf Ihre Warteschlangenmanager hinzuweisen. Weitere Informationen finden Sie im Abschnitt [„WS-Manager unter MSCS-Steuerung einschalten“](#) auf Seite 360.

Wenn Sie einen Warteschlangenmanager aus der MSCS-Steuerung entfernen möchten, verwenden Sie das in [„Entfernen eines Warteschlangenmanagers aus MSCS-Steuerung“](#) auf Seite 367 beschriebene Verfahren.

#### *Einrichtungssymmetrie*

Wenn eine Anwendung von einem Knoten zum anderen wechselt, muss er sich in der gleichen Weise verhalten, unabhängig vom Knoten. Der beste Weg, dies zu gewährleisten, ist, die Umgebungen identisch zu machen.

Wenn Sie können, können Sie einen Cluster mit identischer Hardware, Betriebssystemsoftware, Produktsoftware und Konfiguration auf jedem Computer konfigurieren. Stellen Sie insbesondere sicher, dass die gesamte erforderliche Software auf den beiden Computern in Bezug auf Version, Wartungsstufe, Support-Pacs, Pfade und Exits identisch ist und dass ein allgemeiner Namensbereich (Sicherheitsumgebung) wie in [„MSCS-Sicherheit“](#) auf Seite 356 beschrieben vorhanden ist.

#### *MSCS-Sicherheit*

Für eine erfolgreiche MSCS-Sicherheit befolgen Sie die folgenden Richtlinien.

Die Richtlinien lauten wie folgt:

- Stellen Sie sicher, dass Sie identische Softwareinstallationen auf den einzelnen Computern im Cluster haben.
- Erstellen Sie einen allgemeinen Namensbereich (Sicherheitsumgebung) im gesamten Cluster.
- Machen Sie die Knoten der MSCS-Clustermitglieder einer Domäne, in der das Benutzerkonto, das der *Clustereigner* ist, ein Domänenaccount ist.
- Erstellen Sie die anderen Benutzerkonten im Cluster auch als Domänenkonten, so dass sie auf beiden Knoten verfügbar sind. Dies ist automatisch der Fall, wenn Sie bereits über eine Domäne verfügen und die für WebSphere MQ relevanten Konten Domänenkonten sind. Wenn Sie zurzeit noch keine Domäne haben, können Sie eine *Minidomäne* einrichten, um die Clusterknoten und die relevanten Accounts zu berücksichtigen. Ihr Ziel ist es, Ihren Cluster von zwei Computern wie einer einzigen Rechenressource zu gestalten.

Denken Sie daran, dass ein Konto, das auf einem Computer lokal ist, nicht auf dem anderen Computer vorhanden ist. Selbst wenn Sie ein Konto mit demselben Namen auf dem anderen Computer erstellen, ist die zugehörige Sicherheits-ID (SID) unterschiedlich. Wenn Ihre Anwendung also auf den anderen Knoten verschoben wird, sind die Berechtigungen auf diesem Knoten nicht vorhanden.

Während eines Failover oder einer Verschiebung stellt die MSCS-Unterstützung von WebSphere MQ sicher, dass alle Dateien, die Warteschlangenmanagerobjekte enthalten, über entsprechende Berechtigungen auf dem Zielknoten verfügen. Explizit überprüft der Code, ob die Gruppen "Administratoren" und "mqm" und der Account "SYSTEM" die vollständige Kontrolle haben, und dass, wenn Everyone den Lesezugriff auf den alten Knoten hätte, diese Berechtigung auf dem Zielknoten hinzugefügt wird.

Sie können einen Domänenaccount verwenden, um Ihren WebSphere MQ -Service auszuführen. Stellen Sie sicher, dass sie in der lokalen mqm-Gruppe auf jedem Computer im Cluster vorhanden ist.

#### *Mehrere WS-Manager mit MSCS verwenden*

Wenn Sie mehr als einen Warteschlangenmanager auf einem Computer ausführen, können Sie eine dieser Konfigurationen auswählen.

Die Setups sind wie folgt:

- Alle WS-Manager in einer einzigen Gruppe. Wenn in dieser Konfiguration ein Problem mit einem WS-Manager auftritt, werden alle Warteschlangenmanager in der Gruppenübernahme auf den anderen Computer als Gruppe überstellt.
- Ein einzelner WS-Manager in jeder Gruppe. Wenn in dieser Konfiguration ein Problem mit dem Warteschlangenmanager auftritt, scheitert es allein an dem anderen Computer, ohne die anderen Warteschlangenmanager zu beeinträchtigen.
- Eine Mischung aus den ersten beiden Setups.

#### *Clustermodi*

Es gibt zwei Modi, in denen Sie ein Clustersystem mit WebSphere MQ ausführen können: Aktiv/Passiv oder Aktiv/Aktiv.

**Anmerkung:** Wenn Sie MSCS zusammen mit Microsoft Transaction Server (COM+) verwenden, können Sie den Aktiv/Aktiv-Modus nicht verwenden.

### **Aktiv/Passiver Modus**

Im Modus 'Aktiv/Passiv' verfügt Computer A über die aktive Anwendung und Computer B als Sicherung, die nur verwendet wird, wenn MSCS ein Problem feststellt.

Sie können diesen Modus nur mit einer gemeinsam genutzten Platte verwenden, aber wenn eine Anwendung eine Funktionsübernahme (Failover) bewirkt, müssen **alle** Anwendungen als Gruppe übertragen werden (da nur ein einziger Computer gleichzeitig auf die gemeinsam genutzte Platte zugreifen kann).

Sie können MSCS mit A als *bevorzugten* Computer konfigurieren. Wenn dann der Computer A repariert oder ausgetauscht wurde und wieder ordnungsgemäß funktioniert, erkennt MSCS dies und schaltet die Anwendung automatisch wieder auf Computer A um.

Wenn Sie mehr als einen Warteschlangenmanager ausführen, sollten Sie eine separate gemeinsam genutzte Platte für die einzelnen WS-Manager verwenden. Anschließend wird jeder WS-Manager in eine separate Gruppe in MSCS gestellt. Auf diese Weise kann jeder WS-Manager eine Funktionsübernahme durch den anderen Computer ohne Auswirkungen auf die anderen WS-Manager erreichen.

### **Aktiv/Aktiver Modus**

Im Modus Aktiv/Aktiv verfügen Computer A und B beide über aktive Anwendungen, und die Gruppen auf jedem Computer sind so konfiguriert, dass sie den anderen Computer als Sicherung verwenden. Wenn auf Computer A ein Fehler festgestellt wird, überträgt MSCS die Zustandsdaten an Computer B und leitet die Anwendung dort erneut ein. Computer B führt dann seine eigene Anwendung und A's aus.

Für diese Konfiguration benötigen Sie mindestens zwei gemeinsam genutzte Platten. Sie können MSCS mit A als bevorzugten Computer für A-Anwendungen konfigurieren und B als bevorzugten Computer für B-Anwendungen. Nach dem Failover und der Reparatur wird jede Anwendung automatisch wieder auf ihrem eigenen Computer ausgeführt.

Für WebSphere MQ bedeutet dies, dass Sie beispielsweise zwei Warteschlangenmanager ausführen können, jeweils einen auf A und B, wobei jeder die volle Leistung seines eigenen Computers nutzt. Nach einem Fehler auf dem Computer A werden beide Warteschlangenmanager auf Computer B ausgeführt. Dies bedeutet, die Leistung des einen Computers zu teilen, mit einer verringerten Fähigkeit, große Datenmengen mit Geschwindigkeit zu verarbeiten. Ihre kritischen Anwendungen stehen jedoch weiterhin zur Verfügung, während Sie den Fehler auf A finden und reparieren.

### ***Warteschlangenmanager für die Verwendung mit MSCS erstellen***

Mit dieser Prozedur wird sichergestellt, dass ein neuer Warteschlangenmanager so erstellt wird, dass er für die Vorbereitung und Platzierung unter MSCS-Steuerung geeignet ist.

Sie beginnen, indem Sie den Warteschlangenmanager mit allen zugehörigen Ressourcen auf einem lokalen Laufwerk erstellen und anschließend die Protokolldateien und Datendateien auf eine gemeinsam genutzte Platte migrieren. (Sie können diese Operation umkehren.) Versuchen Sie **nicht**, einen WS-Manager mit seinen Ressourcen auf einem gemeinsam genutzten Laufwerk zu erstellen.

Sie können einen Warteschlangenmanager zur Verwendung mit MSCS auf zwei Arten erstellen: entweder über eine Eingabeaufforderung oder im WebSphere MQ Explorer. Der Vorteil bei der Verwendung einer Eingabeaufforderung besteht darin, dass der Warteschlangenmanager *gestoppt* erstellt und auf *manueller Start* gesetzt ist, der für MSCS bereit ist. (IBM WebSphere MQ Explorer startet automatisch einen neuen Warteschlangenmanager und stellt ihn so ein, dass er nach dem Erstellen automatisch gestartet wird. Sie müssen das ändern.)

## WS-Manager über eine Eingabeaufforderung erstellen

Führen Sie die folgenden Schritte aus, um einen WS-Manager über eine Eingabeaufforderung zu erstellen, der für MSCS verwendet werden soll:

1. Stellen Sie sicher, dass die Umgebungsvariable MQSPREFIX so eingestellt ist, dass sie auf ein lokales Laufwerk (z. B. C:\WebSphere MQ) verweist. Wenn Sie dies ändern, führen Sie einen Warmstart der Maschine durch, damit das Systemkonto die Änderung aufnimmt. Wenn Sie die Variable nicht setzen, wird der Warteschlangenmanager im WebSphere MQ -Standardverzeichnis für Warteschlangenmanager erstellt.
2. Erstellen Sie den WS-Manager mit dem Befehl **crtmqm** . Wenn Sie beispielsweise einen WS-Manager mit dem Namen mscs\_test im Standardverzeichnis erstellen möchten, verwenden Sie Folgendes:

```
crtmqm mscs_test
```

3. Fahren Sie mit dem Abschnitt „[Warteschlangenmanager in MSCS-Speicher versetzen](#)“ auf Seite 358 fort.

## Warteschlangenmanager mit WebSphere MQ Explorer erstellen

Führen Sie die folgenden Schritte aus, um mithilfe von IBM WebSphere MQ Explorer einen Warteschlangenmanager zu erstellen, der mit MSCS verwendet werden soll:

1. Starten Sie IBM WebSphere MQ Explorer über das Startmenü.
2. Erweitern Sie in der Navigator die Baumknoten, um den Baumknoten Queue Managers zu suchen.
3. Klicken Sie mit der rechten Maustaste auf den Baumknoten Queue Managers und wählen Sie New->Queue Manager aus. Die Anzeige 'Warteschlangenmanager erstellen' wird angezeigt.
4. Füllen Sie das Dialogfeld aus (Schritt 1) und klicken Sie dann auf Next>.
5. Füllen Sie den Dialog aus (Schritt 2) und klicken dann auf Next>.
6. Füllen Sie das Dialogfeld (Schritt 3) aus und stellen Sie sicher, dass Start Queue Manager und Create Server Connection Channel nicht ausgewählt sind. Klicken Sie dann auf Next>.
7. Vervollständigen Sie den Dialog (Schritt 4) und klicken Sie anschließend auf Finish.
8. Fahren Sie mit dem Abschnitt „[Warteschlangenmanager in MSCS-Speicher versetzen](#)“ auf Seite 358 fort.

## Warteschlangenmanager in MSCS-Speicher versetzen

Mit dieser Prozedur wird ein vorhandener Warteschlangenmanager konfiguriert, um ihn für die MSCS-Steuerung geeignet zu machen.

Um dies zu erreichen, verschieben Sie die Protokolldateien und Datendateien auf gemeinsam genutzte Platten, um sie im Falle eines Fehlers für den anderen Computer verfügbar zu machen. Der vorhandene Warteschlangenmanager kann z. B. Pfade wie C:\WebSphere MQ\log\<QMname> und C:\WebSphere MQ\qmgrs\<QMname> haben. Versuchen Sie **nicht** , die Dateien manuell zu verschieben. Verwenden Sie das Dienstprogramm, das im Rahmen der WebSphere MQ MSCS-Unterstützung bereitgestellt wird, wie in diesem Abschnitt beschrieben.

Wenn der zu verschiebende Warteschlangenmanager SSL-Verbindungen verwendet und sich das SSL-Schlüsselrepository im Datenverzeichnis des Warteschlangenmanagers auf der lokalen Maschine befindet, wird das Schlüsselrepository mit dem Rest des Warteschlangenmanagers auf die gemeinsam genutzte Platte verschoben. Standardmäßig ist das Warteschlangenmanagerattribut, das die Position des SSL-

Schlüsselrepositorys (SSLKEYR) angibt, auf `MQ_INSTALLATION_PATH\qmgrs\QMGRNAME\ssl\keyge-` setzt, das sich unter dem Datenverzeichnis des Warteschlangenmanagers befindet. `MQ_INSTALLATION_PATH` steht für das übergeordnete Verzeichnis, in dem WebSphere MQ installiert ist. Mit dem Befehl `hamvmqm` wird dieses WS-Manager-Attribut nicht geändert. In dieser Situation müssen Sie das Warteschlangenmanagerattribut `SSLKEYR` mit dem IBM WebSphere MQ Explorer oder mit dem MQSC-Befehl `ALTER QMGR` ändern, um auf die neue SSL-Schlüsselrepositorydatei zu verweisen.

Das Verfahren ist wie folgt:

1. Fahren Sie den Warteschlangenmanager herunter, und prüfen Sie, ob keine Fehler aufgetreten sind.
2. Wenn die Protokolldateien oder die Warteschlangendateien des Warteschlangenmanagers bereits auf einer gemeinsam genutzten Platte gespeichert sind, überspringen Sie den Rest dieser Prozedur und fahren Sie direkt mit „WS-Manager unter MSCS-Steuerung einschalten“ auf Seite 360 fort.
3. Führen Sie ein vollständiges Backup aller Warteschlangen- und Protokolldateien durch und speichern Sie diese Sicherheitskopie in einem sicheren Bereich (in Abschnitt „Warteschlangenmanagerprotokolldateien“ auf Seite 370 wird erläutert, warum dies wichtig ist).
4. Wenn Sie bereits über eine geeignete gemeinsam genutzte Plattenressource verfügen, fahren Sie mit Schritt 6 fort. Andernfalls verwenden Sie den MSCS-Cluster-Administrator, um eine Ressource des Typs *gemeinsam genutzte Platte* mit einer ausreichenden Kapazität zum Speichern der WS-Manager-Protokolldateien und der Daten (Warteschlangen-) Dateien zu erstellen.
5. Testen Sie die gemeinsam genutzte Platte, indem Sie den MSCS-Cluster-Administrator verwenden, um ihn von einem Clusterknoten in den anderen und wieder zurück zu versetzen.
6. Stellen Sie sicher, dass die gemeinsam genutzte Platte online auf dem Clusterknoten ist, auf dem das WS-Manager-Protokoll und die Datendateien lokal gespeichert werden.
7. Führen Sie das Dienstprogramm aus, um den WS-Manager wie folgt zu verschieben:

```
hamvmqm /m qmname /dd "e:\
WebSphere MQ" /ld "e:\
WebSphere MQ\log"
```

Ersetzen Sie *qmname* durch den Namen Ihres Warteschlangenmanagers, *e* durch den Buchstaben Ihres gemeinsam genutzten Plattenlaufwerks und *WebSphere MQ* durch das ausgewählte Verzeichnis. Die Verzeichnisse werden erstellt, wenn sie noch nicht vorhanden sind.

8. Testen Sie den Warteschlangenmanager über IBM WebSphere MQ Explorer, um sicherzustellen, dass er funktioniert. Beispiel:
  - a. Klicken Sie mit der rechten Maustaste auf den Knoten der Baumstruktur des Warteschlangenmanagers und wählen Sie dann `Startaus`. Der WS-Manager wird gestartet.
  - b. Klicken Sie mit der rechten Maustaste auf den Baumknoten `Queues`, wählen Sie `New->Local Queue . . .` aus und geben Sie der Warteschlange einen Namen.
  - c. Klicken Sie auf `Finish`.
  - d. Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie `Put Test Message . . .` aus. Die Anzeige 'Testnachricht einreihen' wird angezeigt.
  - e. Geben Sie einen Nachrichtentext ein, klicken Sie dann auf `Put Test Message` und schließen Sie die Anzeige.
  - f. Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie `Browse Messa-ges . . .` aus. Die Anzeige 'Nachrichten-Browser' wird angezeigt.
  - g. Stellen Sie sicher, dass sich Ihre Nachricht in der Warteschlange befindet und klicken Sie anschließend auf `Close`. Die Anzeige 'Nachrichten-Browser' wird geschlossen.
  - h. Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie `Clear Messa-ges . . .` aus. Die Nachrichten in der Warteschlange werden gelöscht.
  - i. Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie `Delete . . .` aus. Eine Bestätigungsanzeige wird angezeigt. Klicken Sie auf `OK`. Die Warteschlange wird gelöscht.



- j. Klicken Sie mit der rechten Maustaste auf den Knoten der Baumstruktur des Warteschlangenmanagers und wählen Sie dann Stop. . . aus. Die Anzeige 'End Queue Manager' wird angezeigt.
  - k. Klicken Sie auf OK. Der Warteschlangenmanager wird gestoppt.
9. Stellen Sie als WebSphere MQ -Administrator sicher, dass das Startattribut des Warteschlangenmanagers auf manuell gesetzt ist. Setzen Sie im IBM WebSphere MQ Explorer das Startfeld in der Anzeige mit den Eigenschaften des Warteschlangenmanagers auf manuell .
  10. Fahren Sie mit dem Abschnitt „WS-Manager unter MSCS-Steuerung einschalten“ auf Seite 360 fort.

### **WS-Manager unter MSCS-Steuerung einschalten**

Die Tasks, die zum Platzieren eines Warteschlangenmanagers unter MSCS-Steuerung gehören, einschließlich vorausgesetzter Tasks.

### **Bevor Sie einen Warteschlangenmanager unter die MSCS-Steuerung stellen**

Führen Sie die folgenden Tasks aus, bevor Sie einen Warteschlangenmanager unter die MSCS-Steuerung stellen:

1. Stellen Sie sicher, dass IBM WebSphere MQ und der zugehörige MSCS-Support auf beiden Maschinen im Cluster installiert sind und dass die Software auf jedem Computer identisch ist, wie in „IBM WebSphere MQ für MSCS-Clustering einrichten“ auf Seite 355 beschrieben.
2. Mit dem Dienstprogramm **haregtyp** können Sie WebSphere MQ als MSCS-Ressourcentyp auf allen Clusterknoten registrieren. Weitere Informationen enthält der Abschnitt „IBM WebSphere MQ-Unterstützungsprogramme für MSCS“ auf Seite 371.
3. Wenn Sie den Warteschlangenmanager noch nicht erstellt haben, lesen Sie den Abschnitt „Warteschlangenmanager für die Verwendung mit MSCS erstellen“ auf Seite 357.
4. Wenn Sie den Warteschlangenmanager erstellt haben oder er bereits vorhanden ist, stellen Sie sicher, dass Sie die Prozedur in „Warteschlangenmanager in MSCS-Speicher versetzen“ auf Seite 358 ausgeführt haben.
5. Stoppen Sie den Warteschlangenmanager, falls er aktiv ist. Verwenden Sie dazu entweder eine Eingabeaufforderung oder den IBM WebSphere MQ-Explorer.
6. Testen Sie die MSCS-Operation der gemeinsam genutzten Laufwerke, bevor Sie mit einer der folgenden Windows -Prozeduren in diesem Abschnitt fortfahren.

### **Windows Server 2012**



**Achtung:** MSCS-Unterstützung wird in WebSphere MQ 7.5 über eine 32-Bit-DLL bereitgestellt. Aufgrund einer Einschränkung in Windows 2012 wird der IBM WebSphere MQ -Warteschlangenmanager nach einem Neustart nicht übernommen.

Microsoft hat die Verwendung von 32-Bit-DLLs mit Windows 2012 nicht weiter unterstützt. Daher ist derzeit kein Betriebssystemfix für dieses Problem verfügbar. IBM stellt keine 64-Bit-Bibliothek für IBM WebSphere MQ 7.5 bereit.

Ab IBM MQ 8.0 ist eine 64-Bit-Bibliothek verfügbar, sodass Sie diese Version des Produkts für die vollständige MSCS-Funktionalität mit Windows 2012 und höher verwenden müssen.

Gehen Sie wie folgt vor, um einen Warteschlangenmanager unter Windows Server 2012 unter MSCS-Steuerung zu stellen:

1. Melden Sie sich am Cluster-Knotencomputer an, auf dem sich der Warteschlangenmanager befindet, oder melden Sie sich als Benutzer mit Clusterverwaltungsberechtigungen an einer fernen Workstation an und stellen Sie eine Verbindung zu dem Clusterknoten her, der als Host für den Warteschlangenmanager fungiert.
2. Starten Sie das Tool Failover Cluster Management.
3. Klicken Sie mit der rechten Maustaste auf **Failover Cluster Management > Connect Cluster ...** , um eine Verbindung zum Cluster zu öffnen.



4. Im Gegensatz zu dem Gruppenschema, das im MSCS-Clusteradministrator unter früheren Versionen von Windows verwendet wird, verwendet das Tool 'Failover Cluster Management' das Konzept von Services und Anwendungen. Ein konfigurierter Service oder eine konfigurierte Anwendung enthält alle Ressourcen, die erforderlich sind, damit eine Anwendung in einem Cluster zusammengefasst wird. Sie können einen WS-Manager wie folgt unter MSCS konfigurieren:

- a. Klicken Sie auf den Cluster, und wählen Sie **Rolle konfigurieren** aus, um den Konfigurationsassistenten zu starten.
- b. Wählen Sie in der Anzeige "Service oder Anwendung auswählen" die Option **Anderer Server** aus.
- c. Wählen Sie eine geeignete IP-Adresse als Clientzugriffspunkt aus.

Bei dieser Adresse muss es sich um eine nicht verwendete IP-Adresse handeln, die von Clients und anderen Warteschlangenmanagern verwendet werden soll, um eine Verbindung zum *virtuellen* Warteschlangenmanager herzustellen. Bei dieser IP-Adresse handelt es sich nicht um die normale (statische) Adresse eines der beiden Knoten. Es handelt sich um eine zusätzliche Adresse, die zwischen den beiden Knoten *floats* ist. Obwohl MSCS das Routing dieser Adresse handhabt, prüft es **nicht**, ob die Adresse erreicht werden kann.

- d. Ordnen Sie eine Speichereinheit für die exklusive Verwendung durch den Warteschlangenmanager zu. Diese Einheit muss als Ressourceninstanz erstellt werden, bevor sie zugeordnet werden kann.

Sie können ein Laufwerk verwenden, um sowohl die Protokolle als auch die Warteschlangendateien zu speichern, oder Sie können sie über Laufwerke hinweg aufteilen. Wenn jeder WS-Manager über eine eigene gemeinsam genutzte Platte verfügt, stellen Sie sicher, dass alle Laufwerke, die von diesem Warteschlangenmanager verwendet werden, exklusiv für diesen Warteschlangenmanager sind, d. -E., dass nichts anderes auf den Laufwerken basiert. Stellen Sie außerdem sicher, dass Sie für jedes Laufwerk, das der Warteschlangenmanager verwendet, eine Ressourceninstanz erstellen.

Der Ressourcentyp für ein Laufwerk hängt von der von Ihnen verwendeten SCSI-Unterstützung ab. Weitere Informationen finden Sie in den Anweisungen zum SCSI-Adapter. Es können bereits Gruppen und Ressourcen für die einzelnen gemeinsam genutzten Laufwerke vorhanden sein. Ist dies der Fall, müssen Sie die Ressourceninstanz für jedes Laufwerk nicht erstellen. Verschieben Sie ihn von seiner aktuellen Gruppe in die für den Warteschlangenmanager erstellte Gruppe.

Geben Sie für jede Laufwerkressource die möglichen Eigner beider Knoten an. Setzen Sie abhängige Ressourcen auf "none".

- e. Wählen Sie die Ressource **IBM MQSeries MSCS** in der Anzeige "Ressourcentyp auswählen" aus.
- f. Führen Sie die verbleibenden Schritte im Assistenten aus.

5. Bevor die Ressource online geschaltet wird, benötigt die MSCS-Ressource IBM MQSeries zusätzliche Konfigurationsschritte:

- a. Wählen Sie den neu definierten Service aus, der eine Ressource namens 'New IBM MQSeries MSCS' enthält.
- b. Klicken Sie in der MQ-Ressource auf **Eigenschaften**.
- c. Konfigurieren Sie die Ressource:

- Name; Wählen Sie einen Namen aus, der die Identifizierung des Warteschlangenmanagers, für den er gilt, vereinfacht.
- Run in a separate Resource Monitor; für eine bessere Isolation
- Possible owners; beide Knoten festlegen
- Dependencies; Fügen Sie das Laufwerk und die IP-Adresse für diesen Warteschlangenmanager hinzu.

**Warnung:** Wenn Sie diese Abhängigkeiten nicht hinzufügen, bedeutet dies, dass IBM WebSphere MQ versucht, den Warteschlangenmanagerstatus während der Failover auf die falsche Clusterplatte zu schreiben. Da viele Prozesse möglicherweise gleichzeitig versuchen, auf diese Platte zu schreiben, könnten einige IBM WebSphere MQ-Prozesse von der Ausführung blockiert werden.

- Parameters; wie folgt:

- QueueManagerName (erforderlich); der Name des Warteschlangenmanagers, den diese Resource steuern soll. Dieser WS-Manager muss auf dem lokalen Computer vorhanden sein.
- PostOnlineCommand (optional); Sie können ein Programm angeben, das ausgeführt werden soll, wenn die WS-Manager-Ressource ihren Status von offline in online ändert. Nähere Informationen finden Sie unter „[PostOnlineCommand und PreOfflineCommand](#)“ auf Seite 370.
- PreOfflineCommand (optional); Sie können ein Programm angeben, das ausgeführt werden soll, wenn die Warteschlangenmanagerressource ihren Status von "Online" in "Offline" ändert. Nähere Informationen finden Sie unter „[PostOnlineCommand und PreOfflineCommand](#)“ auf Seite 370.

**Anmerkung:** Das Abfrageintervall für *LooksAlive* wird auf den Standardwert 5000 ms gesetzt. Das Abfrageintervall für *isAlive* wird auf den Standardwert 60000 ms gesetzt. Diese Standardwerte können nur geändert werden, nachdem die Ressourcendefinition abgeschlossen wurde. Weitere Informationen finden Sie im Abschnitt „[Zusammenfassung der looksAlive- und isAlive-Abfragen](#)“ auf Seite 367.

- d. Legen Sie optional einen bevorzugten Knoten fest (aber beachten Sie die Kommentare in „[Bevorzugte Knoten verwenden](#)“ auf Seite 371).
  - e. Die *Funktionsübernehmerichtlinie* wird standardmäßig auf sinnvolle Werte gesetzt, Sie können jedoch die Schwellenwerte und Zeiträume, die *Resource Failover* und *Group Failover* steuern, so optimieren, dass sie mit den auf den Warteschlangenmanager platzierten Lasten übereinstimmen.
6. Testen Sie den Warteschlangenmanager, indem Sie ihn online in den MSCS-Clusteradministrator setzen und einer Testworkload unterziehen. Wenn Sie mit einem Testwarteschlangenmanager experimentieren, verwenden Sie den IBM WebSphere MQ-Explorer. Beispiel:
- a. Klicken Sie mit der rechten Maustaste auf den Baumknoten Queues , wählen Sie New->Local Queue . . . aus und geben Sie der Warteschlange einen Namen.
  - b. Klicken Sie auf Finish. Die Warteschlange wird erstellt und in der Inhaltsansicht angezeigt.
  - c. Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie Put Test Message . . . aus. Die Anzeige 'Testnachricht einreihen' wird angezeigt.
  - d. Geben Sie einen Nachrichtentext ein, klicken Sie dann auf Put Test Message und schließen Sie die Anzeige.
  - e. Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie Browse Messages . . . aus. Die Anzeige 'Nachrichten-Browser' wird angezeigt.
  - f. Stellen Sie sicher, dass sich Ihre Nachricht in der Warteschlange befindet, und klicken Sie anschließend auf Close . Die Anzeige 'Nachrichten-Browser' wird geschlossen.
  - g. Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie Clear Messages . . . aus. Die Nachrichten in der Warteschlange werden gelöscht.
  - h. Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie Delete . . . aus. Eine Bestätigungsanzeige wird angezeigt. Klicken Sie auf OK. Die Warteschlange wird gelöscht.
7. Testen Sie, ob der Warteschlangenmanager offline und wieder online mit dem MSCS-Cluster-Administrator ausgeführt werden kann.
8. Simulieren Sie eine Funktionsübernahme.
- Klicken Sie im MSCS-Clusteradministrator auf die Gruppe, die den Warteschlangenmanager enthält, und wählen Sie Move Group aus. Dies kann einige Minuten dauern. (Wenn Sie einen Warteschlangenmanager zu einem anderen Zeitpunkt schnell in einen anderen Knoten verschieben möchten, führen Sie die Prozedur in „[Warteschlangenmanager in MSCS-Speicher versetzen](#)“ auf Seite 358 aus.) Sie können auch mit der rechten Maustaste klicken und Initiate Failure auswählen. Die Aktion (lokaler Neustart oder Failover) hängt vom aktuellen Status und den Konfigurationseinstellungen ab.

## Windows Server 2008

Gehen Sie wie folgt vor, um einen WS-Manager unter Windows Server 2008 unter MSCS-Steuerung zu stellen:

1. Melden Sie sich am Cluster-Knotencomputer an, auf dem sich der Warteschlangenmanager befindet, oder melden Sie sich als Benutzer mit Clusterverwaltungsberechtigungen an einer fernen Workstation an und stellen Sie eine Verbindung zu dem Clusterknoten her, der als Host für den Warteschlangenmanager fungiert.
  2. Starten Sie das Tool Failover Cluster Management.
  3. Klicken Sie mit der rechten Maustaste auf **Failover-Cluster-Management > Cluster verwalten .** , um eine Verbindung zum Cluster zu öffnen.
  4. Im Gegensatz zu dem Gruppenschema, das im MSCS-Clusteradministrator unter früheren Versionen von Windows verwendet wird, verwendet das Tool 'Failover Cluster Management' das Konzept von Services und Anwendungen. Ein konfigurierter Service oder eine konfigurierte Anwendung enthält alle Ressourcen, die erforderlich sind, damit eine Anwendung in einem Cluster zusammengefasst wird. Sie können einen WS-Manager wie folgt unter MSCS konfigurieren:
    - a. Klicken Sie mit der rechten Maustaste auf **Dienste und Anwendungen > Service oder Anwendung konfigurieren ...** , um den Konfigurationsassistenten zu starten.
    - b. Wählen Sie in der Anzeige "Service oder Anwendung auswählen" die Option **Anderer Server** aus.
    - c. Wählen Sie eine geeignete IP-Adresse als Clientzugriffspunkt aus.  
 Bei dieser Adresse muss es sich um eine nicht verwendete IP-Adresse handeln, die von Clients und anderen Warteschlangenmanagern verwendet werden soll, um eine Verbindung zum *virtuellen* Warteschlangenmanager herzustellen. Bei dieser IP-Adresse handelt es sich nicht um die normale (statische) Adresse eines der beiden Knoten. Es handelt sich um eine zusätzliche Adresse, die zwischen den beiden Knoten *floats* ist. Obwohl MSCS das Routing dieser Adresse handhabt, prüft es **nicht** , ob die Adresse erreicht werden kann.
    - d. Ordnen Sie eine Speichereinheit für die exklusive Verwendung durch den Warteschlangenmanager zu. Diese Einheit muss als Ressourceninstanz erstellt werden, bevor sie zugeordnet werden kann.  
 Sie können ein Laufwerk verwenden, um sowohl die Protokolle als auch die Warteschlangendateien zu speichern, oder Sie können sie über Laufwerke hinweg aufteilen. Wenn jeder WS-Manager über eine eigene gemeinsam genutzte Platte verfügt, stellen Sie sicher, dass alle Laufwerke, die von diesem Warteschlangenmanager verwendet werden, exklusiv für diesen Warteschlangenmanager sind, d. -E., dass nichts anderes auf den Laufwerken basiert. Stellen Sie außerdem sicher, dass Sie für jedes Laufwerk, das der Warteschlangenmanager verwendet, eine Ressourceninstanz erstellen.  
 Der Ressourcentyp für ein Laufwerk hängt von der von Ihnen verwendeten SCSI-Unterstützung ab. Weitere Informationen finden Sie in den Anweisungen zum SCSI-Adapter. Es können bereits Gruppen und Ressourcen für die einzelnen gemeinsam genutzten Laufwerke vorhanden sein. Ist dies der Fall, müssen Sie die Ressourceninstanz für jedes Laufwerk nicht erstellen. Verschieben Sie ihn von seiner aktuellen Gruppe in die für den Warteschlangenmanager erstellte Gruppe.  
 Geben Sie für jede Laufwerkressource die möglichen Eigner beider Knoten an. Setzen Sie abhängige Ressourcen auf "none".
  - e. Wählen Sie die Ressource **IBM MQSeries MSCS** in der Anzeige "Ressourcentyp auswählen" aus.
  - f. Führen Sie die verbleibenden Schritte im Assistenten aus.
5. Bevor die Ressource online geschaltet wird, benötigt die MSCS-Ressource IBM MQSeries zusätzliche Konfigurationsschritte:
  - a. Wählen Sie den neu definierten Service aus, der eine Ressource namens 'New IBM MQSeries MSCS' enthält.
  - b. Klicken Sie in der MQ-Ressource auf **Eigenschaften .**
  - c. Konfigurieren Sie die Ressource:
    - Name; Wählen Sie einen Namen aus, der die Identifizierung des Warteschlangenmanagers, für den er gilt, vereinfacht.
    - Run in a separate Resource Monitor; für eine bessere Isolation
    - Possible owners; beide Knoten festlegen

- **Dependencies**; Fügen Sie das Laufwerk und die IP-Adresse für diesen Warteschlangenmanager hinzu.

**Warnung:** Wenn Sie diese Abhängigkeiten nicht hinzufügen, bedeutet dies, dass WebSphere MQ versucht, den Warteschlangenmanagerstatus während des Failovers auf die falsche Clusterplatte zu schreiben. Da viele Prozesse möglicherweise gleichzeitig versuchen, auf diese Platte zu schreiben, könnten einige IBM WebSphere MQ-Prozesse von der Ausführung blockiert werden.

- **Parameters**; wie folgt:

- **QueueManagerName** (erforderlich); der Name des Warteschlangenmanagers, den diese Ressource steuern soll. Dieser WS-Manager muss auf dem lokalen Computer vorhanden sein.
- **PostOnlineCommand** (optional); Sie können ein Programm angeben, das ausgeführt werden soll, wenn die WS-Manager-Ressource ihren Status von offline in online ändert. Nähere Informationen finden Sie unter „PostOnlineCommand und PreOfflineCommand“ auf Seite 370.
- **PreOfflineCommand** (optional); Sie können ein Programm angeben, das ausgeführt werden soll, wenn die Warteschlangenmanagerressource ihren Status von "Online" in "Offline" ändert. Nähere Informationen finden Sie unter „PostOnlineCommand und PreOfflineCommand“ auf Seite 370.

**Anmerkung:** Das Abfrageintervall für *LooksAlive* wird auf den Standardwert 5000 ms gesetzt. Das Abfrageintervall für *isAlive* wird auf den Standardwert 60000 ms gesetzt. Diese Standardwerte können nur geändert werden, nachdem die Ressourcendefinition abgeschlossen wurde. Weitere Informationen finden Sie im Abschnitt „Zusammenfassung der looksAlive- und isAlive-Abfragen“ auf Seite 367.

- Legen Sie optional einen bevorzugten Knoten fest (aber beachten Sie die Kommentare in „Bevorzugte Knoten verwenden“ auf Seite 371).
  - Die *Funktionsübernahmerrichtlinie* wird standardmäßig auf sinnvolle Werte gesetzt, Sie können jedoch die Schwellenwerte und Zeiträume, die *Resource Failover* und *Group Failover* steuern, so optimieren, dass sie mit den auf den Warteschlangenmanager platzierten Lasten übereinstimmen.
- Testen Sie den Warteschlangenmanager, indem Sie ihn online in den MSCS-Clusteradministrator setzen und einer Testworkload unterziehen. Wenn Sie mit einem Testwarteschlangenmanager experimentieren, verwenden Sie den IBM WebSphere MQ-Explorer. Beispiel:
    - Klicken Sie mit der rechten Maustaste auf den Baumknoten Queues , wählen Sie New->Local Queue . . . aus und geben Sie der Warteschlange einen Namen.
    - Klicken Sie auf Finish. Die Warteschlange wird erstellt und in der Inhaltsansicht angezeigt.
    - Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie Put Test Message . . . aus. Die Anzeige 'Testnachricht einreihen' wird angezeigt.
    - Geben Sie einen Nachrichtentext ein, klicken Sie dann auf Put Test Message und schließen Sie die Anzeige.
    - Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie Browse Messages . . . aus. Die Anzeige 'Nachrichten-Browser' wird angezeigt.
    - Stellen Sie sicher, dass sich Ihre Nachricht in der Warteschlange befindet, und klicken Sie anschließend auf Close . Die Anzeige 'Nachrichten-Browser' wird geschlossen.
    - Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie Clear Messages . . . aus. Die Nachrichten in der Warteschlange werden gelöscht.
    - Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie Delete . . . aus. Eine Bestätigungsanzeige wird angezeigt. Klicken Sie auf OK. Die Warteschlange wird gelöscht.
  - Testen Sie, ob der Warteschlangenmanager offline und wieder online mit dem MSCS-Cluster-Administrator ausgeführt werden kann.
  - Simulieren Sie eine Funktionsübernahme.

Klicken Sie im MSCS-Clusteradministrator auf die Gruppe, die den Warteschlangenmanager enthält, und wählen Sie Move Group aus. Dies kann einige Minuten dauern. (Wenn Sie einen Warteschlangenmanager zu einem anderen Zeitpunkt schnell in einen anderen Knoten verschieben möchten,

führen Sie die Prozedur in „Warteschlangenmanager in MSCS-Speicher versetzen“ auf Seite 358 aus.) Sie können auch mit der rechten Maustaste klicken und `Initiate Failure` auswählen. Die Aktion (lokaler Neustart oder Failover) hängt vom aktuellen Status und den Konfigurationseinstellungen ab.

## Windows 2003

Gehen Sie wie folgt vor, um einen Warteschlangenmanager unter Windows 2003 unter MSCS-Steuerung zu stellen:

1. Melden Sie sich am Cluster-Knotencomputer an, auf dem sich der Warteschlangenmanager befindet, oder melden Sie sich als Benutzer mit Clusterverwaltungsberechtigungen an einer fernen Workstation an und stellen Sie eine Verbindung zu dem Clusterknoten her, der als Host für den Warteschlangenmanager fungiert.
2. Starten Sie den MSCS-Clusteradministrator.
3. Öffnen Sie eine Verbindung zum Cluster.
4. Erstellen Sie eine MSCS-Gruppe, die verwendet werden soll, um die Ressourcen für den Warteschlangenmanager zu enthalten. Benennen Sie die Gruppe so, dass es ersichtlich ist, auf welche Warteschlangenmanager sie sich bezieht. Jede Gruppe kann mehrere Warteschlangenmanager enthalten, wie in „Mehrere WS-Manager mit MSCS verwenden“ auf Seite 356 beschrieben.

Verwenden Sie die Gruppe für alle verbleibenden Schritte.

5. Erstellen Sie eine Ressourceninstanz für jedes der logischen SCSI-Laufwerke, die vom Warteschlangenmanager verwendet werden.

Sie können ein Laufwerk verwenden, um sowohl die Protokolle als auch die Warteschlangendateien zu speichern, oder Sie können sie über Laufwerke hinweg aufteilen. Wenn jeder WS-Manager über eine eigene gemeinsam genutzte Platte verfügt, stellen Sie sicher, dass alle Laufwerke, die von diesem Warteschlangenmanager verwendet werden, exklusiv für diesen Warteschlangenmanager sind, d. -E., dass nichts anderes auf den Laufwerken basiert. Stellen Sie außerdem sicher, dass Sie für jedes Laufwerk, das der Warteschlangenmanager verwendet, eine Ressourceninstanz erstellen.

Der Ressourcentyp für ein Laufwerk hängt von der von Ihnen verwendeten SCSI-Unterstützung ab. Weitere Informationen finden Sie in den Anweisungen zum SCSI-Adapter. Es können bereits Gruppen und Ressourcen für die einzelnen gemeinsam genutzten Laufwerke vorhanden sein. Ist dies der Fall, müssen Sie die Ressourceninstanz für jedes Laufwerk nicht erstellen. Verschieben Sie ihn von seiner aktuellen Gruppe in die für den Warteschlangenmanager erstellte Gruppe.

Geben Sie für jede Laufwerkressource die möglichen Eigner beider Knoten an. Setzen Sie abhängige Ressourcen auf "none".

6. Erstellen Sie eine Ressourceninstanz für die IP-Adresse.

Erstellen Sie eine IP-Adressenressource (Ressourcentyp *IP-Adresse*). Bei dieser Adresse muss es sich um eine nicht verwendete IP-Adresse handeln, die von Clients und anderen Warteschlangenmanagern verwendet werden soll, um eine Verbindung zum *virtuellen* Warteschlangenmanager herzustellen. Bei dieser IP-Adresse handelt es sich nicht um die normale (statische) Adresse eines der beiden Knoten. Es handelt sich um eine zusätzliche Adresse, die zwischen den beiden Knoten *floats* ist. Obwohl MSCS das Routing dieser Adresse handhabt, prüft es **nicht**, ob die Adresse erreicht werden kann.

7. Erstellen Sie eine Ressourceninstanz für den WS-Manager.

Erstellen Sie eine Ressource des Typs *IBM WebSphere MQ MSCS*. Der Assistent fordert Sie zur Eingabe verschiedener Elemente auf. Dazu gehören die folgenden:

- Name; Wählen Sie einen Namen aus, der die Identifizierung des Warteschlangenmanagers, für den er gilt, vereinfacht.
- Add to group; verwenden Sie die von Ihnen erstellte Gruppe
- Run in a separate Resource Monitor; für eine bessere Isolation
- Possible owners; beide Knoten festlegen

- **Dependencies**; Fügen Sie das Laufwerk und die IP-Adresse für diesen Warteschlangenmanager hinzu.

**Warnung:** Wenn Sie diese Abhängigkeiten nicht hinzufügen, bedeutet dies, dass WebSphere MQ versucht, den Warteschlangenmanagerstatus während des Failovers auf die falsche Clusterplatte zu schreiben. Da viele Prozesse möglicherweise gleichzeitig versuchen, auf diese Platte zu schreiben, könnten einige IBM WebSphere MQ-Prozesse von der Ausführung blockiert werden.

- **Parameters**; wie folgt:

- **QueueManagerName** (erforderlich); der Name des Warteschlangenmanagers, den diese Ressource steuern soll. Dieser WS-Manager muss auf dem lokalen Computer vorhanden sein.
- **PostOnlineCommand** (optional); Sie können ein Programm angeben, das ausgeführt werden soll, wenn die WS-Manager-Ressource ihren Status von offline in online ändert. Nähere Informationen finden Sie unter „PostOnlineCommand und PreOfflineCommand“ auf Seite 370.
- **PreOfflineCommand** (optional); Sie können ein Programm angeben, das ausgeführt werden soll, wenn die Warteschlangenmanagerressource ihren Status von "Online" in "Offline" ändert. Nähere Informationen finden Sie unter „PostOnlineCommand und PreOfflineCommand“ auf Seite 370.

**Anmerkung:** Das Abfrageintervall für *LooksAlive* wird auf den Standardwert 5000 ms gesetzt. Das Abfrageintervall für *isAlive* wird auf den Standardwert 30000 ms gesetzt. Diese Standardwerte können nur geändert werden, nachdem die Ressourcendefinition abgeschlossen wurde. Weitere Informationen finden Sie im Abschnitt „Zusammenfassung der looksAlive- und isAlive-Abfragen“ auf Seite 367.

- Legen Sie optional einen bevorzugten Knoten fest (aber beachten Sie die Kommentare in „Bevorzugte Knoten verwenden“ auf Seite 371).
- Die *Funktionsübernehmerichtlinie* (wie in den Eigenschaften für die Gruppe definiert) wird standardmäßig auf sinnvolle Werte gesetzt, Sie können jedoch die Schwellenwerte und Zeiträume, die *Resource Failover* und *Group Failover* steuern, so optimieren, dass sie mit den auf den Warteschlangenmanager platzierten Lasten übereinstimmen.
- Testen Sie den Warteschlangenmanager, indem Sie ihn online in den MSCS-Clusteradministrator setzen und einer Testworkload unterziehen. Wenn Sie mit einem Testwarteschlangenmanager experimentieren, verwenden Sie den IBM WebSphere MQ-Explorer. Beispiel:
  - Klicken Sie mit der rechten Maustaste auf den Baumknoten Queues, wählen Sie New->Local Queue... aus und geben Sie der Warteschlange einen Namen.
  - Klicken Sie auf Finish. Die Warteschlange wird erstellt und in der Inhaltsansicht angezeigt.
  - Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie Put Test Message... aus. Die Anzeige 'Testnachricht einreihen' wird angezeigt.
  - Geben Sie einen Nachrichtentext ein, klicken Sie dann auf Put Test Message und schließen Sie die Anzeige.
  - Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie Browse Messages... aus. Die Anzeige 'Nachrichten-Browser' wird angezeigt.
  - Stellen Sie sicher, dass sich Ihre Nachricht in der Warteschlange befindet, und klicken Sie anschließend auf Close. Die Anzeige 'Nachrichten-Browser' wird geschlossen.
  - Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie Clear Messages... aus. Die Nachrichten in der Warteschlange werden gelöscht.
  - Klicken Sie mit der rechten Maustaste auf die Warteschlange und wählen Sie Delete... aus. Eine Bestätigungsanzeige wird angezeigt. Klicken Sie auf OK. Die Warteschlange wird gelöscht.
- Testen Sie, ob der Warteschlangenmanager offline und wieder online mit dem MSCS-Cluster-Administrator ausgeführt werden kann.
- Simulieren Sie eine Funktionsübernahme.

Klicken Sie im MSCS-Clusteradministrator auf die Gruppe, die den Warteschlangenmanager enthält, und wählen Sie Move Group aus. Dies kann einige Minuten dauern. (Wenn Sie einen Warteschlan-

genmanager zu einem anderen Zeitpunkt schnell in einen anderen Knoten verschieben möchten, führen Sie die Prozedur in „Warteschlangenmanager in MSCS-Speicher versetzen“ auf Seite 358 aus.) Sie können auch mit der rechten Maustaste klicken und `Initiate Failure` auswählen. Die Aktion (lokaler Neustart oder Failover) hängt vom aktuellen Status und den Konfigurationseinstellungen ab.

### **Zusammenfassung der *looksAlive*- und *isAlive*-Abfragen**

*looksAlive* und *isAlive* sind Intervalle, in denen MSCS-Aufrufe zurück in die vom bereitgestellten Bibliothekscode gelieferten Ressourcentypen zurückrufen, und fordert, dass die Ressource prüft, ob der Arbeitsstatus von sich selbst bestimmt wird. Dies bestimmt schließlich, ob MSCS über die Ressource fehlschlagen soll.

Bei jedem Ablaufen des *looksAlive* -Intervalls (Standardwert: 5000 ms) wird die WS-Manager-Ressource aufgerufen, um eine eigene Prüfung durchzuführen, um festzustellen, ob der Status zufriedenstellend ist.

Bei jeder Ausführung des *isAlive* -Intervalls (Standardwert: 30000 ms) wird ein weiterer Aufruf an die WS-Manager-Ressource vorgenommen, um eine weitere Überprüfung durchzuführen, um zu ermitteln, ob die Ressource ordnungsgemäß funktioniert. Auf diese Weise können zwei Ebenen der Ressourcentyp-überprüfung aktiviert werden

1. Eine *looksAlive* -Statusprüfung, um festzustellen, ob die Ressource funktionsfähig ist.
2. Eine größere *isAlive* Prüfung, die festlegt, ob die WS-Manager-Ressource aktiv ist.

Wenn die Warteschlangenmanagerressource nicht aktiv sein soll, wird MSCS auf der Basis anderer erweiterter MSCS-Optionen für die Ressource und die zugehörigen abhängigen Ressourcen zu einem anderen Knoten im Cluster ausgelöst. Weitere Informationen finden Sie in der [MSCS-Dokumentation](#) .

### **Entfernen eines Warteschlangenmanagers aus MSCS-Steuerung**

Sie können Warteschlangenmanager aus der MSCS-Steuerung entfernen und sie zur manuellen Verwaltung zurückgeben.

Sie müssen Warteschlangenmanager nicht von der MSCS-Steuerung für Wartungsoperationen entfernen. Sie können das tun, indem Sie einen Warteschlangenmanager vorübergehend offline schalten, indem Sie den MSCS-Clusteradministrator verwenden. Das Entfernen eines WS-Managers von der MSCS-Steuerung ist eine permanente Änderung; nur dann, wenn Sie entscheiden, dass MSCS keine weitere Steuerung des Warteschlangenmanagers mehr haben soll.

Wenn der zu entfernende Warteschlangenmanager SSL-Verbindungen verwendet, müssen Sie das Warteschlangenmanagerattribut `SSLKEYR` mit WebSphere MQ Explorer oder mit dem MQSC-Befehl `ALTER QMGRso` ändern, dass es auf die SSL-Schlüsselrepositorodatei im lokalen Verzeichnis verweist.

Das Verfahren ist wie folgt:

1. Verwenden Sie den MSCS-Cluster-Administrator, wie in „[WS-Manager offline von MSCS übernehmen](#)“ auf Seite 367 beschrieben, um die Warteschlangenmanagerressource offline zu schalten.
2. Löschen Sie die Ressourceninstanz. Der WS-Manager wird dadurch nicht gelöscht.
3. Optional können Sie die WS-Manager-Dateien von gemeinsam genutzten Laufwerken zurück auf lokale Laufwerke migrieren. Informationen dazu finden Sie unter „[WS-Manager aus MSCS-Speicher zurückgeben](#)“ auf Seite 368.
4. Testen Sie den WS-Manager.

### **WS-Manager offline von MSCS übernehmen**

Führen Sie die folgenden Schritte aus, um einen Warteschlangenmanager offline von MSCS zu nehmen:

1. Starten Sie den MSCS-Clusteradministrator.
2. Öffnen Sie eine Verbindung zum Cluster.
3. Wählen Sie `Groups` aus und öffnen Sie die Gruppe mit dem zu verschiebenden Warteschlangenmanager.
4. Wählen Sie die WS-Manager-Ressource aus.



5. Klicken Sie auf die Schaltfläche und wählen Sie *Offline* aus.
6. Warten Sie auf den Abschluss.

## WS-Manager aus MSCS-Speicher zurückgeben

Bei dieser Prozedur wird der Warteschlangenmanager so konfiguriert, dass er sich wieder auf dem lokalen Laufwerk seines Computers befindet, d. h., er wird zu einem *normalen* WebSphere MQ -Warteschlangenmanager. Um dies zu erreichen, verschieben Sie die Protokolldateien und Datendateien von den gemeinsam genutzten Platten. Der vorhandene Warteschlangenmanager kann z. B. Pfade wie E:\WebSphere MQ\Log\<QMname> und E:\WebSphere MQ\qmgrs\<QMname> haben. Versuchen Sie nicht, die Dateien manuell zu verschieben. Verwenden Sie das Dienstprogramm **hamvmqm**, das als Teil der MSCS-Unterstützung für WebSphere MQ bereitgestellt wird:

1. Fahren Sie den Warteschlangenmanager herunter, und prüfen Sie, ob keine Fehler aufgetreten sind.
2. Führen Sie ein vollständiges Backup aller Warteschlangen- und Protokolldateien durch und speichern Sie diese Sicherheitskopie in einem sicheren Bereich (in Abschnitt [„Warteschlangenmanagerprotokoll-dateien“](#) auf Seite 370 wird erläutert, warum dies wichtig ist).
3. Entscheiden Sie, welches lokale Laufwerk verwendet werden soll, und stellen Sie sicher, dass es über ausreichende Kapazität zum Speichern der Protokolldateien und Datendateien (Warteschlangendateien) des Warteschlangenmanagers verfügt.
4. Stellen Sie sicher, dass sich die gemeinsam genutzte Platte, auf der sich die Dateien befinden, online auf dem Clusterknoten befindet, in den das WS-Manager-Protokoll und die Datendateien verschoben werden sollen.
5. Führen Sie das Dienstprogramm aus, um den WS-Manager wie folgt zu verschieben:

```
hamvmqm /m qmname /dd "c:\
WebSphere MQ" /ld "c:\
WebSphere MQ\log"
```

Ersetzen Sie *qmname* durch den Namen Ihres Warteschlangenmanagers, *c* durch den Buchstaben Ihres lokalen Plattenlaufwerks und *WebSphere MQ* durch das ausgewählte Verzeichnis (die Verzeichnisse werden erstellt, wenn sie noch nicht vorhanden sind).

6. Testen Sie den Warteschlangenmanager, um sicherzustellen, dass er funktioniert (wie in [„Warteschlangenmanager in MSCS-Speicher versetzen“](#) auf Seite 358 beschrieben).

## Hinweise und Tipps zur Verwendung von MSCS

Dieser Abschnitt enthält einige allgemeine Informationen, die Sie bei der effektiven Verwendung der WebSphere MQ -Unterstützung für MSCS unterstützen.

Dieser Abschnitt enthält einige allgemeine Informationen, die Sie bei der effektiven Verwendung der WebSphere MQ -Unterstützung für MSCS unterstützen.

Wie lange dauert es, bis ein WS-Manager von einer Maschine zum anderen fehlschlägt? Dies hängt stark von der Auslastung des Warteschlangenmanagers und der Kombination des Datenverkehrs ab, z. B., wie viel von dem Warteschlangenmanager persistent ist, innerhalb des Synchronisationspunkts und wie viel vor dem Fehlschlag festgeschrieben wurde. IBM -Tests haben Failover- und Failback-Zeiten von etwa einer Minute angegeben. Dies war auf einem sehr gering belasteten WS-Manager, und die tatsächlichen Zeiten werden je nach Ladezeit sehr unterschiedlich sein.

### Sicherstellen, dass MSCS funktioniert

Führen Sie die folgenden Schritte aus, um sicherzustellen, dass Sie über einen aktiven MSCS-Cluster verfügen.

Bei den Taskbeschreibungen, die mit [„Warteschlangenmanager für die Verwendung mit MSCS erstellen“](#) auf Seite 357 beginnen, wird davon ausgegangen, dass Sie über einen aktiven MSCS-Cluster verfügen, in dem Sie Ressourcen erstellen, migrieren und löschen können. Wenn Sie sicherstellen möchten, dass Sie über einen solchen Cluster verfügen:

1. Erstellen Sie mit dem MSCS-Clusteradministrator eine Gruppe.



2. Erstellen Sie in dieser Gruppe eine Instanz einer generischen Anwendungsressource und geben Sie dabei die Systemuhr an (Pfadname C:\winnt\system32\clock.exe und Arbeitsverzeichnis von C:\).
3. Stellen Sie sicher, dass Sie die Ressource in den Onlinemodus versetzen können, dass Sie die Gruppe, die sie enthält, in den anderen Knoten verschieben können und dass Sie die Ressource offline schalten können.

#### *Manueller Start*

Für einen Warteschlangenmanager, der von MSCS verwaltet wird, **müssen** Sie das Startattribut auf "manuell" setzen. Dadurch wird sichergestellt, dass die MSCS-Unterstützung für WebSphere MQ den IBM MQSeries -Service ohne sofortigen Start des Warteschlangenmanagers erneut starten kann.

Die MSCS-Unterstützung für WebSphere MQ muss in der Lage sein, den Service erneut zu starten, damit er die Überwachung und Steuerung ausführen kann, muss aber selbst steuern können, welche Warteschlangenmanager ausgeführt werden und auf welchen Maschinen. Weitere Informationen finden Sie im Abschnitt [„Warteschlangenmanager in MSCS-Speicher versetzen“](#) auf Seite 358.

#### *MSCS-und Warteschlangenmanager*

Hinweise zu Warteschlangenmanagern bei der Verwendung von MSCS.

## **Erstellen eines übereinstimmenden Warteschlangenmanagers auf dem anderen Knoten**

Damit das Clustering mit WebSphere MQ funktioniert, benötigen Sie einen identischen Warteschlangenmanager auf Knoten B für jeden auf Knoten A. Es ist jedoch nicht erforderlich, die zweite explizit zu erstellen. Sie können einen Warteschlangenmanager auf einem Knoten erstellen oder vorbereiten, ihn wie in [„Warteschlangenmanager in MSCS-Speicher versetzen“](#) auf Seite 358 beschrieben in den anderen Knoten versetzen und auf diesem Knoten vollständig duplizieren.

## **Standardwarteschlangenmanager**

Verwenden Sie keinen Standard-WS-Manager unter MSCS-Steuerung. Ein Warteschlangenmanager verfügt nicht über eine Eigenschaft, die ihn zur Standardeinstellung macht; WebSphere MQ verwaltet einen eigenen separaten Datensatz. Wenn Sie einen Warteschlangenmanager so einstellen, dass er als Standardwert für den anderen Computer bei der Funktionsübernahme verwendet wird, wird er nicht zum Standardwert dort. Stellen Sie sicher, dass alle Anwendungen auf bestimmte WS-Manager nach Namen verweisen.

## **Löschen eines Warteschlangenmanagers**

Sobald ein WS-Manager einen Knoten verschoben hat, sind seine Details in der Registry auf beiden Computern vorhanden. Wenn Sie den Warteschlangenmanager löschen möchten, tun Sie dies wie üblich auf einem Computer und führen Sie dann das in [„IBM WebSphere MQ-Unterstützungsprogramme für MSCS“](#) auf Seite 371 beschriebene Dienstprogramm aus, um die Registry auf dem anderen Computer zu bereinigen.

## **Unterstützung für vorhandene WS-Manager**

Sie können einen vorhandenen Warteschlangenmanager unter MSCS-Steuerung stellen, vorausgesetzt, Sie können die Protokolldateien und die Warteschlangendateien Ihres Warteschlangenmanagers auf einer Platte, die sich auf dem gemeinsam genutzten SCSI-Bus befindet, zwischen den beiden Maschinen (siehe [Abbildung 62](#) auf Seite 355 ) setzen. Sie müssen den WS-Manager kurz offline schalten, während die MSCS-Ressource erstellt wird.

Wenn Sie einen neuen Warteschlangenmanager erstellen möchten, erstellen Sie ihn unabhängig von MSCS, testen Sie ihn, und legen Sie ihn dann unter MSCS-Steuerung. Unter

- [„Warteschlangenmanager für die Verwendung mit MSCS erstellen“](#) auf Seite 357
- [„Warteschlangenmanager in MSCS-Speicher versetzen“](#) auf Seite 358

- [„WS-Manager unter MSCS-Steuerung einschalten“](#) auf Seite 360

## MSCS-Warteschlangenmanager für die Verwaltung von Telling

Sie können auswählen, welche Warteschlangenmanager unter MSCS-Steuerung platziert werden, indem Sie den MSCS-Clusteradministrator verwenden, um eine Ressourceninstanz für jeden dieser Warteschlangenmanager zu erstellen. In diesem Prozess wird eine Liste der Ressourcen angezeigt, aus denen Sie den Warteschlangenmanager auswählen können, den diese Instanz verwalten soll.

## Warteschlangenmanagerprotokolldateien

Wenn Sie einen Warteschlangenmanager in den MSCS-Speicher versetzen, versetzen Sie seine Protokoll- und Datendateien auf eine gemeinsam genutzte Platte (siehe [„Warteschlangenmanager in MSCS-Speicher versetzen“](#) auf Seite 358).

Es empfiehlt sich, vor dem Verschieben den Warteschlangenmanager ordnungsgemäß zu beenden und eine vollständige Sicherung der Datendateien und Protokolldateien zu erstellen.

## Mehrere Warteschlangenmanager

Die MSCS-Unterstützung von WebSphere MQ ermöglicht Ihnen, mehrere Warteschlangenmanager auf jeder Maschine auszuführen und einzelne Warteschlangenmanager unter MSCS-Steuerung zu stellen.

### *Immer MSCS zum Verwalten von Clustern verwenden*

Versuchen Sie nicht, mithilfe von Steuerbefehlen oder IBM WebSphere MQ Explorer Operationen zum Starten und Stoppen direkt auf einem Warteschlangenmanager unter der Kontrolle von MSCS auszuführen. Verwenden Sie stattdessen MSCS Cluster Administrator, um den WS-Manager online zu schalten oder offline zu schalten.

Die Verwendung des MSCS-Clusteradministrators ist zum Teil die Vermeidung möglicher Verwechslungen durch MSCS, die gemeldet werden, dass der Warteschlangenmanager offline ist, wenn Sie ihn tatsächlich außerhalb der Steuerung von MSCS gestartet haben. Ein schwerwiegender Fehler beim Stoppen eines Warteschlangenmanagers ohne MSCS wird von MSCS als Fehler erkannt und die Übernahme durch den anderen Knoten eingeleitet.

### *Im Aktiv/Aktiv-Modus arbeiten*

Beide Computer im MSCS-Cluster können WS-Manager im Aktiv/Aktiv-Modus ausführen. Es ist nicht erforderlich, eine vollständig inaktive Maschine als Standby-Server zu verwenden (aber Sie können, wenn Sie möchten, im Modus 'Aktiv/Passiv') ausgeführt werden.

Wenn Sie beide Maschinen zur Ausführung der Workload verwenden möchten, stellen Sie jeder Maschine ausreichend Kapazität (Prozessor, Speicher, Sekundärspeicher) zur Verfügung, um die gesamte Clusterauslastung mit einem zufriedenstellenden Leistungsniveau zu führen.

**Anmerkung:** Wenn Sie MSCS zusammen mit Microsoft Transaction Server (COM+) verwenden, können Sie den Aktiv/Aktiv-Modus **nicht** verwenden. Dies liegt daran, dass Sie WebSphere MQ mit MSCS und COM+ verwenden müssen:

- Anwendungskomponenten, die die COM+ -Unterstützung von WebSphere MQ verwenden, müssen auf demselben Computer ausgeführt werden wie DTC (Distributed Transaction Coordinator), ein Teil von COM+.
- Der WS-Manager muss auch auf demselben Computer ausgeführt werden.
- Die DTC muss als MSCS-Ressource konfiguriert werden und kann daher zu einem beliebigen Zeitpunkt auf einem der Computer im Cluster ausgeführt werden.

### *PostOnlineCommand und PreOfflineCommand*

Mit diesen Befehlen können Sie die MSCS-Unterstützung von WebSphere MQ in andere Systeme integrieren. Sie können sie verwenden, um WebSphere MQ -Befehle abzusetzen, d. h. einige Einschränkungen.

Geben Sie diese Befehle in den Parametern für eine Ressource des Typs IBM WebSphere MQ MSCS an. Sie können sie verwenden, um WebSphere MQ MSCS-Unterstützung mit anderen Systemen oder Prozeduren zu integrieren. Sie können z. B. den Namen eines Programms angeben, das eine E-Mail-Nachricht sendet, einen Pager aktiviert oder eine andere Form von Alert generiert, die von einem anderen Überwachungssystem erfasst werden soll.

Der Befehl 'PostOnlineCommand' wird aufgerufen, wenn sich die Ressource von offline in online ändert. Der Befehl 'PreOfflineCommand' wird für einen Wechsel von online in offline aufgerufen. Beim Aufruf werden diese Befehle standardmäßig aus dem Windows-Systemverzeichnis ausgeführt. Da WebSphere MQ einen 32-Bit-Ressourcenüberwachungsprozess verwendet, handelt es sich auf 64-Bit-Systemen unter Windows nicht um \Windows\system32, sondern um das Verzeichnis \Windows\SysWOW64. Weitere Informationen finden Sie in der Microsoft-Dokumentation zur Dateiumleitung in einer Windows x64 -Umgebung. Beide Befehle werden unter dem Benutzerkonto ausgeführt, das für die Ausführung des MSCS-Cluster-Service verwendet wird, und werden asynchron aufgerufen. Die WebSphere MQ MSCS-Unterstützung wartet nicht darauf, dass sie abgeschlossen werden, bevor sie fortgesetzt werden. Dadurch wird das Risiko ausgeschlossen, dass weitere Clusteroperationen blockiert oder verzögert werden.

Sie können diese Befehle auch verwenden, um WebSphere MQ -Befehle abzusetzen, z. B. um Requestkanäle erneut zu starten. Die Befehle werden jedoch zu dem Zeitpunkt ausgeführt, zu dem sich die Statusänderungen des Warteschlangenmanagers ändern, so dass sie nicht für die Ausführung von Funktionen mit langer Laufzeit gedacht sind und keine Annahmen über den aktuellen Status des Warteschlangenmanagers treffen müssen. Es ist durchaus möglich, dass ein Administrator unmittelbar nach dem Online-Betrieb des Warteschlangenmanagers einen Offline-Befehl abgesetzt hat.

Wenn Sie Programme ausführen wollen, die vom Status des Warteschlangenmanagers abhängig sind, sollten Sie die Erstellung von Instanzen des Ressourcentyps MSCS Generic Application in der gleichen MSCS-Gruppe wie die Warteschlangenmanager-Ressource in Betracht ziehen und sie abhängig von der WS-Manager-Ressource machen.

#### *Bevorzugte Knoten verwenden*

Bei Verwendung des Aktiv/Aktiv-Modus kann es hilfreich sein, einen *bevorzugten Knoten* für jeden Warteschlangenmanager zu konfigurieren. Im Allgemeinen ist es jedoch besser, keinen bevorzugten Knoten festzulegen, sondern sich auf einen manuellen Failback zu verlassen.

Im Gegensatz zu anderen relativ statusunabhängigen Ressourcen kann ein Warteschlangenmanager eine Weile dauern, bis er (oder zurück) von einem Knoten zum anderen fehlschlägt. Um unnötige Ausfälle zu vermeiden, testen Sie den wiederhergestellten Knoten, bevor Sie einen WS-Manager wieder in den Knoten zurückschlagen. Dies schließt die Verwendung der *immediate* -Failback-Einstellung aus. Sie können die Failback-Funktion zwischen bestimmten Tageszeiten konfigurieren.

Die sicherste Route ist wahrscheinlich, dass der WS-Manager manuell auf den erforderlichen Knoten zurückversetzt wird, wenn Sie sicher sind, dass der Knoten vollständig wiederhergestellt ist. Dies schließt die Verwendung der Option *preferred node* aus.

#### *Vorgehensweise bei Auftreten von COM+-Fehlern im Anwendungsereignisprotokoll*

Wenn Sie WebSphere MQ in einem neu installierten MSCS-Cluster installieren, finden Sie möglicherweise einen Fehler mit der Quelle COM + und der Ereignis-ID 4691 im Anwendungsereignisprotokoll.

Dies bedeutet, dass Sie versuchen, WebSphere MQ in einer MSCS-Umgebung ( Microsoft Cluster Server) auszuführen, wenn Microsoft Distributed Transaction Coordinator (MSDTC) nicht für die Ausführung in einer solchen Umgebung konfiguriert wurde. Informationen zur Konfiguration von MSDTC in einer Clustenumgebung finden Sie in der Microsoft -Dokumentation.

### **IBM WebSphere MQ-Unterstützungsprogramme für MSCS**

Eine Liste der IBM WebSphere MQ-Unterstützung für MSCS-Dienstprogramme, die Sie an einer Eingabeaufforderung ausführen können.

IBM WebSphere MQ-Unterstützung für MSCS umfasst die folgenden Dienstprogrammprogramme:

#### **Ressourcentyp registrieren/deregistrieren**

haregtyp.exe

Nachdem Sie die Registrierung des IBM WebSphere MQ MSCS-Ressourcentyps *aufgehoben* haben, können Sie keine Ressourcen dieses Typs mehr erstellen. MSCS lässt die Deregistrierung eines Ressourcentyps nicht zu, wenn noch Instanzen dieses Typs in dem Cluster vorhanden sind:

1. Stoppen Sie mit Hilfe des MSCS-Clusteradministrators alle Warteschlangenmanager, die unter MSCS-Steuerung ausgeführt werden, indem Sie sie offline schalten, wie in [„WS-Manager offline von MSCS übernehmen“](#) auf Seite 367 beschrieben.
2. Löschen Sie die Ressourceninstanzen mit Hilfe des MSCS-Clusteradministrators.
3. Heben Sie an einer Eingabeaufforderung die Registrierung des Ressourcentyps auf, indem Sie den folgenden Befehl eingeben:

```
haregtyp /u
```

Wenn Sie den Typ *registrieren* (oder zu einem späteren Zeitpunkt erneut registrieren) möchten, geben Sie an einer Eingabeaufforderung den folgenden Befehl ein:

```
haregtyp /r
```

Nachdem Sie die MSCS-Bibliotheken erfolgreich registriert haben, müssen Sie das System neu starten, wenn Sie dies seit der Installation von IBM WebSphere MQ noch nicht getan haben.

### **WS-Manager in MSCS-Speicher verschieben**

hamvmqm.exe

Siehe [„Warteschlangenmanager in MSCS-Speicher versetzen“](#) auf Seite 358.

### **WS-Manager aus einem Knoten löschen**

hadl1mqm.exe

Angenommen, Sie haben einen Warteschlangenmanager in Ihrem Cluster, der von einem Knoten in einen anderen verschoben wurde, und jetzt möchten Sie ihn zerstören. Verwenden Sie die IBM WebSphere MQ Explorer, um sie auf dem Knoten zu löschen, auf dem sie sich derzeit befindet. Die Registry-Einträge für diese Datei sind noch auf dem anderen Computer vorhanden. Um diese zu löschen, geben Sie den folgenden Befehl an einer Eingabeaufforderung auf diesem Computer ein:

```
hadl1mqm /m qmname
```

Dabei steht qmname für den Namen des zu entfernenden Warteschlangenmanager.

### **Setup-Details prüfen und speichern**

amqmsysn.exe

Dieses Dienstprogramm stellt einen Dialog mit vollständigen Details Ihrer IBM WebSphere MQ MSCS-Support-Konfiguration dar, z. B. wenn Sie sich an den IBM Support wenden. Es gibt eine Option zum Speichern der Details in einer Datei.

## **Warteschlangenmanager mit mehreren Instanzen**

Multi-Instanz-Warteschlangenmanager sind Instanzen desselben Warteschlangenmanagers, die auf verschiedenen Servern konfiguriert sind. Eine Instanz des Warteschlangenmanagers ist als aktive Instanz definiert, die andere ist eine Standby-Instanz. Wenn die aktive Instanz ausfällt, wird der Multi-Instanz-Warteschlangenmanager automatisch auf dem Standby-Server gestartet.

Abbildung 63 auf Seite 373 zeigt eine Multi-Instanz-Konfiguration für QM1. IBM WebSphere MQ ist auf zwei Servern installiert, von denen einer als Ersatz dient. Es wurde ein Warteschlangenmanager (QM1) erstellt. Eine Instanz von QM1 ist aktiv und wird auf einem einzigen Server ausgeführt. Die andere Instanz von QM1 wird in der Bereitschaftsdatenbank auf dem anderen Server ausgeführt, ohne aktive Verarbeitung zu tun, aber bereit ist, von der aktiven Instanz von QM1 zu übernehmen, wenn die aktive Instanz fehlschlägt.

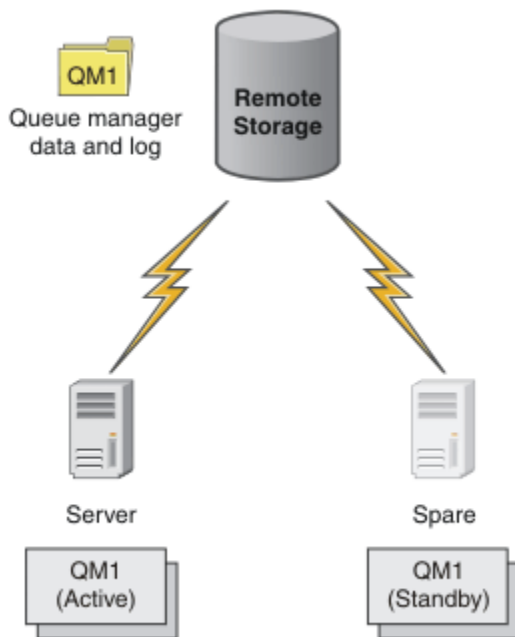


Abbildung 63. Multi-Instanz-Warteschlangenmanager

Wenn Sie einen Warteschlangenmanager als Multi-Instanz-Warteschlangenmanager verwenden wollen, erstellen Sie einen einzelnen Warteschlangenmanager auf einem der Server mit dem Befehl **crtmqm**, indem Sie seine Warteschlangenmanagerdaten und -protokolle in den gemeinsam genutzten Netzspeicher stellen. Verwenden Sie auf dem anderen Server den Befehl **addmqinf**, um einen Verweis auf die Warteschlangenmanagerdaten und -protokolle im Netzspeicher zu erstellen, anstatt den Warteschlangenmanager erneut zu erstellen.

Sie können nun den Warteschlangenmanager von einem der beiden Server aus ausführen. Jeder der Server verweist auf dieselben WS-Manager-Daten und -Protokolle. Es gibt nur einen Warteschlangenmanager, und er ist auf nur einem Server gleichzeitig aktiv.

Der Warteschlangenmanager kann entweder als Einzelinstanzwarteschlangenmanager oder als Warteschlangenmanager mit mehreren Instanzen ausgeführt werden. In beiden Fällen wird nur eine Instanz des Warteschlangenmanagers ausgeführt, Verarbeitungsanforderungen werden verarbeitet. Der Unterschied besteht darin, dass der Server, der die aktive Instanz des Warteschlangenmanagers nicht ausführt, als Standby-Instanz ausgeführt wird, wenn er als Standby-Warteschlangenmanager ausgeführt wird und bereit ist, automatisch von der aktiven Instanz zu übernehmen, wenn der aktive Server ausfällt.

Das einzige Steuerelement, über das Sie über die Instanz verfügen, ist die Reihenfolge, in der Sie den WS-Manager auf den beiden Servern starten. Die erste Instanz, die Lese-/Schreibsperrern für die WS-Manager-Daten erhält, wird zum aktiven Exemplar.

Sie können die aktive Instanz, nachdem sie gestartet wurde, in den anderen Server eintauschen, indem Sie die aktive Instanz mit der Umschaltoption stoppen, um die Steuerung an den Standby-Server zu übertragen.

Die aktive Instanz von QM1 hat exklusiven Zugriff auf die Daten des gemeinsam genutzten Warteschlangenmanagers und protokolliert Ordner, wenn sie ausgeführt wird. Die Standby-Instanz von QM1 erkennt, wenn die aktive Instanz ausgefallen ist, und wird zur aktiven Instanz. Sie übernimmt die QM1-Daten und -Protokolle in dem Status, den sie von der aktiven Instanz hinterlassen haben, und akzeptiert die Verbindungen von Clients und Kanälen.

Die aktive Instanz kann aus verschiedenen Gründen fehlschlagen, die dazu führen, dass die Bereitschaftsdatenbank die folgenden Schritte übernimmt:

- Fehler des Servers, auf dem sich die aktive WS-Manager-Instanz befindet.
- Fehler bei der Verbindung zwischen dem Server, auf dem sich die aktive WS-Manager-Instanz und das Dateisystem befinden.

- Keine Reaktionsfähigkeit der Warteschlangenmanagerprozesse, erkannt von WebSphere MQ, die dann den Warteschlangenmanager herunterfahren.

Sie können die WS-Manager-Konfigurationsinformationen zu mehreren Servern hinzufügen und die beiden Server als Aktiv/Standby-Paar ausführen. Es gibt eine Begrenzung von insgesamt zwei Instanzen. Es können keine zwei Standby-Instanzen und eine aktive Instanz vorhanden sein.

Ein WS-Manager mit mehreren Instanzen ist Teil einer Hochverfügbarkeitslösung. Sie benötigen einige zusätzliche Komponenten, um eine nützliche Hochverfügbarkeitslösung zu erstellen.

- Client- und Kanalverbindungswiederholung zur Übertragung von WebSphere MQ -Verbindungen zu dem Computer, der die aktive Warteschlangenmanagerinstanz übernimmt.
- Ein gemeinsam genutztes Network File System (NFS) mit hoher Leistung, das die Sperren korrekt verwaltet und den Schutz vor einem Datenträger- und Dateiserverfehler bietet.

**Wichtig:** Sie müssen alle WS-Manager-Instanzen mit mehreren Instanzen stoppen, die in Ihrer Umgebung ausgeführt werden, bevor Sie die Wartung auf dem NFS-Laufwerk ausführen können. Stellen Sie sicher, dass die Konfigurationssicherungen des Warteschlangenmanagers im Falle eines NFS-Fehlers wiederhergestellt werden müssen.

- Resiliente Netze und Netzteile, um Single Points of Failure in der Basisinfrastruktur zu eliminieren.
- Anwendungen, die Failover tolerieren. Insbesondere müssen Sie das Verhalten von Transaktionsanwendungen und Anwendungen, die WebSphere MQ -Warteschlangen durchsuchen, genau beachten.
- Überwachung und Verwaltung der aktiven und Standby-Instanzen, um sicherzustellen, dass sie aktiv sind, und um aktive Instanzen erneut zu starten, die fehlgeschlagen sind. Obwohl Warteschlangenmanager mit mehreren Instanzen automatisch erneut gestartet werden, müssen Sie sicher sein, dass Ihre Standby-Instanzen aktiv sind, bereit sind, um zu übernehmen, und dass fehlgeschlagene Instanzen wieder als neue Standby-Instanzen wieder online sind.

WebSphere MQ MQI-Clients und -Kanäle stellen die Verbindung zum Standby-Warteschlangenmanager automatisch wieder her, wenn er aktiv ist. Weitere Informationen zur erneuten Verbindung und zu den anderen Komponenten in einer Hochverfügbarkeitslösung finden Sie in den zugehörigen Themen. Die automatische Wiederherstellung einer Clientverbindung wird von IBM WebSphere MQ Classes for Java nicht unterstützt.

## Unterstützte Plattformen

Sie können einen Warteschlangenmanager mit mehreren Instanzen auf allen anderen Plattformen als z/OS ab Version 7.0.1 erstellen.

Die automatische Clientverbindungswiederholung wird für MQI-Clients ab Version 7.0.1 unterstützt.

### **Erstellen eines Warteschlangenmanagers mit mehreren Instanzen**

Erstellen Sie einen Multi-Instanz-Warteschlangenmanager, indem Sie den Warteschlangenmanager auf einem Server erstellen und IBM WebSphere MQ auf einem anderen Server konfigurieren. Gemeinsam genutzte WS-Manager-Daten und -Protokolle für mehrere Instanzen.

Der größte Aufwand bei der Erstellung eines Multi-Instanz-WS-Managers ist die Aufgabe, die Daten und Protokolldateien des gemeinsam genutzten Warteschlangenmanagers zu konfigurieren. Sie müssen gemeinsam genutzte Verzeichnisse im Netzspeicher erstellen und die Verzeichnisse für andere Server mit Netzfreigaben verfügbar machen. Diese Aufgaben müssen von einer Person mit Administratorberechtigung, z. B. *Root* auf UNIX and Linux-Systemen, durchgeführt werden. Die Schritte sind wie folgt:

1. Erstellen Sie die Freigaben für die Daten- und Protokolldateien.
2. Erstellen Sie den WS-Manager auf einem einzigen Server.
3. Führen Sie den Befehl **dspmqlinf** auf dem ersten Server aus, um die Konfigurationsdaten des Warteschlangenmanagers zu erfassen und in die Zwischenablage zu kopieren.
4. Führen Sie den Befehl **addmqinf** mit den kopierten Daten aus, um die Warteschlangenmanagerkonfiguration auf dem zweiten Server zu erstellen.

Sie führen **crtmqm** nicht aus, um den WS-Manager erneut auf dem zweiten Server zu erstellen.

## Dateizugriffssteuerung

Sie müssen darauf achten, dass der Benutzer und die Gruppe `mqm` auf allen anderen Servern die Berechtigung zum Zugriff auf die Freigaben haben.

Unter UNIX and Linux müssen Sie die `uid` und die `gid` von `mqm` auf allen Systemen gleich machen. Möglicherweise müssen Sie `/etc/passwd` auf jedem System bearbeiten, um eine gemeinsame `uid` und `gid` für `mqm` festzulegen, und anschließend einen Warmstart Ihres Systems.

Unter Microsoft Windows muss die Benutzer-ID, die die Warteschlangenmanagerprozesse ausführt, über vollständige Steuerungsberechtigung für die Verzeichnisse verfügen, die die WS-Manager-Daten und -Protokolldateien enthalten. Sie können die Berechtigung auf zwei Arten konfigurieren:

1. Erstellen Sie einen Warteschlangenmanager mit einer globalen Gruppe als alternativen Sicherheitsprincipal. Autorisieren Sie die globale Gruppe, um vollständigen Steuerungszugriff auf die Verzeichnisse zu haben, die Warteschlangenmanagerdaten und Protokolldateien enthalten. Weitere Informationen finden Sie in [„Gemeinsam genutzte Warteschlangenmanagerdaten, Protokollverzeichnisse und Dateien unter Windows sichern“](#) auf Seite 403. Erstellen Sie die Benutzer-ID, unter der der Warteschlangenmanager ausgeführt wird, ein Mitglied der globalen Gruppe. Sie können keinen lokalen Benutzer zu einem Mitglied einer globalen Gruppe machen, daher müssen die WS-Manager-Prozesse unter einer Domänenbenutzer-ID ausgeführt werden. Die Domänenbenutzer-ID muss ein Mitglied der lokalen Gruppe `mqm` sein. Die Task [„Multi-Instanz-Warteschlangenmanager auf Workstations oder Servern in einer Domäne erstellen“](#) auf Seite 377 veranschaulicht, wie ein WS-Manager mit mehreren Instanzen unter Verwendung von auf diese Weise gesicherten Dateien eingerichtet wird.
2. Erstellen Sie einen Warteschlangenmanager auf dem Domänencontroller, so dass die lokale `mqm`-Gruppe einen Domänenbereich ("domain local") hat. Sichern Sie die Dateifreigabe mit der lokalen `mqm`-Domäne und führen Sie WS-Manager-Prozesse für alle Instanzen eines Warteschlangenmanagers in derselben Domäne aus, die sich in derselben Domäne wie die lokale `mqm`-Gruppe enthält. Die Task [„Multi-Instanz-Warteschlangenmanager auf Domänencontrollern erstellen“](#) auf Seite 393 veranschaulicht, wie ein WS-Manager mit mehreren Instanzen unter Verwendung von auf diese Weise gesicherten Dateien eingerichtet wird.

## Konfigurationsdaten

Konfigurieren Sie so viele Warteschlangenmanagerinstanzen, wie Sie benötigen, indem Sie die Konfigurationsdaten des IBM WebSphere MQ-Warteschlangenmanagers für jeden Server ändern. Auf jedem Server muss die gleiche Version von IBM WebSphere MQ installiert sein, die auf einer kompatiblen Fixversion installiert ist. Die Befehle `dspmqinf` und `addmqinf` unterstützen Sie bei der Konfiguration der zusätzlichen Warteschlangenmanagerinstanzen. Alternativ können Sie die Dateien `mq5.ini` und `qm.ini` direkt bearbeiten. Die Themen [„Mehrinstanz-Warteschlangenmanager unter Linux erstellen“](#) auf Seite 415, [„Multi-Instanz-Warteschlangenmanager auf Workstations oder Servern in einer Domäne erstellen“](#) auf Seite 377 und [„Multi-Instanz-Warteschlangenmanager auf Domänencontrollern erstellen“](#) auf Seite 393 sind Beispiele für die Konfiguration eines Multi-Instanz-Warteschlangenmanagers.

Auf Windows-, UNIX and Linux -Systemen können Sie eine einzelne `mq5.ini`-Datei gemeinsam nutzen, indem Sie sie in die Netzfreigabe stellen und die Umgebungsvariable `AMQ_MQS_INI_LOCATION` so festlegen, dass sie darauf verweist.

## Einschränkungen

1. Konfigurieren Sie mehrere Instanzen desselben Warteschlangenmanagers nur auf Servern, die dasselbe Betriebssystem, dieselbe Architektur und Endian-Format aufweisen. Beide Maschinen müssen zum Beispiel entweder 32-Bit oder 64-Bit sein.
2. Alle IBM WebSphere MQ-Installationen müssen Release-Level 7.0.1 oder höher aufweisen.
3. In der Regel werden aktive Installationen und Standby-Installationen auf derselben Wartungsstufe verwaltet. Lesen Sie die Wartungsanweisungen für jedes Upgrade, um zu überprüfen, ob Sie alle Installationen gemeinsam aktualisieren müssen.



Beachten Sie, dass die Wartungsstufen für die aktiven und passiven WS-Manager identisch sein müssen.

4. Teilen Sie Warteschlangenmanagerdaten und -protokolle nur zwischen Warteschlangenmanagern, die mit demselben IBM WebSphere MQ-Benutzer, derselben Gruppe und demselben Zugriffssteuerungsmechanismus konfiguriert sind.
5. Auf UNIX and Linux-Systemen konfigurieren Sie das gemeinsam genutzte Dateisystem auf dem vernetzten Speicher am besten mit einem unbedingten, unterbrechbaren Mount, statt mit einem bedingten Mount. Eine unterbrechungsfreie Halterung erzwingt die Blockierung des Warteschlangenmanagers, bis die Unterbrechung durch einen Systemaufruf unterbrochen wird. Weiche Mounts stellen keine Datenkonsistenz nach einem Serverausfall sicher.
6. Das gemeinsam genutzte Protokoll und die Datenverzeichnisse können nicht auf einem FAT oder einem NFSv3-Dateisystem gespeichert werden. Bei Multi-Instanz-Warteschlangenmanagern unter Windows muss der Zugriff auf den vernetzten Speicher mit dem CIFS-Protokoll (Common Internet File System), das in Windows-Netzen verwendet wird, ausgeführt werden.

#### *Windows-Domänen und Multi-Instanz-Warteschlangenmanager*

Ein Multi-Instanz-Warteschlangenmanager auf Windows setzt voraus, dass seine Daten und Protokolle gemeinsam genutzt werden. Die Freigabe muss für alle Instanzen des Warteschlangenmanagers, die auf verschiedenen Servern oder Workstations ausgeführt werden, zugänglich sein. Konfigurieren Sie die Warteschlangenmanager und nutzen Sie sie als Teil einer Windows-Domäne gemeinsam. Der Warteschlangenmanager kann auf einer Domänenworkstation oder einem Server oder auf dem Domänencontroller ausgeführt werden.

Lesen Sie vor der Konfiguration eines Warteschlangenmanagers mit mehreren Instanzen „Nicht gemeinsam genutzte WS-Manager-Daten und -Protokollverzeichnisse und -Dateien unter Windows schützen“ auf Seite 406 und „Gemeinsam genutzte Warteschlangenmanagerdaten, Protokollverzeichnisse und Dateien unter Windows sichern“ auf Seite 403, um zu prüfen, wie der Zugriff auf WS-Manager-Daten und -Protokolldateien gesteuert werden kann. Die Themen sind erzieherische Informationen. Wenn Sie gemeinsam genutzte Verzeichnisse für einen Multi-Instanz-Warteschlangenmanager in einer Windows-Domäne einrichten möchten, finden Sie weitere Informationen in „Multi-Instanz-Warteschlangenmanager auf Workstations oder Servern in einer Domäne erstellen“ auf Seite 377.

## **Multi-Instanz-WS-Manager auf Domänenworkstations oder Servern ausführen**

Ab Version 7.1 werden Multi-Instanz-Warteschlangenmanager auf einer Workstation oder einem Server ausgeführt, die/der Mitglied einer Domäne ist. Vor Version 7.1 wurden Multi-Instanz-Warteschlangenmanager nur auf Domänencontrollern ausgeführt (siehe Abschnitt „Multi-Instanz-WS-Manager auf Domänencontrollern ausführen“ auf Seite 377). Um einen Multi-Instanz-Warteschlangenmanager unter Windows auszuführen, benötigen Sie einen Domänencontroller, einen Dateiserver sowie zwei Workstations oder Server, auf denen derselbe Warteschlangenmanager ausgeführt wird, der mit derselben Domäne verbunden ist.

Die Änderung, die die Ausführung eines Warteschlangenmanagers mit mehreren Instanzen auf einem beliebigen Server oder einer Workstation in einer Domäne ermöglicht, besteht darin, dass Sie jetzt einen Warteschlangenmanager mit einer zusätzlichen Sicherheitsgruppe erstellen können. Die zusätzliche Sicherheitsgruppe wird im Befehl `crtmqm` im Parameter `-a` übergeben. Sie sichern die Verzeichnisse, die die Daten des Warteschlangenmanagers enthalten, und Protokolle mit der Gruppe. Die Benutzer-ID, die WS-Manager-Prozesse ausführt, muss ein Mitglied dieser Gruppe sein. Wenn der Warteschlangenmanager auf die Verzeichnisse zugreift, überprüft Windows die Berechtigungen, über die die Benutzer-ID für den Zugriff auf die Verzeichnisse verfügt. Wenn sowohl die Gruppe als auch der Benutzer-ID-Domänenbereich angegeben wird, verfügt die Benutzer-ID, die die WS-Manager-Prozesse ausführt, über Berechtigungsnachweise aus der globalen Gruppe. Wenn der WS-Manager auf einem anderen Server ausgeführt wird, kann die Benutzer-ID, die die WS-Manager-Prozesse ausführt, dieselben Berechtigungsnachweise haben. Die Benutzer-ID muss nicht identisch sein. Es muss ein Mitglied der alternativen Sicherheitsgruppe sowie ein Mitglied der lokalen `mqm`-Gruppe sein.

Die Task zum Erstellen eines Warteschlangenmanagers mit mehreren Instanzen ist dieselbe wie in Version 7.0.1 mit einer Änderung. Sie müssen den zusätzlichen Sicherheitsgruppennamen zu den Parametern



des Befehls **crtmqm** hinzufügen. Die Task wird im Abschnitt [„Multi-Instanz-Warteschlangenmanager auf Workstations oder Servern in einer Domäne erstellen“](#) auf Seite 377 beschrieben.

Es sind mehrere Schritte erforderlich, um die Domäne und die Domänenserver und Workstations zu konfigurieren. Sie müssen wissen, wie Windows den Zugriff eines Warteschlangenmanagers auf seine Daten- und Protokollverzeichnisse autorisiert. Wenn Sie nicht sicher sind, wie Warteschlangenmanagerprozesse berechtigt sind, auf ihre Protokoll- und Datendateien zuzugreifen, lesen Sie den Abschnitt [„Nicht gemeinsam genutzte WS-Manager-Daten und -Protokollverzeichnisse und -Dateien unter Windows schützen“](#) auf Seite 406. Das Thema enthält zwei Tasks, die Ihnen helfen, die Schritte zu verstehen, die erforderlich sind. Die Tasks sind [„Lesen und Schreiben von Daten und Protokolldateien, die von der lokalen mqm-Gruppe autorisiert sind“](#) auf Seite 408 und [„Lesen und Schreiben von Daten- und Protokolldateien, die von einer alternativen lokalen Sicherheitsgruppe autorisiert sind“](#) auf Seite 411. In einem weiteren Abschnitt, [„Gemeinsam genutzte Warteschlangenmanagerdaten, Protokollverzeichnisse und Dateien unter Windows sichern“](#) auf Seite 403, wird erläutert, wie gemeinsam genutzte Verzeichnisse mit Warteschlangenmanagerdaten und Protokolldateien mit der alternativen Sicherheitsgruppe gesichert werden. Der Abschnitt umfasst vier Tasks, die Einrichtung einer Windows-Domäne, die Erstellung einer Dateifreigabe, die Installation von IBM WebSphere MQ for Windows und die Konfiguration eines Warteschlangenmanagers für die Verwendung der Freigabe. Die Aufgaben lauten wie folgt:

1. [„Active Directory und DNS-Domäne für IBM WebSphere MQ“](#) auf Seite 380.
2. [„IBM WebSphere MQ auf einem Server oder einer Workstation in einer Windows-Domäne installieren“](#) auf Seite 384.
3. [„Gemeinsam genutztes Verzeichnis für Warteschlangenmanagerdaten und Protokolldateien erstellen“](#) auf Seite 387.
4. [„Lesen und Schreiben von gemeinsam genutzten Daten und Protokolldateien, die von einer alternativen globalen Sicherheitsgruppe autorisiert sind“](#) auf Seite 390.

Anschließend können Sie die Task [„Multi-Instanz-Warteschlangenmanager auf Workstations oder Servern in einer Domäne erstellen“](#) auf Seite 377 unter Verwendung der Domäne ausführen. Führen Sie diese Tasks aus, um die Konfiguration eines Warteschlangenmanagers mit mehreren Instanzen zu untersuchen, bevor Sie Ihre Kenntnisse in eine Produktionsdomäne übertragen.

## **Multi-Instanz-WS-Manager auf Domänencontrollern ausführen**

In Version 7.0.1 wurden Multi-Instanz-Warteschlangenmanager nur auf Domänencontrollern ausgeführt. Die WS-Manager-Daten konnten mit der Domäne mqm gesichert werden. Wie im Abschnitt [„Gemeinsam genutzte Warteschlangenmanagerdaten, Protokollverzeichnisse und Dateien unter Windows sichern“](#) auf Seite 403 erläutert, können Sie Verzeichnisse, die mit der lokalen mqm-Gruppe auf Workstations oder Servern geschützt sind, nicht gemeinsam nutzen. Auf Domänencontrollern haben jedoch alle Gruppen- und Principals einen Domänenbereich. Wenn Sie IBM WebSphere MQ for Windows auf einem Domänencontroller installieren, werden die Daten des Warteschlangenmanagers und die Protokolldateien mit der Domäne mqm geschützt, die gemeinsam genutzt werden kann. Führen Sie die Schritte in der Task [„Multi-Instanz-Warteschlangenmanager auf Domänencontrollern erstellen“](#) auf Seite 393 aus, um einen Multi-Instanz-Warteschlangenmanager auf Domänencontrollern zu konfigurieren.

### **Zugehörige Informationen**

[Clusterknoten mit Windows 2000, Windows Server 2003 und Windows Server 2008 als Domänencontroller](#)

#### *Multi-Instanz-Warteschlangenmanager auf Workstations oder Servern in einer Domäne erstellen*

Anhand eines Beispiels wird veranschaulicht, wie ein Multi-Instanz-Warteschlangenmanager auf einer Workstation oder einem Server unter Windows eingerichtet wird, der Teil einer Windows-Domäne ist. Der Server muss kein Domänencontroller sein. In der Konfiguration werden die verwendeten Konzepte und nicht die Produktionsmaßstab, sondern die Konzepte veranschaulicht. Das Beispiel basiert auf Windows Server 2008. Möglicherweise weichen die Schritte auf anderen Windows Server-Versionen im Einzelnen ab.

In einer Produktionsmaßstabskonfiguration müssen Sie die Konfiguration möglicherweise an eine vorhandene Domäne anpassen. Sie können z. B. verschiedene Domänengruppen definieren, um unterschiedliche Freigaben zu berechtigen und die Benutzer-IDs zu gruppieren, die Warteschlangenmanager ausführen.

Die Beispielkonfiguration besteht aus drei Servern:

#### **sun**

Ein Domänencontroller unter Windows Server 2008. Sie ist Eigner der *wmq.example.com*-Domäne, die *Sun*, *Mars* und *Venus* enthält. Für die Zwecke der Veranschaulichung wird sie auch als Dateiserver verwendet.

#### **Mars**

Ein Windows Server 2008, der als erster IBM WebSphere MQ -Server verwendet wird. Sie enthält eine Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

#### **Venus**

Ein Windows Server 2008, der als zweiter IBM WebSphere MQ -Server verwendet wird. Sie enthält die zweite Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

Ersetzen Sie die kursiv dargestellten Namen im Beispiel durch die Namen Ihrer Wahl.

## **Vorbereitende Schritte**

Unter Windows müssen Sie das Dateisystem, in dem Sie Warteschlangenmanagerdaten und Protokolldateien speichern möchten, nicht überprüfen. Die Prüfprozedur Verhalten des gemeinsam genutzten Dateisystems überprüfen gilt für UNIX and Linux. Unter Windows sind die Prüfungen immer erfolgreich.

Führen Sie die Schritte in den folgenden Tasks aus. Mit den Tasks wird der Domänencontroller und die Domäne erstellt, IBM WebSphere MQ for Windows auf einem Server installiert und die Dateifreigabe für Daten- und Protokolldateien erstellt. Wenn Sie einen bereits vorhandenen Domänencontroller konfigurieren möchten, ist es unter Umständen sinnvoll, die durchzuführenden Schritte an einem neuen Windows Server 2008 auszuprobieren. Sie können die Schritte an Ihre Domäne anpassen.

1. „Active Directory und DNS-Domäne für IBM WebSphere MQ“ auf Seite 380.
2. „IBM WebSphere MQ auf einem Server oder einer Workstation in einer Windows-Domäne installieren“ auf Seite 384.
3. „Gemeinsam genutztes Verzeichnis für Warteschlangenmanagerdaten und Protokolldateien erstellen“ auf Seite 387.
4. „Lesen und Schreiben von gemeinsam genutzten Daten und Protokolldateien, die von einer alternativen globalen Sicherheitsgruppe autorisiert sind“ auf Seite 390.

## **Informationen zu diesem Vorgang**

Diese Task ist eine der Tasks, die zum Konfigurieren eines Domänencontrollers und zwei Servern in der Domäne ausgeführt werden, um Instanzen eines Warteschlangenmanagers auszuführen. In dieser Task konfigurieren Sie einen zweiten Server (*Venus*), um eine weitere Instanz des Warteschlangenmanagers *QMGR* auszuführen. Führen Sie die Schritte in dieser Task aus, um die zweite Instanz des Warteschlangenmanagers *QMGR* zu erstellen und zu testen, ob sie funktioniert.

Diese Task ist von den vier Tasks im vorherigen Abschnitt getrennt. Sie enthält die Schritte, mit denen ein einzelner Instanz-WS-Manager in einen Warteschlangenmanager mit mehreren Instanzen konvertiert wird. Alle anderen Schritte sind für einzelne oder mehrere Instanzen von Warteschlangenmanagern mit mehreren Instanzen gemeinsam.

## **Vorgehensweise**

1. Konfigurieren Sie einen zweiten Server für die Ausführung von IBM WebSphere MQ for Windows.
  - a) Führen Sie die Schritte der Task „IBM WebSphere MQ auf einem Server oder einer Workstation in einer Windows-Domäne installieren“ auf Seite 384 aus, um einen zweiten Domänenserver zu erstellen. In dieser Folge von Tasks wird der zweite Server als *Venus* bezeichnet.

**Tipp:** Erstellen Sie die zweite Installation. Achten Sie dabei darauf, dass Sie IBM WebSphere MQ auf beiden Servern mit denselben Einstellungen installieren. Wenn die Standardwerte abweichen, müssen Sie unter Umständen die Variablen Präfix und Installationsname in der **QMGR QueueManager**-Zeilegruppe der IBM WebSphere MQ-Konfigurationsdatei `mq5.ini` anpassen. Die Variablen beziehen sich auf Pfade, die sich für jede Installation und jeden Warteschlangenmanager auf jedem Server unterscheiden können. Wenn die Pfade auf jedem Server gleich bleiben, ist es einfacher, einen WS-Manager mit mehreren Instanzen zu konfigurieren.

2. Erstellen Sie eine zweite Instanz von **QMGR** auf *venus*.

- a) Wenn **QMGR** on *mars* nicht vorhanden ist, führen Sie die Task „Lesen und Schreiben von gemeinsam genutzten Daten und Protokolldateien, die von einer alternativen globalen Sicherheitsgruppe autorisiert sind“ auf Seite 390 aus, um sie zu erstellen.
- b) Überprüfen Sie die Werte der Parameter `Prefix` und `InstallationName` für *venus*.

Führen Sie unter *mars* den Befehl **dspmqinf** aus:

```
dspmqinf QMGR
```

Systemantwort:

```
QueueManager:
 Name=QMGR
 Directory=QMGR
 Prefix=C:\Program Files\IBM\WebSphere MQ
 DataPath=\\sun\wmq\data\QMGR
 InstallationName=Installation1
```

- c) Kopieren Sie die maschinenlesbare Form der Zeilegruppe **QueueManager** in die Zwischenablage.

Führen Sie unter *mars* den Befehl **dspmqinf** erneut mit dem Parameter `-o command` aus.

```
dspmqinf -o command QMGR
```

Systemantwort:

```
addmqinf -s QueueManager -v Name=QMGR
-v Directory=QMGR -v Prefix="C:\Program Files\IBM\WebSphere MQ"
-v DataPath=\\sun\wmq\data\QMGR
```

- d) Führen Sie unter *venus* den Befehl **addmqinf** aus der Zwischenablage heraus aus, um eine Instanz des Warteschlangenmanagers unter *venus* zu erstellen.

Passen Sie den Befehl bei Bedarf an, um die Unterschiede in den Parametern `Prefix` oder `InstallationName` zu berücksichtigen.

```
addmqinf -s QueueManager -v Name=QMGR
-v Directory=QMGR -v Prefix="C:\Program Files\IBM\WebSphere MQ"
-v DataPath=\\sun\wmq\data\QMGR
```

WebSphere MQ configuration information added.

3. Starten Sie den Warteschlangenmanager **QMGR** unter *venus*, um Standby-Instanzen zu ermöglichen.

- a) Die Option **QMGR** auf *mars* wird gestoppt.

Führen Sie unter *mars* den Befehl **dspmq** aus:

```
dspmq -m QMGR
```

Die Systemantwort hängt davon ab, wie der WS-Manager gestoppt wurde. Beispiel:

```
C:\Users\Administrator>dspmq -m QMGR
QMNAME(QMGR) STATUS(Ended immediately)
```

- b) Führen Sie unter *venus* den Befehl **strmqm** aus, damit **QMGR** die Bereitschaftsdatenbanken zulässt:

```
strmqm -x QMGR
```

Systemantwort:

```
WebSphere MQ queue manager 'QMGR' starting.
The queue manager is associated with installation 'Installation1'.
5 log records accessed on queue manager 'QMGR' during the log
replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
WebSphere MQ queue manager 'QMGR' started using V7.1.0.0.
```

## Ergebnisse

Führen Sie die folgenden Schritte aus, um den WS-Manager mit mehreren Instanzen zu testen:

1. Führen Sie unter *marsden* Befehl **strmqm** aus, um *QMGR* zu starten, das Bereitschaftsdatenbanken zulässt:

```
strmqm -x QMGR
```

Systemantwort:

```
WebSphere MQ queue manager 'QMGR' starting.
The queue manager is associated with installation 'Installation1'.
A standby instance of queue manager 'QMGR' has been started.
The active instance is running elsewhere.
```

2. Führen Sie unter *venus* den Befehl **endmqm** aus:

```
endmqm -r -s -i QMGR
```

Systemantwort auf *venus*:

```
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ended, permitting switchover to
a standby instance.
```

Und auf *mars*:

```
dspmq
QMNAME(QMGR) STATUS(Running as standby)
C:\Users\wmquser2>dspmq
QMNAME(QMGR) STATUS(Running as standby)
C:\Users\wmquser2>dspmq
QMNAME(QMGR) STATUS(Running)
```

## Nächste Schritte

Ein Mehrinstanz-WS-Manager kann auch mit Musterprogrammen überprüft werden. Informationen hierzu finden Sie im Abschnitt [„Warteschlangenmanager mit mehreren Instanzen unter Windows überprüfen“](#) auf Seite 400.

*Active Directory und DNS-Domäne für IBM WebSphere MQ*

Mit dieser Task erstellen Sie die Domäne *wmq.example.com* auf einem Windows 2008-Domänencontroller namens *sun*. Er konfiguriert die globale Gruppe `Domain_mqm` in der Domäne mit den korrekten Berechtigungen und mit einem Benutzer.

In einer Produktionsmaßstabskonfiguration müssen Sie die Konfiguration möglicherweise an eine vorhandene Domäne anpassen. Sie können z. B. verschiedene Domänengruppen definieren, um unterschiedliche Freigaben zu berechtigen und die Benutzer-IDs zu gruppieren, die Warteschlangenmanager ausführen.

Die Beispielkonfiguration besteht aus drei Servern:

#### ***sun***

Ein Domänencontroller unter Windows Server 2008. Sie ist Eigner der *wmq.example.com*-Domäne, die *Sun*, *mars* und *venus* enthält. Für die Zwecke der Veranschaulichung wird sie auch als Dateiserver verwendet.

#### ***mars***

Ein Windows Server 2008, der als erster IBM WebSphere MQ -Server verwendet wird. Sie enthält eine Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

#### ***venus***

Ein Windows Server 2008, der als zweiter IBM WebSphere MQ -Server verwendet wird. Sie enthält die zweite Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

Ersetzen Sie die kursiv dargestellten Namen im Beispiel durch die Namen Ihrer Wahl.

### **Vorbereitende Schritte**

1. Die Taskschritte sind konsistent mit einem Windows Server 2008, der installiert, aber nicht mit Rollen konfiguriert ist. Wenn Sie einen bereits vorhandenen Domänencontroller konfigurieren möchten, ist es unter Umständen sinnvoll, die durchzuführenden Schritte an einem neuen Windows Server 2008 auszuprobieren. Sie können die Schritte an Ihre Domäne anpassen.

### **Informationen zu diesem Vorgang**

In dieser Task erstellen Sie eine Active Directory- und DNS-Domäne auf einem neuen Domänencontroller. Anschließend nehmen Sie die erforderliche Konfiguration für die Installation von IBM WebSphere MQ auf anderen Servern und Workstations vor, die derselben Domäne beitreten. Führen Sie die Task aus, wenn Sie mit der Installation und Konfiguration von Active Directory nicht vertraut sind, um eine Windows -Domäne zu erstellen. Sie müssen eine Windows -Domäne erstellen, um eine Multi-Instanz-WS-Manager-Konfiguration zu erstellen. Ziel dieser Task ist es nicht, eine optimale Konfiguration einer Windows-Domäne zu erzielen. Für die Bereitstellung von Multi-Instanz-Warteschlangenmanagern in einer Produktionsumgebung ist die Windows-Dokumentation zu konsultieren.

Während der Task führen Sie die folgenden Schritte aus:

1. Installieren Sie Active Directory.
2. Fügen Sie eine Domäne hinzu.
3. Fügen Sie die Domäne dem DNS hinzu.
4. Erstellen Sie die globale Gruppe `Domain\mqm` und geben Sie ihr die richtigen Berechtigungen.
5. Fügen Sie einen Benutzer hinzu und machen Sie ihn zu einem Mitglied der globalen Gruppe `Domain\mqm`.

Diese Task ist eine von einer Gruppe zusammengehöriger Tasks, die den Zugriff auf Warteschlangenmanagerdaten und Protokolldateien veranschaulichen. Die Tasks zeigen, wie ein Warteschlangenmanager erstellt wird, der berechtigt ist, Daten und Protokolldateien zu lesen und zu schreiben, die in einem Verzeichnis Ihrer Wahl gespeichert sind. Sie begleiten die Task [„Windows-Domänen und Multi-Instanz-Warteschlangenmanager“](#) auf Seite 376.

Bei dieser Task lautet der Hostname des Domänencontrollers *sun* und die beiden IBM WebSphere MQ-Server heißen *mars* und *venus*. Die Domäne wird als *wmq.example.com* bezeichnet. Sie können alle in Kursivdruck angezeigten Namen in der Task durch Namen Ihrer Wahl ersetzen.

## Vorgehensweise

1. Melden Sie sich als lokaler Administrator oder Workgroup-Administrator am Domänencontroller (*sun*) an.  
Wenn der Server bereits als Domänencontroller konfiguriert ist, müssen Sie sich als Domänenadministrator anmelden.
2. Führen Sie den Assistenten für die Active Directory-Domänenservices aus.
  - a) Klicken Sie auf **Start > Ausführen ...**. Geben Sie `dcprmo` ein und klicken Sie auf **OK**.  
Wenn die Active Directory-Binärdateien noch nicht installiert sind, installiert Windows die Dateien automatisch.
3. Lassen Sie im ersten Fenster des Assistenten das Kontrollkästchen **Erweiterte Modusinstallation verwenden** leer. Klicken Sie auf **Weiter > Weiter** und anschließend auf **Neue Domäne in einer neuen Gesamtstruktur erstellen > Weiter**.
4. Geben Sie `wmq.example.com` in das Feld **FQDN of the forest root domain** ein. Klicken Sie auf **Weiter**.
5. Wählen Sie im Fenster "Gesamtstrukturfunktionsebene festlegen" **Windows Server 2003** oder höher aus der Liste der **Gesamtstrukturfunktionsebenen > Weiteraus**.  
Die älteste Version von Windows Server, die von IBM WebSphere MQ unterstützt wird, ist Windows Server 2003.
6. Optional: Wählen Sie im Fenster "Domänenfunktionsebene festlegen" **Windows Server 2003** oder höher aus der Liste der **Domänenfunktionsebenen > Weiteraus**.  
Dieser Schritt ist nur erforderlich, wenn Sie als Funktionsebene der Gesamtstruktur **Windows Server 2003** festlegen.
7. Das Fenster "Zusätzliche Domänencontrolleroptionen" wird geöffnet, wobei **DNS-Server** als zusätzliche Option ausgewählt ist. Klicken Sie auf **Weiter** und **Ja**, um das Warnfenster zu löschen.  
 **Tipp:** Wenn bereits ein DNS-Server installiert ist, wird diese Option nicht angezeigt. Wenn Sie diese Task genau verfolgen möchten, entfernen Sie alle Rollen von diesem Domänencontroller und starten Sie sie erneut.
8. Lassen Sie die Verzeichnisse Database, Log Files und SYSVOL unverändert. Klicken Sie auf **Weiter**.
9. Geben Sie ein Kennwort in die Felder **Kennwort** und **Kennwort bestätigen** im Fenster 'Administrator-kennwort für Directory Services Restore Mode' ein. Klicken Sie auf **Weiter > Weiter**. Wählen Sie im letzten Fenster des Assistenten die Option **Reboot on completion** (Warmstart bei Beendigung) aus.
10. Wenn der Domänencontroller einen Warmstart durchgeführt hat, melden Sie sich als `wmq\Administrator` an.  
Der Servermanager wird automatisch gestartet.
11. Öffnen Sie den Ordner `wmq.example.com\Users`.
  - a) Öffnen Sie **Server Manager > Roles > Active Directory Domain Services > wmq.example.com > Users**.
12. Klicken Sie mit der rechten Maustaste auf **Benutzer > Neu > Gruppe**.
  - a) Geben Sie einen Gruppennamen in das Feld **Gruppenname** ein.  
**Anmerkung:** Der bevorzugte Gruppenname ist `Domain\mqm`. Geben Sie ihn genau wie angezeigt ein.
    - Durch den Aufruf der Gruppe `Domain\mqm` ändert sich das Verhalten des Assistenten "Prepare IBM WebSphere MQ" (WebSphere MQ vorbereiten) auf einer Workstation bzw. einem Server der Domäne. Der "IBM WebSphere MQ-Vorbereitungsassistent" fügt die Gruppe `Domain\mqm` automatisch der lokalen Gruppe `mqm` jeder neuen IBM WebSphere MQ-Installation in der Domäne hinzu.
    - Sie können Workstations oder Server in einer Domäne ohne globale `Domain\mqm`-Gruppe installieren. Wenn Sie dies tun, müssen Sie eine Gruppe mit denselben Eigenschaften wie

die Gruppe `Domain mqm` definieren. Machen Sie die Gruppe oder ihre Benutzermitglieder zu Mitgliedern der lokalen Gruppe `mqm`, sobald IBM WebSphere MQ in einer Domäne installiert wird. Sie können Domänenbenutzer in mehrere Gruppen stellen. Erstellen Sie mehrere Domänengruppen, wobei jede Gruppe einer Gruppe von Installationen entspricht, die Sie separat verwalten möchten. Teilen Sie Domänenbenutzer entsprechend den Installationen, die sie verwalten, in verschiedene Domänengruppen auf. Fügen Sie jede Domänengruppe der lokalen Gruppe `mqm` unterschiedlicher IBM WebSphere MQ-Installationen hinzu. Nur Domänenbenutzer in den Domänengruppen, die Mitglieder einer bestimmten lokalen `mqm`-Gruppe sind, können Warteschlangenmanager für diese Installation erstellen, verwalten und ausführen.

- Der Domänenbenutzer, den Sie bei der Installation von IBM WebSphere MQ auf einer Workstation oder einem Server in einer Domäne angeben, muss Mitglied der Gruppe `Domain mqm` oder einer alternativen Gruppe sein, die Sie mit denselben Eigenschaften wie die Gruppe `Domain mqm` definiert haben.

b) Lassen Sie **Global** als **Gruppenbereich** geklickt oder ändern Sie ihn in **Universal**. Lassen Sie **Sicherheit** als **Gruppentyp** geklickt. Klicken Sie auf **OK**.

13. Fügen Sie die Berechtigungen **Allow Read group membership** und **Allow Read groupMembershipSAM** zu den Berechtigungen der globalen Gruppe `Domain mqm` hinzu.

a) Klicken Sie in der Aktionsleiste des Server-Managers auf **Ansicht > Erweiterte Funktionen**

b) Klicken Sie in der Navigationsstruktur des Servermanagers auf **Benutzer**.

c) Klicken Sie im Fenster "Benutzer" mit der rechten Maustaste auf **Domain mqm > Eigenschaften**

d) Klicken Sie auf **Sicherheit > Erweitert > Hinzufügen ...**. Geben Sie `Domain mqm` ein und klicken Sie auf **Namen prüfen > OK**.

Das Feld **Name** wird mit der Zeichenfolge `Domain mqm (domain name\Domain mqm)` ausgefüllt.

e) Klicken Sie auf **Eigenschaften**. Wählen Sie unten in der Liste **Anwenden auf** die Option **Untergeordnete Benutzerobjekte** aus.

f) Wählen Sie in der Liste **Berechtigungen** die Kontrollkästchen **Gruppenzugehörigkeit lesen** und **Gruppenzugehörigkeit lesen groupMembershipSAM Zulassen** aus; klicken Sie auf **OK > Anwenden > OK > OK**.

14. Fügen Sie zwei oder mehr Benutzer zur globalen Gruppe `Domain mqm` hinzu.

Ein Benutzer, im Beispiel `wmquser1`, führt den IBM WebSphere MQ -Service aus und der andere Benutzer, `wmquser2`, wird interaktiv verwendet.

Ein Domänenbenutzer ist für die Erstellung eines Warteschlangenmanagers erforderlich, der die alternative Sicherheitsgruppe in einer Domänenkonfiguration verwendet. Es reicht nicht aus, wenn die Benutzer-ID ein Administrator ist, obwohl ein Administrator über die Berechtigung zum Ausführen des Befehls `crtmqm` verfügt. Der Domänenbenutzer, der ein Administrator sein kann, muss ein Mitglied der lokalen `mqm`-Gruppe sowie der alternativen Sicherheitsgruppe sein.

In diesem Beispiel nehmen Sie `wmquser1`- und `wmquser2`-Member der globalen Gruppe `Domain mqm` ab. Der Assistent "IBM WebSphere MQ" konfiguriert `Domain mqm` automatisch als Mitglied der lokalen `mqm`-Gruppe, wo immer der Assistent ausgeführt wird.

Sie müssen einen anderen Benutzer angeben, um den IBM WebSphere MQ -Service für jede Installation von IBM WebSphere MQ auf einem einzelnen Computer auszuführen. Sie können dieselben Benutzer auf verschiedenen Computern verwenden.

a) Klicken Sie in der Navigationsstruktur des Server-Managers auf **Benutzer > Neu > Benutzer**

b) Geben Sie im Fenster 'Neues Objekt-Benutzer' `wmquser1` in das Feld **Benutzeranmeldename** ein. Geben Sie `WebSphere` in das Feld **Vorname** und `MQ1` in das Feld **Nachname** ein. Klicken Sie auf **Weiter**.

c) Geben Sie ein Kennwort in die Felder **Kennwort** und **Kennwort bestätigen** ein, und inaktivieren Sie das Kontrollkästchen **Benutzer muss Kennwort bei der nächsten Anmeldung ändern**. Klicken Sie auf **Weiter > Fertig stellen**.

- d) Klicken Sie im Fenster "Users" auf **WebSphere MQ > Add to a group ...** . Geben Sie Domain mqm ein und klicken Sie auf **Namen überprüfen > OK > OK**.
- e) Wiederholen Sie die Schritte a bis d , um *WebSphere MQ2* als *wmuser2* hinzuzufügen.
15. IBM WebSphere MQ wird als Service ausgeführt.

Wenn Sie IBM WebSphere MQ als Service ausführen möchten und dem Domänenbenutzer, den Sie von Ihrem Domänenadministrator erhalten haben, danach das Recht zur Ausführung als Service erteilen wollen, führen Sie die folgenden Schritte aus:

- a) Klicken Sie auf **Start> Ausführen...**

Geben Sie den Befehl `secpol.msc` ein und klicken Sie auf **OK** .

- b) Öffnen Sie **Sicherheitseinstellungen > Local Policies (Lokale Richtlinien) > User Rights Assignments** (Zuweisen von Benutzerrechten).

Klicken Sie in der Liste der Richtlinien mit der rechten Maustaste auf **Log on as a service** (Anmelden als Service) > **Eigenschaften**.

- c) Klicken Sie auf **Add User or Group** (Benutzer oder Gruppe hinzufügen).

Geben Sie den Namen des Benutzers ein, den Sie von Ihrem Domänenadministrator erhalten haben, und klicken Sie auf **Check Names** (Namen überprüfen).

- d) Wenn Sie in einem Fenster für die Windows -Sicherheit dazu aufgefordert werden, geben Sie den Benutzernamen und das Kennwort eines Kontobenutzers oder Administrators mit ausreichender Berechtigung ein und klicken Sie auf **OK > Anwenden > OK**.

Schließen Sie das Fenster "Lokale Sicherheitsrichtlinie".

**Anmerkung:** Unter Windows Vista und Windows Server 2008 ist die Benutzerkontensteuerung (User Account Control, UAC) standardmäßig aktiviert.

Diese Funktion schränkt die Operationen ein, die Benutzer (selbst wenn sie zur Gruppe 'Administratoren' gehören) für bestimmte Komponenten des Betriebssystems ausführen können. Zur Umgehung dieser Einschränkung müssen Sie bestimmte Schritte ausführen.

## Nächste Schritte

Fahren Sie mit der nächsten Task, „[IBM WebSphere MQ auf einem Server oder einer Workstation in einer Windows-Domäne installieren](#)“ auf Seite 384, fort.

*IBM WebSphere MQ auf einem Server oder einer Workstation in einer Windows-Domäne installieren*

In dieser Task installieren und konfigurieren Sie IBM WebSphere MQ auf einem Server oder einer Workstation in der *wmq.example.com* Windows -Domäne.

In einer Produktionsmaßstabskonfiguration müssen Sie die Konfiguration möglicherweise an eine vorhandene Domäne anpassen. Sie können z. B. verschiedene Domänengruppen definieren, um unterschiedliche Freigaben zu berechtigen und die Benutzer-IDs zu gruppieren, die Warteschlangenmanager ausführen.

Die Beispielkonfiguration besteht aus drei Servern:

### **sun**

Ein Domänencontroller unter Windows Server 2008. Sie ist Eigner der *wmq.example.com* -Domäne, die *Sun*, *mars* und *venus* enthält. Für die Zwecke der Veranschaulichung wird sie auch als Dateiserver verwendet.

### **mars**

Ein Windows Server 2008, der als erster IBM WebSphere MQ -Server verwendet wird Sie enthält eine Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

### **venus**

Ein Windows Server 2008, der als zweiter IBM WebSphere MQ -Server verwendet wird Sie enthält die zweite Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

Ersetzen Sie die kursiv dargestellten Namen im Beispiel durch die Namen Ihrer Wahl.



## Vorbereitende Schritte

1. Führen Sie die im Abschnitt „Active Directory und DNS-Domäne für IBM WebSphere MQ“ auf Seite [380](#) beschriebenen Schritte zum Erstellen eines Domänencontrollers *sun* für die Domäne *wmq.example.com* aus. Ändern Sie die in Kursivdruck angezeigten Namen Ihrer Konfiguration entsprechend.
2. Im Abschnitt [Hardware- und Softwarevoraussetzungen auf Windows-Systemen](#) finden Sie Informationen zu weiteren Windows-Versionen, auf denen IBM WebSphere MQ ausgeführt werden kann.

## Informationen zu diesem Vorgang

In dieser Task konfigurieren Sie einen Windows Server 2008 mit dem Namen *mars* als Mitglied der *wmq.example.com*-Domäne. Sie installieren IBM WebSphere MQ und konfigurieren die Installation so, dass sie als Mitglied der *wmq.example.com*-Domäne ausgeführt wird.

Diese Task ist eine von einer Gruppe zusammengehöriger Tasks, die den Zugriff auf Warteschlangenmanagerdaten und Protokolldateien veranschaulichen. Die Tasks zeigen, wie ein Warteschlangenmanager erstellt wird, der berechtigt ist, Daten und Protokolldateien zu lesen und zu schreiben, die in einem Verzeichnis Ihrer Wahl gespeichert sind. Sie begleiten die Task „[Windows-Domänen und Multi-Instanz-Warteschlangenmanager](#)“ auf Seite 376.

Bei dieser Task lautet der Hostname des Domänencontrollers *sun* und die beiden IBM WebSphere MQ-Server heißen *mars* und *venus*. Die Domäne wird als *wmq.example.com* bezeichnet. Sie können alle in Kursivdruck angezeigten Namen in der Task durch Namen Ihrer Wahl ersetzen.

## Vorgehensweise

1. Fügen Sie den Domänencontroller *sun.wmq.example.com* zu *mars* als DNS-Server hinzu.
  - a) Melden Sie sich unter *mars* als *mars\Administrator* an und klicken Sie auf **Start**.
  - b) Klicken Sie mit der rechten Maustaste auf **Netz > Eigenschaften > Netzverbindungen verwalten**.
  - c) Klicken Sie auf den Netzadapter, und klicken Sie auf **Eigenschaften**.  
Das System antwortet mit dem Fenster "Merkmale des lokalen Bereichs", in dem die von der Verbindung verwendeten Elemente aufgelistet werden.
  - d) Wählen Sie aus der Liste der Einträge im Fenster 'Local Area Connection Properties' **Internet Protocol Version 4** oder **Internet Protocol Version 6** aus. Klicken Sie auf **Eigenschaften > Erweitert ...** und klicken Sie auf die Registerkarte **DNS**.
  - e) Klicken Sie unter den DNS-Serveradressen auf **Hinzufügen ...**.
  - f) Geben Sie die IP-Adresse des Domänencontrollers ein, bei dem es sich auch um den DNS-Server handelt, und klicken Sie auf **Hinzufügen**.
  - g) Klicken Sie auf **Diese DNS-Suffixe anhängen > Hinzufügen ...**.
  - h) Geben Sie *wmq.example.com* ein und klicken Sie auf **Hinzufügen**.
  - i) Geben Sie *wmq.example.com* in das Feld **DNS-Suffix für diese Verbindung** ein.
  - j) Wählen Sie die Option **Diese Verbindungsadresse in DNS registrieren** und **Das Suffix dieser Verbindung in der DNS-Registrierung verwenden** aus. Klicken Sie auf **OK > OK > Schließen**.
  - k) Öffnen Sie ein Befehlsfenster und geben Sie den Befehl **ipconfig /all** ein, um die TCP/IP-Einstellungen zu überprüfen.
2. Fügen Sie unter *mars* den Computer zur Domäne *wmq.example.com* hinzu.
  - a) Klicken Sie auf **Start**.
  - b) Klicken Sie mit der rechten Maustaste auf **Computer > Eigenschaften**. Klicken Sie im Bereich "Computername", "Domäne" und "Arbeitsgruppeneinstellungen" auf **Einstellungen ändern**.
  - c) Klicken Sie in den Fenstern "Systemeigenschaften" auf **Ändern ...**.
  - d) Klicken Sie auf Domäne, geben Sie *wmq.example.com* ein und klicken Sie auf **OK**.

- e) Geben Sie den **Benutzernamen** und das **Kennwort** des Domänencontrolleradministrators ein, der über die Berechtigung verfügt, dass der Computer der Domäne beitreten kann, und klicken Sie auf **OK**.
  - f) Klicken Sie auf **OK > OK > Schließen > Erneut starten** als Antwort auf die Nachricht "Willkommen bei der *wmq.example.com* -Domäne".
3. Überprüfen Sie, ob der Computer Mitglied der *wmq.example.com*-Domäne ist.
- a) Melden Sie sich unter *sun* an dem Domänencontroller als *wmq\Administrator* an.
  - b) Öffnen Sie **Server Manager > Active Directory Domain Services > wmq.example.com > Computers** und überprüfen Sie, ob *mars* ordnungsgemäß im Fenster 'Computers' aufgelistet ist.
4. Installieren Sie IBM WebSphere MQ for Windows auf *mars*.

Weitere Informationen zum Ausführen des Installationsassistenten von IBM WebSphere MQ for Windows finden Sie im Abschnitt IBM WebSphere MQ-Server unter Windows installieren.

- a) Melden Sie sich unter *mars* als lokaler Administrator an, *mars\Administrator*.
- b) Führen Sie den Befehl **Setup** auf den IBM WebSphere MQ for Windows -Installationsmedien aus.  
Die Launchpad-Anwendung von IBM WebSphere MQ wird gestartet.
- c) Klicken Sie auf **Softwarevoraussetzungen**, um zu überprüfen, ob die vorausgesetzte Software installiert ist.
- d) Klicken Sie auf **Netzwerkconfiguration > Ja**, um eine Domänenbenutzer-ID zu konfigurieren.  
Bei der Task „Active Directory und DNS-Domäne für IBM WebSphere MQ“ auf Seite 380 wird für diese Gruppe von Tasks eine Domänenbenutzer-ID konfiguriert.
- e) Klicken Sie auf **WebSphere MQ Installation**, wählen Sie eine Sprache für die Installation aus und klicken Sie auf 'Launch IBM WebSphere MQ Installer' (Installation von IBM WebSphere MQ starten).
- f) Bestätigen Sie die Lizenzvereinbarung und klicken Sie auf **Next > Next > Install**, um die Standardkonfiguration zu akzeptieren. Warten Sie, bis die Installation abgeschlossen ist, und klicken Sie auf **Fertig stellen**.

Möglicherweise möchten Sie den Namen der Installation ändern, verschiedene Komponenten installieren, ein anderes Verzeichnis für WS-Manager-Daten und -Protokolle konfigurieren oder in einem anderen Verzeichnis installieren. Wenn dies der Fall ist, klicken Sie auf **Angepasst** und nicht auf **Standard**.

IBM WebSphere MQ wird installiert und das Installationsprogramm startet den Assistenten "Prepare IBM WebSphere MQ" (WebSphere MQ vorbereiten).

**Wichtig:** Führen Sie den Assistenten noch nicht aus.

5. Konfigurieren Sie den Benutzer, der den IBM WebSphere MQ-Service ausführen soll, mit der Berechtigung **Run as a service** (Als Service ausführen).
- Wählen Sie aus, ob die lokale *mqm* -Gruppe, die *Domain mqm* -Gruppe oder der Benutzer, der den IBM WebSphere MQ -Service ausführen soll, mit der rechten Maustaste konfiguriert werden soll. In dem Beispiel geben Sie dem Benutzer das Recht.
- a) Klicken Sie auf **Starten > Ausführen**. Geben Sie den Befehl **secpol1.msc** ein und klicken Sie auf **OK**.
  - b) Öffnen Sie **Sicherheitseinstellungen > Lokale Richtlinien > Zuweisen von Benutzerrechten**. Klicken Sie in der Liste der Richtlinien mit der rechten Maustaste auf **Log on as a service > Properties** (Als Service anmelden > Eigenschaften).
  - c) Klicken Sie auf **Add User or Group** (Benutzer oder Gruppe hinzufügen). Geben Sie *wmquser1* ein und klicken Sie auf **Namen überprüfen**.
  - d) Geben Sie den Benutzernamen und das Kennwort eines Domänenadministrators, *wmq\Administrator*, ein und klicken Sie auf **OK > Anwenden > OK**. Schließen Sie das Fenster "Lokale Sicherheitsrichtlinie".

6. Führen Sie den Assistenten "Prepare IBM WebSphere MQ" (WebSphere MQ vorbereiten) aus.

Weitere Informationen zur Ausführung des Assistenten " IBM WebSphere MQvorbereiten" finden Sie im Abschnitt [WebSphere MQ mit dem Assistenten WebSphere MQ vorbereiten](#).

a) Vom IBM WebSphere MQ-Installationsprogramm wird "Prepare IBM WebSphere MQ" automatisch ausgeführt.

Um den Assistenten manuell zu starten, suchen Sie die Verknüpfung zu "Prepare IBM WebSphere MQ" im Ordner **Start > Alle Programme > IBM WebSphere MQ**. Wählen Sie in einer Konfiguration mit mehreren Installationen den entsprechenden Direktaufruf für die IBM WebSphere MQ-Installation aus.

b) Klicken Sie auf **Next** (Weiter) und belassen Sie die Auswahl **Yes** (Ja) auf die Frage "Identify if there is a Windows 2000 or later domain controller in the network" (Stellen Sie fest, ob im Netz ein Domänencontroller unter Windows 2000 oder höher vorhanden ist).

c) Klicken Sie im ersten Fenster 'Configuring IBM WebSphere MQ for Windows for Windows domain users' (WebSphere MQ wird für Windows-Domänenbenutzer konfiguriert) auf **Yes > Next** (Ja > Weiter).

d) Geben Sie im zweiten Fenster ' IBM WebSphere MQ for Windows für Windows -Domänenbenutzer konfigurieren' *wmq* im Feld **Domäne** ein. Geben Sie *wmquser1* in das Feld **Benutzername** und das Kennwort (falls Sie eines festgelegt haben) in das Feld **Kennwort** ein. Klicken Sie auf **Weiter**.

Der Assistent konfiguriert und startet IBM WebSphere MQ mit *wmquser1*.

e) Wählen Sie auf der letzten Seite des Assistenten die Markierungsfelder aus, die Sie benötigen, und klicken Sie auf **Fertig stellen**.

## Nächste Schritte

1. Führen Sie die Task „[Lesen und Schreiben von Daten und Protokolldateien, die von der lokalen mqm-Gruppe autorisiert sind](#)“ auf Seite 408 aus, um zu prüfen, ob die Installation und Konfiguration ordnungsgemäß funktionieren.
2. Führen Sie die Task „[Gemeinsam genutztes Verzeichnis für Warteschlangenmanagerdaten und Protokolldateien erstellen](#)“ auf Seite 387 aus, um eine Dateifreigabe zum Speichern der Daten- und Protokolldateien eines Multi-Instanz-Warteschlangenmanagers zu konfigurieren.

## Zugehörige Konzepte

[Erforderliche Benutzerberechtigungen für einen WebSphere MQ-Windows-Dienst](#)

*Gemeinsam genutztes Verzeichnis für Warteschlangenmanagerdaten und Protokolldateien erstellen*

Diese Task ist eine von einer Gruppe zusammengehöriger Tasks, die den Zugriff auf Warteschlangenmanagerdaten und Protokolldateien veranschaulichen. Die Tasks zeigen, wie ein Warteschlangenmanager erstellt wird, der berechtigt ist, Daten und Protokolldateien zu lesen und zu schreiben, die in einem Verzeichnis Ihrer Wahl gespeichert sind.

In einer Produktionsmaßstabskonfiguration müssen Sie die Konfiguration möglicherweise an eine vorhandene Domäne anpassen. Sie können z. B. verschiedene Domänengruppen definieren, um unterschiedliche Freigaben zu berechtigen und die Benutzer-IDs zu gruppieren, die Warteschlangenmanager ausführen.

Die Beispielkonfiguration besteht aus drei Servern:

### **sun**

Ein Domänencontroller unter Windows Server 2008. Sie ist Eigner der *wmq.example.com*-Domäne, die *sun*, *mars* und *venus* enthält. Für die Zwecke der Veranschaulichung wird sie auch als Dateiserver verwendet.

### **mars**

Ein Windows Server 2008, der als erster IBM WebSphere MQ -Server verwendet wird Sie enthält eine Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

## **venus**

Ein Windows Server 2008, der als zweiter IBM WebSphere MQ -Server verwendet wird Sie enthält die zweite Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

Ersetzen Sie die kursiv dargestellten Namen im Beispiel durch die Namen Ihrer Wahl.

## **Vorbereitende Schritte**

1. Wenn Sie diese Task genau wie dokumentiert ausführen möchten, führen Sie die Schritte in der Task „Active Directory und DNS-Domäne für IBM WebSphere MQ“ auf Seite 380 aus, um die Domäne *sun.wmq.example.com* auf dem Domänencontroller *sun* zu erstellen. Ändern Sie die in Kursivdruck angezeigten Namen Ihrer Konfiguration entsprechend.

## **Informationen zu diesem Vorgang**

Diese Task ist eine von einer Gruppe zusammengehöriger Tasks, die den Zugriff auf Warteschlangenmanagerdaten und Protokolldateien veranschaulichen. Die Tasks zeigen, wie ein Warteschlangenmanager erstellt wird, der berechtigt ist, Daten und Protokolldateien zu lesen und zu schreiben, die in einem Verzeichnis Ihrer Wahl gespeichert sind. Sie begleiten die Task „Windows-Domänen und Multi-Instanz-Warteschlangenmanager“ auf Seite 376.

In der Task erstellen Sie eine Freigabe, die ein Daten- und Protokollverzeichnis enthält, und eine globale Gruppe, um den Zugriff auf die Freigabe zu berechtigen. Sie übergeben den Namen der globalen Gruppe, die die Freigabe für den Befehl **crtmqm** im Parameter **-a** berechtigt. Die globale Gruppe bietet Ihnen die Flexibilität, die Benutzer dieses Anteils von den Benutzern anderer Aktien zu trennen. Wenn Sie diese Flexibilität nicht benötigen, genehmigen Sie die gemeinsame Nutzung mit der Gruppe *Domain\_mqm*, anstatt eine neue globale Gruppe zu erstellen.

Die globale Gruppe, die für die gemeinsame Nutzung in dieser Task verwendet wird, wird als *wmqha* bezeichnet, und die gemeinsam genutzte Gruppe wird als *wmq* bezeichnet. Sie werden auf dem Domänencontroller *sun* in der Windows-Domäne *wmq.example.com* definiert. Die Freigabe verfügt über vollständige Steuerberechtigungen für die globale Gruppe *wmqha*. Ersetzen Sie die italicisierten Namen in der Task durch die Namen Ihrer Wahl.

Für die Zwecke dieser Task ist der Domänencontroller derselbe Server wie der Dateiserver. Teilen Sie in praktischen Anwendungen die Verzeichnis- und Dateiservices zwischen verschiedenen Servern auf Leistung und Verfügbarkeit auf.

Sie müssen die Benutzer-ID, unter der der WS-Manager ausgeführt wird, als Mitglied von zwei Gruppen konfigurieren. Es muss ein Mitglied der lokalen *mqm*-Gruppe auf einem IBM WebSphere MQ-Server und der globalen Gruppe von *wmqha* sein.

Wenn der Warteschlangenmanager in dieser Gruppe von Tasks als Service ausgeführt wird, wird er unter der Benutzer-ID *wmquser1* ausgeführt, sodass *wmquser1* ein Mitglied von *wmqha* sein muss. Wenn der Warteschlangenmanager interaktiv ausgeführt wird, wird er unter der Benutzer-ID *wmquser2* ausgeführt, sodass *wmquser2* ein Mitglied von *wmqha* sein muss. Sowohl *wmquser1* als auch *wmquser2* sind Mitglieder der globalen Gruppe *Domain\_mqm*. *Domain\_mqm* ist Mitglied der lokalen *mqm*-Gruppe auf den *mars*- und *venus* IBM WebSphere MQ -Servern. Daher sind *wmquser1* und *wmquser2* Mitglieder der lokalen *mqm*-Gruppe auf beiden IBM WebSphere MQ -Servern.

## **Vorgehensweise**

1. Melden Sie sich am Domänencontroller (*sun.wmq.example.com*) als Domänenadministrator an.
2. Erstellen Sie die globale Gruppe *wmqha*.
  - a) Öffnen Sie **Server Manager > Roles > Active Directory Domain Services > wmq.example.com > Users**.
  - b) Öffnen Sie den Ordner *wmq.example.com\Users*.
  - c) Klicken Sie mit der rechten Maustaste auf **Benutzer > Neu > Gruppe**.
  - d) Geben Sie *wmqha* in das Feld **Gruppenname** ein.

- e) Lassen Sie **Global** als **Gruppenbereich** und **Sicherheit** als **Gruppentyp** geklickt. Klicken Sie auf **OK**.
3. Fügen Sie die Domänenbenutzer *wmquser1* und *wmquser2* zur globalen Gruppe *wmqha* hinzu.
- Klicken Sie in der Navigationsstruktur des Servermanagers auf **Benutzer** und klicken Sie mit der rechten Maustaste auf *wmqha* > **Eigenschaften** in der Liste der Benutzer.
  - Klicken Sie im Fenster *wmqha* auf die Registerkarte Mitglieder.
  - Klicken Sie auf **Hinzufügen ....** *wmquser1*; *wmquser2* eingeben und auf **Namen prüfen** > **OK** > **Anwenden** > **OK** klicken.
4. Erstellen Sie die Verzeichnisstruktur, in der die Daten und Protokolldateien des Warteschlangenmanagers enthalten sind.
- Öffnen Sie eine Eingabeaufforderung.
  - Geben Sie den folgenden Befehl ein:

```
md c:\wmq\data , c:\wmq\logs
```

5. Autorisieren Sie die globale Gruppe *wmqha*, um die vollständige Steuerungsberechtigung für die *c:\wmq*-Verzeichnisse und die gemeinsame Nutzung zu erhalten.
- Klicken Sie in Windows Explorer mit der rechten Maustaste auf *c:\wmq* > **Eigenschaften**.
  - Klicken Sie auf die Registerkarte **Sicherheit** und anschließend auf **Erweitert** > **Bearbeiten ....**
  - Wählen Sie das Kontrollkästchen **Vererbte Berechtigungen für diesen Objekteigner einschließen** ab. Klicken Sie im Fenster 'Windows-Sicherheit' auf **Kopieren**.
  - Wählen Sie die Zeilen für Benutzer in der Liste der **Berechtigungseinträge** aus und klicken Sie auf **Entfernen**. Übernehmen Sie die Zeilen für SYSTEM, Administratoren und CREATOR OWNER in der Liste der **Berechtigungseinträge**.
  - Klicken Sie auf **Hinzufügen ....** und geben Sie den Namen der globalen Gruppe *wmqha* ein. Klicken Sie auf **Namen überprüfen** > **OK**.
  - Wählen Sie im Fenster "Permission Entry for wmq" die Option **Full Control** in der Liste **Permissions** aus.
  - Klicken Sie auf **OK** > **Anwenden** > **OK** > **OK** > **OK**
  - Klicken Sie in Windows Explorer mit der rechten Maustaste auf *c:\wmq* > **Freigeben für**.
  - Klicken Sie auf **Erweiterte gemeinsame Nutzung**. und wählen Sie das Kontrollkästchen **Diesen Ordner freigeben** aus. Belassen Sie den Freiteilnamen als *wmq*.
  - Klicken Sie auf **Berechtigungen** > **Hinzufügen ...**, und geben Sie den Namen der globalen Gruppe *wmqha* ein. Klicken Sie auf **Namen überprüfen** > **OK**.
  - Wählen Sie *wmqha* in der Liste **Gruppen-oder Benutzernamen** aus. Wählen Sie das Kontrollkästchen **Vollzugriff** in der Liste der **Berechtigungen für wmqha** aus und klicken Sie auf **Anwenden**.
  - Wählen Sie *Administrators* in der Liste **Gruppen-oder Benutzernamen** aus. Wählen Sie das Kontrollkästchen **Vollzugriff** in der Liste **Berechtigungen für -Administratoren** aus; klicken Sie auf **Anwenden** > **OK** > **OK** > **Schließen**.

## Nächste Schritte

Überprüfen Sie, ob Sie Dateien von jedem der IBM WebSphere MQ-Server lesen und in die gemeinsam genutzten Verzeichnisse schreiben können. Überprüfen Sie die IBM WebSphere MQ -Servicebenutzer-ID *wmquser1* und die interaktive Benutzer-ID *wmquser2*.

- Wenn Sie den fernen Desktop verwenden, müssen Sie *wmq\wmquser1* und *wmquser2* zur lokalen Gruppe Remote Desktop Users unter *mars* hinzufügen.
  - Melden Sie sich als *wmq\Administrator* bei *mars* an.
  - Führen Sie den Befehl **lusrmgr.msc** aus, um das Fenster "Lokale Benutzer und Gruppen" zu öffnen.

- c. Klicken Sie auf **Gruppen** . Klicken Sie mit der rechten Maustaste auf **Remote Desktop-Benutzer > Eigenschaften > Hinzufügen ....** Geben Sie *wmquser1*; *wmquser2* ein und klicken Sie auf **Namen überprüfen**.
  - d. Geben Sie den Benutzernamen und das Kennwort des Domänenadministrators *wmq\Administratorein* und klicken Sie auf **OK > Anwenden > OK**.
  - e. Schließen Sie das Fenster 'Lokale Benutzer und Gruppen'.
2. Melden Sie sich bei *mars* als *wmq\wmquser1* an.
    - a. Öffnen Sie ein Windows Explorer-Fenster, und geben Sie `\\sun\wmq` ein.  
Das System antwortet durch Öffnen der *wmq* -Freigabe in *sun.wmq.example.com* und listet die Daten auf und protokolliert Verzeichnisse.
    - b. Überprüfen Sie die Berechtigungen von *wmquser1* , indem Sie eine Datei im Datenunterverzeichnis erstellen, einen Inhalt hinzufügen, lesen und anschließend löschen.
  3. Melden Sie sich bei *mars* als *wmq\wmquser2* an, und wiederholen Sie die Prüfungen.
  4. Führen Sie die nächste Task aus, um einen Warteschlangenmanager zu erstellen, der die gemeinsam genutzten Daten- und Protokollverzeichnisse verwendet (siehe „Lesen und Schreiben von gemeinsam genutzten Daten und Protokolldateien, die von einer alternativen globalen Sicherheitsgruppe autorisiert sind“ auf Seite 390).

*Lesen und Schreiben von gemeinsam genutzten Daten und Protokolldateien, die von einer alternativen globalen Sicherheitsgruppe autorisiert sind*

Diese Task veranschaulicht, wie das Flag -a im Befehl **crtmqm** verwendet wird. Das Flag -a gibt dem WS-Manager Zugriff auf seine Protokoll- und Datendateien in einer fernen Dateifreigabe unter Verwendung der alternativen Sicherheitsgruppe.

In einer Produktionsmaßstabskonfiguration müssen Sie die Konfiguration möglicherweise an eine vorhandene Domäne anpassen. Sie können z. B. verschiedene Domänengruppen definieren, um unterschiedliche Freigaben zu berechtigen und die Benutzer-IDs zu gruppieren, die Warteschlangenmanager ausführen.

Die Beispielkonfiguration besteht aus drei Servern:

#### **sun**

Ein Domänencontroller unter Windows Server 2008. Sie ist Eigner der *wmq.example.com* -Domäne, die *Sun*, *mars* und *venus* enthält. Für die Zwecke der Veranschaulichung wird sie auch als Dateiserver verwendet.

#### **mars**

Ein Windows Server 2008, der als erster IBM WebSphere MQ -Server verwendet wird Sie enthält eine Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

#### **venus**

Ein Windows Server 2008, der als zweiter IBM WebSphere MQ -Server verwendet wird Sie enthält die zweite Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

Ersetzen Sie die kursiv dargestellten Namen im Beispiel durch die Namen Ihrer Wahl.

## **Vorbereitende Schritte**

Führen Sie die Schritte in den folgenden Tasks aus. Mit den Tasks wird der Domänencontroller und die Domäne erstellt, IBM WebSphere MQ for Windows auf einem Server installiert und die Dateifreigabe für Daten- und Protokolldateien erstellt. Wenn Sie einen bereits vorhandenen Domänencontroller konfigurieren möchten, ist es unter Umständen sinnvoll, die durchzuführenden Schritte an einem neuen Windows Server 2008 auszuprobieren. Sie können die Schritte an Ihre Domäne anpassen.

1. „Active Directory und DNS-Domäne für IBM WebSphere MQ“ auf Seite 380.
2. „IBM WebSphere MQ auf einem Server oder einer Workstation in einer Windows-Domäne installieren“ auf Seite 384.
3. „Gemeinsam genutztes Verzeichnis für Warteschlangenmanagerdaten und Protokolldateien erstellen“ auf Seite 387.

## Informationen zu diesem Vorgang

Diese Task ist eine von einer Gruppe zusammengehöriger Tasks, die den Zugriff auf Warteschlangenmanagerdaten und Protokolldateien veranschaulichen. Die Tasks zeigen, wie ein Warteschlangenmanager erstellt wird, der berechtigt ist, Daten und Protokolldateien zu lesen und zu schreiben, die in einem Verzeichnis Ihrer Wahl gespeichert sind. Sie begleiten die Task „[Windows-Domänen und Multi-Instanz-Warteschlangenmanager](#)“ auf Seite 376.

In dieser Task erstellen Sie einen Warteschlangenmanager, der seine Daten speichert und sich in einem fernen Verzeichnis auf einem Dateiserver anmeldet. Für die Zwecke dieses Beispiels ist der Dateiserver der gleiche Server wie der Domänencontroller. Das Verzeichnis, das die Daten- und Protokollordner enthält, wird mit der vollständigen Steuerberechtigung für die globale Gruppe `wmqa` freigegeben.

## Vorgehensweise

1. Melden Sie sich am Domänenserver (`mars`) als lokaler Administrator (`mars\Administrator`) an.
2. Öffnen Sie ein Befehlsfenster.
3. Führen Sie einen Neustart für den IBM IBM WebSphere MQ-Service aus.

Sie müssen den Service erneut starten, damit die Benutzer-ID, unter der er ausgeführt wird, die zusätzlichen Sicherheitsberechtigungs nachweise erhält, die Sie für die Benutzer-ID konfiguriert haben.

Geben Sie die Befehle ein:

```
endmqsvc
strmqsvc
```

Die Systemantworten:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
The MQ service for installation 'Installation1' ended successfully.
Und:
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
The MQ service for installation 'Installation1' started successfully.
```

4. Erstellen Sie den Warteschlangenmanager.

```
crtmqm -a wmq\wmqa -sax -u SYSTEM.DEAD.LETTER.QUEUE -md \\sun\wmq\data -ld \\sun\wmq\logs
QMGR
```

Sie müssen die Domäne (`wmq`) der alternativen Sicherheitsgruppe `wmqa` angeben, indem Sie den vollständigen Domännennamen der globalen Gruppe "`wmq\wmqa`" angeben.

Sie müssen den UNC-Namen (UNC = Universal Naming Convention) für die gemeinsame Nutzung von `\\sun\wmq` festlegen und keinen zugeordneten Laufwerkverweis verwenden.

Systemantwort:

```
WebSphere MQ queue manager created.
Directory '\\sun\wmq\data\QMGR' created.
The queue manager is associated with installation '1'
Creating or replacing default objects for queue manager 'QMGR'
Default objects statistics : 74 created. 0 replaced.
Completing setup.
Setup completed.
```

## Nächste Schritte

Testen Sie den Warteschlangenmanager, indem Sie eine Nachricht in eine Warteschlange einreihen und eine Nachricht erhalten.

1. Starten Sie den Warteschlangenmanager.



```
strmqm QMGR
```

Systemantwort:

```
WebSphere MQ queue manager 'QMGR' starting.
The queue manager is associated with installation '1'.
5 log records accessed on queue manager 'QMGR' during the log
replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
WebSphere MQ queue manager 'QMGR' started using V7.1.0.0.
```

- Erstellen Sie eine Testwarteschlange.

```
echo define qlocal(QTEST) | runmqsc QMGR
```

Systemantwort:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QMGR.
```

```
1 : define qlocal(QTEST)
AMQ8006: WebSphere MQ queue created.
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

- Reihen Sie eine Testnachricht mit dem Beispielprogramm **amqspu**tein.

```
echo 'A test message' | amqsput QTEST QMGR
```

Systemantwort:

```
Sample AMQSPUT0 start
target queue is QTEST
Sample AMQSPUT0 end
```

- Rufen Sie die Testnachricht mit dem Beispielprogramm **amqsget**ab.

```
amqsget QTEST QMGR
```

Systemantwort:

```
Sample AMQSGET0 start
message <A test message>
Wait 15 seconds ...
no more messages
Sample AMQSGET0 end
```

- Stoppen Sie den Warteschlangenmanager.

```
endmqm -i QMGR
```

Systemantwort:

```
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ended.
```

- Löschen Sie den Warteschlangenmanager.

```
dltmqm QMGR
```

Systemantwort:



WebSphere MQ queue manager 'QMGR' deleted.

7. Löschen Sie die von Ihnen erstellten Verzeichnisse.

**Tipp:** Fügen Sie die Option /Q zu den Befehlen hinzu, um zu verhindern, dass der Befehl zum Löschen jeder Datei oder jedes Verzeichnisses auffordert.

```
del /F /S C:\wmq*. *
rmdir /S C:\wmq
```

### *Multi-Instanz-Warteschlangenmanager auf Domänencontrollern erstellen*

Im folgenden Beispiel wird gezeigt, wie ein Warteschlangenmanager mit mehreren Instanzen unter Windows auf Domänencontrollern konfiguriert wird. In der Konfiguration werden die verwendeten Konzepte und nicht die Produktionsmaßstab, sondern die Konzepte veranschaulicht. Das Beispiel basiert auf Windows Server 2008. Möglicherweise weichen die Schritte auf anderen Windows Server-Versionen im Einzelnen ab.

Die Konfiguration verwendet das Konzept einer Minidomäne oder "domainlet"; siehe Windows 2000-, Windows Server 2003- und Windows Server 2008-Clusterknoten als Domänencontroller. Informationen zum Hinzufügen von Warteschlangenmanagern mit mehreren Instanzen zu einer vorhandenen Domäne finden Sie unter „Multi-Instanz-Warteschlangenmanager auf Workstations oder Servern in einer Domäne erstellen“ auf Seite 377.

Die Beispielkonfiguration besteht aus drei Servern:

#### **sun**

Ein Server von Windows Server 2008, der als erster Domänencontroller verwendet wird. Sie definiert die *wmq.example.com* -Domäne, die *sun*, *earth* und *mars* enthält. Sie enthält eine Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

#### **earth**

Ein Windows Server 2008, der als zweiter Domänencontroller IBM WebSphere MQ -Server verwendet wird. Sie enthält die zweite Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

#### **mars**

Ein Windows Server 2008, der als Dateiserver verwendet wird.

Ersetzen Sie die kursiv dargestellten Namen im Beispiel durch die Namen Ihrer Wahl.

## **Vorbereitende Schritte**

1. Unter Windows müssen Sie das Dateisystem, in dem Sie Warteschlangenmanagerdaten und Protokoll-dateien speichern möchten, nicht überprüfen. Die Prüfprozedur Verhalten des gemeinsam genutzten Dateisystems überprüfen gilt für UNIX and Linux. Unter Windows sind die Prüfungen immer erfolgreich.
2. Führen Sie die Schritte in „Active Directory und DNS-Domäne für IBM WebSphere MQ“ auf Seite 380 aus, um den ersten Domänencontroller zu erstellen.
3. Führen Sie die Schritte in „Zweiten Domänencontroller zur Domäne wmq.example.com hinzufügen“ auf Seite 396 aus, um einen zweiten Domänencontroller hinzuzufügen, IBM WebSphere MQ für Windows auf beiden Domänencontrollern zu installieren und die Installationen zu überprüfen.
4. Führen Sie die Schritte in „IBM WebSphere MQ auf Domänencontrollern in der Domäne wmq.example.com installieren“ auf Seite 398 aus, um IBM WebSphere MQ auf den beiden Domänencontrollern zu installieren.

## **Informationen zu diesem Vorgang**

Erstellen Sie auf einem Dateiserver in derselben Domäne eine Freigabe für die Protokoll- und Datenverzeichnisse des Warteschlangenmanagers. Erstellen Sie als Nächstes die erste Instanz eines Multi-Instanz-Warteschlangenmanagers, der die Dateifreigabe auf einem der Domänencontroller verwendet. Erstellen Sie die andere Instanz auf dem anderen Domänencontroller, und überprüfen Sie abschließend die Konfiguration. Sie können die Dateifreigabe auf einem Domänencontroller erstellen.

In dem Beispiel ist *sun* der erste Domänencontroller, *earth* der zweite und *mars* der Dateiserver.

## Vorgehensweise

1. Erstellen Sie die Verzeichnisse, die die WS-Manager-Daten und -Protokolldateien enthalten sollen.

a) Geben Sie unter *mars* den folgenden Befehl ein:

```
md c:\wmq\data , c:\wmq\logs
```

2. Geben Sie die Verzeichnisse frei, die die Daten des Warteschlangenmanagers und die Protokolldateien enthalten sollen.

Sie müssen den vollständigen Steuerzugriff auf die lokale Domänengruppe *mqm* und die Benutzer-ID, die Sie zum Erstellen des Warteschlangenmanagers verwenden, zulassen. In dem Beispiel haben Benutzer-IDs, die Mitglieder von Domain Administrators sind, die Berechtigung zum Erstellen von Warteschlangenmanagern.

Die Dateifreigabe muss sich auf einem Server befinden, der sich in derselben Domäne wie die Domänencontroller befindet. In dem Beispiel befindet sich der Server *mars* in derselben Domäne wie die Domänencontroller.

- a) Klicken Sie in Windows Explorer mit der rechten Maustaste auf **c:\wmq > Eigenschaften**.
  - b) Klicken Sie auf die Registerkarte **Sicherheit** und anschließend auf **Erweitert > Bearbeiten ....**
  - c) Wählen Sie das Kontrollkästchen **Vererbte Berechtigungen für diesen Objekteigner einschließen** ab. Klicken Sie im Fenster 'Windows-Sicherheit' auf **Kopieren**.
  - d) Wählen Sie die Zeilen für Benutzer in der Liste der **Berechtigungseinträge** aus und klicken Sie auf **Entfernen**. Übernehmen Sie die Zeilen für SYSTEM, Administratoren und CREATOR OWNER in der Liste der **Berechtigungseinträge**.
  - e) Klicken Sie auf **Hinzufügen ....** und geben Sie den Namen der lokalen Domänengruppe *mqmein*. Klicken Sie auf **Namen überprüfen**.
  - f) Geben Sie als Antwort auf ein Windows-Sicherheitsfenster den Namen und das Passwort für Domain Administrator ein und klicken Sie auf **OK > OK**.
  - g) Wählen Sie im Fenster "Permission Entry for wmq" die Option **Full Control** in der Liste **Permissions** aus.
  - h) Klicken Sie auf **OK > Anwenden > OK > OK > OK**
    - i) Wiederholen Sie die Schritte e bis h, um Domain Administrators hinzuzufügen.
    - j) Klicken Sie in Windows Explorer mit der rechten Maustaste auf **c:\wmq > Freigeben für**.
    - k) Klicken Sie auf **Erweiterte gemeinsame Nutzung**. und wählen Sie das Kontrollkästchen **Diesen Ordner freigeben** aus. Belassen Sie den Freiteilnamen als *wmq*.
    - l) Klicken Sie auf **Berechtigungen > Hinzufügen ...**, und geben Sie den Namen der lokalen Domänengruppe *mqm*; Domain Administrator ein. Klicken Sie auf **Namen überprüfen**.
    - m) Geben Sie als Antwort auf ein Windows-Sicherheitsfenster den Namen und das Passwort für Domain Administrator ein und klicken Sie auf **OK > OK**.
3. Erstellen Sie den WS-Manager *QMGR* auf dem ersten Domänencontroller, *sun*.

```
crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md \\mars\wmq\data -ld \\mars\wmq\logs QMGR
```

Systemantwort:

```
WebSphere MQ queue manager created.
Directory '\\mars\wmq\data\QMGR' created.
The queue manager is associated with installation 'Installation1'.
Creating or replacing default objects for queue manager 'QMGR'.
Default objects statistics : 74 created. 0 replaced. 0 failed.
Completing setup.
Setup completed.
```

4. Starten Sie den Warteschlangenmanager unter *sun*, so dass eine Standby-Instanz möglich ist.

```
strmqm -x QMGR
```

Systemantwort:

```
WebSphere MQ queue manager 'QMGR' starting.
The queue manager is associated with installation 'Installation1'.
5 log records accessed on queue manager 'QMGR' during the log
replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
WebSphere MQ queue manager 'QMGR' started using V7.1.0.0.
```

5. Erstellen Sie eine zweite Instanz von *QMGR* auf *earth*.

- a) Überprüfen Sie die Werte der Parameter *Prefix* und *InstallationName* für *earth*.

Führen Sie unter *sun* den Befehl **dspmqlinf** aus:

```
dspmqlinf QMGR
```

Systemantwort:

```
QueueManager:
 Name=QMGR
 Directory=QMGR
 Prefix=C:\Program Files\IBM\WebSphere MQ
 DataPath=\\mars\wmq\data\QMGR
 InstallationName=Installation1
```

- b) Kopieren Sie die maschinenlesbare Form der Zeilengruppe **QueueManager** in die Zwischenablage.

Führen Sie unter *sun* den Befehl **dspmqlinf** erneut mit dem Parameter *-o command* aus.

```
dspmqlinf -o command QMGR
```

Systemantwort:

```
addmqinf -s QueueManager -v Name=QMGR
-v Directory=QMGR -v Prefix="C:\Program Files\IBM\WebSphere MQ"
-v DataPath=\\mars\wmq\data\QMGR
```

- c) Führen Sie unter *earth* den Befehl **addmqinf** aus der Zwischenablage heraus aus, um eine Instanz des Warteschlangenmanagers unter *earth* zu erstellen.

Passen Sie den Befehl bei Bedarf an, um die Unterschiede in den Parametern *Präfix* oder *InstallationName* zu berücksichtigen.

```
addmqinf -s QueueManager -v Name=QMGR
-v Directory=QMGR -v Prefix="C:\Program Files\IBM\WebSphere MQ"
-v DataPath=\\mars\wmq\data\QMGR
```

WebSphere MQ configuration information added.

6. Starten Sie die Standby-Instanz des Warteschlangenmanagers auf *earth*.

```
strmqm -x QMGR
```

Systemantwort:

```
WebSphere MQ queue manager 'QMGR' starting.
The queue manager is associated with installation 'Installation1'.
A standby instance of queue manager 'QMGR' has been started. The active
instance is running elsewhere.
```

## Ergebnisse

Stellen Sie sicher, dass der WS-Manager von *sun* in *earth* umschaltet:

1. Führen Sie unter *sun* den folgenden Befehl aus:

```
endmqm -i -r -s QMGR
```

Systemantwort auf *sun*:

```
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ended, permitting switchover to
a standby instance.
```

2. Geben Sie unter *earth* den folgenden Befehl wiederholt ein:

```
dspmq
```

Die Systemantworten:

```
QMNAME(QMGR) STATUS(Running as standby)
QMNAME(QMGR) STATUS(Running as standby)
QMNAME(QMGR) STATUS(Running)
```

## Nächste Schritte

Ein Mehrinstanz-WS-Manager kann auch mit Musterprogrammen überprüft werden. Informationen hierzu finden Sie im Abschnitt [„Warteschlangenmanager mit mehreren Instanzen unter Windows überprüfen“](#) auf Seite 400.

### Zugehörige Tasks

[„Zweiten Domänencontroller zur Domäne \*wmq.example.com\* hinzufügen“](#) auf Seite 396

[„IBM WebSphere MQ auf Domänencontrollern in der Domäne \*wmq.example.com\* installieren“](#) auf Seite 398

### Zugehörige Informationen

[Clusterknoten mit Windows 2000, Windows Server 2003 und Windows Server 2008 als Domänencontroller](#)

*Zweiten Domänencontroller zur Domäne *wmq.example.com* hinzufügen*

Fügen Sie der *wmq.example.com*-Domäne einen zweiten Domänencontroller hinzu, um eine Windows-Domäne zu erstellen, in der WS-Manager mit mehreren Instanzen auf Domänencontrollern und Dateiservern ausgeführt werden.

Die Beispielkonfiguration besteht aus drei Servern:

#### ***sun***

Ein Server von Windows Server 2008, der als erster Domänencontroller verwendet wird. Sie definiert die *wmq.example.com*-Domäne, die *sun*, *earth* und *mars* enthält. Sie enthält eine Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

#### ***earth***

Ein Windows Server 2008, der als zweiter Domänencontroller IBM WebSphere MQ -Server verwendet wird. Sie enthält die zweite Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

#### ***mars***

Ein Windows Server 2008, der als Dateiserver verwendet wird.

Ersetzen Sie die kursiv dargestellten Namen im Beispiel durch die Namen Ihrer Wahl.

## Vorbereitende Schritte

1. Führen Sie die im Abschnitt „Active Directory und DNS-Domäne für IBM WebSphere MQ“ auf Seite 380 beschriebenen Schritte zum Erstellen eines Domänencontrollers *sun* für die Domäne *wmq.example.com* aus. Ändern Sie die in Kursivdruck angezeigten Namen Ihrer Konfiguration entsprechend.
2. Installieren Sie Windows Server 2008 auf einem Server in der Standardworkgroup (WORKGROUP). Für das Beispiel hat der Server den Namen *earth*.

## Informationen zu diesem Vorgang

In dieser Task konfigurieren Sie einen Windows Server 2008 mit dem Namen *earth* als zweiter Domänencontroller in der *wmq.example.com*-Domäne.

Diese Task ist eine von einer Gruppe zusammengehöriger Tasks, die den Zugriff auf Warteschlangenmanagerdaten und Protokolldateien veranschaulichen. Die Tasks zeigen, wie ein Warteschlangenmanager erstellt wird, der berechtigt ist, Daten und Protokolldateien zu lesen und zu schreiben, die in einem Verzeichnis Ihrer Wahl gespeichert sind. Sie begleiten die Task „Windows-Domänen und Multi-Instanz-Warteschlangenmanager“ auf Seite 376.

## Vorgehensweise

1. Fügen Sie den Domänencontroller *sun.wmq.example.com* zu *earth* als DNS-Server hinzu.
  - a) Melden Sie sich unter *earth* als *earth\Administrator* an und klicken Sie auf **Starten** .
  - b) Klicken Sie mit der rechten Maustaste auf **Netz > Eigenschaften > Netzverbindungen verwalten** .
  - c) Klicken Sie auf den Netzadapter, und klicken Sie auf **Eigenschaften** .

Das System antwortet mit dem Fenster "Merkmale des lokalen Bereichs", in dem die von der Verbindung verwendeten Elemente aufgelistet werden.
  - d) Wählen Sie aus der Liste der Einträge im Fenster 'Local Area Connection Properties' **Internet Protocol Version 4** oder **Internet Protocol Version 6** aus. Klicken Sie auf **Eigenschaften > Erweitert ...** und klicken Sie auf die Registerkarte **DNS** .
  - e) Klicken Sie unter den DNS-Serveradressen auf **Hinzufügen ...** .
  - f) Geben Sie die IP-Adresse des Domänencontrollers ein, bei dem es sich auch um den DNS-Server handelt, und klicken Sie auf **Hinzufügen** .
  - g) Klicken Sie auf **Diese DNS-Suffixe anhängen > Hinzufügen ...** .
  - h) Geben Sie *wmq.example.com* ein und klicken Sie auf **Hinzufügen**
  - i) Geben Sie *wmq.example.com* in das Feld **DNS-Suffix für diese Verbindung** ein.
  - j) Wählen Sie die Option **Diese Verbindungsadresse in DNS registrieren** und **Das Suffix dieser Verbindung in der DNS-Registrierung verwenden** aus. Klicken Sie auf **OK > OK > Schließen** .
  - k) Öffnen Sie ein Befehlsfenster und geben Sie den Befehl **ipconfig /all** ein, um die TCP/IP-Einstellungen zu überprüfen.
2. Melden Sie sich als lokaler Administrator oder Workgroup-Administrator am Domänencontroller (*sun*) an.

Wenn der Server bereits als Domänencontroller konfiguriert ist, müssen Sie sich als Domänenadministrator anmelden.
3. Führen Sie den Assistenten für die Active Directory-Domänenservices aus.
  - a) Klicken Sie auf **Start > Ausführen ...** Geben Sie `dcprmo` ein und klicken Sie auf **OK** .

Wenn die Active Directory-Binärdateien noch nicht installiert sind, installiert Windows die Dateien automatisch.
4. Konfigurieren Sie *earth* als zweiten Domänencontroller in der *wmq.example.com*-Domäne.

- a) Lassen Sie im ersten Fenster des Assistenten das Kontrollkästchen **Erweiterte Modusinstallation verwenden** leer. Klicken Sie auf **Next > Next** und klicken Sie auf **Create Add a domain controller to an existing domain > Next**.
- b) Geben Sie *wmq* in das Feld **Geben Sie den Namen einer beliebigen Domäne in dieser Gesamtstruktur ein ...** ein Feld angegeben haben. Klicken Sie auf das Optionsfeld **Alternative Berechtigungsnachweise**, und klicken Sie auf **Festlegen ...**. Geben Sie den Namen und das Kennwort des Domänenadministrators ein und klicken Sie auf **OK > Weiter > Weiter > Weiter**.
- c) Akzeptieren Sie im Fenster 'Additional Domain Controller Options' die ausgewählten Optionen **DNS-Server** und **Globaler Katalog**. Klicken Sie auf **Weiter > Weiter**.
- d) Geben Sie im Administratorkennwort für den Wiederherstellungsmodus der Verzeichnisservices **Kennwort** und **Kennwort bestätigen** ein und klicken Sie auf **Weiter > Weiter**.
- e) Geben Sie das Kennwort des Domänenadministrators ein, wenn Sie dazu aufgefordert werden, **Network Credentials** einzugeben. Wählen Sie im letzten Fenster des Assistenten die Option **Reboot on completion** (Warmstart bei Beendigung) aus.
- f) Nach einer Weile wird möglicherweise ein Fenster mit einem **DCPromo** -Fehler bezüglich der DNS-Delegation geöffnet. Klicken Sie auf **OK**. Der Server wird neu gestartet.

## Ergebnisse

Wenn *earth* einen Warmstart durchgeführt hat, melden Sie sich als Domänenadministrator an. Überprüfen Sie, ob die *wmq.example.com* -Domäne in *earth* repliziert wurde.

## Nächste Schritte

Fahren Sie mit der Installation von IBM WebSphere MQ fort; siehe [„IBM WebSphere MQ auf Domänencontrollern in der Domäne wmq.example.com installieren“](#) auf Seite 398.

## Zugehörige Tasks

[„Active Directory und DNS-Domäne für IBM WebSphere MQ“](#) auf Seite 380

*IBM WebSphere MQ auf Domänencontrollern in der Domäne wmq.example.com installieren*

Installieren und konfigurieren Sie Installationen von IBM WebSphere MQ auf beiden Domänencontrollern in der Domäne *wmq.example.com*.

Geben Sie hier Ihre Kurzbeschreibung an; sie wird für den ersten Absatz verwendet und ist abstrakt.

Die Beispielkonfiguration besteht aus drei Servern:

### **sun**

Ein Server von Windows Server 2008, der als erster Domänencontroller verwendet wird. Sie definiert die *wmq.example.com* -Domäne, die *sun*, *earth* und *mars* enthält. Sie enthält eine Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

### **earth**

Ein Windows Server 2008, der als zweiter Domänencontroller IBM WebSphere MQ -Server verwendet wird. Sie enthält die zweite Instanz des Multi-Instanz-WS-Managers mit dem Namen *QMGR*.

### **mars**

Ein Windows Server 2008, der als Dateiserver verwendet wird.

Ersetzen Sie die kursiv dargestellten Namen im Beispiel durch die Namen Ihrer Wahl.

## Vorbereitende Schritte

1. Führen Sie die im Abschnitt [„Active Directory und DNS-Domäne für IBM WebSphere MQ“](#) auf Seite 380 beschriebenen Schritte zum Erstellen eines Domänencontrollers *sun* für die Domäne *wmq.example.com* aus. Ändern Sie die in Kursivdruck angezeigten Namen Ihrer Konfiguration entsprechend.
2. Führen Sie die Schritte im Abschnitt [„Zweiten Domänencontroller zur Domäne wmq.example.com hinzufügen“](#) auf Seite 396 aus, um einen zweiten Domänencontroller, *earth*, für die Domäne *wmq.example.com* aus.

*ple.com* zu erstellen. Ändern Sie die in Kursivdruck angezeigten Namen Ihrer Konfiguration entsprechend.

3. Im Abschnitt Hardware- und Softwarevoraussetzungen auf Windows-Systemen finden Sie Informationen zu weiteren Windows-Versionen, auf denen IBM WebSphere MQ ausgeführt werden kann.

## Informationen zu diesem Vorgang

Installieren und konfigurieren Sie Installationen von IBM WebSphere MQ auf beiden Domänencontrollern in der Domäne *umq.example.com*.

## Vorgehensweise

1. Installieren Sie IBM WebSphere MQ auf *sun* und *earth*.

Weitere Informationen zum Ausführen des Installationsassistenten von IBM WebSphere MQ for Windows finden Sie im Abschnitt IBM WebSphere MQ-Server unter Windows installieren.

- a) Melden Sie sich sowohl auf *sun* als auch auf *earth* als Domänenadministrator an, *umq\Administrator*.
- b) Führen Sie den Befehl **Setup** auf den IBM WebSphere MQ for Windows -Installationsmedien aus. Die Launchpad-Anwendung von IBM WebSphere MQ wird gestartet.
- c) Klicken Sie auf **Softwarevoraussetzungen**, um zu überprüfen, ob die vorausgesetzte Software installiert ist.
- d) Klicken Sie auf **Netzkonfiguration > Nein**.  
Sie können entweder eine Domänenbenutzer-ID oder nicht für diese Installation konfigurieren. Die Benutzer-ID, die erstellt wird, ist eine lokale Benutzer-ID.
- e) Klicken Sie auf **WebSphere MQ Installation**, wählen Sie eine Sprache für die Installation aus und klicken Sie auf 'Launch IBM WebSphere MQ Installer' (Installation von IBM WebSphere MQ starten).
- f) Bestätigen Sie die Lizenzvereinbarung und klicken Sie auf **Next > Next > Install**, um die Standardkonfiguration zu akzeptieren. Warten Sie, bis die Installation abgeschlossen ist, und klicken Sie auf **Fertig stellen**.

Wenn Sie den Namen der Installation ändern möchten, verschiedene Komponenten installieren, ein anderes Verzeichnis für WS-Manager-Daten und -Protokolle konfigurieren oder in einem anderen Verzeichnis installieren möchten, klicken Sie auf **Angepasst** und nicht auf **Typisch**.

IBM WebSphere MQ wird installiert und das Installationsprogramm startet den Assistenten "Prepare IBM WebSphere MQ" (WebSphere MQ vorbereiten).

Bei der IBM WebSphere MQ for Windows-Installation werden eine lokale Gruppe *mqm* und eine Domänengruppe *Domain mqm* konfiguriert. Sie macht *Domain mqm* zu einem Mitglied von *mqm*. Nachfolgende Domänencontroller in derselben Domäne verwenden die Gruppen *mqm* und *Domain mqm*.

2. Führen Sie sowohl auf *earth* als auch auf *sun* den "Assistenten für die Vorbereitung von IBM WebSphere MQ" aus.

Weitere Informationen zur Ausführung des Assistenten "Prepare IBM WebSphere MQ" finden Sie unter Configuring WebSphere MQ with the Prepare WebSphere MQ wizard.

- a) Das IBM WebSphere MQ -Installationsprogramm führt "Prepare IBM WebSphere MQ" automatisch aus.  
Um den Assistenten manuell zu starten, suchen Sie die Verknüpfung zu "Prepare IBM WebSphere MQ" im Ordner **Start > Alle Programme > IBM WebSphere MQ**. Wählen Sie in einer Konfiguration mit mehreren Installationen den entsprechenden Direktaufruf für die IBM WebSphere MQ-Installation aus.
- b) Klicken Sie auf **Weiter** und lassen Sie **Nein** als Antwort auf die Frage "Identifizieren, ob es einen Domänencontroller von Windows 2000 oder höher im Netz gibt" ausgewählt.<sup>1</sup>



c) Wählen Sie auf der letzten Seite des Assistenten die Markierungsfelder aus, die Sie benötigen, und klicken Sie auf **Fertig stellen** .

Der "Assistent für die Vorbereitung von IBM WebSphere MQ" erstellt einen lokalen Domänenbenutzer MUSR\_MQADMIN auf dem ersten Domänencontroller und einen weiteren lokalen Domänenbenutzer MUSR\_MQADMIN1 auf dem zweiten Domänencontroller. Der Assistent erstellt den IBM WebSphere MQ -Service auf jedem Controller, wobei MUSR\_MQADMIN oder MUSR\_MQADMIN1 der Benutzer ist, der den Service anmeldet.

3. Definieren Sie einen Benutzer, der über die Berechtigung zum Erstellen eines Warteschlangenmanagers verfügt.

Der Benutzer muss das Recht haben, sich lokal anzumelden und ein Mitglied der lokalen mqm-Gruppe zu sein. Domänenbenutzer haben auf Domänencontrollern nicht das Recht, sich lokal anzumelden, aber Administratoren tun dies. Standardmäßig verfügt kein Benutzer über beide Attribute. Fügen Sie in dieser Task Domänenadministratoren zur lokalen mqm-Gruppe hinzu.

- a) Öffnen Sie **Server Manager > Roles > Active Directory Domain Services > wmq.example.com > Users**.
- b) Klicken Sie mit der rechten Maustaste auf **Domänenadministratoren > Zu Gruppe hinzufügen ...** Geben Sie mqmein. Klicken Sie auf **Namen prüfen > OK > OK** .

## Ergebnisse

1. Vergewissern Sie sich, dass der "Assistent für die Vorbereitung von IBM WebSphere MQ " den Domänenbenutzer MUSR\_MQADMIN erstellt hat:
  - a. Öffnen Sie **Server Manager > Roles > Active Directory Domain Services > wmq.example.com > Users**.
  - b. Klicken Sie mit der rechten Maustaste auf **MUSR\_MQADMIN > Eigenschaften ... > Mitglied von** und sehen, dass es ein Mitglied von `Domain users` und `mqmist` ist.
2. Stellen Sie sicher, dass MUSR\_MQADMIN das Recht hat, als Dienst ausgeführt zu werden:
  - a. Klicken Sie auf **Starten > Ausführen**. Geben Sie den Befehl `secpol.msc` ein und klicken Sie auf **OK**.
  - b. Öffnen Sie **Sicherheitseinstellungen > Lokale Richtlinien > Zuweisen von Benutzerrechten**. Klicken Sie in der Liste der Richtlinien mit der rechten Maustaste auf **Log on as a service > Properties** und stellen Sie fest, ob MUSR\_MQADMIN mit dem Recht, sich als Service anzumelden, aufgeführt ist. Klicken Sie auf **OK**.

## Nächste Schritte

1. Führen Sie die Task „Lesen und Schreiben von Daten und Protokolldateien, die von der lokalen mqm-Gruppe autorisiert sind“ auf Seite 408 aus, um zu prüfen, ob die Installation und Konfiguration ordnungsgemäß funktionieren.
2. Kehren Sie zur Task „Multi-Instanz-Warteschlangenmanager auf Domänencontrollern erstellen“ auf Seite 393 zurück, um die Task zur Konfiguration eines Multi-Instanz-Warteschlangenmanagers auf Domänencontrollern auszuführen.

## Zugehörige Konzepte

Erforderliche Benutzerberechtigungen für einen WebSphere MQ-Windows-Dienst

*Warteschlangenmanager mit mehreren Instanzen unter Windows überprüfen*

Verwenden Sie die Beispielprogramme **amqsgshac**, **amqsphac** und **amqsmhac** , um die Konfiguration eines Multi-Instanz-Warteschlangenmanagers zu prüfen. Dieser Abschnitt enthält eine Beispielkonfiguration für

---

<sup>1</sup> Sie können die Installation für die Domäne konfigurieren. Da alle Benutzer und Gruppen auf einem Domänencontroller einen Domänenbereich haben, macht dies keinen Unterschied. Es ist einfacher, IBM WebSphere MQ so zu installieren, als ob es nicht in der Domäne vorhanden ist.



die Überprüfung der Konfiguration eines Multi-Instanz-Warteschlangenmanagers unter Windows Server 2003.

Die Beispielprogramme für hohe Verfügbarkeit verwenden die automatische Clientwiederverbindung. Wenn der verbundene Warteschlangenmanager ausfällt, versucht der Client, die Verbindung zu einem WS-Manager in derselben Warteschlangenmanager-Gruppe wieder herzustellen. Die Beschreibung der Beispiele [Hochverfügbarkeits-Musterprogramme](#) veranschaulicht die Clientwiederverbindung unter Verwendung eines einzigen Instanz-WS-Managers für die Einfachheit. Sie können dieselben Muster mit Warteschlangenmanagern mit mehreren Instanzen verwenden, um eine Konfiguration mit mehreren Instanzen des Warteschlangenmanagers zu überprüfen.

In diesem Beispiel wird die in „[Multi-Instanz-Warteschlangenmanager auf Domänencontrollern erstellen](#)“ auf Seite 393 beschriebene Konfiguration mit mehreren Instanzen verwendet. Überprüfen Sie mit der Konfiguration, ob der Warteschlangenmanager mit mehreren Instanzen in die Standby-Instanz umschaltet. Stoppen Sie den WS-Manager mit dem Befehl **endmqm** und verwenden Sie die Option **-s**, Switchover. Die Clientprogramme stellen die Verbindung zur neuen Warteschlangenmanagerinstanz wieder her und arbeiten nach einer geringfügigen Verzögerung weiterhin mit der neuen Instanz.

Der Client wird in einem VMware -Image mit 400 MB installiert, auf dem Windows XP Service-Pack 2 ausgeführt wird. Aus Sicherheitsgründen ist sie auf demselben nur-Host-Netz von VMWare wie die Domänenserver, auf denen der Multi-Instanz-Warteschlangenmanager ausgeführt wird, verbunden. Der Ordner /MQHA , der die Clientverbindungstabelle enthält, wird gemeinsam genutzt, um die Konfiguration zu vereinfachen.

### **Funktionsübernahme mit WebSphere MQ Explorer überprüfen**

Bevor Sie die Beispielanwendungen verwenden, um die Funktionsübernahme zu überprüfen, führen Sie den WebSphere MQ Explorer auf jedem Server aus. Fügen Sie jedem Explorer die beiden WS-Manager-Instanzen hinzu, indem Sie den Assistenten **Remote Queue Manager hinzufügen > Direkt mit einem Multi-Instanz-Warteschlangenmanager verbinden** verwenden. Stellen Sie sicher, dass beide Instanzen aktiv sind und den Standby-Modus zulassen. Schließen Sie das Fenster, in dem das VMware-Image mit der aktiven Instanz ausgeführt wird, schließen Sie den Server virtuell ab, oder stoppen Sie die aktive Instanz, indem Sie die Umschaltung auf die Standby-Instanz zulassen und Clients erneut verbinden können, um die Verbindung herzustellen.

**Anmerkung:** Wenn Sie den Server ausschalten, stellen Sie sicher, dass es sich nicht um den Host des MQHA -Ordners handelt!

**Anmerkung:** Die Option **Umschalten auf eine Standby-Instanz zulassen** ist unter Umständen nicht im Dialog **Warteschlangenmanager stoppen** verfügbar. Die Option fehlt, weil der WS-Manager als einzelner Instanz-Warteschlangenmanager ausgeführt wird. Sie müssen diese Option ohne die Option **Standby-Instanz zulassen** gestartet haben. Wenn Ihre Anforderung zum Stoppen des Warteschlangenmanagers zurückgewiesen wird, sehen Sie sich das Fenster **Details** an. Möglicherweise ist keine Standby-Instanz aktiv.

### **Funktionsübernahme mit den Musterprogrammen überprüfen**

#### **Wählen Sie einen Server zum Ausführen der aktiven Instanz aus.**

Sie haben möglicherweise einen der Server ausgewählt, um das MQHA -Verzeichnis oder das Dateisystem zu hosten. Wenn Sie die Funktionsübernahme testen möchten, indem Sie das Fenster VMware schließen, in dem der aktive Server ausgeführt wird, stellen Sie sicher, dass es sich nicht um das Hosting von MQHA handelt!

#### **Auf dem Server, auf dem die aktive WS-Manager-Instanz ausgeführt**

1. Ändern Sie *ipaddr1* und *ipaddr2* und speichern Sie die folgenden Befehle in `N:\hasample.tst..`

```
DEFINE QLOCAL(SOURCE) REPLACE
DEFINE QLOCAL(TARGET) REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
MCAUSER(' ') REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNAME('ipaddr1(1414),ipaddr2(1414)') QMNAME(QM1) REPLACE
START CHANNEL(CHANNEL1)
```

```
DEFINE LISTENER(LISTENER.TCP) TRPTYPE(TCP) CONTROL(QMGR)
DISPLAY LISTENER(LISTENER.TCP) CONTROL
DISPLAY LSSTATUS(LISTENER.TCP) STATUS
```

**Anmerkung:** Wenn Sie den Parameter **MCAUSER** leer lassen, wird die Clientbenutzer-ID an den Server gesendet. Die Clientbenutzer-ID muss über die korrekten Berechtigungen für die Server verfügen. Alternativ können Sie den Parameter **MCAUSER** im Kanal SVRCONN auf die Benutzer-ID setzen, die Sie auf dem Server konfiguriert haben.

2. Öffnen Sie eine Eingabeaufforderung mit dem Pfad N:\ und führen Sie den folgenden Befehl aus:

```
runmqsc -m QM1 < hasample.tst
```

3. Stellen Sie sicher, dass der Listener aktiv ist und die Steuerung des Warteschlangenmanagers hat, indem Sie die Ausgabe des Befehls **runmqsc** überprüfen.

```
LISTENER(LISTENER.TCP)CONTROL(QMGR)
LISTENER(LISTENER.TCP)STATUS(RUNNING)
```

Oder verwenden Sie den WebSphere MQ Explorer, um zu erkennen, dass das TCP/IP-Empfangsprogramm aktiv ist und über Control = Queue Manager verfügt.

### Auf dem Client

1. Ordnen Sie das gemeinsam genutzte Verzeichnis C:\MQHA auf dem Server N:\ auf dem Client zu.
2. Öffnen Sie eine Eingabeaufforderung mit dem Pfad N:\ . Setzen Sie die Umgebungsvariable MQCHLLIB so, dass sie auf die Clientkanaldefinitionstabelle (CCDT) auf dem Server verweist:

```
SET MQCHLLIB=N:\data\QM1\@ipcc
```

3. Geben Sie an der Eingabeaufforderung die folgenden Befehle ein:

```
start amqsgbac TARGET QM1
start amqsmbac -s SOURCE -t TARGET -m QM1
start amqsphac SOURCE QM1
```

**Anmerkung:** Wenn Sie Probleme haben, starten Sie die Anwendungen in einer Eingabeaufforderung, damit der Ursachencode auf der Konsole ausgegeben wird, oder schauen Sie sich die Datei AMQERR01.LOG im Ordner N:\data\QM1\errors an.

### Auf dem Server, auf dem die aktive WS-Manager-Instanz ausgeführt

1. Entweder:
  - Schließen Sie das Fenster, in dem das VMware-Image mit der aktiven Serverinstanz ausgeführt wird.
  - Stoppen Sie mithilfe von WebSphere MQ Explorer die aktive Warteschlangenmanagerinstanz, ermöglichen Sie das Umschalten auf die Standby-Instanz und weisen Sie wiederverbindbare Clients an, die Verbindung wiederherzustellen.
2. Die drei Clients erkennen schließlich, dass die Verbindung unterbrochen ist, und stellen Sie dann die Verbindung wieder her. Wenn Sie in dieser Konfiguration das Serverfenster schließen, dauert es ungefähr sieben Minuten, bis alle drei Verbindungen wiederhergestellt sind. Einige Verbindungen werden vor anderen wieder hergestellt.

## Ergebnisse

```
N:\>amqspshac SOURCE QM1
Sample AMQSPHAC start
target queue is SOURCE
message <Message 1>
message <Message 2>
message <Message 3>
message <Message 4>
message <Message 5>
17:05:25 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message <Message 6>
message <Message 7>
message <Message 8>
message <Message 9>
```

```
N:\>amqsmhac -s SOURCE -t TARGET -m QM1
Sample AMQSMHA0 start

17:05:25 : EVENT : Connection Reconnecting (Delay: 97ms)
17:05:48 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:53 : EVENT : Connection Reconnected
```

```
N:\>amqsgshac TARGET QM1
Sample AMQSGHAC start
message <Message 1>
message <Message 2>
message <Message 3>
message <Message 4>
message <Message 5>
17:05:25 : EVENT : Connection Reconnecting (Delay: 156ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message <Message 6>
message <Message 7>
message <Message 8>
message <Message 9>
```

### *Gemeinsam genutzte Warteschlangenmanagerdaten, Protokollverzeichnisse und Dateien unter Windows sichern*

In diesem Thema wird beschrieben, wie Sie eine gemeinsam genutzte Position für WS-Manager-Daten und -Protokolldateien mit einer globalen alternativen Sicherheitsgruppe sichern können. Sie können die Position zwischen verschiedenen Instanzen eines Warteschlangenmanagers, der auf verschiedenen Servern ausgeführt wird, gemeinsam nutzen.

In der Regel legen Sie keine gemeinsam genutzte Position für WS-Manager-Daten und Protokolldateien fest. Wenn Sie IBM WebSphere MQ for Windows installieren, erstellt das Installationsprogramm ein Ausgangsverzeichnis Ihrer Wahl für alle WS-Manager, die auf diesem Server erstellt werden. Es sichert die Verzeichnisse mit der lokalen Gruppe mqm und konfiguriert eine Benutzer-ID für den IBM WebSphere MQ -Service, um auf die Verzeichnisse zuzugreifen.

Wenn Sie einen gemeinsam genutzten Ordner mit einer Sicherheitsgruppe sichern, muss ein Benutzer, der für den Zugriff auf den Ordner berechtigt ist, über die Berechtigungsnachweise der Gruppe verfügen. Angenommen, ein Ordner auf einem fernen Dateiserver ist mit der lokalen mqm -Gruppe auf einem Server mit dem Namen *mars* gesichert. Stellen Sie sicher, dass der Benutzer, der Warteschlangenmanager ausführt, ein Mitglied der lokalen mqm -Gruppe in *mars* verarbeitet. Der Benutzer verfügt über die Berechtigungsnachweise, die mit den Berechtigungsnachweisen des Ordners auf dem fernen Dateiserver übereinstimmen. Mit diesen Berechtigungsnachweisen ist der Warteschlangenmanager in der Lage, auf seine Daten zuzugreifen und die Dateien im Ordner zu Protokolldateien zu verwenden. Der Benutzer, der WS-Manager-Prozesse auf einem anderen Server ausführt, ist Mitglied einer anderen lokalen mqm -Gruppe, die keine übereinstimmenden Berechtigungsnachweise hat. Wenn der WS-Manager auf einem anderen Server als *mars* ausgeführt wird, kann er nicht auf die Daten und Protokolldateien zugreifen, die er bei der Ausführung unter *mars* erstellt hat. Selbst wenn Sie den Benutzer zu einem Domänenbenutzer machen, verfügt er über unterschiedliche Berechtigungsnachweise, da er die Berechtigungsnachweise

aus der lokalen mqm -Gruppe auf *mars* erwerben muss und dies nicht von einem anderen Server aus ausgeführt werden kann.

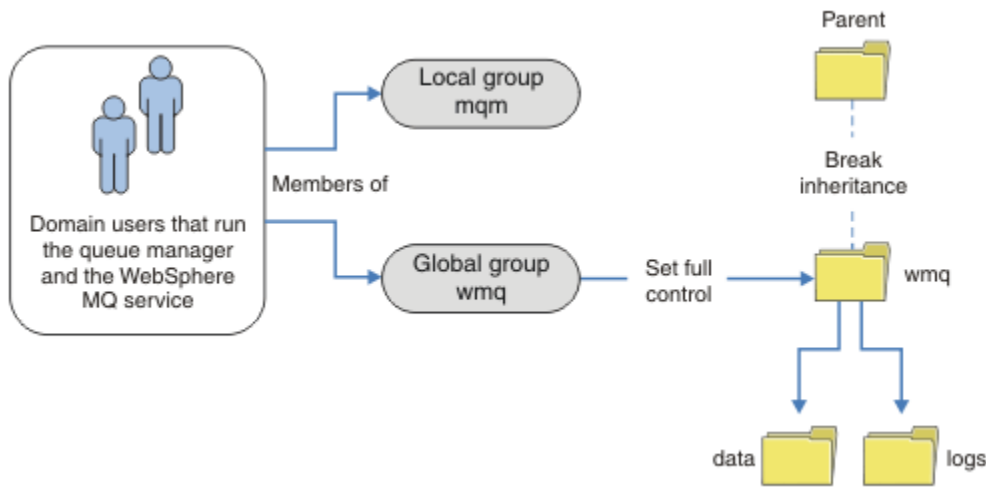
Die Bereitstellung des Warteschlangenmanagers mit einer globalen alternativen Sicherheitsgruppe löst das Problem (siehe [Abbildung 64](#) auf Seite 404). Sichern Sie einen fernen Ordner mit einer globalen Gruppe. Übergeben Sie den Namen der globalen Gruppe an den Warteschlangenmanager, wenn Sie ihn unter *mars* erstellen. Übergeben Sie den globalen Gruppennamen als alternative Sicherheitsgruppe mit dem Parameter `-a[r]` im Befehl `crtmqm`. Wenn Sie den Warteschlangenmanager zur Ausführung auf einem anderen Server übertragen, wird der Name der Sicherheitsgruppe mit dem Namen der Sicherheitsgruppe übertragen. Der Name wird in der Zeilengruppe **AccessMode** in der Datei `qm.ini` als Security-Groupübertragen. Beispiel:

```
AccessMode:
SecurityGroup=wmq\wmq
```

Die Zeilengruppe **AccessMode** in der Datei `qm.ini` enthält auch die Datei `RemoveMQMAccess`. Beispiel:

```
AccessMode:
RemoveMQMAccess=<true>false>
```

Wenn dieses Attribut mit dem Wert `true` angegeben wird und auch eine Zugriffsgruppe angegeben wurde, erhält die lokale Gruppe 'mqm' keinen Zugriff auf die Datendateien des Warteschlangenmanagers.



*Abbildung 64. Sichern von Warteschlangenmanagerdaten und -protokollen mithilfe einer alternativen globalen Sicherheitsgruppe (1)*

Für die Benutzer-ID, mit der WS-Manager-Prozesse ausgeführt werden sollen, um die übereinstimmenden Berechtigungsnachweise der globalen Sicherheitsgruppe zu verwenden, muss die Benutzer-ID auch über einen globalen Geltungsbereich verfügen. Sie können keine lokale Gruppe oder Principal als Mitglied einer globalen Gruppe erstellen. In [Abbildung 64](#) auf Seite 404 werden die Benutzer, die die WS-Manager-Prozesse ausführen, als Domänenbenutzer angezeigt.

Wenn Sie viele IBM WebSphere MQ-Server implementieren, ist die Gruppierung von Benutzern in [Abbildung 64](#) auf Seite 404 nicht praktisch. Sie müssen den Prozess zum Hinzufügen von Benutzern zu lokalen Gruppen für jeden IBM WebSphere MQ -Server wiederholen. Erstellen Sie stattdessen eine globale Domain `mqm`-Gruppe auf dem Domänencontroller und machen Sie die Benutzer, die IBM WebSphere MQ ausführen, zu Mitgliedern der Gruppe Domain `mqm` (siehe [Abbildung 65](#) auf Seite 405). Wenn Sie IBM WebSphere MQ als Domäneninstallation installieren, konfiguriert der Assistent "IBM WebSphere MQ vorbereiten" die Gruppe Domain `mqm` automatisch als Mitglied der lokalen Gruppe `mqm`. Dieselben Benutzer sind sowohl in den globalen Gruppen Domain `mqm` als auch in `wmq` enthalten.

**Tipp:** Dieselben Benutzer können IBM WebSphere MQ auf verschiedenen Servern ausführen, aber auf einem einzelnen Server müssen Sie verschiedene Benutzer haben, um IBM WebSphere MQ als Dienst ausführen zu können, und interaktiv ausgeführt werden. Sie müssen auch für jede Installation auf einem Server unterschiedliche Benutzer haben. In der Regel enthält Domain mqm daher eine Reihe von Benutzern.

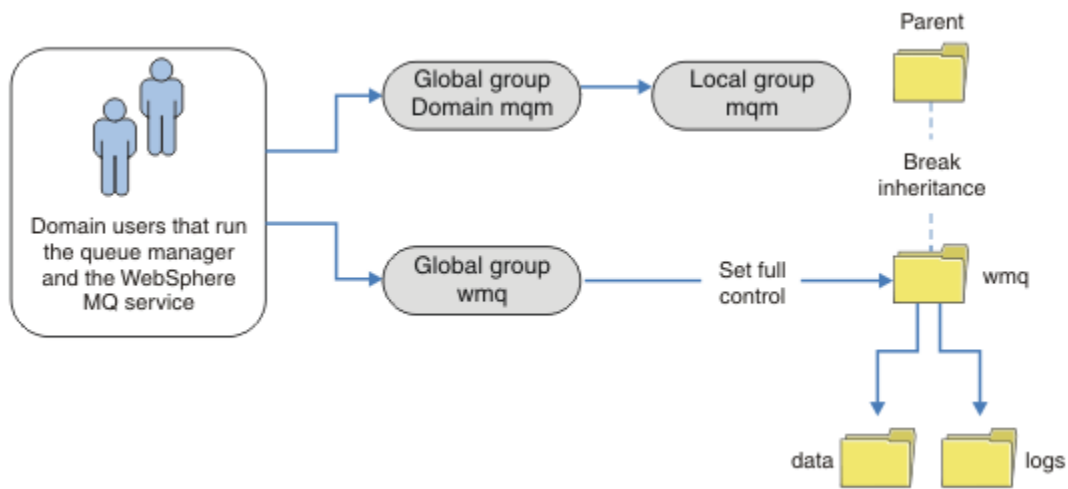


Abbildung 65. Sichern von Warteschlangenmanagerdaten und -protokollen mithilfe einer alternativen globalen Sicherheitsgruppe (2)

Die Organisation in [Abbildung 65](#) auf Seite 405 ist nach dem Stand der Dinge unnötig kompliziert. Die Anordnung hat zwei globale Gruppen mit identischen Mitgliedern. Sie können die Organisation vereinfachen und nur eine globale Gruppe definieren (siehe [Abbildung 66](#) auf Seite 405).

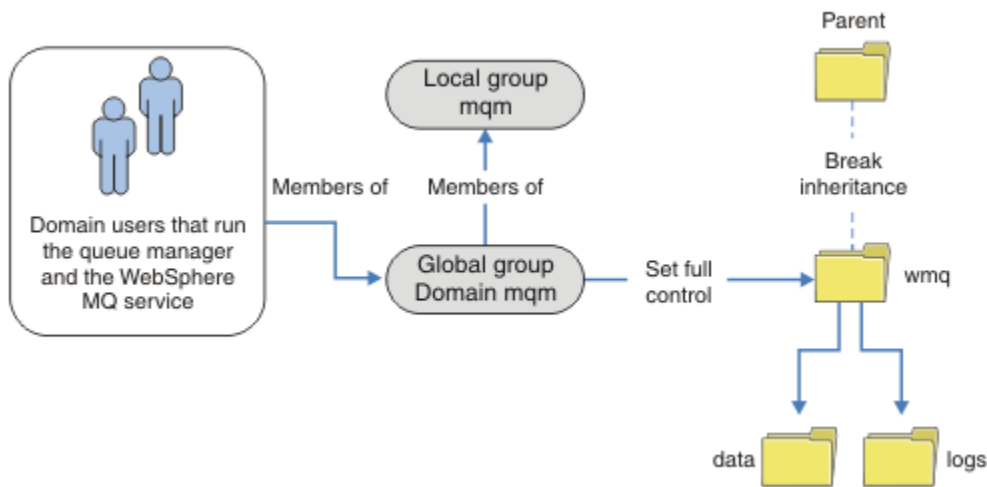


Abbildung 66. Sichern von WS-Manager-Daten und -Protokollen mithilfe einer alternativen globalen Sicherheitsgruppe (3)

Alternativ hierzu benötigen Sie möglicherweise eine feinere Zugriffssteuerung, wobei die verschiedenen Warteschlangenmanager auf verschiedene Ordner zugreifen können. Weitere Informationen finden Sie im Abschnitt [Abbildung 67](#) auf Seite 406. In [Abbildung 67](#) auf Seite 406 sind zwei Gruppen von Domänenbenutzern definiert, in separaten globalen Gruppen, um verschiedene WS-Manager-Protokoll- und Datendateien zu sichern. Es werden zwei verschiedene lokale mqm -Gruppen angezeigt, die sich auf verschiedenen IBM WebSphere MQ -Servern befinden müssen. In diesem Beispiel werden die Warteschlan-

genmanager in zwei Gruppen unterteilt, wobei verschiedene Benutzer den beiden Gruppen zugeordnet sind. Die beiden Gruppen können Test- und Produktionswarteschlangenmanager sein. Die alternativen Sicherheitsgruppen heißen wmq1 und wmq2. Sie müssen die globalen Gruppen wmq1 und wmq2 den richtigen Warteschlangenmanagern hinzufügen, je nachdem, ob sie sich in der Test- oder Produktionsabteilung befinden. Die Konfiguration kann nicht nutzen, dass die Installation von IBM WebSphere MQ Domain mqm an die lokale mqm -Gruppe wie in [Abbildung 66](#) auf Seite 405 weitergibt, da es zwei Benutzergruppen gibt.

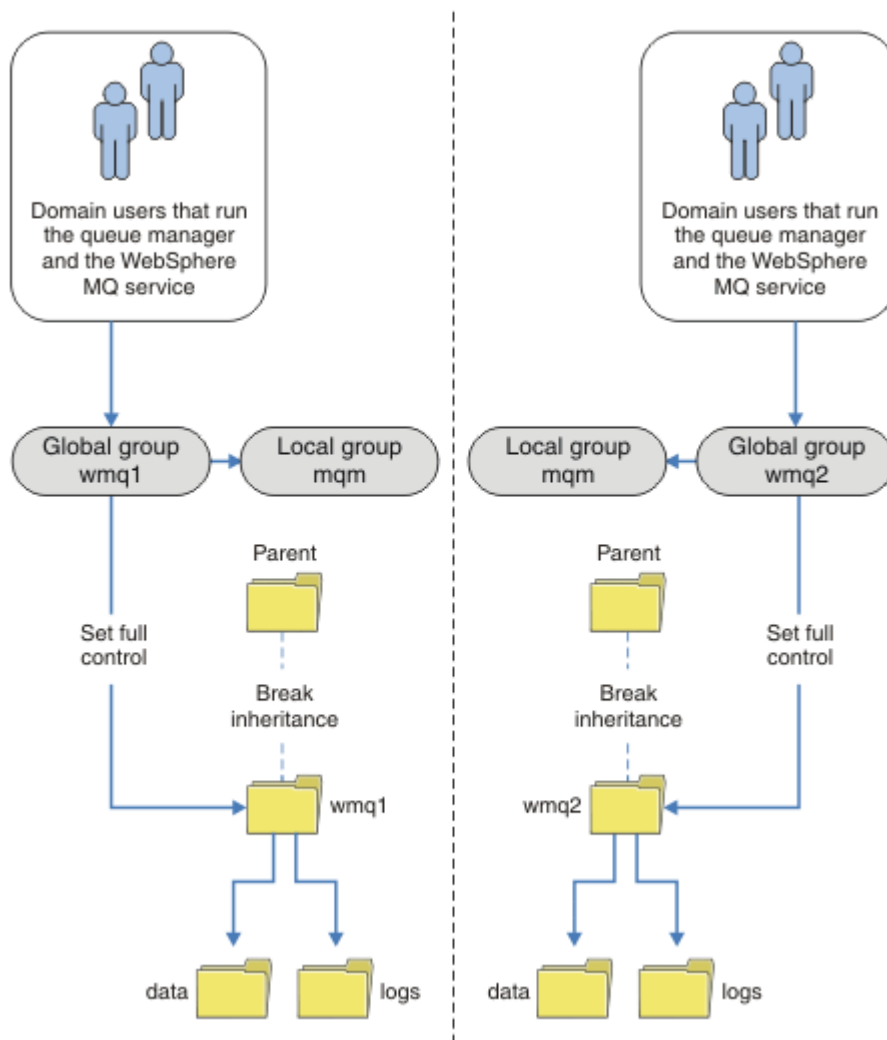


Abbildung 67. Sichern von Warteschlangenmanagerdaten und -protokollen mit einem alternativen globalen Sicherheitsanbieter (4)

Eine alternative Möglichkeit, zwei Abteilungen zu partitionieren, besteht darin, sie in zwei Windows-Domänen zu platzieren. In diesem Fall können Sie zur Verwendung des einfacheren Modells zurückkehren, das in [Abbildung 66](#) auf Seite 405 angezeigt wird.

#### *Nicht gemeinsam genutzte WS-Manager-Daten und -Protokollverzeichnisse und -Dateien unter Windows schützen*

In diesem Thema wird beschrieben, wie Sie eine alternative Position für WS-Manager-Daten und -Protokolldateien sichern können, indem Sie die lokale mqm -Gruppe und eine alternative Sicherheitsgruppe verwenden.

In der Regel wird keine alternative Position für WS-Manager-Daten und -Protokolldateien eingerichtet. Wenn Sie IBM WebSphere MQ for Windows installieren, erstellt das Installationsprogramm ein Ausgangsverzeichnis Ihrer Wahl für alle Warteschlangenmanager, die erstellt werden. Es sichert die Verzeichnisse

mit der lokalen Gruppe mqm und konfiguriert eine Benutzer-ID für den IBM WebSphere MQ -Service, um auf die Verzeichnisse zuzugreifen.

In zwei Beispielen wird gezeigt, wie die Zugriffssteuerung für IBM WebSphere MQ konfiguriert wird. In den Beispielen wird gezeigt, wie ein Warteschlangenmanager mit seinen Daten und Protokollen in Verzeichnissen erstellt wird, die sich nicht auf den von der Installation erstellten Daten und Protokollpfaden befinden. Im ersten Beispiel ermöglichen Sie „Lesen und Schreiben von Daten und Protokolldateien, die von der lokalen mqm-Gruppe autorisiert sind“ auf Seite 408 den Zugriff auf die Warteschlangen- und Protokollverzeichnisse, indem Sie die Berechtigung durch die lokale mqm-Gruppe zulassen. Das zweite Beispiel, „Lesen und Schreiben von Daten- und Protokolldateien, die von einer alternativen lokalen Sicherheitsgruppe autorisiert sind“ auf Seite 411, unterscheidet sich dadurch, dass der Zugriff auf die Verzeichnisse durch eine alternative Sicherheitsgruppe autorisiert wird. Wenn auf die Verzeichnisse von einem Warteschlangenmanager zugegriffen wird, der auf einem einzigen Server ausgeführt wird, müssen Sie die Daten- und Protokolldateien mit der alternativen Sicherheitsgruppe sichern, um verschiedene Warteschlangenmanager mit unterschiedlichen lokalen Gruppen oder Principals zu sichern. Wenn auf die Verzeichnisse von einem Warteschlangenmanager zugegriffen wird, der auf verschiedenen Servern ausgeführt wird, z. B. mit einem Warteschlangenmanager mit mehreren Instanzen, ist die sichere Sicherung der Daten und Protokolldateien mit der alternativen Sicherheitsgruppe die einzige Option (siehe „Gemeinsam genutzte Warteschlangenmanagerdaten, Protokollverzeichnisse und Dateien unter Windows sichern“ auf Seite 403).

Die Konfiguration der Sicherheitsberechtigungen von WS-Manager-Daten und -Protokolldateien ist keine allgemeine Task unter Windows. Wenn Sie IBM WebSphere MQ for Windows installieren, geben Sie entweder Verzeichnisse für WS-Manager-Daten und -Protokolle an, oder akzeptieren Sie die Standardverzeichnisse. Das Installationsprogramm sichert diese Verzeichnisse automatisch mit der lokalen mqm-Gruppe und gibt ihm die vollständige Steuerungsberechtigung. Der Installationsprozess stellt sicher, dass die Benutzer-ID, die Warteschlangenmanager ausführt, Mitglied der lokalen mqm-Gruppe ist. Sie können die anderen Zugriffsberechtigungen für die Verzeichnisse so ändern, dass sie Ihren Zugriffs Voraussetzungen entsprechen.

Wenn Sie das Verzeichnis für Daten- und Protokolldateien an neue Positionen verschieben, müssen Sie die Sicherheit der neuen Speicherpositionen konfigurieren. Sie können die Position der Verzeichnisse ändern, wenn Sie einen Warteschlangenmanager sichern und auf einem anderen Computer wiederherstellen, oder wenn Sie den Warteschlangenmanager als WS-Manager mit mehreren Instanzen ändern. Sie haben die Wahl zwischen zwei Möglichkeiten, die WS-Manager-Daten und die Protokollverzeichnisse an ihrer neuen Position zu sichern. Sie können die Verzeichnisse sichern, indem Sie den Zugriff auf die lokale mqm-Gruppe einschränken, oder Sie können den Zugriff auf eine beliebige Sicherheitsgruppe Ihrer Wahl einschränken.

Es wird mindestens die Anzahl der Schritte zur Sicherung der Verzeichnisse mit der lokalen Gruppe mqm benötigt. Legen Sie die Berechtigungen für die Daten- und Protokollverzeichnisse fest, um die vollständige Steuerung der lokalen mqm-Gruppe zu ermöglichen. Ein typischer Ansatz besteht darin, die vorhandene Gruppe von Berechtigungen zu kopieren und die Vererbung aus dem übergeordneten Element zu entfernen. Anschließend können Sie die Berechtigungen anderer Principals entfernen oder einschränken.

Wenn Sie den Warteschlangenmanager unter einer anderen Benutzer-ID als der vom IBM WebSphere MQ-Vorbereitungsassistenten festgelegte Service ausführen, muss diese Benutzer-ID Mitglied der lokalen Gruppe mqm sein. Die Task „Lesen und Schreiben von Daten und Protokolldateien, die von der lokalen mqm-Gruppe autorisiert sind“ auf Seite 408 führt Sie durch die Schritte.

Sie können auch WS-Manager-Daten und -Protokolldateien mit einer alternativen Sicherheitsgruppe sichern. Der Prozess der Sicherung der WS-Manager-Daten und -Protokolldateien mit der alternativen Sicherheitsgruppe enthält eine Reihe von Schritten, die sich auf Abbildung 68 auf Seite 408 beziehen. Die lokale Gruppe mqm ist ein Beispiel für eine alternative Sicherheitsgruppe.

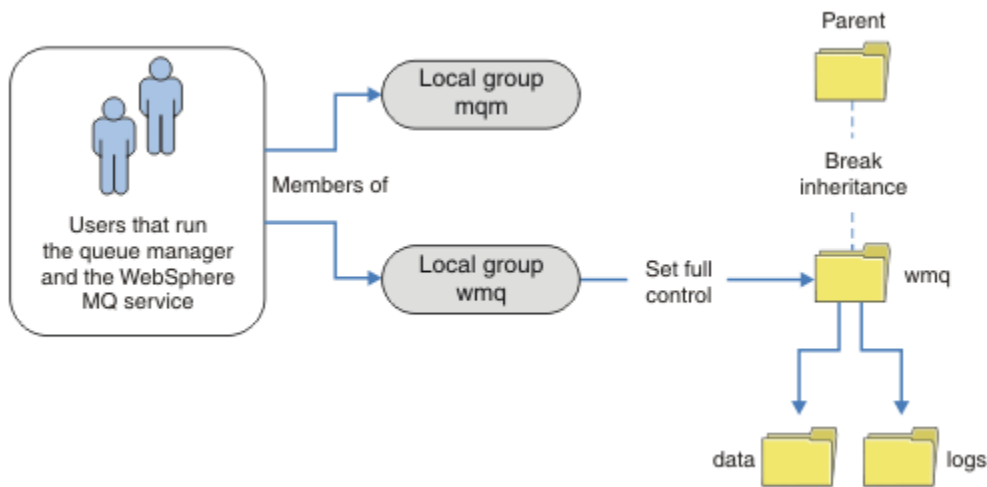


Abbildung 68. Sichern von WS-Manager-Daten und -Protokollen mit einer alternativen lokalen Sicherheitsgruppe, wmq

1. Erstellen Sie entweder separate Verzeichnisse für die WS-Manager-Daten und -Protokolle, ein allgemeines Verzeichnis oder ein gemeinsames übergeordnetes Verzeichnis.
2. Kopieren Sie die vorhandene Gruppe von übernommenen Berechtigungen für die Verzeichnisse oder das übergeordnete Verzeichnis, und ändern Sie sie entsprechend Ihren Anforderungen.
3. Sichern Sie die Verzeichnisse, die den Warteschlangenmanager und die Protokolle enthalten sollen, indem Sie der alternativen Gruppe wmq die vollständige Steuerung der Berechtigung für die Verzeichnisse erteilen.
4. Geben Sie allen Benutzer-IDs, die Warteschlangenmanager ausführen, die Berechtigungsnachweise der alternativen Sicherheitsgruppe oder des Principals an:
  - a. Wenn Sie einen Benutzer als alternativen Sicherheitprincipal definieren, muss der Benutzer derselbe Benutzer sein, unter dem der WS-Manager ausgeführt wird. Der Benutzer muss ein Mitglied der lokalen mqm -Gruppe sein.
  - b. Wenn Sie eine lokale Gruppe als alternative Sicherheitsgruppe definieren, fügen Sie den Benutzer hinzu, den der WS-Manager unter der alternativen Gruppe ausführen wird. Der Benutzer muss auch ein Mitglied der lokalen mqm -Gruppe sein.
  - c. Wenn Sie eine globale Gruppe als alternative Sicherheitsgruppe definieren, finden Sie weitere Informationen in „Gemeinsam genutzte Warteschlangenmanagerdaten, Protokollverzeichnisse und Dateien unter Windows sichern“ auf Seite 403.
5. Erstellen Sie den Warteschlangenmanager unter Angabe der alternativen Sicherheitsgruppe oder des alternativen Principals im Befehl **crtmqm** mit dem Parameter -a .

*Lesen und Schreiben von Daten und Protokolldateien, die von der lokalen mqm-Gruppe autorisiert sind*

Die Task veranschaulicht, wie ein Warteschlangenmanager mit seinen Daten erstellt wird und die Dateien in einem beliebigen Verzeichnis Ihrer Wahl gespeichert werden. Der Zugriff auf die Dateien wird durch die lokale mqm -Gruppe gesichert. Das Verzeichnis wird nicht gemeinsam genutzt.

## Vorbereitende Schritte

1. Installieren Sie IBM WebSphere MQ for Windows als primäre Installation.
2. Führen Sie den Assistenten "Prepare IBM WebSphere MQ" (WebSphere MQ vorbereiten) aus. Konfigurieren Sie für diese Task die Installation entweder für die Ausführung mit einer lokalen Benutzer-ID oder für eine Domänenbenutzer-ID. Beachten Sie allerdings, dass die Installation später ohnehin



für eine Domäne konfiguriert werden muss, um alle Tasks von „[Windows-Domänen und Multi-Instanz-Warteschlangenmanager](#)“ auf Seite 376 ausführen zu können.

3. Melden Sie sich mit Administratorberechtigung an, um den ersten Teil der Task auszuführen.

## Informationen zu diesem Vorgang

Diese Task ist eine von einer Gruppe zusammengehöriger Tasks, die den Zugriff auf Warteschlangenmanagerdaten und Protokolldateien veranschaulichen. Die Tasks zeigen, wie ein Warteschlangenmanager erstellt wird, der berechtigt ist, Daten und Protokolldateien zu lesen und zu schreiben, die in einem Verzeichnis Ihrer Wahl gespeichert sind. Sie begleiten die Task „[Windows-Domänen und Multi-Instanz-Warteschlangenmanager](#)“ auf Seite 376.

Unter Windows können Sie die Standarddaten- und Protokollpfade für IBM WebSphere MQ for Windows in beliebigen Verzeichnissen Ihrer Wahl erstellen. Der Installations- und Konfigurationsassistent erteilt der lokalen `mqm`-Gruppe und der Benutzer-ID, die die Warteschlangenmanagerprozesse ausführt, automatisch Zugriff auf die Verzeichnisse. Wenn Sie einen Warteschlangenmanager erstellen, der unterschiedliche Verzeichnisse für WS-Manager-Daten und -Protokolldateien angibt, müssen Sie die vollständige Steuerberechtigung für die Verzeichnisse konfigurieren.

In diesem Beispiel geben Sie dem Warteschlangenmanager die vollständige Kontrolle über seine Daten- und Protokolldateien, indem Sie der lokalen `mqm`-Gruppe die Berechtigung für das Verzeichnis `c:\wmq` erteilen.

Der Befehl `crtmqm` erstellt einen Warteschlangenmanager, der automatisch gestartet wird, wenn die Workstation mit dem IBM WebSphere MQ -Service gestartet wird.

Die Task ist anschaulich. Sie verwendet bestimmte Werte, die Sie ändern können. Die Werte, die Sie ändern können, sind in Kursivschrift. Führen Sie am Ende der Task die Anweisungen aus, um alle Änderungen zu entfernen, die Sie vorgenommen haben.

## Vorgehensweise

1. Öffnen Sie eine Eingabeaufforderung.
2. Geben Sie den folgenden Befehl ein:

```
md c:\wmq\data , c:\wmq\logs
```

3. Legen Sie die Berechtigungen für die Verzeichnisse fest, um der lokalen `mqm`-Gruppe Lese- und Schreibzugriff zu erteilen.

```
cacls c:\wmq /T /E /G mqm:F
```

Systemantwort:

```
processed dir: c:\wmq
processed dir: c:\wmq\data
processed dir: c:\wmq\logs
```

4. Optional: Wechseln Sie zu einer Benutzer-ID, die Mitglied der lokalen `mqm`-Gruppe ist.

Sie können als Administrator fortfahren, aber für eine realistische Produktionskonfiguration mit einer Benutzer-ID fortfahren, die mehr eingeschränkte Rechte hat. Die Benutzer-ID muss mindestens ein Mitglied der lokalen `mqm`-Gruppe sein.

Wenn die IBM WebSphere MQ-Installation als Teil einer Domäne konfiguriert ist, machen Sie die Benutzer-ID zu einem Mitglied der Gruppe `Domain mqm`. Der Assistent "IBM WebSphere MQ vorbereiten" macht die globale Gruppe `Domain mqm` zu einem Mitglied der lokalen Gruppe `mqm`, d. h., Sie müssen die Benutzer-ID nicht direkt zu einem Mitglied der lokalen Gruppe `mqm` machen.

5. Erstellen Sie den Warteschlangenmanager.

```
crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md c:\wmq\data -ld c:\wmq\logs QMGR
```

Systemantwort:

```
WebSphere MQ queue manager created.
Directory 'c:\wmq\data\QMGR' created.
The queue manager is associated with installation '1'
Creating or replacing default objects for queue manager 'QMGR'
Default objects statistics : 74 created. 0 replaced.
Completing setup.
Setup completed.
```

6. Stellen Sie sicher, dass sich die vom Warteschlangenmanager erstellten Verzeichnisse im Verzeichnis `c:\wmq` befinden.

```
dir c:\wmq /D /B /S
```

7. Stellen Sie sicher, dass die Dateien Lese- und Schreibberechtigung oder vollständige Steuerungs- berechtigung für die lokale mqm-Gruppe haben.

```
cacls c:\wmq*.*
```

## Nächste Schritte

Testen Sie den Warteschlangenmanager, indem Sie eine Nachricht in eine Warteschlange einreihen und eine Nachricht erhalten.

1. Starten Sie den Warteschlangenmanager.

```
strmqm QMGR
```

Systemantwort:

```
WebSphere MQ queue manager 'QMGR' starting.
The queue manager is associated with installation '1'.
5 log records accessed on queue manager 'QMGR' during the log
replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
WebSphere MQ queue manager 'QMGR' started using V7.1.0.0.
```

2. Erstellen Sie eine Testwarteschlange.

```
echo define qlocal(QTEST) | runmqsc QMGR
```

Systemantwort:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QMGR.
```

```
1 : define qlocal(QTEST)
AMQ8006: WebSphere MQ queue created.
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

3. Reihen Sie eine Testnachricht mit dem Beispielprogramm **amqspu**tein.

```
echo 'A test message' | amqsput QTEST QMGR
```

Systemantwort:

```
Sample AMQSPUT0 start
target queue is QTEST
Sample AMQSPUT0 end
```

4. Rufen Sie die Testnachricht mit dem Beispielprogramm **amqsgetab**.

```
amqsget QTEST QMGR
```

Systemantwort:

```
Sample AMQSGETO start
message <A test message>
Wait 15 seconds ...
no more messages
Sample AMQSGETO end
```

5. Stoppen Sie den Warteschlangenmanager.

```
endmqm -i QMGR
```

Systemantwort:

```
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ended.
```

6. Löschen Sie den Warteschlangenmanager.

```
dltmqm QMGR
```

Systemantwort:

```
WebSphere MQ queue manager 'QMGR' deleted.
```

7. Löschen Sie die von Ihnen erstellten Verzeichnisse.

**Tipp:** Fügen Sie die Option /Q zu den Befehlen hinzu, um zu verhindern, dass der Befehl zum Löschen jeder Datei oder jedes Verzeichnisses auffordert.

```
del /F /S C:\wmq*. *
rmdir /S C:\wmq
```

### Zugehörige Konzepte

„Windows-Domänen und Multi-Instanz-Warteschlangenmanager“ auf Seite 376

Ein Multi-Instanz-Warteschlangenmanager auf Windows setzt voraus, dass seine Daten und Protokolle gemeinsam genutzt werden. Die Freigabe muss für alle Instanzen des Warteschlangenmanagers, die auf verschiedenen Servern oder Workstations ausgeführt werden, zugänglich sein. Konfigurieren Sie die Warteschlangenmanager und nutzen Sie sie als Teil einer Windows-Domäne gemeinsam. Der Warteschlangenmanager kann auf einer Domänenworkstation oder einem Server oder auf dem Domänencontroller ausgeführt werden.

### Zugehörige Tasks

„Lesen und Schreiben von Daten- und Protokolldateien, die von einer alternativen lokalen Sicherheitsgruppe autorisiert sind“ auf Seite 411

Diese Task veranschaulicht, wie das Flag -a im Befehl **crtmqm** verwendet wird. Die Markierung stellt dem Warteschlangenmanager eine alternative lokale Sicherheitsgruppe zur Verfügung, mit der er Zugriff auf seine Protokoll- und Datendateien erhält.

„Lesen und Schreiben von gemeinsam genutzten Daten und Protokolldateien, die von einer alternativen globalen Sicherheitsgruppe autorisiert sind“ auf Seite 390

„Multi-Instanz-Warteschlangenmanager auf Workstations oder Servern in einer Domäne erstellen“ auf Seite 377

*Lesen und Schreiben von Daten- und Protokolldateien, die von einer alternativen lokalen Sicherheitsgruppe autorisiert sind*

Diese Task veranschaulicht, wie das Flag -a im Befehl **crtmqm** verwendet wird. Die Markierung stellt dem Warteschlangenmanager eine alternative lokale Sicherheitsgruppe zur Verfügung, mit der er Zugriff auf seine Protokoll- und Datendateien erhält.

## Vorbereitende Schritte

1. Installieren Sie IBM WebSphere MQ for Windows als primäre Installation.
2. Führen Sie den Assistenten "Prepare IBM WebSphere MQ" (WebSphere MQ vorbereiten) aus. Konfigurieren Sie für diese Task die Installation entweder für die Ausführung mit einer lokalen Benutzer-ID oder für eine Domänenbenutzer-ID. Beachten Sie allerdings, dass die Installation später ohnehin für eine Domäne konfiguriert werden muss, um alle Tasks von „Windows-Domänen und Multi-Instanz-Warteschlangenmanager“ auf Seite 376 ausführen zu können.
3. Melden Sie sich mit Administratorberechtigung an, um den ersten Teil der Task auszuführen.

## Informationen zu diesem Vorgang

Diese Task ist eine von einer Gruppe zusammengehöriger Tasks, die den Zugriff auf Warteschlangenmanagerdaten und Protokolldateien veranschaulichen. Die Tasks zeigen, wie ein Warteschlangenmanager erstellt wird, der berechtigt ist, Daten und Protokolldateien zu lesen und zu schreiben, die in einem Verzeichnis Ihrer Wahl gespeichert sind. Sie begleiten die Task „Windows-Domänen und Multi-Instanz-Warteschlangenmanager“ auf Seite 376.

Unter Windows können Sie die Standarddaten- und Protokollpfade für IBM WebSphere MQ for Windows in beliebigen Verzeichnissen Ihrer Wahl erstellen. Der Installations- und Konfigurationsassistent erteilt der lokalen `mqm`-Gruppe und der Benutzer-ID, die die Warteschlangenmanagerprozesse ausführt, automatisch Zugriff auf die Verzeichnisse. Wenn Sie einen Warteschlangenmanager erstellen, der unterschiedliche Verzeichnisse für WS-Manager-Daten und -Protokolldateien angibt, müssen Sie die vollständige Steuerberechtigung für die Verzeichnisse konfigurieren.

In diesem Beispiel stellen Sie dem Warteschlangenmanager eine alternative lokale Sicherheitsgruppe zur Verfügung, die über die vollständige Steuerungsberechtigung für die Verzeichnisse verfügt. Die alternative Sicherheitsgruppe erteilt dem Warteschlangenmanager die Berechtigung zum Verwalten von Dateien im Verzeichnis. Der primäre Zweck der alternativen Sicherheitsgruppe besteht darin, eine alternative globale Sicherheitsgruppe zu autorisieren. Verwenden Sie eine alternative globale Sicherheitsgruppe, um einen WS-Manager mit mehreren Instanzen einzurichten. In diesem Beispiel konfigurieren Sie eine lokale Gruppe, um sich mit der Verwendung einer alternativen Sicherheitsgruppe vertraut zu machen, ohne IBM WebSphere MQ in einer Domäne installieren zu müssen. Es ist ungewöhnlich, eine lokale Gruppe als alternative Sicherheitsgruppe zu konfigurieren.

Der Befehl `crtmqm` erstellt einen Warteschlangenmanager, der automatisch gestartet wird, wenn die Workstation mit dem IBM WebSphere MQ -Service gestartet wird.

Die Task ist anschaulich. Sie verwendet bestimmte Werte, die Sie ändern können. Die Werte, die Sie ändern können, sind in Kursivschrift. Führen Sie am Ende der Task die Anweisungen aus, um alle Änderungen zu entfernen, die Sie vorgenommen haben.

## Vorgehensweise

1. Richten Sie eine alternative Sicherheitsgruppe ein.

Die alternative Sicherheitsgruppe ist normalerweise eine Domänengruppe. In diesem Beispiel erstellen Sie einen Warteschlangenmanager, der eine lokale alternative Sicherheitsgruppe verwendet. Wenn Sie eine lokale alternative Sicherheitsgruppe verwenden, können Sie die Task mit einer IBM WebSphere MQ -Installation ausführen, die nicht Teil einer Domäne ist.

- a) Führen Sie den Befehl `luszmgr.msc` aus, um das Fenster "Lokale Benutzer und Gruppen" zu öffnen.
- b) Klicken Sie mit der rechten Maustaste auf **Gruppen > Neue Gruppe...**
- c) Geben Sie im Feld **Gruppenname** `al\mqm` ein und klicken Sie auf **Erstellen > Schließen**.
- d) Ermitteln Sie die Benutzer-ID, die den Service IBM WebSphere MQ ausführt.
  - i) Klicken Sie auf **Starten > Ausführen ...**, Geben Sie `services.msc` ein und klicken Sie auf **OK**.

- ii) Klicken Sie in der Liste der Services auf den Service IBM IBM WebSphere MQ und anschließend auf die Registerkarte 'Anmelden'.
- iii) Merken Sie sich die Benutzer-ID, und schließen Sie den Service-Explorer
- e) Fügen Sie die Benutzer-ID, die den IBM IBM WebSphere MQ -Service ausführt, zur Gruppe *altnmqm* hinzu. Fügen Sie außerdem die Benutzer-ID hinzu, mit der Sie sich anmelden, um einen Warteschlangenmanager zu erstellen, und führen Sie ihn interaktiv aus.

Windows überprüft die Zugriffsberechtigung des Warteschlangenmanagers auf die Daten- und Protokollverzeichnisse, indem er die Berechtigung der Benutzer-ID überprüft, die Warteschlangenmanagerprozesse ausführt. Die Benutzer-ID muss direkt oder indirekt über eine globale Gruppe ein Mitglied der *altnmqm*-Gruppe, die die Verzeichnisse autorisiert hat, sein.

Wenn Sie IBM WebSphere MQ als Teil einer Domäne installiert haben und die Tasks in „Multi-Instanz-Warteschlangenmanager auf Workstations oder Servern in einer Domäne erstellen“ auf Seite 377 ausführen werden, sind die in „Active Directory und DNS-Domäne für IBM WebSphere MQ“ auf Seite 380 erstellten Domänenbenutzer-IDs *wmquser1* und *wmquser2*.

Wenn Sie den Warteschlangenmanager nicht als Teil einer Domäne installiert haben, lautet die standardmäßige lokale Benutzer-ID, die den IBM IBM WebSphere MQ -Service ausführt, *MUSR\_MQADMIN*. Wenn Sie die Tasks ohne Administratorberechtigung ausführen wollen, erstellen Sie einen Benutzer, der Mitglied der lokalen *mqm* -Gruppe ist.

Befolgen Sie diese Schritte, um *wmquser1* und *wmquser2* zu *altnmqm* hinzuzufügen. Wenn Ihre Konfiguration anders ist, ersetzen Sie Ihre Namen durch die Benutzer-IDs und die Gruppe.

- i) Klicken Sie in der Liste der Gruppen mit der rechten Maustaste auf **altnmqm > Eigenschaften > Hinzufügen ....**
  - ii) Geben Sie im Fenster "Benutzer, Computer oder Gruppen auswählen" *wmquser1; wmquser2* ein und klicken Sie auf **Namen prüfen**.
  - iii) Geben Sie den Namen und das Kennwort eines Domänenadministrators im Fenster 'Windows-Sicherheit' ein und klicken Sie auf **OK > OK > Anwenden > OK**.
2. Öffnen Sie eine Eingabeaufforderung.
  3. Führen Sie einen Neustart für den IBM IBM WebSphere MQ-Service aus.

Sie müssen den Service erneut starten, damit die Benutzer-ID, unter der er ausgeführt wird, die zusätzlichen Sicherheitsberechtigungenachweise erhält, die Sie für die Benutzer-ID konfiguriert haben.

Geben Sie die Befehle ein:

```
endmqsvc
strmqsvc
```

Die Systemantworten:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
The MQ service for installation 'Installation1' ended successfully.
```

Und:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
The MQ service for installation 'Installation1' started successfully.
```

4. Geben Sie den folgenden Befehl ein:

```
md c:\wmq\data , c:\wmq\logs
```

5. Legen Sie die Berechtigungen für die Verzeichnisse fest, um dem lokalen Benutzer *user* Lese- und Schreibzugriff zu erteilen.

```
cacls c:\wmq /T /E /G altnmqm:F
```

Systemantwort:

```
processed dir: c:\wmq
processed dir: c:\wmq\data
processed dir: c:\wmq\logs
```

6. Optional: Wechseln Sie zu einer Benutzer-ID, die Mitglied der lokalen mqm-Gruppe ist.

Sie können als Administrator fortfahren, aber für eine realistische Produktionskonfiguration mit einer Benutzer-ID fortfahren, die mehr eingeschränkte Rechte hat. Die Benutzer-ID muss mindestens ein Mitglied der lokalen mqm-Gruppe sein.

Wenn die IBM WebSphere MQ-Installation als Teil einer Domäne konfiguriert ist, machen Sie die Benutzer-ID zu einem Mitglied der Gruppe Domain mqm. Der Assistent "IBM WebSphere MQ vorbereiten" macht die globale Gruppe Domain mqm zu einem Mitglied der lokalen Gruppe mqm, d. h., Sie müssen die Benutzer-ID nicht direkt zu einem Mitglied der lokalen Gruppe mqm machen.

7. Erstellen Sie den Warteschlangenmanager.

```
crtmqm -a altmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md c:\wmq\data -ld c:\wmq\logs QMGR
```

Systemantwort:

```
WebSphere MQ queue manager created.
Directory 'c:\wmq1\data\QMGR' created.
The queue manager is associated with installation '1'
Creating or replacing default objects for queue manager 'QMGR'
Default objects statistics : 74 created. 0 replaced.
Completing setup.
Setup completed.
```

8. Stellen Sie sicher, dass sich die vom Warteschlangenmanager erstellten Verzeichnisse im Verzeichnis c:\wmq befinden.

```
dir c:\wmq /D /B /S
```

9. Stellen Sie sicher, dass die Dateien Lese- und Schreibberechtigung oder vollständige Steuerungsrechte für die lokale mqm-Gruppe haben.

```
cacls c:\wmq*.*
```

## Nächste Schritte

Testen Sie den Warteschlangenmanager, indem Sie eine Nachricht in eine Warteschlange einreihen und eine Nachricht erhalten.

1. Starten Sie den Warteschlangenmanager.

```
strmqm QMGR
```

Systemantwort:

```
WebSphere MQ queue manager 'QMGR' starting.
The queue manager is associated with installation '1'.
5 log records accessed on queue manager 'QMGR' during the log
replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
WebSphere MQ queue manager 'QMGR' started using V7.1.0.0.
```

2. Erstellen Sie eine Testwarteschlange.

```
echo define qlocal(QTEST) | runmqsc QMGR
```

Systemantwort:

5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.  
Starting MQSC for queue manager QMGR.

```
1 : define qlocal(QTEST)
AMQ8006: WebSphere MQ queue created.
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

3. Reihen Sie eine Testnachricht mit dem Beispielprogramm **amqspu**tein.

```
echo 'A test message' | amqsput QTEST QMGR
```

Systemantwort:

```
Sample AMQSPUT0 start
target queue is QTEST
Sample AMQSPUT0 end
```

4. Rufen Sie die Testnachricht mit dem Beispielprogramm **amqsget**ab.

```
amqsget QTEST QMGR
```

Systemantwort:

```
Sample AMQSGET0 start
message <A test message>
Wait 15 seconds ...
no more messages
Sample AMQSGET0 end
```

5. Stoppen Sie den Warteschlangenmanager.

```
endmqm -i QMGR
```

Systemantwort:

```
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ended.
```

6. Löschen Sie den Warteschlangenmanager.

```
dltmqm QMGR
```

Systemantwort:

```
WebSphere MQ queue manager 'QMGR' deleted.
```

7. Löschen Sie die von Ihnen erstellten Verzeichnisse.

**Tipp:** Fügen Sie die Option /Q zu den Befehlen hinzu, um zu verhindern, dass der Befehl zum Löschen jeder Datei oder jedes Verzeichnisses auffordert.

```
del /F /S C:\wmq*. *
rmdir /S C:\wmq
```

### *Mehrinstanz-Warteschlangenmanager unter Linux erstellen*

Anhand eines Beispiels sehen Sie, wie Sie einen Warteschlangenmanager mit mehreren Instanzen unter Linux konfigurieren können. Die Konfiguration ist klein, um die Konzepte zu veranschaulichen. Das Beispiel basiert auf Linux Red Hat Enterprise 5. Die Schritte unterscheiden sich auf anderen UNIX -Plattformen.

Das Beispiel wird auf einem 2-GHz-Notebook-Computer mit 3 GB RAM eingerichtet, auf dem Windows XP Service-Pack 2 ausgeführt wird. Zwei virtuelle VMware-Maschinen, Server1 und Server2, führen Li-

nux Red Hat Enterprise 5 in 640 MB Images aus. Server1 enthält das Network File System (NFS), die WS-Manager-Protokolle und eine HA-Instanz. Es ist nicht üblich, dass der NFS-Server auch eine der WS-Manager-Instanzen hosten kann. Dies ist die Vereinfachung des Beispiels. Server2 hängt die Protokolle des Server1-Warteschlangenmanagers mit einer Standby-Instanz an. Ein WebSphere MQ MQI-Client wird auf einem zusätzlichen VMware -Image mit 400 MB installiert, das Windows XP Service Pack 2 ausführt und die Beispielanwendungen für hohe Verfügbarkeit ausführt. Alle virtuellen Maschinen werden aus Sicherheitsgründen als Teil eines VMware-Host-Netztes konfiguriert.

**Anmerkung:** Sie sollten nur WS-Manager-Daten auf einem NFS-Server in die Warteschlange stellen. Verwenden Sie im -NFS die folgenden drei Optionen mit dem Befehl mount, um das System sicher zu machen:

**noexec**

Wenn Sie diese Option verwenden, stoppen Sie die Ausführung von Binärdateien auf dem NFS, wodurch verhindert wird, dass ein ferner Benutzer nicht mehr benötigten Code auf dem System ausführen kann.

**nosuid**

Wenn Sie diese Option verwenden, verhindern Sie die Verwendung der Bits "set-user-identifier" und "set-group-identifier bits", die verhindert, dass ein ferner Benutzer höhere Berechtigungen erhält.

**nodev**

Wenn Sie diese Option verwenden, stoppen Sie die Zeichen- und Blockspezial-Einheiten, die verwendet oder definiert werden, wodurch verhindert wird, dass ein ferner Benutzer aus einem chroot-Gefängnis heraus kommt.

**Beispiel**

| <i>Tabelle 30. Veranschaulichung der Konfiguration eines Multi-Instanz-Warteschlangenmanagers unter Linux</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                                                                                                                                                                                                                         |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Server1</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | <b>Server2</b>                                                                                                                                                                                                          |
| Melden Sie sich als <i>Root</i> an.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |                                                                                                                                                                                                                         |
| <p>Befolgen Sie die Anweisungen unter <a href="#">IBM WebSphere MQ installieren</a> , um WebSphere MQ zu installieren, den Benutzer und die Gruppe 'mqm' zu erstellen, wenn diese nicht vorhanden sind, und definieren Sie /var/mqm. Überprüfen Sie, was uid und gid in /etc/passwd auf der ersten Maschine für mqm anzeigt. Beispiel:</p> <pre>mqm:x:501:100:MQ User:/var/mqm:/bin/bash</pre> <p>Stimmen Sie uid und gid für mqm in /etc/passwd auf der zweiten Maschine überein, um sicherzustellen, dass diese Werte identisch sind. Starten Sie diese Maschine erneut, wenn Sie die Werte ändern müssen.</p> |                                                                                                                                                                                                                         |
| Führen Sie die Task <a href="#">Verhalten des gemeinsam genutzten Dateisystems überprüfen</a> aus, um zu überprüfen, ob das Dateisystem mehrere Instanzen von Warteschlangenmanagern mit mehreren Instanzen unterstützt.                                                                                                                                                                                                                                                                                                                                                                                         |                                                                                                                                                                                                                         |
| <p>Erstellen Sie Protokoll- und Datenverzeichnisse in einem allgemeinen Ordner ( /MQHA), der gemeinsam genutzt werden soll. Beispiel:</p> <ol style="list-style-type: none"> <li><b>mkdir</b> /MQHA</li> <li><b>mkdir</b> /MQHA/logs</li> <li><b>mkdir</b> /MQHA/qmgrs</li> </ol>                                                                                                                                                                                                                                                                                                                                | <p>Erstellen Sie den Ordner /MQHA, um das gemeinsam genutzte Dateisystem anzuhängen. Verwenden Sie denselben Pfad wie auf Server1; . Beispiel:</p> <ol style="list-style-type: none"> <li><b>mkdir</b> /MQHA</li> </ol> |
| <p>Stellen Sie sicher, dass Benutzer und Gruppe 'mqm' Eigner der MQHA -Verzeichnisse sind und die Zugriffsberechtigungen für Benutzer und Gruppe auf rwx gesetzt sind. Beispiel: <b>ls -al</b> wird angezeigt.</p> <pre>drwxrwxr-x mqm mqm 4096 Nov 27 14:38 MQDATA</pre> <ol style="list-style-type: none"> <li><b>chown -R</b> mqm:mqm /MQHA</li> <li><b>chmod -R</b> ug+rwx /MQHA</li> </ol>                                                                                                                                                                                                                  |                                                                                                                                                                                                                         |



Tabelle 30. Veranschaulichung der Konfiguration eines Multi-Instanz-Warteschlangenmanagers unter Linux (Forts.)

| Server1                                                                                                                                                                                                                                                                                                                                                                                                                                                  | Server2                                                                                                                                                                                                          |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Erstellen Sie den Warteschlangenmanager:<br><b>crtmqm -ld /MQHA/logs -md /MQHA/qmgrs QM1</b>                                                                                                                                                                                                                                                                                                                                                             |                                                                                                                                                                                                                  |
| Hinzufügen <sup>2</sup> /MQHA *(rw, sync, no_wdelay, fsid=0) zu /etc/exports                                                                                                                                                                                                                                                                                                                                                                             |                                                                                                                                                                                                                  |
| Starten Sie den NFS-Dämon: <b>/etc/init.d/nfs start</b>                                                                                                                                                                                                                                                                                                                                                                                                  | Legen Sie das exportierte Dateisystem /MQHA an:<br><br><b>mount -t nfs4 -o hard,intr Server1:/ /MQHA</b>                                                                                                         |
| Kopieren Sie die Konfigurationsdetails des Warteschlangenmanagers von Server1:<br><br><b>dspmqlnf -o command QM1</b><br><br>und kopieren Sie das Ergebnis in die Zwischenablage:<br><br><b>addmqinf -s QueueManager<br/>-v Name=QM1<br/>-v Directory=QM1<br/>-v Prefix=/var/mqm<br/>-v DataPath=/MQHA/qmgrs/QM1</b>                                                                                                                                      | Fügen Sie den Konfigurationsbefehl des Warteschlangenmanagers in Server2 ein:<br><br><b>addmqinf -s QueueManager<br/>-v Name=QM1<br/>-v Directory=QM1<br/>-v Prefix=/var/mqm<br/>-v DataPath=/MQHA/qmgrs/QM1</b> |
| Starten Sie die Instanzen des Warteschlangenmanagers in beliebiger Reihenfolge mit dem Parameter <b>-x</b> :<br><b>strmqm -x QM1</b>                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                  |
| Der zum Starten der Warteschlangenmanagerinstanzen verwendete Befehl muss von derselben IBM WebSphere MQ-Installation wie der Befehl <b>addmqinf</b> ausgegeben werden. Wenn Sie den Warteschlangenmanager von einer anderen Installation aus starten und stoppen möchten, müssen Sie zuerst die dem Warteschlangenmanager zugeordnete Installation mit dem Befehl <b>setmqm</b> festlegen. Weitere Informationen finden Sie in <a href="#">setmqm</a> . |                                                                                                                                                                                                                  |

#### Multi-Instanz-Warteschlangenmanager unter Linux überprüfen

Verwenden Sie die Beispielprogramme **amqsgbac**, **amqspbac** und **amqsmbac**, um die Konfiguration eines Multi-Instanz-Warteschlangenmanagers zu prüfen. Dieser Abschnitt enthält eine Beispielkonfiguration für die Überprüfung einer Konfiguration mit einem Multi-Instanz-Warteschlangenmanager unter Linux Red Hat Enterprise 5.

Die Beispielprogramme für hohe Verfügbarkeit verwenden die automatische Clientwiederverbindung. Wenn der verbundene Warteschlangenmanager ausfällt, versucht der Client, die Verbindung zu einem WS-Manager in derselben Warteschlangenmanager-Gruppe wieder herzustellen. Die Beschreibung der Beispiele [Hochverfügbarkeits-Musterprogramme](#) veranschaulicht die Clientwiederverbindung unter Verwendung eines einzigen Instanz-WS-Managers für die Einfachheit. Sie können dieselben Muster mit Warteschlangenmanagern mit mehreren Instanzen verwenden, um eine Konfiguration mit mehreren Instanzen des Warteschlangenmanagers zu überprüfen.

Im Beispiel wird die Konfiguration mit mehreren Instanzen verwendet, die in „[Mehrinstanz-Warteschlangenmanager unter Linux erstellen](#)“ auf Seite 415 beschrieben wird. Überprüfen Sie mit der Konfiguration, ob der Warteschlangenmanager mit mehreren Instanzen in die Standby-Instanz umschaltet. Stoppen Sie den WS-Manager mit dem Befehl **endmqm** und verwenden Sie die Option **-s**, Switchover. Die Clientpro-

<sup>2</sup> Der '\*' ermöglicht es allen Maschinen, die diesen einen Mount/MQHA für Lese-/Schreibvorgänge erreichen können. Schränken Sie den Zugriff auf eine Produktionsmaschine ein.

gramme stellen die Verbindung zur neuen Warteschlangenmanagerinstanz wieder her und arbeiten nach einer geringfügigen Verzögerung weiterhin mit der neuen Instanz.

In dem Beispiel wird der Client auf einem Windows XP Service Pack 2-System ausgeführt. Das System dient als Host für zwei VMware Linux-Server, auf denen der Multi-Instanz-Warteschlangenmanager ausgeführt wird.

### Funktionsübernahme mit WebSphere MQ Explorer überprüfen

Bevor Sie die Beispielanwendungen verwenden, um die Funktionsübernahme zu überprüfen, führen Sie den WebSphere MQ Explorer auf jedem Server aus. Fügen Sie jedem Explorer die beiden WS-Manager-Instanzen hinzu, indem Sie den Assistenten **Remote Queue Manager hinzufügen > Direkt mit einem Multi-Instanz-Warteschlangenmanager verbinden** verwenden. Stellen Sie sicher, dass beide Instanzen aktiv sind und den Standby-Modus zulassen. Schließen Sie das Fenster, in dem das VMware-Image ausgeführt wird, mit der aktiven Instanz, den Server virtuell ausschalten oder die aktive Instanz stoppen, um die Umschaltung auf die Standby-Instanz zu ermöglichen.

**Anmerkung:** Wenn Sie den Server ausschalten, stellen Sie sicher, dass es sich nicht um den Server handelt, der /MQHAhostet!

**Anmerkung:** Die Option **Umschalten auf eine Standby-Instanz zulassen** ist unter Umständen nicht im Dialog **Warteschlangenmanager stoppen** verfügbar. Die Option fehlt, weil der WS-Manager als einzelner Instanz-Warteschlangenmanager ausgeführt wird. Sie müssen diese Option ohne die Option **Standby-Instanz zulassen** gestartet haben. Wenn Ihre Anforderung zum Stoppen des Warteschlangenmanagers zurückgewiesen wird, sehen Sie sich das Fenster **Details** an, da möglicherweise keine Standby-Instanz aktiv ist.

### Funktionsübernahme mit den Musterprogrammen überprüfen

#### Wählen Sie einen Server aus, der die aktive Instanz ausführen soll.

Möglicherweise haben Sie einen der Server ausgewählt, um das MQHA-Verzeichnis oder das Dateisystem zu hosten. Wenn Sie die Funktionsübernahme testen möchten, indem Sie das Fenster VMware schließen, in dem der aktive Server ausgeführt wird, stellen Sie sicher, dass es sich nicht um das Hosting von MQHA handelt!

#### Auf dem Server, auf dem die aktive WS-Manager-Instanz ausgeführt

**Anmerkung:** Die Ausführung des SVRCONN -Kanals mit dem Wert mqmfür MCAUSER ist ein komfortabler Schritt, um die Anzahl der Konfigurationsschritte im Beispiel zu reduzieren. Wenn eine andere Benutzer-ID ausgewählt ist und Ihr System anders konfiguriert ist als in dem Beispiel, können Sie Probleme mit der Zugriffsberechtigung haben. Verwenden Sie mqm nicht als MCAUSER auf einem exponierten System; es ist wahrscheinlich, dass die Sicherheit erheblich beeinträchtigt wird.

1. Ändern Sie *ipaddr1* und *ipaddr2* und speichern Sie die folgenden Befehle in /MQHA/hasamples.tst..

```
DEFINE QLOCAL(SOURCE) REPLACE
DEFINE QLOCAL(TARGET) REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
MCAUSER('mqm') REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNAME('ipaddr1(1414),ipaddr2
(1414)') QMNAME(QM1) REPLACE
START CHANNEL(CHANNEL1)
DEFINE LISTENER(LISTENER.TCP) TRPTYPE(TCP) CONTROL(QMGR)
DISPLAY LISTENER(LISTENER.TCP) CONTROL
START LISTENER(LISTENER.TCP)
DISPLAY LSSTATUS(LISTENER.TCP) STATUS
```

2. Öffnen Sie ein Terminalfenster mit dem Pfad /MQHA und führen Sie den folgenden Befehl aus:

```
runmqsc -m QM1 < hasamples.tst
```

3. Stellen Sie sicher, dass der Listener aktiv ist und die Steuerung des Warteschlangenmanagers hat, indem Sie die Ausgabe des Befehls **runmqsc** überprüfen.

```
LISTENER(LISTENER.TCP)CONTROL(QMGR)
LISTENER(LISTENER.TCP)STATUS(RUNNING)
```

Oder verwenden Sie den WebSphere MQ Explorer, um zu erkennen, dass das TCP/IP-Empfangsprogramm aktiv ist und über `Control = Queue Manager` verfügt.

### Auf dem Client

1. Kopieren Sie die Clientverbindungstabelle `AMQCLCHL.TAB` von `/MQHA/qmgrs/QM1.000/@ipcc` auf dem Server auf den `C:\` auf dem Client.
2. Öffnen Sie eine Eingabeaufforderung mit dem Pfad `C:\`, und legen Sie die Umgebungsvariable `MQCHLLIB` so fest, dass sie auf die Definitionstabelle für den Clientkanal (CCDT) verweist.

```
SET MQCHLLIB=C:\
```

3. Geben Sie an der Eingabeaufforderung die folgenden Befehle ein:

```
start amqsghac TARGET QM1
start amqsmhac -s SOURCE -t TARGET -m QM1
start amqsphac SOURCE QM1
```

### Auf dem Server, auf dem die aktive WS-Manager-Instanz ausgeführt

1. Entweder:
  - Schließen Sie das Fenster, in dem das VMware-Image mit der aktiven Serverinstanz ausgeführt wird.
  - Stoppen Sie mithilfe von WebSphere MQ Explorer die aktive Warteschlangenmanagerinstanz, ermöglichen Sie das Umschalten auf die Standby-Instanz und weisen Sie wiederverbindbare Clients an, die Verbindung wiederherzustellen.
2. Die drei Clients erkennen schließlich, dass die Verbindung unterbrochen ist, und stellen Sie dann die Verbindung wieder her. Wenn Sie in dieser Konfiguration das Serverfenster schließen, dauert es ungefähr sieben Minuten, bis alle drei Verbindungen wiederhergestellt sind. Einige Verbindungen werden vor anderen wieder hergestellt.

### Ergebnisse

```
N:\>amqsphac SOURCE QM1
Sample AMQSPHAC start
target queue is SOURCE
message <Message 1>
message <Message 2>
message <Message 3>
message <Message 4>
message <Message 5>
17:05:25 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message <Message 6>
message <Message 7>
message <Message 8>
message <Message 9>
```

```
N:\>amqsmhac -s SOURCE -t TARGET -m QM1
Sample AMQSMHA0 start

17:05:25 : EVENT : Connection Reconnecting (Delay: 97ms)
17:05:48 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:53 : EVENT : Connection Reconnected
```

```

N:\>amqsg hac TARGET QM1
Sample AMQSGHAC start
message <Message 1>
message <Message 2>
message <Message 3>
message <Message 4>
message <Message 5>
17:05:25 : EVENT : Connection Reconnecting (Delay: 156ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message <Message 6>
message <Message 7>
message <Message 8>
message <Message 9>

```

## **Multi-Instanz-WS-Manager löschen**

Um einen Multi-Instanz-Warteschlangenmanager vollständig zu löschen, müssen Sie ihn mit dem Befehl **dltmqm** löschen und anschließend mit den Befehlen **rmvmqinf** oder **dltmqm** Instanzen von anderen Servern entfernen.

Führen Sie den **dltmqm** -Befehl aus, um einen Warteschlangenmanager zu löschen, für den Instanzen auf anderen Servern definiert sind, auf allen Servern, auf denen dieser Warteschlangenmanager definiert ist. Sie müssen den Befehl **dltmqm** nicht auf demselben Server ausführen, auf dem er erstellt wurde. Führen Sie anschließend den Befehl **rmvmqinf** oder **dltmqm** auf allen anderen Servern aus, auf denen der Warteschlangenmanager definiert ist.

Sie können einen WS-Manager nur löschen, wenn er gestoppt wurde. Zu dem Zeitpunkt, zu dem Sie es löschen, werden keine Instanzen ausgeführt, und der Warteschlangenmanager, genau genommen, ist weder ein einzelner noch ein Warteschlangenmanager mit mehreren Instanzen. Es handelt sich lediglich um einen Warteschlangenmanager, der über die zugehörigen WS-Manager-Daten und Protokolle auf einem fernen freigesetzten Verzeichnis verfügt. Wenn Sie einen Warteschlangenmanager löschen, werden seine Warteschlangenmanagerdaten und -Protokolle gelöscht und die Zeilengruppe des Warteschlangenmanagers wird aus der Datei `mq5.ini` auf dem Server entfernt, auf dem Sie den Befehl **dltmqm** ausgegeben haben. Sie müssen Zugriff auf die Netzfreigabe haben, die die Daten und Protokolle des WS-Managers enthält, wenn Sie den Warteschlangenmanager löschen.

Auf anderen Servern, auf denen Sie zuvor Instanzen des Warteschlangenmanagers erstellt haben, befinden sich auch Einträge in den `mq5.ini` -Dateien auf diesen Servern. Sie müssen jeden Server nacheinander besuchen und die Zeilengruppe des Warteschlangenmanagers entfernen, indem Sie den Befehl **rmvmqinf** *Zeilengruppenname des Warteschlangenmanagers* ausführen.

Wenn Sie auf UNIX and Linux-Systemen eine allgemeine `mq5.ini`-Datei in den Netzspeicher gestellt und von allen Servern referenziert haben, indem Sie die Umgebungsvariable `AMQ_MQS_INI_LOCATION` auf jedem Server definieren, müssen Sie den Warteschlangenmanager nur von einem seiner Server löschen, da nur eine `mq5.ini`-Datei zu aktualisieren ist.

### **Beispiel**

#### **Erster Server**

```
dltmqm QM1
```

#### **Andere Server, auf denen Instanzen definiert sind**

```
rmvmqinf QM1, oder
```

```
dltmqm QM1
```

## **Starten und Stoppen eines Multi-Instanz-Warteschlangenmanagers**

Starten und Stoppen eines Warteschlangenmanagers, der als Einzel- oder Multi-Instanz-Warteschlangenmanager konfiguriert ist.

Wenn Sie einen Warteschlangenmanager mit mehreren Instanzen auf einem Serverpaar definiert haben, können Sie den Warteschlangenmanager entweder in einem einzigen Instanz-Warteschlangenmanager oder als WS-Manager mit mehreren Instanzen auf beiden Servern ausführen.

Um einen Warteschlangenmanager mit mehreren Instanzen auszuführen, starten Sie den WS-Manager auf einem der Server mit dem Befehl **strmqm -x QM1** ; die Option **-x** ermöglicht der Instanz die Funktionsübernahme. Sie wird zur *aktiven Instanz*. Starten Sie die Standby-Instanz auf dem anderen Server mit demselben Befehl **strmqm -x QM1** ; die Option **-x** ermöglicht den Start der Instanz als Standby-Instanz.

Der WS-Manager wird jetzt mit einer aktiven Instanz ausgeführt, die alle Anforderungen verarbeitet, und eine Standby-Instanz, die bereit ist, zu übernehmen, wenn die aktive Instanz fehlschlägt. Der aktiven Instanz wird exklusiven Zugriff auf die Daten und Protokolle des Warteschlangenmanagers gewährt. Der Standby-Wartestatus wartet auf exklusiven Zugriff auf die Daten und Protokolle des Warteschlangenmanagers. Wenn der Standby-Server exklusiven Zugriff gewährt wird, wird er zur aktiven Instanz.

Sie können die Steuerung auch manuell zur Standby-Instanz wechseln, indem Sie den Befehl **endmqm -s** für die aktive Instanz absetzen. Der Befehl **endmqm -s** beendet die aktive Instanz, ohne die Bereitschaftsdatenbank herunterzufahren. Die exklusive Zugriffssperre für die WS-Manager-Daten und -Protokolle wird freigegeben, und die Bereitschaftsdatenbank übernimmt die Verwaltung.

Sie können auch einen Warteschlangenmanager starten und stoppen, der mit mehreren Instanzen auf verschiedenen Servern als einzelner Instanz-WS-Manager konfiguriert ist. Wenn Sie den WS-Manager ohne die Option **-x** im Befehl **strmqm** starten, werden die Instanzen des Warteschlangenmanagers, die auf anderen Maschinen konfiguriert sind, daran gehindert, als Standby-Instanzen zu starten. Wenn Sie versuchen, eine andere Instanz zu starten, erhalten Sie die Antwort, dass die WS-Manager-Instanz nicht als Standby-Instanz ausgeführt werden darf.

Wenn Sie die aktive Instanz eines Multi-Instanz-Warteschlangenmanagers mit dem Befehl **endmqm** ohne die Option **-s** stoppen, werden die aktiven Instanzen und die Standby-Instanzen gestoppt. Wenn Sie die Standby-Instanz mit dem Befehl **endmqm** mit der Option **-x** stoppen, wird sie als Standby-Instanz gestoppt und die aktive Instanz wird weiter ausgeführt. Sie können **endmqm** nicht ohne die Option **-x** in der Bereitschaftsdatenbank absetzen.

Es können nur zwei WS-Manager-Instanzen gleichzeitig ausgeführt werden. Eine Instanz ist die aktive Instanz, und die andere Instanz ist eine Standby-Instanz. Wenn Sie zwei Instanzen gleichzeitig starten, hat WebSphere MQ keine Kontrolle darüber, welche Instanz zur aktiven Instanz wird; sie wird vom Netzdateisystem bestimmt. Die erste Instanz, die exklusiven Zugriff auf die WS-Manager-Daten erhält, wird zum aktiven Exemplar.

**Anmerkung:** Bevor Sie einen fehlgeschlagenen Warteschlangenmanager erneut starten, müssen Sie die Verbindung zu den Anwendungen von dieser Instanz des Warteschlangenmanagers trennen. Ist dies nicht der Fall, wird der WS-Manager möglicherweise nicht ordnungsgemäß erneut gestartet.

### ***Gemeinsam genutztes Dateisystem***

Ein Multi-Instanz-Warteschlangenmanager verwendet zur Verwaltung von Warteschlangenmanagerinstanzen ein Netzdateisystem.

Ein Warteschlangenmanager mit mehreren Instanzen automatisiert die Funktionsübernahme mit einer Kombination aus Dateisystemsperrern und gemeinsam genutzten WS-Manager-Daten und -Protokollen. Nur eine Instanz eines Warteschlangenmanagers kann exklusiven Zugriff auf die Daten und Protokolle des gemeinsam genutzten Warteschlangenmanagers haben. Wenn er Zugriff erhält, wird er zur aktiven Instanz. Die andere Instanz, die den exklusiven Zugriff nicht erhält, wartet als Standby-Instanz, bis die WS-Manager-Daten und -Protokolle verfügbar werden.

Das Netzdateisystem ist für die Freigabe der Sperren verantwortlich, die er für die aktive Warteschlangenmanagerinstanz enthält. Wenn die aktive Instanz in irgendeiner Weise ausfällt, gibt das vernetzte Dateisystem die Sperren frei, die sie für die aktive Instanz hält. Sobald die exklusive Sperre freigegeben wird, wartet ein Standby-Warteschlangenmanager, der auf die Sperre wartet, um die Sperre zu erhalten. Ist sie erfolgreich, wird sie zur aktiven Instanz und verfügt über exklusiven Zugriff auf die Daten und Protokolle des Warteschlangenmanagers und die Protokolle im gemeinsam genutzten Dateisystem. Anschließend wird die Verarbeitung fortgesetzt.

Das zugehörige Thema *Unterstützung der Planungsdateisysteme* beschreibt, wie Sie konfigurieren und überprüfen, ob Ihr Dateisystem mehrere Instanzen von Warteschlangenmanagern unterstützt.

Ein WS-Manager mit mehreren Instanzen schützt Sie nicht vor einem Fehler im Dateisystem. Es gibt eine Reihe von Möglichkeiten, Ihre Daten zu schützen.

- Investieren Sie in einen zuverlässigen Speicher, wie z. B. redundante Platteneinheiten (RAID), und schließen Sie sie in ein Netzdateisystem ein, das über Netzausfallsicherheit verfügt.
- Sichern Sie lineare WebSphere MQ -Protokolle auf alternativen Datenträgern. Wenn Ihr primärer Protokolldatenträger ausfällt, stellen Sie die Protokolle auf den alternativen Datenträgern wieder her. Sie können einen Sicherungswarteschlangenmanager verwenden, um diesen Prozess zu verwalten.

### **Mehrere WS-Manager-Instanzen**

Ein Warteschlangenmanager mit mehreren Instanzen ist belastbar, da er eine Standby-WS-Manager-Instanz verwendet, um die Verfügbarkeit des Warteschlangenmanagers nach einem Fehler wiederherzustellen.

Das Replizieren von Warteschlangenmanagerinstanzen ist eine sehr effektive Möglichkeit, die Verfügbarkeit von Warteschlangenmanagerprozessen zu verbessern. Verwenden Sie ein einfaches Verfügbarkeitsmodell, nur zur Veranschaulichung: Wenn die Zuverlässigkeit einer Instanz eines Warteschlangenmanagers 99% beträgt (über ein Jahr beträgt die kumulative Ausfallzeit 3,65 Tage), erhöht das Hinzufügen einer weiteren Instanz des Warteschlangenmanagers die Verfügbarkeit auf 99,99% (über ein Jahr, kumulative Ausfallzeit von etwa einer Stunde).

Dies ist zu einfach ein Modell, um Ihnen praktische numerische Schätzungen der Verfügbarkeit zu geben. Um die Verfügbarkeit realistisch zu modellieren, müssen Sie statistische Daten für die mittlere Zeit zwischen Fehlern (MTBF) und die mittlere Reparaturzeit (MTTR) und die Wahrscheinlichkeitsverteilung der Zeit zwischen Ausfällen und Reparaturzeiten erfassen.

Der Begriff Multi-Instanz-WS-Manager bezeichnet die Kombination aus aktiven und Standby-Instanzen des Warteschlangenmanagers, die die Daten und Protokolle des Warteschlangenmanagers gemeinsam nutzen. Warteschlangenmanager mit mehreren Instanzen schützen Sie vor dem Ausfall von WS-Manager-Prozessen, indem eine Instanz des auf einem Server aktiven WS-Managers und eine weitere Instanz des Warteschlangenmanagers auf einem anderen Server auf einem anderen Server aktiv ist. Diese Instanz kann automatisch übernommen werden, wenn die aktive Instanz fehlschlägt.

### **Failover oder Switchover**

Eine Standby-WS-Manager-Instanz übernimmt die aktive Instanz entweder auf Anforderung (Switchover) oder wenn die aktive Instanz ausfällt (Failover).

- *Switchover* findet statt, wenn eine Standby-Instanz als Antwort auf den Befehl **endmqm -s** gestartet wird, der an die aktive Warteschlangenmanagerinstanz ausgegeben wird. Sie können die **endmqm** -Parameter **-c**, **-i** oder **-p** angeben, um zu steuern, wie der Warteschlangenmanager abrupt gestoppt wird.

**Anmerkung:** Switchover findet nur statt, wenn bereits eine Standby-Warteschlangenmanagerinstanz gestartet wurde. Der Befehl **endmqm -s** gibt die Sperre des aktiven Warteschlangenmanagers frei und lässt das Umschalten zu: Er startet keine Standby-Warteschlangenmanagerinstanz.

- *Failover* tritt auf, wenn die Sperre für Warteschlangenmanagerdaten, die von der aktiven Instanz gehalten werden, freigegeben wird, weil die Instanz anscheinend unerwartet gestoppt wurde (d. h., ohne dass ein Befehl **endmqm** ausgegeben wurde).

Wenn die Standby-Instanz als aktive Instanz übernimmt, schreibt sie eine Nachricht in das Fehlerprotokoll des Warteschlangenmanagers.

Bei einem Ausfall oder Umschalten eines WS-Managers werden die wiederverbindbaren Clients automatisch wieder verbunden. Sie müssen das Flag **-x** nicht in den Befehl **endmqm** einschließen, um die Clientverbindungswiederholung anzufordern. Die automatische Wiederherstellung einer Clientverbindung wird von WebSphere MQ Classes for Java nicht unterstützt.

Wenn Sie feststellen, dass Sie eine fehlgeschlagene Instanz nicht erneut starten können, obwohl ein Failover stattgefunden hat und die Standby-Instanz aktiv geworden ist, überprüfen Sie, ob Anwendungen, die lokal mit der fehlgeschlagenen Instanz verbunden sind, die Verbindung zur fehlgeschlagenen Instanz nicht getrennt haben. Lokal verbundene Anwendungen beenden oder trennen eine fehlgeschlagene WS-Manager-Instanz, um sicherzustellen, dass die fehlgeschlagene Instanz erneut gestartet werden kann.

Alle lokal verbundenen Anwendungen, die gemeinsam genutzte Bindungen verwenden (dies ist die Standardeinstellung), die eine Verbindung zu einer fehlgeschlagenen Instanz enthalten, um zu verhindern, dass die Instanz erneut gestartet wird. Wenn es nicht möglich ist, die lokal verbundenen Anwendungen zu beenden, oder stellen Sie sicher, dass die Verbindung getrennt wird, wenn die lokale Warteschlangenmanagerinstanz fehlschlägt, sollten Sie die Verwendung isolierter Bindungen in Betracht ziehen. Lokal verbundene Anwendungen, die isolierte Bindungen verwenden, verhindern nicht, dass die lokale WS-Manager-Instanz erneut gestartet wird, auch wenn sie nicht getrennt werden.

### **Kanal- und Clientverbindungswiederholung**

Die Verbindung zwischen Kanal und Client ist ein wesentlicher Bestandteil der Wiederherstellung der Nachrichtenverarbeitung, nachdem eine Standby-Warteschlangenmanagerinstanz aktiv geworden ist.

Instanzen von Warteschlangenmanagern mit mehreren Instanzen werden auf Servern mit unterschiedlichen Netzadressen installiert. Sie müssen IBM WebSphere MQ-Kanäle und -Clients mit Verbindungsinformationen für alle Warteschlangenmanagerinstanzen konfigurieren. Wenn eine Bereitschaftsdatenbank übernommen wird, werden Clients und Kanäle automatisch mit der neu aktiven WS-Manager-Instanz an der neuen Netzadresse verbunden. Die automatische Wiederherstellung einer Clientverbindung wird von WebSphere MQ Classes for Java nicht unterstützt.

Das Design unterscheidet sich von der Art und Weise, wie Hochverfügbarkeitsumgebungen, wie z. B. HA-CMP, funktionieren. HA-CMP stellt eine virtuelle IP-Adresse für den Cluster bereit und überträgt die Adresse an den aktiven Server. WebSphere Die MQ -Verbindungswiederholung ändert keine IP-Adressen oder leitet sie nicht um. Es funktioniert, indem die Verbindung unter Verwendung der Netzwerkadressen, die Sie in Kanaldefinitionen und Clientverbindungen definiert haben, wieder hergestellt wird. Als Administrator müssen Sie die Netzadressen in Kanaldefinitionen und Clientverbindungen zu allen Instanzen eines beliebigen Multi-Instanz-Warteschlangenmanagers definieren. Die beste Möglichkeit, Netzadressen für einen Warteschlangenmanager mit mehreren Instanzen zu konfigurieren, hängt von der Verbindung ab:

#### **WS-Manager-Kanäle**

Das Attribut CONNAME von Kanälen ist eine durch Kommas getrennte Liste von Verbindungsnamen, z. B. CONNAME ( ' 127 . 0 . 0 . 1 (1234) , 192 . 0 . 2 . 0 (4321) ' ). Die Verbindungen werden in der Reihenfolge versucht, die in der Verbindungsliste angegeben ist, bis eine Verbindung erfolgreich hergestellt wurde. Wenn keine Verbindung erfolgreich hergestellt wurde, versucht der Kanal, die Verbindung herzustellen.

#### **Clusterkanäle**

In der Regel ist keine zusätzliche Konfiguration erforderlich, um WS-Manager mit mehreren Instanzen in einem Cluster zu bearbeiten.

Wenn ein Warteschlangenmanager eine Verbindung zu einem Repository-WS-Manager herstellt, erkennt das Repository die Netzadresse des Warteschlangenmanagers. Er bezieht sich auf den CONNAME des CLUSRCVR -Kanals auf dem Warteschlangenmanager. Bei TCPIP setzt der Warteschlangenmanager den CONNAME automatisch, wenn Sie ihn weglassen oder ihn in Leerzeichen konfigurieren. Wenn eine Standby-Instanz übernimmt, ersetzt die IP-Adresse der vorherigen aktiven Instanz die IP-Adresse der vorherigen aktiven Instanz als CONNAME .

Wenn dies erforderlich ist, können Sie CONNAME manuell mit der Liste der Netzadressen der Warteschlangenmanagerinstanzen konfigurieren.

#### **Clientverbindungen**

Clientverbindungen können Verbindungslisten oder Warteschlangenmanagergruppen verwenden, um alternative Verbindungen auszuwählen. Weitere Informationen zur Clientverbindungswiederholung zu einem Warteschlangenmanager mit mehreren Instanzen finden Sie unter „[Automatische Clientverbindungswiederholung](#)“ auf Seite 336. Clients müssen kompiliert werden, damit sie mit Clientbibliotheken von WebSphere MQ Version 7.0.1 oder höher ausgeführt werden können. Sie müssen mit mindestens einem Warteschlangenmanager der Version 7.0.1 verbunden sein.

Wenn eine Funktionsübernahme erfolgt, nimmt die Verbindungswiederverbindung einige Zeit in Anspruch. Der Standby-WS-Manager muss den Start beenden. Die Clients, die mit dem fehlgeschlagenen Warteschlangenmanager verbunden waren, müssen den Verbindungsfehler feststellen und eine neue

Clientverbindung starten. Wenn eine neue Clientverbindung den Standby-WS-Manager auswählt, der neu aktiv ist, wird der Client erneut mit demselben Warteschlangenmanager verbunden.

Wenn sich der Client während der Verbindungswiederverbindung in der Mitte eines MQI-Aufrufs befindet, muss er eine erweiterte Wartezeit tolerieren, bevor der Aufruf abgeschlossen wird.

Wenn der Fehler während einer Stapelübertragung auf einem Nachrichtenkanal stattfindet, wird die Stapelverarbeitung rückgängig gemacht und erneut gestartet.

Das Umschalten ist schneller als ein fehlerhaftes Umschalten und dauert nur so lange, wie eine Instanz des Warteschlangenmanagers gestoppt und eine andere Instanz gestartet wird. Bei einem WS-Manager mit nur wenigen Protokollsätzen, die wiedergegeben werden sollen, kann bei bester Umschaltung die Reihenfolge einiger Sekunden in Anspruch nehmen. Um zu schätzen, wie lange die Funktionsübernahme dauert, müssen Sie die Zeit hinzufügen, die für das Erkennen des Fehlers benötigt wird. Am besten ist die Erkennung in der Größenordnung von 10 Sekunden, und kann mehrere Minuten, je nach Netzwerk und Dateisystem.

### **Anwendungswiederherstellung**

Die Anwendungswiederherstellung ist die automatisierte Fortsetzung der Anwendungsverarbeitung nach dem Failover. Die Anwendungswiederherstellung nach dem Failover erfordert sorgfältige Konstruktion. Einige Anwendungen müssen ein Failover-Failover durchgeführt haben.

Das Ziel der Anwendungswiederherstellung ist es, dass die Anwendung die Verarbeitung mit nur einer kurzen Verzögerung fortsetzt. Bevor Sie mit der neuen Verarbeitung fortfahren, muss die Anwendung die Arbeitseinheit, die sie während des Fehlers verarbeitet hat, wieder zurücksenden und erneut übergeben.

Ein Problem bei der Anwendungswiederherstellung besteht darin, den Kontext zu verlieren, der vom WebSphere MQ MQI-Client und dem Warteschlangenmanager gemeinsam genutzt und im Warteschlangenmanager gespeichert wird. Der WebSphere MQ-Client stellt den Großteil des Kontexts wieder her. Allerdings gibt es einige Teile des Kontexts, die nicht zuverlässig wiederhergestellt werden können. In den folgenden Abschnitten werden einige Merkmale der Anwendungswiederherstellung beschrieben und die Auswirkungen auf die Wiederherstellung von Anwendungen, die mit einem Warteschlangenmanager mit mehreren Instanzen verbunden sind, beeinflusst.

### **Transaktionsorientiertes Messaging**

Aus Sicht der Nachrichtenübermittlung ändert ein Failover nicht die persistenten Eigenschaften des Messaging in WebSphere MQ. Wenn Nachrichten persistent sind und ordnungsgemäß in Arbeitseinheiten verwaltet werden, gehen die Nachrichten während einer Funktionsübernahme nicht verloren.

Aus der Perspektive der Transaktionsverarbeitung werden Transaktionen nach einem Failover entweder zurückgesetzt oder festgeschrieben.

Nicht festgeschriebene Transaktionen werden rückgängig gemacht. Nach einem Failover erhält eine Anwendung, deren Verbindung wiederhergestellt werden kann, einen MQRC\_BACKED\_OUT-Ursachencode, um die fehlgeschlagene Transaktion anzuzeigen, und die Transaktion muss durch Ausgeben von MQBACK zurückgesetzt werden. Anschließend muss die Transaktion erneut gestartet werden.

Festgeschriebene Transaktionen sind Transaktionen, die die zweite Phase der zweiphasigen Festschreibung erreicht haben, oder Transaktionen mit einer einzigen Phase (nur Nachrichten), die MQCMIT gestartet haben.

Wenn der Warteschlangenmanager der Transaktionskoordinator ist und MQCMIT die zweite Phase der zweiphasigen Festschreibung vor dem Fehler begonnen hat, wird die Transaktion erfolgreich abgeschlossen. Die Fertigstellung wird unter der Steuerung des Warteschlangenmanagers ausgeführt und wird fortgesetzt, wenn der Warteschlangenmanager erneut ausgeführt wird. In einer wiederverbindbaren Anwendung wird der Aufruf MQCMIT normal beendet.

In einer einphasigen Festschreibung, die nur Nachrichten umfasst, wird eine Transaktion, die die COMMIT-Verarbeitung gestartet hat, normalerweise unter der Steuerung des Warteschlangenmanagers abgeschlossen, sobald sie erneut ausgeführt wird. In einer wiederverbindbaren Anwendung wird MQCMIT normal beendet.



Wiederanschlussfähige Clients können unter der Steuerung des Warteschlangenmanagers als Transaktionskoordinator einzelne Phasentransaktionen verwenden. Der erweiterte transaktionsorientierte Client unterstützt keine erneute Verbindung. Wenn eine erneute Verbindung angefordert wird, wenn der transaktionsorientierte Client eine Verbindung herstellt, ist die Verbindung erfolgreich, aber ohne dass die Verbindung erneut hergestellt werden kann. Die Verbindung verhält sich so, als ob sie nicht wieder angeschlossen werden kann.

## Anwendungsneustart oder -wiederaufnehmen

Failover unterbricht eine Anwendung. Nach einem Fehler kann eine Anwendung von Anfang an erneut gestartet werden, oder sie kann die Verarbeitung nach der Unterbrechung wieder aufnehmen. Letzteres wird als *automatische Clientwiederverbindung* bezeichnet. Die automatische Wiederherstellung einer Clientverbindung wird von WebSphere MQ Classes for Java nicht unterstützt.

Mit einer WebSphere MQ-Clientanwendung können Sie über eine Verbindungsoption festlegen, dass der Client automatisch die Verbindung wiederherstellt. Diese Optionen sind MQCNO\_RECONNECT oder MQCNO\_RECONNECT\_Q\_MGR. Wenn keine Option festgelegt ist, versucht der Client nicht, die Verbindung automatisch wieder herzustellen, und der Fehler des Warteschlangenmanagers MQRC\_CONNECTI-ON\_BROKEN wird an den Client zurückgegeben. Sie können den Client so gestalten, dass er versucht, eine neue Verbindung zu starten, indem Sie einen neuen Aufruf MQCONN oder MQCONNX absetzen.

Serverprogramme müssen erneut gestartet werden. Sie können nicht automatisch durch den Warteschlangenmanager an dem Punkt, an dem sie verarbeitet wurden, wieder verbunden werden, wenn der Warteschlangenmanager oder der Server fehlgeschlagen ist. WebSphere MQ -Serverprogramme werden in der Standby-Warteschlangenmanagerinstanz normalerweise nicht erneut gestartet, wenn eine Instanz eines Multi-Instanz-Warteschlangenmanagers fehlschlägt.

Sie können ein WebSphere MQ-Serverprogramm automatisieren, damit es auf dem Standby-Server auf zwei Arten erneut startet:

1. Packen Sie Ihre Serveranwendung als WS-Manager-Service. Sie wird erneut gestartet, wenn der Standby-WS-Manager erneut gestartet wird.
2. Schreiben Sie Ihre eigene Failoverlogik, die beispielsweise ausgelöst wird, wenn die Failover-Protokollnachricht von einer Standby-WS-Manager-Instanz geschrieben wird, wenn sie gestartet wird. Die Anwendungsinstanz muss dann MQCONN oder MQCONNX aufrufen, nachdem sie gestartet wurde, um eine Verbindung zum Warteschlangenmanager herzustellen.

## Failover wird ermittelt

Einige Anwendungen müssen sich der Funktionsübernahme bewusst sein, andere nicht. Betrachten Sie diese beiden Beispiele.

1. Eine Messaging-Anwendung, die Nachrichten über einen Nachrichtenübertragungskanal empfängt oder empfängt, erfordert normalerweise nicht, dass der Warteschlangenmanager am anderen Ende des Kanals aktiv ist: Es ist unwahrscheinlich, dass es betroffen ist, wenn der Warteschlangenmanager am anderen Ende des Kanals auf einer Standby-Instanz erneut gestartet wird.
2. Eine WebSphere MQ MQI-Clientanwendung verarbeitet persistente Nachrichteneingaben aus einer Warteschlange und reiht persistente Nachrichtenantworten als Teil einer einzelnen Arbeitseinheit in eine andere Warteschlange ein: Wenn sie einen MQRC\_BACKED\_OUT -Ursachencode von MQPUT, MQGET oder MQCMIT innerhalb eines Synchronisationspunkts verarbeitet, indem sie MQBACK ausgibt und die Arbeitseinheit erneut startet, gehen keine Nachrichten verloren. Darüber hinaus muss die Anwendung keine spezielle Verarbeitung ausführen, um einen Verbindungsfehler zu bewältigen.

Nehmen Sie jedoch im zweiten Beispiel an, dass die Anwendung die Warteschlange durchsucht, um die zu verarbeitende Nachricht mit der Option MQGET , MQGMO\_MSG\_UNDER\_CURSOR, auszuwählen. Die Verbindungswiederholung setzt den Anzeigecursor zurück und der Aufruf MQGET gibt nicht die richtige Nachricht zurück. In diesem Beispiel muss die Anwendung eine Übernahme durch eine Funktionsübernahme (Failover) durchgeführt werden. Außerdem muss die Anwendung vor der Ausgabe einer weiteren MQGET für die Nachricht unter dem Cursor den Anzeigecursor wiederherstellen.

Das Sperren des Anzeigecursors ist ein Beispiel dafür, wie der Anwendungskontext nach der Verbindungswiederverbindung geändert wird. Weitere Fälle sind in „Wiederherstellung eines automatisch verbundenen Clients“ auf Seite 426 dokumentiert.

Es stehen nach einem Failover drei alternative Designmuster für WebSphere MQ-Clientanwendungen zur Verfügung. Nur bei einem von ihnen muss das Failover nicht erkannt werden.

### **Keine Verbindungswiederverbindung**

In diesem Muster stoppt die Anwendung die gesamte Verarbeitung für die aktuelle Verbindung, wenn die Verbindung unterbrochen ist. Damit die Anwendung die Verarbeitung fortsetzen kann, muss sie eine neue Verbindung zum WS-Manager herstellen. Die Anwendung ist vollständig für die Übertragung aller erforderlichen Statusinformationen verantwortlich, um die Verarbeitung für die neue Verbindung fortzusetzen. Vorhandene Clientanwendungen, die nach dem Verlust ihrer Verbindung die Verbindung zu einem WS-Manager herstellen, werden auf diese Weise geschrieben.

Der Client empfängt einen Ursachencode, z. B. MQRC\_CONNECTION\_BROKEN oder MQRC\_Q\_MGR\_NOT\_AVAILABLE, vom nächsten MQI-Aufruf, nachdem die Verbindung verloren gegangen ist. Die Anwendung muss alle ihre WebSphere MQ-Statusinformationen, z. B. Warteschlangenkennungen, löschen und einen neuen MQCONN- oder MQCONNX-Aufruf ausgeben, um eine neue Verbindung herzustellen; anschließend muss die Anwendung die WebSphere MQ-Objekte öffnen, die von ihr verarbeitet werden müssen.

Das standardmäßige MQI-Verhalten ist, dass die WS-Manager-Verbindungskennung nicht mehr verwendbar wird, nachdem eine Verbindung zum Warteschlangenmanager verloren gegangen ist. Der Standardwert entspricht der Einstellung der Option MQCNO\_RECONNECT\_DISABLED auf MQCONNX, um die Wiederherstellung der Anwendungsverbindung nach einem Failover zu verhindern.

### **Failover-tolerant**

Schreiben Sie die Anwendung so, dass sie von einem Failover nicht betroffen ist. Manchmal ist eine sorgfältige Fehlerbehandlung ausreichend, um das Failover zu bewältigen.

### **Verbindungskonnektion**

Registrieren Sie einen Ereignishandler MQCBT\_EVENT\_HANDLER mit dem Warteschlangenmanager. Der Ereignishandler wird mit MQRC\_RECONNECTING übergeben, wenn der Client versucht, die Verbindung zum Server wiederherzustellen, und mit MQRC\_RECONNECTED, nachdem die Verbindung erfolgreich wiederhergestellt wurde. Anschließend können Sie eine Routine ausführen, um einen vorhersehbaren Status wieder herzustellen, so dass die Clientanwendung die Verarbeitung fortsetzen kann.

## **Wiederherstellung eines automatisch verbundenen Clients**

Der Failover ist ein unerwartetes Ereignis, und für einen automatisch verbundenen Client, der die Auswirkungen der Verbindungswiederaufschaltung hat, müssen die Auswirkungen vorhersehbar sein.

Ein Hauptelement, bei dem ein unerwarteter Fehler in eine vorhersehbare und zuverlässige Wiederherstellung verwandelt wird, ist die Verwendung von Transaktionen.

Im vorherigen Abschnitt wurde ein Beispiel, „2“ auf Seite 425, für einen WebSphere MQ MQI-Client angegeben, der eine lokale Transaktion zur Koordination von MQGET und MQPUT verwendet. Der Client gibt einen MQCMIT -oder MQBACK -Aufruf als Antwort auf einen Fehler MQRC\_BACKED\_OUT aus und übergibt die zurückgesetzte Transaktion erneut. Der Warteschlangenmanager-Fehler führt dazu, dass die Transaktion zurückgesetzt wird, und das Verhalten der Clientanwendung stellt sicher, dass keine Transaktionen ausgeführt werden und keine Nachrichten verloren gehen.

Nicht der gesamte Programmstatus wird als Teil einer Transaktion verwaltet, und daher werden die Auswirkungen der Neuverbindung schwerer zu verstehen. Sie müssen wissen, wie eine Verbindungswiederholung den Status eines WebSphere MQ-Clients ändert, um Ihre Clientanwendung so zu gestalten, dass sie ein Warteschlangenmanager-Failover übersteht.

Sie können Ihre Anwendung ohne speziellen Failover-Code entwerfen, da die Verbindungsfehler mit der gleichen Logik wie andere Fehler behandelt werden. Alternativ dazu können Sie auch in Erwägung ziehen, dass für die Verbindungswiederholung eine spezielle Fehlerbehandlung erforderlich ist, und einen Ereignis-

nishandler in WebSphere MQ eintragen, um eine Routine zur Handhabung des Failovers auszuführen. Die Routine kann die Verarbeitung der Verbindungswiederherstellung selbst verarbeiten oder eine Markierung setzen, um dem Hauptprogrammthread anzuzeigen, dass die Verarbeitung wiederaufgenommen werden muss, wenn die Verarbeitung wieder aufgenommen werden muss.

Die WebSphere MQ-Clientumgebung erkennt das Failover und stellt nach einer Verbindungswiederholung so viel Kontext wie möglich wieder her, indem einige Statusinformationen im Client gespeichert werden und zusätzliche MQI-Aufrufe im Auftrag der Clientanwendung ausgegeben werden, um ihren WebSphere MQ-Status wiederherzustellen. Zum Beispiel werden die Kennungen für Objekte, die am Point of Failure geöffnet waren, wiederhergestellt, und temporäre dynamische Warteschlangen werden mit demselben Namen geöffnet. Es gibt jedoch Änderungen, die unvermeidlich sind, und Sie brauchen Ihr Design, um mit diesen Änderungen umzugehen. Die Änderungen lassen sich in fünf Arten kategorisieren:

1. Neue oder zuvor nicht diagnostizierte Fehler werden von MQI-Aufrufen zurückgegeben, bis ein konsistenter neuer Kontextstatus durch das Anwendungsprogramm wiederhergestellt wird.

Ein Beispiel für den Empfang eines neuen Fehlers ist der Rückkehrcode MQRC\_CONTEXT\_NOT\_AVAILABLE, wenn versucht wird, Kontext nach dem Speichern des Kontexts vor der Verbindungswiederherstellung zu übergeben. Der Kontext kann nach der Neuverbindung nicht wiederhergestellt werden, da der Sicherheitskontext nicht an ein nicht berechtigtes Clientprogramm übergeben wird. Zu diesem Zweck würde ein zerstörerliches Anwendungsprogramm den Sicherheitskontext abrufen.

In der Regel bearbeiten Anwendungen häufig auftretende und vorhersehbare Fehler sorgfältig und führen unübliche Fehler zu einem generischen Fehlerbehandlungsprogramm zurück. Die Fehlerbehandlungsroutine kann die Verbindung zu WebSphere MQ trennen und wiederherstellen oder sogar das Programm insgesamt stoppen. Zur Verbesserung der Kontinuität könnte es erforderlich sein, einige Fehler auf eine andere Weise zu handhaben.

2. Nicht persistente Nachrichten gehen möglicherweise verloren.
3. Transaktionen werden rückgängig gemacht.
4. MQGET -oder MQPUT -Aufrufe, die außerhalb eines Synchronisationspunkts verwendet werden, werden möglicherweise mit dem Verlust einer Nachricht unterbrochen.
5. Timing-induzierter Fehler, aufgrund eines längeren Wartestatus in einem MQI-Aufruf.

Einige Details zum verlorenen Kontext werden im folgenden Abschnitt aufgelistet.

- Nicht persistente Nachrichten werden gelöscht, es sei denn, sie werden mit der Option NPMCLASS (HIGH) in eine Warteschlange gestellt, und der Fehler des Warteschlangenmanagers hat die Option zum Speichern nicht persistenter Nachrichten beim Herunterfahren nicht unterbrochen.
- Eine nicht permanente Subskription geht verloren, wenn eine Verbindung unterbrochen wird. Bei der Neuverbindung wird sie erneut aufgebaut. Ziehen Sie die Verwendung einer permanenten Subskription in Betracht.
- Das Intervall für den Abwartestatus wird neu berechnet. Wenn der Grenzwert überschritten wird, wird MQRC\_NO\_MSG\_AVAILABLE zurückgegeben. Analog wird die Subskriptionsverfallszeit so neu berechnet, dass sie die gleiche Gesamtverfallszeit erhält.
- Die Position des Suchcursors in einer Warteschlange ist verloren; sie wird in der Regel vor der ersten Nachricht neu erstellt.
  - MQGET -Aufrufe, die MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR oder MQGMO\_MSG\_UNDER\_CURSOR angeben, schlagen mit dem Ursachencode MQRC\_NO\_MSG\_AVAILABLE fehl.
  - Nachrichten, die zum Durchsuchen gesperrt sind, werden entsperrt
  - Durchsuchbare Nachrichten mit dem Geltungsbereich der Handle sind nicht markiert und können erneut durchsucht werden.
  - In den meisten Fällen sind die markierten Nachrichten nicht markiert.
- Der Sicherheitskontext ist verloren. Der Versuch, den gespeicherten Nachrichtenkontext zu verwenden, wie z. B. das Einreihen einer Nachricht mit MQPMO\_PASS\_ALL\_CONTEXT, schlägt mit MQRC\_CONTEXT\_NOT\_AVAILABLE fehl.

- Nachrichtentoken gehen verloren. Beim Verwenden von MQGET mit einem Nachrichtentoken wird der Ursachencode MQRC\_NO\_MSG\_AVAILABLE zurückgegeben.

**Anmerkung:** *MsgId* und *CorrelId*, da sie Teil der Nachricht sind, werden während der Funktionsübernahme mit der Nachricht beibehalten, sodass die MQGET Verwendung von *MsgId* oder *CorrelId* wie erwartet funktioniert.

- Nachrichten, die in eine Warteschlange unter Synchronisationspunkt in einer nicht festgeschriebenen Transaktion gestellt werden, sind nicht mehr verfügbar.
- Die Verarbeitung von Nachrichten in einer logischen Reihenfolge oder in einer Nachrichtengruppe führt zu einem Rückkehrcode von MQRC\_RECONNECT\_INCOMPATIBLE nach der Verbindungswiederverbindung.
- Ein MQI-Aufruf gibt möglicherweise MQRC\_RECONNECT\_FAILED zurück und nicht die allgemeinere MQRC\_CONNECTION\_BROKEN, die Clients normalerweise heute empfangen.
- Eine Verbindungswiederholung während eines MQPUT-Aufrufs außerhalb eines Synchronisationspunkts gibt MQRC\_CALL\_INTERRUPTED zurück, wenn der WebSphere MQ-Client nicht weiß, ob die Nachricht erfolgreich an den Warteschlangenmanager übermittelt wurde. Die Verbindungswiederherstellung während MQCMIT verhält sich ähnlich.
- MQRC\_CALL\_INTERRUPTED wird zurückgegeben - nach einer erfolgreichen Verbindungswiederholung -, wenn der WebSphere MQ-Client keine Antworten vom Warteschlangenmanager erhalten hat, um den Erfolg oder Fehlschlag der folgenden Aktionen anzuzeigen:
  - die Zustellung einer persistenten Nachricht unter Verwendung eines MQPUT-Aufrufs außerhalb des Synchronisationspunkts.
  - die Zustellung einer persistenten Nachricht oder einer Nachricht mit Standardpersistenz unter Verwendung eines MQPUT1-Aufrufs außerhalb des Synchronisationspunkts.
  - Festschreiben einer Transaktion mit Hilfe eines MQCMIT-Aufrufs. Die Antwort wird nur nach einer erfolgreichen Verbindungswiederverbindung zurückgegeben.
- Kanäle werden als neue Instanzen erneut gestartet (sie können auch unterschiedliche Kanäle sein), sodass kein Kanalexitstatus beibehalten wird.
- Temporäre dynamische Warteschlangen werden als Teil des Prozesses zur Wiederherstellung wiederverbindbarer Clients, die temporäre dynamische Warteschlangen geöffnet haben, wiederhergestellt. Es werden keine Nachrichten in einer temporären dynamischen Warteschlange zurückgespeichert, aber Anwendungen, bei denen die Warteschlange geöffnet war oder sich an den Namen der Warteschlange erinnern haben, können die Verarbeitung fortsetzen.

Es besteht die Möglichkeit, dass, wenn die Warteschlange von einer anderen Anwendung als der erstellt wird, die sie erstellt hat, sie möglicherweise nicht schnell genug zurückgeschrieben wird, wenn sie als nächste Referenz angegeben wird. Wenn ein Client beispielsweise eine temporäre dynamische Warteschlange als Empfangswarteschlange für Antworten erstellt und eine Antwortnachricht von einem Kanal in die Warteschlange gestellt werden soll, wird die Warteschlange möglicherweise nicht in der Zeit wiederhergestellt. In diesem Fall würde der Kanal normalerweise die Antwort-Nachricht in die Warteschlange für nicht zustellbare Nachrichten stellen.

Wenn eine Clientanwendung, deren Verbindung wiederhergestellt werden kann, eine temporäre dynamische Warteschlange über den Namen öffnet (weil sie bereits von einer anderen Anwendung erstellt wurde), dann kann der WebSphere MQ-Client die temporäre dynamische Warteschlange im Falle einer Verbindungswiederholung nicht erneut erstellen, weil das Modell, über das sie erstellt wurde, nicht verfügbar ist. In der MQI kann nur eine Anwendung die temporäre dynamische Warteschlange nach Modell öffnen. Andere Anwendungen, die die temporäre dynamische Warteschlange verwenden möchten, müssen MQPUT1- oder -Serverbindungen verwenden oder die erneute Verbindung erneut versuchen, wenn sie fehlschlägt.

Es können nur nicht persistente Nachrichten in eine temporäre dynamische Warteschlange gestellt werden, und diese Nachrichten gehen während der Funktionsübernahme verloren. Dieser Verlust gilt für Nachrichten, die während der erneuten Verbindung mit MQPUT1 in eine temporäre dynamische Warteschlange gestellt werden. Wenn das Failover während des MQPUT1-Befehls auftritt, wird die Nachricht möglicherweise nicht gestellt, obwohl die Nachricht MQPUT1 erfolgreich ist. Eine Fehlerum-

gehung für dieses Problem besteht darin, permanente dynamische Warteschlangen zu verwenden. Jede Serververbindungsanwendung kann die temporäre dynamische Warteschlange nach Namen öffnen, weil sie nicht wieder verbunden werden kann.

### **Datenwiederherstellung und hohe Verfügbarkeit**

Hochverfügbarkeitslösungen unter Verwendung von Warteschlangenmanagern mit mehreren Instanzen müssen einen Mechanismus zum Wiederherstellen von Daten nach einem Speicherfehler enthalten.

Ein Warteschlangenmanager mit mehreren Instanzen erhöht die Verfügbarkeit von WS-Managerprozessen, aber nicht die Verfügbarkeit anderer Komponenten, wie z. B. das Dateisystem, das der Warteschlangenmanager zum Speichern von Nachrichten verwendet, und andere Informationen.

Eine Möglichkeit, Daten hoch verfügbar zu machen, ist die Verwendung von vernetztem, ausfallsicheren Datenspeicher. Sie können entweder Ihre eigene Lösung mit Hilfe eines vernetzten Dateisystems und einer flexiblen Datenspeicherung erstellen, oder Sie können eine integrierte Lösung kaufen. Wenn Sie die Ausfallsicherheit mit Disaster Recovery kombinieren möchten, ist die asynchrone Plattenreplikation verfügbar, die eine Plattenreplikation über mehrere Dutzend oder Hunderte von Kilometern zulässt.

Sie können konfigurieren, wie unterschiedliche WebSphere MQ -Verzeichnisse Speichermedien zugeordnet werden, um die Medien optimal zu nutzen. Für *Multi-Instanz* -Warteschlangenmanager gibt es eine wichtige Unterscheidung zwischen zwei Typen von WebSphere MQ -Verzeichnissen und -Dateien.

#### **Verzeichnisse, die gemeinsam von den Instanzen eines Warteschlangenmanagers gemeinsam genutzt werden müssen.**

Die Informationen, die zwischen verschiedenen Instanzen eines Warteschlangenmanagers gemeinsam genutzt werden müssen, befinden sich in zwei Verzeichnissen: den Verzeichnissen `mqgtr` und `logs`. Die Verzeichnisse müssen sich in einem gemeinsam genutzten Netzdateisystem befinden. Es wird empfohlen, einen Speicherdatenträger zu verwenden, der eine ständige hohe Verfügbarkeit und eine ausgezeichnete Leistung bietet, da die Daten ständig geändert werden, da Nachrichten erstellt und gelöscht werden.

#### **Verzeichnisse und Dateien, die nicht *haben*, um von Instanzen eines Warteschlangenmanagers gemeinsam genutzt zu werden.**

Einige andere Verzeichnisse müssen nicht von verschiedenen Instanzen eines Warteschlangenmanagers gemeinsam genutzt werden und werden schnell mit anderen Verzeichnissen zurückgeschrieben, als mit einem gespiegelten Dateisystem.

- WebSphere Ausführbare MQ -Dateien und das Verzeichnis 'tools'. Durch die erneute Installation oder durch Sichern und Zurückschreiben aus einem gesicherten Dateiarhiv ersetzen.
- Konfigurationsinformationen, die für die Installation als Ganzes geändert werden. Die Konfigurationsdaten werden entweder von WebSphere MQ verwaltet, z. B. von der Datei `mqsc.ini` auf Windows-, UNIX and Linux -Systemen oder von einem Teil Ihres eigenen Konfigurationsmanagements, z. B. **MQSC** -Konfigurationsscripts. Sichern und Zurückschreiben mit einem Dateiarhiv.
- Installationsweite Ausgabe, wie z. B. Traces, Fehlerprotokolle und FFDC-Dateien. Die Dateien werden in den Unterverzeichnissen `errors` und `trace` im Standarddatenverzeichnis gespeichert. Das Standarddatenverzeichnis auf UNIX and Linux -Systemen ist `/var/mqm`. Unter Windows ist das Standarddatenverzeichnis das WebSphere MQ -Installationsverzeichnis.

Sie können auch einen Sicherungswarteschlangenmanager verwenden, um regelmäßige Datenträgersicherungen eines Multi-Instanz-Warteschlangenmanagers mit linearer Protokollierung zu verwenden. Ein Sicherungswarteschlangenmanager stellt keine Wiederherstellung bereit, die so schnell wie von einem gespiegelten Dateisystem ist, und die Änderungen seit der letzten Sicherung werden nicht wiederhergestellt. Der Sicherungswarteschlangenmanager-Mechanismus eignet sich besser für die Verwendung in Off-Site-Szenarios zur Wiederherstellung nach einem Katastrophenfall als die Wiederherstellung eines Warteschlangenmanagers nach einem lokalisierten Speicherfehler.

### **IBM WebSphere MQ-Verfügbarkeitslösungen kombinieren**

Von Anwendungen werden weitere IBM WebSphere MQ-Leistungsmerkmale genutzt, um die Verfügbarkeit zu steigern. Warteschlangenmanager mit mehreren Instanzen ergänzen andere Hochverfügbarkeitsfunktionen.

## **IBM WebSphere MQ-Cluster steigern die Warteschlangenverfügbarkeit**

Sie können die Warteschlangenverfügbarkeit erhöhen, indem Sie mehrere Definitionen einer Clusterwarteschlange erstellen; bis zu einer jeden Warteschlange auf den einzelnen Managern im Cluster.

Angenommen, ein Member des Clusters schlägt fehl, und anschließend wird eine neue Nachricht an eine Clusterwarteschlange gesendet. Wenn die Nachricht *has* nicht in den fehlgeschlagenen Warteschlangenmanager wechseln soll, wird die Nachricht an einen anderen aktiven WS-Manager im Cluster gesendet, der über eine Definition der Warteschlange verfügt.

Obwohl Cluster die Verfügbarkeit erheblich erhöhen, gibt es zwei zusammengehörige Fehlerszenarios, die zu verzögerten Nachrichten führen. Durch die Erstellung eines Clusters mit Multi-Instanz-WS-Managern wird die Wahrscheinlichkeit, dass eine Nachricht verzögert wird, verringert.

### **Marooned-Nachrichten**

Wenn ein Warteschlangenmanager im Cluster fehlschlägt, werden keine weiteren Nachrichten, die an andere WS-Manager im Cluster weitergeleitet werden können, an den fehlgeschlagenen Warteschlangenmanager weitergeleitet. Nachrichten, die bereits gesendet wurden, werden bis zum Neustart des fehlgeschlagenen Warteschlangenmanagers gemieckt.

### **Affinitäten**

Affinität ist der Begriff, der verwendet wird, um Informationen zu beschreiben, die zwischen zwei ansonsten getrennten Berechnungen gemeinsam genutzt werden. Beispielsweise besteht eine Affinität zwischen einer Anwendung, die eine Anforderungsnachricht an einen Server sendet, und der gleichen Anwendung, die die Verarbeitung der Antwort erwartet. Ein weiteres Beispiel wäre eine Folge von Nachrichten, die die Verarbeitung jeder Nachricht in Abhängigkeit von den vorherigen Nachrichten enthält.

Wenn Sie Nachrichten an geclusterte Warteschlangen senden, müssen Sie Affinitäten berücksichtigen. Müssen Sie aufeinanderfolgende Nachrichten an denselben WS-Manager senden, oder kann jede Nachricht an ein beliebigen Member des Clusters gesendet werden?

Wenn Sie Nachrichten an denselben Warteschlangenmanager im Cluster senden müssen und die Nachrichten fehlschlagen, warten die Nachrichten in der Übertragungswarteschlange des Senders, bis der fehlgeschlagene Cluster-WS-Manager erneut ausgeführt wird.

Wenn der Cluster mit Multi-Instanz-WS-Managern konfiguriert ist, ist die Verzögerung, die auf den Neustart des fehlgeschlagenen Warteschlangenmanagers wartet, auf die Reihenfolge einer Minute begrenzt, während die Bereitschaftsdatenbank die Zeit übernimmt. Wenn die Bereitschaftsdatenbank aktiv ist, werden die Nachrichten wieder aufgenommen, die Kanäle zur neu aktivierten Warteschlangenmanagerinstanz gestartet werden und die Nachrichten, die in Übertragungswarteschlangen anstanden, werden gestartet.

Eine Möglichkeit, einen Cluster so zu konfigurieren, dass die Nachrichten, die von einem fehlgeschlagenen Warteschlangenmanager verzögert werden, überwunden werden, besteht darin, zwei verschiedene Warteschlangenmanager auf jedem Server im Cluster zu implementieren und eine aktive und eine als die Standby-Instanz der verschiedenen Warteschlangenmanager zu definieren. Hierbei handelt es sich um eine Aktiv-Standby-Konfiguration, die die Verfügbarkeit des Clusters erhöht.

Neben den Vorteilen einer reduzierten Verwaltung und einer erhöhten Skalierbarkeit stellen Cluster weiterhin zusätzliche Elemente der Verfügbarkeit zur Verfügung, um Multi-Instanz-WS-Manager zu ergänzen. Cluster schützen vor anderen Typen von Fehlern, die sowohl die aktiven als auch die Standby-Instanzen eines Warteschlangenmanagers betreffen.

### **Ununterbrochener Service**

Ein Cluster stellt einen ununterbrochenen Service bereit. Neue Nachrichten, die vom Cluster empfangen werden, werden an aktive Warteschlangenmanager gesendet, die verarbeitet werden sollen. Verlassen Sie sich nicht auf einen Warteschlangenmanager mit mehreren Instanzen, um einen unterbrechungsfreien Service bereitzustellen, da es Zeit für den Standby-WS-Manager benötigt, um den Fehler zu erkennen und seinen Start abzuschließen, damit die Kanäle erneut verbunden werden, und für fehlgeschlagene Nachrichtenstapel, die erneut übergeben werden sollen.

### **Lokalisierter Ausfall**

Es gibt praktische Einschränkungen, wie weit die aktiven, die Standby- und die Dateisystemserver voneinander entfernt werden können, da sie mit einer Millisekundengeschwindigkeit interagieren müssen, um eine akzeptable Leistung zu erzielen.

Clusterwarteschlangenmanager erfordern Interaktionsgeschwindigkeiten in der Größenordnung von vielen Sekunden und können geographisch überall auf der Welt verteilt werden.

### **Betriebsfehler**

Durch die Verwendung von zwei verschiedenen Mechanismen zur Erhöhung der Verfügbarkeit reduzieren Sie die Wahrscheinlichkeit, dass ein Betriebsfehler, wie z. B. ein menschlicher Fehler, Ihre Verfügbarkeitsanstrengungen beeinträchtigt.

## **Gruppen mit gemeinsamer Warteschlange erhöhen die Verfügbarkeit der Nachrichtenverarbeitung**

Gruppen mit gemeinsamer Warteschlange, die nur unter z/OS bereitgestellt werden, ermöglichen es einer Gruppe von Warteschlangenmanagern, die Wartung einer Warteschlange gemeinsam zu nutzen. Wenn ein Warteschlangenmanager ausfällt, verarbeiten die anderen WS-Manager weiterhin alle Nachrichten in der Warteschlange. Mehrinstanz-Warteschlangenmanager werden unter z/OS nicht unterstützt und ergänzen Gruppen mit gemeinsamer Warteschlange nur als Bestandteil einer breiteren Messaging-Architektur.

## **WebSphere MQ Clients erhöhen die Anwendungsverfügbarkeit**

WebSphere MQ MQI-Clientprogramme können je nach Verfügbarkeit des Warteschlangenmanagers, Verbindungsgewichtungen und Affinitäten eine Verbindung zu verschiedenen Warteschlangenmanagern in einer Warteschlangenmanagergruppe herstellen. Wenn Sie eine Anwendung auf einem anderen System als dem Warteschlangenmanager ausführen, auf dem der Warteschlangenmanager ausgeführt wird, können Sie die Gesamtverfügbarkeit einer Lösung verbessern, solange eine Möglichkeit besteht, die Anwendung erneut zu verbinden, wenn die WS-Manager-Instanz, mit der sie verbunden ist, fehlgeschlagen ist.

WS-Manager-Gruppen werden verwendet, um die Clientverfügbarkeit zu erhöhen, indem ein Client aus einem Warteschlangenmanager entfernt wird, der gestoppt wurde, und die Lastverteilung von Clientverbindungen in eine Gruppe von Warteschlangenmanagern, und zwar wie ein IP-Sprayer. Die Clientanwendung darf keine Affinitäten mit dem fehlgeschlagenen Warteschlangenmanager haben, z. B. eine Abhängigkeit in einer bestimmten Warteschlange oder die Verarbeitung kann nicht fortgesetzt werden.

Automatische Clientwiederverbindungs- und Multi-Instanz-Warteschlangenmanager erhöhen die Verfügbarkeit der Clients, indem einige Affinitätsprobleme behoben werden. Die automatische Wiederherstellung einer Clientverbindung wird von WebSphere MQ-Klassen für Java nicht unterstützt.

Sie können die Option `MQCNO MQCNO_RECONNECT_Q_MGR` festlegen, um einen Client zu zwingen, die Verbindung zum selben Warteschlangenmanager erneut herzustellen:

1. Wenn der zuvor verbundene einzelne Instanz-WS-Manager nicht aktiv ist, wird die Verbindung wiederholt, bis der Warteschlangenmanager wieder aktiv ist.
2. Wenn der Warteschlangenmanager als Multi-Instanz-Warteschlangenmanager konfiguriert ist, stellt der Client die Verbindung zu der Instanz wieder her, die aktiv ist.

Durch die automatische Verbindung zu demselben Warteschlangenmanager werden viele der Statusinformationen, die der Warteschlangenmanager im Namen des Clients gespeichert hat, wie z. B. die von ihm geöffneten Warteschlangen und das von ihm subskribierte Topic, wiederhergestellt. Wenn der Client eine dynamische Empfangswarteschlange für Antworten geöffnet hat, um eine Antwort auf eine Anforderung zu empfangen, wird auch die Verbindung zur Warteschlange für Antwortantworten wiederhergestellt.

## **Sicherstellen, dass Nachrichten nicht verloren gehen (Protokollierung)**

WebSphere MQ protokolliert alle Informationen, die Sie für die Wiederherstellung nach einem Warteschlangenmanagerfehler benötigen.

WebSphere MQ zeichnet alle signifikanten Änderungen an den Daten, die vom Warteschlangenmanager gesteuert werden, in einem Wiederherstellungsprotokoll auf.

Dies umfasst das Erstellen und Löschen von Objekten, persistenten Nachrichtenaktualisierungen, Transaktionsstatus, Änderungen an Objektattributen und Kanalaktivitäten. Das Protokoll enthält die Informationen, die Sie benötigen, um alle Aktualisierungen für Nachrichtenwarteschlangen wiederherzustellen, indem Sie folgende Schritte ausführen:

- Datensätze von WS-Manager-Änderungen werden beibehalten
- Datensätze von Warteschlangenaktualisierungen für die Verwendung durch den Neustartprozess beibehalten
- Wiederherstellung von Daten nach einem Hardware- oder Softwarefehler

WebSphere MQ stützt sich jedoch auch auf das Plattensystem, auf dem sich die Dateien befinden. Wenn das Plattensystem selbst unzuverlässig ist, können Informationen, einschließlich Protokollinformationen, immer noch verloren gehen.

## Wie Logs aussehen

Protokolle bestehen aus primären und sekundären Dateien und einer Steuerdatei. Sie definieren die Anzahl und Größe der Protokolldateien und werden dort gespeichert, wo sie im Dateisystem gespeichert werden.

Ein WebSphere MQ -Protokoll besteht aus zwei Komponenten:

1. Eine oder mehrere Dateien mit Protokolldaten.
2. Eine Protokollsteuerdatei

Eine Datei mit Protokolldaten wird auch als Protokollspeicherbereich bezeichnet.

Es gibt eine Reihe von Protokolldateien, die die aufgezeichneten Daten enthalten. Sie können die Anzahl und Größe (wie in „[Konfigurationsinformationen für IBM WebSphere MQ und Warteschlangenmanager ändern](#)“ auf Seite 454 erläutert) definieren oder den Systemstandardwert von drei Dateien übernehmen.

In WebSphere MQ für Windows beträgt der Standardwert für jede der drei Dateien 1 MB. In WebSphere MQ für UNIX and Linux -Systeme ist jede der drei Dateien standardmäßig 4 MB groß.

Wenn Sie einen Warteschlangenmanager erstellen, entspricht die Anzahl der Protokolldateien, die Sie definieren, der Anzahl der zugeordneten *primären* Protokolldateien. Wenn Sie keine Zahl angeben, wird der Standardwert verwendet.

Wenn Sie in WebSphere MQ für Windows den Protokollpfad nicht geändert haben, werden Protokolldateien im folgenden Verzeichnis erstellt:

```
C:\Program Files\IBM\WebSphere MQ\log\<QMGrName>
```

Wenn Sie in WebSphere MQ für UNIX and Linux -Systeme den Protokollpfad nicht geändert haben, werden Protokolldateien im folgenden Verzeichnis erstellt:

```
/var/mqm/log/<QMGrName>
```

WebSphere MQ beginnt mit diesen primären Protokolldateien. Wenn der primäre Protokollspeicherbereich jedoch nicht ausreicht, werden *sekundäre* Protokolldateien zugeordnet. Dies macht sie dynamisch und entfernt sie, wenn die Nachfrage nach Protokollspeicherbereich reduziert wird. Standardmäßig können bis zu zwei sekundäre Protokolldateien zugeordnet werden. Sie können diese Standardzuordnung ändern, wie in „[Konfigurationsinformationen für IBM WebSphere MQ und Warteschlangenmanager ändern](#)“ auf Seite 454 beschrieben.

## Die Protokollsteuerdatei

Die Protokollsteuerdatei enthält die Informationen, die erforderlich sind, um die Verwendung von Protokolldateien zu steuern, wie z. B. ihre Größe und Position sowie den Namen der nächsten verfügbaren Datei.

Die Protokollsteuerdatei dient nur der internen Verwendung des Warteschlangenmanagers.



Der WS-Manager verwaltet die Steuerdaten, die dem Status des Wiederherstellungsprotokolls in der Protokollsteuerdatei zugeordnet sind, und Sie dürfen den Inhalt der Protokollsteuerdatei nicht ändern.

**Anmerkung:** Stellen Sie sicher, dass die Protokolle, die beim Starten eines Warteschlangenmanagers erstellt werden, groß genug sind, um die Größe und das Volumen der Nachrichten aufzunehmen, die von Ihren Anwendungen verarbeitet werden. Sie müssen wahrscheinlich die Standardprotokollnummern und -größen ändern, um Ihre Anforderungen zu erfüllen. Weitere Informationen finden Sie im Abschnitt [„Berechnen der Größe des Protokolls“](#) auf Seite 436.

## Typen der Protokollierung

In WebSphere MQ hängt die Anzahl der Dateien, die für die Protokollierung erforderlich sind, von der Dateigröße, der Anzahl der Nachrichten, die Sie empfangen haben, und der Länge der Nachrichten ab. Es gibt zwei Möglichkeiten, Datensätze von Warteschlangenmanageraktivitäten zu verwalten: Umlaufprotokollierung und lineare Protokollierung.

### Umlaufprotokollierung

Verwenden Sie die Umlaufprotokollierung, wenn Sie möchten, dass die Wiederherstellung erneut gestartet wird. Verwenden Sie das Protokoll, um Transaktionen rückgängig zu machen, die sich in Bearbeitung befanden, als das System gestoppt wurde.

Die Umlaufprotokollierung speichert alle Neustartdaten in einem Ring von Protokolldateien. Die Protokollierung füllt die erste Datei im Ring aus und wird dann in die nächste Datei verschoben, und so weiter, bis alle Dateien voll sind. Anschließend geht es zurück in die erste Datei im Ring und beginnt erneut. Dies wird solange fortgesetzt, wie das Produkt im Gebrauch ist, und hat den Vorteil, dass Sie keine Protokolldateien mehr ausführen.

WebSphere MQ speichert die Protokolleinträge, die erforderlich sind, um den Warteschlangenmanager ohne Datenverlust erneut zu starten, bis sie nicht mehr benötigt werden, um die Wiederherstellung der Warteschlangenmanagerdaten sicherzustellen. Der Mechanismus zum Freigeben von Protokolldateien für die Wiederverwendung wird in [„Verwenden des Prüfpunktprogramms zur Sicherstellung einer vollständigen Wiederherstellung“](#) auf Seite 434 beschrieben.

### Lineare Protokollierung

Verwenden Sie die lineare Protokollierung, wenn Sie sowohl die Wiederherstellung als auch die Datenträgerwiederherstellung (Wiederherstellung verlorener oder beschädigter Daten durch Wiedergabe des Inhalts des Protokolls) verwenden möchten. Die lineare Protokollierung belässt die Protokolldateien in einer fortlaufenden Dateisequenz. Der Speicherbereich wird nicht wiederverwendet, so dass Sie jede Aufzeichnung in einem beliebigen Protokollspeicherbereich, die nicht gelöscht wurde, wiederfinden können.

Da der Plattenspeicherplatz begrenzt ist, müssen Sie möglicherweise über eine Art der Archivierung nachdenken. Es ist eine Verwaltungstask, Ihren Plattenspeicherplatz für das Protokoll zu verwalten, den vorhandenen Speicherplatz nach Bedarf wieder zu verwenden oder zu erweitern.

Die Anzahl der Protokolldateien, die für die lineare Protokollierung verwendet werden, kann je nach Nachrichtenfluss und Alter des Warteschlangenmanagers sehr groß sein. Es gibt jedoch eine Reihe von Dateien, die als *aktiv* bezeichnet werden. Aktive Dateien enthalten die Protokolleinträge, die für den Neustart des WS-Managers erforderlich sind. Gemeinsam werden aktive Protokolldateien als *aktive Protokolldatei* bezeichnet. Die Anzahl der aktiven Protokolldateien ist in der Regel kleiner als die Anzahl primärer Protokolldateien, die in den Konfigurationsdateien definiert sind. (Informationen zum Definieren der Anzahl finden Sie im Abschnitt [„Berechnen der Größe des Protokolls“](#) auf Seite 436.)

Das Schlüsselereignis, das steuert, ob eine Protokolldatei als aktiv bezeichnet wird oder nicht als *checkpoint* bezeichnet wird. Ein WebSphere MQ -Prüfpunkt ist ein Konsistenzpunkt zwischen dem Wiederherstellungsprotokoll und den Objektdateien. Ein Prüfpunkt bestimmt die Gruppe von Protokolldateien, die für die Neustartwiederherstellung erforderlich sind. Protokolldateien, die nicht aktiv sind, sind für die Neustartwiederherstellung nicht erforderlich und werden als inaktiv bezeichnet. In einigen Fällen sind inaktive Protokolldateien für die Datenträgerwiederherstellung erforderlich. (Weitere Informationen zum

Prüfpunktverfahren finden Sie in „Verwenden des Prüfpunktprogramms zur Sicherstellung einer vollständigen Wiederherstellung“ auf Seite 434 .)

Inaktive Protokolldateien können archiviert werden, weil sie nicht für die Wiederherstellung nach einem Neustart erforderlich sind. Inaktive Protokolldateien, die für die Datenträgerwiederherstellung nicht erforderlich sind, können als überflüssige Protokolldateien betrachtet werden. Sie können überflüssige Protokolldateien löschen, wenn sie für Ihre Operation nicht mehr von Interesse sind. Weitere Informationen zur Disposition von Protokolldateien finden Sie in „Protokolle verwalten“ auf Seite 438 .

Wenn ein neuer Prüfpunkt in der zweiten oder späteren primären Protokolldatei aufgezeichnet wird, kann die erste Datei inaktiv werden und eine neue primäre Datei wird formatiert und am Ende des primären Pools hinzugefügt, wobei die Anzahl der für die Protokollierung verfügbaren primären Dateien zurückgeschrieben wird. Auf diese Weise kann der primäre Protokolldateipool als aktuelle Gruppe von Dateien in einer immer erweiterten Liste von Protokolldateien betrachtet werden. Auch hier ist es eine Verwaltungstask, die inaktiven Dateien entsprechend den Anforderungen Ihrer Operation zu verwalten.

Obwohl sekundäre Protokolldateien für die lineare Protokollierung definiert sind, werden sie im normalen Betrieb nicht verwendet. Wenn eine Situation eintritt, die wahrscheinlich auf langlebige Transaktionen zurückzuführen ist, ist es nicht möglich, eine Datei aus dem aktiven Pool zu befreien, da sie möglicherweise noch für einen Neustart erforderlich ist. Sekundärdateien werden formatiert und dem Pool für aktive Protokolldateien hinzugefügt.

Wenn die Anzahl der verfügbaren sekundären Dateien aufgebraucht ist, werden Anforderungen für die meisten weiteren Operationen, die Protokollaktivität erfordern, mit dem Rückkehrcode MQRC\_RESOURCE\_PROBLEM abgelehnt, der an die Anwendung zurückgegeben wird.

Beide Protokollierungstypen können mit einem unerwarteten Stromausfall umgehen, vorausgesetzt, es liegt kein Hardwarefehler vor.

## **Verwenden des Prüfpunktprogramms zur Sicherstellung einer vollständigen Wiederherstellung**

Prüfpunkte synchronisieren Warteschlangenmanagerdaten und -Protokolldateien und markieren einen Konsistenzpunkt, aus dem Protokollsätze gelöscht werden können. Häufiges Prüfpunktverfahren beschleunigt die Wiederherstellung.

Persistente Aktualisierungen von Nachrichtenwarteschlangen werden in zwei Phasen durchgeführt. Zuerst werden die Datensätze, die die Aktualisierung darstellen, in das Protokoll geschrieben. Anschließend wird die Warteschlangendatei aktualisiert. Die Protokolldateien können somit auf dem neuesten Stand sein als die Warteschlangendateien. Um sicherzustellen, dass die Neustartverarbeitung an einem konsistenten Punkt beginnt, verwendet WebSphere MQ Prüfpunkte. Ein Prüfpunkt ist ein Zeitpunkt, an dem der im Protokoll beschriebene Datensatz mit dem Datensatz in der Warteschlange identisch ist. Der Prüfpunkt selbst besteht aus einer Reihe von Protokollsätzen, die für den Neustart des Warteschlangenmanagers erforderlich sind, z. B. der Status aller Transaktionen (Arbeitseinheiten), die zum Zeitpunkt des Prüfpunkts aktiv sind.

WebSphere MQ generiert Prüfpunkte automatisch. Sie werden verwendet, wenn der Warteschlangenmanager gestartet wird, beim Herunterfahren, wenn der Protokollspeicherbereich knapp wird, und nach jeweils 10.000 protokollierten Operationen.

Wenn die Warteschlangen weitere Nachrichten verarbeiten, wird der Prüfpunktsatz mit dem aktuellen Status der Warteschlangen nicht konsistent.

Bei einem Neustart von WebSphere MQ wird der letzte Prüfpunktsatz im Protokoll gefunden. Diese Informationen werden in der Prüfpunktdatei gehalten, die am Ende jedes Prüfpunkts aktualisiert wird. Der Prüfpunktsatz stellt den letzten Konsistenzpunkt zwischen dem Protokoll und den Daten dar. Alle Operationen, die seit dem Prüfpunkt ausgeführt wurden, werden erneut wiedergegeben. Dies wird als "Wiedergabephase" bezeichnet. Die Wiedergabephase bewirkt, dass die Warteschlangen wieder in den logischen Zustand zurückgebracht werden, in dem sie sich vor dem Systemausfall oder dem Systemabschluss befanden. Während der Wiedergabephase wird eine Liste der Transaktionen erstellt, die bei Auftreten des Systemausfalls oder des Systemabschlusses in den Fluten ausgeführt wurden. Die Nachrichten AMQ7229 und AMQ7230 werden ausgegeben, um den Verlauf der Wiedergabephase anzugeben.

Um zu ermitteln, welche Operationen zurückgesetzt oder festgeschrieben werden sollen, greift WebSphere MQ auf jeden aktiven Protokollsatz zu, der einer unvollständigen Transaktion zugeordnet ist. Dies wird als Recovery-Phase bezeichnet. Die Nachrichten AMQ7231, AMQ7232 und AMQ7234 werden ausgegeben, um den Verlauf der Wiederherstellungsphase anzugeben.

Sobald während der Wiederherstellungsphase auf alle erforderlichen Protokollsätze zugegriffen wurde, wird jede aktive Transaktion wiederum aufgelöst, und jede der Transaktion zugeordnete Operation wird entweder zurückgesetzt oder festgeschrieben. Dies wird als Auflösungsphase bezeichnet. Die Nachricht AMQ7233 wird ausgegeben, um den Verlauf der Auflösungsphase anzugeben.

WebSphere MQ verwaltet interne Zeiger auf den Kopf und das Ende des Protokolls. Er versetzt den Kopfzeiger auf den neuesten Prüfpunkt, der mit der Wiederherstellung von Nachrichtendaten konsistent ist.

Prüfpunkte werden verwendet, um die Recovery effizienter zu machen, und um die Wiederverwendung von primären und sekundären Protokolldateien zu steuern.

In Abbildung 69 auf Seite 435 werden alle Datensätze vor dem letzten Prüfpunkt, Prüfpunkt 2, nicht mehr von WebSphere MQ benötigt. Die Warteschlangen können aus den Prüfpunktinformationen und allen späteren Protokolleinträgen wiederhergestellt werden. Für die Umlaufprotokollierung können alle freigegebenen Dateien erneut verwendet werden, bevor der Prüfpunkt wiederverwendet werden kann. Für ein lineares Protokoll müssen die freigegebenen Protokolldateien nicht mehr für den normalen Betrieb zugänglich gemacht werden und werden inaktiv. Im Beispiel wird der Zeigerkopfzeiger so bewegt, dass er auf den letzten Prüfpunkt (Prüfpunkt 2) zeigt, der dann zum neuen Warteschlangenkopf, Head 2, wird. Protokolldatei 1 kann jetzt wiederverwendet werden.

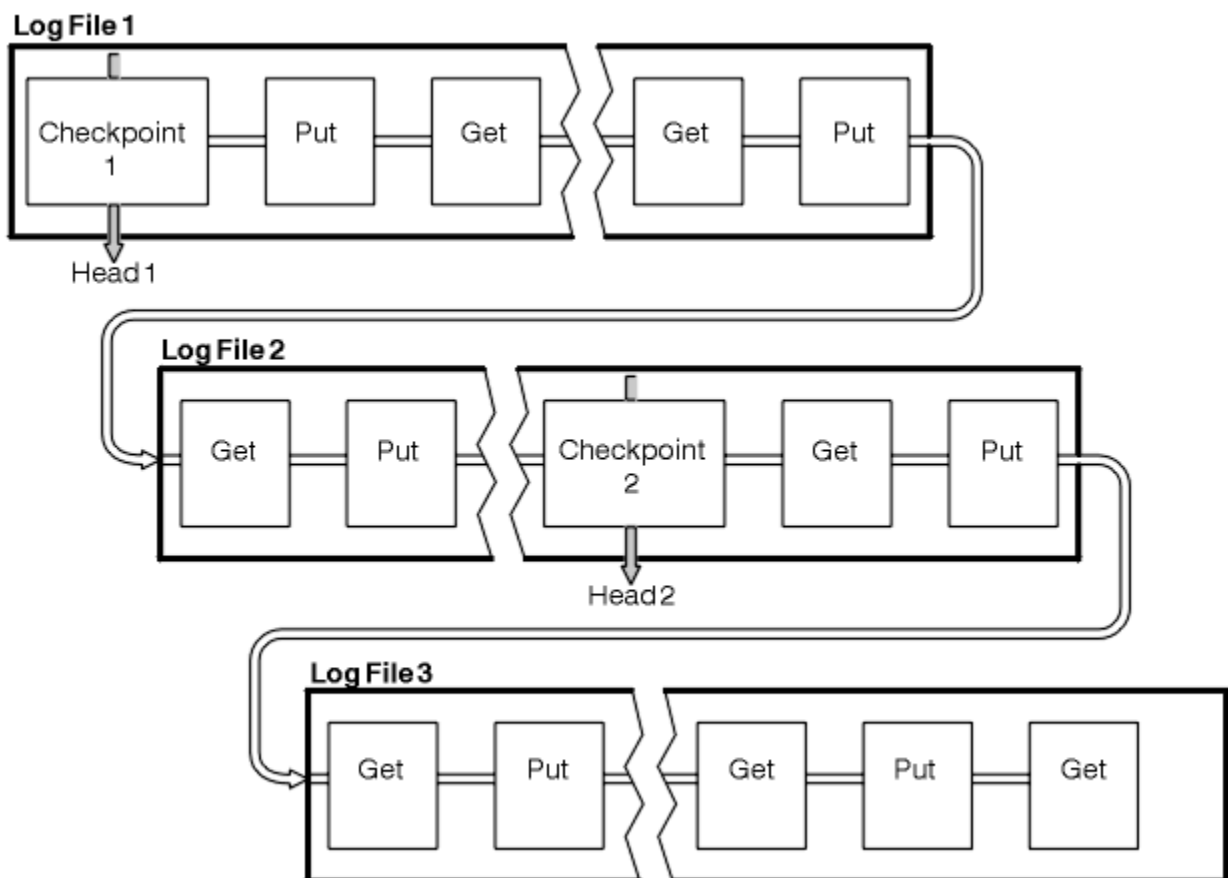


Abbildung 69. Prüfpunktprogramm

### **Prüfpunktverlaufes mit Transaktionen mit langer Laufzeit**

Wie eine Transaktion mit langer Laufzeit die Wiederverwendung von Protokolldateien beeinflusst.

Abbildung 70 auf Seite 436 zeigt, wie sich eine Transaktion mit langer Laufzeit auf die Wiederverwendung von Protokolldateien auswirkt. In dem Beispiel hat eine Transaktion mit langer Laufzeit einen Eintrag in das Protokoll gestellt, das als LR 1 dargestellt wird, nachdem der erste Prüfpunkt angezeigt wurde. Die Transaktion wird (bei Punkt LR 2) bis nach dem dritten Prüfpunkt nicht vollständig ausgeführt. Alle Protokollinformationen von LR 1 werden beibehalten, um die Wiederherstellung dieser Transaktion zu ermöglichen, sofern dies erforderlich ist, bis sie abgeschlossen ist.

Nach Abschluss der Transaktion mit langer Laufzeit wird bei LR 2 der Protokollkopf an Prüfpunkt 3, den zuletzt protokollierten Prüfpunkt, verschoben. Die Dateien, die Protokollsätze vor Prüfpunkt 3, Head 2, enthalten, werden nicht mehr benötigt. Wenn Sie die Umlaufprotokollierung verwenden, kann der Speicherbereich wiederverwendet werden.

Wenn die primären Protokolldateien voll sind, bevor die Transaktion mit langer Laufzeit abgeschlossen ist, werden sekundäre Protokolldateien verwendet, um zu verhindern, dass die Protokolle voll werden.

Wenn der Protokollkopf verschoben wird und Sie die Umlaufprotokollierung verwenden, werden die primären Protokolldateien möglicherweise für die Wiederverwendung in Frage gestellt, und die Protokollfunktion, nachdem die aktuelle Datei gefüllt wurde, verwendet die erste verfügbare Primärdatei erneut. Wenn Sie die lineare Protokollierung verwenden, wird der Protokollkopf immer noch in den aktiven Pool verschoben, und die erste Datei wird inaktiv. Eine neue Primärdatei wird formatiert und der Basis des Pools hinzugefügt, wenn sie für zukünftige Protokollierungsaktivitäten bereit ist.

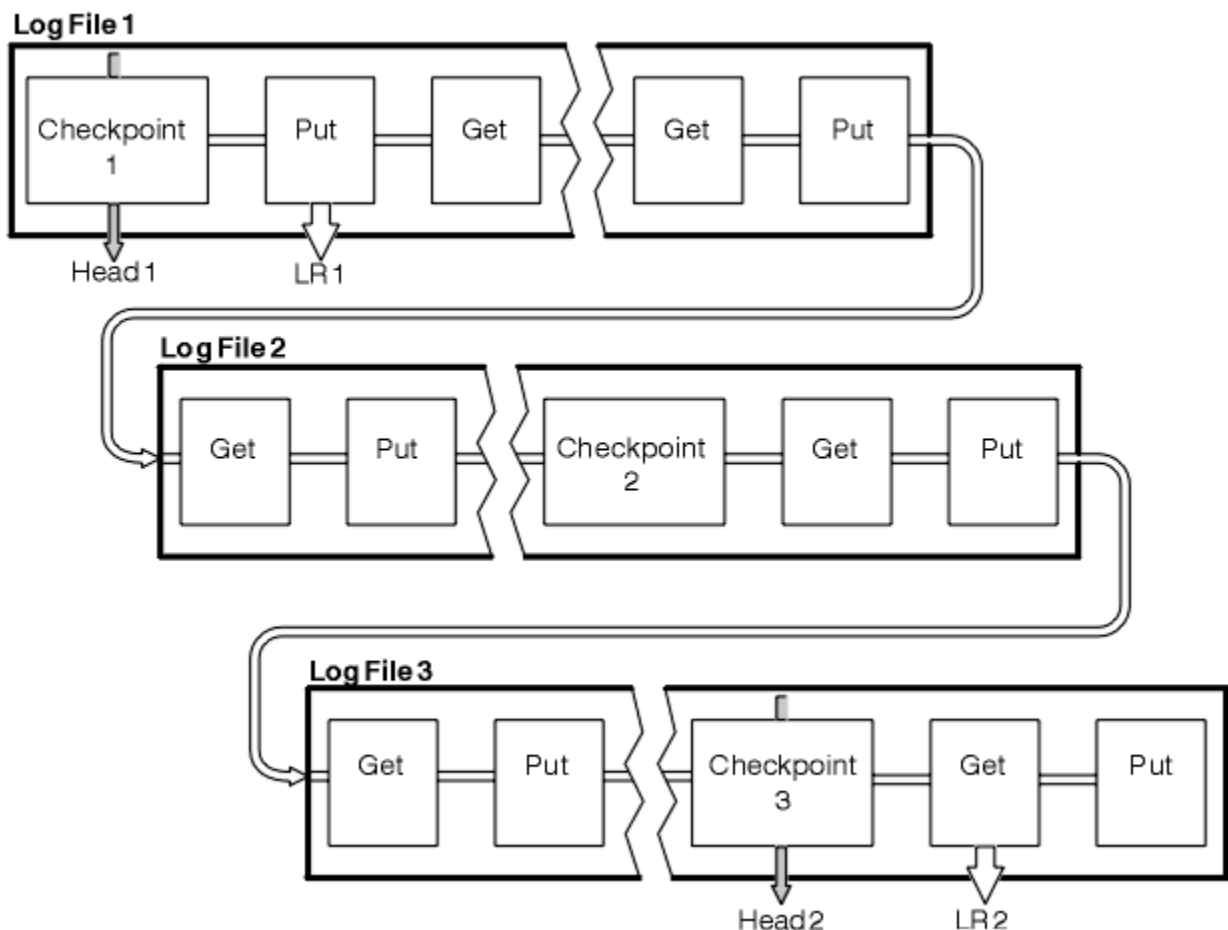


Abbildung 70. Prüfpunktüberprüfung mit einer Transaktion mit langer Laufzeit

## Berechnen der Größe des Protokolls

Schätzen der Größe des Protokollwarteschlangenmanagers.

Nachdem Sie entschieden haben, ob der Warteschlangenmanager die Umlaufprotokollierung oder die lineare Protokollierung verwendet, müssen Sie die Größe des Protokolls schätzen, das der Warteschlan-

genmanager benötigt. Die Größe des Protokolls wird durch die folgenden Protokollkonfigurationsparameter bestimmt:

**LogFilePages**

Die Größe der einzelnen primären und sekundären Protokolldateien in Einheiten von 4-KB-Seiten.

**LogPrimaryFiles**

Die Anzahl der vorab zugeordneten primären Protokolldateien.

**LogSecondaryFiles**

Die Anzahl der sekundären Protokolldateien, die erstellt werden können, wenn die primären Protokolldateien voll sind.

Tabelle 31 auf Seite 437 zeigt das Datenvolumen, das der Warteschlangenmanager für verschiedene Operationen protokolliert. Die meisten Warteschlangenmanageroperationen benötigen einen minimalen Protokollspeicherbereich. Wenn jedoch eine persistente Nachricht in eine Warteschlange eingereiht wird, müssen **alle** Nachrichtendaten in das Protokoll geschrieben werden, damit die Nachricht wiederhergestellt werden kann. Die Größe des Protokolls hängt normalerweise von der Anzahl und Größe der persistenten Nachrichten ab, die der Warteschlangenmanager verarbeiten muss.

*Tabelle 31. Protokolleintragsgrößen (alle Werte sind ungefähr)*

| Operation                             | Größe                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|---------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Persistente Nachricht einreihen       | 750 Byte + Nachrichtenlänge<br>Wenn die Nachricht groß ist, wird sie in Segmente von 261844 Byte unterteilt, wobei jedes Segment zusätzliche 300 Byte hinzufügt.                                                                                                                                                                                                                                                                                                     |
| Nachricht abrufen                     | 260 Byte                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Synchronisationspunkt, Fest-schreiben | 750 Byte                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Synchronisationspunkt, Rollback       | 1000 Byte + 12 Byte für jede GET-oder PUT-Operation, für die ein Rollback durchgeführt werden soll                                                                                                                                                                                                                                                                                                                                                                   |
| Objekt erstellen                      | 1500 Byte                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Objekt löschen                        | 300 Byte                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Attribute ändern                      | 1024 Byte                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Medienimage aufzeichnen               | 800 Byte + Bild<br>Das Bild ist in Segmente von 260.000 Byte unterteilt, wobei jedes Segment zusätzliche 300 Byte hinzufügt.                                                                                                                                                                                                                                                                                                                                         |
| Prüfpunkt                             | 750 Byte + 200 Byte für jede aktive Arbeitseinheit + 380 Byte für jeden Clustersenderkanal, wenn Sie mehrere Clusterübertragungswarteschlangen pro WS-Manager verwenden<br>Zusätzliche Daten können für nicht festgeschriebene Einreihungen oder Abrufe protokolliert werden, die aus Leistungsgründen gepuffert werden.<br>Wenn Sie Clustersenderkanäle haben, werden bei jedem Prüfpunkt zusätzliche 380 Byte pro Clustersenderkanal in das Protokoll geschrieben. |

**Anmerkung:**

1. Sie können die Anzahl der primären und sekundären Protokolldateien bei jedem Start des Warteschlangenmanagers ändern.
2. Sie können die Größe der Protokolldatei nicht ändern. Sie müssen es **vor** der Erstellung des Warteschlangenmanagers festlegen.

3. Die Anzahl primärer Protokolldateien und die Größe der Protokolldatei bestimmen die Größe des Protokollspeicherbereichs, der beim Erstellen des Warteschlangenmanagers vorab zugeordnet wird.
4. Die Gesamtzahl der primären und sekundären Protokolldateien darf auf UNIX and Linux -Systemen nicht größer als 511 und unter Windows nicht größer als 255 sein, was bei Vorhandensein von Transaktionen mit langer Laufzeit die maximale Größe des Protokollspeicherbereichs begrenzt, der dem Warteschlangenmanager für die Neustartwiederherstellung zur Verfügung steht. Die Größe des Protokollspeicherbereichs, den der WS-Manager möglicherweise für die Datenträgerwiederherstellung benötigt, teilt diese Begrenzung nicht mit.
5. Wenn die *Umlaufprotokollierung* verwendet wird, verwendet der Warteschlangenmanager den primären Protokollspeicherbereich wieder. Dies bedeutet, dass das Protokoll des Warteschlangenmanagers kleiner sein kann als das von Ihnen geschätzte Datenvolumen, das der Warteschlangenmanager protokollieren muss. Der WS-Manager ordnet bis zu einem Grenzwert eine sekundäre Protokolldatei zu, wenn eine Protokolldatei voll wird und die nächste primäre Protokolldatei in der Sequenz nicht verfügbar ist.
6. Primäre Protokolldateien werden für die Wiederverwendung während eines Prüfpunkts verfügbar gemacht. Der Warteschlangenmanager berücksichtigt vor dem Setzen eines Prüfpunkts sowohl den primären als auch den sekundären Protokollspeicherbereich, da der Protokollspeicherbereich knapp ist.

Wenn Sie nicht mehr primäre Protokolldateien als sekundäre Protokolldateien definieren, ordnet der Warteschlangenmanager möglicherweise sekundäre Protokolldateien zu, bevor ein Prüfpunkt gesetzt wird. Dadurch werden die primären Protokolldateien für die Wiederverwendung verfügbar.

## Protokolle verwalten

Protokolle sind fast selbstverwaltet, müssen aber manchmal verwaltet werden, um Speicherplatzprobleme zu lösen.

Im Laufe der Zeit werden einige der geschriebenen Protokollsätze für den Neustart des Warteschlangenmanagers unnötig. Wenn Sie die Umlaufprotokollierung verwenden, gibt der Warteschlangenmanager freigegebenen Speicherplatz in den Protokolldateien frei. Diese Aktivität ist für den Benutzer nicht offensichtlich und Sie sehen normalerweise nicht, wie viel Plattenspeicherplatz verwendet wird, da der zugeordnete Speicherplatz schnell wiederverwendet wird.

Von den Protokollsätzen werden nur die Datensätze benötigt, die seit dem Start des letzten vollständigen Prüfpunkts geschrieben wurden, und diejenigen, die von aktiven Transaktionen geschrieben wurden, zum erneuten Starten des Warteschlangenmanagers. Das Protokoll wird möglicherweise gefüllt, wenn ein Prüfpunkt lange Zeit nicht verwendet wurde oder wenn eine Transaktion mit langer Laufzeit vor langer Zeit einen Protokollsatz geschrieben hat. Der WS-Manager versucht oft genug Prüfpunkte zu nehmen, um das erste Problem zu vermeiden.

Wenn eine Transaktion mit langer Laufzeit das Protokoll füllt, schlagen Versuche, Protokollsätze zu schreiben, fehl und einige MQI-Aufrufe geben MQRC\_RESOURCE\_PROBLEM zurück. (Speicherbereich ist reserviert, um alle unvollständigen Transaktionen festzuschreiben oder rückgängig zu machen, daher sollte **MQCMIT** oder **MQBACK** nicht fehlschlagen.)

Der Warteschlangenmanager macht Transaktionen rückgängig, die zu viel Protokollspeicherbereich belegen. Eine Anwendung mit einer Transaktion wird auf diese Weise rückgängig gemacht und kann keine nachfolgenden **MQPUT** - oder **MQGET** -Operationen ausführen, die einen Synchronisationspunkt in derselben Transaktion angeben. Der Versuch, eine Nachricht unter Synchronisationspunkt in diesem Status zu setzen oder abzurufen, gibt den Wert MQRC\_BACKED\_OUT zurück. Die Anwendung kann dann **MQCMIT** ausgeben, das MQRC\_BACKED\_OUT oder **MQBACK** zurückgibt und eine neue Transaktion starten. Wenn die Transaktion, die zu viel Protokollspeicherbereich belegt, rückgängig gemacht wurde, wird ihr Protokollspeicherbereich freigegeben und der Warteschlangenmanager funktioniert weiterhin normal.

Wenn das Protokoll voll ist, wird die Nachricht AMQ7463 ausgegeben. Außerdem wird die Nachricht AMQ7465 ausgegeben, wenn das Protokoll voll wird, da eine Transaktion mit langer Laufzeit die Freigabe des Speicherbereichs verhindert hat.

Wenn Datensätze schneller in das Protokoll geschrieben werden, als das Protokoll sie verarbeiten kann, wird die Nachricht AMQ7466 ausgegeben. Wenn diese Nachricht angezeigt wird, erhöhen Sie die Zahl der Protokolldateien oder verringern Sie das Datenvolumen, das vom Warteschlangenmanager verarbeitet wird.

### **Was passiert, wenn eine Platte voll ist**

Die Protokollierungskomponente des Warteschlangenmanagers kann mit einer vollständigen Platte und mit vollständigen Protokolldateien fertig werden. Wenn die Platte, die das Protokoll enthält, voll ist, gibt der Warteschlangenmanager die Nachricht AMQ6708 aus und es wird ein Fehlerdatensatz erstellt.

Die Protokolldateien werden mit ihrer maximalen Größe erstellt und nicht erweitert, wenn Protokollsätze in sie geschrieben werden. Dies bedeutet, dass WebSphere MQ nur dann über keinen Plattenspeicherplatz mehr verfügt, wenn eine neue Datei erstellt wird. Beim Schreiben eines Datensatzes in das Protokoll kann nicht mehr genügend Speicherplatz zur Verfügung stehen. WebSphere MQ weiß immer, wie viel Speicherplatz in den vorhandenen Protokolldateien verfügbar ist, und verwaltet den Speicherplatz in den Dateien entsprechend.

Wenn Sie das Laufwerk mit den Protokolldateien füllen, können Sie möglicherweise Plattenspeicherplatz frei machen. Wenn Sie ein lineares Protokoll verwenden, kann es einige inaktive Protokolldateien im Protokollverzeichnis geben, und Sie können diese Dateien auf ein anderes Laufwerk oder eine andere Einheit kopieren. Wenn Sie noch nicht über Speicherplatz hinaus, überprüfen Sie, ob die Konfiguration des Protokolls in der Warteschlangenmanagerkonfigurationsdatei korrekt ist. Möglicherweise können Sie die Anzahl primärer oder sekundärer Protokolldateien reduzieren, so dass das Protokoll den verfügbaren Speicherplatz nicht übersteigt. Sie können die Größe der Protokolldateien für einen vorhandenen WS-Manager nicht ändern. Der Warteschlangenmanager geht davon aus, dass alle Protokolldateien dieselbe Größe haben.

### **Protokolldateien verwalten**

Allokalisieren Sie genügend Speicherplatz für Ihre Protokolldateien. Bei linearer Protokollierung können Sie alte Protokolldateien löschen, wenn sie nicht mehr benötigt werden.

Wenn Sie die Umlaufprotokollierung verwenden, stellen Sie sicher, dass genügend Speicherplatz für die Protokolldateien vorhanden ist, wenn Sie Ihr System konfigurieren (siehe „Protokollstandardwerte für IBM WebSphere MQ“ auf Seite 463 und „Warteschlangenmanagerprotokolle“ auf Seite 471). Die Größe des Plattenspeicherplatzes, der vom Protokoll verwendet wird, erhöht sich nicht über die konfigurierte Größe hinaus, einschließlich des Speicherbereichs für sekundäre Dateien, die bei Bedarf erstellt werden sollen.

Wenn Sie ein lineares Protokoll verwenden, werden die Protokolldateien kontinuierlich hinzugefügt, wenn Daten protokolliert werden, und die Menge des verwendeten Plattenspeicherplatzes wird mit der Zeit erhöht. Wenn die Rate der Daten, die protokolliert werden, hoch ist, wird der Plattenspeicherplatz schnell von neuen Protokolldateien verwendet.

Im Laufe der Zeit sind die älteren Protokolldateien für ein lineares Protokoll nicht mehr erforderlich, um den WS-Manager erneut zu starten oder die Datenträgerwiederherstellung beschädigter Objekte auszuführen. Mit den folgenden Methoden lässt sich feststellen, welche Protokolldateien noch benötigt werden:

#### **Protokollfunktionseignisnachrichten**

Wenn diese Option aktiviert ist, werden Protokollfunktionseignisnachrichten generiert, wenn Warteschlangenmanager beginnen, Protokollsätze in eine neue Protokolldatei zu schreiben. Der Inhalt der Protokollfunktionseignisnachrichten gibt die Protokolldateien an, die für den Neustart des Warteschlangenmanagers noch erforderlich sind, und die Datenträgerwiederherstellung. Weitere Informationen zu Ereignisnachrichten der Protokollfunktion finden Sie unter [Protokollierungseignisse](#).

#### **Status des Warteschlangenmanagers**

Durch Ausführen des MQSC-Befehls, DISPLAY QMSTATUS oder des PCF-Befehls "Inquire Queue Manager Status" werden die WS-Manager-Informationen, einschließlich Details der erforderlichen Protokolldateien, zurückgegeben. Weitere Informationen zu MQSC-Befehlen finden Sie im Abschnitt [Scriptbefehle \(MQSC\)](#) und Informationen zu PCF-Befehlen finden Sie im Abschnitt [Verwaltungstasks automatisieren](#).



## WS-Manager-Nachrichten

Der WS-Manager gibt in regelmäßigen Abständen ein Nachrichtenpaar aus, um anzugeben, welche Protokolldateien erforderlich sind:

- Die Nachricht AMQ7467 gibt den Namen der ältesten Protokolldatei an, die zum erneuten Starten des Warteschlangenmanagers erforderlich ist. Diese Protokolldatei und alle neueren Protokolldateien müssen beim Neustart des Warteschlangenmanagers verfügbar sein.
- Die Nachricht AMQ7468 gibt den Namen der ältesten Protokolldatei an, die für die Datenträgerwiederherstellung benötigt wird.

Nur Protokolldateien, die für den Neustart des Warteschlangenmanagers erforderlich sind, sind für aktive Protokolldateien erforderlich. Inaktive Protokolldateien können in ein Archivierungsmedium, wie z. B. ein Band für die Wiederherstellung nach einem Katastrophenfall, kopiert und aus dem Protokollverzeichnis entfernt werden. Inaktive Protokolldateien, die für die Datenträgerwiederherstellung nicht erforderlich sind, können als überflüssige Protokolldateien betrachtet werden. Sie können überflüssige Protokolldateien löschen, wenn sie für Ihre Operation nicht mehr von Interesse sind.

Um "ältere" und "neuere" Protokolldateien zu ermitteln, verwenden Sie die Protokolldateinummer und nicht die Änderungszeiten, die vom Dateisystem angewendet werden.

Wenn eine Protokolldatei nicht gefunden werden kann, wird die Bedienernachricht AMQ6767 ausgegeben. Stellen Sie die Protokolldatei und alle nachfolgenden Protokolldateien dem Warteschlangenmanager zur Verfügung und wiederholen Sie die Operation.

**Anmerkung:** Bei einer Datenträgerwiederherstellung müssen alle erforderlichen Protokolldateien im Protokollverzeichnis gleichzeitig verfügbar sein. Stellen Sie sicher, dass Sie regelmäßig Datenträgerimages von Objekten, die Sie wiederherstellen möchten, verwenden, um zu vermeiden, dass der Plattenspeicherplatz für alle erforderlichen Protokolldateien gehalten wird. Führen Sie den Befehl `rcdmqimg` wie in den folgenden Beispielen gezeigt aus, um ein Medienimage aller Objekte in Ihrem Warteschlangenmanager zu erstellen:

### Unter Windows:

```
rcdmqimg -m QMNAME -t all *
```

### Unter UNIX and Linux

```
rcdmqimg -m QMNAME -t all "*"
```

Wenn Sie `rcdmqimg` ausführen, wird die Folgenummer des Datenträgerprotokolls (LSN) weitergeleitet. Weitere Informationen zu Protokollfolgenummern finden Sie in [„Speicherauszug für den Inhalt des Protokolls mit dem Befehl 'dmpmqlog' erstellen“](#) auf Seite 444. `rcdmqimg` wird nicht automatisch ausgeführt und muss daher manuell oder über eine von Ihnen erstellte automatische Task ausgeführt werden. Weitere Informationen zu diesem Befehl finden Sie in den Informationen zu [rcdmqimg](#) und [dmpmqlog](#).

**Anmerkung:** Die Nachrichten AMQ7467 und AMQ7468 können auch beim Ausführen des Befehls `rcdmqimg` ausgegeben werden.

### *Überflüssige Protokolldateien ermitteln*

Bei der Verwaltung von linearen Protokolldateien ist es wichtig zu wissen, welche Dateien gelöscht oder archiviert werden können. Diese Informationen helfen Ihnen bei dieser Entscheidung.

Verwenden Sie die Änderungszeiten des Dateisystems nicht, um die "älteren" Protokolldateien zu ermitteln. Verwenden Sie nur die Protokolldateinummer. Die Verwendung von Protokolldateien durch den Warteschlangenmanager folgt komplexen Regeln, einschließlich der Vorzuweisung und Formatierung von Protokolldateien, bevor sie benötigt werden. Es werden möglicherweise Protokolldateien mit Änderungszeiten angezeigt, die irreführend wären, wenn Sie versuchen, diese Zeiten für die Bestimmung des relativen Alters zu verwenden.

Um die älteste Protokolldatei zu ermitteln, die für den Neustart des Warteschlangenmanagers erforderlich ist, geben Sie den Befehl `DISPLAY QMSTATUS RECLOGaus`.



Um die älteste Protokolldatei zu ermitteln, die für die Datenträgerwiederherstellung erforderlich ist, geben Sie den Befehl `DISPLAY QMSTATUS MEDIALOG` aus.

Im Allgemeinen impliziert eine niedrigere Protokolldateinummer ein älteres Protokoll. Es sei denn, Sie haben einen sehr hohen Protokolldateiumsatz von der Reihenfolge von 3000 Protokolldateien pro Tag für 10 Jahre, dann müssen Sie nicht für den Zahlenumlauf bei 9.999.999 sorgen. In diesem Fall können Sie jede Protokolldatei mit einer Zahl, die kleiner als der `RECLOG`-Wert ist, archivieren, und Sie können jede Protokolldatei mit einer Zahl löschen, die kleiner ist als die Werte für `RECLOG` und `MEDIALOG`.

Wenn Sie jedoch eine sehr hohe Fluktuation von Protokolldateien haben oder auf andere Weise sicher sein wollen, mit dem allgemeinen Fall fertig zu werden, kann der folgende Algorithmus verwendet werden:

S == Neustartprotokolldateinummer  
(aus `DISPLAY QMSTATUS RECLOG`).

M == Datenträgerwiederherstellungsprotokolldateinummer  
(von `DISPLAY QMSTATUS MEDIALOG`).

Lassen Sie L == eine Protokolldateinummer mit der Berechtigung zum Löschen oder Archivieren die bestimmt werden müssen.

```
Funktion minlog (a, b) {
 if (abs (a-b) < 5000000)
 return min (a, b); # Nicht eingeschlossen.
 else
 return max (a, b); # Umlauf.}
```

Eine Protokolldatei L kann gelöscht werden, wenn  
(L! = S && L! = M && minlog (L, minlog (S, M)) == L).

Eine Protokolldatei L kann archiviert werden, wenn  
(L! = S && minlog (L, S) == L).

#### *Position der Protokolldatei*

Beachten Sie bei der Auswahl einer Position für Ihre Protokolldateien, dass die Operation stark beeinträchtigt wird, wenn WebSphere MQ ein neues Protokoll nicht formatieren kann, weil nicht genügend Plattenspeicherplatz verfügbar ist.

Wenn Sie ein Umlaufprotokoll verwenden, stellen Sie sicher, dass auf dem Laufwerk genügend Speicherplatz für mindestens die konfigurierten primären Protokolldateien vorhanden ist. Geben Sie außerdem Speicherplatz für mindestens eine sekundäre Protokolldatei an, die benötigt wird, wenn das Protokoll wachsen muss.

Wenn Sie ein lineares Protokoll verwenden, können Sie erheblich mehr Speicherplatz in Anspruch haben. Der Speicherplatz, der vom Protokoll verbraucht wird, nimmt kontinuierlich zu, wenn Daten protokolliert werden.

Idealerweise platzieren Sie die Protokolldateien auf einem anderen Plattenlaufwerk als die WS-Manager-Daten. Dies hat Vorteile hinsichtlich der Leistung. Es kann auch möglich sein, die Protokolldateien auf mehreren Plattenlaufwerken in einer spiegelgleichen Anordnung zu platzieren. Dies schützt vor einem Ausfall des Laufwerks, das das Protokoll enthält. Ohne Spiegelung könnten Sie gezwungen sein, zur letzten Sicherung Ihres WebSphere MQ -Systems zurückzukehren.

## **Protokoll für Wiederherstellung verwenden**

Verwendung von Protokollen für die Wiederherstellung nach Fehlern.

Es gibt mehrere Möglichkeiten, Ihre Daten zu beschädigen. WebSphere MQ unterstützt Sie bei der Wiederherstellung von:

- Ein beschädigtes Datenobjekt
- Ein Stromausfall im System

- Ein Kommunikationsfehler

In diesem Abschnitt wird erläutert, wie die Protokolle für die Wiederherstellung nach diesen Problemen verwendet werden.

### **Wiederherstellung nach Stromausfall-oder Kommunikationsfehlern**

WebSphere MQ kann nach Kommunikationsfehlern und Stromausfällen wiederhergestellt werden. Manchmal ist auch bei anderen Arten von Problemen eine Wiederherstellung möglich, beispielsweise wenn eine Datei unbeabsichtigt gelöscht wurde.

Bei einem Kommunikationsfehler verbleiben Nachrichten in Warteschlangen, bis sie von einer empfangenden Anwendung entfernt werden. Wenn die Nachricht übertragen wird, bleibt sie in der Übertragungswarteschlange, bis sie erfolgreich übertragen werden kann. Für die Wiederherstellung nach einem Kommunikationsfehler können Sie die Kanäle in der Regel über den fehlgeschlagenen Link erneut starten.

Wenn Sie die Stromversorgung verlieren und der Warteschlangenmanager erneut gestartet wird, stellt WebSphere MQ die Warteschlangen zum Zeitpunkt des Fehlers in ihrem festgeschriebenen Status wieder her. Dadurch wird sichergestellt, dass keine persistenten Nachrichten verloren gehen. Nicht persistente Nachrichten werden gelöscht und bleiben nicht erhalten, wenn WebSphere MQ abrupt gestoppt wird.

### **Beschädigte Objekte wiederherstellen**

Es gibt Fälle, in denen ein IBM WebSphere MQ-Objekt unbrauchbar werden kann, z. B. wegen unbeabsichtigter Beschädigung. Sie müssen dann entweder Ihr vollständiges System oder einen Teil davon wiederherstellen. Die erforderliche Aktion hängt davon ab, wann der Schaden erkannt wird, ob die ausgewählte Protokollmethode die Datenträgerwiederherstellung unterstützt und welche Objekte beschädigt sind.

## **Datenträgerwiederherstellung**

Durch die Datenträgerwiederherstellung werden Objekte aus Informationen, die in einem linearen Protokoll aufgezeichnet wurden, erneut erstellt. Wenn beispielsweise eine Objektdatei versehentlich gelöscht oder aus einem anderen Grund nicht mehr verwendet werden kann, kann die Datenträgerwiederherstellung erneut erstellt werden. Die Informationen in dem Protokoll, die für die Datenträgerwiederherstellung eines Objekts erforderlich sind, werden als *Datenträgerimage* bezeichnet.

Ein Datenträgerimage ist eine Folge von Protokollsätzen, die ein Bild eines Objekts enthalten, aus dem das Objekt selbst neu erstellt werden kann.

Der erste Protokollsatz, der zum erneuten Erstellen eines Objekts erforderlich ist, wird als *Datenträgerwiederherstellungssatz* bezeichnet; er ist der Anfang des neuesten Medienimages für das Objekt. Der Datenträgerwiederherstellungsdatensatz eines jeden Objekts ist einer der Informationsteile, die während eines Prüfpunkts aufgezeichnet wurden.

Wenn ein Objekt aus seinem Medienimage neu erstellt wird, müssen auch alle Protokollsätze wiedergegeben werden, die die Aktualisierungen beschreiben, die seit dem letzten Abbild für das Objekt ausgeführt wurden.

Betrachten Sie zum Beispiel eine lokale Warteschlange, die ein Image des Warteschlangenobjekts enthält, bevor eine persistente Nachricht in die Warteschlange gestellt wird. Um das aktuellste Image des Objekts erneut zu erstellen, müssen die Protokolleinträge, die das Einreihen der Nachricht in die Warteschlange aufzeichnen, neu wiedergegeben werden, und die Wiedergabe des Images selbst wird nicht mehr angezeigt.

Wenn ein Objekt erstellt wird, enthalten die geschriebenen Protokollsätze genügend Informationen, um das Objekt vollständig neu zu erstellen. Diese Datensätze bilden das erste Medienimage des Objekts. Anschließend zeichnet der Warteschlangenmanager bei jedem Systemabschluss die Datenträgerimages automatisch wie folgt auf:

- Images aller Prozessobjekte und Warteschlangen, die nicht lokal sind
- Images von leeren lokalen Warteschlangen

Medienimages können auch manuell mit dem Befehl **rcdmqimg** aufgezeichnet werden, wie in [rcdmqimg](#) beschrieben. Dieser Befehl schreibt ein Datenträgerimage des IBM WebSphere MQ-Objekts. Wenn ein Datenträgerimage geschrieben wurde, sind nur die Protokolle, die das Datenträgerimage enthalten, und alle nach diesem Zeitpunkt erstellten Protokolle erforderlich, um beschädigte Objekte erneut zu erstellen. Der Vorteil der Erstellung von Datenträgerimages hängt von Faktoren wie der Menge des verfügbaren freien Speichers und der Geschwindigkeit ab, mit der Protokolldateien erstellt werden.

## Wiederherstellung von Medienimages

Ein Warteschlangenmanager stellt einige Objekte während des Starts des Warteschlangenmanagers automatisch von ihrem Medienimage wieder her. Eine Warteschlange wird automatisch wiederhergestellt, wenn sie an einer Transaktion beteiligt war, die beim letzten Abschalten des Warteschlangenmanagers unvollständig war und bei der Neustartverarbeitung als beschädigt oder fehlerhaft erkannt wird.

Sie müssen andere Objekte manuell wiederherstellen, indem Sie den Befehl **rcrmqobj** verwenden, der die Datensätze im Protokoll wiedergibt, um das IBM WebSphere MQ-Objekt erneut zu erstellen. Das Objekt wird aus dem zuletzt im Protokoll gefundenen Image zusammen mit allen anwendbaren Protokollereignissen zwischen dem Zeitpunkt, zu dem das Image gespeichert wurde, und dem Zeitpunkt, zu dem der Befehl zum Erstellen der Neuerstellungs-Datei ausgegeben wurde, erneut erstellt. Wenn ein IBM WebSphere MQ -Objekt beschädigt wird, sind die einzigen gültigen Aktionen, die ausgeführt werden können, entweder, um es zu löschen oder um es mit dieser Methode neu zu erstellen. Nicht persistente Nachrichten können auf diese Weise nicht wiederhergestellt werden.

Weitere Informationen zum Befehl **rcrmqobj** finden Sie unter [rcrmqobj](#).

Die Protokolldatei, die den Datenträgerwiederherstellungssatz enthält, und alle nachfolgenden Protokolldateien müssen im Protokolldateiverzeichnis verfügbar sein, wenn die Datenträgerwiederherstellung eines Objekts versucht wird. Wenn eine erforderliche Datei nicht gefunden werden kann, wird die Bedienernachricht AMQ6767 ausgegeben und die Datenträgerwiederherstellungsoperation schlägt fehl. Wenn Sie keine regulären Medienimages der Objekte verwenden, die Sie erneut erstellen möchten, haben Sie möglicherweise nicht genügend Plattenspeicherplatz, um alle Protokolldateien zu speichern, die für die erneute Erstellung eines Objekts erforderlich sind.

## Wiederanlauf beschädigter Objekte beim Start

Wenn der Warteschlangenmanager während des Starts ein beschädigtes Objekt erkennt, hängt die von ihm abhängige Aktion vom Typ des Objekts und davon ab, ob der Warteschlangenmanager für die Unterstützung der Datenträgerwiederherstellung konfiguriert ist.

Wenn das WS-Manager-Objekt beschädigt ist, kann der Warteschlangenmanager nicht gestartet werden, es sei denn, er kann das Objekt wiederherstellen. Wenn der Warteschlangenmanager mit einem linearen Protokoll konfiguriert ist und somit die Datenträgerwiederherstellung unterstützt, versucht IBM WebSphere MQ automatisch, das WS-Manager-Objekt aus seinen Medienimages erneut zu erstellen. Wenn die ausgewählte Protokollmethode keine Datenträgerwiederherstellung unterstützt, können Sie entweder eine Sicherung des Warteschlangenmanagers zurückschreiben oder den Warteschlangenmanager löschen.

Wenn Transaktionen aktiv waren, als der Warteschlangenmanager gestoppt wurde, sind auch die lokalen Warteschlangen, die die persistenten, nicht festgeschriebenen Nachrichten enthalten oder in diese Transaktionen eingingen, erforderlich, um den WS-Manager erfolgreich zu starten. Wenn eine dieser lokalen Warteschlangen beschädigt ist und der Warteschlangenmanager die Datenträgerwiederherstellung unterstützt, versucht er automatisch, sie aus ihren Medienimages erneut zu erstellen. Wenn eine der Warteschlangen nicht wiederhergestellt werden kann, kann IBM WebSphere MQ nicht gestartet werden.

Wenn beschädigte lokale Warteschlangen, die nicht festgeschriebene Nachrichten enthalten, während der Startverarbeitung auf einem Warteschlangenmanager erkannt werden, der keine Datenträgerwiederherstellung unterstützt, werden die Warteschlangen als beschädigte Objekte markiert und die nicht festgeschriebenen Nachrichten in ihnen werden ignoriert. Dies liegt daran, dass es nicht möglich ist, die Datenträgerwiederherstellung beschädigter Objekte in einem solchen Warteschlangenmanager auszuführen, und die einzige Aktion, die noch vorhanden ist, besteht darin, sie zu löschen. Die Nachricht AMQ7472 wird ausgegeben, um alle Schäden zu melden.

## Beschädigte Objekte zu einem anderen Zeitpunkt wiederherstellen

Die Datenträgerwiederherstellung von Objekten erfolgt automatisch nur während des Systemstarts. Zu einem anderen Zeitpunkt, wenn Objektbeschädigung festgestellt wird, wird die Bedienernachricht AMQ7472 ausgegeben, und die meisten Operationen, die das Objekt verwenden, schlagen fehl. Wenn das WS-Manager-Objekt zu einem beliebigen Zeitpunkt nach dem Start des Warteschlangenmanagers beschädigt ist, führt der Warteschlangenmanager eine bereits emptive Beendigung aus. Wenn ein Objekt beschädigt wurde, können Sie es löschen oder, wenn der Warteschlangenmanager ein lineares Protokoll verwendet, versuchen, es mit dem Befehl `rcrmqobj` aus seinem Datenträgerimage wiederherzustellen. Weitere Informationen finden Sie im Abschnitt [rcrmqobj](#).

## IBM WebSphere MQ -Protokolldateien schützen

Berühren Sie die Protokolldateien nicht, wenn ein Warteschlangenmanager ausgeführt wird. Eine Wiederherstellung ist möglicherweise nicht möglich. Verwenden Sie die Superuser-oder die `mqm`-Berechtigung zum Schutz von Protokolldateien vor unbeabsichtigtes Ändern.

Entfernen Sie die aktiven Protokolldateien nicht manuell, wenn ein IBM WebSphere MQ-Warteschlangenmanager ausgeführt wird. Wenn ein Benutzer die Protokolldateien, die ein Warteschlangenmanager erneut starten muss, versehentlich löscht, gibt IBM WebSphere MQ **keine** Fehler aus und setzt die Verarbeitung der Daten *einschließlich persistenter Nachrichten* fort. Der WS-Manager wird normal beendet, kann aber nicht erneut gestartet werden. Die Wiederherstellung von Nachrichten wird dann nicht mehr möglich.

Benutzer mit der Berechtigung zum Entfernen von Protokollen, die von einem aktiven Warteschlangenmanager verwendet werden, haben auch die Berechtigung zum Löschen anderer wichtiger WS-Manager-Ressourcen (z. B. Warteschlangendateien, Objektkatalog und ausführbare IBM WebSphere MQ-Dateien). Sie können daher beispielsweise aus Unerfahrenheit einen aktiven oder ruhenden WS-Manager in einer Weise beschädigen, die IBM WebSphere MQ nicht selbst schützen kann.

Gehen Sie bei der Übertragung von Superuser-oder `mqm`-Berechtigungen vorsichtig vor.

## Speicherauszug für den Inhalt des Protokolls mit dem Befehl 'dmpmqlog' erstellen

Verwendung des Befehls `dmpmqlog` zum Erstellen eines Speicherauszugs des Warteschlangenmanagerprotokolls.

Verwenden Sie den Befehl `dmpmqlog`, um einen Speicherauszug des Inhalts des Warteschlangenmanagerprotokolls zu erstellen. Standardmäßig wird ein Speicherauszug aller aktiven Protokollsätze erstellt, d. a. der Befehl startet das Erstellen eines Speicherauszugs aus dem Protokollkopf (in der Regel der Anfang des letzten abgeschlossenen Prüfpunkts).

Für das Protokoll kann in der Regel nur ein Speicherauszug erstellt werden, wenn der Warteschlangenmanager nicht aktiv ist. Da der Warteschlangenmanager während der Beendigung einen Prüfpunkt nimmt, enthält der aktive Teil des Protokolls normalerweise eine kleine Anzahl Protokollsätze. Sie können jedoch mit dem Befehl `dmpmqlog` einen Speicherauszug für weitere Protokollsätze erstellen, indem Sie eine der folgenden Optionen verwenden, um die Startposition des Speicherauszugs zu ändern:

- Starten Sie das Dumping von der *Basis* des Protokolls. Die Basis des Protokolls ist der erste Protokollsatz in der Protokolldatei, der den Kopf des Protokolls enthält. Die Menge der zusätzlichen Daten, die in diesem Fall erstellt werden, hängt davon ab, wo sich der Protokollkopf in der Protokolldatei befindet. Wenn die Protokolldatei am Anfang der Protokolldatei steht, wird nur ein kleiner Teil der zusätzlichen Daten erstellt. Befindet sich der Kopf am Ende der Protokolldatei, wird ein Speicherauszug für deutlich mehr Daten erstellt.
- Geben Sie die Startposition des Speicherauszugs als einzelnen Protokollsatz an. Jeder Protokollsatz wird durch eine eindeutige *Protokollfolgennummer (Log Sequence Number, LSN)* identifiziert. Bei der Umlaufprotokollierung kann dieser Startprotokollsatz nicht vor der Basis des Protokolls liegen. Diese Einschränkung gilt nicht für lineare Protokolle. Sie müssen möglicherweise inaktive Protokolldateien erneut instanzieren, bevor Sie den Befehl ausführen. Sie müssen eine gültige Protokollfolgennummer (LSN) aus der vorherigen `dmpmqlog`-Ausgabe als Startposition angeben.

Bei der linearen Protokollierung können Sie beispielsweise die `nextlsn` aus Ihrer letzten `dmpmqlog`-Ausgabe angeben. Der `nextlsn` wird in `Log File Header` angezeigt und gibt die Protokollfolgennummer des nächsten Protokollsatzes an, der geschrieben werden soll. Verwenden Sie diese Funktion als Startposition, um alle Protokollsätze zu formatieren, die seit der letzten Erstellung des Protokollspeichers ausgeschrieben wurden.

- **Nur für lineare Protokolle:** Sie können `dmpmqlog` anweisen, mit der Formatierung von Protokollsätzen aus einem bestimmten Protokolldateibereich zu beginnen. In diesem Fall erwartet `dmpmqlog`, dass sich diese Protokolldatei und alle nachfolgenden Protokolldateien in demselben Verzeichnis wie die aktiven Protokolldateien befinden. Diese Option gilt nicht für Umlaufprotokolle, bei denen `dmpmqlog` vor der Protokollbasis nicht auf Protokollsätze zugreifen kann.

Die Ausgabe des Befehls `dmpmqlog` ist `Log File Header` und eine Reihe formatierter Protokollsätze. Der WS-Manager verwendet mehrere Protokollsätze, um Änderungen an seinen Daten zu erfassen.

Ein Teil der Informationen, die formatiert werden, ist nur intern verwendet. Die folgende Liste enthält die nützlichsten Protokollsätze:

### Protokolldateiheader

Jedes Protokoll hat einen einzelnen Protokolldateiheader, der immer das erste ist, was vom Befehl `dmpmqlog` formatiert wird. Sie enthält die folgenden Felder:

|                       |                                                                                                             |
|-----------------------|-------------------------------------------------------------------------------------------------------------|
| <i>logactive</i>      | Die Anzahl primärer Protokollspeicherbereiche.                                                              |
| <i>loginactive</i>    | Die Anzahl sekundärer Protokollerweiterungen.                                                               |
| <i>Protokollgröße</i> | Die Anzahl der 4-KB-Seiten pro Speicherbereich.                                                             |
| <i>Baselsn</i>        | Die erste Protokollfolgennummer (LSN) in der Protokollspeicherbereichsdatei, die den Protokollkopf enthält. |
| <i>nextlsn</i>        | Die Protokollfolgennummer (LSN) des nächsten Protokollsatzes, der geschrieben werden soll.                  |
| <i>Headlsn</i>        | Die Protokollfolgennummer (LSN) des Protokollsatzes am Kopf des Protokolls.                                 |
| <i>tailsn</i>         | Die Protokollfolgennummer (LSN), die die Endposition des Protokolls angibt.                                 |
| <i>hflag1</i>         | Gibt an, ob das Protokoll CIRCULAR oder LOG RETAIN (linear) ist.                                            |
| <i>HeadExtent-ID</i>  | Die Protokollspeicherbereichsdatei, die den Protokollkopf enthält.                                          |

### Protokollsatz-Header

Jeder Protokollsatz innerhalb des Protokolls hat einen festen Header mit den folgenden Informationen:

|                              |                                                                                                                                                                                                                                                                                                                                                 |
|------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Protokollfolgennummer</i> | Die Protokollfolgennummer.                                                                                                                                                                                                                                                                                                                      |
| <i>LogRecdTyp</i>            | Der Typ des Protokollsatzes.                                                                                                                                                                                                                                                                                                                    |
| <i>XTranid</i>               | Die Transaktions-ID, die diesem Protokollsatz zugeordnet ist (falls vorhanden).<br><br>Der Wert <i>TranType</i> für MQI gibt eine reine WebSphere MQ-Transaktion an. Ein <i>TranType</i> von XA ist an anderen Ressourcenmanagern beteiligt. Aktualisierungen, die in derselben Arbeitseinheit enthalten sind, haben denselben <i>XTranid</i> . |
| <i>QueueName</i>             | Die Warteschlange, die diesem Protokollsatz zugeordnet ist (falls vorhanden).                                                                                                                                                                                                                                                                   |
| <i>QID</i>                   | Die eindeutige interne Kennung für die Warteschlange.                                                                                                                                                                                                                                                                                           |

*PrevLSN* Die Protokollfolgennummer (LSN) des vorherigen Protokollsatzes innerhalb derselben Transaktion (falls vorhanden).

### **WS-Manager starten**

Diese Protokolle, die der Warteschlangenmanager gestartet hat.

*StartDate* Das Datum, an dem der WS-Manager gestartet wurde.

*StartTime* Die Zeit, zu der der WS-Manager gestartet wurde.

### **Warteschlangenmanager stoppen**

Diese Protokolle, die der Warteschlangenmanager gestoppt hat, wurden gestoppt.

*StopDate* Das Datum, an dem der WS-Manager gestoppt wurde.

*StopTime* Die Zeit, zu der der WS-Manager gestoppt wurde.

*ForceFlag* Der Typ des verwendeten Systemabschlusses.

### **Prüfpunkt starten**

Gibt den Start eines Warteschlangenmanagerprüfpunkts an.

### **Prüfpunkt beenden**

Gibt das Ende eines Warteschlangenmanagerprüfpunkts an.

*ChkPtLSN* Die Protokollfolgennummer (LSN) des Protokollsatzes, der diesen Prüfpunkt gestartet hat.

### **Nachricht einreihen**

Diese Nachricht protokolliert eine persistente Nachricht, die in eine Warteschlange gestellt wird. Wenn die Nachricht unter Synchronisationspunkt gesetzt wurde, enthält der Protokollsatz-Header einen Nicht-Nullwert *XTranId*. Der Rest des Datensatzes enthält:

*MapIndex* Eine Kennung für die Nachricht in der Warteschlange. Sie kann verwendet werden, um die entsprechende MQGET abzugleichen, die zum Abrufen dieser Nachricht aus der Warteschlange verwendet wurde. In diesem Fall kann ein nachfolgender *Get Message* -Protokollsatz gefunden werden, der die gleichen QueueName und MapIndex enthält. Zu diesem Zeitpunkt kann die MapIndex -ID für eine nachfolgende Nachricht in diese Warteschlange wiederverwendet werden.

*Daten* Im hexadezimalen Speicherauszug für diesen Protokollsatz befinden sich verschiedene interne Daten, gefolgt vom Nachrichtendeskriptor (eyecatcher MD) und den Nachrichtendaten selbst.

### **Teil einreihen**

Persistente Nachrichten, die zu groß für einen einzelnen Protokollsatz sind, werden als mehrere *Put Part*-Protokollsätze gefolgt von einem einzelnen *Put Message*-Datensatz protokolliert. Wenn *Put Part* -Datensätze vorhanden sind, verkettet das Feld *PrevLSN* die *Put Part* -Datensätze und den endgültigen *Put Message* -Datensatz.

*Daten* Setzt die Nachrichtendaten da fort, wo der vorherige Protokollsatz aufgehört hatte.

### **Nachricht abrufen**

Es werden nur persistente Nachrichten protokolliert. Wenn die Nachricht unter Synchronisationspunkt wurde, enthält der Protokollsatz-Header einen Nicht-Nullwert *XTranId*. Der Rest des Datensatzes enthält:

|                  |                                                                                                                                                                                                                                |
|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>MapIndex</i>  | Bezeichnet die Nachricht, die aus der Warteschlange abgerufen wurde. Der aktuellste <i>Put Message</i> -Protokollsatz, der denselben <i>QueueName</i> und <i>MapIndex</i> enthält, gibt die Nachricht an, die abgerufen wurde. |
| <i>QPriority</i> | Die Priorität der Nachricht, die aus der Warteschlange abgerufen wurde.                                                                                                                                                        |

### Transaktion starten

Gibt den Start einer neuen Transaktion an. Der TranType MQI gibt eine reine WebSphere MQ-Transaktion an. Ein TranType von XA gibt an, dass andere Ressourcenmanager involviert sind. Alle Aktualisierungen, die von dieser Transaktion vorgenommen werden, haben denselben *XTranid*.

### Transaktion vorbereiten

Gibt an, dass der WS-Manager bereit ist, die Aktualisierungen festzuschreiben, die dem angegebenen *XTranid* zugeordnet sind. Dieser Protokollsatz wird als Teil einer zweiphasige Commit mit anderen Ressourcenmanagern geschrieben.

### Commit-Transaktion

Gibt an, dass der Warteschlangenmanager alle Aktualisierungen festgeschrieben hat, die von einer Transaktion vorgenommen wurden.

### Rollback-Transaktion

Gibt die Absicht des WS-Managers an, eine Transaktion rückgängig zu machen.

### Transaktion beenden

Gibt das Ende einer rückgängig gewickelten Transaktion an.

### Transaktionstabelle

Dieser Datensatz wird während des Synchronisationspunkts geschrieben. Er zeichnet den Status jeder Transaktion auf, die persistente Aktualisierungen vorgenommen hat. Für jede Transaktion werden die folgenden Informationen aufgezeichnet:

|                 |                                                                                                  |
|-----------------|--------------------------------------------------------------------------------------------------|
| <i>XTranid</i>  | Die Transaktions-ID.                                                                             |
| <i>FirstLSN</i> | Die Protokollfolgennummer (LSN) des ersten Protokollsatzes, der der Transaktion zugeordnet ist.  |
| <i>LastLSN</i>  | Die Protokollfolgennummer (LSN) des letzten Protokollsatzes, der der Transaktion zugeordnet ist. |

### Transaktionsteilnehmer

Dieser Protokollsatz wird von der Komponente "XA Transaction Manager" des Warteschlangenmanagers geschrieben. Es zeichnet die externen Ressourcenmanager auf, die an Transaktionen beteiligt sind. Für jeden Teilnehmer werden die folgenden Informationen aufgezeichnet:

|                      |                                                                                                                                                                                                                                |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>RM-Name</i>       | Der Name des Ressourcenmanagers.                                                                                                                                                                                               |
| <i>RMID</i>          | Die Kennung des Ressourcenmanagers. Dies wird auch in nachfolgenden <i>Transaction Prepared</i> -Protokollsätzen protokolliert, in denen globale Transaktionen aufgezeichnet werden, an denen der Ressourcenmanager teilnimmt. |
| <i>SwitchFile</i>    | Die Switchloaddatei für diesen Ressourcenmanager.                                                                                                                                                                              |
| <i>XAOpenString</i>  | Die offene XA-Zeichenfolge für diesen Ressourcenmanager.                                                                                                                                                                       |
| <i>XACloseString</i> | Die XA-Zeichenfolge für diesen Ressourcenmanager.                                                                                                                                                                              |

### Transaktion vorbereitet

Dieser Protokollsatz wird von der Komponente "XA Transaction Manager" des Warteschlangenmanagers geschrieben. Sie zeigt an, dass die angegebene globale Transaktion erfolgreich vorbereitet wurde. Jeder der beteiligten Ressourcenmanager wird angewiesen, sich festzuschreiben. Der *RMID* jedes vorbereiteten Ressourcenmanagers wird im Protokollsatz aufgezeichnet. Wenn der WS-Manager selbst an der Transaktion beteiligt ist, wird ein *Participant Entry* mit einem *RMID* von null vorhanden sein.

### Transaktionsforget

Dieser Protokollsatz wird von der Komponente "XA Transaction Manager" des Warteschlangenmanagers geschrieben. Sie folgt dem Protokollsatz *Transaction Prepared*, wenn die Festschreibungsentscheidung an jeden Teilnehmer zugestellt wurde.

### Bereinigungs-warteschlange

In diesem Protokoll wird die Tatsache protokolliert, dass alle Nachrichten in einer Warteschlange gelöscht wurden, z. B. mit dem MQSC-Befehl CLEAR QUEUE.

### Warteschlangenattribute

Dadurch wird die Initialisierung oder Änderung der Attribute einer Warteschlange protokolliert.

### Objekt erstellen

Dies protokolliert die Erstellung eines WebSphere MQ -Objekts.

|                |                                               |
|----------------|-----------------------------------------------|
| <i>ObjName</i> | Der Name des Objekts, das erstellt wurde.     |
| <i>UserId</i>  | Die Benutzer-ID, die die Erstellung ausführt. |

### Objekt löschen

Dies protokolliert das Löschen eines WebSphere MQ -Objekts.

|                |                                           |
|----------------|-------------------------------------------|
| <i>ObjName</i> | Der Name des Objekts, das gelöscht wurde. |
|----------------|-------------------------------------------|

## IBM WebSphere MQ-Warteschlangenmanagerdaten sichern und wiederherstellen

Sicherung von Warteschlangenmanagern und Warteschlangenmanagerdaten.

In regelmäßigen Abständen können Sie Maßnahmen ergreifen, um WS-Manager vor möglichen Beschädigungen durch Hardwarefehler zu schützen. Es gibt drei Möglichkeiten zum Schutz eines Warteschlangenmanagers:

### Sichern Sie die WS-Manager-Daten.

Wenn die Hardware fehlschlägt, wird möglicherweise ein WS-Manager gestoppt. Wenn die Protokoll-daten eines Warteschlangenmanagers aufgrund des Hardwarefehls verloren gehen, kann der WS-Manager möglicherweise nicht erneut gestartet werden. Wenn Sie WS-Manager-Daten sichern, können Sie möglicherweise einige oder alle verloren gegangenen WS-Manager-Daten wiederherstellen.

Im Allgemeinen sind die weniger Daten, die Sie im Falle eines Hardwarefehls verloren haben, bei einem Hardwarefehler, der zu einem Verlust der Integrität des Wiederherstellungsprotokolls führt, im Allgemeinen die weniger Daten, die Sie sichern.

Damit WS-Manager-Daten gesichert werden können, muss der Warteschlangenmanager nicht aktiv sein.

Informationen zum Sichern und Wiederherstellen von WS-Manager-Daten finden Sie unter:

- „WS-Manager-Daten sichern“ auf Seite 449.
- „WS-Manager-Daten zurückschreiben“ auf Seite 450.

### Sicherungswarteschlangenmanager verwenden

Wenn der Hardwarefehler schwer wiegend ist, kann ein Warteschlangenmanager möglicherweise nicht wiederhergestellt werden. Wenn der nicht wiederherstellbare Warteschlangenmanager in dieser Situation über einen dedizierten Sicherungswarteschlangenmanager verfügt, kann der Sicherungswarteschlangenmanager an Stelle des nicht wiederherstellbaren Warteschlangenmanagers aktiviert werden. Bei regelmäßiger Aktualisierung kann das Protokoll des Sicherungswarteschlangenmanagers Protokoll-daten enthalten, die das letzte vollständige Protokoll des nicht wiederherstellbaren Warteschlangenmanagers umfassen.

Ein Sicherungswarteschlangenmanager kann aktualisiert werden, während der vorhandene WS-Manager noch aktiv ist.



Informationen zum Erstellen und Aktivieren eines Sicherungswarteschlangenmanagers finden Sie in folgenden Abschnitten:

- [„Sicherungswarteschlangenmanager erstellen“](#) auf Seite 451.
- [„Sicherungswarteschlangenmanager starten“](#) auf Seite 452.

### **Nur die Konfiguration des WS-Managers sichern**

Wenn die Hardware fehlschlägt, wird möglicherweise ein WS-Manager gestoppt. Falls bei dem Hardwarefehler sowohl die Konfigurations- als auch die Protokolldaten des Warteschlangenmanagers verloren gehen, kann der Warteschlangenmanager weder neu gestartet noch anhand des Protokolls wiederhergestellt werden. Wenn Sie die Warteschlangenmanagerkonfiguration sichern, könnten Sie den Warteschlangenmanager und alle zugehörigen Objekte anhand der gespeicherten Definitionen neu erstellen.

Damit die Warteschlangenmanagerkonfiguration gesichert werden kann, muss der Warteschlangenmanager aktiv sein.

Informationen zum Sichern und Wiederherstellen der WS-Manager-Konfiguration finden Sie unter:

- [„WS-Manager-Konfiguration sichern“](#) auf Seite 453
- [„Warteschlangenmanagerkonfiguration wird zurückgespeichert“](#) auf Seite 453

### **WS-Manager-Daten sichern**

Das Sichern von WS-Manager-Daten kann Ihnen dabei helfen, den möglichen Datenverlust durch Hardwarefehler zu schützen.

### **Vorbereitende Schritte**

Stellen Sie sicher, dass der Warteschlangenmanager nicht aktiv ist. Wenn Sie versuchen, einen aktiven Warteschlangenmanager zu sichern, ist die Sicherungskopie unter Umständen nicht konsistent, wenn beim Kopieren der Dateien gerade eine Aktualisierung vorgenommen wurde. Falls möglich, stoppen Sie Ihren Warteschlangenmanager, indem Sie den Befehl `endmqm -w` ausführen (eine Beendigung im Wartestatus). Verwenden Sie nur dann, wenn dies fehlschlägt, den Befehl `endmqm -i` (eine sofortige Beendigung).

### **Informationen zu diesem Vorgang**

Führen Sie die folgenden Tasks aus, um eine Sicherungskopie der Daten eines Warteschlangenmanagers zu erstellen:

1. Suchen Sie nach den Verzeichnissen, unter denen der WS-Manager seine Daten und seine Protokolldateien mit den Informationen in den Konfigurationsdateien platziert. Weitere Informationen finden Sie unter [„Konfigurationsinformationen für IBM WebSphere MQ und Warteschlangenmanager ändern“](#) auf Seite 454.

**Anmerkung:** Möglicherweise haben Sie Schwierigkeiten, die Namen zu verstehen, die im Verzeichnis angezeigt werden. Die Namen werden umgesetzt, um sicherzustellen, dass sie mit der Plattform kompatibel sind, auf der Sie WebSphere MQ verwenden. Weitere Informationen zu Namensumsetzungen finden Sie unter [Understanding WebSphere MQ file names](#).

2. Übernehmen Sie Kopien aller Daten- und Protokolldateiverzeichnisse des Warteschlangenmanagers, einschließlich aller Unterverzeichnisse.

Stellen Sie sicher, dass Sie keine Dateien, insbesondere die Protokollsteuerdatei, wie in [„Wie Logs aussehen“](#) auf Seite 432 beschrieben, und die Konfigurationsdateien, wie im Abschnitt [„Initialisierungs- und Konfigurationsdateien“](#) auf Seite 76 beschrieben, vergessen. Einige der Verzeichnisse sind möglicherweise leer, aber Sie benötigen alle Verzeichnisse, um die Sicherung zu einem späteren Zeitpunkt zurückschreiben zu können.

3. Die Eignerschaften der Dateien beibehalten. Für WebSphere MQ für UNIX and Linux -Systeme können Sie dies mit dem Befehl `tar tun`. (Wenn Sie über Warteschlangen mit mehr als 2 GB verfügen, können

Sie den Befehl `tar` nicht verwenden. Weitere Informationen hierzu finden Sie im Abschnitt [Große Warteschlangen aktivieren](#).

**Anmerkung:** Wenn Sie ein Upgrade auf WebSphere MQ Version 7.5 und höher durchführen, stellen Sie sicher, dass Sie eine Sicherung der Datei `.ini` und der Registry-Einträge erstellen. Die Warteschlangenmanagerinformationen werden in der Datei `.ini` gespeichert und können zum Zurücksetzen auf eine frühere Version von WebSphere MQ verwendet werden.

## WS-Manager-Daten zurückschreiben

Führen Sie die folgenden Schritte aus, um eine Sicherung der Daten eines Warteschlangenmanagers wiederherzustellen.

### Vorbereitende Schritte

Stellen Sie sicher, dass der Warteschlangenmanager nicht aktiv ist.

### Informationen zu diesem Vorgang

Gehen Sie wie folgt vor, um eine Sicherung der Daten eines Warteschlangenmanagers wiederherzustellen:

1. Suchen Sie die Verzeichnisse, unter denen der WS-Manager seine Daten und seine Protokolldateien platziert, indem Sie die Informationen in den Konfigurationsdateien verwenden.
2. Leeren Sie die Verzeichnisse, in die Sie die Sicherungsdaten stellen wollen.
3. Kopieren Sie die Daten des Sicherungswarteschlangenmanagers und die Protokolldateien an die richtigen Stellen.
4. Aktualisieren Sie die Konfigurationsinformationsdateien.

Überprüfen Sie die sich ergebende Verzeichnisstruktur, um sicherzustellen, dass alle erforderlichen Verzeichnisse vorhanden sind.

Weitere Informationen zu IBM WebSphere MQ -Verzeichnissen und -Unterverzeichnissen finden Sie unter [Verzeichnisstruktur auf Windows -Systemen](#) und [Verzeichnisinhalt auf UNIX and Linux -Systemen](#).

Stellen Sie sicher, dass Sie über eine Protokollsteuerdatei sowie über die Protokolldateien verfügen. Stellen Sie außerdem sicher, dass die IBM WebSphere MQ -und Warteschlangenmanagerkonfigurationsdateien konsistent sind, damit WebSphere MQ an den richtigen Stellen nach den wiederhergestellten Daten suchen kann.

Sichern Sie bei der Umlaufprotokollierung die WS-Manager-Daten und die Protokolldateiverzeichnisse gleichzeitig, so dass Sie eine konsistente Gruppe von WS-Manager-Daten und -Protokollen wiederherstellen können.

Zur linearen Protokollierung sichern Sie die WS-Manager-Daten und die Protokolldateiverzeichnisse zur gleichen Zeit. Es ist möglich, nur die WS-Manager-Datendateien zurückzuspeichern, wenn eine entsprechende vollständige Folge von Protokolldateien verfügbar ist.

**Anmerkung:** Wenn Sie ein Upgrade auf WebSphere MQ Version 7.5 und höher durchführen, stellen Sie sicher, dass Sie eine Sicherung der Datei `.ini` und der Registry-Einträge erstellen. Die Warteschlangenmanagerinformationen werden in der Datei `.ini` gespeichert und können zum Zurücksetzen auf eine frühere Version von WebSphere MQ verwendet werden.

### Ergebnisse

Wenn die Daten ordnungsgemäß gesichert und wiederhergestellt wurden, wird der WS-Manager jetzt gestartet.

### Sicherungswarteschlangenmanager verwenden

Zu einem vorhandenen Warteschlangenmanager kann es einen dedizierten Sicherungswarteschlangenmanager geben.

Ein Sicherungswarteschlangenmanager ist eine inaktive Kopie des vorhandenen Warteschlangenmanagers. Wenn der vorhandene Warteschlangenmanager aufgrund eines schwerwiegenden Hardwarefehlers nicht wiederhergestellt werden kann, kann der Sicherungswarteschlangenmanager online gebracht werden, um den nicht wiederherstellbaren WS-Manager zu ersetzen.

Die vorhandenen Warteschlangenmanagerprotokolldateien müssen regelmäßig in den Sicherungswarteschlangenmanager kopiert werden, um sicherzustellen, dass der Sicherungswarteschlangenmanager eine effektive Methode für die Wiederherstellung nach einem Katastrophenfall bleibt. Der vorhandene Warteschlangenmanager muss nicht gestoppt werden, damit die Protokolldateien kopiert werden. Sie sollten jedoch nur dann eine Protokolldatei kopieren, wenn der WS-Manager die Schreibversuche abgeschlossen hat. Da das vorhandene Warteschlangenmanagerprotokoll ständig aktualisiert wird, gibt es immer eine geringfügige Diskrepanz zwischen dem vorhandenen WS-Managerprotokoll und den Protokolldaten, die in das Sicherungs-WS-Managerprotokoll kopiert werden. Durch regelmäßige Aktualisierungen des Sicherungswarteschlangenmanagers wird die Diskrepanz zwischen den beiden Protokollen minimiert.

Wenn ein Sicherungswarteschlangenmanager benötigt wird, um ihn online zu stellen, muss er aktiviert und dann gestartet werden. Die Voraussetzung, einen Sicherungswarteschlangenmanager vor dem Start zu aktivieren, ist eine vorbeugende Maßnahme zum Schutz vor einem versehentlichen Start eines Sicherungswarteschlangenmanagers. Sobald ein Sicherungswarteschlangenmanager aktiviert ist, kann er nicht länger aktualisiert werden.

Weitere Informationen zum Erstellen, Aktualisieren und Starten eines Sicherungswarteschlangenmanagers finden Sie in den folgenden Themen:

- [„Sicherungswarteschlangenmanager erstellen“ auf Seite 451](#)
- [„Sicherungswarteschlangenmanager aktualisieren“ auf Seite 452](#)
- [„Sicherungswarteschlangenmanager starten“ auf Seite 452](#)

## Sicherungswarteschlangenmanager erstellen

Sie können einen Sicherungswarteschlangenmanager nur verwenden, wenn Sie die lineare Protokollierung verwenden.

Gehen Sie wie folgt vor, um einen Sicherungswarteschlangenmanager für einen vorhandenen Warteschlangenmanager zu erstellen:

1. Erstellen Sie mit dem Steuerbefehl `crtmqmeinen` Sicherungswarteschlangenmanager für den vorhandenen Warteschlangenmanager. Für den Sicherungswarteschlangenmanager ist Folgendes erforderlich:
  - Damit werden dieselben Attribute wie der vorhandene Warteschlangenmanager verwendet, z. B. der Name des Warteschlangenmanagers, der Protokollierungstyp und die Größe der Protokolldatei.
  - Befinden Sie sich auf derselben Plattform wie der vorhandene Warteschlangenmanager.
  - Die Codeversion ist gleich oder höher als die Codeversion des vorhandenen Warteschlangenmanagers.
2. Übernehmen Sie Kopien aller vorhandenen Daten- und Protokolldateiverzeichnisse des vorhandenen Warteschlangenmanagers, einschließlich aller Unterverzeichnisse, wie in [„WS-Manager-Daten sichern“ auf Seite 449](#) beschrieben.
3. Überschreiben Sie die Daten- und Protokolldateiverzeichnisse des Sicherungswarteschlangenmanagers, einschließlich aller Unterverzeichnisse, mit den Kopien, die aus dem vorhandenen Warteschlangenmanager übernommen wurden.
4. Führen Sie den folgenden Steuerbefehl auf dem Sicherungswarteschlangenmanager aus:

```
stmqm -i BackupQMName
```

Dieser Parameter markiert den Warteschlangenmanager als Sicherungswarteschlangenmanager in WebSphere MQ und gibt alle kopierten Protokollspeicherbereiche wieder, um den Sicherungswarteschlangenmanager in Schritt mit dem vorhandenen Warteschlangenmanager zu bringen.

## Sicherungswarteschlangenmanager aktualisieren

Um sicherzustellen, dass ein Sicherungswarteschlangenmanager eine effektive Methode für die Wiederherstellung nach einem Katastrophenfall bleibt, muss er regelmäßig aktualisiert werden.

Durch die regelmäßige Aktualisierung wird die Diskrepanz zwischen dem Backup-WS-Manager-Protokoll und dem aktuellen WS-Manager-Protokoll verringert. Es ist nicht nötig, den Warteschlangenmanager für eine Sicherung zu stoppen.

Gehen Sie wie folgt vor, um einen Sicherungswarteschlangenmanager zu aktualisieren:

1. Geben Sie den folgenden Scriptbefehl (MQSC) für den Warteschlangenmanager aus, der gesichert werden soll:

```
RESET QMGR TYPE(ADVANCELOG)
```

Dadurch wird das Schreiben in das aktuelle Protokoll gestoppt und die Protokollierung des Warteschlangenmanagers in den nächsten Protokollspeicherbereich fortgeschrieben. Dadurch wird sichergestellt, dass alle Informationen, die bis zur aktuellen Zeit protokolliert wurden, gesichert werden.

2. Rufen Sie die (neue) aktuelle Nummer des aktiven Protokollspeicherbereichs ab, indem Sie den folgenden Scriptbefehl (MQSC) auf dem zu sicherenden WS-Manager absetzen:

```
DIS QMSTATUS CURRLOG
```

3. Kopieren Sie die aktualisierten Protokollspeicherbereichsdateien aus dem Protokollverzeichnis des aktuellen Warteschlangenmanagers in das Protokollverzeichnis des Sicherungswarteschlangenmanagers-kopieren Sie alle Protokollspeicherbereiche seit der letzten Aktualisierung und bis (aber nicht einschließlich) zu dem in Schritt 2 angegebenen aktuellen Speicherbereich. Kopieren Sie nur Protokollspeicherbereichsdateien, die mit "S. .." beginnen.

4. Geben Sie auf dem Sicherungswarteschlangenmanager den folgenden Steuerbefehl aus:

```
strmqm -r BackupQMName
```

Dadurch werden alle kopierten Protokollreplays und der Sicherungswarteschlangenmanager mit dem Warteschlangenmanager in Schritt gebracht. Wenn die Wiedergabe abgeschlossen ist, empfangen Sie eine Nachricht, die alle für die Neustartwiederherstellung erforderlichen Protokollweiterungen und alle für die Datenträgerwiederherstellung erforderlichen Protokollspeicherbereiche enthält.

**Warnung:** Wenn Sie eine non-contiguous -Protokollgruppe in das Protokollverzeichnis des Sicherungswarteschlangenmanagers kopieren, werden nur die Protokolle bis zu dem Punkt wiedergegeben, an dem das erste fehlende Protokoll gefunden wird.

## Sicherungswarteschlangenmanager starten

Sie können einen Sicherungswarteschlangenmanager für einen nicht wiederherstellbaren Warteschlangenmanager ersetzen.

Gehen Sie hierfür wie folgt vor:

1. Führen Sie den folgenden Steuerbefehl aus, um den Sicherungswarteschlangenmanager zu aktivieren:

```
strmqm -a BackupQMName
```

Der Sicherungswarteschlangenmanager ist aktiviert. Jetzt, da er aktiv ist, kann der Sicherungswarteschlangenmanager nicht mehr aktualisiert werden.

2. Führen Sie den folgenden Steuerbefehl aus, um den Sicherungswarteschlangenmanager zu starten:

```
strmqm BackupQMName
```

WebSphere MQ betrachtet dies als Neustartwiederherstellung und verwendet das Protokoll des Sicherungswarteschlangenmanagers. Während der letzten Aktualisierung für den Sicherungswarteschlangenmanager ist eine Wiedergabe aufgetreten, daher werden nur die aktiven Transaktionen vom zuletzt aufgezeichneten Prüfpunkt rückgängig gemacht.

Wenn ein nicht wiederherstellbarer Warteschlangenmanager durch einen Sicherungswarteschlangenmanager ersetzt wird, können einige der Warteschlangenmanagerdaten aus dem nicht wiederherstellbaren Warteschlangenmanager verloren gehen. Die Menge der verloren gegangenen Daten hängt davon ab, wie kürzlich der Sicherungswarteschlangenmanager zuletzt aktualisiert wurde. Je mehr zuletzt die letzte Aktualisierung durchgeführt wurde, der Verlust des Datenverlusts in weniger Warteschlangenmanagern.

3. Starten Sie alle Kanäle erneut.

Überprüfen Sie die sich ergebende Verzeichnisstruktur, um sicherzustellen, dass alle erforderlichen Verzeichnisse vorhanden sind.

Weitere Informationen zu WebSphere MQ -Verzeichnissen und -Unterverzeichnissen finden Sie unter [Dateisystemunterstützung planen](#).

Stellen Sie sicher, dass Sie über eine Protokollsteuerdatei sowie über die Protokolldateien verfügen. Prüfen Sie außerdem, ob die Konfigurationsdateien von WebSphere MQ und des Warteschlangenmanagers konsistent sind, damit WebSphere MQ an den richtigen Stellen nach den wiederhergestellten Daten suchen kann.

Wenn die Daten ordnungsgemäß gesichert und wiederhergestellt wurden, wird der WS-Manager jetzt gestartet.

**Anmerkung:** Auch wenn die Warteschlangenmanagerdaten und Protokolldateien in verschiedenen Verzeichnissen gespeichert werden, sollten Sie die Verzeichnisse gleichzeitig sichern und wiederherstellen. Wenn die WS-Manager-Daten und -Protokolldateien unterschiedlich sind, befindet sich der WS-Manager nicht in einem gültigen Status und wird wahrscheinlich nicht gestartet. Wenn er gestartet wird, sind Ihre Daten wahrscheinlich beschädigt.

## WS-Manager-Konfiguration sichern

Wenn Sie die Konfiguration eines Warteschlangenmanagers sichern, kann dies hilfreich sein, um den Warteschlangenmanager anhand seiner Definitionen neu zu erstellen.

Gehen Sie wie folgt vor, um eine Sicherungskopie der Konfiguration eines Warteschlangenmanagers zu erstellen:

1. Vergewissern Sie sich, dass der WS-Manager aktiv ist.
2. a. Unter AIX, HP-UX, Linux, Solaris oder Windows: Führen Sie den Befehl "Dump MQ Configuration" (`dmpmqcfg`) mit der Standardformatierungsoption (`-f mqsc`) MQSC und allen Attributen (`-a`) aus. Verwenden Sie die Standardausgabeumleitung, um die Definitionen in einer Datei zu speichern. Beispiel:

```
dmpmqcfg -m MYQMGR -a > /mq/backups/MYQMGR.mqsc
```

## Warteschlangenmanagerkonfiguration wird zurückgespeichert

Führen Sie die folgenden Schritte aus, um eine Sicherung der Konfiguration eines Warteschlangenmanagers wiederherzustellen.

Gehen Sie wie folgt vor, um eine Sicherung der Konfiguration eines Warteschlangenmanagers wiederherzustellen:

1. Vergewissern Sie sich, dass der WS-Manager aktiv ist. Beachten Sie, dass der Warteschlangenmanager möglicherweise neu erstellt wurde, wenn die Beschädigung der Daten und Protokolle nicht auf anderem Wege behoben werden kann.
2. Führen Sie je nach Plattform einen der folgenden Befehle aus:

- a. Unter AIX, HP-UX, Linux, Solaris oder Windows: Führen Sie `runmqsc` für den Warteschlangenmanager aus und verwenden Sie die Standardeingabeumleitung, um die Definitionen aus einer Scriptdatei wiederherzustellen, die mit dem Befehl "Dump MQ Configuration" (`dmpmqcfg`) generiert wurde. Beispiel:

```
runmqsc MYQMGR < /mq/backups/MYQMGR.mqsc
```

### Zugehörige Verweise

[dmpmqcfg](#)

## Konfigurationsinformationen für IBM WebSphere MQ und Warteschlangenmanager ändern

---

Sie können das Verhalten von IBM WebSphere MQ oder einem einzelnen Warteschlangenmanager den Anforderungen Ihrer Installation entsprechend ändern.

Sie können die Konfigurationsdaten von IBM WebSphere MQ ändern, indem Sie die Werte bestimmter Konfigurationsattribute (oder -parameter) von IBM WebSphere MQ ändern.

Die Attributinformationen können Sie direkt in den Konfigurationsdateien von IBM WebSphere MQ ändern. Unter IBM WebSphere MQ für Windows und Linux (x86 -und x86-64 -Plattformen) können die IBM WebSphere MQ -Konfigurationsdateien mit IBM WebSphere MQ Explorer bearbeitet werden.

Unter Windows können die Konfigurationsinformationen, wie im Abschnitt [amqmdain](#) beschrieben, auch mit dem Befehl 'amqmdain' geändert werden.

Um weitere Informationen zur Konfiguration von IBM WebSphere MQ und Warteschlangenmanagern für Ihre Plattform zu erhalten, beachten Sie die folgenden Unterabschnitte:

### Zugehörige Konzepte

„Konfiguration“ auf Seite 5

Erstellen Sie einen oder mehrere Warteschlangenmanager auf einem oder mehreren Computern, und konfigurieren Sie sie auf Ihren Entwicklungs-, Test- und Produktionssystemen, um Nachrichten zu verarbeiten, die Ihre Geschäftsdaten enthalten.

### Zugehörige Tasks

[Planung](#)

[WebSphere MQ verwalten](#)

## Konfigurationsdaten auf UNIX, Linux, and Windows-Systemen ändern

Konfigurationsattribute werden auf Knoten- und Warteschlangenmanagerebene in Konfigurationsdateien gespeichert.

Auf Windows-, UNIX and Linux -Plattformen können Sie IBM WebSphere MQ -Konfigurationsattribute wie folgt ändern:

- Eine IBM WebSphere MQ -Konfigurationsdatei (**mqc.ini**), um Änderungen für IBM WebSphere MQ auf dem gesamten Knoten zu übernehmen. Pro Knoten gibt es eine Datei 'mqc.ini'.
- Eine Warteschlangenmanagerkonfigurationsdatei (**qm.ini**), um Änderungen für bestimmte Warteschlangenmanager zu bewirken. Für jeden Warteschlangenmanager auf dem Knoten steht eine Datei des Typs 'qm.ini' zur Verfügung.

Die Clientkonfigurationsoptionen werden separat in der Clientkonfigurationsdatei gespeichert.

Eine Konfigurationsdatei (oder **Zeilegruppen**-Datei) enthält eine oder mehrere Zeilegruppen. Hierbei handelt es sich um Gruppen von Zeilen in der Datei '.ini', die eine gemeinsame Funktion haben oder einen Systembereich definieren, beispielsweise Protokollfunktionen, Kanalfunktionen und installierbare Services.

Da die IBM WebSphere MQ-Konfigurationsdatei zur Lokalisierung der zu den Warteschlangenmanagern gehörigen Daten benötigt wird, würde das Fehlen bzw. eine falsche Einstellung dieser Datei dazu führen,

dass keine oder einige der MQSC-Befehle fehlschlagen. Auch die Verbindung zwischen Anwendungen und einem nicht in der IBM WebSphere MQ-Konfigurationsdatei definierten Warteschlangenmanager funktioniert nicht.

Alle Änderungen, die Sie an einer Konfigurationsdatei vornehmen, werden in der Regel erst beim nächsten Start des Warteschlangenmanagers wirksam.

Unter Windows und Linux (x86- und x86-64-Plattformen) können Sie Konfigurationsinformationen in IBM WebSphere MQ Explorer ändern.

Auf Windows-Systemen können Sie auch den Befehl `amqmdain` verwenden, um die Konfigurationsdateien zu bearbeiten.

Weitere Informationen zu den Konfigurationsoptionen unter Windows, UNIX and Linux finden Sie in den folgenden Abschnitten:

### **Zugehörige Konzepte**

[„Konfiguration“ auf Seite 5](#)

Erstellen Sie einen oder mehrere Warteschlangenmanager auf einem oder mehreren Computern, und konfigurieren Sie sie auf Ihren Entwicklungs-, Test- und Produktionssystemen, um Nachrichten zu verarbeiten, die Ihre Geschäftsdaten enthalten.

[„Konfigurationsinformationen für IBM WebSphere MQ und Warteschlangenmanager ändern“ auf Seite 454](#)

Sie können das Verhalten von IBM WebSphere MQ oder einem einzelnen Warteschlangenmanager den Anforderungen Ihrer Installation entsprechend ändern.

### **Zugehörige Tasks**

[Planung](#)

[WebSphere MQ verwalten](#)

### **Zugehörige Verweise**

[„Attribute zum Ändern von IBM WebSphere MQ -Konfigurationsinformationen“ auf Seite 460](#)

Ändern Sie unter IBM WebSphere MQ für Windows-Systeme und unter IBM WebSphere MQ für Linux (x86- und x86-64-Plattformen) die Konfigurationsdaten mithilfe von IBM WebSphere MQ Explorer. Auf anderen Systemen ändern Sie die Informationen, indem Sie die Konfigurationsdatei 'mqsc.ini' bearbeiten.

[„Konfigurationsdaten des Warteschlangenmanagers ändern“ auf Seite 467](#)

Über die hier beschriebenen Attribute wird die Konfiguration eines einzelnen Warteschlangenmanagers geändert. Sie überschreiben alle Einstellungen für WebSphere MQ.

## **Konfigurationsdateien bearbeiten**

Bearbeiten Sie die Konfigurationsdateien mit Hilfe von Befehlen oder einem Standardtexteditor.

Bevor Sie eine Konfigurationsdatei editieren, sichern Sie sie so, dass Sie eine Kopie haben, auf die Sie zurückgreifen können, wenn die Notwendigkeit besteht.

Sie können die Konfigurationsdateien entweder wie folgt bearbeiten:

- Automatisch unter Verwendung von Befehlen, die die Konfiguration von Warteschlangenmanagern auf dem Knoten ändern
- Manuell unter Verwendung eines Standardtexteditors

Sie können die Standardwerte in den WebSphere MQ -Konfigurationsdateien nach der Installation bearbeiten.

Wenn Sie einen falschen Wert in einem Konfigurationsdateiattribut festlegen, wird der Wert ignoriert und eine Bedienernachricht ausgegeben, um das Problem anzugeben. (Der Effekt ist der gleiche wie das Fehlen des Attributs vollständig.)

Wenn Sie einen neuen WS-Manager erstellen, gilt Folgendes:

- Sichern Sie die Konfigurationsdatei WebSphere MQ .
- Sichern Sie die neue WS-Manager-Konfigurationsdatei.



Kommentare können in Konfigurationsdateien aufgenommen werden, indem ein ";" oder ein "#"-Zeichen vor dem Kommentartext hinzugefügt wird. Wenn Sie ein ";" oder ein "#"-Zeichen verwenden möchten, ohne dass es einen Kommentar darstellt, können Sie dem Zeichen ein "\"-Zeichen als Präfix voranstellen und es wird als Teil der Konfigurationsdaten verwendet.

### **Wann müssen Sie eine Konfigurationsdatei bearbeiten?**

Bearbeiten Sie eine Konfigurationsdatei, um die Sicherung wiederherzustellen, einen Warteschlangenmanager zu verschieben, den Standardwarteschlangenmanager zu ändern oder den IBM Support zu unterstützen.

Möglicherweise müssen Sie eine Konfigurationsdatei bearbeiten, wenn Sie zum Beispiel:

- Sie verlieren eine Konfigurationsdatei. (Wiederherstellung von der Sicherung, wenn Sie können.)
- Sie müssen einen oder mehrere WS-Manager in ein neues Verzeichnis verschieben.
- Sie müssen den Standardwarteschlangenmanager ändern. Dies kann passieren, wenn Sie versehentlich den vorhandenen WS-Manager löschen.
- Ihr IBM Support Center empfiehlt Ihnen, dies zu tun.

### **Konfigurationsdateiprioritäten**

Der Wert eines Attributs wird an mehreren Stellen definiert. Attribute, die in Befehlen festgelegt werden, haben Vorrang vor Attributen in Konfigurationsdateien.

Die Attributwerte einer Konfigurationsdatei werden entsprechend den folgenden Prioritäten festgelegt:

- Die in der Befehlszeile eingegebenen Parameter haben Vorrang vor den Werten, die in den Konfigurationsdateien definiert sind.
- Werte, die in den qm.ini-Dateien definiert sind, haben Vorrang vor den Werten, die in der Datei mqs.ini definiert sind.

### **Die IBM WebSphere MQ -Konfigurationsdatei mqs.ini**

Die IBM WebSphere MQ-Konfigurationsdatei "mqs.ini" enthält Informationen, die für alle Warteschlangenmanager auf dem Knoten relevant sind. Sie wird automatisch während der Installation erstellt.

Bei IBM WebSphere MQ for UNIX and Linux -Produkten lauten das Datenverzeichnis und das Protokollverzeichnis immer `/var/mqm` bzw. `/var/mqm/log`.

Auf Windows-Systemen werden der Speicherort des Datenverzeichnisses `mqs.ini` und der Speicherort des Protokollverzeichnisses in der Registry gespeichert, da sie wechselnde Speicherorte haben können.

Außerdem befinden sich auf Windows -Systemen die Informationen zur Installationskonfiguration (enthalten in `mqinst.ini` auf IBM WebSphere MQ for UNIX and Linux -Systemen) in der Registry, da unter Windows keine Datei `mqinst.ini` vorhanden ist.

Die Datei `mqs.ini` für Windows -Systeme wird durch den im Schlüssel `HKLM\SOFTWARE\IBM\WepSphere MQ` angegebenen `WorkPath` angegeben. Sie enthält:

- Die Namen der WS-Manager
- Der Name des Standard-WS-Managers.
- Die Position der Dateien, die jedem von ihnen zugeordnet sind.

Die bereitgestellte Zeilengruppe `LogDefaults` für eine neue Installation von IBM WebSphere MQ enthält keine expliziten Werte für die Attribute. Das Fehlen eines Attributs bedeutet, dass der Standardwert für diesen Wert bei der Erstellung eines neuen Warteschlangenmanagers verwendet wird. Die Standardwerte werden für die Zeilengruppe `LogDefaults` in [Abbildung 71 auf Seite 457](#) angezeigt. Der Wert `null` für das Attribut `LogBufferPages` bedeutet 512.

Wenn Sie einen anderen Wert als den Standardwert benötigen, müssen Sie diesen Wert explizit in der Zeilengruppe `LogDefaults` angeben.



```

#*****#
#* Module Name: mqs.ini **#
#* Type : WebSphere MQ Machine-wide Configuration File **#
#* Function : Define WebSphere MQ resources for an entire machine **#
#*****#
#* Notes : **#
#* 1) This is the installation time default configuration **#
#* **#
#*****#
AllQueueManagers:
#*****#
#* The path to the qmgrs directory, below which queue manager data **#
#* is stored **#
#*****#
DefaultPrefix=/var/mqm

LogDefaults:
 LogPrimaryFiles=3
 LogSecondaryFiles=2
 LogFilePages=4096
 LogType=CIRCULAR
 LogBufferPages=0
 LogDefaultPath=/var/mqm/log

QueueManager:
 Name=saturn.queue.manager
 Prefix=/var/mqm
 Directory=saturn!queue!manager
 InstallationName=Installation1

QueueManager:
 Name=pluto.queue.manager
 Prefix=/var/mqm
 Directory=pluto!queue!manager
 InstallationName=Installation2

DefaultQueueManager:
 Name=saturn.queue.manager

ApiExitTemplate:
 Name=OurPayrollQueueAuditor
 Sequence=2
 Function=EntryPoint
 Module=/usr/ABC/auditor
 Data=123

ApiExitCommon:
 Name=MQPoliceman
 Sequence=1
 Function=EntryPoint
 Module=/usr/MQPolice/tmqp
 Data=CheckEverything

```

Abbildung 71. Beispiel einer IBM WebSphere MQ -Konfigurationsdatei für UNIX -Systeme

## Warteschlangenmanagerkonfigurationsdateien, qm.ini

Die Konfigurationsdatei qm.inieines Warteschlangenmanagers enthält relevante Informationen für einen bestimmten Warteschlangenmanager.

Für jeden Warteschlangenmanager gibt es eine WS-Manager-Konfigurationsdatei. Die Datei 'qm.ini' wird automatisch bei der Erstellung des Warteschlangenmanagers, dem sie zugeordnet ist, ebenfalls erstellt.

**V 7.5.0.9** Ab IBM WebSphere MQ Version 7.5.0Fixpack 9 prüft der Befehl **strmqm** die Syntax der Zeilengruppen CHANNELS und SSL in der Datei qm. ini, bevor er den Warteschlangenmanager vollständig startet. Dies macht es einfacher, Fehler zu erkennen und schnell zu korrigieren, falls **strmqm** feststellt, dass die Datei qm. ini Fehler enthält. Weitere Informationen finden Sie im Abschnitt [strmqm](#).

## Position der qm.ini -Dateien



Auf UNIX and Linux -Systemen befindet sich die Datei qm.ini im Stammverzeichnis der Verzeichnisbaumstruktur, die vom Warteschlangenmanager belegt wird. Der Pfad und der Name einer Konfigurationsdatei für einen WS-Manager mit dem Namen QMNAME sind beispielsweise:

```
/var/mqm/qmgrs/QMNAME/qm.ini
```

Auf Windows -Systemen wird die Position der Datei qm.ini durch den WorkPath angegeben, der im Schlüssel HKLM\SOFTWARE\IBM\WebSphere MQ angegeben ist. Der Pfad und der Name einer Konfigurationsdatei für einen WS-Manager mit dem Namen QMNAME sind beispielsweise:

```
C:\Program Files\IBM\WebSphere MQ\qmgrs\QMNAME\qm.ini
```

Der Name des WS-Managers kann bis zu 48 Zeichen lang sein. Dies garantiert jedoch nicht, dass der Name gültig oder eindeutig ist. Daher wird basierend auf dem Namen des Warteschlangenmanagers ein Verzeichnisname generiert. Dieser Prozess wird als *Namensumsetzung* bezeichnet. Eine Beschreibung finden Sie unter [Understanding WebSphere MQ file names](#).

## Beispiel für eine qm.ini -Datei



Das folgende Beispiel zeigt, wie Gruppen von Attributen in einer Konfigurationsdatei des Warteschlangenmanagers in IBM WebSphere MQ für UNIX and Linux -Systeme angeordnet werden können.

```
##* Module Name: qm.ini ##*
##* Type : WebSphere MQ queue manager configuration file ##*
##* Function : Define the configuration of a single queue manager ##*
##* ##*
##* Notes : ##*
##* 1) This file defines the configuration of the queue manager ##*
##* ##*
##* ##*
ExitPath:
 ExitsDefaultPath=/var/mqm/exits
 ExitsDefaultPath64=/var/mqm/exits64

Service:
 Name=AuthorizationService
 EntryPoints=13

ServiceComponent:
 Service=AuthorizationService
 Name=MQSeries.UNIX.auth.service
 Module=opt/mqm/bin/amqzfu
 ComponentDataSize=0

Log:
 LogPrimaryFiles=3
 LogSecondaryFiles=2
 LogFilePages=4096
 LogType=CIRCULAR
 LogBufferPages=01
 LogPath=/var/mqm/log/saturn!queue!manager/

AccessMode:
 SecurityGroup=wmq\wmq

XAResourceManager:
 Name=DB2 Resource Manager Bank
 SwitchFile=/usr/bin/db2swit
 XAOpenString=MQBankDB
 XACloseString=
 ThreadOfControl=THREAD
```

```

Channels: 2
 MaxChannels=200
 MaxActiveChannels=100
 MQIBindType=STANDARD

AccessMode:
 SecurityGroup=wmq\wmq
TCP:
 KeepAlive = Yes
 SvrSndBuffSize=32768
 SvrRcvBuffSize=32768
 Connect_Timeout=0

QMErrorLog:
 ErrorLogSize=262144
 ExcludeMessage=7234
 SuppressMessage=9001,9002,9202
 SuppressInterval=30

ApiExitLocal:
 Name=ClientApplicationAPIChecker
 Sequence=3
 Function=EntryPoint
 Module=/usr/Dev/ClientAppChecker
 Data=9.20.176.20

```

### Anmerkungen:

1. Der Wert 0 für `LogBufferPages` ergibt einen Wert von 512.
2. Weitere Informationen zur Zeilengruppe 'Channel' finden Sie in [„Initialisierungs- und Konfigurationsdateien“](#) auf Seite 76.
3. Die maximale Anzahl an XAResourceManager-Zeilengruppen ist auf 255 begrenzt. Sie sollten jedoch nur eine kleine Anzahl von Zeilengruppen verwenden, um eine Verschlechterung der Transaktionsleistung zu vermeiden.

WebSphere MQ unter Unix verwendet Konfigurationsdateien mit der Dateierweiterung `.ini`, z. B. `qm.ini`. Es gibt einige Dienstprogramme in WebSphereMQ, z. B. **setmqm**, die eine temporäre Sicherungskopie der Dateien erstellen. Beispiel: Die Datei `qm.ini` erstellt eine Sicherungskopie mit dem Namen `qm.ini.bak`. Ein Dienstprogramm ändert die Datei `qm.ini`, speichert die aktualisierte Datei und löscht dann die Datei `qm.ini.bak`. Wenn das Dienstprogramm die Datei `qm.ini` nicht speichern kann, stellt es den Inhalt der Datei `qm.ini` aus der Sicherungsdatei `qm.ini.bak` wieder her und löscht dann die Datei `qm.ini.bak`.

Wenn eine Datei `qm.ini.bak` vorhanden ist, setzt das Dienstprogramm die Datei `qm.ini` mit dem Inhalt der Datei `qm.ini.bak` zurück und löscht die Datei `qm.ini.bak`. Aus diesem Grund sollten Sie keine Sicherungskopien der `*.ini`-Dateien mit der Dateierweiterung `.bak` erstellen, da diese Sicherungsdateien möglicherweise von WebSphere MQ -Dienstprogrammen gelöscht werden.

Informationen darüber, wann Ihre Änderungen wirksam werden, erhalten Sie im Abschnitt [„Konfigurationsdaten auf UNIX, Linux, and Windows-Systemen ändern“](#) auf Seite 454.

## Installationskonfigurationsdatei, mqinst.ini

### UNIX and Linux-Systeme

Die Installationskonfigurationsdatei `mqinst.ini` enthält Informationen zu allen IBM WebSphere MQ-Installationen auf einem UNIX- oder Linux-System.

Die Datei `mqinst.ini` befindet sich im `/etc/opt/mqm`-Verzeichnis auf UNIX and Linux -Systemen. Sie enthält Informationen darüber, welche Installation (falls vorhanden) die primäre Installation sowie die folgenden Informationen für jede Installation enthält:

- Der Installationsname
- Die Installationsbeschreibung
- Die Installations-ID
- Der Installationspfad

Diese Datei darf nicht direkt bearbeitet oder referenziert werden, da ihr Format nicht festgelegt ist und sich ändern kann. Stattdessen sollten Sie zum Erstellen, Löschen, Abfragen und Ändern der Werte in der Datei 'mqinst.ini' folgende Befehle verwenden:

- [crtmqinst](#) zum Erstellen von Einträgen.
- [dlmqinst](#) zum Löschen von Einträgen.
- [dspmqinst](#) , um Einträge anzuzeigen.
- [setmqinst](#) zum Festlegen von Einträgen.

Die Installationskennung (nur für die interne Verwendung) wird automatisch festgelegt und darf nicht geändert werden.

## Windows-Systeme

Die Konfigurationsdaten der Installation sind auf Windows-Systemen in folgendem Schlüssel gespeichert:

```
HKLM\SOFTWARE\IBM\WebSphere MQ\Installation\
```

Dieser Schlüssel darf nicht direkt bearbeitet oder referenziert werden, da sein Format nicht fixiert ist und sich ändern kann. Stattdessen sollten Sie zum Abfragen und Ändern der Werte in der Registry folgende Befehle verwenden:

- [dspmqinst](#) , um Einträge anzuzeigen.
- [setmqinst](#) zum Festlegen von Einträgen.

Unter Windows stehen die Befehle **crtmqinst** und **dlmqinst** nicht zur Verfügung. Die Installations- und Deinstallationsprozesse bearbeiten die Erstellung und das Löschen der erforderlichen Registry-Einträge.

## Attribute zum Ändern von IBM WebSphere MQ -Konfigurationsinformationen

Ändern Sie unter IBM WebSphere MQ für Windows-Systeme und unter IBM WebSphere MQ für Linux (x86 -und x86-64 -Plattformen) die Konfigurationsdaten mithilfe von IBM WebSphere MQ Explorer. Auf anderen Systemen ändern Sie die Informationen, indem Sie die Konfigurationsdatei 'mqs.ini' bearbeiten.

Attribute für bestimmte Komponenten finden Sie in den folgenden Unterabschnitten:

### Zugehörige Konzepte

[„Konfiguration“ auf Seite 5](#)

Erstellen Sie einen oder mehrere Warteschlangenmanager auf einem oder mehreren Computern, und konfigurieren Sie sie auf Ihren Entwicklungs-, Test- und Produktionssystemen, um Nachrichten zu verarbeiten, die Ihre Geschäftsdaten enthalten.

[„Konfigurationsinformationen für IBM WebSphere MQ und Warteschlangenmanager ändern“ auf Seite 454](#)

Sie können das Verhalten von IBM WebSphere MQ oder einem einzelnen Warteschlangenmanager den Anforderungen Ihrer Installation entsprechend ändern.

### Zugehörige Tasks

[Planung](#)

[WebSphere MQ verwalten](#)

### Zugehörige Verweise

[„Konfigurationsdaten des Warteschlangenmanagers ändern“ auf Seite 467](#)

Über die hier beschriebenen Attribute wird die Konfiguration eines einzelnen Warteschlangenmanagers geändert. Sie überschreiben alle Einstellungen für WebSphere MQ.

## Alle WS-Manager

Verwenden Sie die Eigenschaftenseite General und Extended WebSphere MQ im IBM WebSphere MQ Explorer oder die Zeilengruppe AllQueueManagers in der Datei mqs.ini , um die folgenden Informationen zu allen Warteschlangenmanagern anzugeben.

### DefaultPrefix=Verzeichnisname

Dieses Attribut gibt den Pfad zum Verzeichnis 'qmgrs' an, in dem die WS-Manager-Daten gespeichert werden.

Wenn Sie das Standardpräfix für den Warteschlangenmanager ändern, replizieren Sie die Verzeichnisstruktur, die zur Installationszeit erstellt wurde.

Insbesondere müssen Sie die qmgrs-Struktur erstellen. Stoppen Sie WebSphere MQ, bevor Sie das Standardpräfix ändern, und starten Sie WebSphere MQ erst erneut, nachdem Sie die Strukturen an die neue Position verschoben und das Standardpräfix geändert haben.

**Anmerkung:** Löschen Sie auf UNIX and Linux -Systemen nicht das Verzeichnis `/var/mqm/errors` und auf Windows -Systemen nicht das Verzeichnis `\errors`.

Als Alternative zum Ändern des Standardpräfix können Sie die Umgebungsvariable MQSPREFIX verwenden, um DefaultPrefix für den `crtmqm` -Befehl zu überschreiben.

Aufgrund der Einschränkungen des Betriebssystems müssen Sie den angegebenen Pfad so kurz halten, dass die Summe der Pfadlänge und aller WS-Manager-Namen maximal 70 Zeichen lang ist.

### ConvEBCDICNewline = NL\_TO\_LF | TABLE | ISO

EBCDIC-Codepages enthalten ein Zeilenvorschubzeichen (NL), das von ASCII-Codepages nicht unterstützt wird (obwohl einige ISO-Varianten von ASCII ein Äquivalent enthalten).

Verwenden Sie das Attribut ConvEBCDICNewline, um anzugeben, wie WebSphere MQ das EBCDIC-NL-Zeichen in das ASCII-Format konvertiert.

#### NL\_TO\_LF

Konvertieren Sie das EBCDIC-NL-Zeichen (X'15 ') in das ASCII-Zeilenvorschubzeichen, LF (X'0A'), für alle EBCDIC-zu-ASCII-Konvertierungen.

NL\_TO\_LF ist der Standardwert.

#### TABELLE

Konvertieren Sie das EBCDIC-NL-Zeichen entsprechend den Konvertierungstabellen, die auf Ihrer Plattform für alle EBCDIC-zu-ASCII-Konvertierungen verwendet werden.

Der Effekt dieses Konvertierungstyps kann von Plattform zu Plattform und von Sprache zu Sprache variieren. Selbst auf der gleichen Plattform kann das Verhalten variieren, wenn Sie andere CCSIDs verwenden.

#### ISO

Konvertieren:

- ISO-CCSIDs mit der Methode TABLE
- Alle anderen CCSIDs mit der Methode NL\_TO\_CF

Mögliche ISO-CCSIDs werden in [Tabelle 32](#) auf Seite 461 angezeigt.

| CCSID | Codetruppe |
|-------|------------|
| 819   | ISO8859-1  |
| 912   | ISO8859-2  |
| 915   | ISO8859-5  |
| 1089  | ISO8859-6  |
| 813   | ISO8859-7  |
| 916   | ISO8859-8  |
| 920   | ISO8859-9  |
| 1051  | roman8     |

Wenn die ASCII-CCSID kein ISO-Subset ist, nimmt ConvEBCDICNewline standardmäßig NL\_TO\_LF an.

## Standard-WS-Manager

Verwenden Sie die Eigenschaftenseite General WebSphere MQ im IBM WebSphere MQ Explorer oder die Zeilengruppe DefaultQueueManager in der mqs.ini, um den Standardwarteschlangenmanager anzugeben.

### Name =standardwarteschlangenmanager

Der Standardwarteschlangenmanager verarbeitet alle Befehle, für die kein Warteschlangenmanagername explizit angegeben wurde. Das Attribut DefaultQueueManager wird automatisch aktualisiert, wenn Sie einen neuen Standardwarteschlangenmanager erstellen. Wenn Sie versehentlich einen neuen Standardwarteschlangenmanager erstellen und dann zum ursprünglichen Warteschlangenmanager zurückkehren möchten, ändern Sie das Attribut DefaultQueueManager manuell.

## Exiteigenschaften

Verwenden Sie die Eigenschaftenseite Extended IBM WebSphere MQ in IBM WebSphere MQ Explorer oder die Zeilengruppe ExitProperties in der Datei mqs.ini, um Konfigurationsoptionen anzugeben, die von Exitprogrammen des Warteschlangenmanagers verwendet werden.

### CLWLMode=SAFE|FAST

Mit dem Exit für Clusterauslastung (CLWL) können Sie angeben, welche Clusterwarteschlange im Cluster als Antwort auf einen MQI-Aufruf geöffnet werden soll (z. B. MQOPEN, MQPUT). Der CLWL-Exit wird abhängig von dem Wert, den Sie im Attribut CLWLMode angeben, entweder im Modus FAST oder im Modus SAFE ausgeführt. Wenn Sie das Attribut CLWLMode nicht angeben, wird der Exit für Clusterauslastung im SAFE-Modus ausgeführt.

#### SAFE

Führen Sie den CLWL-Exit in einem separaten Prozess aus dem Warteschlangenmanager aus. Dies ist die Standardeinstellung.

Tritt bei der Ausführung im SAFE-Modus ein Problem mit dem vom Benutzer geschriebenen CLWL-Exit auf, geschieht Folgendes:

- Der CLWL-Serverprozess (amqzlw0) schlägt fehl.
- Der WS-Manager startet den CLWL-Serverprozess erneut.
- Der Fehler wird Ihnen im Fehlerprotokoll gemeldet. Wenn ein MQI-Aufruf in Bearbeitung ist, erhalten Sie eine Benachrichtigung in Form eines Rückkehrcodes.

Die Integrität des Warteschlangenmanagers bleibt erhalten.

**Anmerkung:** Die Ausführung des CLWL-Exits in einem separaten Prozess kann sich auf die Leistung auswirken.

#### FAST

Führen Sie den Cluster-Exit inline im WS-Manager-Prozess aus.

Wenn Sie diese Option angeben, wird die Leistung verbessert, da die Prozesse, die mit der Ausführung im SAFE-Modus verbunden sind, vermieden werden, dies jedoch zu Lasten der Integrität des Warteschlangenmanagers geht. Sie sollten den CLWL-Exit nur im FAST-Modus ausführen, wenn Sie davon überzeugt sind, dass es **keine** Probleme mit dem CLWL-Exit gibt, und Sie sind besonders besorgt über die Leistung.

Tritt ein Problem auf, wenn der CLWL-Exit im FAST-Modus ausgeführt wird, schlägt der Warteschlangenmanager fehl, und Sie laufen Gefahr, dass die Integrität des Warteschlangenmanagers beeinträchtigt wird, der beeinträchtigt wird.

## Protokollstandardwerte für IBM WebSphere MQ

Verwenden Sie die Eigenschaftenseite `Default log settings` (Standardprotokolleinstellungen) von IBM WebSphere MQ im IBM WebSphere MQ Explorer oder die Zeilengruppe `LogDefaults` in der Datei `mqs.ini`, um Informationen zu Protokollstandardwerten für alle Warteschlangenmanager anzugeben.

Wenn die Zeilengruppe nicht vorhanden ist, werden die MQ-Standardwerte verwendet. Die Protokollattribute werden beim Erstellen eines Warteschlangenmanagers als Standardwerte verwendet, können aber überschrieben werden, wenn Sie die Protokollattribute im Befehl `crtmqm` angeben. Ausführliche Informationen zu diesem Befehl finden Sie in [crtmqm](#).

Sobald ein WS-Manager erstellt wurde, werden die Protokollattribute für diesen Warteschlangenmanager aus den in „Warteschlangenmanagerprotokolle“ auf Seite 471 beschriebenen Einstellungen übernommen.

Das Standardpräfix (angegeben in „Alle WS-Manager“ auf Seite 460) und der Protokollpfad, die für den jeweiligen Warteschlangenmanager angegeben sind (in „Warteschlangenmanagerprotokolle“ auf Seite 471 angegeben), ermöglichen dem Warteschlangenmanager und seinem Protokoll, sich auf verschiedenen physischen Laufwerken zu befinden. Dies ist die empfohlene Methode, obwohl sie sich standardmäßig auf demselben Laufwerk befinden.

Informationen zum Berechnen der Protokollgrößen finden Sie in „Berechnen der Größe des Protokolls“ auf Seite 436.

**Anmerkung:** Die in der folgenden Parameterliste angegebenen Grenzwerte sind Grenzwerte, die von WebSphere MQ festgelegt werden. Durch die Begrenzung des Betriebssystems kann die maximal mögliche Protokollgröße reduziert werden.

### **LogPrimaryFiles=3|2-254 ( Windows) |2-510 (UNIX and Linux -Systeme)**

Die Protokolldateien, die beim Erstellen des Warteschlangenmanagers zugeordnet werden.

Die Mindestanzahl primärer Protokolldateien beträgt 2 und die maximale Anzahl 254 unter Windows bzw. 510 auf UNIX and Linux -Systemen. Der Standardwert ist 3.

Die Gesamtzahl der primären und sekundären Protokolldateien darf unter Windows nicht größer als 255 und auf UNIX and Linux -Systemen nicht kleiner als 511 sein.

Der Wert wird geprüft, wenn der WS-Manager erstellt oder gestartet wird. Sie können sie ändern, nachdem der WS-Manager erstellt wurde. Eine Änderung des Werts ist jedoch erst wirksam, wenn der Warteschlangenmanager erneut gestartet wird und der Effekt möglicherweise nicht sofort ausgeführt wird.

### **LogSecondaryFiles=2|1-253 ( Windows) |1-509 (UNIX and Linux -Systeme)**

Die Protokolldateien, die zugeordnet werden, wenn die Primärdateien erschöpft sind.

Die minimale Anzahl sekundärer Protokolldateien ist 1 und die maximale Anzahl 253 unter Windows bzw. 509 auf Systemen mit UNIX and Linux . Die Standardanzahl ist 2.

Die Gesamtzahl der primären und sekundären Protokolldateien darf unter Windows nicht größer als 255 und auf UNIX and Linux -Systemen nicht kleiner als 511 sein.

Der Wert wird geprüft, wenn der Warteschlangenmanager gestartet wird. Sie können diesen Wert ändern, aber Änderungen werden erst wirksam, wenn der Warteschlangenmanager erneut gestartet wird, und selbst dann wird der Effekt möglicherweise nicht sofort wirksam.

### **LogFilePages =Anzahl**

Die Protokolldateien werden in einer Reihe von Dateien mit dem Namen "Protokolldateien" festgehalten. Die Protokolldateigröße wird in Einheiten von 4-KB-Seiten angegeben.

Die Standardanzahl der Protokolldateiseiten beträgt 4096, wobei eine Protokolldateigröße von 16 MB angegeben wird.

Auf UNIX and Linux -Systemen beträgt die Mindestanzahl der Protokolldateiseiten 64 und unter Windows beträgt die Mindestanzahl der Protokolldateiseiten 32. In beiden Fällen beträgt die maximale Anzahl 65.535.

**Anmerkung:** Die Größe der Protokolldateien, die bei der Erstellung des Warteschlangenmanagers angegeben wurden, kann für einen Warteschlangenmanager nicht geändert werden.

### **LogType=CIRCULAR| LINEAR**

Der Typ des zu verwendenden Protokolls. Der Standardwert ist CIRCULAR.

#### **CIRCULAR**

Starten Sie die Wiederherstellung nach einem Neustart mit Hilfe des Protokolls, um Transaktionen rückgängig zu machen, die sich in Bearbeitung befanden, als das System gestoppt wurde

Eine genauere Beschreibung des Protokolltyps CIRCULAR (Umlaufprotokollierung) finden Sie im Abschnitt [„Typen der Protokollierung“](#) auf Seite 433.

#### **LINEAR**

Sowohl für die Wiederherstellung nach einem Neustart als auch für die Datenträger-oder Vorwärtswiederherstellung (durch das Erstellen verlorener oder beschädigter Daten durch Wiedergabe des Inhalts des Protokolls).

Eine ausführlichere Erläuterung der linearen Protokollierung finden Sie in [„Typen der Protokollierung“](#) auf Seite 433 .

Wenn Sie den Standardwert ändern möchten, können Sie entweder das Attribut LogType bearbeiten oder die lineare Protokollierung mit dem Befehl `crtmqm` angeben. Nach der Erstellung eines Warteschlangenmanagers kann die Protokollierungsmethode nicht mehr geändert werden.

### **LogBufferPages=0|0-4096**

Die Größe des Speichers, der den Pufferdatensätzen für das Schreiben zugeordnet ist, wobei die Größe der Puffer in Einheiten von 4-KB-Seiten angegeben wird.

Die Mindestanzahl der Pufferseiten beträgt 18 und der Maximalwert 4096. Größere Puffer führen zu einem höheren Durchsatz, insbesondere bei größeren Nachrichten.

Wenn Sie 0 (die Standardeinstellung) angeben, wählt der Warteschlangenmanager die Größe aus. In WebSphere MQ Version 7.0 ist dies 512 (2048 KB).

Wenn Sie eine Zahl im Bereich von 1 bis 17 angeben, nimmt der WS-Manager standardmäßig den Wert 18 (72 KB) an. Wenn Sie eine Zahl im Bereich von 18 und 4096 angeben, verwendet der Warteschlangenmanager die angegebene Zahl, um den zugeordneten Speicher festzulegen.

### **LogDefaultPath =Verzeichnisname**

Das Verzeichnis, in dem sich die Protokolldateien für einen WS-Manager befinden. Das Verzeichnis befindet sich auf einer lokalen Einheit, in die der Warteschlangenmanager schreiben kann, und zwar vorzugsweise auf einem anderen Laufwerk aus den Nachrichtenwarteschlangen. Die Angabe eines anderen Laufwerks bietet einen zusätzlichen Schutz im Falle eines Systemausfalls.

Der Standardwert lautet:

- `<DefaultPrefix>\log` für WebSphere MQ für Windows , wobei `<DefaultPrefix>` der Wert ist, der im Attribut `DefaultPrefix` auf der Eigenschaftenseite von `All Queue Managers WebSphere MQ` angegeben ist. Dieser Wert wird bei der Installation festgelegt.
- `/var/mqm/log` für WebSphere MQ für UNIX and Linux -Systeme

Alternativ können Sie den Namen eines Verzeichnisses mit dem Befehl `crtmqm` mit dem Flag `-ld` angeben. Wenn ein Warteschlangenmanager erstellt wird, wird auch ein Verzeichnis unter dem WS-Manager-Verzeichnis erstellt, und dieses Verzeichnis wird verwendet, um die Protokolldateien zu speichern. Der Name dieses Verzeichnisses basiert auf dem Namen des Warteschlangenmanagers. Dadurch wird sichergestellt, dass der Protokolldateipfad eindeutig ist und dass er auch Einschränkungen bezüglich der Länge von Verzeichnisnamen entspricht.

Wenn Sie `-ld` im Befehl `crtmqm` nicht angeben, wird der Wert des Attributs `LogDefaultPath` in der Datei `mq5.ini` verwendet.

Der Name des Warteschlangenmanagers wird an den Verzeichnisnamen angehängt, um sicherzustellen, dass mehrere WS-Manager unterschiedliche Protokollverzeichnisse verwenden.



Wenn der Warteschlangenmanager erstellt wird, wird in den Protokollattributen in den Konfigurationsinformationen ein LogPath -Wert erstellt, der den vollständigen Verzeichnisnamen für das Protokoll des Warteschlangenmanagers enthält. Dieser Wert wird verwendet, um das Protokoll zu lokalisieren, wenn der Warteschlangenmanager gestartet oder gelöscht wird.

**LogWriteIntegrity =SingleWrite|DoubleWrite| TripleWrite**

Die Methode, die die Protokollfunktion verwendet, um Protokollsätze zuverlässig zu schreiben.

**TripleWrite**

Dies ist die Standardmethode.

Beachten Sie, dass Sie **DoubleWrite** auswählen können, aber wenn Sie dies tun, interpretiert das System diese Option als **TripleWrite** .

**SingleWrite**

**Einfach schreiben** sollte nur verwendet werden, wenn das Dateisystem bzw. die Einheit, auf der sich das WebShere MQ-Wiederherstellungsprotokoll befindet, die Atomizität von 4-KB-Schreibvorgängen explizit garantiert.

Das heißt, wenn ein Schreiben einer 4-KB-Seite aus irgendeinem Grund fehlschlägt, sind die einzigen beiden möglichen Status entweder das Vorimage oder das Nachimage. Ein Zwischenzustand sollte nicht möglich sein.

**Advanced Configuration and Power Interface (ACPI)**

Auf der Eigenschaftenseite ACPI WebSphere MQ im IBM WebSphere MQ Explorer können Sie angeben, wie sich WebSphere MQ verhalten soll, wenn das System eine Aussetzanforderung empfängt.

Windows unterstützt den ACPI-Standard (Advanced Configuration and Power Interface). Dadurch können Windows -Benutzer mit ACPI-aktivierter Hardware Kanäle stoppen und erneut starten, wenn das System in den Aussetzmodus wechselt und den Modus wieder aufnimmt.

Beachten Sie, dass die auf der Eigenschaftenseite ACPI WebSphere MQ angegebenen Einstellungen nur angewendet werden, wenn der Alertmonitor aktiv ist. Wenn der Alertmonitor ausgeführt wird, ist das Symbol Alertmonitor in der Taskleiste vorhanden.

**DoDialog=J | N**

Zeigt den Dialog zum Zeitpunkt der Aussetzungsanforderung an.

**DenySuspend=Y | N**

Verweigert die Aussetzungsanforderung. Dies wird verwendet, wenn DoDialog = N ist, oder wenn DoDialog=Y und ein Dialog nicht angezeigt werden kann, z. B., weil der Notizbuchdeckel geschlossen ist.

**CheckChannelsRunning=Y | N**

Prüft, ob Kanäle aktiv sind. Das Ergebnis kann das Ergebnis der anderen Einstellungen bestimmen.

In der folgenden Tabelle sind die Auswirkungen der einzelnen Kombinationen dieser Parameter aufgeführt:

| DoDialog | DenySuspend | Aktive CheckChannels | Action                                                                                                          |
|----------|-------------|----------------------|-----------------------------------------------------------------------------------------------------------------|
| N        | N           | N                    | Akzeptieren Sie die Aussetzungsanforderung.                                                                     |
| N        | N           | Y                    | Akzeptieren Sie die Aussetzungsanforderung.                                                                     |
| N        | Y           | N                    | Die Aussetzanforderung zurückweisen.                                                                            |
| N        | Y           | Y                    | Wenn Kanäle aktiv sind, wird die Aussetzungsanforderung verweigert. Wenn die Anforderung nicht akzeptiert wird. |

|   |   |   |                                                                                                                                                                           |
|---|---|---|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Y | N | N | Zeigen Sie den Dialog an (siehe <a href="#">Hinweis</a> ; akzeptieren Sie die Aussetzungsanforderung). Dies ist die Standardeinstellung.                                  |
| Y | N | Y | Wenn keine Kanäle aktiv sind, akzeptieren Sie die Aussetzungsanforderung. Wenn sie den Dialog anzeigen (siehe <a href="#">Hinweis</a> ), akzeptieren Sie die Anforderung. |
| Y | Y | N | Zeigt den Dialog an ( <a href="#">Hinweis</a> ; verweigert die Aussetzungsanforderung).                                                                                   |
| Y | Y | Y | Wenn keine Kanäle aktiv sind, akzeptieren Sie die Aussetzungsanforderung. Wenn sie den Dialog anzeigen ( <a href="#">Hinweis</a> ; verweigern Sie die Anforderung).       |

**Anmerkung:** Wenn in den Fällen, in denen der Dialog angezeigt werden soll, der Dialog nicht angezeigt werden kann (z. B. weil die Notizbuch-ID geschlossen ist), wird die Option 'DenySuspend' verwendet, um zu ermitteln, ob die Aussetzungsanforderung akzeptiert oder verweigert wird.

## API-Exits

Verwenden Sie den IBM WebSphere MQ Explorer oder den Befehl `amqmdain`, um die Einträge für API-Exits zu ändern.

Verwenden Sie die Eigenschaftenseite von Exits IBM WebSphere MQ in der Zeilengruppe IBM WebSphere MQ Explorer oder in der Zeilengruppe `ApiExitTemplate` und `ApiExitCommon` in der Datei `mqs.ini`, um API-Exitroutinen für alle Warteschlangenmanager zu identifizieren. Auf Windows-Systemen können Sie auch den Befehl `amqmdain` verwenden, um die Einträge für API-Exits zu ändern. (Zur Angabe von API-Exit-Routinen für einzelne Warteschlangenmanager wird die Zeilengruppe `ApiExitLocal` wie in „API-Exits“ auf Seite 481 beschrieben verwendet.)

Eine vollständige Beschreibung der Attribute für diese Zeilengruppen finden Sie im Abschnitt [API-Exits konfigurieren](#).

## Warteschlangenmanager

Für jeden WS-Manager gibt es eine Zeilengruppe `QueueManager`. Verwenden Sie die Zeilengruppe, um die Position des Warteschlangenmanagerverzeichnisses anzugeben.

Auf Windows-, UNIX and Linux -Systemen gibt es für jeden Warteschlangenmanager eine `QueueManager`-Zeilengruppe. Diese Attribute geben den Namen des Warteschlangenmanagers und den Namen des Verzeichnisses an, das die Dateien enthält, die diesem WS-Manager zugeordnet sind. Der Name des Verzeichnisses basiert auf dem Namen des Warteschlangenmanagers, wird aber umgesetzt, wenn der Name des WS-Managers kein gültiger Dateiname ist. Weitere Informationen zur Namensumsetzung finden Sie im Abschnitt [Informationen zu WebSphere MQ -Dateinamen](#).

### **Name =Warteschlangenmanagername**

Der Name des Warteschlangenmanagers.

### **Prefix =Präfix**

Gibt an, wo die Warteschlangenmanagerdateien gespeichert sind. Standardmäßig ist dieser Wert mit dem Wert identisch, der im Attribut `DefaultPrefix` der Informationen zu allen WS-Managern angegeben ist.

### **Directory =Name**

Der Name des Unterverzeichnisses unter dem Verzeichnis `<prefix>\QMGRS`, in dem die Warteschlangenmanager-Dateien gespeichert sind. Dieser Name basiert auf dem Namen des Warteschlangenmanagers, kann aber umgesetzt werden, wenn ein doppelter Name vorhanden ist oder wenn der Name des WS-Managers kein gültiger Dateiname ist.

**DataPath=Pfad**

Ein expliziter Datenpfad, der beim Erstellen des Warteschlangenmanagers bereitgestellt wurde, überschreibt Prefix und Directory als Pfad zu den WS-Manager-Daten.

**InstallationName=Name**

Der Name der WebSphere MQ -Installation, die diesem Warteschlangenmanager zugeordnet ist. Befehle aus dieser Installation müssen bei der Interaktion mit diesem WS-Manager verwendet werden. Wenn kein Wert für InstallationName vorhanden ist, wird der WS-Manager einer Installation von WebSphere MQ vor Version 7.1 zugeordnet.

**Zugehörige Konzepte**

„WS-Manager einer Installation zuordnen“ auf Seite 18

Wenn Sie einen WS-Manager erstellen, wird er automatisch der Installation zugeordnet, die den **crtmqm**-Befehl ausgegeben hat. Unter UNIX, Linux, and Windows können Sie die Installation ändern, die einem Warteschlangenmanager zugeordnet ist, indem Sie den Befehl **setmqm** verwenden.

**Sicherheit**

Verwenden Sie die Zeilengruppe Security in der Datei `qm.ini`, um Optionen für den OAM (Object Authority Manager) anzugeben.

**ClusterQueueAccessControl = RQMName | Xmitq**

Legen Sie dieses Attribut fest, um die Zugriffssteuerung von Clusterwarteschlangen oder vollständig qualifizierten Warteschlangen zu überprüfen, die auf Cluster-WS-Managern gehostet sind.

**RQMName**

Die Profile, die auf die Zugriffssteuerung von fern gehosteten Warteschlangen überprüft werden, sind benannte Warteschlangen oder benannte WS-Manager-Profile.

**XMITQ**

Die Profile, die auf die Zugriffssteuerung von fern gehosteten Warteschlangen überprüft werden, werden in `SYSTEM.CLUSTER.TRANSMIT.QUEUE` aufgelöst.

Xmitq ist der Standardwert.

**GroupModel=GlobalGroups**

Dieses Attribut bestimmt, ob der OAM globale Gruppen prüft, wenn er die Gruppenzugehörigkeit eines Benutzers in Windows bestimmt.

Standardmäßig werden globale Gruppen nicht überprüft.

**GlobalGroups**

Der OAM überprüft globale Gruppen.

Wenn GlobalGroups festgelegt ist, akzeptieren die Berechtigungsbefehle **setmqaut**, **dspmqa** und **dmpmqaut** globale Gruppennamen (siehe Parameter **setmqaut -g**).

**Anmerkung:** Wenn Sie `ClusterQueueAccessControl=RQMName` festlegen und eine angepasste Implementierung des Berechtigungsservice mit weniger als `MQZAS_VERSION_6` ergibt, wird der Warteschlangenmanager nicht gestartet. Geben Sie in dieser Instanz entweder `ClusterQueueAccessControl=Xmitq` ein oder führen Sie ein Upgrade des angepassten Berechtigungsservice auf `MQZAS_VERSION_6` oder höher aus.

**Konfigurationsdaten des Warteschlangenmanagers ändern**

Über die hier beschriebenen Attribute wird die Konfiguration eines einzelnen Warteschlangenmanagers geändert. Sie überschreiben alle Einstellungen für WebSphere MQ.

Auf UNIX and Linux -Systemen ändern Sie die Konfigurationsdaten des Warteschlangenmanagers, indem Sie die Konfigurationsdatei `qm.ini` bearbeiten. Wenn Sie eine Zeilengruppe in `qm.ini` definieren, müssen Sie nicht jedes Element in einer neuen Zeile starten. Sie können entweder ein Semikolon (;) oder ein Nummernzeichen (#) verwenden, um einen Kommentar anzugeben.

Auf Windows -und Linux -Systemen (x86 -und x86-64 -Plattformen) können Sie einige Konfigurationsdaten mithilfe von IBM WebSphere MQ Explorer ändern. Da es jedoch erhebliche Auswirkungen auf die

Änderung installierbarer Services und ihrer Komponenten hat, sind die installierbaren Services in der IBM WebSphere MQ Explorer schreibgeschützt. Daher müssen Sie alle Änderungen an installierbaren Services vornehmen, indem Sie **regedit** unter Windows verwenden und die Datei qm.ini unter UNIX and Linux bearbeiten.

Weitere Informationen zum Ändern der Konfigurationsinformationen des Warteschlangenmanagers finden Sie in den folgenden Unterabschnitten:

### Zugehörige Konzepte

[„Konfiguration“ auf Seite 5](#)

Erstellen Sie einen oder mehrere Warteschlangenmanager auf einem oder mehreren Computern, und konfigurieren Sie sie auf Ihren Entwicklungs-, Test- und Produktionssystemen, um Nachrichten zu verarbeiten, die Ihre Geschäftsdaten enthalten.

[„Konfigurationsinformationen für IBM WebSphere MQ und Warteschlangenmanager ändern“ auf Seite 454](#)

Sie können das Verhalten von IBM WebSphere MQ oder einem einzelnen Warteschlangenmanager den Anforderungen Ihrer Installation entsprechend ändern.

### Zugehörige Tasks

[Planung](#)

[WebSphere MQ verwalten](#)

### Zugehörige Verweise

[„Attribute zum Ändern von IBM WebSphere MQ -Konfigurationsinformationen“ auf Seite 460](#)

Ändern Sie unter IBM WebSphere MQ für Windows-Systeme und unter IBM WebSphere MQ für Linux (x86- und x86-64-Plattformen) die Konfigurationsdaten mithilfe von IBM WebSphere MQ Explorer. Auf anderen Systemen ändern Sie die Informationen, indem Sie die Konfigurationsdatei 'mqm.ini' bearbeiten.

## Zugriffsmodus

**Access Mode** gilt nur für Windows -Server. Die Zeilengruppe AccessMode wird durch die Option -a [r] im Befehl **crtmqm** festgelegt. Ändern Sie die Zeilengruppe 'AccessMode' nicht, nachdem der WS-Manager erstellt wurde.

Zugriffsgruppe verwenden (-a [r] ) Option des Befehls **crtmqm** zur Angabe einer Windows-Sicherheitsgruppe, deren Mitgliedern uneingeschränkter Zugriff auf alle Warteschlangenmanager-Datendateien erteilt wird. Die Gruppe kann abhängig von der verwendeten Syntax entweder eine lokale oder eine globale Gruppe sein. Folgende Syntax ist für den Gruppennamen gültig:

*LocalGroup*

*Domänenname \ Name der globalen Gruppe*

*Name der globalen Gruppe@Domänenname*

Sie müssen die zusätzliche Zugriffsgruppe definieren, bevor Sie den Befehl 'crtmqm' mit der Option -a [r] ausführen.

Wenn Sie die Gruppe mit -ar anstelle von -a angeben, erhält die lokale Gruppe 'mqm' keinen Zugriff auf die Datendateien des Warteschlangenmanagers. Verwenden Sie diese Option, wenn das Dateisystem, in dem sich die WS-Manager-Datendateien befinden, keine Zugriffssteuerungseinträge für lokal definierte Gruppen unterstützt.

Die Gruppe ist üblicherweise eine globale Sicherheitsgruppe, die verwendet wird, um Multi-Instanz-Warteschlangenmanagern Zugriff auf einen gemeinsam genutzten Ordner für Warteschlangenmanagerdaten und Protokolle zu ermöglichen. Verwenden Sie die zusätzliche Sicherheitszugriffsgruppe, um Lese- und Schreibberechtigungen für den Ordner festzulegen oder darin enthaltene Warteschlangenmanagerdaten und Protokolldateien freizugeben.

Die zusätzliche Sicherheitszugriffsgruppe ist eine Alternative zur Verwendung der lokalen Gruppe mqm, um Berechtigungen für den Ordner festzulegen, der Warteschlangenmanagerdaten und -protokolle enthält. Im Gegensatz zur lokalen Gruppe mqm können Sie die zusätzliche Sicherheitszugriffsgruppe als lokale oder globale Gruppe definieren. Es muss eine globale Gruppe sein, um Berechtigungen für die freigegebenen

nen Ordner festlegen zu können, die die Daten und Protokolldateien enthalten, die von Multi-Instanz-Warteschlangenmanagern verwendet werden.

Das Windows-Betriebssystem überprüft die Zugriffsberechtigungen zum Lesen und Schreiben von Warteschlangenmanager-Daten und -Protokolldateien. Es prüft die Berechtigungen der Benutzer-ID, die Warteschlangenmanager-Prozesse ausführt. Welche Benutzer-ID überprüft wird, hängt davon ab, ob Sie den Warteschlangenmanager als Service oder ob Sie ihn interaktiv gestartet haben. Wenn Sie den Warteschlangenmanager als Service gestartet haben, überprüft das Windows-Betriebssystem die Benutzer-ID, die Sie mit dem Assistenten zur **Vorbereitung von IBM WebSphere MQ** konfiguriert haben. Wenn Sie den Warteschlangenmanager interaktiv gestartet haben, überprüft das Windows-Betriebssystem die Benutzer-ID, die den Befehl **strmqm** ausgeführt hat.

Die Benutzer-ID muss Mitglied der lokalen mqm-Gruppe sein, um den Warteschlangenmanager zu starten. Wenn die Benutzer-ID zu der zusätzlichen Sicherheitszugriffsgruppe gehört, hat der Warteschlangenmanager Lese- und Schreibzugriff auf die Dateien, für die diese Berechtigungen durch Verwendung der Gruppe erteilt wurden.

**Einschränkung:** Eine zusätzliche Sicherheitszugriffsgruppe kann nur im Windows-Betriebssystem angegeben werden. Wenn Sie in anderen Betriebssystemen eine zusätzliche Sicherheitszugriffsgruppe angeben, gibt der Befehl **crtmqm** einen Fehler zurück.

### Zugehörige Konzepte

[„Nicht gemeinsam genutzte WS-Manager-Daten und -Protokollverzeichnisse und -Dateien unter Windows schützen“ auf Seite 406](#)

[„Gemeinsam genutzte Warteschlangenmanagerdaten, Protokollverzeichnisse und Dateien unter Windows sichern“ auf Seite 403](#)

### Zugehörige Tasks

[„Multi-Instanz-Warteschlangenmanager auf Workstations oder Servern in einer Domäne erstellen“ auf Seite 377](#)

### Zugehörige Verweise

[crtmqm](#)

## Installierbare Services

Sie ändern die installierbaren Services unter Windows mithilfe von **regedit** und unter UNIX and Linux mithilfe der Zeilengruppe **Service** in der Datei **qm.ini**.

**Anmerkung:** Es hat erhebliche Auswirkungen auf die Änderung installierbarer Services und ihrer Komponenten. Aus diesem Grund sind die installierbaren Services im WebSphere MQ Explorer schreibgeschützt.

Um installierbare Services auf Windows -Systemen zu ändern, verwenden Sie **regedit** oder auf UNIX and Linux -Systemen die Zeilengruppe **Service** in der Datei **qm.ini**. Für jede Komponente in einem Service müssen Sie auch den Namen und den Pfad des Moduls angeben, das den Code für diese Komponente enthält. Auf UNIX and Linux -Systemen verwenden Sie hierfür die Zeilengruppe **ServiceComponent**.

### **Name =AuthorizationService|NameService**

Der Name des erforderlichen Service.

#### **AuthorizationService**

Für WebSphere MQ wird die Berechtigungsservicekomponente als Objektberechtigungsmanager oder OAM bezeichnet.

Die Zeilengruppe **AuthorizationService** und die zugehörige Zeilengruppe **ServiceComponent** werden bei der Erstellung des Warteschlangenmanagers automatisch hinzugefügt. Fügen Sie andere **ServiceComponent** -Zeilengruppen manuell hinzu.

#### **NameService**

Standardmäßig wird kein Namensservice bereitgestellt. Wenn Sie einen Namensservice benötigen, müssen Sie die Zeilengruppe **NameService** manuell hinzufügen.

**EntryPoints=Anzahl Einträge**

Die Anzahl der Eingangspunkte, die für den Service definiert wurden. Dazu gehören die Initialisierungs- und Beendigungseingangspunkte

**SecurityPolicy=Default|NTSIDsRequired (nur WebSphere MQ for Windows )**

Das Attribut SecurityPolicy gilt nur, wenn der angegebene Service der Standardberechtigungs-service ist, d. h. der OAM. Mit dem Attribut "SecurityPolicy" können Sie die Sicherheitsrichtlinie für die einzelnen Warteschlangenmanager angeben. Folgende Werte sind möglich:

**Default**

Verwenden Sie die Standardsicherheitsrichtlinie, die wirksam werden soll. Wenn für eine bestimmte Benutzer-ID keine Windows-Sicherheits-ID (NT SID) an den OAM übergeben wurde, wird versucht, die entsprechende SID durch Durchsuchen der relevanten Sicherheitsdatenbanken zu erhalten.

**NTSIDsRequired**

Übergeben Sie eine NT-SID an den OAM, wenn Sie Sicherheitsprüfungen durchführen.

Weitere Informationen finden Sie unter [Windows -Sicherheitskennungen \(SIDs\)](#).

**SharedBindingsUserId=Benutzertyp**

Das Attribut SharedBindingsUserId gilt nur, wenn es sich bei dem angegebenen Service um den Standardberechtigungs-service handelt, d. h. den OAM. Das Attribut SharedBindingsUserId wird nur in Verbindung mit gemeinsam genutzten Bindungen verwendet. Mit diesem Wert können Sie angeben, ob das Feld *UserIdentifier* in der *IdentityContext* -Struktur aus der Funktion MQZ\_AUTHENTICATE\_USER die effektive Benutzer-ID oder die reale Benutzer-ID ist. Informationen zur Funktion MQZ\_AUTHENTICATE\_USER finden Sie unter [MQZ\\_AUTHENTICATE\\_USER-Benutzer authentifizieren](#). Folgende Werte sind möglich:

**Default**

Der Wert des Feldes *UserIdentifier* wird als reale Benutzer-ID festgelegt.

**Reelle Zahl**

Der Wert des Feldes *UserIdentifier* wird als reale Benutzer-ID festgelegt.

**Effektiv**

Der Wert des Feldes *UserIdentifier* wird als effektive Benutzer-ID festgelegt.

**FastpathBindingsUserId=Benutzertyp**

Das Attribut FastpathBindingsUserId gilt nur, wenn es sich bei dem angegebenen Service um den Standardberechtigungs-service handelt, d. h. den OAM. Das Attribut 'FastpathBindingsUserId' wird nur in Verbindung mit den Direktaufrufbindungen verwendet. Mit diesem Wert können Sie angeben, ob das Feld *UserIdentifier* in der *IdentityContext* -Struktur aus der Funktion MQZ\_AUTHENTICATE\_USER die effektive Benutzer-ID oder die reale Benutzer-ID ist. Informationen zur Funktion MQZ\_AUTHENTICATE\_USER finden Sie unter [MQZ\\_AUTHENTICATE\\_USER-Benutzer authentifizieren](#). Folgende Werte sind möglich:

**Default**

Der Wert des Feldes *UserIdentifier* wird als reale Benutzer-ID festgelegt.

**Reelle Zahl**

Der Wert des Feldes *UserIdentifier* wird als reale Benutzer-ID festgelegt.

**Effektiv**

Der Wert des Feldes *UserIdentifier* wird als effektive Benutzer-ID festgelegt.

**IsolatedBindingsUserId =Benutzertyp**

Das Attribut IsolatedBindingsUserId gilt nur, wenn der angegebene Service der Standardberechtigungs-service ist, d. h. der OAM. Das Attribut IsolatedBindingsUserId wird nur für isolierte Bindungen verwendet. Mit diesem Wert können Sie angeben, ob das Feld *UserIdentifier* in der *IdentityContext* -Struktur aus der Funktion MQZ\_AUTHENTICATE\_USER die effektive Benutzer-ID oder die reale Benutzer-ID ist. Informationen zur Funktion MQZ\_AUTHENTICATE\_USER finden Sie unter [MQZ\\_AUTHENTICATE\\_USER-Benutzer authentifizieren](#). Folgende Werte sind möglich:

**Default**

Der Wert des Feldes *UserIdentifier* wird als effektive Benutzer-ID festgelegt.

**Reelle Zahl**

Der Wert des Feldes *UserIdentifier* wird als reale Benutzer-ID festgelegt.

**Effektiv**

Der Wert des Feldes *UserIdentifier* wird als effektive Benutzer-ID festgelegt.

Weitere Informationen zu installierbaren Services und Komponenten finden Sie unter [Installierbare Services und Komponenten für UNIX, Linux und Windows](#).

Weitere allgemeine Informationen zu Sicherheitsservices finden Sie unter [Setting up security on Windows, UNIX and Linux -Systeme](#).

**Zugehörige Verweise**

[Referenzinformationen zu installierbaren Services](#)

**Servicekomponenten**

Sie müssen Servicekomponenteninformationen angeben, wenn Sie einen neuen installierbaren Service hinzufügen. Verwenden Sie auf Windows -Systemen `regedit` und auf UNIX and Linux -Systemen die Zeilengruppe `ServiceComponent` in der Datei `qm.ini`. Die Berechtigungsservice-Zeilengruppe ist standardmäßig vorhanden, und die zugeordnete Komponente (OAM) ist aktiv.

Geben Sie die Servicekomponenten wie folgt an:

**Service =*ServiceName***

Der Name des erforderlichen Service. Dieser Wert muss mit dem Wert übereinstimmen, der im Attribut `Name` der Servicekonfigurationsinformationen angegeben ist.

**Name =*Komponentenname***

Der beschreibende Name der Servicekomponente. Dieser Name muss eindeutig sein und darf nur Zeichen enthalten, die für die Namen von WebSphere MQ -Objekten (z. B. Warteschlangennamen) gültig sind. Dieser Name tritt in Bedienernachrichten auf, die durch den Service generiert werden. Es wird empfohlen, diesen Namen mit einer Unternehmensmarke oder einer ähnlichen Unterscheidungszeichenfolge zu beginnen.

**Module =*modulname***

Der Name des Moduls, das den Code für diese Komponente enthält. Dies muss ein vollständiger Pfadname sein.

**ComponentDataSize =*Größe***

Die Größe des Komponentendatenbereichs (in Byte), der an die Komponente in jedem Aufruf übergeben wurde. Geben Sie Null an, wenn keine Komponentendaten erforderlich sind.

Weitere Informationen zu installierbaren Services und Komponenten finden Sie unter [Installierbare Services und Komponenten für UNIX, Linux und Windows](#).

**Warteschlangenmanagerprotokolle**

Verwenden Sie die Eigenschaftenseite des Log -Warteschlangenmanagers im IBM WebSphere MQ Explorer oder die Zeilengruppe `Log` in der Datei `qm.ini`, um Informationen zur Protokollierung auf einem Warteschlangenmanager anzugeben.

Standardmäßig werden diese Einstellungen aus den Standardprotokolleinstellungen des Warteschlangenmanagers übernommen (Beschreibung siehe „[Protokollstandardwerte für IBM WebSphere MQ](#)“ auf Seite 463). Ändern Sie diese Einstellungen nur, wenn Sie diesen Warteschlangenmanager auf eine andere Weise konfigurieren möchten.

Informationen zum Berechnen der Protokollgrößen finden Sie in „[Berechnen der Größe des Protokolls](#)“ auf Seite 436.

**Anmerkung:** Die in der folgenden Parameterliste angegebenen Grenzwerte werden von WebSphere MQ festgelegt. Durch die Begrenzung des Betriebssystems kann die maximal mögliche Protokollgröße reduziert werden.

**LogPrimaryFiles =3 |2-254 ( Windows ) |2-510 ( UNIX and Linux -Systeme)**

Die Protokolldateien, die beim Erstellen des Warteschlangenmanagers zugeordnet werden.



Die Mindestanzahl primärer Protokolldateien beträgt 2 und die maximale Anzahl 254 unter Windows bzw. 510 auf UNIX and Linux -Systemen. Der Standardwert ist 3.

Die Gesamtzahl der primären und sekundären Protokolldateien darf unter Windows nicht größer als 255 und auf UNIX and Linux -Systemen nicht kleiner als 511 sein.

Der Wert wird geprüft, wenn der WS-Manager erstellt oder gestartet wird. Sie können sie ändern, nachdem der WS-Manager erstellt wurde. Eine Änderung des Werts ist jedoch erst wirksam, wenn der Warteschlangenmanager erneut gestartet wird und der Effekt möglicherweise nicht sofort ausgeführt wird.

#### **LogSecondaryFiles = 2 | 1-253 ( Windows ) | 1-509 ( UNIX and Linux -Systeme)**

Die Protokolldateien, die zugeordnet werden, wenn die Primärdateien erschöpft sind.

Die minimale Anzahl sekundärer Protokolldateien ist 1 und die maximale Anzahl 253 unter Windows bzw. 509 auf Systemen mit UNIX and Linux . Die Standardanzahl ist 2.

Die Gesamtzahl der primären und sekundären Protokolldateien darf unter Windows nicht größer als 255 und auf UNIX and Linux -Systemen nicht kleiner als 511 sein.

Der Wert wird geprüft, wenn der Warteschlangenmanager gestartet wird. Sie können diesen Wert ändern, aber Änderungen werden erst wirksam, wenn der Warteschlangenmanager erneut gestartet wird, und selbst dann wird der Effekt möglicherweise nicht sofort wirksam.

#### **LogFilePages = Anzahl**

Die Protokolldaten werden in einer Reihe von Dateien mit dem Namen "Protokolldateien" festgehalten. Die Protokolldateigröße wird in Einheiten von 4-KB-Seiten angegeben.

Die Standardanzahl der Protokolldateiseiten beträgt 4096, wobei eine Protokolldateigröße von 16 MB angegeben wird.

Auf UNIX and Linux -Systemen beträgt die Mindestanzahl der Protokolldateiseiten 64 und unter Windows beträgt die Mindestanzahl der Protokolldateiseiten 32. In beiden Fällen beträgt die maximale Anzahl 65.535.

**Anmerkung:** Die Größe der Protokolldateien, die bei der Erstellung des Warteschlangenmanagers angegeben wurden, kann für einen Warteschlangenmanager nicht geändert werden.

#### **LogType=CIRCULAR | LINEAR**

Der Typ der Protokollierung, der vom Warteschlangenmanager verwendet werden soll. Sie können den Typ der Protokollierung nicht ändern, die nach der Erstellung des Warteschlangenmanagers verwendet werden soll. Informationen zum Erstellen eines Warteschlangenmanagers mit dem Typ der Protokollierung, die Sie benötigen, finden Sie in der Beschreibung des Attributs LogType in „[Protokollstandardwerte für IBM WebSphere MQ](#)“ auf Seite 463.

##### **CIRCULAR**

Starten Sie die Wiederherstellung nach einem Neustart mit Hilfe des Protokolls, um Transaktionen rückgängig zu machen, die sich in Bearbeitung befanden, als das System gestoppt wurde

Eine genauere Beschreibung des Protokolltyps CIRCULAR (Umlaufprotokollierung) finden Sie im Abschnitt „[Typen der Protokollierung](#)“ auf Seite 433.

##### **LINEAR**

Sowohl für die Wiederherstellung nach einem Neustart als auch für die Datenträger- oder Vorwärtswiederherstellung (durch das Erstellen verlorener oder beschädigter Daten durch Wiedergabe des Inhalts des Protokolls).

Eine ausführlichere Erläuterung der linearen Protokollierung finden Sie in „[Typen der Protokollierung](#)“ auf Seite 433 .

#### **LogBufferSeiten = 0 | 0-4096**

Die Größe des Speichers, der den Pufferdatensätzen für das Schreiben zugeordnet ist, wobei die Größe der Puffer in Einheiten von 4-KB-Seiten angegeben wird.

Die Mindestanzahl der Pufferseiten beträgt 18 und der Maximalwert 4096. Größere Puffer führen zu einem höheren Durchsatz, insbesondere bei größeren Nachrichten.



Wenn Sie 0 (die Standardeinstellung) angeben, wählt der Warteschlangenmanager die Größe aus. In WebSphere MQ Version 7.0 ist dies 512 (2048 KB).

Wenn Sie eine Zahl im Bereich von 1 bis 17 angeben, nimmt der WS-Manager standardmäßig den Wert 18 (72 KB) an. Wenn Sie eine Zahl im Bereich von 18 bis 4096 angeben, verwendet der WS-Manager die angegebene Zahl, um den zugeordneten Speicher festzulegen.

Der Wert wird geprüft, wenn der Warteschlangenmanager gestartet wird. Der Wert kann innerhalb der angegebenen Grenzwerte erhöht oder verringert werden. Eine Änderung des Werts ist jedoch erst wirksam, wenn der Warteschlangenmanager das nächste Mal gestartet wird.

### **LogPath=Verzeichnisname**

Das Verzeichnis, in dem sich die Protokolldateien für einen WS-Manager befinden. Dieser muss auf einer lokalen Einheit vorhanden sein, in die der Warteschlangenmanager schreiben kann, und zwar vorzugsweise auf einem anderen Laufwerk aus den Nachrichtenwarteschlangen. Die Angabe eines anderen Laufwerks bietet einen zusätzlichen Schutz im Falle eines Systemausfalls.

Der Standardwert lautet:

- C:\Program Files\IBM\WebSphere MQ\log in WebSphere MQ für Windows .
- /var/mqm/log in WebSphere MQ für UNIX and Linux -Systeme.

Sie können den Namen eines Verzeichnisses mit dem Befehl `crtmqm` mit dem Flag `-ld` angeben. Wenn ein Warteschlangenmanager erstellt wird, wird auch ein Verzeichnis unter dem WS-Manager-Verzeichnis erstellt, und dieses Verzeichnis wird verwendet, um die Protokolldateien zu speichern. Der Name dieses Verzeichnisses basiert auf dem Namen des Warteschlangenmanagers. Dadurch wird sichergestellt, dass der Protokolldateipfad eindeutig ist und dass er auch Einschränkungen bezüglich der Länge von Verzeichnisnamen entspricht.

Wenn Sie `-ld` im Befehl `crtmqm` nicht angeben, wird der Wert des Attributs `LogDefaultPath` verwendet.

In WebSphere MQ für UNIX and Linux -Systeme müssen die Benutzer-ID 'mqm' und die Gruppe 'mqm' über vollständige Berechtigungen für die Protokolldateien verfügen. Wenn Sie die Position dieser Dateien ändern, müssen Sie diese Berechtigungen selbst zuweisen. Dies ist nicht erforderlich, wenn sich die Protokolldateien an den Standardpositionen befinden, die im Lieferumfang des Produkts enthalten sind.

### **LogWriteIntegrity =SingleWrite|DoubleWrite| TripleWrite**

Die Methode, die die Protokollfunktion verwendet, um Protokollsätze zuverlässig zu schreiben.

#### **TripleWrite**

Dies ist die Standardmethode.

Beachten Sie, dass Sie **DoubleWrite** auswählen können, aber wenn Sie dies tun, interpretiert das System diese Option als **TripleWrite** .

#### **SingleWrite**

**Einfach schreiben** sollte nur verwendet werden, wenn das Dateisystem bzw. die Einheit, auf der sich das WebShere MQ-Wiederherstellungsprotokoll befindet, die Atomizität von 4-KB-Schreibvorgängen explizit garantiert.

Das heißt, wenn ein Schreiben einer 4-KB-Seite aus irgendeinem Grund fehlschlägt, sind die einzigen beiden möglichen Status entweder das Vorimage oder das Nachimage. Ein Zwischenzustand sollte nicht möglich sein.

## **Eingeschränkter Modus**

Diese Option gilt nur für UNIX and Linux -Systeme. Die Zeilengruppe `RestrictedMode` wird über die Option `-g` im Befehl `crtmqm` festgelegt. Ändern Sie diese Zeilengruppe nicht, nachdem der WS-Manager erstellt wurde. Wenn Sie die Option `-g` nicht verwenden, wird die Zeilengruppe nicht in der Datei 'qm.ini' erstellt.

Es gibt einige Verzeichnisse im Datenverzeichnis eines Warteschlangenmanagers, in denen IBM WebSphere MQ-Anwendungen Dateien erstellen, während sie mit dem Warteschlangenmanager verbunden

sind. Damit Anwendungen Dateien in diesen Verzeichnissen erstellen können, wird ihnen der World-Write-Zugriff gewährt:

- `/var/mqm/sockets/QMgrName/@ipcc/ssem/hostname/`
- `/var/mqm/sockets/QMgrName/@app/ssem/hostname/`
- `/var/mqm/sockets/QMgrName/zsocketapp/hostname/`

Dabei steht `<QMGRNAME>` für den Namen des Warteschlangenmanagers und `<hostname>` für den Hostnamen.

Auf einigen Systemen ist es nicht akzeptabel, allen Benutzern Schreibzugriff auf diese Verzeichnisse zu erteilen. Beispiel: Die Benutzer, die keinen Zugriff auf den Warteschlangenmanager benötigen. Der eingeschränkte Modus ändert die Berechtigungen der Verzeichnisse, in denen WS-Manager-Daten gespeichert werden. Auf die Verzeichnisse kann dann nur von den Mitgliedern der angegebenen Anwendungsgruppe zugegriffen werden. Die Berechtigungen für den gemeinsam genutzten Speicher von System V IPC, der für die Kommunikation mit dem Warteschlangenmanager verwendet wird, werden ebenfalls auf die gleiche Weise geändert.

Die Anwendungsgruppe ist der Name der Gruppe mit Mitgliedern, die über die Berechtigung zum Führen der folgenden Elemente verfügen:


- Ausführung von MQI-Anwendungen
- Aktualisierung aller IPCC-Ressourcen
- Ändern der Inhalte einiger Warteschlangenmanagerverzeichnisse

Gehen Sie wie folgt vor, um den eingeschränkten Modus für einen WS-Manager

- Der Ersteller des Warteschlangenmanagers muss sich in der Gruppe `mqm` und in der Anwendungsgruppe befinden.
- Die `mqm` -Benutzer-ID muss in der Anwendungsgruppe enthalten sein.
- Alle Benutzer, die den Warteschlangenmanager verwalten möchten, müssen sich in der Gruppe `mqm` und in der Anwendungsgruppe befinden.
- Alle Benutzer, die IBM WebSphere MQ-Anwendungen ausführen wollen, müssen sich in der Anwendungsgruppe befinden.

Jeder MQCONN- oder MQCONNX-Aufruf, der von einem nicht in der Anwendungsgruppe enthaltenen Benutzer ausgegeben wird, schlägt mit dem Ursachencode `MQRC_Q_MGR_NOT_AVAILABLE` fehl.

Der eingeschränkte Modus wird mit dem IBM WebSphere MQ-Berechtigungs-service betrieben. Daher müssen Sie Benutzern auch die Berechtigung erteilen, eine Verbindung mit IBM WebSphere MQ herzustellen und auf die Ressourcen zuzugreifen. Dies geschieht mit dem IBM WebSphere MQ-Berechtigungs-service.

 Weitere Informationen zur Konfiguration des IBM WebSphere MQ Authorization Service finden Sie unter [Setting up security on Windows, UNIX and Linux systems](#).

Verwenden Sie nur den eingeschränkten Modus von IBM WebSphere MQ, wenn die vom Berechtigungs-service bereitgestellte Steuerung keine ausreichende Isolation von Warteschlangenmanagerressourcen bietet.

## XA-Ressourcenmanager

Auf der Eigenschaftenseite des `XA resource manager` -Warteschlangenmanagers im IBM WebSphere MQ Explorer oder in der Zeilengruppe `XAResourceManager` in der Datei `qm.ini` können Sie die folgenden Informationen zu den Ressourcenmanagern angeben, die an den vom Warteschlangenmanager koordinierten globalen Arbeitseinheiten beteiligt sind.

Fügen Sie die Konfigurationsinformationen für den XA-Ressourcenmanager für jede Instanz eines Ressourcenmanagers, der an globalen Arbeitseinheiten beteiligt ist, manuell hinzu; es werden keine Standardwerte bereitgestellt.

Weitere Informationen zu Ressourcenmanagerattributen finden Sie unter [Datenbankkoordinierung](#).

**Name =name (obligatorisch)**

Dieses Attribut gibt die Ressourcenmanagerinstanz an.

Der Wert für Name kann bis zu 31 Zeichen lang sein. Sie können den Namen des Ressourcenmanagers verwenden, wie er in der XA-Switch-Struktur definiert ist. Wenn Sie jedoch mehr als eine Instanz desselben Ressourcenmanagers verwenden, müssen Sie für jede Instanz einen eindeutigen Namen erstellen. Sie können die Eindeutigkeit sicherstellen, indem Sie z. B. den Namen der Datenbank in die Zeichenfolge Name einschließend.

WebSphere MQ verwendet den Wert für Name in Nachrichten und in der Ausgabe des Befehls `dspmqrtn`.

Ändern Sie den Namen einer Ressourcenmanagerinstanz nicht oder löschen Sie den zugehörigen Eintrag aus den Konfigurationsdaten, sobald der zugehörige Warteschlangenmanager gestartet wurde und der Name des Ressourcenmanagers wirksam ist.

**SwitchFile=Name (obligatorisch)**

Der vollständig qualifizierte Name der Ladedatei, die die XA-Switchstruktur des Ressourcenmanagers enthält.

Wenn Sie einen 64-Bit-Warteschlangenmanager mit 32-Bit-Anwendungen verwenden, sollte der Wert von name nur den Basisnamen der Ladedatei enthalten, die die XA-Switchstruktur des Ressourcenmanagers enthält.

Die 32-Bit-Datei wird aus dem durch `ExitsDefaultPath` angegebenen Pfad in die Anwendung geladen.

Die 64-Bit-Datei wird aus dem durch `ExitsDefaultPath64` angegebenen Pfad in den WS-Manager geladen.

**XAOpenString=Zeichenfolge (optional)**

Die Zeichenfolge der Daten, die an den Eingangspunkt `xa_open` des Ressourcenmanagers übergeben werden sollen. Der Inhalt der Zeichenfolge hängt vom Ressourcenmanager selbst ab. Die Zeichenfolge könnte z. B. die Datenbank angeben, auf die diese Instanz des Ressourcenmanagers zugreifen soll. Weitere Informationen zum Definieren dieses Attributs finden Sie unter:

- [Ressourcenmanager-Konfigurationsinformationen für DB2](#)
- [Ressourcenmanagerkonfigurationsinformationen für Oracle hinzufügen](#)
- [Ressourcenmanager-Konfigurationsinformationen für Sybase hinzufügen](#)
- [Ressourcenmanager-Konfigurationsinformationen für Informix](#)

und ziehen Sie die Dokumentation zu Ihrem Ressourcenmanager nach der entsprechenden Zeichenfolge zu Rate.

**XACloseString=Zeichenfolge (optional)**

Die Zeichenfolge der Daten, die an den Eingangspunkt `xa_close` des Ressourcenmanagers übergeben werden sollen. Der Inhalt der Zeichenfolge hängt vom Ressourcenmanager selbst ab. Weitere Informationen zum Definieren dieses Attributs finden Sie unter:

- [Ressourcenmanager-Konfigurationsinformationen für DB2](#)
- [Ressourcenmanagerkonfigurationsinformationen für Oracle hinzufügen](#)
- [Ressourcenmanager-Konfigurationsinformationen für Sybase hinzufügen](#)
- [Ressourcenmanager-Konfigurationsinformationen für Informix](#)

und ziehen Sie die Datenbankdokumentation für die entsprechende Zeichenfolge zu Rate.

**ThreadOfControl=THREAD|PROCESS**

Dieses Attribut ist für WebSphere MQ für Windows obligatorisch. Der Warteschlangenmanager verwendet diesen Wert für die Serialisierung, wenn er den Ressourcenmanager von einem seiner eigenen Multithread-Prozesse aus aufrufen muss.

## THREAD

Der Ressourcenmanager ist vollständig *Threadwissen*. In einem Multithread- WebSphere MQ -Prozess können XA-Funktionsaufrufe an den externen Ressourcenmanager aus mehreren Threads gleichzeitig erfolgen.

## PROCESS

Der Ressourcenmanager ist nicht *threadsicher*. In einem Multithread-Prozess von WebSphere MQ kann jeweils nur ein XA-Funktionsaufruf an den Ressourcenmanager abgesetzt werden.

Der Eintrag `ThreadOfControl` gilt nicht für XA-Funktionsaufrufe, die vom WS-Manager in einem Multithread-Anwendungsprozess abgesetzt werden. Im Allgemeinen erfordert eine Anwendung, die parallele Arbeitseinheiten in verschiedenen Threads hat, diesen Modus der Operation, die von jedem der Ressourcenmanager unterstützt wird.

## Attribute von Zeilengruppen für Kanäle

Diese Attribute bestimmen die Konfiguration eines Kanals.

Diese Informationen gelten nicht für WebSphere MQ für die z/OS -Plattform.

Verwenden Sie die Eigenschaftenseite des `Channels` -Warteschlangenmanagers im WebSphere MQ Explorer oder die Zeilengruppe `CHANNELS` in der Datei `qm.ini`, um Informationen zu Kanälen anzugeben.

### **MaxChannels=100 |Anzahl**

Die maximale Anzahl zulässiger *aktueller* Kanäle.

Der Wert muss im Bereich von 1 bis 65535 liegen. Der Standardwert ist 100.

### **MaxActiveChannels =MaxChannels\_Wert**

Die maximale Anzahl der Kanäle, die zu einem beliebigen Zeitpunkt *aktiv* sein dürfen. Der Standardwert ist der Wert, der für das Attribut `MaxChannels` angegeben ist.

### **MaxInitiators=3 |Anzahl**

Die maximale Anzahl der Initiatoren. Der Standardwert und der Maximalwert sind 3.

### **MQIBindType= FASTPATH | STANDARD**

Die Bindung für Anwendungen:

#### **FASTPATH**

Kanäle werden über `MQCONN FASTPATH` verbunden; es gibt keinen Agentenprozess.

#### **STANDARD**

Kanäle verbinden sich mit `STANDARD`.

### **PipeLineLength =1 |Anzahl**

Die maximale Anzahl gleichzeitiger Threads, die ein Kanal verwenden wird. Der Standardwert ist 1. Alle Werte größer als 1 werden wie der Wert 2 behandelt.

Wenn Sie Pipelining verwenden, konfigurieren Sie die Warteschlangenmanager an beiden Enden des Kanals, um einen *PipeLineLength* größer als 1 zu verwenden.

**Anmerkung:** Pipelining ist nur für TCP/IP-Kanäle wirksam.

### **AdoptNewMCA =N0| SVR | SDR | RCVR | CLUSRCVR | ALL | FASTPATH**

Wenn WebSphere MQ eine Anforderung zum Starten eines Kanals empfängt, aber feststellt, dass bereits eine Instanz des Kanals aktiv ist, muss in einigen Fällen die vorhandene Kanalinstanz gestoppt werden, damit die neue gestartet werden kann. Mit dem Attribut `AdoptNewMCA` können Sie steuern, welche Kanaltypen auf diese Weise beendet werden können.

Wenn Sie das Attribut `AdoptNewMCA` für einen bestimmten Kanaltyp angeben, der neue Kanal jedoch nicht gestartet werden kann, weil bereits eine übereinstimmende Kanalinstanz ausgeführt wird, gilt Folgendes:

1. Der neue Kanal versucht, die vorherige zu stoppen, indem er sie zum Beenden anfordert.
2. Wenn der vorherige Kanalserver nicht auf diese Anforderung reagiert, wenn das Warteintervall für `AdoptNewMCATimeout` abläuft, wird der Thread oder der Prozess für den vorherigen Kanalserver beendet.

3. Wenn der vorherige Kanalserver nach Schritt 2 nicht beendet wurde und das Warteintervall für `AdoptNewMCATimeout` ein zweites Mal abläuft, beendet WebSphere MQ den Kanal mit dem Fehler `CHANNEL IN USE`.

Die Funktion "AdoptNewMCA" gilt für Server-, Sender-, Empfänger- und Clusterempfängerkanäle. Im Fall eines Sende- oder Serverkanals kann im empfangenden Warteschlangenmanager nur eine Instanz eines Kanals mit einem bestimmten Namen ausgeführt werden. Im Fall eines Empfangs- oder Clusterempfängerkanals werden möglicherweise mehrere Instanzen eines Kanals mit einem bestimmten Namen im empfangenden Warteschlangenmanager ausgeführt, aber es kann jeweils nur eine Instanz von einem bestimmten fernen Warteschlangenmanager ausgeführt werden.

**Anmerkung:** `AdoptNewMCA` wird auf Requester- oder Serververbindungskanälen nicht unterstützt.

Geben Sie einen oder mehrere durch Kommas oder Leerzeichen voneinander getrennte Werte aus der folgenden Liste an:

**Nein**

Das Feature `AdoptNewMCA` ist nicht erforderlich. Dies ist die Standardeinstellung.

**SVR**

Verwalten Sie Serverkanäle.

**SDR**

Adoptions-Senderkanäle.

**RCVR**

Empfänger-Channels.

**CLUSRCVR**

Clusterempfängerkanäle verwalten.

**ALLE**

Alle Kanaltypen mit Ausnahme von `FASTPATH`-Kanälen werden hinzugefügt.

**FASTPATH**

Geben Sie den Kanal an, wenn es sich um einen `FASTPATH`-Kanal handelt. Dies geschieht nur, wenn auch der entsprechende Kanaltyp angegeben ist, z. B. `AdoptNewMCA=RCVR, SVR, FASTPATH`.

**Achtung!:** Das Attribut "AdoptNewMCA" verhält sich möglicherweise in unvorhersehbarer Weise mit `FASTPATH`-Kanälen. Gehen Sie mit großer Vorsicht vor, wenn Sie das Attribut `AdoptNewMCA` für `FASTPATH`-Kanäle aktivieren.

**AdoptNewMCATimeout=60 | 1-3600**

Die Zeit in Sekunden, die die neue Kanalinstanz wartet, bis die alte Kanalinstanz beendet wird. Geben Sie einen Wert im Bereich von 1 bis 3600 an. Der Standardwert ist 60.

**AdoptNewMCACheck = QM | ADRESSE | NAME | ALL**

Der Typ der Überprüfung, die erforderlich ist, wenn das Attribut `AdoptNewMCA` aktiviert wird. Wenn möglich, führen Sie eine vollständige Überprüfung durch, um die Kanäle vor dem Herunterfahren, versehentlich oder böswillig zu schützen. Überprüfen Sie mindestens, ob die Kanalnamen übereinstimmen.

Geben Sie einen oder mehrere der folgenden Werte an, die im Fall von `QM`, `NAME` oder `ALL` durch Kommas oder Leerzeichen getrennt sind:

**QM**

Überprüfen Sie, ob die Namen der WS-Manager übereinstimmen.

Beachten Sie, dass der Name des Warteschlangenmanagers selbst und nicht die QMID übereinstimmt.

**ADDRESS**

Überprüfen Sie die IP-Adresse der DFV-Quelle. Zum Beispiel die TCP/IP-Adresse.

**Anmerkung:** Durch Komma getrennte `CONNNAME`-Werte gelten für Zieladressen und sind daher für diese Option nicht relevant.

Wenn ein Warteschlangenmanager mit mehreren Instanzen von `hosta` zu `hostb` nicht ausgeführt werden kann, verwenden alle abgehenden Kanäle dieses Warteschlangenmanagers die Quellen-IP-Adresse von `hostb`. Wenn sich dieser Wert nicht von `hosta` unterscheidet, kann `AdoptNewMCACheck = ADDRESS` nicht übereinstimmen.

Sie können SSL oder TLS mit gegenseitiger Authentifizierung verwenden, um zu verhindern, dass ein Angreifer einen vorhandenen aktiven Kanal aufbricht. Alternativ können Sie eine HACMP-Typ-lösung mit IP-Übernahme anstelle von Warteschlangenmanagern mit mehreren Instanzen verwenden oder eine Netzlastausgleichsfunktion verwenden, um die Quellen-IP-Adresse zu maskieren.

**NAME**

Überprüfen Sie, ob die Kanalnamen übereinstimmen.

**ALLE**

Suchen Sie nach übereinstimmenden WS-Managernamen, der DFV-Adresse und nach übereinstimmenden Kanalnamen.

Der Standardwert ist `AdoptNewMCACheck=NAME , ADDRESS , QM`.

**Zugehörige Konzepte**

„Kanalstatus“ auf Seite 61

Ein Kanal kann zu einem beliebigen Zeitpunkt in einem von vielen Status sein. Einige Staaten haben auch Unterzustände. Aus einem bestimmten Zustand kann ein Kanal in andere Zustände übergehen.

**TCP, LU62, NETBIOS und SPX**

Verwenden Sie diese WS-Manager-Eigenschaftenseiten oder Zeilengruppen in der Datei `qm.ini`, um Netzprotokollkonfigurationsparameter anzugeben. Sie überschreiben die Standardattribute für Kanäle.

**TCP**

Verwenden Sie die Eigenschaftenseite TCP des Warteschlangenmanagers in IBM WebSphere MQ Explorer oder die Zeilengruppe TCP in der Datei '`qm.ini`', um TCP/IP-Konfigurationsparameter (Transmission Control Protocol/Internet Protocol) anzugeben.

**Port =1414|Portnummer**

Die Standardportnummer (in Dezimalschreibweise) für TCP/IP-Sitzungen. Die *bekannte* Portnummer für WebSphere MQ ist 1414.

**Library1 =DLLName1 (nur WebSphere MQ für Windows )**

Der Name der TCP/IP-Sockets-DLL.

Der Standardwert ist `WSOCK32`.

**KeepAlive=NO|JA**

Schalten Sie die KeepAlive-Funktion ein oder aus. `KeepAlive=YES` bewirkt, dass TCP/IP in regelmäßigen Abständen überprüft, ob das andere Ende der Verbindung noch verfügbar ist. Ist dies nicht der Fall, wird der Kanal geschlossen.

**ListenerBacklog=number**

Überschreiben Sie die Standardanzahl ausstehender Anforderungen für den TCP/IP-Listener.

Beim Empfang über TCP/IP wird eine maximale Anzahl ausstehender Verbindungsanforderungen festgelegt. Dies kann als *Rückstand* von Anforderungen betrachtet werden, die auf den TCP/IP-Port warten, bis der Listener die Anforderung akzeptiert hat. Die Werte des Standard-Listener-Rückstandsprotokolls werden in [Tabelle 33](#) auf Seite 478 angezeigt.

| <i>Tabelle 33. Ausstehende Standardverbindungsanforderungen (TCP)</i> |                                         |
|-----------------------------------------------------------------------|-----------------------------------------|
| <b>Plattform</b>                                                      | <b>Standardwert für ListenerBacklog</b> |
| Windows -Server                                                       | 100                                     |
| Windows -Workstation                                                  | 5                                       |
| Linux                                                                 | 100                                     |

| <i>Tabelle 33. Ausstehende Standardverbindungsanforderungen (TCP) (Forts.)</i> |                                         |
|--------------------------------------------------------------------------------|-----------------------------------------|
| <b>Plattform</b>                                                               | <b>Standardwert für ListenerBacklog</b> |
| Solaris                                                                        | 100                                     |
| HP-UX                                                                          | 20                                      |
| AIX V4.2 oder höher                                                            | 100                                     |
| AIX V4.1 oder früher                                                           | 10                                      |

**Anmerkung:** Einige Betriebssysteme unterstützen einen größeren Wert als der angezeigte Standardwert. Verwenden Sie diese Option, um das Erreichen der Verbindungsgrenze zu vermeiden.

Umgekehrt können einige Betriebssysteme die Größe des TCP-Rückstageprotokolls begrenzen, so dass der effektive TCP-Rückstand kleiner als hier angefordert werden kann.

Wenn der Rückstand die in [Tabelle 33 auf Seite 478](#) angezeigten Werte erreicht, wird die TCP/IP-Verbindung abgelehnt und der Kanal kann nicht starten. Bei Nachrichtenkanälen führt dies dazu, dass der Kanal in einen RETRY-Status eingeht und die Verbindung zu einem späteren Zeitpunkt erneut versucht. Bei Clientverbindungen erhält der Client einen MQRC\_Q\_MGR\_NOT\_AVAILABLE-Ursachencode von MQCONN und wiederholt die Verbindung zu einem späteren Zeitpunkt.

**SvrSndBuffSize=32768|zahl**

Die Größe des TCP/IP-Sendepuffers (in Byte), der vom Serverende eines Clientverbindungs-Serververbindungskanals verwendet wird.

**SvrRcvBuffSize=32768|zahl**

Die Größe des TCP/IP-Empfangspuffers (in Byte), der vom Serverende eines Clientverbindungs-Serververbindungskanals verwendet wird.

**Connect\_Timeout=0|zahl**

Die Anzahl der Sekunden, bevor ein Versuch unternommen wird, das Socket-Zeitlimit zu verbinden. Der Standardwert 0 gibt an, dass kein Verbindungszeitlimit vorhanden ist.

**LU62 (nur WebSphere MQ für Windows )**

Verwenden Sie die Eigenschaftenseite LU6 . 2 des Warteschlangenmanagers in IBM WebSphere MQ Explorer oder die Zeilengruppe LU62 in der Datei 'qm.ini', um Konfigurationsparameter für die SNA LU 6.2-Protokolle anzugeben.

**TP-Name**

Der Name des TP-Namens, der auf dem fernen Standort gestartet werden soll.

**Library1 =DL-LName 1**

Der Name der APPC-DLL.

Der Standardwert ist WCPIC32.

**Library2 =DLL-Name2**

Wie Library1, wird verwendet, wenn der Code in zwei separaten Bibliotheken gespeichert ist.

Der Standardwert ist WCPIC32.

**NETBIOS (nur WebSphere MQ für Windows )**

Verwenden Sie die Eigenschaftenseite Netbios des Warteschlangenmanagers in IBM WebSphere MQ Explorer oder die Zeilengruppe NETBIOS in der Datei 'qm.ini', um Konfigurationsparameter für die NetBIOS-Protokolle anzugeben.

**LocalName=Name**

Der Name, unter dem diese Maschine im LAN bekannt ist.

**AdapterNum=0|Adapternummer**

Die Nummer des LAN-Adapters. Der Standardwert ist Adapter 0.

**NumSess=1|Anzahl\_Sitzungen**

Die Anzahl der Sitzungen, die zugeordnet werden sollen. Der Standardwert ist 1.

**NumCmds=1|Anzahl\_Befehle**

Die Anzahl der Befehle, die zugeordnet werden sollen. Der Standardwert ist 1.

**NumNames=1|Anzahl\_Namen**

Die Anzahl der zuzuordnende Namen. Der Standardwert ist 1.

**Library1 =DLLName1**

Der Name der NetBIOS-DLL.

Der Standardwert ist NETAPI32.

**SPX (nur WebSphere MQ für Windows )**

Verwenden Sie die Eigenschaftenseite SPX des Warteschlangenmanagers in IBM WebSphere MQ Explorer oder die Zeilengruppe SPX in der Datei 'qm.ini', um Konfigurationsparameter für die SPX-Protokolle anzugeben.

**Socket =5E86|Socketnummer**

Die SPX-Socket-Nummer in Hexadezimalschreibweise. Der Standardwert ist X'5E86 '.

**BoardNum=0|Adapternummer**

Die LAN-Adapternummer. Der Standardwert ist Adapter 0.

**KeepAlive = NO | YES**

Schalten Sie die KeepAlive-Funktion ein oder aus.

KeepAlive=YES bewirkt, dass SPX in regelmäßigen Abständen überprüft, ob das andere Ende der Verbindung noch verfügbar ist. Ist dies nicht der Fall, wird der Kanal geschlossen.

**Library1 =DLLName1**

Der Name der SPX-DLL.

Der Standardwert ist WSOCK32.DLL.

**Library2 =DLL-Name2**

Dasselbe gilt für LibraryName1, das verwendet wird, wenn der Code in zwei separaten Bibliotheken gespeichert wird.

Der Standardwert ist WSOCK32.DLL.

**ListenerBacklog=number**

Überschreiben Sie die Standardanzahl ausstehender Anforderungen für den SPX-Listener.

Beim Empfang auf SPX wird eine maximale Anzahl ausstehender Verbindungsanforderungen festgelegt. Dies kann als *Rückstand* von Anforderungen betrachtet werden, die auf das SPX-Socket warten, damit der Listener die Anforderung akzeptiert. Die Werte des Standard-Listener-Rückstandsprotokolls werden in [Tabelle 34 auf Seite 480](#) angezeigt.

| <i>Tabelle 34. Ausstehende Standardverbindungsanforderungen (SPX)</i> |                                         |
|-----------------------------------------------------------------------|-----------------------------------------|
| <b>Plattform</b>                                                      | <b>Standardwert für ListenerBacklog</b> |
| Windows -Server                                                       | 100                                     |
| Windows -Workstation                                                  | 5                                       |

**Anmerkung:** Einige Betriebssysteme unterstützen einen größeren Wert als der angezeigte Standardwert. Verwenden Sie diese Option, um das Erreichen der Verbindungsgrenze zu vermeiden.

Umgekehrt können einige Betriebssysteme die Größe des SPX-Rückprotokolls begrenzen, so dass der effektive SPX-Rückstand kleiner als hier angefordert werden kann.

Wenn der Rückstand die in [Tabelle 34 auf Seite 480](#) angezeigten Werte erreicht, wird die SPX-Verbindung abgelehnt und der Kanal kann nicht starten. Bei Nachrichtenkanälen führt dies dazu, dass der Kanal in einen RETRY-Status eingeht und die Verbindung zu einem späteren Zeitpunkt erneut versucht. Bei Clientverbindungen empfängt der Client einen Ursachencode MQRC\_Q\_MGR\_NOT\_AVAILABLE von MQCONN und sollte die Verbindung zu einem späteren Zeitpunkt wiederholen.



## Exitpfad

Über die Warteschlangenmanager-Eigenschaftenseite `Exits` in IBM WebSphere MQ Explorer oder die Zeilengruppe `ExitPath` in der Datei `qm.ini` können Sie den Pfad für Benutzerexitprogramme auf dem Warteschlangenmanagersystem angeben.

### **ExitsDefaultPath =zeichenfolge**

Das Attribut `"ExitsDefaultPath"` gibt die Position von an:

- 32-Bit-Kanalexits für Clients
- 32-Bit-Kanalexits und Datenkonvertierungsexits für Server
- Nicht qualifizierte XA-Switchloaddateien

### **ExitsDefaultPath64 =zeichenfolge**

Das Attribut `"ExitsDefaultPath64"` gibt die Position von an:

- 64-Bit-Kanalexits für Clients
- 64-Bit-Kanalexits und Datenkonvertierungsexits für Server
- Nicht qualifizierte XA-Switchloaddateien

## API-Exits

Verwenden Sie für einen Server die Eigenschaftenseite `Exits` des Warteschlangenmanagers im IBM WebSphere MQ Explorer oder die Zeilengruppe `ApiExitLocal` in der Datei `qm.ini`, um API-Exitroutinen für einen Warteschlangenmanager zu identifizieren. Ändern Sie für einen Client die Zeilengruppe `ApiExitLocal` in der Datei `mqlclient.ini`, um API-Exitroutinen für einen Warteschlangenmanager zu identifizieren.

Auf Windows-Systemen können Sie auch den Befehl `'amqmdain'` verwenden, um die Einträge für API-Exits zu ändern. (Um API-Exitroutinen für alle Warteschlangenmanager zu identifizieren, verwenden Sie die Zeilengruppen `ApiExitCommon` und `ApiExitTemplate`, wie in [„API-Exits“](#) auf Seite 466 beschrieben.)

Beachten Sie, dass die Nachricht vom Server nicht umgesetzt werden muss, damit der API-Exit ordnungsgemäß funktioniert. Nachdem der API-Exit die Nachricht verarbeitet hat, muss die Nachricht dann auf dem Client konvertiert werden. Daher ist es erforderlich, dass Sie alle Konvertierungsexits auf dem Client installiert haben.

Eine vollständige Beschreibung der Attribute für diese Zeilengruppen finden Sie im Abschnitt [API-Exits konfigurieren](#).

## Zeilengruppe `QMErrorLog` unter UNIX, Linux, and Windows

Verwenden Sie die Eigenschaftenseite des Extended -Warteschlangenmanagers im WebSphere MQ Explorer oder die Zeilengruppe `QMErrorLog` in der Datei `qm.ini`, um die Operation und den Inhalt der Fehlerprotokolle des Warteschlangenmanagers anzupassen.



**Achtung:** Sie können die Änderungen nur mit WebSphere MQ Explorer vornehmen, wenn Sie einen lokalen Warteschlangenmanager auf der Plattform Windows verwenden.

### **ErrorLogSize =maxsize**

Gibt die Größe des Fehlerprotokolls des Warteschlangenmanagers an, bei dem es in die Sicherung kopiert wird. `maxsize` muss im Bereich von 32768 bis 2147483648 Byte liegen. Wenn `ErrorLogSize` nicht angegeben wird, wird der Standardwert von 2097152 Byte (2 MB) verwendet.

### **ExcludeMessage=msgIds**

Gibt Nachrichten an, die nicht in das Fehlerprotokoll des Warteschlangenmanagers geschrieben werden sollen. Wenn Ihr WebSphere MQ -System stark ausgelastet ist und viele Kanäle gestoppt und gestartet werden, wird eine große Anzahl von Informationsnachrichten an die z/OS -Konsole und das Hardcopy-Protokoll gesendet. Die WebSphere MQ-IMS -Bridge und der Puffermanager erzeugen möglicherweise auch eine große Anzahl von Informationsnachrichten, sodass das Ausschließen von

Nachrichten verhindert, dass Sie bei Bedarf eine große Anzahl von Nachrichten empfangen. *msgIds* ist eine durch Kommas getrennte Liste mit einer Auswahl der folgenden Nachrichten-IDs:

- 5211-Maximale Länge des Eigenschaftsnamens überschritten.
- 5973-Verteilte Publish/Subscribe-Subskription unterdrückt
- 5974-Verteilte Publish/Subscribe-Veröffentlichung unterdrückt
- 6254-Das System konnte die gemeinsam genutzte Bibliothek nicht dynamisch laden.
- 7234-Anzahl der geladenen Nachrichten
- 9001-Kanalprogramm normal beendet
- 9002-Kanalprogramm gestartet
- 9202-Ferner Host nicht verfügbar
- 9208-Fehler beim Empfangen vom Host
- 9209-Verbindung geschlossen
- 9228-Kanalantworter kann nicht gestartet werden
- 9489-Maximale Anzahl der SVRCONN-Instanzen überschritten
- 9490-SVRCONN max-Instanzen pro Clientgrenzwert überschritten
- 9508-Verbindung zum WS-Manager kann nicht hergestellt werden
- 9524-Ferner WS-Manager nicht verfügbar
- 9528-Benutzer hat das Schließen des Kanals angefordert
- 9558-Ferner Kanal ist nicht verfügbar
- 9637-Kanal fehlt ein Zertifikat
- 9776-Kanal wurde von Benutzer-ID blockiert
- 9777-Kanal wurde durch NOACCESS-Zuordnung blockiert
- 9782-Verbindung wurde durch Adresse blockiert
- 9999-Kanalprogramm abnormal beendet

### **SuppressMessage=msgIds**

Gibt Nachrichten an, die nur einmal in einem angegebenen Zeitintervall in das Fehlerprotokoll des Warteschlangenmanagers geschrieben werden. Wenn Ihr WebSphere MQ -System stark ausgelastet ist und viele Kanäle gestoppt und gestartet werden, wird eine große Anzahl von Informationsnachrichten an die z/OS -Konsole und das Hardcopy-Protokoll gesendet. Die WebSphere MQ-IMS -Bridge und der Puffermanager erzeugen möglicherweise auch eine große Anzahl von Informationsnachrichten, sodass die Unterdrückung von Nachrichten verhindert, dass Sie eine Reihe sich wiederholender Nachrichten empfangen, wenn Sie sie benötigen. Das Zeitintervall wird durch *SuppressInterval* angegeben. *msgIds* ist eine durch Kommas getrennte Liste mit einer Auswahl der folgenden Nachrichten-IDs:

- 5211-Maximale Länge des Eigenschaftsnamens überschritten.
- 5973-Verteilte Publish/Subscribe-Subskription unterdrückt
- 5974-Verteilte Publish/Subscribe-Veröffentlichung unterdrückt
- 6254-Das System konnte die gemeinsam genutzte Bibliothek nicht dynamisch laden.
- 7234-Anzahl der geladenen Nachrichten
- 9001-Kanalprogramm normal beendet
- 9002-Kanalprogramm gestartet
- 9202-Ferner Host nicht verfügbar
- 9208-Fehler beim Empfangen vom Host
- 9209-Verbindung geschlossen
- 9228-Kanalantworter kann nicht gestartet werden
- 9489-Maximale Anzahl der SVRCONN-Instanzen überschritten
- 9490-SVRCONN max-Instanzen pro Clientgrenzwert überschritten
- 9508-Verbindung zum WS-Manager kann nicht hergestellt werden
- 9524-Ferner WS-Manager nicht verfügbar
- 9528-Benutzer hat das Schließen des Kanals angefordert
- 9558-Ferner Kanal ist nicht verfügbar
- 9637-Kanal fehlt ein Zertifikat

9776-Kanal wurde von Benutzer-ID blockiert  
9777-Kanal wurde durch NOACCESS-Zuordnung blockiert  
9782-Verbindung wurde durch Adresse blockiert  
9999-Kanalprogramm abnormal beendet

Wenn in SuppressMessage und ExcludeMessage dieselbe Nachrichten-ID angegeben ist, wird die Nachricht ausgeschlossen.

### **SuppressInterval=Länge**

Gibt das Zeitintervall (in Sekunden) an, in dem Nachrichten, die in SuppressMessage angegeben sind, nur einmal in das Fehlerprotokoll des Warteschlangenmanagers geschrieben werden. *length* muss im Bereich von 1 bis 86400 Sekunden liegen. Wenn SuppressInterval nicht angegeben ist, wird der Standardwert von 30 Sekunden verwendet.

## **Standardbindungsart des WS-Managers**

Verwenden Sie die Eigenschaftenseite des Extended -Warteschlangenmanagers im IBM WebSphere MQ Explorer oder die Zeilengruppe Connection in der Datei qm.ini, um den Standardbindungstyp anzugeben.

### **DefaultBindTyp =SHARED|ISOLATED**

Wenn DefaultBindType auf ISOLATED gesetzt ist, werden Anwendungen und der Warteschlangenmanager in separaten Prozessen ausgeführt, und es werden keine Ressourcen von ihnen gemeinsam genutzt.

Wenn DefaultBindType auf SHARED gesetzt ist, werden Anwendungen und der Warteschlangenmanager in separaten Prozessen ausgeführt, aber einige Ressourcen werden von ihnen gemeinsam genutzt.

Der Standardwert ist SHARED.

## **SSL- und TLS-Zeilengruppe der Konfigurationsdatei des Warteschlangenmanagers**

Verwenden Sie die SSL-Zeilengruppe der Warteschlangenmanager-Konfigurationsdatei, um SSL- oder TLS-Kanäle auf Ihrem Warteschlangenmanager zu konfigurieren.

### **OCSP (Online Certificate Status Protocol)**

Ein Zertifikat kann die Erweiterung "AuthorityInfoAccess" enthalten. Diese Erweiterung gibt einen Server an, der über das Online Certificate Status Protocol (OCSP) kontaktiert werden soll. Um SSL- oder TLS-Kanäle in Ihrem Warteschlangenmanager zu ermöglichen, die Erweiterungen von AuthorityInfoAccess zu verwenden, stellen Sie sicher, dass der in ihnen benannte OCSP-Server verfügbar ist, ordnungsgemäß konfiguriert ist und über das Netz zugänglich ist. Weitere Informationen hierzu finden Sie im Abschnitt [Mit widerrufenen Zertifikaten arbeiten](#).

### **CrlDistributionPoint (CDP)**

Ein Zertifikat kann eine CrlDistributionPoint-Erweiterung enthalten. Diese Erweiterung enthält eine URL, die sowohl das Protokoll, das zum Download einer Zertifikatswiderrufsliste (CRL) verwendet wird, als auch den zu kontaktierbaren Server enthält.

Wenn Sie SSL- oder TLS-Kanäle in Ihrem Warteschlangenmanager für die Verwendung von "CrlDistributionPoint"-Erweiterungen zulassen möchten, stellen Sie sicher, dass der in ihnen benannte CDP-Server verfügbar und korrekt konfiguriert ist und über das Netz zugänglich ist.

## **Die SSL-Zeilengruppe**

Verwenden Sie die SSL-Zeilengruppe in der Datei qm.ini, um zu konfigurieren, wie SSL- oder TLS-Kanäle in Ihrem Warteschlangenmanager versuchen, die folgenden Funktionen zu verwenden, und wie sie reagieren, wenn Probleme bei der Verwendung auftreten.

Wenn der angegebene Wert nicht einer der gültigen Werte ist, wird in jedem der folgenden Fälle der Standardwert übernommen. Es werden keine Fehlermeldungen geschrieben, die darauf hingewiesen werden, dass ein ungültiger Wert angegeben wurde.

### **CDPCheckExtensions=YES|NO**

CDPCheckExtensions gibt an, ob SSL- oder TLS-Kanäle auf diesem Warteschlangenmanager versuchen, CDP-Server zu überprüfen, die in den Zertifikatserweiterungen von CrlDistributionPoint aufgeführt sind.

- YES: SSL- oder TLS-Kanäle versuchen, die CDP-Server zu überprüfen, um festzustellen, ob ein digitales Zertifikat widerrufen wird.
- NO: SSL- oder TLS-Kanäle versuchen nicht, CDP-Server zu überprüfen. Dieser Wert stellt den Standardwert dar.

### **OCSPAAuthentication=REQUIRED|WARN|OPTIONAL**

OCSPAAuthentication gibt die Aktion an, die ausgeführt werden soll, wenn ein Widerrufsstatus nicht von einem OCSP-Server bestimmt werden kann.

Wenn die OCSP-Prüfung aktiviert ist, versucht ein SSL- oder TLS-Kanalprogramm, einen OCSP-Server zu kontaktieren.

Wenn das Kanalprogramm keine OCSP-Server kontaktieren kann oder wenn kein Server den Widerrufsstatus des Zertifikats bereitstellen kann, wird der Wert des Parameters OCSPAAuthentication verwendet.

- **ERFORDERLICH** : Wenn der Widerrufsstatus nicht bestimmt wird, wird die Verbindung mit einem Fehler geschlossen. Dieser Wert stellt den Standardwert dar.
- **WARNUNG** : Wenn der Widerrufsstatus nicht bestimmt wird, wird eine Warnung in das Fehlerprotokoll des Warteschlangenmanagers geschrieben, aber die Verbindung kann fortgesetzt werden.
- **OPTIONAL** : Wenn der Widerrufsstatus nicht festgestellt werden kann, kann die Verbindung unbeaufsichtigt fortgesetzt werden. Es werden keine Warnungen oder Fehler ausgegeben.

### **OCSPCheckExtensions=YES|NO**

OCSPCheckExtensions gibt an, ob SSL- und TLS-Kanäle auf diesem Warteschlangenmanager versuchen, die OCSP-Server zu überprüfen, die in den Zertifikatserweiterungen von AuthorityInfoAccess aufgeführt sind.

- YES: SSL- und TLS-Kanäle versuchen, die OCSP-Server zu überprüfen, um festzustellen, ob ein digitales Zertifikat widerrufen wird. Dieser Wert stellt den Standardwert dar.
- NO: SSL- oder TLS-Kanäle versuchen nicht, die OCSP-Server zu überprüfen.

### **SSLHTTPProxyName=Zeichenfolge**

Die Zeichenfolge ist entweder der Hostname oder die Netzadresse des HTTP-Proxy-Servers, der von GSKit für OCSP-Prüfungen verwendet werden soll. Auf die Adresse kann optional eine in Klammern gesetzte Portnummer folgen. Wenn Sie die Portnummer nicht angeben, wird der Standard-HTTP-Port 80 verwendet. Auf den Plattformen HP-UX PA-RISC und Sun Solaris SPARC und für 32-Bit-Clients unter AIX kann die Netzadresse nur eine IPv4 -Adresse sein; auf anderen Plattformen kann sie eine IPv4 -oder IPv6 -Adresse sein.

Dieses Attribut kann erforderlich sein, wenn z. B. eine Firewall den Zugriff auf die URL des OCSP-Responder verhindert.

## **Exiteigenschaften**

Verwenden Sie die Eigenschaftenseite "Clusterwarteschlangenmanager" aus der Zeilengruppe IBM WebSphere MQ Explorer oder die Aktion "ExitPropertiesLocal" in der Datei "qm.ini", um Informationen zu Exiteigenschaften in einem Warteschlangenmanager anzugeben. Alternativ können Sie ihn mit dem Befehl **amqmdain** festlegen.

Diese Einstellung wird standardmäßig aus dem Attribut CLWLMode in der Zeilengruppe 'ExitProperties' der maschinenweiten Konfiguration übernommen (siehe „Exiteigenschaften“ auf Seite 462 ). Ändern Sie diese Einstellung nur, wenn Sie diesen Warteschlangenmanager auf eine andere Weise konfigurieren möchten. Dieser Wert kann für einzelne Warteschlangenmanager mit dem Attribut "Clusterworkloadmodus" auf der Eigenschaftenseite "Clusterwarteschlangenmanager" überschrieben werden.

#### **CLWLMode=SAFE|FAST**

Mit dem Exit für Clusterauslastung (CLWL) können Sie angeben, welche Clusterwarteschlange im Cluster als Antwort auf einen MQI-Aufruf geöffnet werden soll (z. B. MQOPEN, MQPUT). Der CLWL-Exit wird abhängig von dem Wert, den Sie im Attribut CLWLMode angeben, entweder im Modus FAST oder im Modus SAFE ausgeführt. Wenn Sie das Attribut CLWLMode nicht angeben, wird der Exit für Clusterauslastung im SAFE-Modus ausgeführt.

#### **SAFE**

Führen Sie den CLWL-Exit in einem separaten Prozess aus dem Warteschlangenmanager aus. Dies ist die Standardeinstellung.

Tritt bei der Ausführung im SAFE-Modus ein Problem mit dem vom Benutzer geschriebenen CLWL-Exit auf, geschieht Folgendes:

- Der CLWL-Serverprozess (amqzlw0) schlägt fehl.
- Der WS-Manager startet den CLWL-Serverprozess erneut.
- Der Fehler wird Ihnen im Fehlerprotokoll gemeldet. Wenn ein MQI-Aufruf in Bearbeitung ist, erhalten Sie eine Benachrichtigung in Form eines Rückkehrcodes.

Die Integrität des Warteschlangenmanagers bleibt erhalten.

**Anmerkung:** Die Ausführung des CLWL-Exits in einem separaten Prozess kann sich auf die Leistung auswirken.

#### **FAST**

Führen Sie den Cluster-Exit inline im WS-Manager-Prozess aus.

Wenn Sie diese Option angeben, wird die Leistung verbessert, da die Prozesse, die mit der Ausführung im SAFE-Modus verbunden sind, vermieden werden, dies jedoch zu Lasten der Integrität des Warteschlangenmanagers geht. Sie sollten den CLWL-Exit nur im FAST-Modus ausführen, wenn Sie davon überzeugt sind, dass es **keine** Probleme mit dem CLWL-Exit gibt, und Sie sind besonders besorgt über die Leistung.

Tritt ein Problem auf, wenn der CLWL-Exit im FAST-Modus ausgeführt wird, schlägt der Warteschlangenmanager fehl, und Sie laufen Gefahr, dass die Integrität des Warteschlangenmanagers beeinträchtigt wird, der beeinträchtigt wird.

## **Subpool**

Diese Zeilengruppe wird von WebSphere MQ erstellt. Ändern Sie sie nicht.

Der Zeilengruppensubpool und das Attribut ShortSubpoolName innerhalb dieser Zeilengruppe werden automatisch von WebSphere MQ geschrieben, wenn Sie einen Warteschlangenmanager erstellen. WebSphere MQ wählt einen Wert für ShortSubpoolaus. Ändern Sie diesen Wert nicht.

Der Name entspricht einem Verzeichnis und einer symbolischen Verbindung, die im Verzeichnis /var/mqm/sockets erstellt wurden und die WebSphere MQ für die interne Kommunikation zwischen den aktiven Prozessen verwendet.

## **HP Integrity NonStop Server konfigurieren**

Verwenden Sie diese Informationen als Hilfe bei der Konfiguration Ihrer Installation des IBM WebSphere MQ -Clients für HP Integrity NonStop Server .

Weitere Informationen zur Konfiguration eines Clients mithilfe einer Konfigurationsdatei finden Sie im Abschnitt „Client mit einer Konfigurationsdatei konfigurieren“ auf Seite 136.

Weitere Informationen zur Konfiguration eines Clients mithilfe von Umgebungsvariablen finden Sie im Abschnitt [„WebSphere MQ -Umgebungsvariablen verwenden“](#) auf Seite 156.

Wenn Sie IBM WebSphere MQ -Client für HP Integrity NonStop Server -Operationen unter TMF/Gateway ausführen, finden Sie in den Unterabschnitten Informationen zur Konfiguration des TMF/Gateways. Es wird eine Übersicht über den Gateway-Prozess, die Konfiguration des Gateways für die Ausführung unter Pathway und die Konfiguration der Clientinitialisierungsdatei bereitgestellt, damit Ihr IBM WebSphere MQ -Client für HP Integrity NonStop Server das TMF-Gateway erreichen kann.

Dieser Abschnitt enthält außerdem IBM WebSphere MQ Client for HP Integrity NonStop Server -spezifische Informationen zum Erteilen von Berechtigungen für Kanäle.

## Übersicht über den Gateway-

Die HP NonStop Transaction Management Facility (TMF) stellt Services bereit, die es einem Gateway-Prozess ermöglichen, sich als Ressourcenmanager zu registrieren. Der von IBM WebSphere MQ bereitgestellte TMF/Gateway-Prozess wird unter Pathway ausgeführt.

IBM WebSphere MQ registriert einen einzelnen Gateway-Prozess für jeden Warteschlangenmanager, der von TMF koordiniert wird. Daher müssen Sie für jeden Warteschlangenmanager, der an TMF-koordinierten Arbeitseinheiten teilnehmen soll, ein separates TMF/Gateway konfigurieren. Diese Registrierung ist so, dass jeder WS-Manager ein unabhängiger Ressourcenmanager ist, und zu Verwaltungszwecken, die Registrierung jedes WS-Managers einmal mit HP NonStop TMF zu einer leicht verständlichen Zuordnung.

Bei mehreren Installationen von IBM WebSphere MQ müssen Sie einen einzelnen Gateway-Prozess aus einer dieser Installationen für jeden Warteschlangenmanager benennen, der von TMF koordiniert werden soll.

Die Schnittstelle zum Gateway-Prozess unterstützt alle Clients mit derselben Version oder früher.

Weitere Informationen zur Verwaltung des Gateway-Prozesses finden Sie im Abschnitt [HP Integrity Non-Stop Server verwalten](#).

## Gateway für die Ausführung unter Pathway konfigurieren

TMF/Gateway ist die Schnittstelle zwischen der HP NonStop Transaction Management Facility (TMF) und IBM WebSphere MQ, die es TMF ermöglicht, als Transaktionskoordinator für IBM WebSphere MQ-Transaktionen zu wirken.

Das durch IBM WebSphere MQ bereitgestellte TMF-Gateway konvertiert TMF-koordinierte Transaktionen in XA-koordinierte Transaktionen (eXtended Architecture), um mit dem fernen Warteschlangenmanager zu kommunizieren.

Sie müssen über einen TMF/Gateway pro Warteschlangenmanager verfügen, der koordiniert werden muss, und die Clientkonfiguration ist erforderlich, damit der Client eine Verbindung zum richtigen Gateway herstellen kann.

Das TMF/Gateway kann alle Mechanismen verwenden, die dem Client für die Kommunikation mit einem Warteschlangenmanager zur Verfügung stehen. Konfigurieren Sie das TMF/Gateway so, wie Sie es für Ihre anderen Anwendungen tun würden.

Das TMF/Gateway ist kein HP Integrity NonStop Server-Prozesspaar und ist für die Ausführung in einer Pathway-Umgebung konzipiert. Der TMF/Gateway erstellt permanente Ressourcen innerhalb von TMF, die er bei nachfolgenden Ausführungen wiederverwendet, daher muss das TMF/Gateway immer unter derselben Benutzerberechtigung ausgeführt werden.

### Server-Klasse definieren

TMF/Gateway wird als Serverklasse in einer Pathway-Umgebung gehostet. Um die Serverklasse zu definieren, müssen Sie die folgenden Serverattribute festlegen:

#### **PROCESSTYPE=OSS**

Gibt den Typ der Server in der Serverklasse an. Bei dem Gateway-Prozess handelt es sich um ein Multithread-OSS-Programm. Dieses Attribut ist obligatorisch und muss auf OSS gesetzt werden.

**MAXSERVERS=1**

Gibt die maximale Anzahl der Serverprozesse in dieser Serverklasse an, die gleichzeitig ausgeführt werden können. Es kann nur ein einziger Gateway-Prozess für jeden WS-Manager geben. Dieses Attribut ist obligatorisch und muss auf 1 gesetzt werden.

**NUMSTATIC=1**

Gibt die maximale Anzahl der statischen Server in dieser Serverklasse an. Der Gateway-Prozess muss als statischer Server ausgeführt werden. Dieses Attribut ist obligatorisch und muss auf 1 gesetzt werden.

**TMF=ON**

Gibt an, ob Server in dieser Serverklasse Datendateien sperren und aktualisieren können, die vom TMF-Subsystem überwacht werden. Der Gateway-Prozess wirkt in den TMF-Transaktionen von IBM WebSphere MQ-Clientanwendungen mit; daher muss dieses Attribut auf ON gesetzt sein.

**PROGRAM=<OSS installation path>/opt/mqm/bin/runmqtmf**

Für IBM WebSphere MQ -Client für IBM WebSphere MQ muss dieses Attribut runmqtmf sein. Dieses Attribut muss der absolute OSS-Pfadname sein. Der Fall ist bedeutsam.

**ARGLIST=-m < Warteschlangenmanagername>[-c < Kanalname>] [-p < Port>] [-h < Hostname>] [-n < max threads>]**

Diese Attribute stellen Parameter für den Gateway-Prozess bereit, wobei Folgendes gilt:

- `QMGRName` ist der Name des Warteschlangenmanagers für diesen Gateway-Prozess. Wenn Sie eine Gruppe mit gemeinsamer Warteschlange (oder eine andere Portverteilungstechnologie) verwenden, muss dieser Parameter auf einen bestimmten Warteschlangenmanager ausgerichtet sein. Dieser Parameter ist obligatorisch.
- `channel name` ist der Name des Serverkanals auf dem Warteschlangenmanager, der vom Gateway-Prozess verwendet werden soll. Dieser Parameter ist optional.
- `port` ist der TCP/IP-Port des Warteschlangenmanagers. Dieser Parameter ist optional.
- `host name` steht für den Hostnamen des Warteschlangenmanagers. Dieser Parameter ist optional.
- `max threads` ist die maximale Anzahl der Worker-Threads, die vom Gateway-Prozess erstellt werden. Dieser Parameter kann ein Wert von 10 oder größer sein. Der niedrigste Wert, der verwendet wird, ist 10, auch wenn ein Wert kleiner als 10 angegeben ist. Wenn kein Wert angegeben wird, erstellt der Gateway-Prozess bis zu maximal 50 Threads.

Verwenden Sie die Attribute `-c`, `-p` und `-h` als alternative Methode zur Bereitstellung von Verbindungsinformationen für das Gateway, zusätzlich zu den in „[TMF/Gateway mit Umgebungsvariablen konfigurieren](#)“ auf Seite 488 beschriebenen. Wenn Sie einen oder mehrere, aber nicht alle Attribute `-c`, `-p` und `-h` angeben, geben die Attribute, die Sie nicht standardmäßig angeben, die folgenden Werte an:

- `channel name` nimmt standardmäßig den Wert `SYSTEM.DEF.SVRCONN` an
- `host name` nimmt standardmäßig den Wert `localhost` an
- `port` nimmt standardmäßig den Wert `1414` an

Wenn einer der Parameter, die Sie angeben, ungültig ist, gibt der TMF/Gateway die Diagnosenachricht [AMQ5379](#) in das Fehlerprotokoll aus und wird beendet.

**OWNER=ID**

Die Benutzer-ID, unter der das Gateway ausgeführt wird und dem der Warteschlangenmanager die Berechtigung "connect" erteilen muss.

**SECURITY="value"**

Gibt die Benutzer in Bezug auf das Attribut `owner` an, die über eine IBM WebSphere MQ -Clientanwendung auf das Gateway zugreifen können.

`LINKDEPTH` und `MAXLINKS` müssen mit Werten konfiguriert werden, die für die erwartete Anzahl von IBM WebSphere MQ-Clientanwendungen, die möglicherweise gleichzeitig mit dem Gateway kommunizieren möchten, geeignet sind. Wenn diese Werte zu niedrig sind, wird die Fehlernachricht [AMQ5399](#), die von Clientanwendungen ausgegeben wird, möglicherweise angezeigt.



Weitere Informationen zu diesen Serverattributen finden Sie im Handbuch *HP NonStop TS/MP 2.5 System Management Manual*.

## TMF/Gateway mit Umgebungsvariablen konfigurieren

Eine der am häufigsten verwendeten Methoden zum Definieren des TMF/Gateways ist die Festlegung der Umgebungsvariablen MQSERVER, z. B.:

```
ENV MQSERVER=<channel name>/<transport>/<host name>(<listener port>)
```

ENV am Anfang des Befehls ist Pathway Notation.

## Clientinitialisierungsdatei konfigurieren

Bei Verwendung von HP NonStop Transaction Management Facility (TMF) müssen Sie über eine IBM WebSphere MQ-Clientinitialisierungsdatei verfügen, damit Ihr IBM WebSphere MQ-Client für HP Integrity NonStop Server das TMF-Gateway erreichen kann.

Eine IBM WebSphere MQ-Clientinitialisierungsdatei für HP Integrity NonStop Server kann sich an verschiedenen Positionen befinden. Sie finden weitere Informationen hierzu unter [„Position der Clientkonfigurationsdatei“](#) auf Seite 138.

Ausführliche Informationen zum Inhalt der Konfigurationsdatei sowie ein Beispiel finden Sie in [„Client mit einer Konfigurationsdatei konfigurieren“](#) auf Seite 136. Verwenden Sie die TMF-Zeilengruppe, um den TMF-Warteschlangenmanager und die Serverdetails anzugeben. Weitere Informationen finden Sie in [„Zeilengruppen 'TMF' und 'TMF/Gateway'“](#) auf Seite 156.

Es folgt ein Beispiel für die Einträge für einen IBM WebSphere MQ -Client für HP Integrity NonStop Server :

```
TMF :
 PathMon=$PSD1P

TmfGateway :
 QManager=MQ5B
 Server=MQ-MQ5B

TmfGateway :
 QManager=MQ5C
 Server=MQ-MQ5C
```

Weitere Informationen zum Konfigurieren eines Clients unter Verwendung von Umgebungsvariablen finden Sie in [„WebSphere MQ -Umgebungsvariablen verwenden“](#) auf Seite 156.

## Berechtigungen für Kanäle erteilen

Das Erteilen von Berechtigungen an Kanäle auf einem IBM WebSphere MQ-Client für HP Integrity NonStop Server ist mit der Vorgehensweise bei anderen Betriebssystemen identisch; allerdings muss die Kennung des Eigners, unter dessen Namen das Gateway ausgeführt wird, bekannt sein.

Sie können dann die Identifikation des Eigners des Gateways verwenden, um die entsprechenden Berechtigungen zu erteilen. Der wichtige Unterschied besteht darin, dass die Erteilung von Berechtigungen für WS-Manager-Kanäle nicht unter der Berechtigung einer Anwendung liegt.

Verwenden Sie den Befehl **setmqaut**, um eine Berechtigung zu erteilen, d. h. einem IBM WebSphere MQ -Prinzipal oder einer Benutzergruppe die Berechtigung zum Ausführen einer Operation zu erteilen, und um eine Berechtigung zu widerrufen, d. h. die Berechtigung zum Ausführen einer Operation zu entfernen.



## Bemerkungen

---

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden.

Möglicherweise bietet IBM die in dieser Dokumentation 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 IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte 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 dieser Dokumentation ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

IBM Europe  
IBM Europe, Middle East and 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:

Lizenzierung von geistigem Eigentum

IBM Japan, Ltd.

**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 Angaben in dieser Veröffentlichung werden in regelmäßigen Zeitabständen aktualisiert. Die Änderungen werden in Überarbeitungen oder in Technical News Letters (TNLs) bekanntgegeben. IBM kann 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.

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 and 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.

Die 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. Um diese so realistisch wie möglich zu gestalten, enthalten sie auch Namen von Personen, Firmen, Marken und Produkten. Sämtliche dieser Namen sind fiktiv. Ähnlichkeiten mit Namen und Adressen tatsächlicher Unternehmen oder Personen sind zufällig.

#### COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Musterprogramme, die in Quellensprache geschrieben sind. Sie dürfen diese Musterprogramme kostenlos (d. h. ohne Zahlung an IBM) 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 Musterprogramme geschrieben werden. 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 WebSphere 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](http://www.ibm.com/legal/copytrade.shtml). Weitere Produkt- und Servicennamen können Marken von IBM oder anderen Unternehmen sein.

Microsoft und Windows sind 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 (<http://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: