

7.5

Planung für IBM WebSphere MQ

IBM

Hinweis

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen unter „Bemerkungen“ auf Seite 169 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

Planung.....	5
IBM MQ und IBM MQ Appliance -lokale Hinweise zur Umsetzung der DSGVO.....	5
IBM WebSphere MQ-Architektur entwerfen.....	15
Architekturen mit einem einzelnen Warteschlangenmanager.....	15
Architekturen mit einem einzelnen Warteschlangenmanager.....	16
Architekturen mit einem einzelnen Warteschlangenmanager.....	16
Architekturen mit einem einzelnen Warteschlangenmanager.....	16
Architekturen mit mehreren Warteschlangenmanagern.....	17
Punkt-zu-Punkt-Messaging.....	21
Einführung in Publish/Subscribe.....	21
Behandlung nicht zugestellter Nachrichten mit der Steuerroutine der Warteschlange für nicht zustellbare Nachrichten.....	118
Planung für mehrere Installationen.....	128
Primäre Installation auswählen.....	129
Speicher- und Leistungsanforderungen planen.....	134
Erforderer Plattenspeicherplatz.....	135
Dateisystemunterstützung planen.....	136
IBM WebSphere MQ -und UNIX System V IPC-Ressourcen.....	163
Gemeinsam genutzter Speicher unter AIX.....	163
Prioritätswerte für UNIX-Prozesse festlegen.....	164
IBM WebSphere MQ -Clientumgebung unter HP Integrity NonStop Server planen.....	164
HP Integrity NonStop Server -Umgebung vorbereiten.....	164
IBM WebSphere MQ und HP NonStop TMF.....	165
HP NonStop TMF verwenden.....	165
Bemerkungen.....	169
Informationen zu Programmierschnittstellen.....	170
Marken.....	171

Planung

Bei der Planung Ihrer IBM WebSphere MQ -Umgebung müssen Sie die zu konfigurierende IBM WebSphere MQ -Architektur, den Ressourcenbedarf, den Bedarf an Protokollierungs- und Sicherungsfunktionen berücksichtigen. Verwenden Sie die Links in diesem Abschnitt, um die Umgebung zu planen, in der IBM WebSphere MQ ausgeführt wird.

Bevor Sie Ihre IBM WebSphere MQ -Umgebung planen, machen Sie sich mit den IBM WebSphere MQ -Basiskonzepten vertraut. Lesen Sie die Themen unter [Technische Übersicht](#).

Zugehörige Konzepte

[Verfügbarkeit, Wiederherstellung und Neustart](#)

Zugehörige Tasks

[Migration](#)

[Installieren](#)

[Konfiguration](#)

[WebSphere MQ verwalten](#)

[Sicherstellen, dass Nachrichten nicht verloren gehen \(Protokollierung\)](#)

IBM MQ und IBM MQ Appliance -lokale Hinweise zur Umsetzung der DSGVO

Für folgende Produkt-IDs:

- 5724-H72 IBM MQ
- 5655-AV9 IBM MQ Advanced für z/OS
- 5655-AV1 IBM MQ Advanced für z/OS, Value Unit Edition
- 5655-AM9 IBM MQ Advanced Message Security für z/OS
- 5725-Z09 IBM MQ Appliance M2001
- 5725-S14 IBM MQ Appliance M2000
- 5655-MQ9 IBM MQ für z/OS
- 5655-VU9 IBM MQ für z/OS Value Unit Edition
- 5639-L92 IBM MQ Internet Pass-Thru
- 5655-MF9 IBM MQ Managed File Transfer für z/OS
- 5655-ADV IBM WebSphere MQ Advanced für z/OS
- 5655-AMS IBM WebSphere MQ Advanced Message Security für z/OS
- 5724-R10 IBM WebSphere MQ File Transfer Edition für Multiplatforms
- 5724-A39 IBM WebSphere MQ für HP NonStop Server
- 5724-A38 IBM WebSphere MQ für HP OpenVMS
- 5655-W97 IBM WebSphere MQ für z/OS
- 5655-VU8 IBM WebSphere MQ für z/OS Value Unit Edition
- 5655-VUE IBM WebSphere MQ für z/OS Value Unit Edition
- 5725-C79 IBM WebSphere MQ Hypervisor Edition für Red Hat Enterprise Linux® für x86
- 5725-F22 IBM WebSphere MQ Hypervisor für AIX
- 5655-MFT IBM WebSphere MQ Managed File Transfer für z/OS

Hinweis:

Dieses Dokument soll Ihnen bei den Vorbereitungen für die Umsetzung der EU-Datenschutz-Grundverordnung (DSGVO) helfen. Es enthält Informationen zu den Features von IBM MQ, die Sie konfigurieren können, sowie Aspekte zur Verwendung des Produkts, die Sie berücksichtigen sollten, um Ihre Organisation bei der Umsetzung der DSGVO zu unterstützen. Diese Informationen sind keine erschöpfende Liste, da die Kunden viele Möglichkeiten haben, Funktionen auszuwählen und zu konfigurieren, und die große Vielfalt an Möglichkeiten, die das Produkt in sich selbst und mit Anwendungen und Systemen anderer Hersteller verwendet werden kann.

Es liegt allein in der Verantwortung der Kunden, die Einhaltung der verschiedenen Gesetze und Verordnungen sicherzustellen, z. B. der DSGVO. Es obliegt allein den Kunden, sich von kompetenter juristischer Stelle zu Inhalt und Auslegung aller relevanten Gesetze und Vorschriften beraten zu lassen, die ihre Geschäftstätigkeit und die von ihnen eventuell einzuleitenden Maßnahmen zur Einhaltung dieser Gesetze und Vorschriften betreffen.

Die hier beschriebenen Produkte, Services und anderen Funktionen sind nicht für alle Kundensituationen geeignet und können eine eingeschränkte Verfügbarkeit haben. IBM erteilt keine Rechts- oder Steuerberatung und gibt keine Garantie bezüglich der Konformität von IBM Produkten oder Services mit den geltenden Gesetzen und Vorschriften.

Inhaltsverzeichnis

1. [DSGVO](#)
2. [DSGVO-bezogene Produktkonfiguration](#)
3. [Datenlebenszyklus](#)
4. [Datenerfassung](#)
5. [Datenspeicher](#)
6. [Datenzugriff](#)
7. [Datenverarbeitung](#)
8. [Datenlöschung](#)
9. [Datenüberwachung](#)
10. [Funktionalität zur Einschränkung der Nutzung von personenbezogenen Daten](#)
11. [Dateiverwaltung](#)

DSGVO

Die Datenschutz-Grundverordnung (DSGVO) wurde von der Europäischen Union ("EU") verabschiedet und gilt seit dem 25. Mai 2018.

Warum ist die DSGVO wichtig?

Mit der DSGVO wird ein strengerer datenschutzrechtlicher Rahmen für die Verarbeitung personenbezogener Daten geschaffen. Folgende Änderungen sieht die DSGVO vor:

- Neue und erweiterte Rechte für Einzelpersonen
- Eine erweiterte Definition von personenbezogenen Daten
- Neue Verpflichtungen für "Auftragsverarbeiter"
- Potenzielle finanzielle Sanktionen bei Nichtkonformität
- Zwingende Meldepflicht bei Datenschutzverletzungen

Weitere Informationen zur DSGVO finden Sie hier:

- [Informationsportal zur EU-Datenschutz-Grundverordnung](#)
- ibm.com/GDPR-Website

Produktkonfiguration -Überlegungen zur DSGVO-Bereitschaft

In den folgenden Abschnitten finden Sie Hinweise zur Konfiguration von IBM MQ , um Ihre Organisation bei der Umsetzung der DSGVO zu unterstützen.

Datenlebenszyklus

IBM MQ ist ein transaktionsorientiertes, nachrichtenorientiertes Middlewareprodukt, das Anwendungen den asynchronen Austausch von anwendungsbereitgestellten Daten ermöglicht. IBM MQ unterstützt eine Reihe von Messaging-APIs, -Protokollen und -Bridges für die Verbindung von Anwendungen. Daher kann IBM MQ verwendet werden, um viele Formen von Daten auszutauschen, von denen einige potenziell der DSGVO unterliegen könnten. Es gibt mehrere Produkte anderer Anbieter, mit denen IBM MQ Daten austauschen kann. Bei einigen dieser Produkte handelt es sich um IBM eigene Produkte, aber es gibt auch viele andere Produkte, die von anderen Technologieanbietern bereitgestellt werden. Auf der Website [Software Product Compatibility Reports](#) sind Listen der zugehörigen Software aufgeführt. Für Überlegungen zur DSGVO-Bereitschaft eines Fremdprodukts sollten Sie die Produktdokumentation zu Rate ziehen. IBM MQ -Administratoren steuern die Art und Weise, in der IBM MQ durch die Definition von Warteschlangen, Themen und Subskriptionen mit Daten interagiert, die sie durchlaufen.

Welche Typen von Daten fließen durch IBM MQ?

Da IBM MQ einen asynchronen Messaging-Service für Anwendungsdaten bereitstellt, gibt es keine verbindliche Antwort auf diese Frage, da die Anwendungsfälle je nach Anwendungsimplementierung variieren. Anwendungsnachrichtendaten werden in Warteschlangendateien (Seitengruppen oder Coupling-Facility unter z/OS), Protokollen und Archiven persistent gespeichert und die Nachricht kann selbst Daten enthalten, die der DSGVO unterliegen. Die von den Anwendungen bereitgestellten Nachrichtendaten können auch in Dateien enthalten sein, die zu Fehlerbestimmungszwecken erfasst werden, wie z. B. Fehlerprotokolle, Tracedateien und FFSTs. Unter z/OS können von der Anwendung bereitgestellte Nachrichtendaten auch in Adressräume oder Coupling-Facility-Speicherauszüge eingeschlossen werden.

Im Folgenden finden Sie einige typische Beispiele für personenbezogene Daten, die mithilfe von IBM MQ ausgetauscht werden:

- Mitarbeiter des Kunden (z. B. IBM MQ kann verwendet werden, um die Lohnbuchhaltungs- oder HR-Systeme des Kunden zu verbinden)
- Die personenbezogenen Daten des Kunden selbst (z. B. IBM MQ kann von einem Kunden verwendet werden, um Daten zwischen Anwendungen auszutauschen, die sich auf seine Kunden beziehen, wie z. B. das Aufnehmen von Sales-Leads und das Speichern von Daten in ihrem CRM-System.
- Die sensiblen personenbezogenen Daten des Kunden selbst (z. B. IBM MQ) können in Branchenkontexten verwendet werden, in denen der Austausch personenbezogener Daten erforderlich ist, z. B. HL7-based Gesundheitsdatensätze bei der Integration klinischer Anwendungen).

Neben den von der Anwendung bereitgestellten Nachrichtendaten verarbeitet IBM MQ die folgenden Datentypen:

- Authentifizierungsnachweise (z. B. Benutzername und Kennwort, API-Schlüssel usw.)
- Technisch identifizierbare personenbezogenen Daten (z. B. Einheiten-IDs, Nutzungskennungen, IP-Adresse usw. bei Verknüpfung mit einer Einzelperson)

Personenbezogene Daten, die für den Onlinekontakt mit IBM verwendet werden

IBM MQ Kunden können Onlinekommentare/Feedback/Anforderungen zur Kontaktaufnahme mit IBM zu IBM MQ -Themen auf verschiedene Arten einreichen, in erster Linie:

- Öffentlicher Kommentarbereich auf Seiten im [IBM MQ-Bereich auf IBM Developer](#)
- Bereich der öffentlichen Kommentare auf Seiten von [IBM MQ-Produktinformationen in IBM Documentation](#)
- Öffentliche Kommentare in den [IBM Unterstützungsforen](#)
- Öffentliche Kommentare in der [IBM RFE-Community von IBM Developer](#)

In der Regel werden nur der Name und die E-Mail-Adresse des Kunden verwendet, um persönliche Antworten für den Betreff des Kontakts zu ermöglichen, und die Verwendung personenbezogener Daten entspricht der [IBM Online-Datenschutzerklärung](#).

Datenerfassung

IBM MQ kann zum Erfassen personenbezogener Daten verwendet werden. Bei der Beurteilung Ihrer Verwendung von IBM MQ und Ihrer Anforderungen an die Einhaltung der DSGVO sollten Sie die Typen personenbezogener Daten berücksichtigen, die unter Ihren Umständen IBM durchlaufen. Sie können z. B. Aspekte wie die folgenden berücksichtigen:

- Wie kommen die Daten zu Ihren Warteschlangenmanagern? (Über welche Protokolle? Sind die Daten verschlüsselt? Sind die Daten signiert?)
- Wie werden Daten von Ihren Warteschlangenmanagern gesendet? (Über welche Protokolle? Sind die Daten verschlüsselt? Sind die Daten signiert?)
- Wie werden Daten beim Durchlaufen eines Warteschlangenmanagers gespeichert? (Jede Messaging-Anwendung hat das Potenzial, Nachrichtendaten in statusabhängige Medien zu schreiben, selbst wenn eine Nachricht nicht persistent ist. Sind Sie sich bewusst, wie die Messaging-Funktionen Aspekte der Anwendungsnachrichtendaten, die das Produkt passieren, zugänglich machen könnten?)
- Wie werden Berechtigungsnachweise erfasst und gespeichert, die von IBM MQ für den Zugriff auf Anwendungen anderer Anbieter benötigt werden?

IBM MQ muss möglicherweise mit anderen Systemen und Services kommunizieren, für die eine Authentifizierung erforderlich ist, z. B. LDAP. Bei Bedarf werden Authentifizierungsdaten (Benutzer-IDs, Kennwörter) von IBM MQ für die Verwendung in einer solchen Kommunikation konfiguriert und gespeichert. Soweit möglich, sollten Sie die Verwendung persönlicher Berechtigungsnachweise für die IBM MQ -Authentifizierung vermeiden. Berücksichtigen Sie den Schutz des Speichers, der für Authentifizierungsdaten verwendet wird. (Siehe Datenspeicherung weiter unten.)

Datenspeicherung

Wenn Nachrichtendaten über Warteschlangenmanager übertragen werden, speichert IBM MQ diese Daten (möglicherweise mehrere Kopien davon) direkt auf statusabhängigen Datenträgern. IBM MQ -Benutzer sollten den Schutz von Nachrichtendaten in Erwägung ziehen, während sie sich im Ruhezustand befinden.

In den folgenden Punkten werden Bereiche hervorgehoben, in denen IBM MQ von der Anwendung bereitgestellte Daten persistent speichert, die Benutzer bei der Einhaltung der DSGVO berücksichtigen sollten.

- Anwendungsnachrichtenwarteschlangen:

IBM MQ stellt Nachrichtenwarteschlangen für den asynchronen Datenaustausch zwischen Anwendungen bereit. Nicht persistente und persistente Nachrichten, die in einer Warteschlange gespeichert sind, werden in statusabhängige Datenträger geschrieben.

- Dateiübertragungsagentenwarteschlangen:

IBM MQ Managed File Transfer verwendet Nachrichtenwarteschlangen, um die zuverlässige Übertragung von Dateidaten zu koordinieren. Dateien, die personenbezogene Daten enthalten, und Datensätze von Übertragungen werden in diesen Warteschlangen gespeichert.

- Übertragungswarteschlangen:

Damit Nachrichten zuverlässig zwischen Warteschlangenmanagern übertragen werden können, werden Nachrichten temporär in Übertragungswarteschlangen gespeichert.

- Warteschlangen für nicht zustellbare Nachrichten:

Wenn Nachrichten in bestimmten Situationen nicht in die Zielwarteschlange eingereicht werden können, werden sie stattdessen in einer Warteschlange für nicht zustellbare Nachrichten gespeichert, sofern eine für den Warteschlangenmanager konfiguriert ist.

- Rücksetzwarteschlangen:

JMS- und XMS-Messaging-Schnittstellen stellen eine Funktion bereit, die es ermöglicht, dass vergiftete Nachrichten nach einer Reihe von Backouts in eine Backout-Warteschlange verschoben werden können, damit andere gültige Nachrichten verarbeitet werden können.

- AMS-Fehlerwarteschlange:

IBM MQ Advanced Message Security verschiebt Nachrichten, die einer Sicherheitsrichtlinie nicht entsprechen, in das SYSTEM.PROTECTION.ERROR.QUEUE Fehlerwarteschlange auf ähnliche Weise wie die Warteschlange für nicht zustellbare Nachrichten.

- Ständige Veröffentlichungen:

IBM MQ stellt eine Funktion für ständige Veröffentlichungen bereit, mit der subscribierende Anwendungen eine frühere Veröffentlichung zurückerufen können.

Weitere Informationen finden Sie hier:

- [Protokollierung: Stellen Sie sicher, dass die Nachrichten nicht verloren gehen.](#)
- [MFT Agent-Warteschlangeneinstellungen](#)
- [Übertragungswarteschlange definieren](#)
- [Verwenden der Warteschlange für dead-Mail](#)
- [Behandlung von Giftnachrichten in IBM MQ-Klassen für JMS](#)
- [AMS-Fehlerbehandlung](#)
- [Zurückgehaltene Veröffentlichungen](#)

In den folgenden Punkten werden Bereiche hervorgehoben, in denen IBM MQ indirekt von Anwendungen bereitgestellte Daten persistent speichern kann, die Benutzer auch bei der Einhaltung der DSGVO berücksichtigen sollten.

- Trace-Route-Messaging:

IBM MQ stellt Trace-Routenfunktionen bereit, die die Route aufzeichnen, die eine Nachricht zwischen Anwendungen nimmt. Die generierten Ereignisnachrichten können technisch identifizierbare personenbezogene Daten, wie z. B. IP-Adressen, enthalten.

- Anwendungsaktivitätstrace:

IBM MQ stellt einen Trace für Anwendungsaktivitäten bereit, der die Messaging-API-Aktivitäten von Anwendungen und Kanälen aufzeichnet. Der Trace für Anwendungsaktivitäten kann den Inhalt der von der Anwendung bereitgestellten Nachrichtendaten in Ereignisnachrichten aufzeichnen.

- Service-Trace:

IBM MQ stellt Service-Trace-Funktionen bereit, die die internen Codepfade aufzeichnen, über die Nachrichtendaten fließen. Im Rahmen dieser Funktionen kann IBM MQ den Inhalt der von der Anwendung bereitgestellten Nachrichtendaten in Tracedateien aufzeichnen, die auf Platte gespeichert sind.

- Warteschlangenmanagerereignisse:

IBM MQ kann Ereignisnachrichten generieren, die personenbezogene Daten wie Berechtigungs-, Befehls- und Konfigurationsereignisse enthalten können.

Weitere Informationen finden Sie hier:

- [Trace-Route-Messaging](#)
- [Trace verwenden](#)
- [Ereignisüberwachung](#)
- [WS-Manager-Ereignisse](#)

Wenn Sie den Zugriff auf Kopien der von den Anwendungen bereitgestellten Nachrichtendaten schützen möchten, können Sie die folgende Aktionen in Betracht ziehen:

- Schränken Sie den privilegierten Benutzerzugriff auf IBM MQ -Daten im Dateisystem ein, z. B. die Einschränkung der Benutzerzugehörigkeit zur Gruppe 'mqm' auf UNIX -Plattformen.

- Beschränken Sie den Anwendungszugriff auf IBM MQ -Daten über dedizierte Warteschlangen und Zugriffssteuerung. Vermeiden Sie, wenn möglich, eine unnötige gemeinsame Nutzung von Ressourcen, wie z. B. Warteschlangen, durch mehrere Anwendungen und sorgen Sie für eine differenzierte Zugriffssteuerung für Warteschlangen- und Themenressourcen.
- Verwenden Sie IBM MQ Advanced Message Security , um die End-to-End-Signatur und/oder Verschlüsselung von Nachrichtendaten bereitzustellen.
- Verwenden Sie die Verschlüsselung auf Datei- oder Datenträgerebene, um den Inhalt des Verzeichnisses zu schützen, in dem die Traceprotokolle gespeichert werden.
- Nach dem Hochladen des Service-Trace auf IBM können Sie Ihre Service-Trace-Dateien und FFST-Daten löschen, wenn Sie Bedenken hinsichtlich des Inhalts haben, der möglicherweise personenbezogene Daten enthält.

Weitere Informationen finden Sie hier:

- [Privilegierte Benutzer](#)
- [Unterstützung von Dateisystemen auf Multiplatforms planen](#)

Ein IBM MQ -Administrator kann einen Warteschlangenmanager mit Berechtigungsnachweisen (Benutzername und Kennwort, API-Schlüssel usw.) konfigurieren. für 3rd wie LDAP, IBM Cloud Product Insights, Salesforce usw. Diese Daten werden in der Regel im Datenverzeichnis des Warteschlangenmanagers gespeichert, das durch Dateisystemberechtigungen geschützt ist.

Bei der Erstellung eines IBM MQ -Warteschlangenmanagers wird das Datenverzeichnis mit einer gruppenbasierten Zugriffssteuerung eingerichtet, sodass IBM MQ die Konfigurationsdateien lesen und die Berechtigungsnachweise für die Verbindung zu diesen Systemen verwenden kann. IBM MQ -Administratoren werden als privilegierte Benutzer betrachtet und sind Mitglieder dieser Gruppe, sodass sie Lesezugriff auf die Dateien haben. Einige Dateien sind verschleiert, aber sie sind nicht verschlüsselt. Aus diesem Grund sollten Sie die folgenden Aktionen in Betracht ziehen, um den Zugriff auf Berechtigungsnachweise vollständig zu schützen:

- Schränken Sie den privilegierten Benutzerzugriff auf IBM MQ -Daten ein, z. B. die Einschränkung der Zugehörigkeit zur Gruppe 'mqm' auf UNIX -Plattformen.
- Verwenden Sie die Verschlüsselung auf Datei- oder Datenträgerebene, um den Inhalt des Datenverzeichnisses des Warteschlangenmanagers zu schützen.
- Verschlüsseln Sie die Backups des Produktionskonfigurationsverzeichnisses und speichern Sie sie mit entsprechenden Zugriffssteuerungen.
- Sie können Prüfprotokolle für Authentifizierungsfehler, Zugriffssteuerung und Konfigurationsänderungen mit Sicherheits-, Befehls- und Konfigurationsereignissen bereitstellen.

Weitere Informationen finden Sie hier:

- [IBM MQ schützen](#)

Datenzugriff

Auf IBM MQ -Warteschlangenmanagerdaten kann über die folgenden Produktschnittstellen zugegriffen werden, von denen einige für den Zugriff über eine ferne Verbindung und andere für den Zugriff über eine lokale Verbindung konzipiert sind.

- IBM MQ -Konsole [nur fern]
- IBM MQ REST-API [nur fern]
- MQI [Lokal und Fern]
- JMS [Lokal und Fern]
- XMS [Lokal und Fern]
- IBM MQ Telemetry (MQTT) [nur fern]
- IBM MQ Light (AMQP) [Nur fern]
- IBM MQ IMS -Bridge [Nur lokal]

- IBM MQ CICS -Bridge [Nur lokal]
- IBM MQ Bridge for HTTP [Nur fern]
- IBM MQ MFT-Protokollbrücken [nur fern]
- IBM MQ Connect:Direct -Bridges [nur fern]
- IBM MQ Bridge to Salesforce [nur fern]
- IBM MQ Bridge to Blockchain [nur fern]
- IBM MQ MQAI [lokal und fern]
- IBM MQ PCF-Befehle [lokal und fern]
- IBM MQ MQSC-Befehle [lokal und fern]
- IBM MQ Explorer [lokal und fern]

Die Schnittstellen sind so konzipiert, dass Benutzer Änderungen an einem IBM MQ -Warteschlangenmanager und darin gespeicherten Nachrichten vornehmen können. Die Verwaltungs- und Messaging-Operationen sind so gesichert, dass es drei Stufen gibt, wenn eine Anforderung gestellt wird.

- Authentifizierung
- Rollenzuordnung
- Autorisierung

Authentifizierung:

Wenn die Nachricht oder die Verwaltungsoperation von einer lokalen Verbindung angefordert wurde, ist die Quelle dieser Verbindung ein laufender Prozess auf demselben System. Der Benutzer, der den Prozess ausführt, muss alle vom Betriebssystem zur Verfügung gestellten Authentifizierungsschritte durchlaufen haben. Der Benutzername des Eigners des Prozesses, von dem aus die Verbindung hergestellt wurde, wird als Identität bestätigt. Dies könnte z. B. der Name des Benutzers sein, der die Shell ausführt, von der eine Anwendung gestartet wurde. Die möglichen Formen der Authentifizierung für lokale Verbindungen sind:

1. Bestätigter Benutzername (lokales Betriebssystem)
2. Optionaler Benutzername und Kennwort (OS, LDAP oder benutzerdefinierte Repositories von Drittanbietern)

Wenn die Verwaltungsaktion von einer fernen Verbindung angefordert wurde, erfolgt die Kommunikation mit IBM MQ über eine Netzchnittstelle. Die folgenden Formen der Identität können für die Authentifizierung über Netzwerkverbindungen dargestellt werden:

1. Bestätigter Benutzername (vom fernen Betriebssystem)
2. Benutzername und Kennwort (Betriebssystem, LDAP oder benutzerdefinierte Repositories von Drittanbietern)
3. Quellennetzadresse (z. B. IP-Adresse)
4. Digitales X.509-Zertifikat (gegenseitige SSL/TLS-Authentifizierung)
5. Sicherheitstokens (z. B. LTPA2-Token)
6. Andere benutzerdefinierte Sicherheit (Funktionalität, die von Drittanbieterexits bereitgestellt wird)

Rollenzuordnung:

In der Rollenzuordnungsstufe können die Berechtigungsnachweise, die in der Authentifizierungsstufe bereitgestellt wurden, einer alternativen Benutzer-ID zugeordnet werden. Wenn die zugeordnete Benutzer-ID fortgesetzt werden darf (z. B. wenn Benutzer mit Verwaltungsaufgaben durch Kanalauthentifizierungsregeln blockiert werden), wird die zugeordnete Benutzer-ID in die letzte Phase weitergeleitet, wenn Aktivitäten für IBM MQ -Ressourcen autorisiert werden.

Autorisierung:

IBM MQ bietet verschiedenen Benutzern die Möglichkeit, unterschiedliche Berechtigungen für verschiedene Messaging-Ressourcen, wie z. B. Warteschlangen, Themen und andere Warteschlangenmanagerobjekte, zu haben.

Protokollierungsaktivität:

Einige Benutzer von IBM MQ müssen möglicherweise einen Prüfdatensatz für den Zugriff auf MQ -Ressourcen erstellen. Beispiele für wünschenswerte Prüfprotokolle können Konfigurationsänderungen enthalten, die Informationen über die Änderung enthalten, die zusätzlich zu den angeforderten Änderungen enthalten sind.

Für die Implementierung dieser Anforderung stehen die folgenden Informationsquellen zur Verfügung:

1. Ein Warteschlangenmanager IBM MQ kann so konfiguriert werden, dass er Befehlsereignisse generiert, wenn ein Verwaltungsbefehl erfolgreich ausgeführt wurde.
2. Ein IBM MQ
3. Ein Warteschlangenmanager IBM MQ kann so konfiguriert werden, dass er ein Berechtigungsereignis erzeugt, wenn eine Berechtigungsprüfung für eine Ressource fehlschlägt.
4. Fehlermeldungen, die auf fehlgeschlagene Berechtigungsprüfungen hinweisen, werden in die Fehlerprotokolle des Warteschlangenmanagers geschrieben.
5. Die IBM MQ -Webkonsole schreibt Prüfnachrichten in ihre Protokolle, wenn die Authentifizierung, Berechtigungsprüfungen fehlschlagen oder wenn Warteschlangenmanager erstellt, gestartet, gestoppt oder gelöscht werden.

Wenn Sie diese Art von Lösungen in Betracht ziehen, sollten Benutzer von IBM MQ die folgenden Punkte berücksichtigen:

- Ereignisnachrichten sind nicht persistent, so dass beim erneuten Starten eines Warteschlangenmanagers die Informationen verloren gehen. Alle Ereignismonitore sollten so konfiguriert werden, dass sie ständig alle verfügbaren Nachrichten konsumieren und den Inhalt auf persistente Datenträger übertragen.
- IBM MQ privilegierte Benutzer verfügen über ausreichende Berechtigungen für inaktivierte Ereignisse, zum Löschen von Protokollen oder zum Löschen von Warteschlangenmanagern.

Weitere Informationen zum Sichern des Zugriffs auf IBM MQ -Daten und zum Bereitstellen eines Prüfprotokolls finden Sie in den folgenden Abschnitten:

- [IBM MQ-Sicherheitsmechanismen](#)
- [Konfigurationsereignisse](#)
- [Befehlsereignisse](#)
- [Fehlerprotokolle](#)

Datenverarbeitung

Verschlüsselung mit einer PKI-Infrastruktur (Public Key Infrastructure):

Sie können Netzverbindungen zu IBM MQ sichern, um TLS zu verwenden, das auch die gegenseitige Authentifizierung der einleitenden Seite der Verbindung bereitstellen kann.

Die Verwendung der PKI-Sicherheitsfunktionen, die von Transportmechanismen bereitgestellt werden, ist der erste Schritt zur Sicherung der Datenverarbeitung mit IBM MQ. Ohne weitere Sicherheitsfunktionen zu aktivieren, besteht das Verhalten einer konsumierenden Anwendung jedoch darin, alle Nachrichten zu verarbeiten, die an sie übermittelt wurden, ohne zu überprüfen, wo der Ursprung der Nachricht ist oder ob die Nachricht während der Übertragung geändert wurde.

Benutzer von IBM MQ , die für die Verwendung der Funktionalität von Advanced Message Security (AMS) lizenziert sind, können über die Definition und Konfiguration von Sicherheitsrichtlinien steuern, wie Anwendungen personenbezogene Daten in Nachrichten verarbeiten. Sicherheitsrichtlinien ermöglichen es, dass digitale Signatur und/oder Verschlüsselung auf Nachrichtendaten zwischen Anwendungen angewendet werden können.

Es ist möglich, Sicherheitsrichtlinien zu verwenden, um eine digitale Signatur zu fordern und zu validieren, wenn Nachrichten konsumiert werden, um sicherzustellen, dass Nachrichten authentisch sind. Die AMS-Verschlüsselung stellt eine Methode zur Verfügung, mit der Nachrichtendaten von einem lesbaren Formular in eine verschlüsselte Version konvertiert werden, die nur von einer anderen Anwendung decodiert werden kann, wenn es sich um den beabsichtigten Empfänger oder die Nachricht handelt und Zugriff auf den richtigen Entschlüsselungsschlüssel hat.

Weitere Informationen zur Verwendung von SSL und Zertifikaten zum Sichern Ihrer Netzverbindungen finden Sie in den folgenden Abschnitten in der IBM MQ V9 Produktdokumentation:

- [TLS-Sicherheit für IBM MQ konfigurieren](#)
- [AMS-Übersicht](#)

Datenlöschung

IBM MQ stellt Befehle und Benutzerschnittstellenaktionen zum Löschen von Daten bereit, die dem Produkt bereitgestellt wurden. Auf diese Weise können Benutzer von IBM MQ mit Funktionen Daten löschen, die sich auf bestimmte Einzelpersonen beziehen, falls dies erforderlich sein sollte.

- Bereiche des IBM MQ -Verhaltens, die für die Einhaltung der DSGVO beim Löschen von Kundendaten zu berücksichtigen sind
 - Löschen von Nachrichtendaten, die in einer Anwendungswarteschlange gespeichert sind, durch:
 - Einzelne Nachrichten unter Verwendung der Messaging-API oder -Tools oder unter Verwendung von Nachrichtenverfallszeit entfernen
 - Angeben, dass Nachrichten nicht persistent sind, in einer Warteschlange gehalten werden, in der die nicht persistente Nachrichtenklasse normal ist und der Warteschlangenmanager erneut gestartet wird.
 - Die Warteschlange wird administrativ gelöscht.
 - Die Warteschlange wird gelöscht.
 - Gespeicherter Veröffentlichungsdaten, die in einem Thema gespeichert sind, löschen von:
 - Angeben, dass Nachrichten nicht persistent sind und den Warteschlangenmanager erneut starten.
 - Die aufbewahrten Daten durch neue Daten ersetzen oder die Nachrichtenablaufzeit verwenden.
 - Die Themenzeichenfolge wird administrativ gelöscht.
 - Löschen Sie Daten, die auf einem Warteschlangenmanager gespeichert sind, indem Sie den gesamten Warteschlangenmanager löschen.
 - Löschen Sie die Daten, die von den Service-Trace-Befehlen gespeichert werden, indem Sie die Dateien im Traceverzeichnis löschen.
 - Löschen Sie FFST-Daten, die gespeichert werden, indem Sie die Dateien im Fehlerverzeichnis löschen.
 - Löschen Sie Adressräume und Coupling-Facility-Speicherauszüge (unter z/OS).
 - Löschen Sie Archiv-, Backup-oder andere Kopien dieser Daten.
- Bereiche des IBM MQ -Verhaltens, die für die Einhaltung der DSGVO beim Löschen von Kontodaten zu berücksichtigen sind
 - Sie können Kontodaten und Vorgaben, die von IBM MQ gespeichert wurden, für die Verbindung zu Warteschlangenmanagern und 3rd -Partieservices löschen, indem Sie (einschließlich Archivierung, Sicherung oder anderweitig replizierten Kopien davon) löschen:
 - Authentifizierungsdaten des Warteschlangenmanagers, die Berechtigungsnachweise speichern.
 - WS-Manager-Berechtigungsdatensätze, die auf Benutzer-IDs verweisen.
 - WS-Manager-Kanalauthentifizierungsregeln, die bestimmte IP-Adressen, DNS oder Benutzer-IDs von Zertifikaten zuordnen oder blockieren.

- Berechtigungsnachweisdateien, die von IBM MQ Managed File Transfer Agent, Logger und MQ Explorer MFT-Plug-in für die Authentifizierung bei Warteschlangenmanagern und Dateiservern verwendet werden.
- X.509 digitale Zertifikate, die Informationen zu einer Person aus Keystores darstellen oder enthalten, die von SSL/TLS-Verbindungen oder IBM MQ Advanced Message Security (AMS) verwendet werden können.
- Einzelne Benutzerkonten von IBM MQ Appliance, einschließlich Verweis auf diese Konten in Systemprotokolldateien.
- IBM MQ Explorer-Arbeitsbereichsmetadaten und Eclipse -Einstellungen.
- IBM MQ Explorer-Kennwortspeicher gemäß Angabe in den [Kennwortvorgaben](#).
- Konfigurationsdateien für IBM MQ Console und mqweb-Server.
- Salesforce-Verbindungsdatenkonfigurationsdateien.
- Konfigurationsdateien für blockkettenbezogene Verbindungsdaten.
- IBM Cloud Product Insights -Verbindungsdaten unter der Zeilengruppe ReportingService in qm.ini und APIKeyFile.

Weitere Informationen finden Sie hier:

- [IBM MQ Bridge für Salesforce konfigurieren](#)
- [IBM MQ für die Verwendung mit Blockchain konfigurieren](#)
- [MFT-und IBM MQ-Verbindungsauthentifizierung](#)
- [Berechtigungsnachweise für einen Dateiserver mithilfe der Datei "ProtocolBridgeCredentials.xml" zuordnen](#)
- [Benutzer und Rollen von IBM MQ Console konfigurieren](#)

Datenüberwachung

IBM MQ stellt eine Reihe von Überwachungsfunktionen bereit, die Benutzer nutzen können, um ein besseres Verständnis der Leistung von Anwendungen und Warteschlangenmanagern zu erhalten.

IBM MQ stellt außerdem eine Reihe von Funktionen bereit, mit denen Fehlerprotokolle des Warteschlangenmanagers verwaltet werden können.

Weitere Informationen finden Sie hier:

- [IBM MQ-Netz überwachen](#)
- [Diagnosenachrichtenservices](#)
- [QMErrorLog-Service](#)

IBM MQ stellt eine Funktion zur Verfügung, mit der Benutzer Informationen in einem IBM Cloud Product Insights -Service veröffentlichen können, sodass der Benutzer von IBM MQ Start- und Nutzungsinformationen des Warteschlangenmanagers anzeigen kann.

Weitere Informationen finden Sie hier:

- [IBM MQ für die Verwendung mit IBM Cloud Product Insights-Service in IBM Cloud konfigurieren](#)

Funktionalität für die Einschränkung der Verwendung von persönlichen Daten

Mithilfe der in diesem Dokument zusammengefassten Funktionen ermöglicht IBM MQ einem Endbenutzer, die Nutzung seiner personenbezogenen Daten einzuschränken.

IBM MQ -Nachrichtenwarteschlangen sollten nicht auf dieselbe Weise wie eine Datenbank als permanenter Datenspeicher verwendet werden. Dies gilt insbesondere für die Verarbeitung von Anwendungsdaten, die der DSGVO unterliegen.

Im Gegensatz zu einer Datenbank, in der Daten über eine Suchabfrage gefunden werden können, kann es schwierig sein, Nachrichtendaten zu finden, es sei denn, Sie kennen die Warteschlangen-, Nachrichten- und Korrelations-IDs einer Nachricht.

Wenn Nachrichten, die die Daten einer Einzelperson enthalten, leicht identifiziert und lokalisiert werden können, ist es möglich, mithilfe der Standardnachrichtenfunktionen von IBM MQ auf Nachrichtendaten zuzugreifen oder diese zu ändern.

Dateiverwaltung

1. IBM MQ Managed File Transfer führt keine Malware-Scans für übertragene Dateien durch. Dateien werden übertragen, und es wird eine Integritätsprüfung durchgeführt, um sicherzustellen, dass die Dateidaten während der Übertragung nicht geändert werden. Die Quell- und Ziel-Prüfsummen werden als Teil der Übertragungsstatus-Publikation veröffentlicht. Es wird empfohlen, dass Endbenutzer Malware-Scans entsprechend ihrer Umgebung implementieren, bevor MFT die Datei überträgt und nachdem MFT eine Datei an einen fernen Endpunkt übermittelt.
2. IBM MQ Managed File Transfer führt keine Aktionen auf der Basis des MIME-Typs oder der Dateierweiterung aus. MFT liest die Dateien und überträgt die Bytes genau so, wie sie aus der Eingabedatei gelesen werden.

IBM WebSphere MQ -Architektur entwerfen

In diesem Abschnitt finden Sie Informationen zu den verschiedenen Architekturen, die IBM WebSphere MQ für Punkt-zu-Punkt- und Publish/Subscribe-Messaging unterstützt.

Bevor Sie Ihre IBM WebSphere MQ -Architektur planen, machen Sie sich mit den IBM WebSphere MQ -Basiskonzepten vertraut und lesen Sie die Themen unter [IBM WebSphere MQ -Technische Übersicht](#).

IBM WebSphere MQ-Architekturen reichen von einfachen Architekturen mit einem einzigen Warteschlangenmanager bis hin zu komplexeren Netzen miteinander verbundener Warteschlangenmanager. Mehrere Warteschlangenmanager sind über verteilte Warteschlangenverfahren miteinander verbunden. Weitere Informationen zum Planen eines einzelnen Warteschlangenmanagers und mehrerer WS-Manager-Architekturen finden Sie in den folgenden Themen:

- [„Architekturen auf der Basis eines einzelnen Warteschlangenmanagers“](#) auf Seite 15
- [„Architekturen auf der Basis von mehreren Warteschlangenmanagern“](#) auf Seite 17
- [„Netze und Netzplanung“](#) auf Seite 17
- [WebSphere MQ-Verfahren für verteilte Nachrichtenübermittlung](#)

Wenn Sie mehrere logisch zusammengehörige Warteschlangenmanager benötigen, die Daten und Anwendungen gemeinsam nutzen müssen, können sie in einem Cluster gruppiert werden. Die Verwendung von Clustern ermöglicht es Warteschlangenmanagern, miteinander zu kommunizieren, ohne dass zusätzliche Kanaldefinitionen oder Definitionen ferner Warteschlangen eingerichtet werden müssen, was ihre Konfiguration und Verwaltung vereinfacht. Weitere Informationen zur Verwendung von Clustern finden Sie unter [Funktionsweise von Clustern](#).

Zugehörige Konzepte

[„Planung“](#) auf Seite 5

Bei der Planung Ihrer IBM WebSphere MQ -Umgebung müssen Sie die zu konfigurierende IBM WebSphere MQ -Architektur, den Ressourcenbedarf, den Bedarf an Protokollierungs- und Sicherungsfunktionen berücksichtigen. Verwenden Sie die Links in diesem Abschnitt, um die Umgebung zu planen, in der IBM WebSphere MQ ausgeführt wird.

Zugehörige Tasks

[Konfiguration](#)

Architekturen auf der Basis eines einzelnen Warteschlangenmanagers

Zu den einfachsten IBM WebSphere MQ-Architekturen gehören solche, für die nur ein einzelner Warteschlangenmanager konfiguriert und verwendet wird.

Machen Sie sich vor der Planung Ihrer IBM WebSphere MQ -Architektur mit den IBM WebSphere MQ -Basiskonzepten vertraut (siehe [Einführung in IBM WebSphere MQ](#)).

Eine Reihe möglicher Architekturen mit einem einzigen Warteschlangenmanager werden in den folgenden Abschnitten beschrieben:

- [„Einzelwarteschlangenmanager mit lokalen Anwendungen, die auf einen Service zugreifen“ auf Seite 16](#)
- [„Einzelnes Warteschlangenmanager mit fernen Anwendungen, die auf einen Service als Clients zugreifen“ auf Seite 16](#)
- [„Einzelwarteschlangenmanager mit einer Publish/Subscribe-Konfiguration“ auf Seite 16](#)

Einzelwarteschlangenmanager mit lokalen Anwendungen, die auf einen Service zugreifen

Die erste Architektur auf der Basis eines einzigen Warteschlangenmanagers besteht darin, dass die Anwendungen, die auf einen Service zugreifen, auf demselben System ausgeführt werden wie die Anwendungen, die den Service bereitstellen. Ein IBM WebSphere MQ-Warteschlangenmanager stellt die asynchrone Kommunikation zwischen den Anwendungen, die den Service anfordern, und den Anwendungen, die den Service bereitstellen, bereit. Dies bedeutet, dass die Kommunikation zwischen den Anwendungen auch dann fortgesetzt werden kann, wenn eine der Anwendungen für einen längeren Zeitraum offline ist.

Einzelnes Warteschlangenmanager mit fernen Anwendungen, die auf einen Service als Clients zugreifen

Die zweite Architektur auf der Basis eines einzelnen Warteschlangenmanagers verfügt über die Anwendungen, die über Remotezugriff von den Anwendungen ausgeführt werden, die den Service bereitstellen. Die fernen Anwendungen werden auf verschiedenen Systemen für die Services ausgeführt. Die Anwendungen stellen eine Verbindung als Clients mit dem einzelnen Warteschlangenmanager her. Dies bedeutet, dass der Zugriff auf einen Service mehreren Systemen über einen einzigen Warteschlangenmanager zur Verfügung gestellt werden kann.

Eine Einschränkung dieser Architektur besteht darin, dass eine Netzverbindung verfügbar sein muss, damit eine Anwendung ausgeführt werden kann. Die Interaktion zwischen der Anwendung und dem WS-Manager über die Netzverbindung erfolgt synchron.

Einzelwarteschlangenmanager mit einer Publish/Subscribe-Konfiguration

Eine alternative Architektur, die einen einzelnen WS-Manager verwendet, ist die Verwendung einer Publish/Subscribe-Konfiguration. Beim Publish/Subscribe-Messaging können Sie den Anbieter von Informationen von den Konsumenten dieser Informationen entkoppeln. Dies unterscheidet sich vom Punkt-zu-Punkt-Messaging-Stile in den zuvor beschriebenen Architekturen, wo die Anwendungen Informationen über die Zielanwendung kennen müssen, z. B. den Namen der Warteschlange, in die Nachrichten gestellt werden sollen. Mit IBM WebSphere MQ Publish/Subscribe veröffentlicht die sendende Anwendung eine Nachricht mit einem angegebenen Thema auf der Basis des Betreffs der Informationen. IBM WebSphere MQ sorgt für die Verteilung der Nachricht an Anwendungen, die mithilfe einer Subskription ihr Interesse an diesem Inhalt angemeldet haben. Die empfangenden Anwendungen müssen außerdem nichts über die Quelle der Nachrichten wissen, um sie zu empfangen. Weitere Informationen zum Publish/Subscribe-Messaging finden Sie im Abschnitt [Einführung in das WebSphere MQ-Publish/Subscribe-Messaging](#). Ein Beispiel für Publish/Subscribe-Messaging mit einem einzelnen Warteschlangenmanager finden Sie unter [Beispiel einer Publish/Subscribe-Konfiguration mit einem einzelnen Warteschlangenmanager](#).

Zugehörige Konzepte

[„IBM WebSphere MQ -Architektur entwerfen“ auf Seite 15](#)

In diesem Abschnitt finden Sie Informationen zu den verschiedenen Architekturen, die IBM WebSphere MQ für Punkt-zu-Punkt- und Publish/Subscribe-Messaging unterstützt.

Zugehörige Informationen

[Einführung in WebSphere MQ](#)

Architekturen auf der Basis von mehreren Warteschlangenmanagern

Sie können Methoden zur Steuerung der Warteschlangen für verteilte Nachrichten verwenden, um eine IBM WebSphere MQ-Architektur zu erstellen, die die Konfiguration und Verwendung mehrerer Warteschlangenmanager umfasst.

Machen Sie sich vor der Planung Ihrer IBM WebSphere MQ -Architektur mit den IBM WebSphere MQ -Basiskonzepten vertraut (siehe [Einführung in IBM WebSphere MQ](#)).

Eine IBM WebSphere MQ-Architektur kann geändert werden, ohne dass Änderungen an Anwendungen, die Services bereitstellen, erforderlich sind, indem zusätzliche Warteschlangenmanager hinzugefügt werden.

Anwendungen können auf derselben Maschine wie ein Warteschlangenmanager gehostet werden und dann asynchrone Kommunikation mit einem Service erhalten, der auf einem anderen Warteschlangenmanager auf einem anderen System gehostet wird. Alternativ können Anwendungen, die auf einen Service zugreifen, als Clients eine Verbindung zu einem Warteschlangenmanager herstellen, der dann den asynchronen Zugriff auf den Service auf einem anderen Warteschlangenmanager bereitstellt.

Routes, die verschiedene Warteschlangenmanager und ihre Warteschlangen verbinden, werden mithilfe von verteilten Warteschlangenverfahren definiert. Die Warteschlangenmanager in der Architektur werden über Kanäle miteinander verbunden. Kanäle werden verwendet, um Nachrichten automatisch von einem Warteschlangenmanager in eine andere Richtung in eine andere Richtung zu versetzen, abhängig von der Konfiguration der Warteschlangenmanager.

Eine allgemeine Übersicht über die Planung eines IBM WebSphere MQ -Netztes finden Sie unter „[Netze und Netzplanung](#)“ auf Seite 17.

Informationen zur Planung von Kanälen für Ihre IBM WebSphere MQ -Architektur finden Sie im Abschnitt [WebSphere MQ Verfahren für verteiltes Messaging](#).

Über die verteilte Warteschlangenverwaltung können Sie die Kommunikation zwischen Warteschlangenmanagern erstellen und überwachen. Weitere Informationen zur verteilten Warteschlangenverwaltung finden Sie im Abschnitt [Einführung in die verteilte Warteschlangenverwaltung](#).

Zugehörige Konzepte

[Einführung in WebSphere MQ](#)

„[IBM WebSphere MQ -Architektur entwerfen](#)“ auf Seite 15

In diesem Abschnitt finden Sie Informationen zu den verschiedenen Architekturen, die IBM WebSphere MQ für Punkt-zu-Punkt- und Publish/Subscribe-Messaging unterstützt.

Zugehörige Tasks

[Warteschlangenmanager erstellen und verwalten](#)

Netze und Netzplanung

WebSphere MQ sendet und empfängt Daten zwischen Anwendungen und über Netze mit Warteschlangenmanagern und Kanälen. Die Netzplanung umfasst die Definition von Anforderungen zum Erstellen eines Frameworks für die Verbindung dieser Systeme über ein Netz.

Kanäle können zwischen Ihrem System und jedem anderen System, mit dem Sie Kommunikation benötigen, erstellt werden. Multi-Hop-Kanäle können erstellt werden, um eine Verbindung zu Systemen herzustellen, auf denen Sie keine direkten Verbindungen haben. Die in den Szenarios beschriebenen Nachrichtenkanalverbindungen werden in [Abbildung 1 auf Seite 18](#) als Netzdiagramm dargestellt.

Namen der Kanal- und Übertragungswarteschlangen

Der Übertragungswarteschlange kann ein beliebiger Name gegeben werden. Um jedoch Unklarheiten zu vermeiden, können Sie ihnen dieselben Namen wie die Namen des Zielwarteschlangenmanagers oder die Aliasnamen des Warteschlangenmanagers geben. Dadurch wird die Übertragungswarteschlange der

Route zugeordnet, die sie verwenden, und gibt einen klaren Überblick über parallele Routen, die über temporäre (mehrere-hackte) Warteschlangenmanager erstellt werden.

Es ist nicht so klar, dass die Kanalnamen abgeschnitten sind. Die Kanalnamen in [Abbildung 1 auf Seite 18](#) für QM2 müssen sich beispielsweise für eingehende und abgehende Kanäle unterscheiden. Alle Kanalnamen können noch ihre Namen für die Übertragungswarteschlange enthalten, aber sie müssen qualifiziert sein, um sie eindeutig zu machen.

In WSM2 gibt es beispielsweise einen WSM3-Kanal von WSM1 und ein WSM3-Kanal zu QM3. Als eindeutige Namen könnten Sie für den ersten beispielsweise 'QM3_von_QM1' und für den zweiten 'QM3_von_QM2' wählen. Auf diese Weise zeigen die Kanalnamen den Namen der Übertragungswarteschlange im ersten Teil des Namens an. Die Richtung und der benachbarte WS-Manager-Name werden im zweiten Teil des Namens angezeigt.

Eine Tabelle mit den vorgeschlagenen Kanalnamen für [Abbildung 1 auf Seite 18](#) finden Sie in [Tabelle 1 auf Seite 18](#).

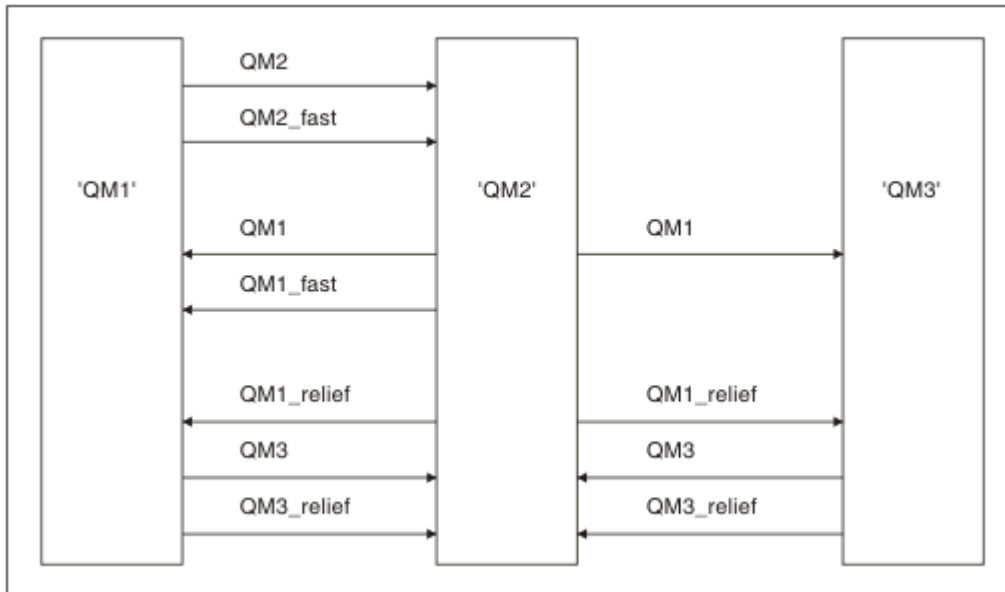


Abbildung 1. Netzdiagramm mit allen Kanälen

Tabelle 1. Beispiel für Kanalnamen			
Leitwegname	Kanal des Warteschlangenmanagers	Name der Übertragungswarteschlange	Empfohlene Kanalbezeichnung
QM1	QM1 & QM2	WSM1 (bei WSM2)	QM1.from.QM2
QM1	QM2 & QM3	WSM1 (bei WSM3)	QM1.from.QM3
QM1_fast	QM1 & QM2	QM1_schnell (bei WSM2)	QM1_fast.from.QM2
QM1_relief	QM1 & QM2	QM1_relief (bei QM2)	QM1_relief.from.QM2
QM1_relief	QM2 & QM3	QM1_relief (bei QM3)	QM1_relief.from.QM3
QM2	QM1 & QM2	WSM2 (bei WSM1)	QM2.from.QM1
QM2_fast	QM1 & QM2	QM2_schnell (bei QM1)	QM2_fast.from.QM1
QM3	QM1 & QM2	WSM3 (bei WSM1)	QM3.from.QM1
QM3	QM2 & QM3	WSM3 (bei WSM2)	QM3.from.QM2

Tabelle 1. Beispiel für Kanalnamen (Forts.)

Leitwegname	Kanal des Warteschlangenmanagers	Name der Übertragungswarteschlange	Empfohlene Kanalbezeichnung
QM3_relief	QM1 & QM2	QM3_relief (bei QM1)	QM3_relief.from.QM1
QM3_relief	QM2 & QM3	QM3_relief (bei QM2)	QM3_relief.from.QM2

Anmerkung:

1. Unter WebSphere MQ für z/OS sind Warteschlangenmanagernamen auf vier Zeichen begrenzt.
2. Nennen Sie alle Kanäle in Ihrem Netzwerk eindeutig. Wie in [Tabelle 1 auf Seite 18](#) gezeigt, ist dies eine gute Möglichkeit, die Namen der Quellen- und Zielwarteschlangenmanager in den Kanalnamen zu verwenden.

Netzplaner

Bei der Erstellung eines Netzes wird davon ausgegangen, dass eine andere, übergeordnete Funktion von *network planner* vorhanden ist, deren Pläne von den anderen Mitgliedern des Teams implementiert werden.

Für weit verbreitete Anwendungen ist es ökonomischer, in Bezug auf lokale Zugriffsseiten für die Konzentration des Nachrichtenverkehrs zu denken. Verwenden Sie die Breitband-Verbindungen zwischen den lokalen Zugriffsseiten (siehe [Abbildung 2 auf Seite 20](#)).

In diesem Beispiel gibt es zwei Hauptsysteme und eine Reihe von Satellitensystemen. Die tatsächliche Konfiguration hängt von den Geschäftsaspekten ab. Es gibt zwei Konzentratortwarteschlangenmanager, die sich in praktischen Centern befinden. Jeder QM-Konzentrator verfügt über Nachrichtenkanäle zu den lokalen WS-Managern:

- Der QM-Konzentrator 1 verfügt über Nachrichtenkanäle zu jedem der drei lokalen WS-Manager QM1, QM2 und QM3. Die Anwendungen, die diese WS-Manager verwenden, können über die QM-Konzentratoren miteinander kommunizieren.
- Der QM-Konzentrator 2 verfügt über Nachrichtenkanäle zu jedem der drei lokalen WS-Manager QM4, QM5 und QM6. Die Anwendungen, die diese WS-Manager verwenden, können über die QM-Konzentratoren miteinander kommunizieren.
- Die QM-Konzentratoren haben Nachrichtenkanäle untereinander, so dass jede Anwendung in einem WS-Manager Nachrichten mit jeder anderen Anwendung in einem anderen WS-Manager austauschen kann.

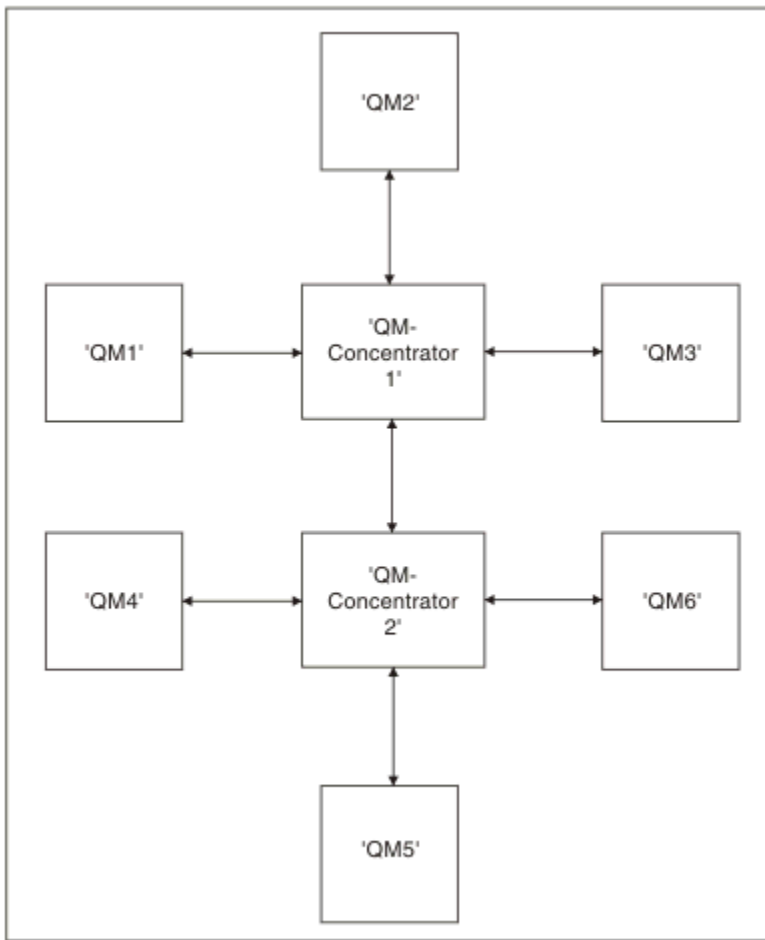


Abbildung 2. Netzdiagramm mit QM-Konzentratoren

Clustering

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.

Die folgenden Informationen setzen voraus, dass der Benutzer grundlegende Kenntnisse über IBM WebSphere MQ -Cluster hat. Diese Informationen sind nicht als "Einheitslösung" gedacht, sondern versuchen, gemeinsame Ansätze für häufig auftretende Probleme zu nutzen.

Cluster stellen einen Mechanismus für die Verbindung von Warteschlangenmanagern auf eine Weise bereit, die die Erstkonfiguration, die zum Einrichten des Systems erforderlich ist, und die fortlaufende Verwaltung vereinfacht. Je größer die Konfiguration, desto größer der Vorteil.

Bei der Planung von Clustersystemen ist Vorsicht geboten, um sicherzustellen, dass sie ordnungsgemäß funktionieren, und um die für das System erforderliche Verfügbarkeit und Reaktionsfähigkeit sicherzustellen, insbesondere bei größeren oder komplexeren Clustersystemen.

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 unter Konzepte der übergreifenden Kommunikation und Funktionsweise von Clustern vertraut sind.

Was sind Cluster und warum werden sie verwendet?

Cluster bieten zwei wichtige Vorteile:

- Cluster vereinfachen die Verwaltung von IBM WebSphere MQ -Netzen, für die normalerweise viele Objektdefinitionen für Kanäle, Übertragungswarteschlangen und ferne Warteschlangen konfiguriert werden müssen. Besonders gravierend ist dieser Aufwand in großen Netzen, die sich häufig ändern und in denen die Verbindung vieler Warteschlangenmanager erforderlich ist. Bei einer solchen Architektur ist die Konfiguration und Verwaltung ohne Cluster besonders anspruchsvoll.
- In Clustern kann die Arbeitslast der Nachrichtenübertragung über mehrere Warteschlangen und Warteschlangenmanager verteilt werden. Eine solche Verteilung sieht die Verteilung der Arbeitslast einer einzelnen Warteschlange auf identische Instanzen dieser Warteschlange auf mehreren Warteschlangenmanagern vor. Durch diese Verteilung erreichen Sie zum Einen eine größere Fehlersicherheit des Systems, zum Anderen verbessern Sie aber auch die Skalierleistung besonders aktiver Nachrichtenflüsse im System. In einer solchen Umgebung verfügt jede Instanz der verteilten Warteschlangen über konsumierende Anwendungen, die die Nachrichten verarbeiten.

Zugehörige Informationen

[Clustering: Bewährte Verfahren](#)

Punkt-zu-Punkt-Messaging

Die einfachste Form des Messaging in IBM WebSphere MQ ist das Punkt-zu-Punkt-Messaging.

Beim Punkt-zu-Punkt-Messaging muss eine sendende Anwendung Informationen zu einer anderen empfangenden Anwendung erhalten, damit sie eine Nachricht an diese Anwendung senden kann. Die sendende Anwendung muss beispielsweise den Namen der Warteschlange kennen, an die die Informationen gesendet werden sollen, und möglicherweise muss auch der Name eines Warteschlangenmanagers angegeben werden.

Ein alternativer Messaging-Stil, den Sie mit IBM WebSphere MQ verwenden können, ist Publish/Subscribe-Messaging. Das Publish/Subscribe-Messaging ermöglicht die Entkopplung des Informationsanbieters von den Informationsempfängern. Die Sender- und Empfängeranwendungen müssen untereinander keine Informationen über die gesendeten und empfangenen Nachrichten austauschen. Weitere Informationen zum Publish/Subscribe-Messaging finden Sie im Abschnitt [Einführung in das WebSphere MQ-Publish/Subscribe-Messaging](#).

Zugehörige Informationen

[Anwendungen entwickeln](#)

[WebSphere MQ-Nachrichten](#)

Einführung in IBM WebSphere MQ Publish/Subscribe-Messaging

Das Publish/Subscribe-Messaging ermöglicht die Entkopplung des Informationsanbieters von den Informationsempfängern. Die Sender- und Empfängeranwendungen müssen untereinander keine Informationen über die gesendeten und empfangenen Nachrichten austauschen.

Bevor eine Punkt-zu-Punkt-Anwendung in IBM WebSphere MQ eine Nachricht an eine andere Anwendung senden kann, werden Informationen zu dieser Anwendung benötigt. Beispielsweise ist der Name der Warteschlange erforderlich, an die die Information gesendet werden soll, und möglicherweise muss auch ein Warteschlangenmanagername angegeben werden.

Durch IBM WebSphere MQ Publish/Subscribe entfällt für eine Anwendung die Notwendigkeit, Informationen zu der Zielanwendung zu haben. Alles, was die sendende Anwendung tun muss, ist, eine IBM WebSphere MQ -Nachricht einzureihen, die die gewünschten Informationen enthält, und ihr ein Thema zuzuordnen, das das Thema der Informationen bezeichnet, und IBM WebSphere MQ die Verteilung dieser Informationen zu überlassen. Ebenso muss auch die Zielanwendung keine Angaben zur Quelle der empfangenen Informationen haben.

[Abbildung 3 auf Seite 22](#) zeigt das einfachste Publish/Subscribe-System. Es gibt einen Publisher, einen Warteschlangenmanager und einen Subskribenten. Eine Subskription wird vom Subskribenten an den Warteschlangenmanager gesendet, eine Veröffentlichung wird vom Publisher an den Warteschlangenmanager gesendet und die Veröffentlichung wird dann vom Warteschlangenmanager an den Subskribenten weitergeleitet.

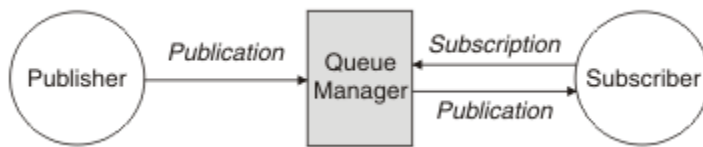


Abbildung 3. Einfache Publish/Subscribe-Konfiguration

Ein typisches Publish/Subscribe-System hat mehr als einen Publisher und mehr als einen Subskribenten und häufig mehr als einen Warteschlangenmanager. Eine Anwendung kann sowohl als Veröffentlichungsanwendung als auch als Subskribent fungieren.

Übersicht über Publish/Subscribe-Komponenten

Publish/Subscribe ist der Mechanismus, über den Subskribenten von Publishern Informationen in Form von Nachrichten erhalten. Die Interaktionen zwischen Publishern und Subskribenten werden von Warteschlangenmanagern über Standardfunktionen von WebSphere MQ gesteuert.

Ein typisches Publish/Subscribe-System hat mehr als einen Publisher und mehr als einen Subskribenten und häufig mehr als einen Warteschlangenmanager. Eine Anwendung kann sowohl als Veröffentlichungsanwendung als auch als Subskribent fungieren.

Der Bereitsteller von Informationen wird als *Publisher* bezeichnet. Publisher stellen Informationen zu einem Thema bereit, ohne irgendetwas über die Anwendungen, für die diese Informationen bestimmt sind, wissen zu müssen. Publisher erstellen diese Informationen in Form von Nachrichten, so genannten *Veröffentlichungen*, deren Thema sie definieren und die sie veröffentlichen.

Der Konsument der Informationen wird als *Subskribent* bezeichnet. Subskribenten erstellen *Subskriptionen*, die das Thema beschreiben, an dem sie interessiert sind. Eine Subskription bestimmt also, welche Veröffentlichungen an den Subskribenten weitergeleitet werden. Subskribenten können mehrere Subskriptionen erstellen und Informationen von vielen verschiedenen Publishern erhalten.

Veröffentlichte Informationen werden in einer WebSphere MQ -Nachricht gesendet und der Betreff der Informationen wird durch das zugehörige *Thema* angegeben. Das Thema wird bei der Veröffentlichung der Informationen vom Publisher festgelegt. Der Subskribent wiederum gibt die Themen an, zu denen er Veröffentlichungen erhalten möchte. Der Subskribent erhält nur Informationen zu den von ihm subskribierten Themen.

Durch Themen können beim Publish/Subscribe-Messaging Informationsbereitsteller von den Konsumenten der Informationen getrennt werden, denn hier ist es im Gegensatz zum Punkt-zu-Punkt-Messaging nicht erforderlich, das Ziel einer Nachricht anzugeben.

Die Interaktionen zwischen Publishern und Subskribenten werden von einem Warteschlangenmanager gesteuert. Der Warteschlangenmanager empfängt die Nachrichten der Publisher und die Subskriptionen (zu bestimmten Themen) der Subskribenten. Die Aufgabe des Warteschlangenmanagers ist es, die veröffentlichten Nachrichten an die Subskribenten weiterzuleiten, die ihr Interesse an den Themen der Nachrichten registriert haben.

Standardfunktionen von WebSphere MQ werden zum Verteilen von Nachrichten verwendet, sodass Ihre Anwendungen alle Funktionen nutzen können, die für vorhandene WebSphere MQ -Anwendungen verfügbar sind. Für die zuverlässige Nachrichtenübermittlung können Sie also auch persistente Nachrichten verwenden. Ebenso können Ihre Nachrichten Teil einer transaktionsorientierten Arbeitseinheit sein, durch die sichergestellt wird, dass Nachrichten nur dann dem Subskribenten zugestellt werden, wenn sie vom Publisher festgeschrieben wurden.

Beispiel für die Publish/Subscribe-Konfiguration für einen einzelnen Warteschlangenmanager

Abbildung 4 auf Seite 23 zeigt eine grundlegende Publish/Subscribe-Konfiguration für einen einzelnen Warteschlangenmanager. Das Beispiel zeigt die Konfiguration für einen Nachrichtenservice, der Informationen von verschiedenen Publishern zu mehreren Themen bereitstellt:

- Publisher 1 veröffentlicht Sportergebnisse unter dem Thema "Sport"
- Publisher 2 veröffentlicht Aktienkurse unter dem Thema "Stock" (Aktien)
- Publisher 3 veröffentlicht Filmkritiken unter dem Thema "Films" (Filme) und das Fernsehprogramm unter dem Thema "TV"

Drei Subskribenten haben ihr Interesse an verschiedenen Themen angemeldet, sodass ihnen der Warteschlangenmanager die entsprechenden Informationen zusendet:

- Subskribent 1 erhält die Sportergebnisse und Aktienkurse
- Subskribent 2 erhält die Filmkritiken
- Subskribent 3 erhält die Sportergebnisse

Keiner der Subskribenten ist am Thema "TV" interessiert, das Fernsehprogramm wird daher nicht verteilt.

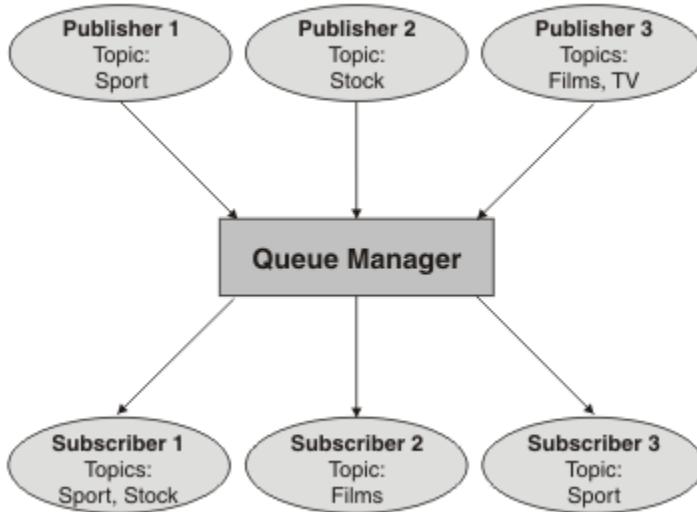


Abbildung 4. Beispiel für die Publish/Subscribe-Konfiguration für einen einzelnen Warteschlangenmanager

Veröffentlichungskomponenten und Veröffentlichungen

In WebSphere MQ Publish/Subscribe ist ein Publisher eine Anwendung, die einem Warteschlangenmanager Informationen zu einem angegebenen Thema in Form einer WebSphere MQ -Standardnachricht, die als Veröffentlichung bezeichnet wird, zur Verfügung stellt. Eine Veröffentlichungskomponente kann Informationen zu mehreren Themen veröffentlichen.

Veröffentlichungskomponenten verwenden das Verb MQPUT, um eine Nachricht in ein zuvor geöffnetes Thema einzureihen. Diese Nachricht ist eine Veröffentlichung. Der lokale Warteschlangenmanager leitet die Veröffentlichung dann an alle Subskribenten weiter, die Subskriptionen zum Thema der Veröffentlichung eingerichtet haben. Eine veröffentlichte Nachricht kann von mehreren Subskribenten gelesen werden.

Zusätzlich zum Verteilen von Veröffentlichungen an alle lokalen Subskribenten mit entsprechenden Subskriptionen kann ein Warteschlangenmanager eine Veröffentlichung auch an beliebige andere Warteschlangenmanager weiterleiten, die entweder direkt oder über ein Netz von Warteschlangenmanagern, die Subskribenten für dieses Thema verwalten, mit ihm verbunden sind.

In einem Publish/Subscribe-Netz mit WebSphere MQ kann eine veröffentlichende Anwendung auch ein Subskribent sein.

Veröffentlichungen unter Synchronisationspunkt

Veröffentlichungskomponenten können MQPUT- oder MQPUT1-Aufrufe im Rahmen einer Synchronisationspunktverarbeitung ausgeben, um alle Nachrichten einzuschließen, die in einer Arbeitseinheit an Subskribenten geliefert werden. Wenn die Option MQPMO_RETAIN oder die Themabereitstellungsoptionen NPMGDLV und PMSGDLV mit dem Wert ALL oder ALLDUR angegeben werden, verwendet der War-

teschlangenmanager interne MQPUT- oder MQPUT1-Aufrufe unter Synchronisationspunktverarbeitung innerhalb des Geltungsbereichs des MQPUT- oder MQPUT1-Aufrufs der Veröffentlichungskomponente.

Status- und Ereignisinformationen

Bei Veröffentlichungen kann es sich um Statusveröffentlichungen (z. B. der aktuelle Aktienkurs) oder um Ereignisveröffentlichungen (z. B. ein Aktienverkauf) handeln.

Statusveröffentlichungen

Statusveröffentlichungen enthalten Informationen zum aktuellen Status oder Zustand von etwas, beispielsweise den Kurs einer Aktie oder den aktuellen Stand eines Fußballspiels. Sobald eine Änderung eintritt (sich also beispielsweise der Aktienpreis oder der Punktestand ändert), wird die vorherige Statusinformation nicht mehr benötigt, da sie durch die neue Information ersetzt wird.

Ein Subskribent möchte zu Beginn seiner Subskription die aktuelle Version der Statusinformationen erhalten. Diese möchte er aktualisiert haben, sobald sich der Status ändert.

Eine Veröffentlichung mit Statusinformationen wird meist als ständige Veröffentlichung veröffentlicht. Ein neuer Subskribent möchte die aktuellen Statusinformationen selbstverständlich sofort erhalten und nicht auf ein Ereignis warten müssen, das die erneute Veröffentlichung der Informationen auslöst. Subskribenten erhalten die ständige Veröffentlichung eines Themas automatisch bei der Subskription, es sei denn, der Subskribent verwendet die Option MQSO_PUBLICATIONS_ON_REQUEST oder MQSO_NEW_PUBLICATIONS_ONLY.

Ereignisveröffentlichungen

Ereignisveröffentlichungen enthalten Informationen zu einzelnen Ereignissen, beispielsweise zu einem Aktienhandel oder einem gefallenen Tor in einem Fußballspiel. Jedes Ereignis ist unabhängig von anderen Ereignissen.

Ein Subskribent möchte die Informationen zu einem Ereignis sofort erhalten, wenn es eintritt.

Ständige Veröffentlichungen

Eine Veröffentlichung wird nach dem Senden an alle interessierten Subskribenten standardmäßig gelöscht. Für den Fall, dass später weitere Subskribenten ihr Interesse an der Veröffentlichung anmelden, kann der Publisher aber auch festlegen, dass eine Kopie der Veröffentlichung aufbewahrt wird.

An alle interessierte Subskribenten gesendete Veröffentlichungen können im Falle von Ereignisdaten meist gelöscht werden, im Falle von Statusinformationen häufig aber nicht. Wird eine solche Nachricht aufbewahrt, müssen neue Subskribenten nicht auf eine erneute Veröffentlichung der Informationen warten, um die anfänglichen Statusinformationen zu erhalten. Ein Subskribent eines Aktienkurses erhielte den aktuellen Kurs dann sofort, ohne dass er auf eine Änderung (und damit erneute Veröffentlichung) des Aktienkurses warten müsste.

Der Warteschlangenmanager kann pro Thema nur eine Veröffentlichung aufbewahren. Die vorhandene ständige Veröffentlichung eines Themas wird daher gelöscht, sobald beim Warteschlangenmanager für das gleiche Thema eine neue ständige Veröffentlichung eingeht. Jedoch kann es passieren, dass die vorhandene Veröffentlichung nicht exakt gleichzeitig mit der Ankunft der neuen Veröffentlichung gelöscht wird. Daher sollten Sie ständige Veröffentlichungen zum gleichen Thema möglichst nur von einem Publisher aus senden.

Mit der Subskriptionsoption MQSO_NEW_PUBLICATIONS_ONLY können Subskribenten angeben, dass sie keine ständigen Veröffentlichungen erhalten wollen. Genauso ist es aber auch möglich, Duplikatkopien ständiger Veröffentlichungen anzufordern.

Eventuell möchten Sie in bestimmten Fällen keine ständigen Veröffentlichungen erstellen, nicht einmal für Statusinformationen. Dies kann zum Beispiel in den folgenden Situationen der Fall sein:

- Wenn alle Subskriptionen zu einem Thema abgeschlossen wurden, bevor zu diesem Thema Veröffentlichungen vorlagen, und Sie nicht davon ausgehen, dass neue Subskriptionen eingehen (bzw. keine

neuen Subskriptionen zulassen), besteht kein Grund zur Aufbewahrung ständiger Veröffentlichungen, da alle Veröffentlichungen bereits bei ihrer ersten Veröffentlichung allen Subskribenten zugestellt wurden.

- Wenn Veröffentlichungen in einer sehr schnellen Abfolge versendet werden, zum Beispiel jede Sekunde, erhält ein neuer Subskribent bzw. ein Subskribent nach einem Recovery den aktuellen Status nahezu sofort nach dem Abschluss der Subskription. Es besteht für solche Veröffentlichungen daher kein Grund zur Aufbewahrung.
- Bei sehr großen Veröffentlichungen und/oder sehr vielen Themen benötigen Sie erheblichen Speicherplatz zur Aufbewahrung der ständigen Veröffentlichungen zu allen Themen. In einer Umgebung mit mehreren Warteschlangenmanagern werden ständige Veröffentlichungen auf allen Warteschlangenmanagern im Netz gespeichert, auf denen eine entsprechende Subskription vorliegt.

In die Überlegung, ob ständige Veröffentlichungen erstellt werden sollen, sollte auch einfließen, wie die abonnierenden Anwendungen nach einem Fehler wiederhergestellt werden. Wenn der Publisher keine ständigen Veröffentlichungen verwendet, muss die abonnierende Anwendung ihren aktuellen Status eventuell lokal speichern.

Zur Aufbewahrung einer ständigen Veröffentlichung verwenden Sie die Put-Option MQPMO_RETAIN. Wenn diese Option angegeben, aber eine ständige Veröffentlichung nicht möglich ist, wird die betreffende Nachricht nicht veröffentlicht und der Aufruf schlägt mit MQRC_PUT_NOT_RETAINED fehl.

Eine ständige Veröffentlichung ist durch die Nachrichteneigenschaft MQIsRetained gekennzeichnet. Die Persistenz einer Nachricht wird bei ihrer ursprünglichen Veröffentlichung festgelegt und bleibt danach unverändert.

Veröffentlichungen unter Synchronisationspunkt

In der Publish/Subscribe-Funktion von IBM WebSphere MQ kann der Synchronisationspunkt von Publishern oder intern vom Warteschlangenmanager verwendet werden.

Publisher verwenden den Synchronisationspunkt, wenn sie MQPUT/MQPUT1-Aufrufe mit der Option MQPMO_SYNCPOINT absetzen. Alle Nachrichten, die an Subskribenten zugestellt werden, werden bei der Berechnung der maximalen Anzahl von nicht festgeschriebenen Nachrichten in einer Arbeitseinheit berücksichtigt. Dieser Grenzwert wird über das Warteschlangenmanager-Attribut MAXUMSGS angegeben. Sobald der Grenzwert erreicht wird, empfängt der Publisher den Ursachencode 2024 (07E8) (RC2024): MQRC_SYNCPOINT_LIMIT_REACHED.

Wenn ein Publisher MQPUT/MQPUT1-Aufrufe unter Verwendung von MQPMO_NO_SYNCPOINT mit der Option MQPMO_RETAIN oder den Themenzustelloptionen NPMMSGDLV/PMSGDLV mit den Werten ALL oder ALLDUR absetzt, verwendet der Warteschlangenmanager interne Synchronisationspunkte, um zu garantieren, dass die Nachrichten wie angefordert zugestellt werden. Der Publisher kann den Ursachencode 2024 (07E8) (RC2024): MQRC_SYNCPOINT_LIMIT_REACHED empfangen, wenn der Grenzwert im Bereich des Publisheraufrufs MQPUT/MQPUT1 erreicht wird.

Subskribenten und Subskriptionen

In WebSphere MQ Publish/Subscribe ist ein Subskribent eine Anwendung, die Informationen zu einem bestimmten Thema von einem Warteschlangenmanager in einem Publish/Subscribe-Netz anfordert. Ein Subskribent kann Nachrichten zu einem oder mehreren Themen von einem oder mehreren Publishern erhalten.

Subskriptionen können manuell mit einem WebSphere MQ-Scriptbefehl oder durch eine Anwendung erstellt werden. Diese Subskriptionen werden dem lokalen Warteschlangenmanager mit Informationen zu den Veröffentlichungen übergeben, die der Subskribent erhalten möchte:

- Das Thema, an dem der Subskribent interessiert ist. Wenn Platzhalter verwendet werden, können dies auch mehrere Themen sein.
- Eine optionale Auswahlzeichenfolge, die auf die veröffentlichten Nachrichten angewendet wird.
- Eine Kennung für eine Warteschlange (bekannt als *Warteschlange für Subskribenten*), in die ausgewählte Veröffentlichungen eingereiht werden sollen, und die optionale CorrelId.

Der lokale Warteschlangenmanager speichert die Subskriptionsinformationen und untersucht diese, sobald er eine Veröffentlichung erhält, um festzustellen, ob ihm eine Subskription mit dem Thema und der Auswahlzeichenfolge dieser Veröffentlichung vorliegt. Bei einer übereinstimmenden Subskription leitet der Warteschlangenmanager die Veröffentlichung an die entsprechende Subskribentenwarteschlange weiter. Die Informationen, die ein Warteschlangenmanager zu Subskriptionen speichert, können mit den Befehlen DIS SUB und DIS SBSTATUS angezeigt werden.

Eine Subskription wird nur im Falle der folgenden Ereignisse gelöscht:

- Der Subskribent hebt die Subskription mit dem Aufruf MQCLOSE auf (nur bei nicht persistenten Subskriptionen).
- Die Subskription läuft ab.
- Die Subskription wird vom Systemadministrator mit dem Befehl DELETE SUB gelöscht.
- Die Subskribentenanwendung wird beendet (nur bei nicht persistenten Subskriptionen).
- Der Warteschlangenmanager wird gestoppt oder neu gestartet (nur bei nicht persistenten Subskriptionen).

Achten Sie darauf, dass Sie beim Abrufen von Nachrichten im MQGET-Aufruf die richtigen Optionen angeben. Wenn Ihre Anwendung nur Nachrichten für eine Subskription verarbeitet, sollten Sie mindestens `get-by-correlid` verwenden, wie im C-Beispielprogramm `amqssbxa.c` und unter Nicht verwalteter MQ-Subskribent veranschaulicht. Die zu verwendende **CorrelId** wird von MQSUB im MQSD zurückgegeben. **SubCorrelId**.

Verwaltete Warteschlangen und Publish/Subscribe

Wenn Sie eine Subskription erstellen, können Sie wählen, ob Sie die verwaltete Warteschlangensteuerung verwenden möchten. Wenn Sie verwaltete Warteschlangen verwenden, wird beim Erstellen einer Subskription automatisch eine Subskriptionswarteschlange erstellt. Verwaltete Warteschlangen werden entsprechend der Persistenz der Subskription automatisch bereinigt. Wenn Sie verwaltete Warteschlangen verwenden, brauchen Sie sich keine Gedanken mehr darüber zu machen, ob auch schon Warteschlangen für die Veröffentlichungen erstellt sind. Ebenso werden nicht konsumierte Veröffentlichungen automatisch aus den Subskribentenwarteschlangen gelöscht, sobald die Verbindung einer nicht persistenten Subskription geschlossen wird.

Wenn eine Anwendung keine bestimmte Warteschlange als Subskribentenwarteschlange, das Ziel für die empfangenen Veröffentlichungen, verwenden muss, kann sie die *verwalteten Subskriptionen* mit der Subskriptionsoption MQSO_MANAGED verwenden. Bei der Erstellung einer verwalteten Subskription gibt der Warteschlangenmanager dem Subskribenten eine Objektkennung für eine Subskribentenwarteschlange zurück, die der Warteschlangenmanager erstellt und in die er die Veröffentlichungen für diese Subskription einreicht. Mithilfe dieser Objektkennung können Sie die Warteschlange suchen, ihren Inhalt abrufen und sie untersuchen. Deren Attribute können Sie allerdings nicht einstellen, es sei denn, Sie haben explizit Zugriff auf temporäre dynamische Warteschlangen.

Die Persistenz einer Subskription bestimmt, ob die verwaltete Warteschlange erhalten bleibt, wenn die Verbindung zwischen der abonnierenden Anwendung und dem Warteschlangenmanager getrennt wird.

Verwaltete Subskriptionen sind besonders praktisch in Verbindung mit nicht persistenten Subskriptionen, da andernfalls nach einer Beendigung der Verbindung der abonnierenden Anwendung nicht konsumierte Nachrichten in der Subskribentenwarteschlange verbleiben und auf dem Warteschlangenmanager für immer Speicherplatz belegen würden. Bei einer verwalteten Subskription ist die verwaltete Warteschlange hingegen eine temporäre dynamische Warteschlange und wird als solche zusammen mit nicht konsumierten Nachrichten gelöscht, wenn die Verbindung aus einer der folgenden Gründe getrennt wird:

- MQCLOSE wird mit MQCO_REMOVE_SUB ausgeführt und der verwaltete Hobj wird geschlossen.
- Eine Verbindung geht an eine Anwendung verloren, die eine nicht persistente Subskription (MQSO_NON_DURABLE) verwendet.
- Eine Subskription wird entfernt, da sie abgelaufen ist und der verwaltete Hobj geschlossen wurde.

Verwaltete Subskriptionen können auch bei persistenten Subskriptionen verwendet werden. In diesem Fall möchten Sie aber vermutlich nicht konsumierte Nachrichten in der Subskribentenwarteschlange be-

lassen, sodass diese nach einer Wiederherstellung der Verbindung abgerufen werden können. Aus diesem Grund werden verwaltete Warteschlangen für persistente Subskriptionen als permanente dynamische Warteschlangen erstellt und bleiben erhalten, wenn die Verbindung der abonnierenden Anwendung zum Warteschlangenmanager getrennt wird.

Sie können einen Ablauf für Ihre Subskription festlegen, wenn Sie eine permanente dynamische verwaltete Warteschlange verwenden möchten, damit die Warteschlange zwar auch nach der Unterbrechung der Verbindung noch vorhanden ist, sie aber nicht unbegrenzt bestehen bleibt.

Wenn Sie eine verwaltete Warteschlange löschen, erhalten Sie eine Fehlernachricht.

An die Namen der verwalteten Warteschlangen wird am Ende eine Zeitmarke angefügt. Auf diese Weise erhält jede Warteschlange einen eindeutigen Namen.

Subskriptionspermanenz

Subskriptionen können persistent oder nicht persistent sein. Die Persistenz einer Subskription bestimmt, was mit der Subskription geschieht, wenn die abonnierende Anwendung vom Warteschlangenmanager getrennt wird.

Permanente Subskriptionen

Permanente Subskriptionen bleiben erhalten, wenn die Verbindung der subscribierenden Anwendung zum Warteschlangenmanager geschlossen wird. Wenn eine Subskription permanent ist und die Verbindung der subscribierenden Anwendung getrennt wird, bleibt die Subskription erhalten und kann von der subscribierenden Anwendung verwendet werden, wenn sie erneut eine Verbindung herstellt und die Subskription mit dem SubName anfordert, der bei der Erstellung der Subskription zurückgegeben wurde.

Bei einer Subskription ist ein Subskriptionsname (SubName) erforderlich. Subskriptionsnamen müssen innerhalb eines Warteschlangenmanagers eindeutig sein, sodass diese eine Subskription zweifelsfrei identifizieren. Dieses Identifikationsmittel ist erforderlich, wenn Sie eine Subskription angeben, die wiederaufgenommen werden soll, wenn Sie entweder die Kennung absichtlich für die Subskription geschlossen haben (mit der Option MQCO_KEEP_SUB) oder die Verbindung zum Warteschlangenmanager getrennt wurden. Eine vorhandene Subskription können Sie mit dem Aufruf MQSUB mit der Option MQSO_RESUME wiederaufnehmen. Die Subskriptionsnamen können mit dem Befehl DISPLAY SBSTATUS mit SUBTYPE ALL oder ADMIN angezeigt werden.

Wird eine persistente Subskription nicht mehr von der abonnierenden Anwendung benötigt, so kann diese mit dem Funktionsaufruf MQCLOSE und der Option MQCO_REMOVE_SUB bzw. manuell mit dem WebSphere MQ-Scriptbefehl DELETE SUB gelöscht werden.

Ob permanente Subskriptionen für ein Thema vorgenommen werden können, kann mit dem Themenattribut **DURSUB** gesteuert werden.

Während der Rückgabe eines MQSUB-Aufrufs mit der Option MQSO_RESUME wird der Subskriptionsablauf auf die ursprüngliche Ablaufzeit der Subskription gesetzt, nicht auf die verbleibende Ablaufzeit.

Ein Warteschlangenmanager sendet einer persistenten Subskription weiterhin Veröffentlichungen, selbst wenn zur Subskribentenanwendung keine Verbindung mehr besteht. Dies führt in der Subskribentenwarteschlange zu einem Rückstau. Die einfachste Methode, dies zu vermeiden, ist, wann immer möglich, die Verwendung von nicht persistenten Subskriptionen. Ist eine persistente Subskription absolut erforderlich, so lässt sich ein solcher Rückstau aber auch vermeiden, wenn die Subskription mit der Option Ständige Veröffentlichungen erstellt wird. Ein Subskribent kann dann mit dem Aufruf MQSUBRQ steuern, wann er Veröffentlichungen empfängt.

Nicht persistente Subskriptionen

Nicht persistente Subskriptionen bleiben nur so lange bestehen, wie die Verbindung einer Subskribentenanwendung zu einem Warteschlangenmanager geöffnet bleibt. Die Subskription wird entfernt, wenn die abonnierende Anwendung absichtlich oder durch eine Verbindungsunterbrechung vom Warteschlangenmanager getrennt wird. Beim Schließen der Verbindung werden die Informationen zur Subskription vom Warteschlangenmanager gelöscht. Die Subskription tritt dann auch bei der Anzeige der Subskriptionen

mit dem Befehl DISPLAY SBSTATUS nicht mehr in Erscheinung. Nach der Beendigung der Subskription werden keine weiteren Nachrichten mehr in die Subskribentenwarteschlange gestellt.

Was bei nicht persistenten Subskriptionen mit noch nicht konsumierten Veröffentlichungen in der Subskribentenwarteschlange geschieht, wird wie folgt bestimmt:

- Wenn eine abonnie-rende Anwendung ein verwaltetes Ziel verwendet, werden alle noch nicht konsumierten Veröffentlichungen automatisch entfernt.
- Hat die abonnie-rende Anwendung bei der Subskription eine Kennung ihrer eigenen Subskribentenwarteschlange bereitgestellt, so werden nicht konsumierte Veröffentlichungen nicht automatisch entfernt. In diesem Fall ist die Anwendung selbst dafür verantwortlich, die Warteschlange zu bereinigen. Falls die Warteschlange auch von anderen Subskribenten oder von Punkt-zu-Punkt-Anwendungen genutzt wird, darf die Warteschlange vermutlich nicht vollständig geleert werden.

Obwohl dies für nicht permanente Subskriptionen nicht erforderlich ist, wird ein Subskriptionsname, sofern angegeben, vom Warteschlangenmanager verwendet. Subskriptionsnamen müssen innerhalb eines Warteschlangenmanagers eindeutig sein, sodass diese eine Subskription zweifelsfrei identifizieren.

Auswahlzeichenfolgen

Eine *Auswahlzeichenfolge* ist ein Ausdruck, der auf eine Veröffentlichung angewendet wird, um festzustellen, ob sie mit einer Subskription übereinstimmt. Auswahlzeichenfolgen können Platzhalterzeichen einschließen.

Beim Einrichten einer Subskription können Sie neben einem Thema auch eine Auswahlzeichenfolge angeben, um Veröffentlichungen anhand ihrer Nachrichteneigenschaften auszuwählen.

Themen

Ein Thema ist der Betreff der Informationen, die in einer Publish/Subscribe-Nachricht veröffentlicht werden.

Nachrichten in Punkt-zu-Punkt-Systemen werden an eine bestimmte Zieladresse gesendet. Nachrichten in betreffbasierten Publish/Subscribe-Systemen werden auf Basis des Betreffs, der den Inhalt der Nachricht beschreibt, an Subskribenten gesendet. Bei inhaltsbasierten Systemen werden Nachrichten auf Basis des Nachrichteninhalts an Subskribenten gesendet.

Das Publish/Subscribe-System von IBM WebSphere MQ ist ein betreffbasiertes Publish/Subscribe-System. Ein Publisher erstellt eine Nachricht und veröffentlicht sie mit einer Themenzeichenfolge, die dem Betreff der Veröffentlichung am besten entspricht. Um Veröffentlichungen zu empfangen, erstellt eine Subskribent eine Subskription mit einer Zeichenfolge mit Mustererkennung, um Veröffentlichungsthemen auszuwählen. Der Warteschlangenmanager liefert Veröffentlichungen an Subskribenten, deren Subskriptionen dem Veröffentlichungsthema entsprechen und für den Empfang von Veröffentlichungen autorisiert sind. Der Artikel „Themenzeichenfolgen“ auf Seite 29 beschreibt die Syntax von Themenzeichenfolgen, die den Betreff einer Veröffentlichung angeben. Subskribenten erstellen zudem auch Themazeichenfolgen, um die zu empfangenden Themen auszuwählen. Die Themazeichenfolgen, die Subskribenten erstellen, können eines von zwei Platzhalterschemas zur Musterübereinstimmung mit den Themazeichenfolgen in Veröffentlichungen enthalten. Informationen zur Musterübereinstimmung finden Sie in „Platzhalterschemas“ auf Seite 30.

Beim betreffbasierten Publish/Subscribe sind Publisher oder Administratoren für die Klassifizierung von Betreffs in Artikeln verantwortlich. Normalerweise werden Betreffs hierarchisch in Themenstrukturen organisiert. Mit dem Zeichen ' / ' werden in der Themazeichenfolge Unterthemen erstellt. Beispiele zu Themenstrukturen finden Sie in „Themenstrukturen“ auf Seite 36. Themen sind Knoten in einer Themenstruktur. Themen können Blattknoten ohne weitere Unterthemen oder Zwischenknoten mit Unterthemen sein.

Parallel zur Verwaltung von Betreffs in hierarchischen Themenstrukturen können Sie administrativen Themenobjekten Themen zuordnen. Sie weisen einem Thema Attribute zu, z. B., ob das Thema in einem Cluster verteilt wird, indem Sie es einem administrativen Themenobjekt zuordnen. Die Zuordnung findet über die Benennung des Themas unter Verwendung des Attributs TOPICSTR des administrativen Themenobjekts statt. Wenn Sie ein administratives Themenobjekt nicht explizit einem Thema zuordnen,

übernimmt das Thema die Attribute seines nächsten Vorgängers in der Themenstruktur, den Sie einem administrativen Themenobjekt *zugeordnet* haben. Wenn Sie überhaupt keine übergeordneten Themen definiert haben, übernimmt diese aus SYSTEM.BASE.TOPIC. Eine Beschreibung der administrativen Themenobjekte finden Sie in „Verwaltungsthemenobjekte“ auf Seite 39.

Anmerkung: Auch wenn Sie alle Attribute eines Themas von SYSTEM.BASE.TOPIC: Definieren Sie ein Stammthema für Ihre Themen, die direkt von SYSTEM.BASE.TOPIC. Im Themenbereich 'US-Länder' USA/Alabama, USA/Alaska usw. ist USA das Stammthema. Hauptzweck eines Stammthemas ist es, getrennte, nicht überlappende Themenbereiche zu erstellen, um zu verhindern, dass Veröffentlichungen den falschen Subskriptionen entsprechen. Zudem können Sie auf diese Weise die Attribute Ihres Stammthemas ändern, um den gesamten Themenbereich entsprechend zu ändern. Sie können beispielsweise den Namen für das Attribut **CLUSTER** festlegen.

Wenn Sie ein Thema als Publisher oder Subskribent referenzieren, können Sie eine Themenzeichenfolge angeben, die auf ein Themenobjekt verweist, oder Sie können beides tun. In diesem Fall definiert die von Ihnen angegebene Themenzeichenfolge ein Unterthema des Themenobjekts. Der Warteschlangenmanager identifiziert das Thema, indem er die Themazeichenfolge an das Präfix der im Themaobjekt genannten Themazeichenfolge anhängt. Dabei fügt er ein weiteres ' / '-Zeichen zwischen die beiden Themazeichenfolgen ein, z. B. *Themazeichenfolge/Objektzeichenfolge*. Unter „Themenzeichenfolgen kombinieren“ auf Seite 34 finden Sie weitere Informationen hierzu. Die daraus resultierende Themazeichenfolge dient zur Ermittlung des Themas und dessen Zuordnung zu einem administrativen Themenobjekt. Das administrative Themenobjekt ist nicht zwingend dasselbe Themenobjekt wie das Themenobjekt, das dem Masterthema entspricht.

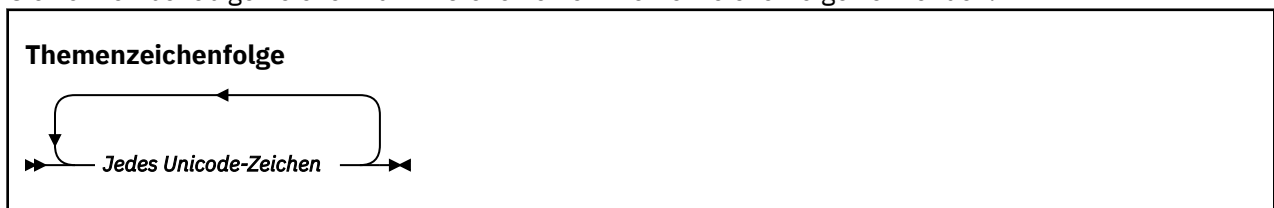
Beim inhaltsbasierten Publish/Subscribe definieren Sie, welche Nachrichten Sie empfangen möchten, indem Sie Auswahlzeichenfolgen angeben, die den Inhalt jeder Nachricht durchsuchen. WebSphere MQ stellt eine temporäre Form von inhaltsbasiertem Publish/Subscribe unter Verwendung von Nachrichtenselektoren bereit, die Nachrichteneigenschaften anstelle des vollständigen Inhalts der Nachricht scannen (siehe Selektoren). Normalerweise werden Nachrichtenselektoren so eingesetzt, dass zunächst ein Thema abonniert und die Auswahl dann mit einer numerischen Eigenschaft klassifiziert wird. Über den Selektor können Sie anzugeben, dass Sie nur an Werten in einem bestimmten Bereich interessiert sind. Dies ist mit Zeichen oder themenbasierten Platzhaltern nicht möglich. Wenn Sie auf der Basis des vollständigen Inhalts der Nachricht filtern müssen, müssen Sie WebSphere Message Broker verwenden.

Themenzeichenfolgen

Kennsatzinformationen, die Sie mithilfe einer Themenzeichenfolge als ein Thema veröffentlichen. Abonnieren Sie Themengruppen, indem Sie entweder zeichenbasierte oder themenbasierte Platzhalterthemenzeichenfolgen verwenden.

Themen

Eine *Themenzeichenfolge* ist eine Zeichenfolge, die das Thema einer Publish/Subscribe-Nachricht angibt. Sie können beliebige Zeichen zum Erstellen einer Themenzeichenfolge verwenden.



Im Publish/Subscribe von Version 7 haben drei Zeichen eine besondere Bedeutung. Sie sind zwar überall in einer Themenzeichenfolge zulässig, sollten aber mit Vorsicht verwendet werden. Die Verwendung der Sonderzeichen wird im Abschnitt „Themenbasiertes Platzhalterschema“ auf Seite 30 erläutert.

Schrägstrich (/)

Dies ist das Trennzeichen für Themenebenen. Verwenden Sie das Zeichen ' / ', um eine Themenstruktur für das Thema zu erstellen.

Vermeiden Sie, wenn es möglich ist, leere Themenebenen (' / / '). Denn diese entsprechen Knoten in der Themenhierarchie ohne Themenzeichenfolge. Ein führender oder abschließender ' / ' in ei-

ner Themenzeichenfolge entspricht einem führenden oder abschließenden leeren Knoten und sollte ebenfalls vermieden werden.

Das Hashzeichen (#)

Wird in Kombination mit '/' verwendet, um in Subskriptionen ein Platzhalterzeichen für mehrere Ebenen zu erstellen. Geben Sie acht bei der Verwendung einer Kombination aus '#' und '/' in Themenzeichenfolgen, die zum Benennen von veröffentlichten Themen dienen. „[Beispiele für Themenzeichenfolgen](#)“ auf Seite 30 zeigt eine sinnvolle Verwendung von '#'.

Die Zeichenfolgen '.../#/...', '#/...' und '.../ #' haben in Subskriptionsthemenzeichenfolgen eine besondere Bedeutung. Die Zeichenfolgen stimmen mit allen Themen auf einer oder mehreren Ebenen in der Themenhierarchie überein. Wenn Sie ein Thema mit einer dieser Folgen erstellt haben, könnten Sie dieses Thema deshalb nicht abonnieren, ohne auch alle Themen auf mehreren Ebenen in der Themenhierarchie zu abonnieren.

Pluszeichen (+)

Wird in Kombination mit '/' verwendet, um in Subskriptionen ein Platzhalterzeichen für eine einzelne Ebene zu erstellen. Geben Sie acht bei der Verwendung einer Kombination aus '+' und '/' in Themenzeichenfolgen, die zum Benennen von veröffentlichten Themen dienen.

Die Zeichenfolgen '.../+/...', '+/...' und '.../ +' haben in Subskriptionsthemenzeichenfolgen eine besondere Bedeutung. Die Zeichenfolgen stimmen mit allen Themen auf einer einzelnen Ebene in der Themenhierarchie überein. Wenn Sie ein Thema mit einer dieser Folgen erstellt haben, könnten Sie dieses Thema deshalb nicht abonnieren, ohne auch alle Themen auf einer einzelnen Ebene in der Themenhierarchie zu abonnieren.

Beispiele für Themenzeichenfolgen

```
IBM/Business Area#/Results
IBM/Diversity/%African American
```

Platzhalterschemas

Es gibt zwei Platzhalterschemas, die zum Abonnieren mehrerer Themen verwendet werden. Die Auswahl des Schemas ist eine Subskriptionsoption.

MQSO_WILDCARD_TOPIC

Wählen Sie zu abonnierende Themen mithilfe des themenbasierten Platzhalterschemas aus.

Dies ist der Standard, wenn kein Platzhalterschema explizit ausgewählt ist.

MQSO_WILDCARD_CHAR

Wählen Sie zu abonnierende Themen mithilfe des zeichenbasierten Platzhalterschemas aus.

Legen Sie eins der Schemas fest, indem Sie den Parameter **wschema** im Befehl DEFINE SUB angeben. Weitere Informationen finden Sie unter [DEFINE SUB](#).

Anmerkung: Subskriptionen, die vor WebSphere MQ Version 7.0 erstellt wurden, verwenden immer das zeichenbasierte Platzhalterschema.

Beispiele

```
IBM/+/Results
#/Results
IBM/Software/Results
IBM/*ware/Results
```

Themenbasiertes Platzhalterschema

Themenbasierte Platzhalterzeichen ermöglichen es Subskribenten, mehrere Themen gleichzeitig zu abonnieren.

Themenbasierte Platzhalter sind eine leistungsfähige Funktion des Themensystems in WebSphere MQ Publish/Subscribe. Die Platzhalter für mehrere Ebenen bzw. für nur eine Ebene können für Subskriptionen, jedoch nicht innerhalb eines Themas durch den Publisher einer Nachricht verwendet werden.

Mithilfe des themenbasierten Platzhalterschemas können Sie Veröffentlichungen gruppiert nach Themen-ebene auswählen. Sie können für *jede Ebene in der Themenhierarchie* auswählen, ob die Zeichenfolge in der Subskription für die betreffende Themenebene mit der Zeichenfolge in der Veröffentlichung genau übereinstimmen muss oder nicht. Beispiel: Die Subskription IBM/+/Results wählt alle Themen aus,

```
IBM/Software/Results
IBM/Services/Results
IBM/Hardware/Results
```

Es gibt zwei Typen von Platzhalterzeichen:

Platzhalterzeichen für mehrere Ebenen

- Das Platzhalterzeichen für mehrere Ebenen wird in Subskriptionen verwendet. Bei Verwendung in einer Veröffentlichung wird es als Literal behandelt.
- Das Platzhalterzeichen '#' für mehrere Ebenen deckt alle Ebenen innerhalb eines Themas ab. Für das Themenstrukturbeispiel bedeutet dies, dass Sie bei einer Subskription von 'USA/Alaska/#' Nachrichten zu den Themen 'USA/Alaska' und 'USA/Alaska/Juneau' empfangen.
- Das Platzhalterzeichen für mehrere Ebenen kann auf keine oder mehrere Ebenen zutreffen. Deshalb kann 'USA/#' auch die einzelne Ebene 'USA' abdecken, wobei '#' auf keine Ebene zutrifft. Das Trennzeichen für Themenebenen ist in diesem Kontext bedeutungslos, weil es keine zu trennenden Ebenen gibt.
- Das Platzhalterzeichen für mehrere Ebenen ist nur wirksam, wenn es allein steht oder neben dem Trennzeichen für Themenebenen. Deshalb sind '#' und 'USA/#' gültige Themen, wobei das Zeichen '#' als Platzhalter behandelt wird. Aber obwohl 'USA#' ebenfalls eine gültige Themenzeichenfolge ist, wird das Zeichen '#' nicht als Platzhalter angesehen und hat deshalb keine besondere Bedeutung. Weitere Informationen finden Sie in [„Wenn themenbasierte Platzhalterzeichen unwirksam sind“](#) auf Seite 33.

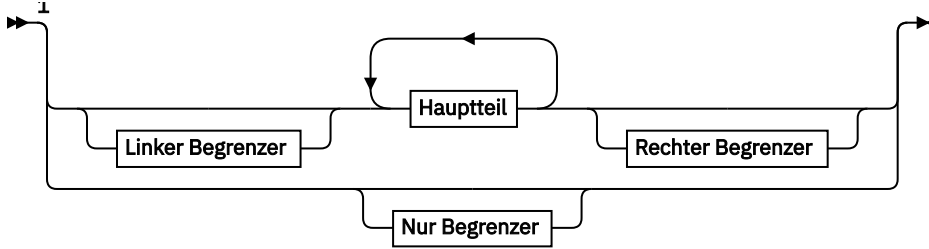
Platzhalterzeichen für einzelne Ebenen

- Das Platzhalterzeichen für einzelne Ebenen wird in Subskriptionen verwendet. Bei Verwendung in einer Veröffentlichung wird es als Literal behandelt.
- Das Platzhalterzeichen '+' für einzelne Ebenen trifft nur auf eine einzige Themenebene zu. Beispielsweise deckt 'USA/+' die Ebene 'USA/Alabama' ab, aber nicht die Ebene 'USA/Alabama/Auburn'. Da das Platzhalterzeichen für einzelne Ebenen nur für eine einzige Ebene zutrifft, deckt 'USA/+' auch nicht die Ebene 'USA' ab.
- Das Platzhalterzeichen für einzelne Ebenen kann auf jeder Ebene in der Themenstruktur und in Verbindung mit dem Platzhalterzeichen für mehrere Ebenen verwendet werden. Es muss neben dem Trennzeichen für Themenebenen angegeben werden, außer wenn es allein angegeben wird. Deshalb sind '+' und 'USA/+' gültige Themen, wobei das Zeichen '+' als Platzhalter behandelt wird. Aber obwohl 'USA+' ebenfalls eine gültige Themenzeichenfolge ist, wird das Zeichen '+' nicht als Platzhalter angesehen und hat deshalb keine besondere Bedeutung. Weitere Informationen finden Sie in [„Wenn themenbasierte Platzhalterzeichen unwirksam sind“](#) auf Seite 33.

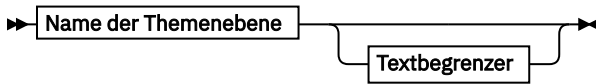
Die Syntax für das themenbasierte Platzhalterschema umfasst keine Escapezeichen. Ob '#' und '+' als Platzhalterzeichen behandelt werden oder nicht, hängt von ihrem Kontext ab. Weitere Informationen finden Sie in [„Wenn themenbasierte Platzhalterzeichen unwirksam sind“](#) auf Seite 33.

Anmerkung: Anfang und Ende einer Themenzeichenfolge werden auf besondere Weise behandelt. Wenn Sie '\$' verwenden, um das Ende der Zeichenfolge anzugeben, ist '\$#/...' ein Platzhalter für mehrere Ebenen und '\$/#/...' ist ein leerer Knoten im Stammverzeichnis, gefolgt von einem Platzhalter für mehrere Ebenen.

Themenbasierte Platzhalterzeichenfolge



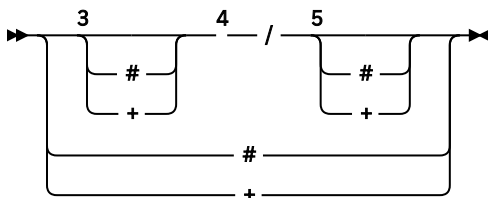
Hauptteil



Name der Themenebene



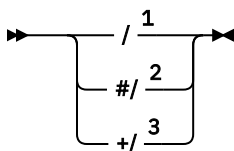
Nur Begrenzer



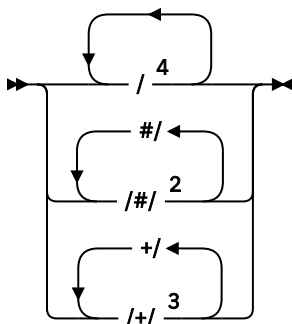
Anmerkungen:

- 1 Eine Nullzeichenfolge oder leere Zeichenfolge als Themenzeichenfolge ist ungültig.
- 2 Es wird empfohlen, keine *, ?, % in Ebenennamenzeichenfolgen zu verwenden, um die Kompatibilität zwischen zeichenbasierten und themenbasierten Platzhalterschemas zu gewährleisten.
- 3 Diese Fälle entsprechen dem Muster von *Linker Begrenzer*.
- 4 / ohne Platzhalterzeichen trifft auf ein einzelnes leeres Thema zu.
- 5 Diese Fälle entsprechen dem Muster von *Rechter Begrenzer*.
- 6 Trifft auf jedes Thema zu.
- 7 Trifft auf jedes Thema zu, für das es nur eine Ebene gibt.

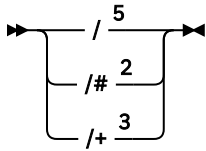
Linker Begrenzer



Textbegrenzer



Rechter Begrenzer



Anmerkungen:

- 1 Die Themenzeichenfolge beginnt mit einem leeren Thema.
- 2 Trifft auf keine oder mehrere Ebenen zu. Mehrere übereinstimmende Zeichenfolgen für mehrere Ebenen haben dieselbe Wirkung wie eine einzige übereinstimmende Zeichenfolge für mehrere Ebenen.
- 3 Trifft auf genau eine Ebene zu.
- 4 // ist ein leeres Thema - ein Themenobjekt ohne Themenzeichenfolge.
- 5 Die Themenzeichenfolge endet mit einem leeren Thema.

Wenn themenbasierte Platzhalterzeichen unwirksam sind

Die Platzhalterzeichen '+' und '#' haben keine besondere Bedeutung, wenn sie in einer Themenebene mit anderen Zeichen (einschließlich sich selbst) gemischt werden.

Dies bedeutet, dass Themen, die in einer Themenebene das Zeichen '+' oder '#' zusammen mit anderen Zeichen enthalten, veröffentlicht werden können.

Betrachten Sie beispielsweise die beiden folgenden Themen:

1. level0/level1/+/level4/#
2. level0/level1/#+/level4/level#

Im ersten Beispiel werden die Zeichen '+' und '#' als Platzhalterzeichen behandelt und sind deshalb in einer Themenzeichenfolge, für die Veröffentlichungen erfolgen sollen, nicht gültig, wohingegen sie in einer Subskription gültig sind.

Im zweiten Beispiel werden die Zeichen '+' und '#' nicht als Platzhalterzeichen behandelt, weshalb für die Themenzeichenfolge sowohl Veröffentlichungen als auch Subskriptionen möglich sind.

Beispiele

```
IBM/+/Results
#/Results
IBM/Software/Results
```

Zeichenbasiertes Platzhalterschema

Mit dem zeichenbasierten Platzhalterschema können Sie Themen auf Grundlage des Abgleichs von konventionellen Zeichen auswählen.

Mit der Zeichenfolge '*' können Sie alle Themen auf mehreren Ebenen in einer Themenhierarchie auswählen. Die Verwendung von '*' im zeichenbasierten Platzhalterschema entspricht der Verwendung der themenbasierten Platzhalterzeichenfolge '#'

'x*/y' entspricht 'x#/y' im themenbasierten Schema und wählt alle Themen in der Themenhierarchie zwischen den Ebenen 'x' und 'y' aus, wobei 'x' und 'y' Themennamen sind, die nicht in der Gruppe von Ebenen enthalten sind, die vom Platzhalterzeichen zurückgegeben werden.

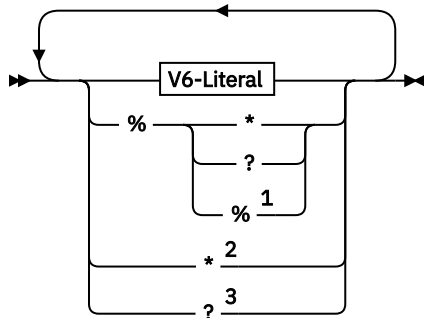
'/+' im themenbasierten Schema hat keine exakte Entsprechung im zeichenbasierten Schema.

'IBM*/Results' würde auch 'IBM/Patents/Software/Results' auswählen. Nur wenn die Gruppe der Themennamen in jeder Ebene der Hierarchie eindeutig ist, können Sie in jedem Fall Abfragen mit den beiden Schemas erstellen, die identische Übereinstimmungen ergeben.

Bei einer allgemeinen Verwendung haben '*' und '?' im zeichenbasierten Schema keine Entsprechungen im themenbasierten Schema. Das themenbasierte Schema führt keinen unvollständigen Abgleich mithilfe von Platzhaltern aus. Die zeichenbasierte Platzhaltersubskription 'IBM/*ware/Results' hat keine themenbasierte Entsprechung.

Anmerkung: Abgleiche mithilfe von Subskriptionen mit Platzhalterzeichen sind langsamer als Abgleiche mithilfe von themenbasierten Subskriptionen.

Zeichenbasierte Platzhalterzeichen



V6-Literal

► Alle Unicode-Zeichen außer *,? und % ◄

Anmerkungen:

- 1 Bedeutet "Folgendem Zeichen Escapezeichen voranstellen", damit es als Literal verwendet wird. Auf '%' muss entweder '*', '?' oder '%' folgen. Weitere Informationen finden Sie in „Beispiele für Themenzeichenfolgen“ auf Seite 30.
- 2 Bedeutet "Übereinstimmung mit null oder mehr Zeichen" in einer Subskription.
- 3 Bedeutet "Übereinstimmung mit genau einem Zeichen" in einer Subskription.

Beispiele

```
IBM/*/Results
IBM/*ware/Results
```

Themenzeichenfolgen kombinieren

Wenn Sie Subskriptionen erstellen oder Themen öffnen, um Nachrichten für sie veröffentlichen zu können, kann die Themenzeichenfolge gebildet werden, indem zwei separate Unterthemenzeichenfolgen (oder "Unterthemen") kombiniert werden. Das eine Unterthema wird von der Anwendung oder vom Verwaltungsbefehl als Themenzeichenfolge bereitgestellt und das andere ist die Themenzeichenfolge, die einem Themenobjekt zugeordnet ist. Sie können eines der Unterthemen als Themenzeichenfolge verwenden oder die Unterthemen kombinieren, um einen neuen Themennamen zu bilden.

Wenn Sie beispielsweise eine Subskription mit dem MQSC-Befehl **DEFINE SUB** definieren, kann der Befehl entweder **TOPICSTR** (Themenzeichenfolge) oder **TOPICOBJ** (Themenobjekt) als Attribut oder beides zusammen verwenden. Wird nur **TOPICOBJ** angegeben, wird die Themenzeichenfolge, die diesem Themenobjekt zugeordnet ist, als Themenzeichenfolge verwendet. Wird nur **TOPICSTR** angegeben, wird eben diese als Themenzeichenfolge verwendet. Wenn beide angegeben werden, werden sie zu einer einzelnen Themenzeichenfolge im Format **TOPICOBJ/TOPICSTR** verknüpft, wobei die konfigurierte Themenzeichenfolge **TOPICOBJ** immer an erster Stelle steht und die beiden Teile der Zeichenfolge immer durch ein "/"-Zeichen getrennt sind.

Entsprechend wird in einem MQI-Programm von MQOPEN der vollständige Themename erstellt. Er besteht aus zwei Feldern, die in Publish/Subscribe-MQI-Aufrufen verwendet werden, in der aufgeführten Reihenfolge:

1. Das Attribut **TOPICSTR** des Themenobjekts, das im Feld **ObjectName** benannt ist.

2. Der Parameter **ObjectString**, der das von der Anwendung bereitgestellte Unterthema definiert.

Die daraus resultierende Themenzeichenfolge wird im Parameter **ResObjectString** zurückgegeben.

Diese Felder gelten als vorhanden, wenn das erste Zeichen jedes Felds kein Leerzeichen oder Nullzeichen ist und die Feldlänge größer als null ist. Wenn nur eines der Felder vorhanden ist, wird es unverändert als Themenname verwendet. Wenn keines der beiden Felder einen Wert enthält, schlägt der Aufruf mit dem Ursachencode MQRC_UNKNOWN_OBJECT_NAME oder MQRC_TOPIC_STRING_ERROR fehl, falls der vollständige Themenname ungültig ist.

Wenn beide Felder vorhanden sind, wird das Zeichen "/" zwischen die beiden Elemente des entstandenen kombinierten Themennamens eingefügt.

Tabelle 2 auf Seite 35 zeigt Beispiele für die Verkettung von Themenzeichenfolgen:

<i>Tabelle 2. Beispiele für die Verkettung von Themenzeichenfolgen</i>			
TOPICSTR des Themenobjekts	Themenzeichenfolge, die von der Anwendung oder dem Befehl DEFINE SUB bereitgestellt wird.	Vollständiger Themenname	Kommentar
Fußball/Ergebnisse	' '	Fußball/Ergebnisse	Es wird nur TOPICSTR des Themenobjekts verwendet.
' '	Fußball/Ergebnisse	Fußball/Ergebnisse	Es wird nur ObjectString/TOPICSTR verwendet.
Fußball	Ergebnisse	Fußball/Ergebnisse	Am Verkettungspunkt wird das Zeichen "/" hinzugefügt.
Fußball	/Ergebnisse	Fußball//Ergebnisse	Zwischen den beiden Zeichenfolgen entsteht ein "leerer Knoten". Dies ist anders als bei "Fußball/Ergebnisse".
/Fußball	Ergebnisse	/Fußball/Ergebnisse	Das Thema beginnt mit einem "leeren Knoten". Dies ist anders als bei "Fußball/Ergebnisse".

Das Zeichen "/" wird als Sonderzeichen betrachtet, das Struktur für den vollständigen Themennamen in „Themenstrukturen“ auf Seite 36 bereitstellt. Das Zeichen "/" darf zu keinem anderen Zweck verwendet werden, da die Themenstruktur betroffen ist. Das Thema "/Football" entspricht nicht dem Thema "Football".

Anmerkung: Wenn Sie beim Erstellen einer Subskription ein Themenobjekt verwenden, wird der Wert der Themenzeichenfolge des Themenobjekts zum Zeitpunkt der Definition in der Subskription festgelegt. Nachfolgende Änderungen des Themenobjekts haben keine Auswirkung auf die Themenzeichenfolge, für die die Subskription definiert ist.

Platzhalterzeichen in Themenzeichenfolgen

Die folgenden Platzhalterzeichen sind Sonderzeichen:

- Pluszeichen (+)
- Nummernzeichen (#)
- Stern (*)

- Fragezeichen (?)

Platzhalterzeichen haben nur eine besondere Bedeutung, wenn sie von einer Subskription verwendet werden. Diese Zeichen werden nicht als ungültig betrachtet, wenn sie woanders verwendet werden, aber Sie müssen sehr genau wissen, wie sie verwendet werden, und verzichten eventuell lieber darauf, diese Zeichen bei der Veröffentlichung oder Definition von Themenobjekten in Ihren Themenzeichenfolgen zu verwenden.

Bei Veröffentlichungen für eine Themenzeichenfolge mit # oder + in Kombination mit anderen Zeichen (einschließlich sich selbst) innerhalb einer Themenebene kann die Themenzeichenfolge mit beiden Platzhalterschemas subskribiert werden.

Bei Veröffentlichungen für eine Themenzeichenfolge mit # oder + als dem einzigen Zeichen zwischen zwei /-Zeichen kann die Themenzeichenfolge nicht explizit von einer Anwendung subskribiert werden, die das Platzhalterschema MQSO_WILDCARD_TOPIC verwendet. Diese Situation führt dazu, dass die Anwendung mehr Veröffentlichungen abrufen als erwartet.

Sie sollten kein Platzhalterzeichen in der Themenzeichenfolge eines definierten Themenobjekts verwenden. Wenn doch, wird das Zeichen bei Verwendung des Objekts durch eine Publisher als Literal und bei Verwendung durch eine Subskription als Platzhalterzeichen behandelt. Dies kann zu Unklarheiten führen.

Mustercodeausschnitt

Dieser Codeausschnitt, der dem Beispielprogramm [Beispiel 2: Publisher eines variablen Themas](#) entnommen wurde, kombiniert ein Themenobjekt mit einer variablen Themenzeichenfolge:

```
MQOD      td = {MQOD_DEFAULT}; /* Object Descriptor          */
td.ObjectType = MQOT_TOPIC; /* Object is a topic    */
td.Version = MQOD_VERSION_4; /* Descriptor needs to be V4 */
strncpy(td.ObjectName, topicName, MQ_TOPIC_NAME_LENGTH);
td.ObjectString.VSPtr = topicString;
td.ObjectString.VSLength = (MQLONG)strlen(topicString);
td.ResObjectString.VSPtr = resTopicStr;
td.ResObjectString.VSBufSize = sizeof(resTopicStr)-1;
MQOPEN(Hconn, &td, MQOO_OUTPUT | MQOO_FAIL_IF QUIESCING, &Hobj, &CompCode, &Reason);
```

Themenstrukturen

Jedes Thema, das Sie definieren, wird in der Themenstruktur durch ein Element oder einen Knoten dargestellt. Die Themenstruktur kann entweder leer sein, um ganz neu zu beginnen, oder Themen enthalten, die zuvor mithilfe von WebSphere MQ-Scriptbefehlen oder PCF-Befehlen definiert wurden. Sie können ein neues Thema definieren, indem Sie entweder die Befehle zum Erstellen von Themen verwenden oder das Thema zum ersten Mal in einer Veröffentlichung oder Subskription angeben.

Obwohl Sie zum Definieren der Themenzeichenfolge eines Themas eine beliebige Zeichenfolge verwenden können, sollten Sie eine Themenzeichenfolge auswählen, die in eine hierarchische Baumstruktur passt. Ein sorgfältiger Entwurf von Themenzeichenfolgen und Themenstrukturen erleichtert folgende Vorgänge:

- Abonnieren mehrerer Themen
- Erstellen von Sicherheitsrichtlinien

Obwohl Sie eine Themenstruktur auch als flache, lineare Struktur entwerfen können, ist es besser, eine Themenstruktur in einer hierarchische Struktur mit einem oder mehreren Stammthemen zu erstellen. Weitere Informationen zur Sicherheitsplanung und zu Themen finden Sie im Abschnitt [Publish/Subscribe-Sicherheit](#).

In [Abbildung 5 auf Seite 37](#) zeigt ein Beispiel für eine Themenstruktur mit einem einzigen Stammthema.

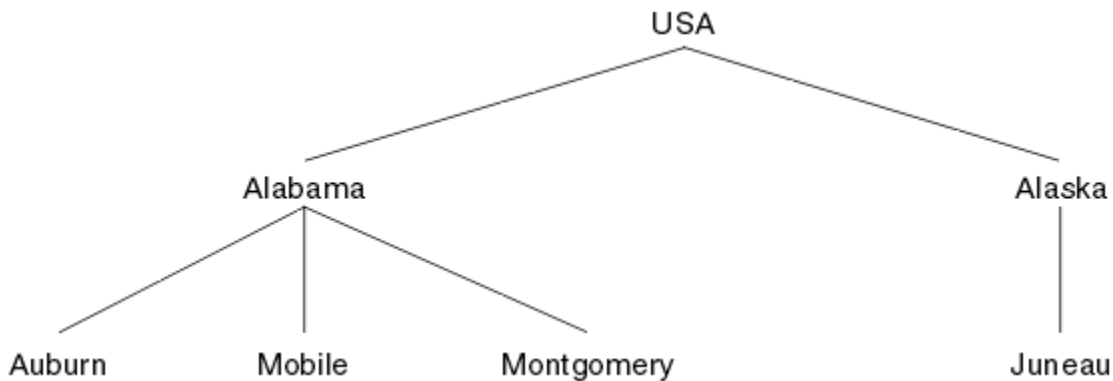


Abbildung 5. Beispiel für eine Themenstruktur

Jede Zeichenfolge in der Abbildung steht für einen Knoten in der Themenstruktur. Eine vollständige Themenzeichenfolge entsteht durch die Zusammenfassung von Knoten auf einer oder mehreren Ebenen in der Themenstruktur. Ebenen werden durch das Zeichen "/" getrennt. Eine vollständig angegebene Themenzeichenfolge hat das Format "Stamm/Ebene2/Ebene3".

Die gültigen Themen in der Themenstruktur in [Abbildung 5 auf Seite 37](#) lauten wie folgt:

"USA"
 "USA/Alabama"
 "USA/Alaska"
 "USA/Alabama/Auburn"
 "USA/Alabama/Mobile"
 "USA/Alabama/Montgomery"
 "USA/Alaska/Juneau"

Beachten Sie beim Entwerfen von Themenzeichenfolgen und Themenstrukturen, dass der Warteschlangenmanager nicht die Themenzeichenfolge selbst interpretiert oder auch nur versucht, eine Bedeutung daraus abzuleiten. Er verwendet die Themenzeichenfolge nur, um ausgewählte Nachrichten an Subskribenten des betreffenden Themas zu senden.

Für die Erstellung und den Inhalt einer Themenstruktur gelten folgende Prinzipien:

- Die Anzahl Ebenen in einer Themenstruktur ist nicht begrenzt.
- Der Länge des Namens einer Ebene in einer Themenstruktur ist nicht begrenzt.
- Es können beliebig viele Stammknoten vorhanden sein, d. h., es kann beliebig viele Themenstrukturen geben.

Reduzieren der Anzahl unerwünschter Themen in der Themenstruktur

Die Leistung eines Publish/Subscribe-Systems wird verbessert, indem die Anzahl unerwünschter Themen in der Themenstruktur reduziert wird. Was ist ein unerwünschtes Thema und wie können Sie sie entfernen?

Sie können eine große Anzahl von Themen erstellen, ohne die Leistung zu beeinträchtigen. Einige Methoden zum Verwenden von Publish/Subscribe führen jedoch dazu, dass Themenstrukturen kontinuierlich erweitert werden. Eine außergewöhnlich große Anzahl von Themen wird einmal erstellt und nie wieder verwendet. Die wachsende Anzahl von Themen kann zu einem Leistungsproblem werden.

Wie können Sie Designs vermeiden, die zu einer großen und wachsenden Zahl unerwünschter Themen führen? Was können Sie tun, um dem WS-Manager zu helfen, unerwünschte Themen aus der Themenstruktur zu entfernen?

Der WS-Manager erkennt ein unerwünschtes Thema, da es seit 30 Minuten ungenutzt ist. Der WS-Manager entfernt nicht verwendete Themen aus der Themenstruktur für Sie. Die Dauer von 30 Minuten kann durch Ändern des Warteschlangenmanagerattributs **TREELIFE** geändert werden. Sie können dem Warteschlangenmanager helfen, unerwünschte Themen zu entfernen, indem Sie sicherstellen, dass der

WS-Manager nicht mehr verwendet wird. Der Abschnitt „Was ist ein nicht verwendetes Thema?“ auf Seite 38 erläutert, was ein nicht verwendetes Thema ist.

Ein Programmierer, der jede Anwendung entwirft und insbesondere eine lange laufende Anwendung entwirft, berücksichtigt seine Ressourcennutzung: Wie viel Ressourcen benötigt das Programm, gibt es irgendwelche unbegrenzten Anforderungen, und irgendwelche Ressourcenlecks? Topics sind eine Resource, die Publish/Subscribe-Programme verwenden. Scruönisieren Sie die Verwendung von Themen wie alle anderen Ressourcen, die ein Programm verwendet.

Was ist ein nicht verwendetes Thema?

Bevor Sie definieren, was ein nicht verwendetes Thema ist, was zählt eigentlich zu einem Thema?

Wenn eine Themenzeichenfolge wie USA/Alabama/Auburn in ein Thema konvertiert wird, wird das Thema der Themenstruktur hinzugefügt. Weitere Themenknoten und die entsprechenden Themen werden in der Baumstruktur erstellt, falls erforderlich. Die Themenzeichenfolge USA/Alabama/Auburn wird in eine Baumstruktur mit drei Themen konvertiert.

- USA
- USA/Alabama
- USA/Alabama/Auburn

Um alle Themen in der Themenstruktur anzuzeigen, verwenden Sie den **runmqsc** -Befehl `DISPLAY TPSTATUS(' #') TYPE(TOPIC)`.

Ein nicht verwendetes Thema in der Themenstruktur weist die folgenden Eigenschaften auf.

Sie ist einem Themenobjekt nicht zugeordnet.

Ein Verwaltungsthemenobjekt verfügt über eine Themenzeichenfolge, die sie einem Thema zuordnet. Wenn Sie das Themenobjekt Alabama definieren und das Thema USA/Alabama, dem es zugeordnet werden soll, nicht vorhanden ist, wird das Thema aus der Themenzeichenfolge erstellt. Wenn das Thema vorhanden ist, werden das Themenobjekt und das Thema unter Verwendung der Themenzeichenfolge verknüpft.

Es hat keine ständige Veröffentlichung

Ein Thema mit einer ständigen Veröffentlichung entsteht, wenn ein Publisher eine Nachricht mit der Option MQPMO_RETAIN in ein Thema einreicht.

Verwenden Sie den **runmqsc** -Befehl `DISPLAY TPSTATUS(' USA/Alabama ') RETAINED`, um zu überprüfen, ob USA/Alabama über eine ständige Veröffentlichung verfügt. Die Antwort ist YES oder NO.

Mit dem **runmqsc** -Befehl `CLEAR TOPICSTR(' USA/Alabama ') CLTRTYPE(RETAINED)` können Sie eine ständige Veröffentlichung aus USA/Alabama entfernen.

Es hat keine untergeordneten Themen

USA/Alabama/Auburn ist ein Thema ohne untergeordnete Themen. USA/Alabama/Auburn ist das direkte untergeordnete Thema von USA/Alabama.

Zeigen Sie die direkten untergeordneten Elemente von USA/Alabama mit dem **runmqsc** -Befehl `DISPLAY TPSTATUS(' USA/Alabama/+')` an.

Es sind keine aktiven Publisher für den Knoten vorhanden.

Ein aktiver Publisher für einen Knoten ist eine Anwendung, die das Thema für die Ausgabe geöffnet hat.

Eine Anwendung öffnet beispielsweise das Themenobjekt mit dem Namen **Alabama** mit den Optionen zum Öffnen MQ00_OUTPUT.

Verwenden Sie den **runmqsc** -Befehl `DISPLAY TPSTATUS(' USA/Alabama/#') TYPE(PUB) ACT-CONN`, um aktive Publisher für USA/Alabama und alle untergeordneten Elemente anzuzeigen.

Es sind keine aktiven Subskribenten für den Knoten vorhanden.

Ein aktiver Subskribent kann entweder eine permanente Subskription oder eine Anwendung sein, die eine Subskription für ein Thema bei MQSUBregistriert und nicht geschlossen hat.

Verwenden Sie den **runmqsc** -Befehl `DISPLAY TPSTATUS('USA/Alabama') TYPE(SUB) ACT-CONN`, um aktive Subskriptionen für USA/Alabama anzuzeigen.

Verwenden Sie den **runmqsc** -Befehl `DISPLAY TPSTATUS('USA/Alabama/#') TYPE(SUB) ACT-CONN`, um aktive Subskriptionen für USA/Alabama und alle untergeordneten Elemente anzuzeigen.

Anzahl der Themen in einer Themenstruktur reduzieren

Zusammenfassend gibt es eine Reihe von Möglichkeiten, die Anzahl der Themen in einer Themenstruktur zu reduzieren.

TREELIFE ändern

Ein nicht verwendetes Thema hat standardmäßig eine Lebensdauer von 30 Minuten. Sie können die Lebensdauer eines nicht verwendeten Themas kleiner machen.

Der Befehl **runmqsc** , `ALTER QMGR TREELIFE(900)`, reduziert beispielsweise die Lebensdauer eines nicht verwendeten Themas von 30 Minuten auf 15 Minuten.

Ausnahmsweise Neustart des WS-Managers

Wenn der Warteschlangenmanager erneut gestartet wird, wird die Themenstruktur von Themenobjekten, Knoten mit ständigen Veröffentlichungen und permanenten Subskriptionen erneut initialisiert. Themen, die durch den Betrieb von Veröffentlichungs- und Subskribentenprogrammen erstellt wurden, werden eliminiert.

Verwenden Sie den **runmqsc** Befehl `DISPLAY TPSTATUS('#') TYPE(TOPIC)` regelmäßig, um alle Themen aufzulisten und zu überprüfen, ob die Anzahl zunimmt.

Als letztes Mittel, wenn das Wachstum in unerwünschten Themen die Ursache von Leistungsproblemen in der Vergangenheit ist, starten Sie den Warteschlangenmanager erneut.

Verwaltungsthemenobjekte

Mit einem Verwaltungsthemenobjekt können Sie Themen bestimmte, nicht standardmäßige Attribute zuweisen.

Abbildung 6 auf Seite 39 zeigt das übergeordnete Thema Sport an, unterteilt in mehrere Unterthemen für verschiedene Sportarten und in Form einer Themenstruktur:

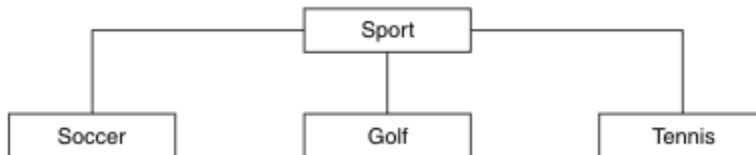


Abbildung 6. Visualisierung einer Themenstruktur

Abbildung 7 auf Seite 39 zeigt, wie die Themenstruktur weiter unterteilt werden kann, um unterschiedliche Informationsarten für jede Sportart zu trennen:



Abbildung 7. Erweiterte Themenstruktur

Zur Erstellung der gezeigten Themenstruktur müssen keine Objekte für Verwaltungsthemen definiert werden. Alle Knoten in dieser Baumstruktur sind durch eine Themenzeichenfolge definiert, die in einer Veröffentlichung oder Subskription erstellt wurde. Jedes Thema der Baumstruktur übernimmt seine Attribute von den ihm übergeordneten Elementen. Attribute werden von dem übergeordneten Themenobjekt übernommen, da standardmäßig alle Attribute auf ASPARENT eingestellt sind. In diesem Beispiel besitzt daher jedes Thema die gleichen Attribute wie das Thema Sport. Das Thema Sport seinerseits hat kein Verwaltungsthemenobjekt und übernimmt seine Attribute von `SYSTEM.BASE.TOPIC`.

Es wird dringend davon abgeraten, Berechtigungen für Benutzer, die nicht der Gruppe mqm angehören, auf Stammknotenebene der Themenstruktur (also auf der Ebene `SYSTEM.BASE.TOPIC`) zu erteilen, weil die Berechtigungen übernommen werden, aber nicht eingeschränkt werden können. Das heißt, wenn Sie Berechtigungen auf dieser Ebene erteilen, erteilen Sie damit Berechtigungen für die gesamte Struktur. Erteilen Sie die Berechtigung besser auf einer niedrigeren Themenebene in der Hierarchie.

Mit Verwaltungsthemenobjekten können bestimmte Attribute für bestimmte Knoten der Verzeichnisstruktur definiert werden. Im folgenden Beispiel wird das Verwaltungsthemenobjekt definiert, um die Eigenschaft für permanente Subskriptionen `DURSUB` des Fußballthemas auf den Wert `NO` zu setzen:

```
DEFINE TOPIC (FOOTBALL.EUROPEAN)
  TOPICSTR('Sport/Soccer')
  DURSUB(NO)
  DESCR('Administrative topic object to disallow durable subscriptions')
```

Die Verzeichnisstruktur kann nun visuell dargestellt werden als:

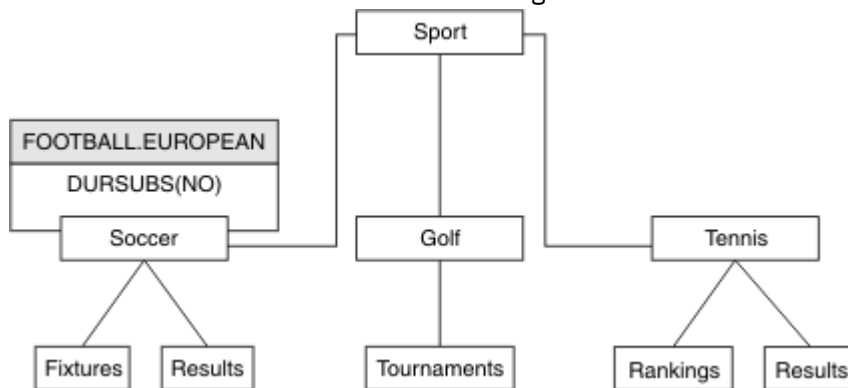


Abbildung 8. Visualisierung eines dem Thema 'Sport/Fußball' zugeordneten Verwaltungsthemenobjekts

Haben Anwendungen Themen abonniert, die in der Verzeichnisstruktur dem Thema 'Fußball' untergeordnet sind, können sie auch weiterhin die vor dem Hinzufügen des Verwaltungsthemenobjekts gültige Themenzeichenfolge verwenden. Eine Anwendung kann jetzt jedoch so geschrieben werden, dass sie den Objektnamen `FOOTBALL.EUROPEAN` anstelle der Zeichenfolge `/Sport/Soccer` subskribiert. Um beispielsweise `/Sport/Soccer/Results` zu subskribieren, kann eine Anwendung `MQSD.ObjectName` als `FOOTBALL.EUROPEAN` und `MQSD.ObjectString` als `Results` angeben.

Auf diese Weise können Sie einen Teil der Verzeichnisstruktur vor den Anwendungsentwicklern verbergen. Wenn Sie ein Verwaltungsthemenobjekt in einem bestimmten Knoten der Verzeichnisstruktur definieren, können Anwendungsentwickler somit ihre eigenen Themen diesem unterordnen. Die Entwickler müssen dabei das übergeordnete Thema, jedoch keine weiteren Knoten in der übergeordneten Baumstruktur kennen.

Attribute übernehmen

Sind in einer Themenstruktur viele Verwaltungsthemenobjekte vorhanden, übernimmt jedes Verwaltungsthemenobjekt standardmäßig die Attribute des ihm am nächsten übergeordneten Verwaltungsthemas. Das vorhergehende Beispiel wurde in [Abbildung 9 auf Seite 41](#) erweitert:

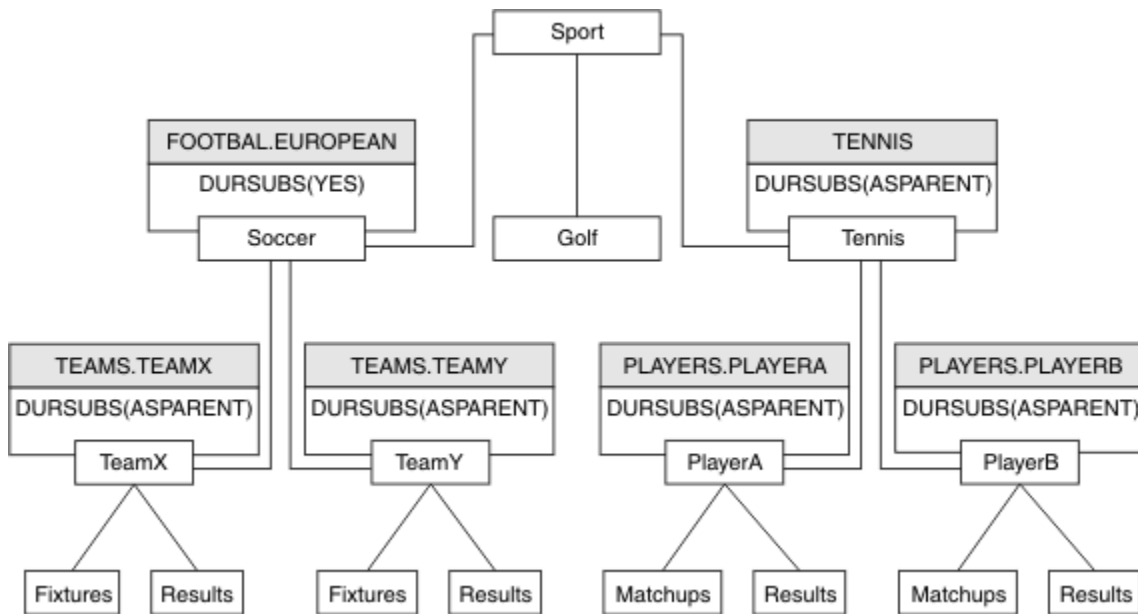


Abbildung 9. Themenstruktur mit mehreren Verwaltungsthemenobjekten

Verwenden Sie beispielsweise die Übernahme, um allen untergeordneten Themen von /Sport/Soccer die Eigenschaft zu geben, dass Subskriptionen nicht permanent sind. Ändern Sie das Attribut DURSUB von FOOTBALL . EUROPEAN in NO.

Dieses Attribut können Sie mit folgendem Befehl festlegen:

```
ALTER TOPIC(FOOTBALL.EUROPEAN) DURSUB(NO)
```

Für alle Verwaltungsthemenobjekte untergeordneter Themen von Sport/Soccer ist die Eigenschaft DURSUB auf den Standardwert ASPARENT gesetzt. Nachdem Sie den DURSUB -Eigenschaftswert von FOOTBALL . EUROPEAN in NO geändert haben, übernehmen die untergeordneten Themen von Sport/Soccer den DURSUB -Eigenschaftswert NO . Alle untergeordneten Themen von Sport/Tennis übernehmen den Wert von DURSUB aus dem Objekt SYSTEM . BASE . TOPIC . SYSTEM . BASE . TOPIC hat den Wert YES.

Der Versuch, eine permanente Subskription für das Thema Sport/Soccer/TeamX/Results zu erstellen, schlägt jetzt fehl. Der Versuch, eine permanente Subskription für Sport/Tennis/PlayerB/Results zu erstellen, wäre jedoch erfolgreich.

Verwendung von Platzhaltern mit der Eigenschaft WILDCARD steuern

Mit der MQSC-Eigenschaft **Topic WILDCARD** oder der entsprechenden PCF-Eigenschaft **Topic WildcardOperation** können Sie die Zustellung von Veröffentlichungen an Subskribentenanwendungen steuern, die Namen von Platzhalterthemenzeichenfolgen verwenden. Die Eigenschaft WILDCARD kann einen von zwei möglichen Werten haben:

WILDCARD

Aktionen von Subskriptionen mit Platzhaltern bezüglich dieses Themas.

PASSTHRU

Subskriptionen für ein Thema mit Platzhalter, das weniger spezifisch ist als die Themenzeichenfolge für dieses Themenobjekt, empfangen Veröffentlichungen zu diesem Thema und zu spezifischeren Themenzeichenfolge.

BLOCK

Subskriptionen für ein Thema mit Platzhalter, das weniger spezifisch ist als die Themenzeichenfolge für dieses Themenobjekt, empfangen keine Veröffentlichungen zu diesem Thema und zu spezifischeren Themenzeichenfolge.

Der Wert für dieses Attribut wird bei der Definition von Subskriptionen verwendet. Wenn Sie dieses Attribut ändern, ist die Gruppe von Themen, die bereits durch vorhandene Subskriptionen abgedeckt

sind, nicht durch die Änderung betroffen. Dieses Szenario gilt auch, wenn sich durch die Erstellung oder das Löschen von Themenobjekten die Topologie ändert; die Themen mit Subskriptionen, die nach der Änderung des Attributs WILDCARD erstellt wurden, werden mit der geänderten Topologie erstellt. Wenn die Themen mit den vorhandenen Subskriptionen übereinstimmen sollen, müssen Sie den Warteschlangenmanager neu starten.

Führen Sie die Schritte im Beispiel „Beispiel: Publish/Subscribe-Cluster für Sport erstellen“ auf Seite 46 aus, um die in [Abbildung 10](#) auf Seite 42 gezeigte Themenstruktur zu erstellen.

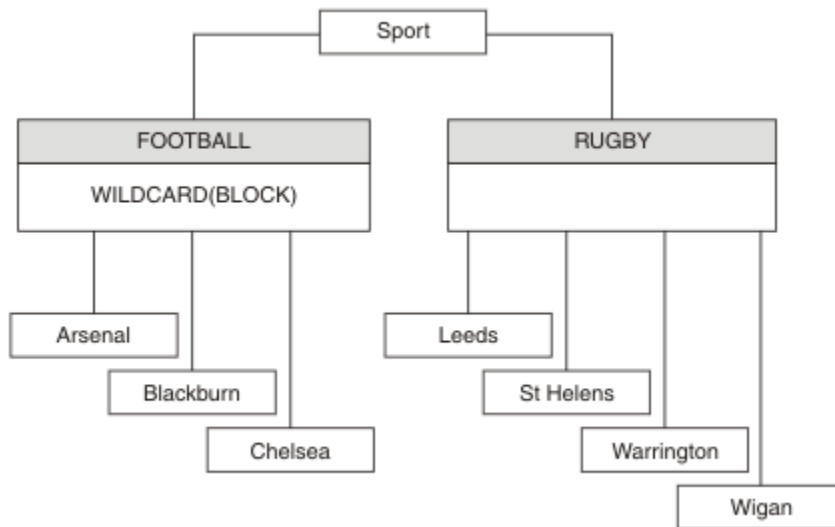


Abbildung 10. Themenstruktur, die die WILDCARD-Eigenschaft BLOCK verwendet

Ein Subskribent, der die Platzhalter-Themenzeichenfolge # verwendet, empfängt alle Veröffentlichungen für das Thema Sport und die Unterverzeichnisstruktur Sport/Rugby. Der Subskribent empfängt keine Veröffentlichungen für die untergeordnete Baumstruktur Sport/Football, weil der Wert der Eigenschaft WILDCARD des Themas Sport/Football BLOCK ist.

Die Standardeinstellung lautet PASSTHRU. Sie können den Wert der Eigenschaft WILDCARD PASSTHRU auf Knoten in der Baumstruktur Sport setzen. Wenn die Knoten nicht über den Eigenschaftswert WILDCARD verfügen BLOCK, ändert die Einstellung PASSTHRU nicht das Verhalten, das von Subskribenten für Knoten in der Sports -Baumstruktur beobachtet wird.

Erstellen Sie im Beispiel Subskriptionen, um zu sehen, wie sich die Platzhaltereinstellung auf die bereitgestellten Veröffentlichungen auswirkt (siehe dazu [Abbildung 14](#) auf Seite 48). Führen Sie den Publish-Befehl in [Abbildung 17](#) auf Seite 49 aus, um einige Veröffentlichungen zu erstellen.

```
pub QMA
```

Abbildung 11. Für QMA publizieren

Die Ergebnisse werden in [Tabelle 3](#) auf Seite 42 gezeigt. Beachten Sie, wie die Einstellung des WILDCARD-Eigenschaftswerts BLOCK verhindert, dass Subskriptionen mit Platzhaltern Veröffentlichungen für Themen empfangen, die innerhalb des Platzhalterumfangs liegen.

Subskription	Themenzeichenfolge	Empfangene Veröffentlichungen	Hinweise
SPORTS	Sports/#	Sports Sports/Rugby Sports/Rugby/Leeds	Alle Veröffentlichungen zur Unterverzeichnisstruktur 'Football', die von WILDCARD (BLOCK) auf Sports/Football blockiert werden

Tabelle 3. Auf QMA empfangene Veröffentlichungen (Forts.)

Subskription	Themenzeichenfolge	Empfangene Veröffentlichungen	Hinweise
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) on Sports/Football verhindert Platzhaltersubskription unter Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Standardwert WILDCARD für 'Sport/Rugby' verhindert keine Platzhaltersubskription auf Leeds.

Anmerkung:

Angenommen, eine Subskription hat ein Platzhalterzeichen, das einem Themenobjekt mit dem Wert BLOCK der Eigenschaft WILDCARD entspricht. Wenn die Subskription auch eine Themenzeichenfolge rechts neben dem entsprechenden Platzhalter hat, empfängt die Subskription niemals Veröffentlichungen. Die nicht geblockten Veröffentlichungen sind Veröffentlichungen für Themen, die dem geblockten Platzhalter übergeordnet sind. Veröffentlichungen für Themen, die dem Thema mit dem Eigenschaftswert BLOCK untergeordnet sind, werden vom Platzhalter geblockt. Subskriptionsthemen-Zeichenfolgen, die ein Thema rechts neben dem Platzhalter einschließen, empfangen daher niemals passende Veröffentlichungen.

Wenn der Eigenschaftswert WILDCARD auf BLOCK gesetzt wird, ist keine Subskription mithilfe einer Themenzeichenfolge möglich, die Platzhalter beinhaltet. Eine solche Subskription ist normal. Die Subskription hat ein explizites Thema, das dem Thema mit einem Themenobjekt entspricht, das den WILDCARD-Eigenschaftswert BLOCK hat. Sie verwendet Platzhalter für Themen, die dem WILDCARD-Eigenschaftswert BLOCK über- oder untergeordnet sind. Im Beispiel in [Abbildung 10](#) auf Seite 42 kann eine Subskription wie Sports/Football/# Veröffentlichungen empfangen.

Platzhalter und Cluster-Topics

Definitionen von Cluster-Topics werden an jeden Warteschlangenmanager in einem Cluster weitergegeben. Die Subskription eines Cluster-Topics auf einem Warteschlangenmanager in einem Cluster führt dazu, dass der Warteschlangenmanager Proxy-Subskriptionen erstellt. Eine Proxy-Subskription wird in jedem Warteschlangenmanager im Cluster erstellt. Subskriptionen, die Themenzeichenfolgen mit Platzhaltern verwenden, kombiniert mit Cluster-Topics, lassen das Verhalten schwer vorhersagen. Das Verhalten wird im folgenden Beispiel erläutert.

In der Clusterkonfiguration für das Beispiel „[Beispiel: Publish/Subscribe-Cluster für Sport erstellen](#)“ auf Seite 46 hat QMB dieselbe Gruppe von Subskriptionen wie QMA, aber QMB hat keine Veröffentlichungen empfangen, nachdem der Publisher in QMA veröffentlicht wurde (siehe [Abbildung 11](#) auf Seite 42). Obwohl die Themen Sports/Football und Sports/Rugby Clusterthemen sind, verweisen die in fullsubs.tst definierten Subskriptionen nicht auf ein Clusterthema. Es werden keine Proxy-Subskriptionen von QMB an QMA weitergegeben. Ohne Proxy-Subskriptionen werden keine Veröffentlichungen für QMA an QMB weitergeleitet.

Einige Subskriptionen, wie z. B. Sports/#/Leeds, scheinen möglicherweise auf ein Clusterthema zu verweisen, in diesem Fall Sports/Rugby. Die Sports/#/Leeds-Subskription wird tatsächlich in das Themenobjekt SYSTEM.BASE.TOPIC aufgelöst.

Die Regel zum Auflösen des Themenobjekts, das von einer Subskription wie Sports/#/Leeds referenziert wird, lautet wie folgt. Schneiden Sie die Themenzeichenfolge bis zum ersten Platzhalter ab. Suchen Sie links über die Themenzeichenfolge nach dem ersten Thema mit einem zugehörigen Verwaltungsthemenobjekt. Das Themenobjekt kann einen Clusternamen angeben oder ein lokales Themenobjekt definieren. Im Beispiel Sports/#/Leeds ist die Themenzeichenfolge nach dem Abschneiden Sports, die kein Themenobjekt hat, und Sports/#/Leeds übernimmt daher von SYSTEM.BASE.TOPIC, einem lokalen Themenobjekt.

Um zu sehen, wie die Subskription von Cluster-Topics die Weitergabe von Platzhaltern ändern kann, führen Sie das Stapelscript `upsubs.bat` aus. Das Script löscht die Subskriptionswarteschlangen und fügt die Subskriptionen des Cluster-Topics zu `fullsubs.tst` hinzu. Führen Sie `puba.bat` erneut aus, um einen Stapel von Veröffentlichungen zu erstellen (siehe [Abbildung 11](#) auf Seite 42).

[Tabelle 4](#) auf Seite 44 zeigt das Ergebnis, nachdem zwei neue Subskriptionen dem Warteschlangenmanager hinzugefügt wurden, in dem die Veröffentlichungen veröffentlicht wurden. Wie erwartet, empfangen die neuen Subskriptionen je eine Veröffentlichung, und die Anzahl der von den anderen Subskriptionen empfangenen Veröffentlichungen ist unverändert. Die unerwarteten Resultate treten im anderen Cluster-Warteschlangenmanager auf, siehe [Tabelle 5](#) auf Seite 44.

Subskription	Themenzeichenfolge	Empfangene Veröffentlichungen	Hinweise
SPORTS	Sports/#	Sports Sports/Rugby Sports/Rugby/Leeds	Alle Veröffentlichungen zur Unterverzeichnisstruktur 'Football', die von WILDCARD (BLOCK) auf Sports/Football blockiert werden
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) on Sports/Football verhindert Platzhaltersubskription unter Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Standardwert WILDCARD für 'Sport/Rugby' verhindert keine Platzhaltersubskription auf Leeds.
FARSENAL	Sports/Football/Arsenal	Sports/Football/Arsenal	Arsenal empfängt eine Veröffentlichung, da die Subskription keinen Platzhalter hat.
FLEEDS	Sports/Rugby/Leeds	Sports/Rugby/Leeds	Leeds würde für jedes Ereignis eine Veröffentlichung empfangen.

[Tabelle 5](#) auf Seite 44 zeigt die Resultate, nachdem die beiden neuen Subskriptionen zu QMB hinzugefügt und auf QMA veröffentlicht wurden. QMB empfing keine Veröffentlichungen ohne diese beiden neuen Subskriptionen. Wie erwartet empfangen die beiden neuen Subskriptionen Veröffentlichungen, da Sports/FootBall und Sports/Rugby Clusterthemen sind. QMB weitergeleitete Proxy-Subskriptionen für Sports/Football/Arsenal und Sports/Rugby/Leeds an QMA, die dann die Veröffentlichungen an QMBgesendet haben

Das unerwartete Ergebnis ist, dass die beiden Subskriptionen Sports/# und Sports/#/Leeds , die zuvor keine Veröffentlichungen empfangen haben, jetzt Veröffentlichungen empfangen. Dies liegt daran, dass die Sports/Football/Arsenal -und Sports/Rugby/Leeds -Veröffentlichungen, die für die anderen Subskriptionen an QMB weitergeleitet werden, jetzt für jeden Subskribenten verfügbar sind, der QMB zugeordnet ist. Folglich empfangen die Subskriptionen für die lokalen Themen Sports/# und Sports/#/Leeds die Veröffentlichung Sports/Rugby/Leeds . Sports/#/Arsenal erhält weiterhin keine Veröffentlichung, weil für Sport/Fußball die Eigenschaft WILDCARD auf BLOCKgesetzt ist.

Subskription	Themenzeichenfolge	Empfangene Veröffentlichungen	Hinweise
SPORTS	Sports/#	Sports/Rugby/Leeds	All publications to Football subtree blocked by WILDCARD (BLOCK) on Sports/Football

Tabelle 5. Auf QMB empfangene Veröffentlichungen (Forts.)

Subskription	Themenzeichenfolge	Empfangene Veröffentlichungen	Hinweise
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) on Sports/ Football verhindert Platzhaltersubskription unter Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Der Standardwert WILDCARD für Sport/Rugby verhindert nicht das Platzhalterabonnement für Leeds .
FARSENAL	Sports/Football/Arsenal	Sports/Football/Arsenal	Arsenal empfängt eine Veröffentlichung, da die Subskription keinen Platzhalter hat.
FLEEDS	Sports/Rugby/Leeds	Sports/Rugby/Leeds	Leeds würde für jedes Ereignis eine Veröffentlichung empfangen.

Bei den meisten Anwendungen ist es nicht erwünscht, dass eine Subskription das Verhalten einer anderen Subskription beeinflusst. Eine wichtige Verwendung der Eigenschaft WILDCARD mit dem Wert BLOCK ist es, dass sich die Subskriptionen für dieselbe Themenzeichenfolge mit Platzhalterzeichen einheitlich verhalten. Unabhängig davon, ob sich die Subskription in demselben Warteschlangenmanager wie der Publisher oder in einem anderen Warteschlangenmanager befindet, sind die Resultate der Subskription dieselben.

Platzhalter und Datenströme

WebSphere MQ Version 6-Datenströme werden Themen von WebSphere MQ Version 7 zugeordnet (siehe „Datenströme und Themen“ auf Seite 49). In der Standardzuordnung, die von **strmqbrk** in Version 7 ausgeführt wird, werden alle Themen im Datenstrom Sports dem Thema Sportszugeordnet. Alle Themen im Datenstrom Business werden dem Thema Businesszugeordnet.

Eine Subskription in WebSphere MQ Version 6 für * im Datenstrom Sports empfängt alle Veröffentlichungen in der Baumstruktur Sports und keine Veröffentlichungen in der Baumstruktur Business . Dieselbe Subskription in Version 7 würde alle Veröffentlichungen in der Baumstruktur Sports und alle Veröffentlichungen in der Baumstruktur Business empfangen. Um dieses Verhalten zu blockieren, legt **strmqbrk** die Eigenschaft WILDCARD fest, wenn Datenströme auf Version 7 migriert werden. Sie setzt sie auf den Wert BLOCK für jedes der Themen der höchsten Ebene, die aus einem Datenstrom migriert wurden. Die Eigenschaft WILDCARD von Sports und Business wird durch die Konvertierung aus den Streams der Version 6 mit den Namen Sports und Businessauf den Wert BLOCK gesetzt.

Bei einer neuen Anwendung, die für die Publish/Subscribe-API geschrieben wurde, hat dies zur Folge, dass eine Subskription für * keine Veröffentlichungen empfängt. Um alle Sportveröffentlichungen zu erhalten, müssen Sie Sports/*oder Sports/# und ähnlich für Business -Veröffentlichungen subskribieren.

Das Verhalten einer vorhandenen Publish/Subscribe-Anwendung in der Warteschlange ändert sich nicht, wenn der Publish/Subscribe-Broker auf WebSphere MQ Version 7 migriert wird. Die Eigenschaft **Stream-Name** im Befehl **Publish**, **Register Publisher**oder **Subscriber** wird dem Namen des Themas zugeordnet, in das der Datenstrom migriert wurde.

Platzhalter und Subskriptionspunkte

WebSphere Message Broker-Subskriptionspunkte werden Themen von WebSphere MQ Version 7 zugeordnet; siehe „Subskriptionspunkte und Themen“ auf Seite 51. In der Standardzuordnung, die von **migmqbrk** in Version 7 ausgeführt wird, werden alle Themen im Subskriptionspunkt Sports dem Thema Sportszugeordnet. Alle Themen im Subskriptionspunkt Business würden dem Thema Businesszugeordnet.

Eine Subskription für WebSphere Message Broker Version 6 für * im Sports -Subskriptionspunkt empfängt alle Veröffentlichungen in der Baumstruktur Sports und keine Veröffentlichungen in der Baumstruktur Business . Dieselbe Subskription in Version 7 würde alle Veröffentlichungen in der Baumstruktur Sports und alle Veröffentlichungen in der Baumstruktur Business empfangen. Um dieses Verhalten zu blockieren, legt **migmqbrk** die Eigenschaft WILDCARD fest, wenn Subskriptionspunkte auf Version 7 migriert werden. Sie setzt sie auf den Wert BLOCK für jedes der Themen der höchsten Ebene, die von einem Subskriptionspunkt migriert werden. Die Eigenschaft WILDCARD von Sports und Business wird durch die Konvertierung aus den WebSphere Message Broker-Subskriptionspunkten Sports und Business auf den Wert BLOCK gesetzt.

Wenn eine neue Anwendung für die Publish/Subscribe-API geschrieben wird, empfängt eine Subskription von * keine Veröffentlichungen. Um alle Sportveröffentlichungen zu erhalten, müssen Sie Sports/*oder Sports/# und ähnlich für Business -Veröffentlichungen subscribieren.

Das Verhalten einer vorhandenen Publish/Subscribe-Anwendung in der Warteschlange ändert sich nicht, wenn der Publish/Subscribe-Broker auf WebSphere MQ Version 7 migriert wird. Die Eigenschaft **Sub-Point** im Befehl **Publish, Register Publisher** oder **Subscriber** wird dem Namen des Themas zugeordnet, auf das die Subskription migriert wurde.

Beispiel: Publish/Subscribe-Cluster für Sport erstellen

Die folgenden Schritte erstellen einen Cluster, CL1, mit vier Warteschlangenmanagern: zwei vollständigen Repositories, CL1A und CL1B , und zwei Teilrepositories, QMA und QMB. Die vollständigen Repositories enthalten nur Clusterdefinitionen. QMA wird als Cluster-Topic-Host festgelegt. Permanente Subskriptionen sind in QMA und QMB definiert.

Anmerkung: Das Beispiel ist für Windowscodiert. Wenn Sie das Beispiel auf anderen Plattformen konfigurieren und testen möchten, müssen Sie den Programmcode für [Create qmgrs.bat](#) und [create pub.bat](#) ändern.

1. Erstellen Sie die Scriptdateien.
 - a. [Create topics.tst](#)
 - b. [Create wildsubs.tst](#)
 - c. [Create fullsubs.tst](#)
 - d. [Create qmgrs.bat](#)
 - e. [create pub.bat](#)
2. Führen Sie [Create qmgrs.bat](#) aus, um die Konfiguration zu erstellen.

```
qmgrs
```

Erstellen Sie die Themen in [Abbildung 10](#) auf [Seite 42](#). Das Script in [Abbildung 5](#) erstellt die Clusterthemen Sports/Football und Sports/Rugby.

Anmerkung: Die Option REPLACE ersetzt nicht die TOPICSTR-Eigenschaften eines Themas. TOPICSTR ist eine Eigenschaft, die im Beispiel gezielt variiert wird, um verschiedene Themenstrukturen zu testen. Wenn Sie ein Thema ändern möchten, müssen Sie das Thema zuerst löschen.

```

DELETE TOPIC ('Sports')
DELETE TOPIC ('Football')
DELETE TOPIC ('Arsenal')
DELETE TOPIC ('Blackburn')
DELETE TOPIC ('Chelsea')
DELETE TOPIC ('Rugby')
DELETE TOPIC ('Leeds')
DELETE TOPIC ('Wigan')
DELETE TOPIC ('Warrington')
DELETE TOPIC ('St. Helens')

DEFINE TOPIC ('Sports')      TOPICSTR('Sports')
DEFINE TOPIC ('Football')   TOPICSTR('Sports/Football') CLUSTER(CL1) WILDCARD(BLOCK)
DEFINE TOPIC ('Arsenal')    TOPICSTR('Sports/Football/Arsenal')
DEFINE TOPIC ('Blackburn')  TOPICSTR('Sports/Football/Blackburn')
DEFINE TOPIC ('Chelsea')    TOPICSTR('Sports/Football/Chelsea')
DEFINE TOPIC ('Rugby')      TOPICSTR('Sports/Rugby') CLUSTER(CL1)
DEFINE TOPIC ('Leeds')      TOPICSTR('Sports/Rugby/Leeds')
DEFINE TOPIC ('Wigan')      TOPICSTR('Sports/Rugby/Wigan')
DEFINE TOPIC ('Warrington') TOPICSTR('Sports/Rugby/Warrington')
DEFINE TOPIC ('St. Helens') TOPICSTR('Sports/Rugby/St. Helens')

```

Abbildung 12. Themen löschen und erstellen: topics.tst

Anmerkung: Löschen Sie die Themen, da REPLACE keine Themenzeichenfolgen ersetzt.

Erstellen Sie Subskriptionen mit Platzhaltern. Die Platzhalter entsprechen den Themen mit Themenobjekten in Abbildung 10 auf Seite 42. Erstellen Sie für jede Subskription eine Warteschlange. Beim Ausführen bzw. erneuten Ausführen des Scripts werden die Warteschlangen geleert, und die Subskriptionen werden gelöscht.

Anmerkung: Die Option REPLACE ersetzt nicht die die Eigenschaften TOPICOBJ oder TOPICSTR einer Subskription. TOPICOBJ oder TOPICSTR sind die Eigenschaften, die im Beispiel sinnvoll variiert werden, um verschiedene Subskriptionen zu testen. Um sie zu ändern, löschen Sie zunächst die Subskription.

```

DEFINE QLOCAL(QSPORTS) REPLACE
DEFINE QLOCAL(QSARSENAL) REPLACE
DEFINE QLOCAL(QSLEEDS) REPLACE
CLEAR QLOCAL(QSPORTS)
CLEAR QLOCAL(QSARSENAL)
CLEAR QLOCAL(QSLEEDS)

DELETE SUB (SPORTS)
DELETE SUB (SARSENAL)
DELETE SUB (SLEEDS)
DEFINE SUB (SPORTS) TOPICSTR('Sports/#') DEST(QSPORTS)
DEFINE SUB (SARSENAL) TOPICSTR('Sports+/Arsenal') DEST(QSARSENAL)
DEFINE SUB (SLEEDS) TOPICSTR('Sports+/Leeds') DEST(QSLEEDS)

```

Abbildung 13. Platzhaltersubskriptionen erstellen: wildsubs.tst

Erstellen Sie Subskriptionen, die die Cluster-Topic-Objekte referenzieren.

Anmerkung:

Das Begrenzungszeichen / wird automatisch zwischen die von TOPICOBJ referenzierte Themenzeichenfolge und die von TOPICSTR definierte Themenzeichenfolge eingefügt.

Die Definition DEFINE SUB(FARSENAL) TOPICSTR('Sports/Football/Arsenal') DEST(QFARSENAL) erstellt dieselbe Subskription. Die Verwendung von TOPICOBJ ist eine zeiteffiziente Methode, bereits definierte Themenzeichenfolgen zu referenzieren. Die erstellte Subskription referenziert das Themenobjekt nicht länger.

```

DEFINE QLOCAL(QFARSENAL) REPLACE
DEFINE QLOCAL(QRLEEDS) REPLACE
CLEAR QLOCAL(QFARSENAL)
CLEAR QLOCAL(QRLEEDS)

DELETE SUB (FARSENAL)
DELETE SUB (RLEEDS)
DEFINE SUB (FARSENAL) TOPICOBJ('Football') TOPICSTR('Arsenal') DEST(QFARSENAL)
DEFINE SUB (RLEEDS) TOPICOBJ('Rugby') TOPICSTR('Leeds') DEST(QRLEEDS)

```

Abbildung 14. Subskriptionen löschen und erstellen: fullsubs.tst

Erstellen Sie einen Cluster mit zwei Repositorys. Erstellen Sie zwei Teilrepositorys zum Veröffentlichen und Abonnieren. Führen Sie das Script erneut aus, um alles zu löschen und neu zu starten. Das Script erstellt zudem die Themenhierarchie und die ursprünglichen Platzhaltersubskriptionen.

Anmerkung:

Schreiben Sie auf anderen Plattformen ein ähnliches Script, oder geben Sie alle Befehle ein. Mit einem Script lässt sich alles schneller löschen und mit einer identischen Konfiguration erneut starten.

```

@echo off
set port.CL1B=1421
set port.CL1A=1420
for %%A in (CL1A CL1B QMA QMB) do call :createQM %%A
call :configureQM CL1A CL1B %port.CL1B% full
call :configureQM CL1B CL1A %port.CL1A% full
for %%A in (QMA QMB) do call :configureQM %%A CL1A %port.CL1A% partial
for %%A in (topics.tst wildsubs.tst) do runmqsc QMA < %%A
for %%A in (wildsubs.tst) do runmqsc QMB < %%A
goto:eof

:createQM
echo Configure Queue manager %1
endmqm -p %1
for %%B in (dlt crt str) do %%Bmqm %1
goto:eof

:configureQM
if %1==CL1A set p=1420
if %1==CL1B set p=1421
if %1==QMA set p=1422
if %1==QMB set p=1423
echo configure %1 on port %p% connected to repository %2 on port %3 as %4 repository
echo DEFINE LISTENER(LST%1) TRPTYPE(TCP) PORT(%p%) CONTROL(QMGR) REPLACE | runmqsc %1
echo START LISTENER(LST%1) | runmqsc %1
if full==%4 echo ALTER QMGR REPOS(CL1) DEADQ(SYSTEM.DEAD.LETTER.QUEUE) | runmqsc %1
echo DEFINE CHANNEL(TO.%2) CHLTYPE(CLUSSDR) TRPTYPE(TCP) CONNAME('LOCALHOST(%3)') CLUSTER(CL1)
REPLACE | runmqsc %1
echo DEFINE CHANNEL(TO.%1) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CONNAME('LOCALHOST(%p%)') CLUS
TER(CL1) REPLACE | runmqsc %1
goto:eof

```

Abbildung 15. Warteschlangenmanager erstellen: qmgrs.bat

Aktualisieren Sie die Konfiguration durch Hinzufügen der Subskriptionen zu den Cluster-Topics.

```

@echo off
for %%A in (QMA QMB) do runmqsc %%A < wildsubs.tst
for %%A in (QMA QMB) do runmqsc %%A < upsubs.tst

```

Abbildung 16. Subskriptionen aktualisieren: upsubs.bat

Führen Sie pub.bat mit einem Warteschlangenmanager als Parameter aus, um Nachrichten zu veröffentlichen, die die Veröffentlichungsthema-Zeichenfolge enthalten. Pub.bat verwendet das Beispielprogramm amqspub.


```

@echo off
@rem Provide queue manager name as a parameter
set S=Sports
set S=6 Sports/Football Sports/Football/Arsenal
set S=6 Sports/Rugby Sports/Rugby/Leeds
for %%B in (6) do echo %%B | amqspub %%B %1

```

Abbildung 17. Veröffentlichen: pub.bat

Datenströme und Themen

Das eingereichte Publish/Subscribe basiert auf dem Konzept eines Veröffentlichungsdatenstroms, das es im Modell für integriertes Publish/Subscribe nicht gibt. Im eingereichten Publish/Subscribe bieten Datenströme die Möglichkeit, den Informationsfluss für unterschiedliche Themen zu trennen. In IBM WebSphere MQ Version 6.0 wird ein Datenstrom als Warteschlange implementiert, die auf jedem Broker definiert ist, der den Datenstrom unterstützt. Jede Warteschlange trägt denselben Namen (den Namen des Datenstroms). Ab IBM WebSphere MQ Version 7.0 wird ein Datenstrom als ein Thema der höchsten Ebene implementiert, das durch eine Verwaltungsaktion einer anderen Themen-ID zugeordnet werden kann.

Der Standarddatenstrom SYSTEM.BROKER.DEFAULT.STREAM wird für alle Broker und Warteschlangenmanager in einem Netz automatisch eingerichtet und es ist keine zusätzliche Konfiguration erforderlich, um den Standarddatenstrom verwenden zu können. Sie können sich den Standarddatenstrom als nicht benannten Themenspeicherbereich vorstellen. Themen, die im Standarddatenstrom veröffentlicht werden, sind sofort für alle verbundenen Version 6.0 -Broker und alle Warteschlangenmanager ab Version 7.0 mit aktiviertem eingereichtem Publish/Subscribe verfügbar. Benannte Datenströme gleichen separaten, benannten Themenspeicherbereichen. Der benannte Datenstrom muss auf jedem Broker definiert werden, auf dem er verwendet wird.

Wenn Sie ein Thema definieren, ist es für Publish/Subscribe-Broker von Version 6.0 sowie für Publisher und Subskribenten, die in einer höheren Version von IBM WebSphere MQ ausgeführt werden, ohne besondere Konfiguration verfügbar.

Wenn sich die Publisher und Subskribenten auf verschiedenen Warteschlangenmanagern befinden, ist für den Austausch der Veröffentlichungen und Subskriptionen zwischen den Brokern keine weitere Konfiguration erforderlich, sobald die Broker in derselben Brokerhierarchie miteinander verbunden sind. Diese Interoperabilität funktioniert auch umgekehrt.

Benannte Datenströme

Ein Lösungsentwickler, der mit dem Programmiermodell für eingereichtes Publish/Subscribe arbeitet, kann beschließen, alle Sportveröffentlichungen in einen benannten Stream namens Sport zu stellen. In Version 6.0 wird ein Datenstrom häufig automatisch auf andere Broker repliziert, die die Modellwarteschlange SYSTEM.BROKER.MODEL.STREAM verwenden. Damit der Datenstrom für einen Warteschlangenmanager verfügbar ist, der in Version 7.0 aufwärts ausgeführt wird und für den eingereichtes Publish/Subscribe aktiviert ist, muss der Datenstrom jedoch manuell hinzugefügt werden.

Wenn Sie einen Warteschlangenmanager von Version 6.0 migrieren, können Sie den Befehl **strmqbrk** migriert Version 6.0 benannte Datenströme auf Themen migrieren. Der Datenstrom Sport wird dem Thema Sport zugeordnet. Dies gilt nicht für z/OS.

Eingereichte Publish/Subscribe-Anwendungen, die Soccer/Results im Datenstrom Sport subskribieren, funktionieren unverändert. Integrierte Publish/Subscribe-Anwendungen, die das Thema Sport mit MQSUB subskribieren und die Themenzeichenfolge Soccer/Results angeben, empfangen dieselben Veröffentlichungen ebenfalls.

Wenn das Thema Soccer/Result von **strmqbrk** erstellt wird, wird es als untergeordnetes Element des Themas Sport mit der Themenzeichenfolge Sport definiert. Eine Subskription für Soccer/Results wird als Subskription für Sport/Soccer/Results realisiert, sodass Veröffentlichungen für den Sport-Datenstrom einem anderen Bereich im Topicbereich zugeordnet werden als Veröffentlichungen für einen anderen Datenstrom, z. B. Business.

Es gibt Szenarios, für die die von **strmqbrk** ausgeführte automatische Migration nicht die Antwort ist, und Sie müssen Datenströme manuell hinzufügen. Die Aufgabe zum Hinzufügen eines Datenstroms wird im Abschnitt Datenstrom hinzufügen beschrieben. Möglicherweise müssen Sie Datenströme aus drei Gründen manuell hinzufügen.

1. Sie verwalten weiterhin Publish/Subscribe-Anwendungen auf Warteschlangenmanagern der Version 6, die mit neu geschriebenen Publish/Subscribe-Anwendungen interagieren, die auf späteren Warteschlangenmanagern ausgeführt werden.
2. Sie entwickeln weiterhin Ihre Anwendungen mit eingereihem Publish/Subscribe, die auf Warteschlangenmanagern einer höheren Version ausgeführt werden, statt die Anwendungen auf die MQI-Schnittstelle für integriertes Publish/Subscribe zu migrieren.
3. Die Standardzuordnung von Datenströmen zu Themen führt zu einer "Kollision" im Themenspeicherbereich und Veröffentlichungen in einem Datenstrom verfügen über dieselbe Themenzeichenfolge wie Veröffentlichungen aus anderen Quellen.

Berechtigungen

Standardmäßig gibt es im Stammverzeichnis der Themenstruktur mehrere Themenobjekte: `SYSTEM.BASE.TOPIC`, `SYSTEM.BROKER.DEFAULT.STREAM` und `SYSTEM.BROKER.DEFAULT.SUBPOINT`. Berechtigungen (z. B. für die Veröffentlichung oder Subskription) werden durch die Berechtigungen auf dem `SYSTEM.BASE.TOPIC` festgelegt; alle Berechtigungen auf `SYSTEM.BROKER.DEFAULT.STREAM` oder `SYSTEM.BROKER.DEFAULT.SUBPOINT` werden ignoriert. Wenn entweder `SYSTEM.BROKER.DEFAULT.STREAM` oder `SYSTEM.BROKER.DEFAULT.SUBPOINT` gelöscht und mit einer nicht leeren Themenzeichenfolge erneut erstellt werden, werden die für diese Objekte definierten Berechtigungen auf dieselbe Weise wie ein normales Themenobjekt verwendet.

Zuordnung von Datenströmen zu Themen

Ein Datenstrom mit eingereihem Publish/Subscribe wird in Version 7.0 aufwärts abgebildet, indem eine Warteschlange mit demselben Namen wie dem des Datenstroms erstellt wird. Manchmal wird die Warteschlange auch Datenstromwarteschlange genannt, weil sie von Anwendungen mit eingereihem Publish/Subscribe so wahrgenommen wird. Die Warteschlange wird für die Publish/Subscribe-Steuerkomponente identifiziert, indem sie zu der speziellen Namensliste namens `SYSTEM.QPUBSUB.QUEUE.NAMELIST` hinzugefügt wird. Sie können so viele Datenströme wie nötig hinzufügen, indem Sie der Namensliste weitere spezielle Warteschlangen hinzufügen. Schließlich müssen Sie Themen hinzufügen, die dieselben Namen wie die Datenströme und dieselben Themenzeichenfolgen wie der Datenstromname aufweisen, damit Sie Inhalte in den Themen veröffentlichen und abonnieren können.

Unter außergewöhnlichen Umständen können Sie für die Themen, die den Datenströmen entsprechen, jedoch beliebige Themenzeichenfolgen angeben, die Sie bei der Auswahl der Themen definieren. Zweck der Themenzeichenfolge ist es, dem Thema einen eindeutigen Namen im Themenspeicherbereich zu geben. Normalerweise erfüllt der Datenstromname diesen Zweck perfekt. Manchmal kollidieren ein Datenstromname und ein vorhandener Themename. Um das Problem zu lösen, können Sie eine andere Themenzeichenfolge für das Thema, das dem Datenstrom zugeordnet ist, auswählen. Stellen Sie bei der Auswahl einer Themenzeichenfolge sicher, dass sie eindeutig ist.

Die Themenzeichenfolge, die in der Themendefinition definiert ist, ist wie üblich im Vergleich zu der Themenzeichenfolge, die von Publishern und Subskribenten über die MQI-Aufrufe `MQOPEN` oder `MQSUB` bereitgestellt werden, mit einem Präfix versehen. Anwendungen, die auf Themen verweisen, die Themenobjekte verwenden, sind von der Auswahl der Präfixthemenzeichenfolge nicht betroffen. Deshalb können Sie eine beliebige Themenzeichenfolge auswählen, sofern die Veröffentlichungen im Themenspeicherbereich eindeutig bleiben.

Für die Neuordnung von unterschiedlichen Datenströmen zu anderen Themen ist es wichtig, dass die Präfixe, die für die Themenzeichenfolgen verwendet werden, eindeutig sind, damit eine Themengruppe komplett von einer anderen Themengruppe getrennt werden kann. Sie müssen eine allgemeine Benennungskonvention für Themen definieren, die strikt befolgt werden muss, damit die Zuordnung funktioniert. Wenn in Version 7.0 eine Kollision von Themenzeichenfolgen aufgetreten ist, können Sie die Themenspeicherbereiche mithilfe von Datenströmen trennen. Ab Version 7.0 verwenden Sie den

Präfixmechanismus, um eine Themenzeichenfolge einem anderen Bereich im Themenspeicherbereich zuzuordnen.

Anmerkung: Wenn Sie einen Datenstrom löschen, löschen Sie zuerst alle Subskriptionen im Datenstrom. Diese Aktion ist dann besonders wichtig, wenn Subskriptionen von anderen Brokern in der Brokerhierarchie stammen.

Beispiel

In [Abbildung 18](#) auf Seite 51 hat das Thema 'Sport' die Themenzeichenfolge 'xyz', was dazu führt, dass Veröffentlichungen, die aus dem Datenstrom 'Sport' stammen, mit der Zeichenfolge 'xyz' im Topicbereich des Warteschlangenmanagers der Version 7 als Präfix versehen werden. Wenn in Version 7 Inhalt im Thema 'Sport' veröffentlicht oder subskribiert wird, wird die Themenzeichenfolge mit dem Präfix 'xyz' versehen. Wenn die Veröffentlichung an einen Subskribenten der Version 6 übertragen wird, wird das Präfix 'xyz' aus der Veröffentlichung entfernt und in den Datenstrom 'Sport' gestellt. Umgekehrt wird das Präfix 'xyz' zur Themenzeichenfolge hinzugefügt, wenn eine Veröffentlichung von Version 6 auf Version 7 vom Datenstrom 'Sport' zum Thema 'Sport' fließt.

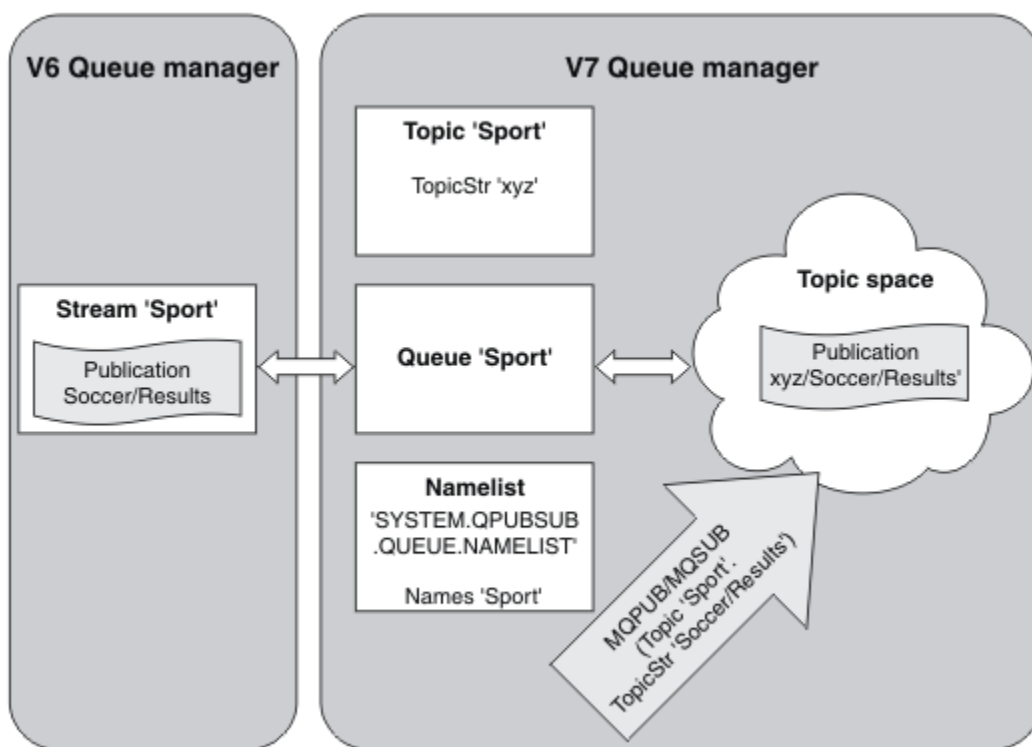


Abbildung 18. Datenströme der Version 6, die mit Themen der Version 7 koexistieren

Subskriptionspunkte und Themen

Ein Subskriptionspunkt, der verwendet wird, um Veröffentlichungen von einer bestimmten Gruppe von Veröffentlichungsknoten in WebSphere MQ Event Broker und Message Broker anzufordern. Benannte Subskriptionspunkte werden durch Themen und Themenobjekte emuliert.

Die Migrationsprozedur für WebSphere MQ Event Broker V6.0 auf WebSphere MQ V7.0.1 **migmbrrk** konvertiert benannte Subskriptionspunkte in Themen und Themenobjekte. Ein Subskriptionspunkt wird automatisch migriert, wenn er über eine ständige Veröffentlichung oder einen registrierten Subskribenten verfügt. **migmbrrk** erstellt Themenobjekte aus benannten Subskriptionspunkten. Der Name des Subskriptionspunkts wird zum Namen des Themenobjekts und zur Themenzeichenfolge selbst. Das Topic-Objekt wird `SYSTEM.QPUBSUB.SUBPOINT.NAMELIST` hinzugefügt.

Wenn ein Themenobjekt mit demselben Namen vorhanden ist, führt **migmbrrk** eine der beiden folgenden Aktionen aus.

1. Wenn das Themenobjekt eine andere Themenzeichenfolge hat oder wenn der Name des Subskriptionspunkts länger als ein Objektname ist, erstellt **migmbbrk** ein Themenobjekt mit einem generierten Namen.
2. Wenn das Themenobjekt dieselbe Themenzeichenfolge hat, fügt **migmbbrk** das vorhandene Objekt der Namensliste hinzu.

Informationen zum manuellen Hinzufügen von Subskriptionspunkten finden Sie im Abschnitt [Subskriptionspunkt hinzufügen](#).

Subskriptionspunkte in WebSphere MQ Event Broker

Veröffentlichungsknoten werden in einem WebSphere MQ -Nachrichtenfluss zum Filtern und Übertragen von Nachrichten an Subskribenten verwendet. Publisher legen normalerweise keine Subskriptionspunkte auf Veröffentlichungsknoten fest. Subskribenten registrieren ein Interesse an einer bestimmten Themen-Gruppe und geben normalerweise auch keine Subskriptionspunkte an.

Mit einem Subskriptionspunkt können Sie auswählen, welche Veröffentlichungsknoten Nachrichten an eine Subskription weiterleiten. Der Subskribent qualifiziert sich für eine Gruppe von Themen mit dem Namen eines Subskriptionspunkts.

Ordnen Sie der Eigenschaft **Subscription point** des Veröffentlichungsknotens einen Namen zu, um den Namen des Subskriptionspunkts festzulegen.

Die Eigenschaft Subskriptionspunkt steuert, ob eine Veröffentlichung zu einem Thema an Subskribenten desselben Themas weitergeleitet wird. Veröffentlichungen von Veröffentlichungsknoten mit einem benannten Subskriptionspunkt werden nur an Subskribenten desselben Subskriptionspunkts weitergeleitet. Veröffentlichungen von Veröffentlichungsknoten ohne benannten Subskriptionspunkt (Standardwert) werden nur an Subskribenten weitergeleitet, die keinen Subskriptionspunkt benannt haben.

Knoten mit einem benannten Subskriptionspunkt senden Publish -Befehlsnachrichten im Format MQRFH2 , wobei die Eigenschaft **SubPoint** festgelegt ist. Subskriptionen für einen benannten Subskriptionspunkt müssen die Eigenschaft **SubPoint** in der MQRFH2 Register subscriber -Befehlsnachricht festlegen.

Subskriptionspunkte in WebSphere MQ

WebSphere MQ ordnet Subskriptionspunkte verschiedenen Topicbereichen innerhalb der Themenstruktur WebSphere MQ zu. Themen in Befehlsnachrichten ohne Subskriptionspunkt werden unverändert dem Stammelement der Themenstruktur WebSphere MQ zugeordnet und übernehmen Eigenschaften von SYSTEM.BASE.TOPIC.

Befehlsnachrichten mit einem Subskriptionspunkt werden mithilfe der Liste der Themenobjekte in SYSTEM.QPUBSUB.SUBPOINT.NAMELIST verarbeitet. Der Name des Subskriptionspunktes in der Befehlsnachricht wird für jedes Themenobjekt in der Liste mit der Themenzeichenfolge abgeglichen. Wenn eine Übereinstimmung gefunden wird, wird der Name des Subskriptionspunktes als Themenknoten der Themenzeichenfolge vorangestellt. Das Thema übernimmt die Eigenschaften vom zugehörigen Themenobjekt in SYSTEM.QPUBSUB.SUBPOINT.NAMELIST.

Die Verwendung von Subskriptionspunkten führt dazu, dass für jeden Subskriptionspunkt ein separater Themenspeicherbereich erstellt wird. Das Stammelement des Themenspeicherbereichs ist ein Thema, das denselben Namen wie der Subskriptionspunkt hat. Themen in den einzelnen Themenspeicherbereichen übernehmen die Eigenschaften von dem Themenobjekt, das denselben Namen wie der Subskriptionspunkt hat.

Alle Eigenschaften, die in dem entsprechenden Themenobjekt nicht festgelegt sind, werden wie üblich von SYSTEM.BASE.TOPIC übernommen.

Vorhandene Publish/Subscribe-Anwendungen in der Warteschlange, die MQRFH2 -Nachrichtenheader verwenden, funktionieren weiterhin, indem sie die Eigenschaft **SubPoint** in den Befehlsnachrichten Publish oder Register subscriber festlegen. Der Subskriptionspunkt wird in der Befehlsnachricht

mit der Themenzeichenfolge kombiniert und das sich ergebende Thema wird wie alle anderen Themen verarbeitet.

Eine neue WebSphere MQ V7 -Anwendung ist von Subskriptionspunkten nicht betroffen. Wenn es ein Thema verwendet, das von einem der übereinstimmenden Themenobjekte übernimmt, interagiert es mit einer Anwendung in der Warteschlange, die den übereinstimmenden Subskriptionspunkt verwendet.

Beispiel

Eine vorhandene Publish/Subscribe-Anwendung von WebSphere MQ Event Broker in einem Verbund verwendet Subskriptionspunkte, um Aktienkurse in verschiedenen Währungen zu veröffentlichen. Der Spotpreis in Dollar der IBM -Aktie wird unter Verwendung des Subskriptionspunkts USD und des Themas NYSE/IBM/SPOT veröffentlicht. Der Sterling-Preis wird mit demselben Thema und dem Subskriptionspunkt GBP veröffentlicht.

Die Migrationsprozedur für WebSphere MQ erstellt zwei Themenobjekte, GBP und USD, mit den entsprechenden Themenzeichenfolgen 'GBP' und 'USD'.

Vorhandene Publisher für das Thema NYSE/IBM/SPOT, die zur Ausführung in WebSphere MQ migriert wurden und den Subskriptionspunkt USD verwenden, erstellen Veröffentlichungen für das Thema USD/NYSE/IBM/SPOT. Ebenso erstellen vorhandene Subskribenten von NYSE/IBM/SPOT, die den Subskriptionspunkt USD verwenden, Subskriptionen in USD/NYSE/IBM/SPOT.

Abonnieren Sie den Dollarpreis in einem Publish/Subscribe-Programm der Version 7, indem Sie MQSUB aufrufen. Erstellen Sie eine Subskription mit dem Themenobjekt USD und der Themenzeichenfolge 'NYSE/IBM/SPOT', wie im 'C'-Codefragment dargestellt.

```
strncpy(sd.ObjectName, "USD", MQ_TOPIC_NAME_LENGTH);  
sd.ObjectString.VSPtr = "NYSE/IBM/SPOT";  
sd.ObjectString.VSLength = MQVS_NULL_TERMINATED;  
MQSUB(Hconn, &sd, &Hobj, &Hsub, &CompCode, &Reason);
```

Überlegen Sie, ob Ihre WebSphere MQ Event Broker-Anwendungen im Verbund immer die Subskriptionspunkte USD und GBP verwenden. Ist dies der Fall, erstellen Sie die Themenobjekte USD und GBP nur einmal als Clusterthemen auf dem Cluster-Topic-Host. Sie müssen Schritt [../com.ibm.mq.mig.doc/q007670_dita#q007670_/clusterstep](#) der Migrationsprozedur nicht ausführen, um SYSTEM.BASE.TOPIC auf jedem Warteschlangenmanager im Cluster in ein Clusterthema zu ändern. Führen Sie stattdessen die folgenden Schritte durch:

1. Legen Sie das Attribut CLUSTER der Themenobjekte USD und GBP auf dem Clusterthemenhost fest.
2. Löschen Sie alle Kopien der Themenobjekte USD und GBP auf anderen Warteschlangenmanagern im Cluster.
3. Vergewissern Sie sich, dass USD und GBP in SYSTEM.QPUBSUB.SUBPOINT.NAMELIST auf jedem Warteschlangenmanager im Cluster definiert sind.

Verteiltes Publish/Subscribe

Dieser Abschnitt enthält Informationen darüber, wie Publish/Subscribe-Messaging zwischen Warteschlangenmanagern ausgeführt werden kann, und die beiden verschiedenen WS-Manager-Topologien, die verwendet werden können, um Warteschlangenmanager, Cluster und Hierarchien zu verbinden.

Warteschlangenmanager können mit anderen Warteschlangenmanagern in Ihrem WebSphere MQ Publish/Subscribe-System kommunizieren, sodass Subskribenten einen Warteschlangenmanager abonnieren und Nachrichten empfangen können, die ursprünglich in einem anderen Warteschlangenmanager veröffentlicht wurden. Im Abschnitt [Abbildung 19 auf Seite 54](#) finden Sie eine Abbildung, die dies verdeutlicht.

[Abbildung 19 auf Seite 54](#) zeigt ein Publish/Subscribe-System mit zwei Warteschlangenmanagern.

- Die veröffentlichende Stelle 4 veröffentlicht auf Warteschlangenmanager 2 Wettervorhersagen unter dem Thema 'Wetter' sowie Verkehrsmeldungen für wichtige Straßenverbindungen unter dem Thema 'Verkehr'.

- Subskribent 4 verwendet ebenfalls diesen Warteschlangenmanager und hat das Thema 'Verkehr' subskribiert.
- Subskribent 3 hat zusätzlich das Thema 'Wetter' subskribiert, obwohl er den Warteschlangenmanager einer anderen veröffentlichenden Stelle verwendet. Dies ist möglich, da die Warteschlangenmanager miteinander verbunden sind.

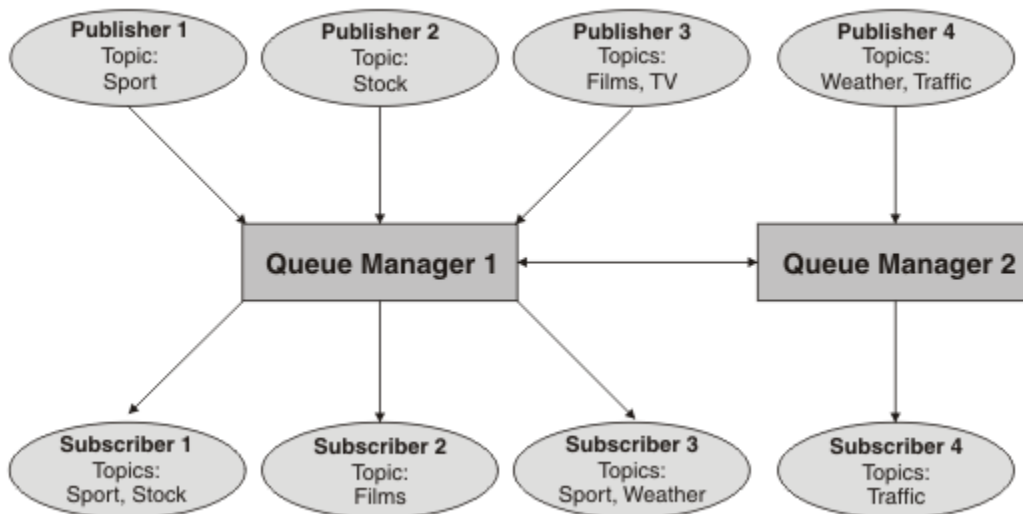


Abbildung 19. Beispiel für Publish/Subscribe mit zwei Warteschlangenmanagern

Wie funktioniert verteiltes Publish/Subscribe?

WebSphere MQ Publish/Subscribe verwendet Proxy-Subskriptionen, um sicherzustellen, dass Subskribenten Nachrichten empfangen können, die für ferne Warteschlangenmanager veröffentlicht werden.

Verteiltes Publish/Subscribe verwendet dieselben Komponenten wie die verteilte Steuerung von Warteschlangen, um Netze von Warteschlangenmanagern und folglich die Anwendungen, die eine Verbindung zu diesen Warteschlangenmanagern herstellen, zu verbinden. Weitere Informationen zum Messaging zwischen Warteschlangenmanagern und den Komponenten, die Verbindungen zwischen Warteschlangenmanagern herstellen, finden Sie in der Dokumentation zur *übergreifenden Kommunikation*.

Subskribenten müssen nichts über die Standardsubskriptionsoperation in einem verteilten Publish/Subscribe-System hinaus tun. Wenn eine Subskription auf einem Warteschlangenmanager vorgenommen wird, verwaltet der WS-Manager den Prozess, durch den die Subskription an verbundene Warteschlangenmanager weitergegeben wird. Proxy-Subskriptionen fließen an alle Warteschlangenmanager im Netz. Sie werden erstellt, um sicherzustellen, dass Veröffentlichungen an den Warteschlangenmanager zurückgeleitet werden, auf dem die ursprüngliche Subskription erstellt wurde (siehe [Abbildung 20](#) auf Seite 55).

Eine Veröffentlichung wird nur an einen fernen Warteschlangenmanager weitergegeben, wenn eine Subskription für dieses Thema auf diesem fernen Warteschlangenmanager vorhanden ist.

Ein Warteschlangenmanager konsolidiert alle erstellten Subskriptionen, unabhängig davon, ob er von lokalen Anwendungen oder von fernen Warteschlangenmanagern erstellt wird. Er erstellt Proxy-Subskriptionen für die Themen der Subskriptionen mit ihren Nachbarn, sofern keine Subskription vorhanden ist (siehe [Abbildung 21](#) auf Seite 55).

Wenn eine Anwendung Informationen veröffentlicht, leitet der empfangende Warteschlangenmanager sie an alle Anwendungen weiter, die gültige Subskriptionen auf fernen Warteschlangenmanagern haben. Er kann sie über einen oder mehrere temporäre Warteschlangenmanager weiterleiten (siehe [Abbildung 22](#) auf Seite 56).

Subskribent 1 registriert eine Subskription für ein bestimmtes Thema auf dem Warteschlangenmanager in Asien (1). Die Subskription für dieses Thema wird an alle anderen WS-Manager im Netz (2,3, 4) weitergeleitet.

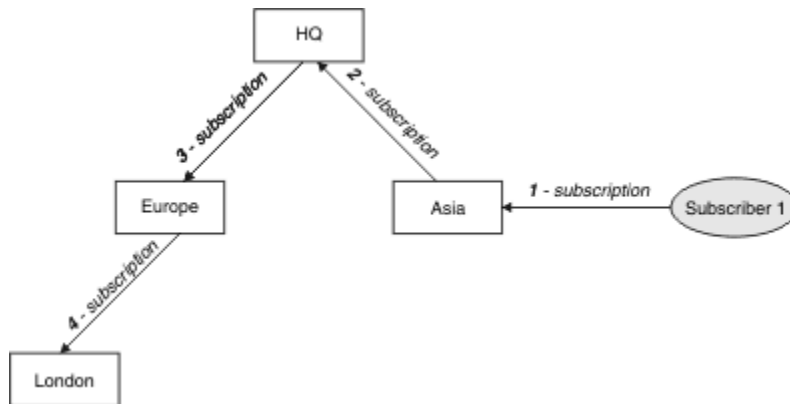


Abbildung 20. Weitergabe von Subskriptionen über ein Warteschlangenmanagernetzwerk

Subskribent 2 registriert eine Subskription für dasselbe Thema wie in [Abbildung 20](#) auf Seite 55 auf dem Warteschlangenmanager der Zentrale (5). Die Subskription für dieses Thema wird an den Warteschlangenmanager in Asien weitergeleitet, sodass er erkennt, dass Subskriptionen an anderer Stelle im Netz vorhanden sind. (6). Die Subskription wird nicht an den Warteschlangenmanager Europa weitergeleitet, da bereits eine Subskription für dieses Thema registriert wurde (siehe Schritt 3 in [Abbildung 20](#) auf Seite 55).

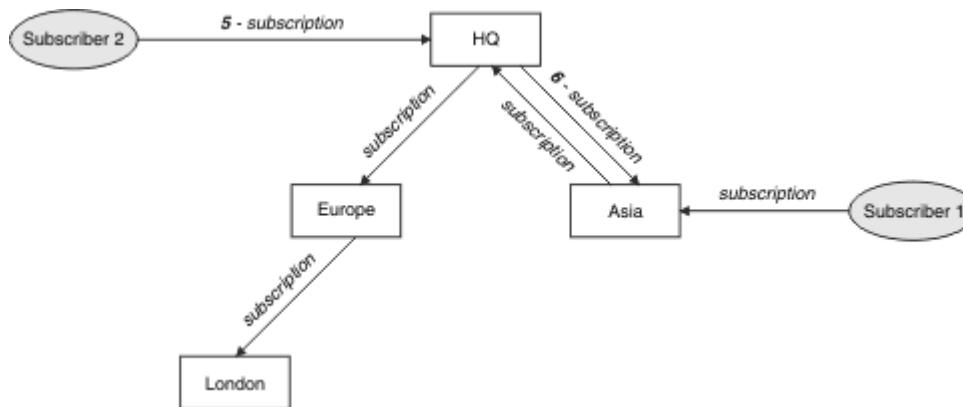


Abbildung 21. Mehrere Subskriptionen

Ein Publisher sendet eine Veröffentlichung zu demselben Thema wie in Abbildung 21 auf Seite 55 an den Warteschlangenmanager Europa (7). Für dieses Thema besteht eine Subskription von HQ nach Europa, sodass die Veröffentlichung an den HQ-Warteschlangenmanager (8) weitergeleitet wird. Es gibt jedoch keine Subskription von London nach Europa (nur von Europa nach London), sodass die Veröffentlichung nicht an den Warteschlangenmanager in London weitergeleitet wird. Der Warteschlangenmanager der Zentrale sendet die Veröffentlichung direkt an Subskribent 2 und an den Warteschlangenmanager in Asien (9). Die Veröffentlichung wird an Teilnehmer 1 aus Asien (10) weitergeleitet.

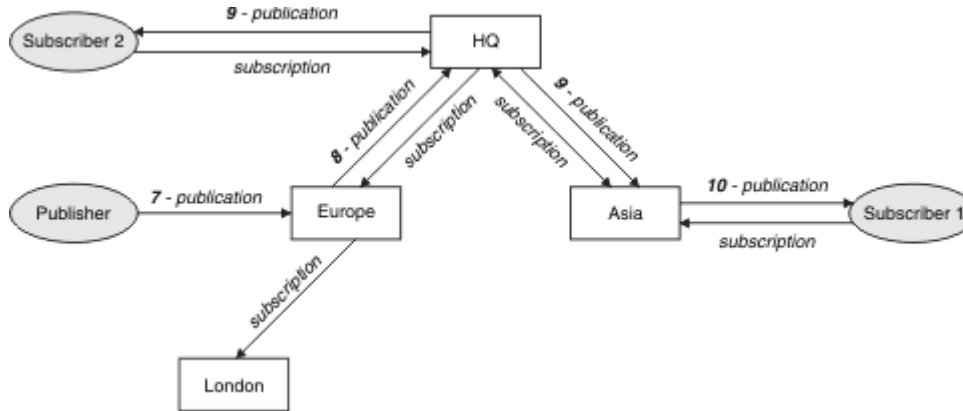


Abbildung 22. Weitergabe von Veröffentlichungen über ein Warteschlangenmanagernetzwerk

Wenn ein Warteschlangenmanager beliebige Veröffentlichungen oder Subskriptionen an einen anderen Warteschlangenmanager sendet, wird seine eigene Benutzer-ID in der Nachricht festgelegt. Wenn Sie eine Publish/Subscribe-Hierarchie verwenden und der eingehende Kanal so eingerichtet ist, dass er Nachrichten mit der Berechtigung der Benutzer-ID in der Nachricht einreicht, müssen Sie die Benutzer-ID des sendenden Warteschlangenmanagers berechtigen (siehe „Standard-Benutzer-IDs mit einer WS-Manager-Hierarchie verwenden“ auf Seite 115). Wenn Sie Publish/Subscribe-Cluster verwenden, erfolgt die Berechtigung durch den Cluster.

Aufgrund der miteinander verbundenen Art von Publish/Subscribe-Warteschlangenmanagern dauert es lange, bis Proxy-Subskriptionen um alle Knoten im Netz weitergegeben werden. Ferne Veröffentlichungen werden nicht unbedingt sofort subskribiert. Sie können die Subskriptionsverzögerung eliminieren, indem Sie das **Topic**-Attribut PROXYSUB mit dem Wert FORCE verwenden, wie in „Weitere Informationen zu Routing-Mechanismen“ auf Seite 57 beschrieben.

Die Subskriptionsoperation wird beendet, wenn die Proxy-Subskription in die entsprechende Übertragungswarteschlange für jeden direkt verbundenen Warteschlangenmanager eingereicht wurde. Die Subskriptionsoperation wartet nicht auf die Weitergabe der Proxy-Subskription an den Rest der Topologie.

Proxy-Subskriptionen werden dem Namen des Warteschlangenmanagers zugeordnet, der sie erstellt hat. Wenn Warteschlangenmanager in der Hierarchie denselben Namen haben, kann dies dazu führen, dass Veröffentlichungen sie nicht erreichen. Um dieses Problem wie beim Punkt-zu-Punkt-Messaging zu vermeiden, geben Sie Warteschlangenmanagern eindeutige Namen, insbesondere wenn sie direkt oder indirekt in einem WebSphere MQ -Netz verbunden sind.

In einem verteilten Publish/Subscribe-Netz kann der Fluss von Veröffentlichungen und Subskriptionen unter Verwendung des Veröffentlichungs- und Subskriptionsbereichs gesteuert und gegebenenfalls eingeschränkt werden.

Aggregation von Proxy-Subskriptionen und Aggregation von Veröffentlichungen

Verteilte Publish/Subscribe-Veröffentlichungen und Proxy-Subskriptionen werden zusammengefasst, um die Anzahl der Nachrichten zu minimieren, die zwischen Publish/Subscribe-Warteschlangenmanagern übergeben werden.

Eine Proxy-Subskription ist eine Subskription, die von einem Warteschlangenmanager für Themen eingerichtet wird, die auf einem anderen Warteschlangenmanager veröffentlicht werden. Sie erstellen Proxy-

Subskriptionen nicht explizit, der Warteschlangenmanager tut dies in Ihrem Namen; siehe [„Wie funktioniert verteiltes Publish/Subscribe?“](#) auf Seite 54.

Sie können Warteschlangenmanager in einer Publish/Subscribe-Hierarchie oder in einem Publish/Subscribe-Cluster verbinden. Proxy-Subskriptionen fließen zwischen den verbundenen Warteschlangenmanagern. Proxy-Subskriptionen bewirken, dass Veröffentlichungen zu einem Thema, das von einem Publisher erstellt wurde, der mit einem Warteschlangenmanager verbunden ist, von Subskribenten zu diesem Thema empfangen werden, die mit anderen Warteschlangenmanagern verbunden sind; siehe [„Publish/Subscribe-Topologien“](#) auf Seite 67.

Eine Proxy-Subskription fließt zwischen Warteschlangenmanagern für jede einzelne Themenzeichenfolge, für die eine Subskription eingerichtet wurde.

Sie können den Fluss von Proxy-Subskriptionen und Veröffentlichungen zwischen verbundenen Warteschlangenmanagern mithilfe der **Topic** -Attribute `PUBSCOPE` und `SUBSCOPE` beschränken. Sie können den Ablauf von Proxy-Subskriptionen, die Platzhalterzeichen enthalten, auch einschränken, indem Sie das **Topic** -Attribut `WILDCARD` auf `BLOCK` setzen (siehe [„Regeln für Platzhalter“](#) auf Seite 59).

Proxy-Subskriptionen werden zwischen Warteschlangenmanagern asynchron zur Erstellung von Subskriptionen übertragen. Sie können die Latenzzeit für das Warten auf die Weitergabe einer Proxy-Subskription an alle verbundenen Warteschlangenmanager reduzieren, indem Sie das **Topic** -Attribut `PROXYSUB` für das Thema oder ein übergeordnetes Element eines Themas, das subskribiert wird, auf `FORCE` setzen (siehe [„Weitere Informationen zu Routing-Mechanismen“](#) auf Seite 57).

Zusammenfassung von Proxy-Subskriptionen

Proxy-Subskriptionen werden mithilfe eines Duplikateliminiierungssystems zusammengefasst. Für eine einzelne aufgelöste Themenzeichenfolge wird bei der ersten lokalen Subskription oder empfangenen Proxy-Subskription eine Proxy-Subskription gesendet. Nachfolgende Subskriptionen für dieselbe Themenzeichenfolge nutzen diese bestehende Proxy-Subskription.

Die Proxy-Subskription wird aufgehoben, nachdem die letzte lokale Subskription oder empfangene Proxy-Subskription aufgehoben wurde.

In Publish/Subscribe-Topologien mit vielen Tausenden von Subskriptionen für einzelne Themenzeichenfolgen oder wenn sich das Vorhandensein dieser Subskriptionen schnell ändern kann, muss der Systemaufwand für die Weitergabe der Proxy-Subskription berücksichtigt werden. Die einzelnen Proxy-Subskriptionen können konsolidiert werden, indem das Themenattribut **PROXYSUB** auf `FORCE` gesetzt wird. Weitere Informationen zu Routing-Mechanismen und zur Leistung von Clusterthemen finden Sie unter [„Weitere Informationen zu Routing-Mechanismen“](#) auf Seite 57.

Zusammenfassung von Veröffentlichungen

Wenn es auf einem Warteschlangenmanager mehrere Subskriptionen für dieselbe Themenzeichenfolge gibt, senden andere Warteschlangenmanager in der Publish/Subscribe-Topologie nur eine einzige Kopie von jeder Veröffentlichung für die betreffende Themenzeichenfolge. Bei Eingang der Nachricht liefert der lokale Warteschlangenmanager eine Kopie der Nachricht an alle übereinstimmenden Subskriptionen.

Wenn die Proxy-Subskriptionen Platzhalterzeichen enthalten, ist es möglich, dass mehrere Proxy-Subskriptionen mit der Themenzeichenfolge einer einzelnen Veröffentlichung übereinstimmen. Wird auf einem Warteschlangenmanager eine Nachricht veröffentlicht, die mit mehreren, von einem einzelnen verbundenen Warteschlangenmanager erstellte Proxy-Subskriptionen übereinstimmt, wird nur eine einzige Kopie der Veröffentlichung an den fernen Warteschlangenmanager weitergeleitet, um alle vorhandenen Proxy-Subskriptionen zu bedienen.

Weitere Informationen zu Routing-Mechanismen

Publish everywhere ist ein alternativer Weiterleitungsmechanismus für die Weiterleitung einzelner Proxy-Subskriptionen. Die Weiterleitung einzelner Proxy-Subskriptionen bedeutet, dass nur Veröffentlichungen mit einer übereinstimmenden Subskription für die Themenzeichenfolge an einen fernen Messaging-Server gesendet werden. *Publish überall* oder *Broadcast* leitet alle Veröffentlichungen, die auf einem Messaging-Server veröffentlicht werden, an alle anderen Messaging-Server in einem verteilten Publish/Subscribe-

Netz weiter. Die empfangenden Messaging-Server stellen dann die Veröffentlichungen zu, die lokalen Subskriptionen entsprechen.

Jeder Mechanismus hat seine Vorzüge, aber auch Einschränkungen.

Weiterleitung einzelner Proxy-Subskriptionen

Dieser Mechanismus führt zu dem geringsten Datenverkehr zwischen den Veröffentlichungen des Warteschlangenmanagers, da nur die Veröffentlichungen gesendet werden, die mit Subskriptionen auf einem Warteschlangenmanager übereinstimmen.

Es gilt jedoch auch Folgendes:

- Jede einzelne Themenzeichenfolge, die subskribiert wird, führt zu einer Proxy-Subskription, die an alle anderen Warteschlangenmanager in der Publish/Subscribe-Topologie gesendet wird. Dieser Messaging-Aufwand kann von Bedeutung sein, wenn viele Tausende von Subskriptionen erstellt oder gelöscht werden müssen (z. B. alle nicht permanenten Subskriptionen nach einem Neustart eines Warteschlangenmanagers) oder wenn sich die Gruppe von Subskriptionen schnell ändert und jede zu einer anderen Themenzeichenfolge gehört.
- Proxy-Subskriptionen werden über asynchrones Messaging an andere Warteschlangenmanager übertragen. Daher kommt es zu einer Verzögerung zwischen der Erstellung einer Subskription und der Erstellung, Zustellung und Verarbeitung von Proxy-Subskriptionen durch die anderen Warteschlangenmanager. Nachrichten, die in diesen Warteschlangenmanagern in diesem Intervall veröffentlicht werden, werden nicht an die ferne Subskription zugestellt.

Überall veröffentlichen

Mit diesem Mechanismus:

- Es gibt keinen Proxy-Subskriptionsaufwand pro Themenzeichenfolge auf dem System, was bedeutet, dass die schnelle Erstellung, Löschung oder Änderung von Subskriptionen nicht zu einer erhöhten Netzauslastung und -verarbeitung führt.
- Es gibt keine Verzögerung zwischen der Erstellung einer Subskription und Veröffentlichungen, die an einen Warteschlangenmanager übertragen werden, da sie immer an alle Warteschlangenmanager übertragen werden. Daher gibt es kein Fenster, in dem Veröffentlichungen nicht an neu erstellte ferne Subskriptionen zugestellt werden.

Es gilt jedoch auch Folgendes:

- Alle Veröffentlichungen werden an alle Warteschlangenmanager in der Publish/Subscribe-Topologie gesendet, was potenziell zu einem übermäßigen Datenaustausch im Netz führen kann, wenn die Veröffentlichungen nicht über übereinstimmende Subskriptionen auf jedem Warteschlangenmanager verfügen.

Sie können den Mechanismus "überall veröffentlichen" verwenden, wenn Sie erwarten, dass eine Veröffentlichung von einem beträchtlichen Teil Ihrer Warteschlangenmanager im Cluster oder in der Hierarchie subskribiert wird, oder wenn der Aufwand für die Proxy-Subskription aufgrund der Häufigkeit von Subskriptionsänderungen zu hoch ist. Diese Arbeitsmethode kann in diesen Fällen effektiver sein als in anderen Fällen, in denen ein erhöhter Messaging-Datenverkehr auftritt, wenn Veröffentlichungen an alle Warteschlangenmanager und nicht an die Warteschlangenmanager mit übereinstimmenden Subskriptionen gesendet werden.

Ein Mechanismus "überall veröffentlichen" kann in verteilten IBM WebSphere MQ -Publish/Subscribe-Topologien aktiviert werden, indem das Attribut **PROXYSUB** für ein übergeordnetes Themenobjekt auf FORCE gesetzt wird.

Details zum Inaktivieren einzelner Proxy-Subskriptionen finden Sie unter „[Einzelne Proxy-Subskriptionen inaktivieren](#)“ auf Seite 82.

Wenn diese erzwungene Proxy-Subskription in der gesamten Topologie weitergegeben wird, erhalten alle neuen Subskriptionen sofort alle Veröffentlichungen von anderen verbundenen Warteschlangenmanagern, ohne dass Latenzzeiten auftreten.

Bei der Konfiguration eines solchen Systems ist Vorsicht geboten. Keine Themenobjekte unterhalb eines Themas, für das **PROXYSUB** auf FORCE gesetzt ist, dürfen sich in einem anderen Cluster oder Hierarchiedatenstrom befinden als der Knoten, für den **PROXYSUB** auf FORCE gesetzt ist. Ebenso dürfen untergeord-

nete Themenobjekte das Attribut **WILDCARD** nicht auf BLOCK setzen. In beiden Fällen kann dies dazu führen, dass veröffentlichte Nachrichten nicht ordnungsgemäß von einem Warteschlangenmanager zu einem anderen übertragen werden.

Selbst wenn **PROXYSUB** auf FORCE gesetzt ist, wird eine Proxy-Subskription für jede einzelne Themenzeichenfolge, die subskribiert wird, weiterhin weitergegeben. Wenn die Anzahl und Häufigkeit von Subskriptionen hoch genug ist, um einen erheblichen Systemaufwand zu verursachen, können sie für alle Themen auf einem Warteschlangenmanager inaktiviert werden. Details zum Inaktivieren einzelner Proxy-Subskriptionen finden Sie unter [„Einzelne Proxy-Subskriptionen inaktivieren“](#) auf Seite 82.

Multicast-und Subskriptionslatenz

Die Subskriptionslatenzzeit und die Option PROXYSUB (FORCE) können zur Verwaltung einer Proxy-Subskription verwendet werden.

Es besteht beispielsweise das potenzielle Problem, dass eine Proxy-Subskription von QM_B zu QM_A aufgehoben wird, nachdem alle Subskribenten getrennt wurden. Diese Situation ist möglicherweise nicht erwünscht, wenn der Multicastverkehr fortgesetzt werden soll, auch wenn die Unicastverbindung zum Warteschlangenmanager beendet wird. Multicast für WebSphere MQ verwaltet die Proxy-Subskription für kurze Zeit, falls ein neuer Subskribent eine Verbindung herstellt, indem jeder Proxy-Subskription eine Latenzzeit von einigen Minuten hinzugefügt wird, damit sie nicht zum Zeitpunkt der Beendigung des letzten Subskribenten aufgehoben werden.

Sie können auch die Option PROXYSUB (FORCE) für das Thema verwenden, um sicherzustellen, dass eine ausstehende Proxy-Subskription immer aussteht. Sie müssen sicherstellen, dass die Nachrichten, die über die Warteschlangen fließen, für die meiste Zeit, in der die Subskription aktiv ist, von mindestens einem Subskribenten benötigt werden. Wenn PROXYSUB (FORCE) festgelegt ist, wird möglicherweise eine Proxy-Subskription vor der ersten lokalen Subskription oder der ersten empfangenen Proxy-Subskription gesendet und nicht storniert, auch wenn die letzte lokale Subskription oder die letzte empfangene Proxy-Subskription storniert wurde.

Wenn die Subskription noch nicht erstellt wurde, können Sie mithilfe der Peer-to-Peer-Kommunikation sicherstellen, dass die Nachrichtenübertragungen fortgesetzt werden. Weitere Informationen finden Sie unter [Hochverfügbarkeit für Multicasting](#).

Regeln für Platzhalter

Platzhalterzeichen in Proxy-Subskriptionen werden für die Verwendung von Topic-Platzhalterzeichen konvertiert.

Wenn eine Subskription für ein Platzhalterzeichen empfangen wird, kann dies ein Zeichen sein, das von WebSphere MQ Version 6.0 verwendet wird. Es kann sich auch um ein Thema handeln, das von WebSphere Message Broker Version 6.0 und WebSphere MQ Version 7.0 verwendet wird.

- Zeichenplatzhalter verwenden `*`, um ein beliebiges Zeichen darzustellen, einschließlich `/`.
- Themenplatzhalter verwenden `#`, um einen Teil des Topicbereichs zwischen `/` Zeichen darzustellen.

In WebSphere MQ Version 7.0 werden alle Proxy-Subskriptionen für die Verwendung von Topic-Platzhalterzeichen konvertiert. Wenn ein Platzhalterzeichen gefunden wird, wird es durch ein `#`-Zeichen ersetzt, zurück zur nächsten `/`. Zum Beispiel wird `/aaa/bbb/c*d` umgewandelt in `/aaa/bbb/#`. Die Umwandlung führt dazu, dass ferne Warteschlangenmanager etwas mehr Veröffentlichungen senden, als explizit subskribiert wurden. Die zusätzlichen Veröffentlichungen werden vom lokalen Warteschlangenmanager ausgefiltert, wenn er die Veröffentlichungen an seine lokalen Subskribenten weitergibt.

Verwendung von Platzhaltern mit der Eigenschaft WILDCARD steuern

Mit der MQSC-Eigenschaft **Topic** WILDCARD oder der entsprechenden PCF-Eigenschaft `Topic wildcardOperation` können Sie die Zustellung von Veröffentlichungen an Subskribentenanwendungen steuern, die Namen von Platzhalterthemenzeichenfolgen verwenden. Die Eigenschaft WILDCARD kann einen von zwei möglichen Werten haben:

WILDCARD

Aktionen von Subskriptionen mit Platzhaltern bezüglich dieses Themas.

PASSTHRU

Subskriptionen für ein Thema mit Platzhalter, das weniger spezifisch ist als die Themenzeichenfolge für dieses Themenobjekt, empfangen Veröffentlichungen zu diesem Thema und zu spezifischeren Themenzeichenfolge.

BLOCK

Subskriptionen für ein Thema mit Platzhalter, das weniger spezifisch ist als die Themenzeichenfolge für dieses Themenobjekt, empfangen keine Veröffentlichungen zu diesem Thema und zu spezifischeren Themenzeichenfolge.

Der Wert für dieses Attribut wird bei der Definition von Subskriptionen verwendet. Wenn Sie dieses Attribut ändern, ist die Gruppe von Themen, die bereits durch vorhandene Subskriptionen abgedeckt sind, nicht durch die Änderung betroffen. Dieses Szenario gilt auch, wenn sich durch die Erstellung oder das Löschen von Themenobjekten die Topologie ändert; die Themen mit Subskriptionen, die nach der Änderung des Attributs WILDCARD erstellt wurden, werden mit der geänderten Topologie erstellt. Wenn die Themen mit den vorhandenen Subskriptionen übereinstimmen sollen, müssen Sie den Warteschlangenmanager neu starten.

Führen Sie die Schritte im Beispiel „Beispiel: Publish/Subscribe-Cluster für Sport erstellen“ auf Seite 46 aus, um die in [Abbildung 10](#) auf Seite 42 gezeigte Themenstruktur zu erstellen.

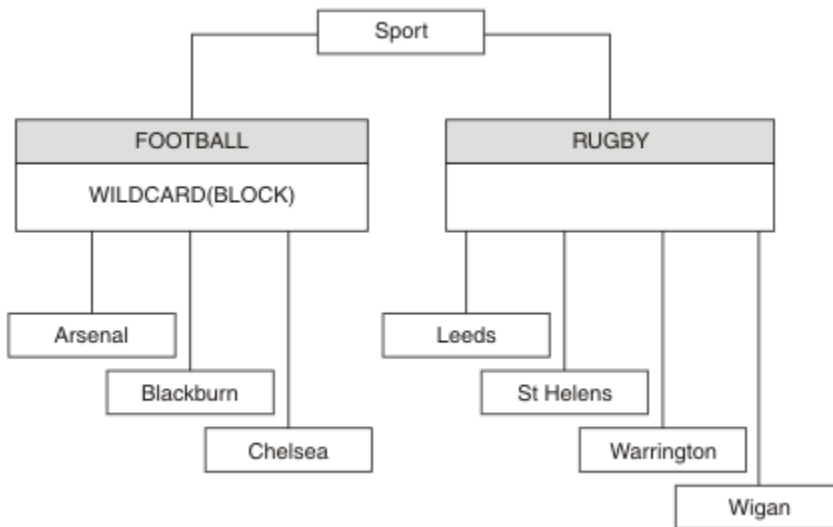


Abbildung 23. Themenstruktur, die die WILDCARD-Eigenschaft BLOCK verwendet

Ein Subskribent, der die Platzhalter-Themenzeichenfolge # verwendet, empfängt alle Veröffentlichungen für das Thema Sport und die Unterverzeichnisstruktur Sport/Rugby. Der Subskribent empfängt keine Veröffentlichungen für die untergeordnete Baumstruktur Sport/Football, weil der Wert der Eigenschaft WILDCARD des Themas Sport/Football BLOCK ist.

Die Standardeinstellung lautet PASSTHRU. Sie können den Wert der Eigenschaft WILDCARD PASSTHRU auf Knoten in der Baumstruktur Sport setzen. Wenn die Knoten nicht über den Eigenschaftswert WILDCARD verfügen BLOCK, ändert die Einstellung PASSTHRU nicht das Verhalten, das von Subskribenten für Knoten in der Sports -Baumstruktur beobachtet wird.

Erstellen Sie im Beispiel Subskriptionen, um zu sehen, wie sich die Platzhalterereinstellung auf die bereitgestellten Veröffentlichungen auswirkt (siehe dazu [Abbildung 14](#) auf Seite 48). Führen Sie den Publish-Befehl in [Abbildung 17](#) auf Seite 49 aus, um einige Veröffentlichungen zu erstellen.

```
pub QMA
```

Abbildung 24. Für QMA publizieren

Die Ergebnisse werden in Tabelle 3 auf Seite 42 gezeigt. Beachten Sie, wie die Einstellung des WILD-CARD-Eigenschaftswerts BLOCK verhindert, dass Subskriptionen mit Platzhaltern Veröffentlichungen für Themen empfangen, die innerhalb des Platzhalterumfangs liegen.

Tabelle 6. Auf QMA empfangene Veröffentlichungen			
Subskription	Themenzeichenfolge	Empfangene Veröffentlichungen	Hinweise
SPORTS	Sports/#	Sports Sports/Rugby Sports/Rugby/Leeds	Alle Veröffentlichungen zur Unterverzeichnisstruktur 'Football', die von WILDCARD (BLOCK) auf Sports/Football blockiert werden
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) on Sports/Football verhindert Platzhaltersubskription unter Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Standardwert WILDCARD für 'Sport/Rugby' verhindert keine Platzhaltersubskription auf Leeds.

Anmerkung:

Angenommen, eine Subskription hat ein Platzhalterzeichen, das einem Themenobjekt mit dem Wert BLOCK der Eigenschaft WILDCARD entspricht. Wenn die Subskription auch eine Themenzeichenfolge rechts neben dem entsprechenden Platzhalter hat, empfängt die Subskription niemals Veröffentlichungen. Die nicht geblockten Veröffentlichungen sind Veröffentlichungen für Themen, die dem geblockten Platzhalter übergeordnet sind. Veröffentlichungen für Themen, die dem Thema mit dem Eigenschaftswert BLOCK untergeordnet sind, werden vom Platzhalter geblockt. Subskriptionsthemen-Zeichenfolgen, die ein Thema rechts neben dem Platzhalter einschließen, empfangen daher niemals passende Veröffentlichungen.

Wenn der Eigenschaftswert WILDCARD auf BLOCK gesetzt wird, ist keine Subskription mithilfe einer Themenzeichenfolge möglich, die Platzhalter beinhaltet. Eine solche Subskription ist normal. Die Subskription hat ein explizites Thema, das dem Thema mit einem Themenobjekt entspricht, das den WILDCARD-Eigenschaftswert BLOCK hat. Sie verwendet Platzhalter für Themen, die dem WILDCARD-Eigenschaftswert BLOCK über- oder untergeordnet sind. Im Beispiel in [Abbildung 10](#) auf Seite 42 kann eine Subskription wie Sports/Football/# Veröffentlichungen empfangen.

Platzhalter und Cluster-Topics

Definitionen von Cluster-Topics werden an jeden Warteschlangenmanager in einem Cluster weitergegeben. Die Subskription eines Cluster-Topics auf einem Warteschlangenmanager in einem Cluster führt dazu, dass der Warteschlangenmanager Proxy-Subskriptionen erstellt. Eine Proxy-Subskription wird in jedem Warteschlangenmanager im Cluster erstellt. Subskriptionen, die Themenzeichenfolgen mit Platzhaltern verwenden, kombiniert mit Cluster-Topics, lassen das Verhalten schwer vorhersagen. Das Verhalten wird im folgenden Beispiel erläutert.

In der Clusterkonfiguration für das Beispiel „[Beispiel: Publish/Subscribe-Cluster für Sport erstellen](#)“ auf Seite 46 hat QMB dieselbe Gruppe von Subskriptionen wie QMA, aber QMB hat keine Veröffentlichungen empfangen, nachdem der Publisher in QMA veröffentlicht wurde (siehe [Abbildung 11](#) auf Seite 42). Obwohl die Themen Sports/Football und Sports/Rugby Clusterthemen sind, verweisen die in `fullsubs.tst` definierten Subskriptionen nicht auf ein Clusterthema. Es werden keine Proxy-Subskriptionen von QMB an QMA weitergegeben. Ohne Proxy-Subskriptionen werden keine Veröffentlichungen für QMA an QMB weitergeleitet.

Einige Subskriptionen, wie z. B. Sports/#/Leeds, scheinen möglicherweise auf ein Clusterthema zu verweisen, in diesem Fall Sports/Rugby. Die Sports/#/Leeds-Subskription wird tatsächlich in das Themenobjekt SYSTEM.BASE.TOPIC aufgelöst.

Die Regel zum Auflösen des Themenobjekts, das von einer Subskription wie Sports/#/Leeds referenziert wird, lautet wie folgt. Schneiden Sie die Themenzeichenfolge bis zum ersten Platzhalter ab. Suchen Sie links über die Themenzeichenfolge nach dem ersten Thema mit einem zugehörigen Verwaltungsthemenobjekt. Das Themenobjekt kann einen Clusternamen angeben oder ein lokales Themenobjekt definieren. Im Beispiel Sports/#/Leeds ist die Themenzeichenfolge nach dem Abschneiden Sports, die kein Themenobjekt hat, und Sports/#/Leeds übernimmt daher von SYSTEM.BASE.TOPIC, einem lokalen Themenobjekt.

Um zu sehen, wie die Subskription von Cluster-Topics die Weitergabe von Platzhaltern ändern kann, führen Sie das Stapelscript `upsubs.bat` aus. Das Script löscht die Subskriptionswarteschlangen und fügt die Subskriptionen des Cluster-Topics zu `fullsubs.tst` hinzu. Führen Sie `puba.bat` erneut aus, um einen Stapel von Veröffentlichungen zu erstellen (siehe [Abbildung 11](#) auf Seite 42).

[Tabelle 4](#) auf Seite 44 zeigt das Ergebnis, nachdem zwei neue Subskriptionen dem Warteschlangenmanager hinzugefügt wurden, in dem die Veröffentlichungen veröffentlicht wurden. Wie erwartet, empfangen die neuen Subskriptionen je eine Veröffentlichung, und die Anzahl der von den anderen Subskriptionen empfangenen Veröffentlichungen ist unverändert. Die unerwarteten Resultate treten im anderen Cluster-Warteschlangenmanager auf, siehe [Tabelle 5](#) auf Seite 44.

<i>Tabelle 7. Auf QMA empfangene Veröffentlichungen</i>			
Subskription	Themenzeichenfolge	Empfangene Veröffentlichungen	Hinweise
SPORTS	Sports/#	Sports Sports/Rugby Sports/Rugby/Leeds	Alle Veröffentlichungen zur Unterverzeichnisstruktur 'Football', die von WILDCARD (BLOCK) auf Sports/Football blockiert werden
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) on Sports/Football verhindert Platzhaltersubskription unter Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Standardwert WILDCARD für 'Sport/Rugby' verhindert keine Platzhaltersubskription auf Leeds.
FARSENAL	Sports/Football/Arsenal	Sports/Football/Arsenal	Arsenal empfängt eine Veröffentlichung, da die Subskription keinen Platzhalter hat.
FLEEDS	Sports/Rugby/Leeds	Sports/Rugby/Leeds	Leeds würde für jedes Ereignis eine Veröffentlichung empfangen.

[Tabelle 5](#) auf Seite 44 zeigt die Resultate, nachdem die beiden neuen Subskriptionen zu QMB hinzugefügt und auf QMA veröffentlicht wurden. QMB empfing keine Veröffentlichungen ohne diese beiden neuen Subskriptionen. Wie erwartet empfangen die beiden neuen Subskriptionen Veröffentlichungen, da Sports/FootBall und Sports/Rugby Clusterthemen sind. QMB weitergeleitete Proxy-Subskriptionen für Sports/Football/Arsenal und Sports/Rugby/Leeds an QMA, die dann die Veröffentlichungen an QMB gesendet haben

Das unerwartete Ergebnis ist, dass die beiden Subskriptionen Sports/# und Sports/#/Leeds, die zuvor keine Veröffentlichungen empfangen haben, jetzt Veröffentlichungen empfangen. Dies liegt daran, dass die Sports/Football/Arsenal- und Sports/Rugby/Leeds-Veröffentlichungen, die für die anderen Subskriptionen an QMB weitergeleitet werden, jetzt für jeden Subskribenten verfügbar sind, der QMB zugeordnet ist. Folglich empfangen die Subskriptionen für die lokalen Themen Sports/# und Sports/#/Leeds die Veröffentlichung Sports/Rugby/Leeds. Sports/#/Arsenal erhält weiterhin keine Veröffentlichung, weil für Sport/Fußball die Eigenschaft WILDCARD auf BLOCK gesetzt ist.

Subskription	Themenzeichenfolge	Empfangene Veröffentlichungen	Hinweise
SPORTS	Sports/#	Sports/Rugby/Leeds	All publications to Football subtree blocked by WILDCARD (BLOCK) on Sports/Football
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) on Sports/Football verhindert Platzhaltersubskription unter Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Der Standardwert WILDCARD für Sport/Rugby verhindert nicht das Platzhalterabonnement für Leeds .
FARSENAL	Sports/Football/Arsenal	Sports/Football/Arsenal	Arsenal empfängt eine Veröffentlichung, da die Subskription keinen Platzhalter hat.
FLEEDS	Sports/Rugby/Leeds	Sports/Rugby/Leeds	Leeds würde für jedes Ereignis eine Veröffentlichung empfangen.

Bei den meisten Anwendungen ist es nicht erwünscht, dass eine Subskription das Verhalten einer anderen Subskription beeinflusst. Eine wichtige Verwendung der Eigenschaft WILDCARD mit dem Wert BLOCK ist es, dass sich die Subskriptionen für dieselbe Themenzeichenfolge mit Platzhalterzeichen einheitlich verhalten. Unabhängig davon, ob sich die Subskription in demselben Warteschlangenmanager wie der Publisher oder in einem anderen Warteschlangenmanager befindet, sind die Resultate der Subskription dieselben.

Platzhalter und Datenströme

WebSphere MQ Version 6-Datenströme werden Themen von WebSphere MQ Version 7 zugeordnet (siehe „Datenströme und Themen“ auf Seite 49). In der Standardzuordnung, die von **strmqbrk** in Version 7 ausgeführt wird, werden alle Themen im Datenstrom Sports dem Thema Sports zugeordnet. Alle Themen im Datenstrom Business werden dem Thema Business zugeordnet.

Eine Subskription in WebSphere MQ Version 6 für * im Datenstrom Sports empfängt alle Veröffentlichungen in der Baumstruktur Sports und keine Veröffentlichungen in der Baumstruktur Business. Dieselbe Subskription in Version 7 würde alle Veröffentlichungen in der Baumstruktur Sports und alle Veröffentlichungen in der Baumstruktur Business empfangen. Um dieses Verhalten zu blockieren, legt **strmqbrk** die Eigenschaft WILDCARD fest, wenn Datenströme auf Version 7 migriert werden. Sie setzt sie auf den Wert BLOCK für jedes der Themen der höchsten Ebene, die aus einem Datenstrom migriert wurden. Die Eigenschaft WILDCARD von Sports und Business wird durch die Konvertierung aus den Streams der Version 6 mit den Namen Sports und Business auf den Wert BLOCK gesetzt.

Bei einer neuen Anwendung, die für die Publish/Subscribe-API geschrieben wurde, hat dies zur Folge, dass eine Subskription für * keine Veröffentlichungen empfängt. Um alle Sportveröffentlichungen zu erhalten, müssen Sie Sports/* oder Sports/# und ähnlich für Business -Veröffentlichungen subscribieren.

Das Verhalten einer vorhandenen Publish/Subscribe-Anwendung in der Warteschlange ändert sich nicht, wenn der Publish/Subscribe-Broker auf WebSphere MQ Version 7 migriert wird. Die Eigenschaft **Stream-Name** im Befehl **Publish**, **Register Publisher** oder **Subscriber** wird dem Namen des Themas zugeordnet, in das der Datenstrom migriert wurde.

Platzhalter und Subskriptionspunkte

WebSphere Message Broker-Subskriptionspunkte werden Themen von WebSphere MQ Version 7 zugeordnet; siehe „Subskriptionspunkte und Themen“ auf Seite 51. In der Standardzuordnung, die von

migmqbrk in Version 7 ausgeführt wird, werden alle Themen im Subskriptionspunkt Sports dem Thema Sportszugeordnet. Alle Themen im Subskriptionspunkt Business würden dem Thema Businesszugeordnet.

Eine Subskription für WebSphere Message Broker Version 6 für * im Sports -Subskriptionspunkt empfängt alle Veröffentlichungen in der Baumstruktur Sports und keine Veröffentlichungen in der Baumstruktur Business . Dieselbe Subskription in Version 7 würde alle Veröffentlichungen in der Baumstruktur Sports und alle Veröffentlichungen in der Baumstruktur Business empfangen. Um dieses Verhalten zu blockieren, legt **migmqbrk** die Eigenschaft WILDCARD fest, wenn Subskriptionspunkte auf Version 7 migriert werden. Sie setzt sie auf den Wert BLOCK für jedes der Themen der höchsten Ebene, die von einem Subskriptionspunkt migriert werden. Die Eigenschaft WILDCARD von Sports und Business wird durch die Konvertierung aus den WebSphere Message Broker-Subskriptionspunkten Sports und Business auf den Wert BLOCK gesetzt.

Wenn eine neue Anwendung für die Publish/Subscribe-API geschrieben wird, empfängt eine Subskription von * keine Veröffentlichungen. Um alle Sportveröffentlichungen zu erhalten, müssen Sie Sports/*oder Sports/# und ähnlich für Business -Veröffentlichungen abonnieren.

Das Verhalten einer vorhandenen Publish/Subscribe-Anwendung in der Warteschlange ändert sich nicht, wenn der Publish/Subscribe-Broker auf WebSphere MQ Version 7 migriert wird. Die Eigenschaft **Sub-Point** im Befehl **Publish, Register Publisher** oder **Subscriber** wird dem Namen des Themas zugeordnet, auf das die Subskription migriert wurde.

Beispiel: Publish/Subscribe-Cluster für Sport erstellen

Die folgenden Schritte erstellen einen Cluster, CL1, mit vier Warteschlangenmanagern: zwei vollständigen Repositories, CL1A und CL1B , und zwei Teilrepositories, QMA und QMB. Die vollständigen Repositories enthalten nur Clusterdefinitionen. QMA wird als Cluster-Topic-Host festgelegt. Permanente Subskriptionen sind in QMA und QMB definiert.

Anmerkung: Das Beispiel ist für Windowscodiert. Wenn Sie das Beispiel auf anderen Plattformen konfigurieren und testen möchten, müssen Sie den Programmcode für [Create qmgrs.bat](#) und [create pub.bat](#) ändern.

1. Erstellen Sie die Scriptdateien.
 - a. [Create topics.tst](#)
 - b. [Create wildsubs.tst](#)
 - c. [Create fullsubs.tst](#)
 - d. [Create qmgrs.bat](#)
 - e. [create pub.bat](#)
2. Führen Sie [Create qmgrs.bat](#) aus, um die Konfiguration zu erstellen.

```
qmgrs
```

Erstellen Sie die Themen in [Abbildung 10 auf Seite 42](#). Das Script in [Abbildung 5](#) erstellt die Clusterthemen Sports/Football und Sports/Rugby.

Anmerkung: Die Option REPLACE ersetzt nicht die TOPICSTR-Eigenschaften eines Themas. TOPICSTR ist eine Eigenschaft, die im Beispiel gezielt variiert wird, um verschiedene Themenstrukturen zu testen. Wenn Sie ein Thema ändern möchten, müssen Sie das Thema zuerst löschen.


```

DELETE TOPIC ('Sports')
DELETE TOPIC ('Football')
DELETE TOPIC ('Arsenal')
DELETE TOPIC ('Blackburn')
DELETE TOPIC ('Chelsea')
DELETE TOPIC ('Rugby')
DELETE TOPIC ('Leeds')
DELETE TOPIC ('Wigan')
DELETE TOPIC ('Warrington')
DELETE TOPIC ('St. Helens')

DEFINE TOPIC ('Sports')      TOPICSTR('Sports')
DEFINE TOPIC ('Football')   TOPICSTR('Sports/Football') CLUSTER(CL1) WILDCARD(BLOCK)
DEFINE TOPIC ('Arsenal')    TOPICSTR('Sports/Football/Arsenal')
DEFINE TOPIC ('Blackburn')  TOPICSTR('Sports/Football/Blackburn')
DEFINE TOPIC ('Chelsea')    TOPICSTR('Sports/Football/Chelsea')
DEFINE TOPIC ('Rugby')      TOPICSTR('Sports/Rugby') CLUSTER(CL1)
DEFINE TOPIC ('Leeds')      TOPICSTR('Sports/Rugby/Leeds')
DEFINE TOPIC ('Wigan')      TOPICSTR('Sports/Rugby/Wigan')
DEFINE TOPIC ('Warrington') TOPICSTR('Sports/Rugby/Warrington')
DEFINE TOPIC ('St. Helens') TOPICSTR('Sports/Rugby/St. Helens')

```

Abbildung 25. Themen löschen und erstellen: topics.tst

Anmerkung: Löschen Sie die Themen, da REPLACE keine Themenzeichenfolgen ersetzt.

Erstellen Sie Subskriptionen mit Platzhaltern. Die Platzhalter entsprechen den Themen mit Themenobjekten in [Abbildung 10](#) auf Seite 42. Erstellen Sie für jede Subskription eine Warteschlange. Beim Ausführen bzw. erneuten Ausführen des Scripts werden die Warteschlangen geleert, und die Subskriptionen werden gelöscht.

Anmerkung: Die Option REPLACE ersetzt nicht die die Eigenschaften TOPICOBJ oder TOPICSTR einer Subskription. TOPICOBJ oder TOPICSTR sind die Eigenschaften, die im Beispiel sinnvoll variiert werden, um verschiedene Subskriptionen zu testen. Um sie zu ändern, löschen Sie zunächst die Subskription.

```

DEFINE QLOCAL(QSPORTS) REPLACE
DEFINE QLOCAL(QSARSENAL) REPLACE
DEFINE QLOCAL(QSLEEDS) REPLACE
CLEAR QLOCAL(QSPORTS)
CLEAR QLOCAL(QSARSENAL)
CLEAR QLOCAL(QSLEEDS)

DELETE SUB (SPORTS)
DELETE SUB (SARSENAL)
DELETE SUB (SLEEDS)
DEFINE SUB (SPORTS) TOPICSTR('Sports/#') DEST(QSPORTS)
DEFINE SUB (SARSENAL) TOPICSTR('Sports+/Arsenal') DEST(QSARSENAL)
DEFINE SUB (SLEEDS) TOPICSTR('Sports+/Leeds') DEST(QSLEEDS)

```

Abbildung 26. Platzhaltersubskriptionen erstellen: wildsubs.tst

Erstellen Sie Subskriptionen, die die Cluster-Topic-Objekte referenzieren.

Anmerkung:

Das Begrenzungszeichen / wird automatisch zwischen die von TOPICOBJ referenzierte Themenzeichenfolge und die von TOPICSTR definierte Themenzeichenfolge eingefügt.

Die Definition `DEFINE SUB(FARSENAL) TOPICSTR('Sports/Football/Arsenal') DEST(QFARSENAL)` erstellt dieselbe Subskription. Die Verwendung von TOPICOBJ ist eine zeiteffiziente Methode, bereits definierte Themenzeichenfolgen zu referenzieren. Die erstellte Subskription referenziert das Themenobjekt nicht länger.

```

DEFINE QLOCAL(QFARSENAL) REPLACE
DEFINE QLOCAL(QRLEEDS) REPLACE
CLEAR QLOCAL(QFARSENAL)
CLEAR QLOCAL(QRLEEDS)

DELETE SUB (FARSENAL)
DELETE SUB (RLEEDS)
DEFINE SUB (FARSENAL) TOPICOBJ('Football') TOPICSTR('Arsenal') DEST(QFARSENAL)
DEFINE SUB (RLEEDS) TOPICOBJ('Rugby') TOPICSTR('Leeds') DEST(QRLEEDS)

```

Abbildung 27. Subskriptionen löschen und erstellen: fullsubs.tst

Erstellen Sie einen Cluster mit zwei Repositorys. Erstellen Sie zwei Teilrepositorys zum Veröffentlichen und Abonnieren. Führen Sie das Script erneut aus, um alles zu löschen und neu zu starten. Das Script erstellt zudem die Themenhierarchie und die ursprünglichen Platzhaltersubskriptionen.

Anmerkung:

Schreiben Sie auf anderen Plattformen ein ähnliches Script, oder geben Sie alle Befehle ein. Mit einem Script lässt sich alles schneller löschen und mit einer identischen Konfiguration erneut starten.

```

@echo off
set port.CL1B=1421
set port.CL1A=1420
for %%A in (CL1A CL1B QMA QMB) do call :createQM %%A
call :configureQM CL1A CL1B %port.CL1B% full
call :configureQM CL1B CL1A %port.CL1A% full
for %%A in (QMA QMB) do call :configureQM %%A CL1A %port.CL1A% partial
for %%A in (topics.tst wildsubs.tst) do runmqsc QMA < %%A
for %%A in (wildsubs.tst) do runmqsc QMB < %%A
goto:eof

:createQM
echo Configure Queue manager %1
endmqm -p %1
for %%B in (dlt crt str) do %%Bmqm %1
goto:eof

:configureQM
if %1==CL1A set p=1420
if %1==CL1B set p=1421
if %1==QMA set p=1422
if %1==QMB set p=1423
echo configure %1 on port %p% connected to repository %2 on port %3 as %4 repository
echo DEFINE LISTENER(LST%1) TRPTYPE(TCP) PORT(%p%) CONTROL(QMGR) REPLACE | runmqsc %1
echo START LISTENER(LST%1) | runmqsc %1
if full==%4 echo ALTER QMGR REPOS(CL1) DEADQ(SYSTEM.DEAD.LETTER.QUEUE) | runmqsc %1
echo DEFINE CHANNEL(TO.%2) CHLTYPE(CLUSSDR) TRPTYPE(TCP) CONNAME('LOCALHOST(%3)') CLUSTER(CL1)
REPLACE | runmqsc %1
echo DEFINE CHANNEL(TO.%1) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CONNAME('LOCALHOST(%p%)') CLUSTER(CL1)
REPLACE | runmqsc %1
goto:eof

```

Abbildung 28. Warteschlangenmanager erstellen: qmgrs.bat

Aktualisieren Sie die Konfiguration durch Hinzufügen der Subskriptionen zu den Cluster-Topics.

```

@echo off
for %%A in (QMA QMB) do runmqsc %%A < wildsubs.tst
for %%A in (QMA QMB) do runmqsc %%A < upsubs.tst

```

Abbildung 29. Subskriptionen aktualisieren: upsubs.bat

Führen Sie pub.bat mit einem Warteschlangenmanager als Parameter aus, um Nachrichten zu veröffentlichen, die die Veröffentlichungsthema-Zeichenfolge enthalten. Pub.bat verwendet das Beispielprogramm amqspub.

```

@echo off
@rem Provide queue manager name as a parameter
set S=Sports
set S=6 Sports/Football Sports/Football/Arsenal
set S=6 Sports/Rugby Sports/Rugby/Leeds
for %%B in (6) do echo %%B | amqspub %%B %1

```

Abbildung 30. Veröffentlichen: pub.bat

Publish/Subscribe-Topologien

Eine *Publish/Subscribe-Topologie* besteht aus Warteschlangenmanagern und den Verbindungen zwischen ihnen, die Publish/Subscribe-Anwendungen unterstützen.

Eine Publish/Subscribe-Anwendung kann aus einem Netz von Warteschlangenmanagern bestehen, die miteinander verbunden sind. Die Warteschlangenmanager können sich alle auf dem gleichen physischen System befinden oder über mehrere physische Systeme verteilt sein. Durch die Verbindung von Warteschlangenmanagern können Veröffentlichungen von einer Anwendung über einen beliebigen Warteschlangenmanager im Netz empfangen werden.

Dies bietet die folgenden Vorteile:

- Clientanwendungen können mit einem in der Nähe befindlichen Warteschlangenmanager und nicht mit einem fernen Warteschlangenmanager kommunizieren, wodurch die Antwortzeiten verbessert werden.
- Durch die Verwendung mehrerer Warteschlangenmanager können mehrere Subskribenten unterstützt werden.

Sie können Warteschlangenmanager, die Publish/Subscribe-Messaging durchführen, auf zwei verschiedene Arten anordnen: Cluster und Hierarchien. Beispiele für einen einfachen Cluster und eine einfache Hierarchie finden Sie unter [Abbildung 31 auf Seite 67](#) und [Abbildung 32 auf Seite 68](#). Weitere Informationen zu diesen beiden Topologien sowie Informationen zu den für Sie am besten geeigneten Topologien finden Sie in den Informationen in diesem Abschnitt der Produktdokumentation.

Es ist möglich, beide Topologien in Kombination zu verwenden, indem Cluster in einer Hierarchie miteinander verknüpft werden.

Cluster

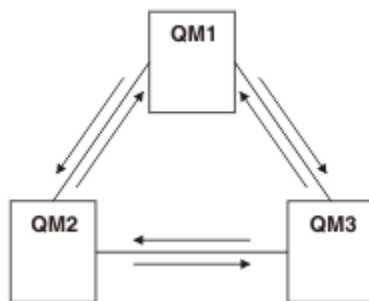


Abbildung 31. Einfacher Publish/Subscribe-Cluster

Hierarchy

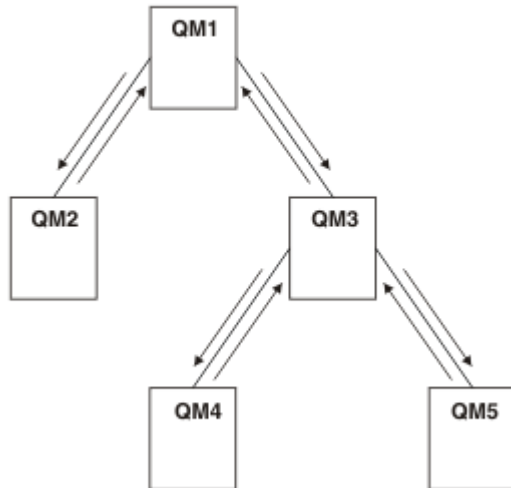


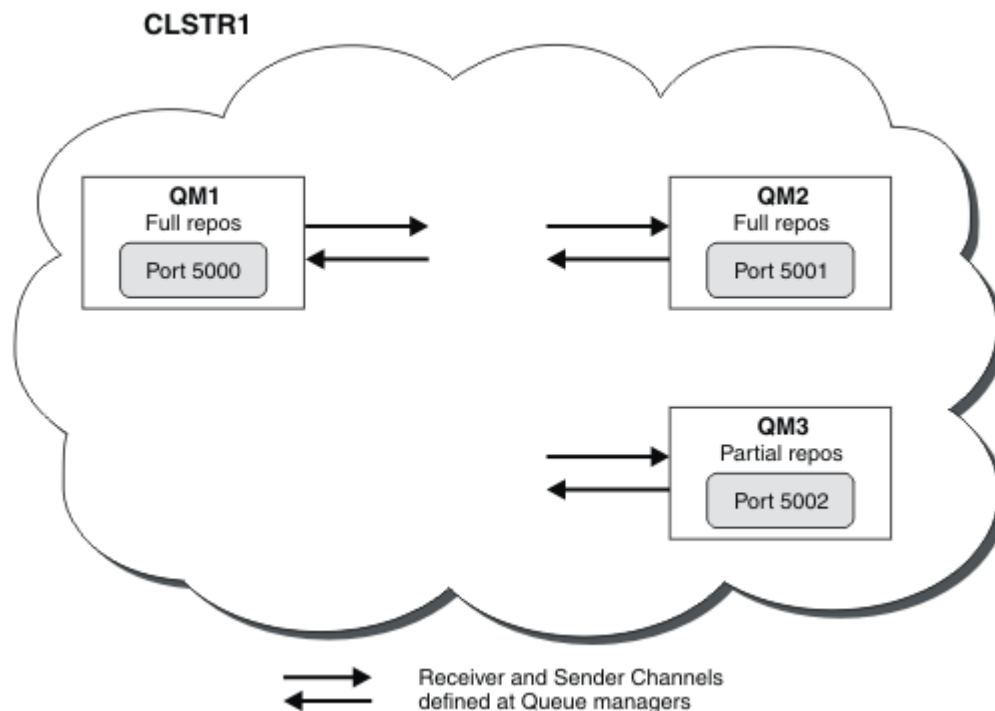
Abbildung 32. Einfache Publish/Subscribe-Hierarchie

Publish/Subscribe-Cluster einrichten: Szenario 1

Fügen Sie einem Cluster zwei Warteschlangenmanager als vollständige Repositories hinzu und definieren Sie die Kanäle zwischen ihnen.

Informationen zu diesem Vorgang

Das folgende Diagramm enthält drei Warteschlangenmanager: QM1, QM2 und



QM3:

QM1 und QM2 sind vollständige Repositories in einem Cluster und QM3 ist ein Teilrepository.

Szenario 1 fügt QM1 und QM2 als vollständige Repositories zum Cluster DEMO hinzu.

Szenario 2 fügt QM3 dem Cluster DEMO als Teilrepository hinzu.

Diese Tasks erfordern mindestens ein Befehlsfenster.

Vorgehensweise

1. Legen Sie QM1 und QM2 als vollständige Repositorys des DEMO -Clusters fest:

```
alter QMGR REPOS(DEMO)
```

2. Definieren und starten Sie ein Empfangsprogramm für QM1:

```
define listener(QM1_LS) TRPTYPE(TCP) CONTROL(QMGR) PORT(5000)
start listener(QM1_LS)
```

3. Definieren und starten Sie ein Empfangsprogramm für QM2:

```
define listener(QM2_LS) TRPTYPE(TCP) CONTROL(QMGR) PORT(5001)
start listener(QM2_LS)
```

4. Definieren Sie einen Empfängerkanal für QM1:

```
DEFINE CHANNEL(DEMO.QM1) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CONNAME('$HOSTNAME(5000)') CLUSQ
TER(DEMO) DESCR('TCP Cluster-receiver channel for queue manager QM1')
```

5. Definieren Sie einen Senderkanal von QM1 zu QM2:

```
DEFINE CHANNEL(DEMO.QM2) CHLTYPE(CLUSSDR) TRPTYPE(TCP) CONNAME('$HOSTNAME(5001)') CLUSQ
TER(DEMO) DESCR('TCP Cluster-sender channel from QM1 to queue manager QM2')
```

6. Definieren Sie einen Empfängerkanal für QM2:

```
DEFINE CHANNEL(DEMO.QM2) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CONNAME('$HOSTNAME(5001)') CLUSQ
TER(DEMO) DESCR('TCP Cluster-receiver channel for queue manager QM2')
```

7. Definieren Sie einen Senderkanal von QM2 zu QM1:

```
DEFINE CHANNEL(DEMO.QM1) CHLTYPE(CLUSSDR) TRPTYPE(TCP) CONNAME('$HOSTNAME(5000)') CLUSQ
TER(DEMO) DESCR('TCP Cluster-sender channel from qm2 to qm1')
```

8. Definieren Sie das Clusterthema scores unter QM1:

```
define topic(scores) TOPICSTR(/football) CLUSTER(DEMO)
```

9. Überprüfen Sie die Konfiguration mit den folgenden Befehlen:

```
display topic(scores) type(all) clusinfo
display clusqmgr(*)
display chstatus(*)
```

10. Testen Sie die Konfiguration mit zwei Befehlsfenstern:

- a. Geben Sie den folgenden Befehl im ersten Befehlsfenster ein:

```
/opt/mqm/samp/bin/amqspub /FOOTBALL/scores QM1
```

- b. Geben Sie den folgenden Befehl im zweiten Befehlsfenster ein:

```
/opt/mqm/samp/bin/amqssub /FOOTBALL/scores QM2
```

Zugehörige Tasks

[WebSphere MQ-Cluster verwalten](#)

[Neuen Cluster einrichten](#)

Publish/Subscribe-Cluster einrichten: Szenario 2

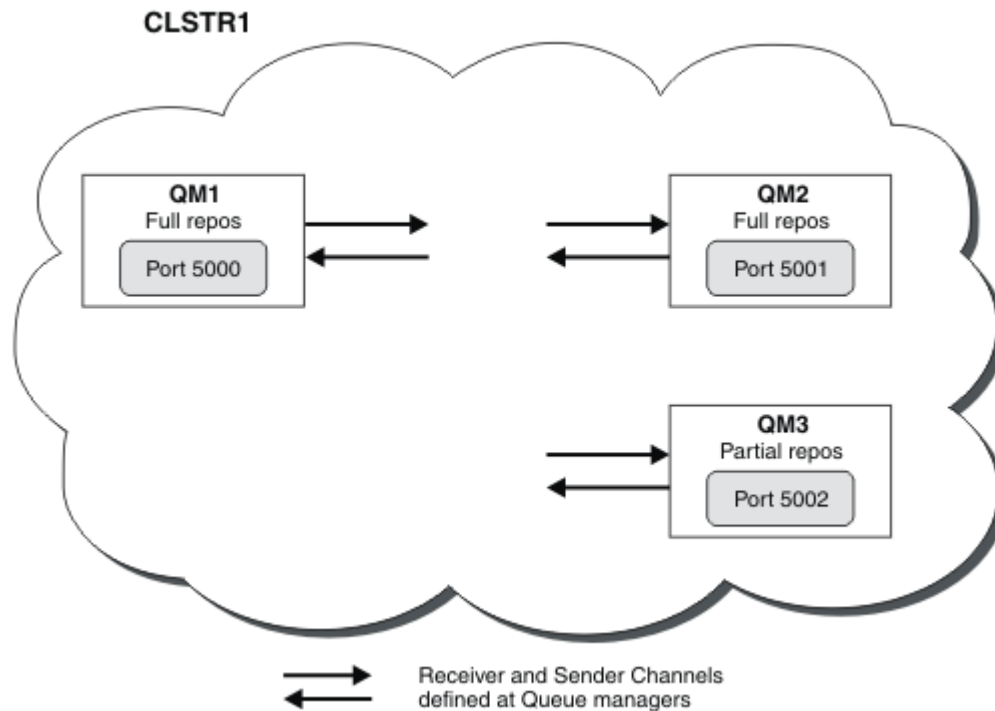
Fügen Sie dem Cluster einen dritten Warteschlangenmanager als Teilrepository hinzu.

Vorbereitende Schritte

Sie müssen die Task in „[Publish/Subscribe-Cluster einrichten: Szenario 1](#)“ auf Seite 68 abgeschlossen haben, bevor Sie diese Task ausführen.

Informationen zu diesem Vorgang

Das folgende Diagramm enthält drei Warteschlangenmanager: QM1, QM2 und



QM3:

QM1 und QM2 sind vollständige Repositorys in einem Cluster und QM3 ist ein Teilrepository.

Szenario 1 fügt QM1 und QM2 als vollständige Repositorys zum Cluster DEMO hinzu.

Szenario 2 fügt QM3 dem Cluster DEMO als Teilrepository hinzu.

Diese Tasks erfordern mindestens ein Befehlsfenster.

Vorgehensweise

1. Definieren und starten Sie ein Empfangsprogramm für QM3:

```
define listener(QM3_LS) TRPTYPE(TCP) CONTROL(QMGR) PORT(5002)
start listener(QM3_LS)
```

2. Definieren Sie einen Empfängerkanal für QM3:

```
DEFINE CHANNEL(DEMO.QM3) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CONNAME('$HOSTNAME(5002)') CLUSTER
(DEMO) DESCR('TCP Cluster-receiver channel for queue manager QM3')
```

3. Definieren Sie einen Senderkanal von QM3 zu QM1:

```
DEFINE CHANNEL(DEMO.QM1) CHLTYPE(CLUSSDR) TRPTYPE(TCP) CONNAME('$HOSTNAME(5000)') CLUSTER(DEMO)
DESCR('TCP Cluster-sender channel from qm3 to qm1')
```

4. Überprüfen Sie die Konfiguration mit den folgenden Befehlen:

```
display topic(scores) type(all) clusinfo
display clusqmgr(*)
display chstatus(*)
```

5. Testen Sie die Konfiguration mit zwei Befehlsfenstern:

- a. Geben Sie den folgenden Befehl im ersten Befehlsfenster ein:

```
/opt/mqm/samp/bin/amqspub /FOOTBALL/scores QM2
```

- b. Geben Sie den folgenden Befehl im zweiten Befehlsfenster ein:

Publish/Subscribe-Cluster

Ein Publish/Subscribe-Cluster ist ein IBM WebSphere MQ -Standardcluster miteinander verbundener Warteschlangenmanager, bei dem Veröffentlichungen automatisch von Veröffentlichungsanwendungen zu Subskriptionen verschoben werden, die auf einem beliebigen Warteschlangenmanager im Cluster vorhanden sind.

Ein Cluster für Publish/Subscribe-Messaging unterscheidet sich nicht von einem IBM WebSphere MQ-Standardcluster. Das bedeutet, dass sich die Warteschlangenmanager innerhalb eines Publish/Subscribe-Clusters auf physisch getrennten Computern befinden können und bei Bedarf automatisch über Clusterkanäle miteinander verbunden werden. Informationen zur Planung und Konfiguration eines IBM WebSphere MQ -Clusters finden Sie unter Funktionsweise von Clustern.

Ein Publish/Subscribe-Cluster wird erstellt, wenn ein Clusterthemenobjekt definiert wird, indem das Attribut **CLUSTER** für das Thema festgelegt wird, das von einem Warteschlangenmanager im Cluster konfiguriert wird. Die Themendefinition wird an alle Member des Clusters weitergegeben. Sie können das Thema und eine beliebige Themenzeichenfolge darunter in der Themenstruktur auf jedem Warteschlangenmanager im Cluster veröffentlichen und subscribieren. Veröffentlichungen werden automatisch an Subskribenten weitergegeben, die mit anderen Warteschlangenmanagern im Cluster verbunden sind.

Eine Publish/Subscribe-Aktivität ohne Cluster kann auch in einem Publish/Subscribe-Cluster stattfinden, indem mit Themenzeichenfolgen gearbeitet wird, die nicht unter ein Clusterthemenobjekt fallen. Diese Anordnung unterscheidet sich von einer Publish/Subscribe-Hierarchie, bei der alle Subskriptionen über die gesamte Hierarchie weitergegeben werden. In beiden Fällen ist eine feinere Steuerung über den Subskriptions- und Veröffentlichungsbereich verfügbar.

Die Verwendung von Clustern in einer Publish/Subscribe-Topologie bietet folgende Vorteile:

- Nachrichten, die für eine Subskription auf einem bestimmten Warteschlangenmanager im selben Cluster bestimmt sind, werden direkt zu diesem Warteschlangenmanager transportiert und müssen keinen temporären Warteschlangenmanager durchlaufen. Dies verbessert die Leistung und optimiert den Publish/Subscribe-Datenverkehr zwischen Warteschlangenmanagern im Vergleich zu einer hierarchischen Topologie.
- Da alle Warteschlangenmanager direkt miteinander verbunden sind, gibt es in dieser Topologie keinen Single Point of Failure. Wenn ein Warteschlangenmanager nicht verfügbar ist, können Subskriptionen auf anderen WS-Managern im Cluster weiterhin Nachrichten von Publishern auf verfügbaren Warteschlangenmanagern empfangen.
- In einem System, das mehrere separate Cluster enthält, z. B. wenn die Cluster geografisch verteilt sind, ist es möglich, Cluster in einer Clusterhierarchie zu verbinden. Diese Verbindung wird erstellt, indem ein einzelner WS-Manager in jedem Cluster verknüpft wird, um den Fluss von Veröffentlichungen und Subskriptionen über das Netz zu ermöglichen (siehe „Topicbereiche mehrerer Cluster kombinieren“ auf Seite 105). Sie können auch steuern, welche Veröffentlichungen von einem Cluster zu einem anderen übertragen werden (siehe „Themenbereiche in mehreren Clustern kombinieren und isolieren“ auf Seite 107).
- Eine subscribierende Anwendung kann eine Verbindung zu ihrem nächsten Warteschlangenmanager herstellen, um ihre eigene Leistung zu verbessern. Der Warteschlangenmanager empfängt alle Nachrichten, die der Subskriptionsregistrierung des Clients entsprechen, von allen Warteschlangenmanagern im Cluster.

Die Leistung einer Clientanwendung wird auch für andere Services verbessert, die von diesem Warteschlangenmanager angefordert werden. Eine Clientanwendung kann sowohl Publish/Subscribe als auch Punkt-zu-Punkt-Messaging verwenden.

- Die Anzahl der Clients und Subskriptionen für jeden Warteschlangenmanager kann reduziert werden, indem dem Cluster weitere Warteschlangenmanager hinzugefügt werden, um die Arbeitslast zu teilen. Veröffentlichungen werden automatisch an die Clients auf den neuen Warteschlangenmanagern verteilt. Bei einigen Verwendungsmustern kann dieser Prozess eine Publish/Subscribe-Cluster-Topologie hoch skalierbar machen.

Hinweise zur Verwendung von Clustern in Publish/Subscribe:

- Alle Warteschlangenmanager in einem Publish/Subscribe-Cluster werden automatisch auf alle anderen Warteschlangenmanager im Cluster aufmerksam gemacht. Dieser Prozess unterscheidet sich für einen Punkt-zu-Punkt-Cluster, in dem nur Warteschlangenmanager bekannt sind, die für einen Warteschlangenmanager von Interesse sind.
- Warteschlangenmanager in einem Publish/Subscribe-Cluster, die mindestens eine Subskription für ein Clusterthema enthalten, erstellen automatisch Clustersenderkanäle zu allen anderen Warteschlangenmanagern im Cluster. Die Warteschlangenmanager senden auch Informationen zu den Subskriptionen für jeden von ihnen, auch wenn die empfangenden Warteschlangenmanager keine Nachrichten in Clustertiteln veröffentlichen.
- Die erste Subskription auf einem WS-Manager in einer Themenzeichenfolge unter einem Clusterthema führt dazu, dass eine Nachricht an alle anderen Warteschlangenmanager im Cluster gesendet wird. In ähnlicher Weise wird auch die letzte Subskription für eine zu löschende Themenzeichenfolge in einer Nachricht angezeigt. Je mehr einzelne Themenzeichenfolgen unter einem Clusterthema verwendet werden, desto mehr Kommunikation zwischen Warteschlangenmanagern findet statt.



Vorsicht:

Aus den zuvor in diesem Abschnitt aufgeführten Gründen kann die Einführung eines Clusterthemas in einem großen IBM WebSphere MQ -Cluster (d. h. einem Cluster, der viele Warteschlangenmanager enthält) sofort zu einer zusätzlichen Belastung jedes Warteschlangenmanagers im Cluster führen und in einigen Situationen zu einer Leistungsminderung führen. Weitere Informationen finden Sie im Abschnitt „Clusterthemenleistung“ auf Seite 79.

Die Einführung von Publish/Subscribe in einem Cluster von Warteschlangenmanagern, insbesondere einem vorhandenen Cluster, muss sorgfältig geplant werden, um diese Leistungseinbußen zu berücksichtigen.

Wenn bekannt ist, dass ein Cluster die Leistungseinbußen bei Publish/Subscribe nicht bewältigen kann, kann die Publish/Subscribe-Clusterfunktionalität in Warteschlangenmanagern mit dem Parameter **PSCLUS** inaktiviert werden. Der Parameter **PSCLUS** dient primär dazu, die schwerwiegenden Probleme zu stoppen, die bei der Erstellung eines Publish/Subscribe-Clusters auftreten können, indem versehentlich oder falsch ein Clusterthema definiert wird. Weitere Informationen zum Inaktivieren dieser Funktionalität finden Sie unter „Cluster-Publish/Subscribe in einem Cluster blockieren“ auf Seite 78.

Publish/Subscribe-Clustering: Bewährte Verfahren

Dieser Abschnitt enthält Anleitungen zur Planung und Verwaltung von IBM WebSphere MQ Publish/Subscribe-Clustern. Die Informationen basieren auf Tests und Feedback von Kunden.

Die folgenden Informationen setzen voraus, dass der Benutzer über grundlegende Kenntnisse zu IBM WebSphere MQ -Clustern, Publish/Subscribe, verfügt und mit den Themen in „Verteiltes Publish/Subscribe“ auf Seite 53 vertraut ist. Diese Informationen sind nicht als "one size fits all" -Lösung gedacht, sondern versuchen, gemeinsame Ansätze für allgemeine Probleme zu teilen.

Publish/Subscribe-Cluster

Bei einem Cluster besteht bei Bedarf eine direkte Verbindung zwischen Warteschlangenmanagern im Cluster. Wenn ein Cluster für Punkt-zu-Punkt-Messaging verwendet wird, kennt jeder Warteschlangenmanager im Cluster nur die Informationen zu anderen Clusterressourcen, z. B. zu anderen Warteschlangenmanagern im Cluster und Clusterwarteschlangen, wenn Anwendungen, die eine Verbindung zu ihnen herstellen, ihre Verwendung anfordern, d. h., sie arbeiten auf einer bedarfsgerechten Basis.

Ein Publish/Subscribe-Cluster ist ein Cluster von Warteschlangenmanagern mit den üblichen CLUSDR -und CLUSRCVR -Kanaldefinitionen. Ein Publish/Subscribe-Cluster enthält jedoch auch mindestens ein TOPIC -Objekt, das auf mindestens einem Warteschlangenmanager im Cluster definiert ist, auf dem das Themenobjekt einen Clusternamen angegeben hat.

Wenn ein Themenobjekt im Cluster definiert ist, kann eine Anwendung, die mit einem Warteschlangenmanager im Cluster verbunden ist, dieses Thema oder einen beliebigen Knoten in der Themenstruktur unter

diesem Thema abonnieren und Veröffentlichungen zu diesem Thema von anderen Warteschlangenmanagern im Cluster empfangen. Dieser Prozess wird durch die Erstellung von Proxy-Subskriptionen auf allen anderen Warteschlangenmanagern im Cluster erreicht, die den Warteschlangenmanager angeben, auf dem die Subskription vorhanden ist. Wenn also eine Veröffentlichung zu dem betreffenden Thema auf ihrem Warteschlangenmanager erfolgt, wissen sie, dass sie sie an andere geeignete Mitglieder des Clusters weiterleiten und von dort aus an die einzelnen Anwendungssubskriptionen übergeben.

Um diese Zustellung zu erreichen, muss jeder Warteschlangenmanager im Cluster die Identität jedes anderen Warteschlangenmanagers im Cluster kennen, sobald einem Cluster ein Thema hinzugefügt wird. Dieses Wissen wird über die Warteschlangenmanager mit vollständigem Repository des Clusters weitergegeben. Veröffentlichte Nachrichten auf einem Warteschlangenmanager werden nur an andere Warteschlangenmanager im Cluster gesendet, die als Host für Subskriptionen desselben Themas bekannt sind. Um diesen Prozess zu erreichen, wenn eine Anwendung eine Subskription für ein Thema erstellt, das in einem Cluster zusammengefasst ist, muss dieser Warteschlangenmanager über Clustersenderkanäle direkt mit jedem anderen Warteschlangenmanager im Cluster kommunizieren, um die Proxy-Subskriptionen weiterzugeben.

Dieser Prozess unterscheidet sich erheblich von den begrenzten Informationen und der Konnektivität, die erforderlich sind, wenn Sie einen Cluster für die Punkt-zu-Punkt-Bereitstellung verwenden. Daher unterscheiden sich die Anforderungen in einem Publish/Subscribe-Cluster von den Anforderungen in einem Punkt-zu-Punkt-Cluster (einer ohne Clusterthemen).

Die Verwendung von Clusterthemen macht die Erweiterung der Publish/Subscribe-Domäne zwischen Warteschlangenmanagern einfach, kann aber zu Problemen führen, wenn die Mechanismen und Auswirkungen nicht verstanden werden und in Bezug auf den Cluster, der für Publish/Subscribe verwendet wird, berücksichtigt werden. Die folgenden Best Practices sollen Ihnen bei diesem Verständnis und bei der Vorbereitung helfen.

Zusammenfassend lässt sich sagen, dass die Auswirkungen von Cluster-Publish/Subscribe auf die Leistung eines großen Clusters abträglich sein können und sorgfältig abgewogen und verstanden werden müssen, bevor versucht wird, Publish/Subscribe in einem vorhandenen Cluster zu verwenden. Zum Beispiel sogar die einfache Erstellung eines Clusterthemenobjekts. Möglicherweise ist es besser, mit einem kleinen neuen Cluster zu beginnen, der der Publish/Subscribe-Aktivität zugeordnet ist, und den Cluster von dort aus zu vergrößern.

Publish/Subscribe-Topologie entwerfen

Wie bereits beschrieben, gibt es Kapazitäts- und Leistungsaspekte bei der Verwendung von Publish/Subscribe in einem Cluster. Daher ist es ein bewährtes Verfahren, die Notwendigkeit von Publish/Subscribe über Warteschlangenmanager hinweg sorgfältig zu berücksichtigen und sie auf die Anzahl der Warteschlangenmanager zu begrenzen, die sie benötigen. Nachdem die minimale Gruppe von Warteschlangenmanagern identifiziert wurde, die Veröffentlichungen und Subskriptionen für eine Gruppe von Themen durchführen müssen, können sie zu Mitgliedern eines Clusters werden, der nur sie und keine anderen Warteschlangenmanager enthält.

Dies gilt insbesondere für einen eingerichteten Cluster, der bereits gut für Punkt-zu-Punkt-Messaging funktioniert. Aus diesem Grund empfiehlt es sich, bei der Umwandlung eines vorhandenen großen Clusters in einen Publish/Subscribe-Cluster zunächst einen separaten Cluster für die Publish/Subscribe-Arbeit zu erstellen, in dem die Anwendungen ausprobiert werden können, anstatt den aktuellen Cluster zu verwenden. Es ist möglich, vorhandene Warteschlangenmanager, die bereits in einem oder mehreren Punkt-zu-Punkt-Clustern vorhanden sind, weiterhin zu verwenden. Die Untergruppe dieser Warteschlangenmanager muss Mitglieder des neuen Publish/Subscribe-Clusters sein. Dieser neue Cluster muss jedoch über separate Warteschlangenmanager verfügen, die als vollständige Repositories konfiguriert sind, um die zusätzliche Last von den vorhandenen vollständigen Clusterrepositories zu isolieren.

Wenn Sie feststellen, dass ein Cluster aufgrund seiner Größe oder aktuellen Arbeitslast nicht für Publish/Subscribe verwendet werden soll, empfiehlt es sich, zu verhindern, dass dieser Cluster durch die einfache Erstellung eines Clusterthemas auf einem beliebigen Warteschlangenmanager im Cluster unerwartet in einen Publish/Subscribe-Cluster umgewandelt wird. Verwenden Sie die Warteschlangenmanagereigenschaft **PSCLUS**, um dieses Design zu erreichen. Weitere Informationen hierzu finden Sie im Abschnitt Cluster-Publish/Subscribe in einem Cluster unterbinden.

Es ist auch wichtig, sorgfältig auszuwählen, welche Themen zum Cluster hinzugefügt werden sollen: Je höher die Themenstruktur dieser Themen ist, desto breiter werden sie. Aus diesem Grund wird nicht empfohlen, den Topic-Stammknoten in den Cluster zu stellen, ohne das Verhalten zu berücksichtigen. Globale Themen möglichst naheliegend machen, z. B. durch mit einem übergeordneten Qualifikationsmerkmal in der Themenzeichenfolge: /global oder /cluster.

Größe von Systemen festlegen

Publish/Subscribe-Cluster erfordern viele Kanäle, da sich das Modell vom Punkt-zu-Punkt-Messaging unterscheidet: Jeder Warteschlangenmanager muss mit allen anderen Warteschlangenmanagern in diesem Cluster kommunizieren. Das Punkt-zu-Punkt-Modell ist ein Opt-in-Modell, aber Publish/Subscribe-Cluster haben eine unterschiedslose Natur mit einem Subskriptionsfan-out. Daher müssen die Warteschlangenmanager mit vollständigem Repository und alle Warteschlangenmanager, die lokale Subskriptionen in einem Publish/Subscribe-Cluster enthalten, gleichzeitig über die Kapazität verfügen, Kanäle zu jedem Mitglied des Clusters einzurichten.

Es ist am besten sicherzustellen, dass jeder Warteschlangenmanager im Publish/Subscribe-Cluster diese Kapazität erreichen kann, aber es wird anerkannt, dass Warteschlangenmanager, die nie als Host für Subskriptionen bekannt sind, keine Kanäle mit jedem anderen Warteschlangenmanager aufbauen müssen und daher diese Kapazität nicht benötigen.

Allerdings ist Vorsicht geboten, da eine versehentliche Subskription, die auf einem solchen Warteschlangenmanager erstellt wird, oder ein Versuch, einen solchen Warteschlangenmanager mit den anderen im Cluster manuell zu resynchronisieren, dazu führt, dass alle Kanäle gleichzeitig gestartet werden. Weitere Informationen finden Sie in „Resynchronisation von Proxy-Subskriptionen“ auf Seite 74.

Cluster-Publish/Subscribe ermöglicht die Zustellung veröffentlichter Nachrichten auf einem Warteschlangenmanager an Subskriptionen auf anderen Warteschlangenmanagern. Bei Punkt-zu-Punkt-Messaging können sich die Kosten für die Übertragung von Nachrichten zwischen Warteschlangenmanagern nachteilig auf die Leistung auswirken. Daher muss versucht werden, nach Möglichkeit Subskriptionen für Themen in denselben Warteschlangenmanagern zu erstellen, in denen Nachrichten veröffentlicht werden.

Ein weiterer Aspekt ist die Auswirkung der Weitergabe von Proxy-Subskriptionen auf die Leistung des Systems. Normalerweise sendet ein Warteschlangenmanager eine Proxy-Subskriptionsnachricht an jeden anderen Warteschlangenmanager im Cluster, wenn die erste Subskription für eine bestimmte Clusterthemenzeichenfolge (nicht nur ein konfiguriertes Themenobjekt) erstellt wird. Wenn eine Publish/Subscribe-Lösung aus vielen eindeutigen Themenzeichenfolgen besteht, die subskribiert werden, oder die Themen häufig subskribiert und nicht subskribiert werden, kann eine erhebliche Menge an Proxy-Subskriptionsdatenverkehr zwischen allen Warteschlangenmanagern in einem Cluster generiert werden, was die Gesamtleistung des Systems beeinträchtigt. Informationen zu Möglichkeiten zur Reduzierung des Systemaufwands für Proxy-Subskriptionen finden Sie in „Clusterthemenleistung“ auf Seite 79.

Resynchronisation von Proxy-Subskriptionen

Unter normalen Umständen stellen Warteschlangenmanager automatisch sicher, dass die Proxy-Subskriptionen im System die Subskriptionen auf jedem Warteschlangenmanager im Cluster korrekt wiedergeben.

Bei Bedarf können Sie jedoch die lokalen Subskriptionen eines Warteschlangenmanagers mit den Proxy-Subskriptionen, die er im Cluster weitergegeben hat, mit dem Befehl `REFRESH QMGR TYPE (PROXYSUB)` manuell resynchronisieren.

Anmerkung: Die Resynchronisation erzeugt vorübergehend eine plötzliche zusätzliche Proxy-Subskriptionslast im Cluster, die von dem Warteschlangenmanager ausgeht, auf dem der Befehl ausgegeben wird. Verwenden Sie es daher nur, wenn Sie in der IBM WebSphere MQ -Service-, IBM WebSphere MQ -Dokumentation oder Fehlerprotokollierung dazu aufgefordert werden.

Ein Beispiel für den Fall, dass die Resynchronisation erforderlich ist, ist, wenn ein Warteschlangenmanager seine Proxy-Subskriptionen nicht ordnungsgemäß weitergeben kann, weil möglicherweise ein Kanal gestoppt wurde und nicht alle Nachrichten für die Übertragung in die Warteschlange eingereicht werden können oder weil ein Bedienerfehler dazu geführt hat, dass Nachrichten fälschlicherweise aus der Warte-

schlange SYSTEM . CLUSTER . TRANSMIT . QUEUE gelöscht wurden. In this situation, first rectify the original problem (for example by restarting the channel) then issue the **REFRESH QMGR TYPE (PROXYSUB)** command on the queue manager. Beachten Sie, dass Veröffentlichungen, die aufgrund fehlender Proxy-Subskriptionen verpasst wurden, für die betroffenen Subskriptionen nicht wiederhergestellt werden. Dieser Nachteil muss berücksichtigt werden.

Für die Resynchronisation muss der Warteschlangenmanager Kanäle zu allen anderen Warteschlangenmanagern im Cluster starten. Der Warteschlangenmanager, den Sie aktualisieren, muss in der Lage sein, mit allen anderen Warteschlangenmanagern im Cluster zu kommunizieren.

Clusterthemen

Clusterthemen sind Verwaltungsthemen, für die das Attribut **cluster** definiert ist. Informationen zu Clusterthemen werden an alle Mitglieder eines Clusters übertragen und mit lokalen Themen kombiniert, um auf jedem Warteschlangenmanager einen anderen Themenbereich zu erstellen.

Wenn Sie ein Clusterthema für einen Warteschlangenmanager definieren, wird diese Clusterthemendefinition an die Warteschlangenmanager mit vollständigem Repository gesendet. Anschließend leiten die vollständigen Repositories die Clusterthemendefinition an alle Warteschlangenmanager im Cluster weiter, sodass das Clusterthema für alle Bereitsteller und Subskribenten verfügbar ist, die in einem Warteschlangenmanager im Cluster vorhanden sind. Der Warteschlangenmanager, in dem ein Clusterthema erstellt wird, wird als Clusterthemenhost bezeichnet. Das Clusterthema kann von jedem Warteschlangenmanager im Cluster verwendet werden, aber alle Änderungen an einem Clusterthema müssen auf dem Warteschlangenmanager vorgenommen werden, auf dem dieses Thema definiert ist (der Host), an dem die Änderung über die vollständigen Repositories an alle Mitglieder des Clusters weitergegeben wird.

Auf jedem Warteschlangenmanager wird ein einzelner Topic-Namespaces aus den lokalen und Cluster-Topic-Definitionen erstellt, die er kennt. Wenn eine Anwendung ein Thema subskribiert, das in ein Clusterthema aufgelöst wird, erstellt IBM WebSphere MQ eine Proxy-Subskription und sendet sie direkt von dem Warteschlangenmanager, in dem die Subskription erstellt wird, an alle anderen Mitglieder des Clusters. Im Gegensatz zum Clusterthema selbst werden Proxy-Subskriptionen nicht über die Warteschlangenmanager mit vollständigem Repository übertragen.

Nachrichten, die zu einem Thema veröffentlicht werden, werden an jede Subskription gesendet, die dem Warteschlangenmanager bekannt ist, mit dem der Publisher verbunden ist. Wenn eine dieser Subskriptionen Proxy-Subskriptionen ist, wird eine Kopie der veröffentlichten Nachricht an den Warteschlangenmanager gesendet, von dem die Proxy-Subskription stammt. Der empfangende Warteschlangenmanager sendet dann eine Kopie der Nachricht an jede lokale Subskription. Dieser Prozess stellt sicher, dass der Subskribent eines Clusterthemas Veröffentlichungen von Publishern empfängt, die mit einem der Warteschlangenmanager im Cluster verbunden sind, und dass die minimale Anzahl veröffentlichter Nachrichten über den Cluster weitergegeben wird.

Wenn Sie über ein Clusterthema und ein lokales Themenobjekt verfügen, hat das lokale Thema Vorrang. Weitere Informationen finden Sie im Abschnitt „Mehrere Clusterthemendefinitionen“ auf Seite 77.

Weitere Informationen zu den Befehlen zum Anzeigen von Clusterthemen finden Sie unter den folgenden zugehörigen Links:

Platzhaltersubskriptionen

Proxy-Subskriptionen werden erstellt, wenn lokale Subskriptionen an eine Themenzeichenfolge vorgenommen werden, die in einem Clusterthemenobjekt aufgelöst wird oder darunter ist. Bei einer Subskription mit Platzhalterzeichen höher in der Themenhierarchie als ein beliebiges Clusterthema erstellt hat, hat es keine Proxy-Subskriptionen, die um den Cluster für das übereinstimmende Clusterthema gesendet werden, und daher keine Veröffentlichungen von anderen Mitgliedern des Clusters empfängt. Sie erhält jedoch Veröffentlichungen aus dem lokalen WS-Manager.

Wenn eine andere Anwendung jedoch eine Themenzeichenfolge subskribiert, die in oder aufgelöst wird, unterhalb des Clusterthemas werden Proxy-Subskriptionen generiert und Veröffentlichungen an diesen WS-Manager weitergegeben werden. Bei Ankunft der ursprünglichen, höheren Wildcard-Subskription wird als rechtmäßiger Empfänger dieser Veröffentlichungen betrachtet und erhält eine Kopie.

Dieses Verhalten unterscheidet sich von lokal veröffentlichten Nachrichten zu denselben Themen. Wenn dieses Verhalten nicht erforderlich ist, führt die Einstellung **WILDCARD (BLOCK)** für das Clusterthema dazu, dass das ursprüngliche Platzhalterzeichen nicht als legitime Subskription betrachtet wird und keine Veröffentlichungen (lokal oder von einer anderen Stelle im Cluster) für das Clusterthema oder dessen Unterthemen empfängt.

Zugehörige Konzepte

[Mit Verwaltungsthemen arbeiten](#)

[Mit Subskriptionen arbeiten](#)

Zugehörige Verweise

[ANZEIGEN TOPIC](#)

[ANZEIGEN TPSTATUS](#)

[ANZEIGEN SUB](#)

Clusterthemenattribute

Zum Entwerfen und Verwalten von Publish/Subscribe-Clustern ist ein gutes Verständnis der Clusterthemenattribute erforderlich.

Ein Themenobjekt hat eine Reihe von Attributen, die für Publish/Subscribe-Topologien mit mehreren Warteschlangenmanagern gelten. Wenn Sie einen IBM WebSphere MQ -Cluster zum Erstellen einer solchen Topologie verwenden, weisen diese Attribute das folgende Verhalten auf.

PROXYSUB

- **PROXYSUB** ist ein Attribut, das steuert, wann Proxy-Subskriptionen vorgenommen werden. Details dazu, warum Sie den Standardwert von FIRSTUSE für dieses Attribut ändern sollten, finden Sie unter „[Weitere Informationen zu Routing-Mechanismen](#)“ auf Seite 57.
- Wie bei anderen Attributen eines Clusterthemas wird das Attribut **PROXYSUB** an jeden Warteschlangenmanager im Cluster weitergegeben, nicht nur an den Warteschlangenmanager, auf dem das Thema definiert wurde. Dies führt sofort dazu, dass jeder Warteschlangenmanager im Cluster eine Proxy-Subskription mit Platzhalterzeichen für jeden anderen Warteschlangenmanager erstellt. Das Ergebnis dieses Prozesses ist, dass jeder Warteschlangenmanager Clustersenderkanäle zu jedem anderen Warteschlangenmanager erstellt und alle veröffentlichten Nachrichten an jeden Warteschlangenmanager gesendet werden.

PUBSCOPE und SUBSCOPE

PUBSCOPE und **SUBSCOPE** legen fest, ob dieser Warteschlangenmanager Veröffentlichungen an Warteschlangenmanager in der Topologie (Publish/Subscribe-Cluster oder Hierarchie) weitergibt oder den Geltungsbereich auf den lokalen Warteschlangenmanager beschränkt. Sie können den entsprechenden Job programmgesteuert mit MQPMO_SCOPE_QMGR/MQSO_SCOPE_QMGR ausführen.

- **PUBSCOPE** Wenn ein Clusterthemenobjekt mit **PUBSCOPE (QMGR)** definiert ist, wird die Definition mit dem Cluster gemeinsam genutzt, aber der Bereich der Veröffentlichungen, die auf diesem Thema basieren, ist lokal und wird nicht an andere Warteschlangenmanager im Cluster gesendet.
- **SUBSCOPE** Wenn ein Clusterthemenobjekt mit **SUBSCOPE (QMGR)** definiert ist, wird die Definition mit dem Cluster gemeinsam genutzt, aber der Bereich der Subskriptionen, die auf diesem Thema basieren, ist nur lokal. Daher werden keine Proxy-Subskriptionen an andere Warteschlangenmanager im Cluster gesendet.

Diese beiden Attribute werden im Allgemeinen zusammen verwendet, um einen Warteschlangenmanager von der Interaktion mit anderen Mitgliedern des Clusters zu bestimmten Themen zu trennen. Der Warteschlangenmanager veröffentlicht oder empfängt keine Veröffentlichungen zu diesen Themen in und von anderen Mitgliedern des Clusters. Diese Situation verhindert nicht die Veröffentlichung oder Subskription, wenn Themenobjekte in Unterabschnitten definiert sind.

Wenn Sie **SUBSCOPE** in einer lokalen Definition eines Themas auf QMGR setzen, werden andere WS-Manager im Cluster nicht daran gehindert, ihre Proxy-Subskriptionen an den Warteschlangenmanager weiterzugeben, wenn sie eine Clusterversion des Themas mit **SUBSCOPE (ALL)** verwenden. Wenn die lokale Definition jedoch auch **PUBSCOPE** auf QMGR setzt, werden diese Proxy-Subskriptionen keine Veröffentlichungen von diesem Warteschlangenmanager gesendet.

Mehrere Clusterthemendefinitionen

Eine lokale Themendefinition überschreibt eine über Fernzugriff definierte Clusterthemendefinition mit demselben Namen. Die Erstellung mehrerer Definitionen desselben Clusterthemas auf verschiedenen Warteschlangenmanagern in einem Cluster ist ebenfalls möglich. Beide Szenarios erfordern jedoch gewisse Vorsicht, die Gründe werden in diesem Abschnitt erläutert.

Wie bei Clusterwarteschlangen führt das Vorhandensein mehrerer Definitionen desselben Clusterthemenobjekts in einem Cluster dazu, dass unterschiedliche Eigenschaften für jede definiert werden können. Es ist nicht einfach zu bestimmen, welche Version der Themendefinition für jeden Warteschlangenmanager im Cluster sichtbar ist, und es ist daher schwierig, das erwartete Verhalten festzulegen.

Wenn zwei oder mehr Clusterthemendefinitionen für eine einzelne Themenzeichenfolge unterschiedliche Attribute aufweisen oder in mehreren Clustern vorhanden sind, werden Nachrichten (AMQ5465 & AMQ5466) in das Fehlerprotokoll geschrieben und die zuletzt empfangene Clusterthemendefinition wird verwendet.

Der Clusterthemenhost-Warteschlangenmanager darf die Themendefinition nicht löschen und verbleibt im Cluster, um sicherzustellen, dass das Clusterthema weiterhin allen Mitgliedern des Clusters bekannt ist. Es ist nicht unbedingt erforderlich, dass dieser Host-Warteschlangenmanager ständig verfügbar ist, weil die Clusterthemendefinition von den Warteschlangenmanagern mit vollständigem Repository und von allen anderen Warteschlangenmanagern in ihren Teilclusterrepositorys zwischengespeichert wird. Dadurch ist eine Verfügbarkeit von mindestens 60 Tagen sichergestellt, wenn der Host-Warteschlangenmanager nicht verfügbar ist. Weitere Informationen zu diesem Thema finden Sie unter [„Wichtige Rollen für Publish/Subscribe-Cluster-Warteschlangenmanager“](#) auf Seite 83.

Clusterthemendefinition lokal überschreiben

Es kann erforderlich sein, das Verhalten eines Clusterthemas auf bestimmten Warteschlangenmanagern im Cluster zu überschreiben. Diese Überschreibung kann erreicht werden, indem ein lokales Themenobjekt definiert wird, um ein Clusterthemenobjekt mit derselben Themenzeichenfolge zu überschreiben, und es verwendet wird, um nur für lokal verbundene Subskribenten zu veröffentlichen.

Auch wenn eine lokale Definition eines Themas erstellt wird, um ein Clusterthema in einem Warteschlangenmanager zu überschreiben, empfängt der Warteschlangenmanager weiterhin Proxy-Subskriptionen von anderen Mitgliedern des Clusters, die die Clusterthemendefinition verwenden. Standardmäßig werden lokal veröffentlichte Nachrichten weiterhin an die fernen Warteschlangenmanager gesendet, um die Proxy-Subskriptionen zu berücksichtigen. Wenn diese Anordnung nicht erforderlich ist, geben Sie **PUBSCOPE** (QMGR) im lokalen Themenobjekt an, um sicherzustellen, dass Publisher-Anwendungen, die mit diesem Warteschlangenmanager verbunden sind, nur für lokale Subskribenten veröffentlichen.

Clusterthemendefinition ändern

Wenn Sie eine Clusterthemendefinition ändern müssen, ändern Sie sie auf demselben Warteschlangenmanager, auf dem sie definiert wurde, auf dem Clusterthemenhost. Erstellen Sie keine Definition desselben Clusterthemas auf einem anderen Warteschlangenmanager im Cluster. Die erneute Definition des Themas führt zu zwei Cluster-Topic-Hosts für dasselbe Cluster-Thema.

Die mehrfache Definition eines Clusterthemas führt zu potenziell widersprüchlichen Definitionen und der Möglichkeit, dass unterschiedliche WS-Manager unterschiedliche Definitionen zu unterschiedlichen Zeiten verwenden.

Clusterthemendefinition in einen anderen WS-Manager im Cluster verschieben

You might need to move a cluster topic definition from one queue manager in the cluster to another, for example when decommissioning a queue manager from the cluster. Wenn Sie eine Clusterthemendefinition auf einen anderen Warteschlangenmanager im Cluster verschieben möchten, ohne den Veröffentlichungsfluss zu unterbrechen, müssen Sie die folgenden Schritte ausführen. Das Beispiel verschiebt eine Definition von QM1 nach QM2.

1. Erstellen Sie ein Duplikat der Clusterthemendefinition in QM2 mit denselben Attributen wie die Definition von QM1.
2. Warten Sie, bis die neue Definition durch die Warteschlangenmanager mit vollständigem Repository im Cluster weitergegeben wird. Die Weitergabe kann bestimmt werden, indem die Clusterthemen auf jedem Cluster-Member mit dem Befehl (**DISPLAY CLUSTER**) angezeigt und auf eine Definition aus QM2 geprüft werden.
3. Löschen Sie die Clusterthemendefinition aus QM1.

Nachdem die ursprüngliche Definition aus QM1 gelöscht wurde, kann die Definition unter QM2 bei Bedarf geändert werden, ohne dass es zu einem Konflikt in den Eigenschaften kommt.

Clusterthemendefinition auf einem fehlgeschlagenen Warteschlangenmanager ersetzen

Im vorherigen Szenario ist es möglicherweise nicht möglich, die Definition aus QM1 zu löschen, wenn QM1 für einen bestimmten Zeitraum nicht verfügbar ist. In diesem Szenario ist es zulässig, beide Definitionen zu verwenden.

Wenn es dann erforderlich wird, die Clusterthemendefinition zu ändern, ist es möglich, die Version in QM2 zu ändern, da die QM2 -Definition neuer als die QM1 -Definition ist und daher Vorrang hat. In diesem Zeitraum werden jedoch Fehler in die Fehlerprotokolle der Warteschlangenmanager geschrieben, da es eine in Konflikt stehende Clusterthemendefinition gibt. Beheben Sie den Fehler so schnell wie möglich, indem Sie die duplizierte Clusterthemendefinition aus QM1 entfernen, wenn sie erneut gestartet werden kann.

Alternativ kann der Befehl **RESET CLUSTER** verwendet werden, um die Ausgabe des Warteschlangenmanagers zu erzwingen, wenn QM1 nie in den Cluster zurückkehren wird (z. B. unerwartete Stilllegung nach einem katastrophalen Hardwarefehler). **RESET CLUSTER** löscht automatisch alle Themenobjekte, die sich auf dem Zielwarteschlangenmanager befinden.

Cluster-Publish/Subscribe in einem Cluster blockieren

Die Einführung von Publish/Subscribe in einem Cluster von Warteschlangenmanagern, insbesondere in einem vorhandenen Cluster, muss sorgfältig geplant werden, um Leistungseinbußen zu berücksichtigen.

Die Einführung eines Clusterthemas in einem großen IBM WebSphere MQ -Cluster (ein Cluster, der viele Warteschlangenmanager enthält) kann sofort zu einer zusätzlichen Belastung jedes Warteschlangenmanagers im Cluster und in einigen Situationen zu einer Leistungsminderung führen. Daher muss die Einführung von Publish/Subscribe sorgfältig geplant werden. Weitere Informationen finden Sie im Abschnitt „Clusterthemenleistung“ auf Seite 79.

Wenn bekannt ist, dass ein Cluster den Aufwand für Publish/Subscribe nicht aufnehmen konnte, ist es möglich, die Publish/Subscribe-Clusterfunktionalität in Warteschlangenmanagern zu deaktivieren, indem das Warteschlangenmanagerattribut **PSCLUS** auf **DISABLED** gesetzt wird.

Wenn Sie **PSCLUS** auf **DISABLED** setzen, werden drei Aspekte der Warteschlangenmanagerfunktionalität geändert:

- Ein Administrator dieses Warteschlangenmanagers kann ein Topic-Objekt nicht mehr als Cluster definieren.
- Eingehende Themendefinitionen oder Proxy-Subskriptionen von anderen Warteschlangenmanagern werden zurückgewiesen (es wird eine Warnung protokolliert, um den Administrator über eine falsche Konfiguration zu informieren).
- Vollständige Repositories nutzen Informationen zu jedem Warteschlangenmanager nicht mehr automatisch mit allen anderen Teilrepositories gemeinsam, wenn sie eine Themendefinition empfangen.

Obwohl **PSCLUS** ein Parameter jedes einzelnen Warteschlangenmanagers in einem Cluster ist, ist es nicht beabsichtigt, die Publish/Subscribe-Subskription in einer Untergruppe von Warteschlangenmanagern im Cluster selektiv zu inaktivieren. Abgesehen von allem anderen würde diese Methode dazu führen, dass häufige Fehlermeldungen angezeigt werden, da Proxy-Subskriptionen und Themendefinitionen ständig angezeigt und abgelehnt werden. Im Idealfall sollten bei Verwendung dieser Option alle Warteschlangen-

manager im Cluster konsistent inaktiviert werden. Wenn ein Warteschlangenmanager an einem oder mehreren Publish/Subscribe-Clustern und einem oder mehreren traditionellen Clustern teilnimmt, muss **PSCLUS** auf diesem Warteschlangenmanager auf ENABLED gesetzt werden. Lesen Sie die folgenden Informationen zur Inaktivierung in den vollständigen Repositorys.

Wichtig: Die Einstellung von **PSCLUS** auf DISABLED für alle Warteschlangenmanager mit vollständigem Repository im Cluster verhindert, dass sich eine Clusterthemendefinition in einem falsch konfigurierten Teilrepository auf andere Warteschlangenmanager im Cluster auswirkt. In solchen Szenarios wird die Inkonsistenz in den Fehlerprotokollen der vollständigen Warteschlangenmanager-Repositorys gemeldet.

Wenn sich ein traditioneller Punkt-zu-Punkt-Cluster mit einem Publish/Subscribe-Cluster überschneidet, ist es wichtig, in jedem eine separate Gruppe vollständiger Repositorys zu verwenden. Bei dieser Anordnung können Themendefinitionen und Informationen zu 'allen Warteschlangenmanagern' nur im Publish/Subscribe-Cluster fließen.

Es gibt einige Vorbehalte bei der Verwendung dieses Parameters, die helfen, inkonsistente Konfigurationen zu vermeiden. Es können keine Clusterthemenobjekte in einem Cluster vorhanden sein, dessen Mitglied dieser Warteschlangenmanager ist, wenn er von ENABLED in DISABLED geändert wird. Alle solchen Themen (auch fern definierte) müssen gelöscht werden, bevor diese Funktion inaktiviert wird.

Weitere Informationen zu **PSCLUS** finden Sie in [ALTER QMGR \(PSCLUS\)](#).

Clusterthemenleistung

Die Leistungsmerkmale von Clusterthemen müssen besonders berücksichtigt werden, da sie sich von den Leistungsmerkmalen von Clusterwarteschlangen unterscheiden und eine schlecht berücksichtigte Nutzung potenziell eine Quelle von Leistungsproblemen in großen oder nicht ausgeglichenen Clustern sein kann.

Auswirkung von Publish/Subscribe auf die Leistung verringern

Es gibt zwei Workloadquellen auf einem Warteschlangenmanager in einem Cluster: die direkte Behandlung von Nachrichten für Anwendungsprogramme und die Handhabung von Nachrichten und Kanälen, die für die Verwaltung des Clusters erforderlich sind. In einem typischen Punkt-zu-Punkt-Cluster ist die Auslastung des Clustersystems weitgehend auf Informationen beschränkt, die bei Bedarf explizit von den Clustermitgliedern angefordert werden (siehe einen Vergleich in [„Leistungsmerkmale von Publish/Subscribe-Clustern“](#) auf Seite 83). Daher können Sie in einem anderen Cluster als einem sehr großen Cluster, z. B. einem Cluster, der Tausende von Warteschlangenmanagern enthält, den Leistungseffekt bei der Verwaltung des Clusters unter Berücksichtigung der Leistung des Warteschlangenmanagers weitgehend abziehen.

In einem Publish/Subscribe-Cluster werden Informationen wie Clusterthemen und Proxy-Subskriptionen an alle Mitglieder eines Clusters übertragen, unabhängig davon, ob alle Clusterwarteschlangenmanager aktiv am Publish/Subscribe-Messaging teilnehmen. Dieser Prozess kann eine erhebliche zusätzliche Belastung für das System verursachen. Daher müssen Sie die Auswirkungen der Clusterverwaltung auf die Leistung des Warteschlangenmanagers sowohl bei der Ablaufsteuerung als auch bei der Größe berücksichtigen.

Beachten Sie die folgenden beiden Vorschläge, um die Auswirkungen des Publish/Subscribe-Cluster-Managements auf die Leistung eines Clusters zu verringern:

1. Führen Sie Cluster-, Topic- und Subskriptionsaktualisierungen zu Off-Peak-Zeiten des Tages aus.
2. Wenn Sie erwägen, Publish/Subscribe-Themen zu einem vorhandenen großen Cluster hinzuzufügen, nur weil der Cluster bereits vorhanden ist, überlegen Sie, ob Sie eine viel kleinere Untergruppe von Warteschlangenmanagern definieren können, die an Publish/Subscribe beteiligt sind, und diesen Cluster zu einem "überlappenden" Cluster machen. Dieser Cluster ist dann der Cluster, in dem die Clusterthemen definiert sind. Obwohl einige WS-Manager jetzt in zwei Clustern vorhanden sind, wird die Gesamtwirkung von Publish/Subscribe reduziert:
 - a. Die Größe des Publish/Subscribe-Clusters ist kleiner.
 - b. Warteschlangenmanager, die sich nicht im Publish/Subscribe-Cluster befinden, sind wesentlich weniger von den Auswirkungen des Clusterverwaltungsdatenverkehrs betroffen.

Balancieren von Produzenten und Verbrauchern

Ein wichtiges Konzept für die asynchrone Messaging-Leistung ist *balance*. Wenn die Nachrichtenkonsumenten nicht mit den Nachrichtenerzeugern in Einklang stehen, besteht die Gefahr, dass sich ein Rückstand nicht konsumierter Nachrichten aufbauen kann und die Leistung mehrerer Anwendungen ernsthaft beeinträchtigt wird.

In einer Punkt-zu-Punkt-Messaging-Topologie ist die Beziehung zwischen Nachrichtenkonsumenten und Nachrichtenproduzenten leicht verständlich. Sie können Schätzungen von Nachrichtenproduktion und -verbrauch, Warteschlange nach Warteschlange, Kanal nach Kanal abrufen. Wenn ein Mangel an Ausgewogenheit besteht, werden die Engpässe schnell erkannt und dann behoben.

Es ist schwieriger zu prüfen, ob Publisher und Subskribenten in einer Publish/Subscribe-Topologie ausgeglichen sind. Beginnen Sie bei jeder Subskription, die in ein Clusterthema aufgelöst wird, und arbeiten Sie zurück zu den Warteschlangenmanagern, die Publisher für das Thema haben. Berechnen Sie die Anzahl der Veröffentlichungen, die von jedem WS-Manager zu jedem Subskribenten fließen.

Jede Veröffentlichung, die einer Subskription auf einem fernen Warteschlangenmanager im Cluster entspricht (basierend auf Proxy-Subskriptionen), wird in die `SYSTEM.CLUSTER.TRANSMIT.QUEUE` eingereiht. Wenn mehrere ferne WS-Manager über Proxy-Subskriptionen für diese Veröffentlichung verfügen, werden mehrere Kopien der Nachricht in die Übertragungswarteschlange eingereiht, die für verschiedene Clustersenderkanäle bestimmt ist.

Diese Veröffentlichungen richten sich an die `SYSTEM.INTER.QMGR.PUBS`-Warteschlange auf den fernen Warteschlangenmanagern. Jeder Warteschlangenmanager verarbeitet Nachrichten, die in dieser Warteschlange ankommen, und stellt sie den richtigen Subskriptionen auf diesem Warteschlangenmanager zu.

Überwachen Sie daher die Auslastung an den folgenden Stellen, an denen Engpässe auftreten können:

- Die einzelnen Subskriptionswarteschlangen selbst:
 - Dieser Engpass bedeutet, dass die subscribierende Anwendung die Veröffentlichungen nicht so schnell konsumiert, wie sie veröffentlicht werden.
- Die `SYSTEM.INTER.QMGR.PUBS`-Warteschlange:
 - Der Warteschlangenmanager empfängt Veröffentlichungen von einem oder mehreren fernen Warteschlangenmanagern schneller, als er sie an die lokalen Subskriptionen verteilen kann.
- Die Clusterkanäle zwischen dem Veröffentlichungswarteschlangenmanager, den subscribierenden Warteschlangenmanagern und den Clusterübertragungswarteschlangen (standardmäßig `SYSTEM.CLUSTER.TRANSMIT.QUEUE`) auf dem Veröffentlichungswarteschlangenmanager:
 - Entweder ist mindestens ein Clusterkanal nicht aktiv oder Nachrichten werden schneller im lokalen Warteschlangenmanager veröffentlicht, als die Kanäle sie dem fernen Warteschlangenmanager zustellen können.
- Wenn die Veröffentlichungsanwendung eine Publish/Subscribe-Schnittstelle in der Warteschlange verwendet, müssen auch die `SYSTEM.BROKER.DEFAULT.STREAM`-Warteschlange und alle anderen Datenstromwarteschlangen, die in der `SYSTEM.QPUBSUB.QUEUE.NAMELIST` aufgelistet sind, sowie die `SYSTEM.BROKER.DEFAULT.SUBPOINT`-Warteschlange und alle anderen Unterpunktwarteschlangen, die in der `SYSTEM.QPUBSUB.SUBPOINT.NAMELIST` aufgelistet sind, berücksichtigt werden:
 - Nachrichten werden von lokalen Veröffentlichungsanwendungen schneller eingereiht, als der lokale Warteschlangenmanager die Nachrichten verarbeiten kann.

Hinweise zur Subskriptionsleistung

Wie zuvor beschrieben, muss bei einer Subskription auf einem Warteschlangenmanager für eine Themenzeihenfolge, die in ein Clusterthema aufgelöst wird, dieser Warteschlangenmanager sicherstellen, dass jeder andere Warteschlangenmanager im Cluster über eine Proxy-Subskription für das Thema verfügt. Um dieses Ergebnis zu erzielen, erstellt und sendet der Warteschlangenmanager eine Proxy-Subskription an jeden anderen Warteschlangenmanager im Cluster.

Wenn Sie die Standardkonfiguration verwenden, führt die Erstellung einer Subskription für ein Clusterthema nur dann nicht dazu, dass neue Proxy-Subskriptionen gesendet werden, wenn auf dem lokalen Warteschlangenmanager bereits eine Subskription für genau dieselbe Themenzeichenfolge vorhanden ist. In dieser Situation sind keine zusätzlichen Proxy-Subskriptionen erforderlich, da eingehende Veröffentlichungen allen übereinstimmenden Subskriptionen zugestellt werden, nicht nur der ursprünglichen Subskription für die Themenzeichenfolge.

Eine Alternative zur Standardkonfiguration finden Sie unter [„Einzelne Proxy-Subskriptionen inaktivieren“](#) auf Seite 82.

Subskriptionsselektoren werden nicht berücksichtigt, sodass zwei Subskriptionen für dieselbe Themenzeichenfolge, aber mit unterschiedlichen Selektoren weiterhin Proxy-Subskriptionen gemeinsam nutzen. Dies kann auch bedeuten, dass Veröffentlichungen, die mit der Themenzeichenfolge übereinstimmen, an den Warteschlangenmanager des Subskribenten weitergegeben werden, auch wenn die Veröffentlichung nicht mit dem Selektor der Subskription übereinstimmt.

Eine entsprechende Nachricht für die Proxy-Subskriptionsnachricht wird erstellt und an alle Warteschlangenmanager gesendet, wenn die letzte Subskription für eine Themenzeichenfolge aus einem Warteschlangenmanager gelöscht wird. Dieser Prozess entfernt die Proxy-Subskriptionen aus den fernen Warteschlangenmanagern.

Aus diesen Gründen können die Größe des Clusters und die Häufigkeit von Subskriptionen für verschiedene Themenzeichenfolgen eine erhebliche Belastung des Clusters selbst darstellen und müssen bei der Planung des Clusters und der von den Publish/Subscribe-Anwendungen zu verwendenden Themen berücksichtigt werden.

Wenn Sie die Auslastung des Systems durch den Proxy-Subskriptionsdatenverkehr berücksichtigen, überwachen Sie zusätzlich zu den im Abschnitt [„Balancieren von Produzenten und Verbrauchern“](#) auf Seite 80 aufgelisteten Warteschlangen auch die folgenden Warteschlangen.

- Die Warteschlange SYSTEM.INTER.QMGR.FANREQ auf dem Subskribentenwarteschlangenmanager.
- Die Warteschlange SYSTEM.INTER.QMGR.CONTROL auf allen anderen Warteschlangenmanagern im Cluster.

Jeder signifikante Nachrichtenrückstand in diesen Warteschlangen impliziert, dass entweder die Änderungsrate der Subskription für das System zu hoch ist oder ein WS-Manager im Cluster nicht ordnungsgemäß funktioniert. Entweder weil die Publish/Subscribe-Unterstützung inaktiviert ist (siehe **PSMODE** in [ALTER QMGR](#)) oder weil ein Problem auftritt, das eine weitere Untersuchung erfordert. Überprüfen Sie dann die Fehlerprotokolle des Warteschlangenmanagers.

Datenverkehr für Proxy-Subskription reduzieren

Wenn der Aufwand für die Proxy-Subskription hoch ist, sollten Schritte unternommen werden, um dies zu reduzieren. Dies kann durch allgemeine Themenkonsolidierung oder durch Wechsel zu einem Broadcast-Modell für Veröffentlichungen zwischen Warteschlangenmanagern möglich sein.

Es ist eine allgemeine Publish/Subscribe-Empfehlung, dass die Verwendung von Themenzeichenfolgen bewertet wird, um festzustellen, ob sie konsolidiert werden können, um die Gesamtbelastung der Ressourcen für das System zu reduzieren. Die Verwendung vieler unterschiedlicher, transientser Themenzeichenfolgen führt zu einem gewissen Verwaltungsaufwand für jeden Warteschlangenmanager in dem System, an dem Publisher oder Subskriptionen angehängt sind. Durch die Reduzierung der Anzahl und des transienten Charakters von Themenzeichenfolgen und damit der Publisher und Subskriptionen für diese Themenzeichenfolgen verringert sich die Auswirkung auf das System.

Eine Methode zum Reduzieren des Proxy-Subskriptionsdatenverkehrs besteht darin, Subskriptionen für dieselbe Themenzeichenfolge auf demselben Warteschlangenmanager zu lokalisieren. Diese Methode ermöglicht es diesem Warteschlangenmanager, eine einzelne Proxy-Subskription an die anderen Warteschlangenmanager zu senden, anstatt mehrere Warteschlangenmanager zu haben, die Proxy-Subskriptionen senden, jeweils für ihre eigene Gruppe von Subskriptionen für dieselbe Themenzeichenfolge. Dieses Verfahren optimiert auch das Routing von Veröffentlichungen im Cluster.

Einzelne Proxy-Subskriptionen inaktivieren

In einigen Situationen, in denen die Gruppe unterschiedlicher Themenzeichenfolgen, die in einem Cluster subskribiert werden, groß ist und sich ständig ändert, kann es besser sein, von einem Weitergabemodell für Subskriptionen zu einem Broadcastmodell für Veröffentlichungen zu wechseln. Bei diesem bevorzugten Modell wird jede Veröffentlichung zu einem Clusterthema automatisch an jeden Warteschlangenmanager im Cluster gesendet, unabhängig davon, ob Subskriptionen auf diesen Warteschlangenmanagern vorhanden sind.

Die empfangenden Warteschlangenmanager können dann die Nachrichten an die vorhandenen lokalen Subskriptionen zustellen oder die Nachricht löschen. In diesem Modell müssen keine einzelnen Proxy-Subskriptionen erstellt und gelöscht werden, wenn Subskriptionen vorhanden sind. Bei Ausführung in diesem Modus ist es wahrscheinlich, dass die Ressourcenauslastung der veröffentlichten Nachrichten zunimmt, wenn alle Veröffentlichungen an alle Warteschlangenmanager gesendet werden. Daher müssen die Warteschlangenmanager im Cluster über die Kapazität zur Verarbeitung dieser zusätzlichen Arbeitslast verfügen.

Aktivieren Sie ein Broadcastmodell mit den folgenden Konfigurationsschritten:

1. Jeder Warteschlangenmanager, der Subskriptionen hostet, muss so konfiguriert werden, dass er keine Proxy-Subskriptionen sendet, die mit lokalen Subskriptionen zu Clusterthemen übereinstimmen. Für diese Konfiguration muss der folgende Optimierungsparameter in jeder `qm.ini`-Datei des Warteschlangenmanagers festgelegt werden, bevor Clusterthemen definiert oder Subskriptionen im Cluster erstellt werden:

```
TuningParameters:  
  pscProxySubFlags=1
```

2. Nachdem der Optimierungsparameter festgelegt wurde, müssen alle Warteschlangenmanager erneut gestartet werden.
3. Nach dem Neustart der Warteschlangenmanager können die Clusterthemen definiert werden. Jedes Clusterthema muss **PROXYSUB** auf **FORCE** setzen.

Verhalten umkehren

Führen Sie die folgenden Schritte aus, um den zuvor in „[Einzelne Proxy-Subskriptionen inaktivieren](#)“ auf Seite 82 beschriebenen Betriebsmodus umzukehren:

1. Entfernen Sie den Optimierungsparameter für jeden Warteschlangenmanager aus der Datei `qm.ini`.
2. Starten Sie jeden Warteschlangenmanager erneut.
3. Setzen Sie den Befehl **REFRESH QMGR TYPE (PROXYSUB)** auf jedem Warteschlangenmanager ab, der Subskriptionen hostet.
4. Setzen Sie **PROXYSUB** für das Clusterthema bzw. die Clusterthemen auf **FIRSTUSE**.



Vorsicht: Wenn sowohl bei der Aktivierung als auch bei der Umkehrung dieses Verhaltens alle Schritte nicht in der dokumentierten Reihenfolge ausgeführt werden, findet möglicherweise kein korrekter Ablauf von Veröffentlichungen zu Subskriptionen statt.

Hinweis: Auswirkung der Einstellung PROXYSUB (auf FORCE)

Wie zuvor in diesem Abschnitt beschrieben, kann das Topic-Attribut **PROXYSUB (FORCE)** den Datenverkehr der Proxy-Subskription reduzieren, aber es muss mit Vorsicht verwendet werden. Das Attribut **PROXYSUB (FORCE)** wird an jeden Warteschlangenmanager im Cluster weitergegeben, nicht nur an den Warteschlangenmanager, auf dem das Thema definiert wurde. Dies führt sofort dazu, dass jeder Warteschlangenmanager im Cluster eine Proxy-Subskription mit Platzhalterzeichen für jeden anderen Warteschlangenmanager erstellt. Das Ergebnis dieses Prozesses ist, dass jeder Warteschlangenmanager Clustersenderkanäle zu jedem anderen Warteschlangenmanager erstellt und alle veröffentlichten Nachrichten an jeden Warteschlangenmanager gesendet werden.

Das Festlegen dieser Eigenschaft in einem großen oder ausgelasteten Cluster kann zu einer zusätzlichen Belastung der Systemressourcen führen.

Leistungsmerkmale von Publish/Subscribe-Clustern

Es ist wichtig zu berücksichtigen, wie sich eine Änderung der Attribute eines Publish/Subscribe-Clusters wie das Hinzufügen eines Warteschlangenmanagers, eines Themas oder einer Subskription zum Cluster auf die Leistung von Anwendungen auswirkt, die im Cluster ausgeführt werden.

Einen Punkt-zu-Punkt-Cluster mit einem Publish/Subscribe-Cluster in Bezug auf zwei Management-Tasks vergleichen.

Zunächst ein Punkt-zu-Punkt-Cluster:

1. Wenn eine neue Clusterwarteschlange definiert wird, werden die Zielinformationen mit einer Push-Operation an die Warteschlangenmanager mit vollständigem Repository übertragen und nur an andere Cluster-Member gesendet, wenn sie zum ersten Mal auf eine Clusterwarteschlange verweisen (z. B. versucht eine Anwendung, sie zu öffnen). Diese Informationen werden dann lokal vom WS-Manager zwischengespeichert, um die Notwendigkeit zu entfernen, die Informationen bei jedem Zugriff auf die Warteschlange fern abzurufen.
2. Das Hinzufügen eines WS-Managers zu einem Cluster wirkt sich nicht direkt auf die Auslastung anderer WS-Manager aus. Die Informationen zum neuen Warteschlangenmanager werden in die vollständigen Repositories verschoben, aber die Kanäle für den neuen Warteschlangenmanager von anderen WS-Managern im Cluster werden nur erstellt und gestartet, wenn der Datenverkehr zum oder vom neuen WS-Manager fließt.

Kurz gesagt, die Arbeitslast auf einem Warteschlangenmanager in einem Punkt-zu-Punkt-Cluster hängt mit dem Nachrichtenverkehr zusammen, den er für Anwendungsprogramme verarbeitet, und ist nicht direkt mit der Größe des Clusters verbunden.

Zweitens ein Publish/Subscribe-Cluster:

1. Wenn ein neues Clusterthema definiert wird, werden die Informationen mit einer Push-Operation an die Warteschlangenmanager mit vollständigem Repository und von dort direkt an alle Mitglieder des Clusters übertragen. Dies führt dazu, dass Kanäle für jedes Member des Clusters aus den vollständigen Repositories gestartet werden, wenn sie noch nicht gestartet wurden.
2. Wenn eine Subskription für ein Clusterthema in einer neuen Themenzeichenfolge erstellt wird, werden die Informationen direkt von diesem Warteschlangenmanager an alle anderen Member des Clusters übertragen, wodurch Kanäle für jedes Member des Clusters von diesem Warteschlangenmanager aus gestartet werden, falls dies noch nicht begonnen hat.
3. Wenn ein neuer Warteschlangenmanager in einen vorhandenen Cluster aufgenommen wird, werden Informationen zu allen Clusterthemen aus dem vollständigen WS-Manager-Repository an ihn übertragen. Der neue Warteschlangenmanager synchronisiert dann das Wissen über alle Subskriptionen für Clusterthemen im Cluster mit allen Mitgliedern des Clusters, wodurch Kanäle für jedes Mitglied des Clusters vom neuen Warteschlangenmanager erstellt und gestartet werden.

Zusammenfassend lässt sich sagen, dass die Clusterverwaltungslast auf jedem Warteschlangenmanager im Cluster mit der Anzahl der Warteschlangenmanager, Clusterthemen und Proxy-Subskriptionen im Cluster wächst, unabhängig von der lokalen Verwendung dieser Clusterthemen auf jedem Warteschlangenmanager.

Wichtige Rollen für Publish/Subscribe-Cluster-Warteschlangenmanager

Ähnlich wie Punkt-zu-Punkt-Cluster gibt es zwei Schlüsselrollen für Warteschlangenmanager in einem Publish/Subscribe-Cluster: als Warteschlangenmanager mit vollständigem Repository und als Clusterthemenhosts.

Vollständiger Repository

Ein Warteschlangenmanager mit vollständigem Repository hat die Aufgabe, Objektdefinitionen an andere Mitglieder eines Clusters zu übertragen. Bei Publish/Subscribe-Clustern werden Clusterthemenobjektdefinitionen an andere Mitglieder des Clusters übertragen.

Cluster-Topic-Host

Ein Clusterthemenhost ist ein Warteschlangenmanager, in dem ein Clusterthemenobjekt definiert ist. Sie können Clusterthemenobjekte auf jedem Warteschlangenmanager im Publish/Subscribe-Cluster definieren. Das Clusterthemenobjekt wird mit Push-Operation an die Warteschlangenmanager mit

vollständigem Repository übertragen, die es dann mit Push-Operation an alle anderen Warteschlangenmanager im Cluster übertragen, wo es für die Verwendung durch Publisher und Subskribenten, die auf allen Warteschlangenmanagern im Cluster ausgeführt werden, zwischengespeichert wird.

Verfügbarkeit und Management

Sie sollten zwei vollständige Repositories in einem Cluster definieren, um die Verfügbarkeit von Clusterthemendefinitionen im Cluster zu maximieren.

Wie bei in Warteschlangen eingereichten Messaging-Clustern ist es in Publish/Subscribe-Clustern, die nur zwei hoch verfügbare Computer unter vielen Computern haben, sinnvoll, die hoch verfügbaren Computer als vollständige Repositories zu definieren.

In Clustern in Warteschlangen können Sie die Verfügbarkeit und den Durchsatz einer Clusterwarteschlange erhöhen, indem Sie dieselbe Clusterwarteschlange auf mehreren Warteschlangenmanagern im Cluster *definieren*. Nachrichten werden dann gleichmäßig auf sie verteilt. In Publish/Subscribe-Clustern hingegen ist ein Clusterthema auf allen Warteschlangenmanagern im Cluster *verfügbar*, es erfolgt jedoch kein Lastausgleich für Publish/Subscribe-Datenverkehr. Stattdessen sollten separate Subskriptionen und Publisher auf verschiedene Warteschlangenmanager verteilt werden, um die Publish/Subscribe-Last zu verteilen. Wenn der Warteschlangenmanager, auf dem Sie das Clusterthema definiert haben, nicht mehr verfügbar ist, verarbeiten die anderen WS-Manager weiterhin Publish/Subscribe-Anforderungen für das Thema.

Wenn der Warteschlangenmanager, auf dem Sie das Clusterthemenobjekt definiert haben, jedoch nie wieder verfügbar gemacht wird, werden die zwischengespeicherten Themenobjekte auf den anderen Warteschlangenmanagern gelöscht und das Thema ist nicht mehr verfügbar. Dieser Prozess findet nach mindestens 60 Tagen (je nachdem, wann die Themendefinition zuletzt aktualisiert wurde) ab dem Zeitpunkt der Nichtverfügbarkeit der Themendefinition statt.

Mit dem 60-Tage-Zeitraum zur Wiederherstellung des Warteschlangenmanagers, auf dem Sie Clusterthemenobjekte definiert haben, sind keine besonderen Maßnahmen erforderlich, um einen Clusterthemenhost hoch verfügbar zu machen. Der Zeitraum von 60 Tagen reicht aus, um technischen Problemen gerecht zu werden; der Zeitraum von 60 Tagen wird wahrscheinlich nur aufgrund von Verwaltungsfehlern überschritten. Um diese Möglichkeit abzuschwächen, schreiben alle Member des Clusters stündlich Fehlerprotokollnachrichten, dass ihr zwischengespeichertes Clusterthemenobjekt nicht aktualisiert wurde, wenn der Clusterthemenhost nicht verfügbar ist. Reagieren Sie auf diese Nachricht, indem Sie sicherstellen, dass der Warteschlangenmanager, auf dem das Clusterthemenobjekt definiert ist, aktiv ist.

Sie können das Verfahren zum Definieren desselben Clusterthemenobjekts auf anderen Warteschlangenmanagern anwenden. Jede Definition führt dazu, dass ein zusätzliches Clusterthemenobjekt mit Push-Operation an die anderen Warteschlangenmanager im Cluster übertragen wird, einschließlich der anderen Clusterthemenhosts. Wenn nun ein Cluster-Topic-Host über 60 Tage nicht mehr verfügbar ist, wird nur seine Version des Cluster-Topic-Objekts von den anderen Hosts entfernt. Die anderen Versionen des Clusterthemenobjekts bleiben erhalten. Es ist erforderlich, dass alle Definitionen für ein bestimmtes Thema in einem Cluster identisch sind. Andernfalls ist es schwierig festzustellen, welche Themendefinition von einem Warteschlangenmanager verwendet wird. Die neueste Kopie auf einem Host ist immer das verwendete Clusterthemenobjekt.

Wägen Sie den zusätzlichen Schutz mehrerer Clusterthemendefinitionen gegen eine erhöhte Verwaltungskomplexität ab: Bei zunehmender Komplexität besteht eine größere Wahrscheinlichkeit von menschlichem Fehler.

Im Gegensatz zum Hosting einer Clusterwarteschlange führt der Host-Warteschlangenmanager für eine Clusterthemendefinition keinen zusätzlichen Anwendungsnachrichtenverkehr ein. Dieser Datenverkehr ist auf die Warteschlangenmanager beschränkt, auf denen die Subskriptionen erstellt und die Nachrichten veröffentlicht werden. Es ist möglich, das Clusterthema auf einem Warteschlangenmanager zu hosten, der dies nicht tut. Diese Situation bedeutet, dass es zwar nicht obligatorisch ist, dass es jedoch oft sinnvoll ist, die Clusterthemen auf den Warteschlangenmanagern mit vollständigem Repository für den Cluster zu hosten, da diese Warteschlangenmanager möglicherweise mit höheren Verfügbarkeitsstufen bereitgestellt werden und eine engere administrative Steuerung auf ihnen haben. Diese Anordnung verrin-

gert die Möglichkeit, die Definitionen oder sogar den Warteschlangenmanager falsch zu ändern oder zu löschen.

Überlappende Clusterunterstützung und Publish/Subscribe

Bei IBM WebSphere MQ -Clustern kann ein einzelner Warteschlangenmanager Mitglied mehrerer Cluster sein. Diese Anordnung wird als überlappende Cluster bezeichnet. Clusterthemen in Publish/Subscribe-Clustern verhalten sich anders als Warteschlangen, wenn Cluster in einem Warteschlangenmanager überlappt werden. Dieses Verhalten muss klar verstanden werden, wenn Cluster-Publish/Subscribe mit überlappenden Clustern verwendet werden.

Im Gegensatz zu einer Warteschlange gibt es keine Möglichkeit, eine Themendefinition mehreren Clustern zuzuordnen. Daher ist der Bereich der in einem Cluster erstellten Proxy-Subskriptionen auf den einzelnen Cluster beschränkt, in dem das Clusterthema definiert ist. Jeder Warteschlangenmanager verfügt jedoch über eine einzelne Themenstruktur, die alle lokalen Themen und alle bekannten Clusterthemen aus allen Clustern enthält, zu denen sie gehören. Aus diesem Grund ist es möglich, ein solches System zu planen, bei dem das Publish/Subscribe-Verhalten schwer verständlich ist.

Mehrere Publish/Subscribe-Cluster integrieren

Bei Punkt-zu-Punkt-Nachrichten besteht ein Grund dafür, dass ein einzelner Warteschlangenmanager Mitglied mehrerer Cluster ist, darin, ein Cluster-Gateway zwischen zwei Clustern zu erstellen. Weitere Informationen zu diesem Thema finden Sie unter [Überlappende Cluster](#) . Dieses Cluster-Gateway ermöglicht die Weiterleitung von Punkt-zu-Punkt-Nachrichten, die aus einem Cluster stammen, an Abfragen in einem anderen Cluster. Publish/Subscribe-Cluster übernehmen die Möglichkeit, von traditionellen WS-Manager-Clustern überlappt zu werden. Sie können diesen Mechanismus jedoch nicht verwenden, um Veröffentlichungen und Subskriptionen von einem Cluster an einen anderen weiterzuleiten.

Um Veröffentlichungen und Subskriptionen von Warteschlangenmanagern in einem Cluster an einen anderen zu übergeben, müssen Sie die Warteschlangenmanager über eine Publish/Subscribe-Hierarchie miteinander verknüpfen. Diese Anordnung kann erreicht werden, indem explizit eine hierarchische Beziehung zwischen einem Warteschlangenmanager in einem Cluster und einem anderen Warteschlangenmanager in dem anderen Cluster erstellt wird. Diese Beziehung ermöglicht den Ablauf aller Proxy-Subskriptionen zwischen den Clustern und damit alle übereinstimmenden Veröffentlichungen. Weitere Informationen zu dieser Beziehung finden Sie unter [„Publish/Subscribe-Hierarchien“](#) auf [Seite 88](#).

Eine Möglichkeit, den Fluss von Veröffentlichungen und Subskriptionen zwischen Clustern zu begrenzen, besteht darin, einen Gateway-WS-Manager zu verwenden, der sich in keinem Cluster befindet ([siehe „Themenbereiche in mehreren Clustern kombinieren und isolieren“](#) auf [Seite 107](#)).

Überlappende Cluster, einzelne Themenstruktur

Jeder Warteschlangenmanager verfügt über eine [Themenstruktur](#) , die lokale Themen und alle bekannten Clusterthemen enthält. Eine weitere Überlegung mit überlappenden zwei Clustern, die beide Publish/Subscribe verwenden, besteht darin, dass ein Warteschlangenmanager in jedem Cluster ein Clusterthema mit demselben Namen oder unterschiedlich benannte Clusterthemen mit derselben Themenzeichenfolge definieren kann. Auf den Warteschlangenmanagern, die Mitglieder beider Cluster sind, treten Konflikte auf, wenn sie über die mehreren Clusterthemendefinitionen informiert werden, eine für jeden Cluster. Der Warteschlangenmanager hat ein Problem gemeldet, aber der Warteschlangenmanager arbeitet weiterhin und verwendet nur die neueste Clusterthemendefinition. Daher wird das Verhalten nicht deterministisch und kann nicht als Grundlage herangezogen werden.

Aus diesem Grund müssen sich überschneidende Cluster, die Cluster-Publish/Subscribe verwenden, ihren Topicdefinitionsnamensbereich für alle Cluster berücksichtigen und ihre Topicobjekte benennen und ihre Topiczeichenfolgen entsprechend strukturieren. Anschließend können Sie Warteschlangenmanager in der Überlappung verwenden, um beide Cluster vorhersehbar zu veröffentlichen und zu abonnieren.

In [Abbildung 33](#) auf [Seite 86](#) sind T_B und T_C Themendefinitionen, die sich nicht überschneiden. Ein Publisher, der mit QM3 verbunden ist und sich im Cluster überschneidet, kann in beiden Topics in ihren jeweiligen Clustern veröffentlichen. Ein Subskribent, der in der Überlappung mit QM3 verbunden ist, kann Themen in beiden Clustern abonnieren.

Eine alternative Möglichkeit, über [Abbildung 33 auf Seite 86](#) nachzudenken, besteht darin, die Proxy-Subskriptionen zu berücksichtigen. Eine Anwendung, die mit dem Warteschlangenmanager QM3 verbunden ist und ein Thema subskribiert, das in das Themenobjekt T_B aufgelöst wird (das nur in CLUSTER 1 vorhanden ist), führt dazu, dass Proxy-Subskriptionen vom Warteschlangenmanager QM3 an Warteschlangenmanager QM1 und QM2 gesendet werden. Eine Anwendung, die mit Warteschlangenmanager QM3 verbunden ist, subskribiert ein Thema, das in das Themenobjekt T_C aufgelöst wird (das nur in CLUSTER 2 vorhanden ist). Die Subskription führt dazu, dass Proxy-Subskriptionen vom Warteschlangenmanager QM3 nur an Warteschlangenmanager QM4 und QM5 gesendet werden.

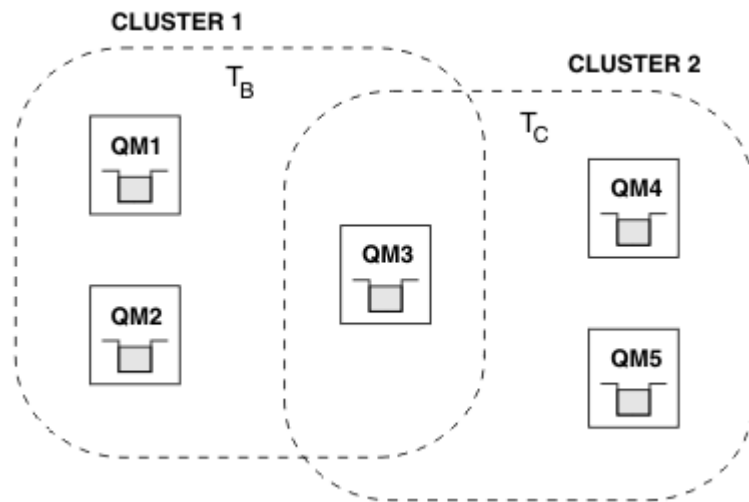


Abbildung 33. Überlappende Cluster: Zwei Cluster, die jeweils verschiedene Themen subskribieren

Publisher und Subskribenten von Warteschlangenmanagern, die sich nicht in der Überlappung befinden, können nur Themen in ihrem Cluster veröffentlichen und subskribieren. Beispiel: Ein Subskribent einer Themenzeichenfolge unter QM2 empfängt keine Nachrichten, die zu derselben Themenzeichenfolge veröffentlicht wurden, die von QM5 veröffentlicht wurde, unabhängig vom Clustering der Themen. Um diese Anordnung zu erreichen, ist eine Publish/Subscribe-Hierarchie erforderlich.

Überlappende Cluster, Subskriptionen mit Platzhaltern

Aus den Gründen im vorherigen Abschnitt dieses Abschnitts muss bei der Verwendung von Platzhalterzeichen für die Subskription von Themen in einem Warteschlangenmanager, der Mitglied mehrerer Cluster ist, Vorsicht geboten werden.

Im vorherigen Beispiel wird angenommen, dass die beiden Themenobjekte wie folgt konfiguriert wurden:

- T_B : Themenname 'Football', Cluster 'CLUSTER1'. Themenzeichenfolge '/Sport/Football'
- T_C : Themenname 'Tennis', Cluster 'CLUSTER2'. Themenzeichenfolge '/Sport/Tennis'

In diesem Szenario sind die beiden Clusterthemen klar voneinander getrennt, ohne Überlappung im Themennamen oder in der Themenzeichenfolge.

Eine Anwendung, die mit QM3 verbunden ist, kann eine Subskription für '/Sport/Football' und eine Subskription für '/Sport/Tennis' erstellen. Sie erhalten dann alle Veröffentlichungen aus den beiden Clustern. Wie in „[Verwaltungsthemenobjekte](#)“ auf Seite 39 beschrieben, wird dieses Modell jedoch nicht als Clusterthema in einem der Cluster erkannt und es werden keine Proxy-Subskriptionen erstellt, wenn sie '/Sport/#' subskribieren und sowohl unter '/Sport/Football' als auch unter '/Sport/Tennis' Veröffentlichungen empfangen wollen. Sie würden dann Veröffentlichungen von anderen Warteschlangenmanagern in beiden Clustern verpassen.

Wie bereits beschrieben, ist es nicht zulässig, ein Clusterthema für '/Sport/#' sowohl in CLUSTER 1 als auch in CLUSTER 2 zu erstellen, da diese Clusterthemen Konflikte verursachen würden und Informationsnachrichten in die Fehlerprotokolle geschrieben werden, um dies anzugeben. However, it is 'allowed' to create such a topic in just one of the clusters, say CLUSTER 1. Jetzt würde eine Subskription für '/Sport/#' in QM3 dazu führen, dass Proxy-Subskriptionen nur an die Warteschlan-

genmanager in CLUSTER 1 gesendet werden, sodass Veröffentlichungen für ' /Sport/Tennis ' von QM4 oder QM5 weiterhin nicht empfangen werden können.

Die einzige Lösung in diesem Szenario ist die Fortsetzung der Erstellung von zwei separaten Subskriptionen.

Hinweise zu REFRESH CLUSTER für Publish/Subscribe-Cluster

Die Ausgabe des Befehls **REFRESH CLUSTER** führt dazu, dass der Warteschlangenmanager vorübergehend lokal gehaltene Informationen zu einem Cluster löscht, einschließlich aller Clusterthemen und der zugehörigen Proxy-Subskriptionen.

Die Zeit, die von der Ausgabe des Befehls **REFRESH CLUSTER** bis zu dem Punkt, an dem der Warteschlangenmanager die erforderlichen Informationen für das Cluster-Publish/Subscribe erhält, benötigt wird, hängt von der Größe des Clusters, der Verfügbarkeit und der Reaktionsfähigkeit der Warteschlangenmanager mit vollständigem Repository ab.

Während der Aktualisierungsverarbeitung erfolgt die Unterbrechung des Publish/Subscribe-Datenverkehrs in einem Publish/Subscribe-Cluster. Bei großen Clustern kann die Verwendung des Befehls **REFRESH CLUSTER** den Cluster unterbrechen, während er in Bearbeitung ist, und danach in 27-Tage-Intervallen, wenn die Clusterobjekte automatisch Statusaktualisierungen an alle interessierten Warteschlangenmanager senden. Nähere Informationen hierzu erhalten Sie im Abschnitt Die Aktualisierung in einem großen Cluster kann sich auf die Leistung und Verfügbarkeit auswirken. Aus diesen Gründen darf der **REFRESH CLUSTER**-Befehl nur in einem Publish/Subscribe-Cluster verwendet werden, wenn er von Ihrem IBM Support Center geleitet wird.

Die Unterbrechung des Clusters kann extern als die folgenden Symptome auftreten:

- Subskriptionen zu Clusterthemen in diesem WS-Manager erhalten keine Veröffentlichungen von Publishern, die mit anderen WS-Managern im Cluster verbunden sind.
- Nachrichten, die in Clusterthemen in diesem Warteschlangenmanager veröffentlicht werden, werden nicht an Subskriptionen auf anderen Warteschlangenmanagern weitergegeben.
- Subskriptionen für Clusterthemen in diesem Warteschlangenmanager, die in diesem Zeitraum erstellt wurden, senden nicht konsistent Proxy-Subskriptionen an andere Mitglieder des Clusters.
- Subskriptionen für Clusterthemen auf diesem Warteschlangenmanager, die in diesem Zeitraum gelöscht wurden, entfernen nicht konsistent die Proxy-Subskriptionen von anderen Mitgliedern des Clusters.
- 10-Sekunden-Pausen oder länger, bei Nachrichtenübermittlung.
- **MQPUT**-Fehler, z. B. MQRC_PUBLICATION_FAILURE.
- Veröffentlichungen, die in der Warteschlange für nicht zustellbare Nachrichten mit dem Grund MQRC_UNKNOWN_REMOTE_Q_MGR platziert wurden

Aus diesen Gründen müssen Publish/Subscribe-Anwendungen in den Quiescemodus versetzt werden, bevor der Befehl **REFRESH CLUSTER** ausgegeben wird.

Siehe auch Hinweise zur Verwendung von REFRESH CLUSTER und Clustering: Best Practices für REFRESH CLUSTER verwenden.

Nachdem ein **REFRESH CLUSTER** -Befehl auf einem WS-Manager in einem Publish/Subscribe-Cluster ausgegeben wurde, warten Sie, bis alle Cluster-Warteschlangenmanager und Clusterthemen erfolgreich aktualisiert wurden, und synchronisieren Sie dann die Proxy-Subskriptionen wie in „Resynchronisation von Proxy-Subskriptionen“ auf Seite 74 beschrieben. Diese Anordnung erfordert, dass Clustersenderkanäle von diesem Warteschlangenmanager zu allen anderen Warteschlangenmanagern im Cluster gestartet werden. Wenn alle Proxy-Subskriptionen ordnungsgemäß resynchronisiert wurden, starten Sie Ihre Publish/Subscribe-Anwendungen erneut.

Wenn die Ausführung eines **REFRESH CLUSTER** -Befehls viel Zeit in Anspruch nimmt, können Sie ihn überwachen, indem Sie sich die CURDEPTH von SYSTEM.CLUSTER.COMMAND.QUEUE ansehen.

Zugehörige Konzepte

Anwendungsprobleme bei der Ausführung von REFRESH CLUSTER

Clustering: Best Practices für REFRESH CLUSTER verwenden

Zugehörige Verweise

MQSC-Befehlsreferenz: REFRESH CLUSTER

Publish/Subscribe-Hierarchien

Warteschlangenmanager können in einer Hierarchie gruppiert werden, wobei die Hierarchie einen oder mehrere Warteschlangenmanager enthält, die direkt verbunden sind. Warteschlangenmanager werden über eine Beziehung zwischen übergeordnetem und untergeordnetem Element zur Verbindungszeit miteinander verbunden. Wenn zwei Warteschlangenmanager zum ersten Mal miteinander verbunden werden, wird der untergeordnete Warteschlangenmanager mit dem übergeordneten Warteschlangenmanager verbunden.

Wenn die übergeordneten und untergeordneten Warteschlangenmanager in einer Hierarchie verbunden sind, gibt es keinen funktionalen Unterschied zwischen ihnen, bis Sie die Verbindung zwischen Warteschlangenmanagern und der Hierarchie trennen.

Anmerkung: Hierarchische IBM WebSphere MQ -Verbindungen erfordern, dass das Warteschlangenmanagerattribut PSMODE auf ENABLED gesetzt ist.

Hierarchy

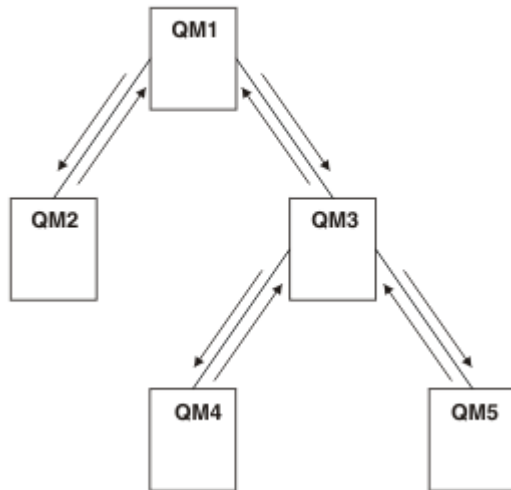


Abbildung 34. Einfache Publish/Subscribe-Hierarchie

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.

Beispiel für Publish/Subscribe-Hierarchie: Szenario 1

Richten Sie eine Publish/Subscribe-Hierarchietopologie unter Verwendung von Punkt-zu-Punkt-Kanälen mit dem Aliasnamen des Warteschlangenmanagers ein.

Informationen zu diesem Vorgang

Diese Szenarios richten eine Publish/Subscribe-Hierarchie auf unterschiedliche Weise ein, um die Verbindung zwischen Warteschlangenmanagern herzustellen. Diese Szenarien verwenden alle einen übergeordneten Warteschlangenmanager mit dem Namen QM1 und zwei untergeordnete Warteschlangenmanager mit dem Namen QM2 und QM3.

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

Szenario 1, Teil 1: Warteschlangenmanager erstellen

Vorgehensweise

1. Erstellen und starten Sie mit folgenden Befehlen die drei Warteschlangenmanager QM1, QM2 und QM3:

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

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

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

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

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

Informationen zu diesem Vorgang

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

Vorgehensweise

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

```
DEFINE QLOCAL(QM1.XMITQ) USAGE(XMITQ)
DEFINE QREMOTE (QM1) RNAME('') RQMNAME(QM1) XMITQ(QM1.XMITQ)
DEFINE CHANNEL('QM2.TO.QM1') CHLTYPE(SDR) CONNAME('localhost(9999)') XMITQ(QM1.XMITQ)
TRPTYPE(TCP)
DEFINE CHANNEL('QM1.TO.QM2') CHLTYPE(RCVR) TRPTYPE(TCP)
```

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

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

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

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

4. Starten Sie die entsprechenden Empfangsprogramme auf den Warteschlangenmanagern:

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

5. Starten Sie folgende Kanäle:

- a. Auf QM1:

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

b. Auf QM2:

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

c. Auf QM3:

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

6. Überprüfen Sie, ob alle Kanäle gestartet wurden:

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

Szenario 1, Teil 3: Warteschlangenmanager verbinden und ein Thema definieren

Informationen zu diesem Vorgang

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

Vorgehensweise

1. Setzen Sie auf QM2 und QM3 den übergeordneten Warteschlangenmanager auf QM1:

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

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

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

3. Definieren Sie ein Themenobjekt:

```
define topic(FOOTBALL) TOPICSTR('Sport/Soccer')
```

Szenario 1, Teil 4: Thema veröffentlichen und abonnieren

Informationen zu diesem Vorgang

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

Vorgehensweise

1. Führen Sie im ersten Befehlsfenster folgenden Befehl aus:

```
amqspub Sport/Soccer QM2
```

2. Führen Sie im zweiten Befehlsfenster folgenden Befehl aus:

```
amqssub Sport/Soccer QM1
```

3. Führen Sie im dritten Befehlsfenster folgenden Befehl aus:

```
amqssub Sport/Soccer QM3
```

Ergebnisse

Die amqssub.exe-Anwendungen im zweiten und dritten Befehlsfenster empfangen die Nachrichten, die im ersten Befehlsfenster veröffentlicht wurden.

Zugehörige Tasks

„[Beispiel für Publish/Subscribe-Hierarchie: Szenario 2](#)“ auf Seite 93

Richten Sie eine Publish/Subscribe-Hierarchietopologie unter Verwendung von Punkt-zu-Punkt-Kanälen ein, wobei der Name der Übertragungswarteschlange mit dem Namen des fernen Warteschlangenmanagers identisch ist.

„[Beispiel für Publish/Subscribe-Hierarchie: Szenario 3](#)“ auf Seite 95

Hinzufügen eines Warteschlangenmanagers zu einer Hierarchietopologie unter Verwendung eines Clusterkanals

Beispiel für Publish/Subscribe-Hierarchie: Szenario 2

Richten Sie eine Publish/Subscribe-Hierarchietopologie unter Verwendung von Punkt-zu-Punkt-Kanälen ein, wobei der Name der Übertragungswarteschlange mit dem Namen des fernen Warteschlangenmanagers identisch ist.

Informationen zu diesem Vorgang

Diese Szenarios richten eine Publish/Subscribe-Hierarchie auf unterschiedliche Weise ein, um die Verbindung zwischen Warteschlangenmanagern herzustellen. Diese Szenarien verwenden alle einen übergeordneten Warteschlangenmanager mit dem Namen QM1 und zwei untergeordnete Warteschlangenmanager mit dem Namen QM2 und QM3.

Szenario 2 ist in kleinere Abschnitte aufgeteilt, um dem Prozess einfacher folgen zu können. In diesem Szenario werden Teil 1, Teil 3 und Teil 4 von Szenario 1 aus dem Abschnitt „[Beispiel für Publish/Subscribe-Hierarchie: Szenario 1](#)“ auf Seite 90 wiederverwendet.

Szenario 2, Teil 1: Warteschlangenmanager erstellen und PSMODE festlegen

Vorgehensweise

1. Erstellen und starten Sie mit folgenden Befehlen die drei Warteschlangenmanager QM1, QM2 und QM3:

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

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

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

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

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

Szenario 2, Teil 2: Punkt-zu-Punkt-Kanalverbindungen

Informationen zu diesem Vorgang

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

Vorgehensweise

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

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

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

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

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

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

4. Starten Sie die entsprechenden Empfangsprogramme auf den Warteschlangenmanagern:

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

5. Starten Sie folgende Kanäle:

- a. Auf QM1:

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

- b. Auf QM2:

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

- c. Auf QM3:

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

6. Überprüfen Sie, ob alle Kanäle gestartet wurden:

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

Szenario 2, Teil 3: Warteschlangenmanager verbinden und ein Thema definieren

Informationen zu diesem Vorgang

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

Vorgehensweise

1. Setzen Sie auf QM2 und QM3 den übergeordneten Warteschlangenmanager auf QM1:

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

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

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

3. Definieren Sie ein Themenobjekt:

```
define topic(FOOTBALL) TOPICSTR('Sport/Soccer')
```

Szenario 2, Teil 4: Thema veröffentlichen und abonnieren

Informationen zu diesem Vorgang

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

Vorgehensweise

1. Führen Sie im ersten Befehlsfenster folgenden Befehl aus:

```
amqspub Sport/Soccer QM2
```

2. Führen Sie im zweiten Befehlsfenster folgenden Befehl aus:

```
amqssub Sport/Soccer QM1
```

3. Führen Sie im dritten Befehlsfenster folgenden Befehl aus:

```
amqssub Sport/Soccer QM3
```

Ergebnisse

Die `amqssub.exe`-Anwendungen im zweiten und dritten Befehlsfenster empfangen die Nachrichten, die im ersten Befehlsfenster veröffentlicht wurden.

Zugehörige Tasks

„[Beispiel für Publish/Subscribe-Hierarchie: Szenario 1](#)“ auf Seite 90

Richten Sie eine Publish/Subscribe-Hierarchietopologie unter Verwendung von Punkt-zu-Punkt-Kanälen mit dem Aliasnamen des Warteschlangenmanagers ein.

„[Beispiel für Publish/Subscribe-Hierarchie: Szenario 3](#)“ auf Seite 95

Hinzufügen eines Warteschlangenmanagers zu einer Hierarchietopologie unter Verwendung eines Clusterkanals

Beispiel für Publish/Subscribe-Hierarchie: Szenario 3

Hinzufügen eines Warteschlangenmanagers zu einer Hierarchietopologie unter Verwendung eines Clusterkanals

Informationen zu diesem Vorgang

Diese Szenarios richten eine Publish/Subscribe-Hierarchie auf unterschiedliche Weise ein, um die Verbindung zwischen Warteschlangenmanagern herzustellen. Diese Szenarien verwenden alle einen übergeordneten Warteschlangenmanager mit dem Namen QM1 und zwei untergeordnete Warteschlangenmanager mit dem Namen QM2 und QM3.

Szenario 3 ist in kleinere Abschnitte aufgeteilt, um dem Prozess einfacher folgen zu können. In diesem Szenario werden Teil 1, Teil 3 und Teil 4 von Szenario 1 aus dem Abschnitt „Beispiel für Publish/Subscribe-Hierarchie: Szenario 1“ auf Seite 90 wiederverwendet.

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

Szenario 2, Teil 1: Warteschlangenmanager erstellen und PSMODE festlegen

Vorgehensweise

1. Erstellen und starten Sie mit folgenden Befehlen die drei Warteschlangenmanager QM1, QM2 und QM3:

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

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

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

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

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

Szenario 2, Teil 2: Punkt-zu-Punkt-Kanalverbindungen

Informationen zu diesem Vorgang

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

Vorgehensweise

1. Setzen Sie auf QM1 und QM2 den Parameter **REPOS** auf den Namen des Clusters DEMO:

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

2. Starten Sie die entsprechenden Empfangsprogramme auf den Warteschlangenmanagern:

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

3. Definieren Sie auf jedem Warteschlangenmanager den Clusterempfängerkanal:

- a. Auf QM1:

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

- b. Auf QM2:

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

- c. Auf QM3:

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

4. Definieren Sie auf jedem Warteschlangenmanager im Cluster einen Clustersenderkanal zu einem vollständigen Repository:

a. Auf QM1:

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

b. Auf QM2:

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

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

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

Szenario 2, Teil 3: Warteschlangenmanager verbinden und ein Thema definieren

Informationen zu diesem Vorgang

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

Vorgehensweise

1. Setzen Sie auf QM2 und QM3 den übergeordneten Warteschlangenmanager auf QM1:

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

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

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

3. Definieren Sie ein Themenobjekt:

```
define topic(FOOTBALL) TOPICSTR('Sport/Soccer')
```

Szenario 2, Teil 4: Thema veröffentlichen und abonnieren

Informationen zu diesem Vorgang

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

Vorgehensweise

1. Führen Sie im ersten Befehlsfenster folgenden Befehl aus:

```
amqspub Sport/Soccer QM2
```

2. Führen Sie im zweiten Befehlsfenster folgenden Befehl aus:

```
amqssub Sport/Soccer QM1
```

3. Führen Sie im dritten Befehlsfenster folgenden Befehl aus:

```
amqssub Sport/Soccer QM3
```

Ergebnisse

Die amqssub.exe-Anwendungen im zweiten und dritten Befehlsfenster empfangen die Nachrichten, die im ersten Befehlsfenster veröffentlicht wurden.

Zugehörige Tasks

„[Beispiel für Publish/Subscribe-Hierarchie: Szenario 1](#)“ auf Seite 90

Richten Sie eine Publish/Subscribe-Hierarchietopologie unter Verwendung von Punkt-zu-Punkt-Kanälen mit dem Aliasnamen des Warteschlangenmanagers ein.

„[Beispiel für Publish/Subscribe-Hierarchie: Szenario 2](#)“ auf Seite 93

Richten Sie eine Publish/Subscribe-Hierarchietopologie unter Verwendung von Punkt-zu-Punkt-Kanälen ein, wobei der Name der Übertragungswarteschlange mit dem Namen des fernen Warteschlangenmanagers identisch ist.

Ablauf von Veröffentlichungen und Subskriptionen steuern

Warteschlangenmanager, die miteinander in eine verteilte Publish/Subscribe-Topologie miteinander verbunden sind, nutzen gemeinsam einen gemeinsamen Topic-Bereich. Sie können den Ablauf von Veröffentlichungen und Subskriptionen in der Topologien steuern, indem Sie auswählen, ob jede Veröffentlichung und Subskription lokal oder global sein soll.

Lokale Veröffentlichungen und Subskriptionen werden nicht außerhalb des Warteschlangenmanagers weitergegeben, mit dem der Bereitsteller oder Subskribent verbunden ist.

Sie können den Umfang der Themenbereiche steuern, die durch die Verbindung von Warteschlangenmanagern in Clustern oder Hierarchien erstellt werden. In einem Publish/Subscribe-Cluster muss das Themenobjekt 'clustered' sein oder alle Elemente bleiben lokal und die Veröffentlichung oder Subskription hat keine Auswirkung.

Eine Subskription, die mit Themenzeichenfolgen in verschiedenen Veröffentlichungen übereinstimmt, kann in verschiedene Themenobjekte aufgelöst werden. Diese werden als überlappende Themen bezeichnet. Das Themenobjekt, das einer Veröffentlichung für eine bestimmte Übereinstimmung zugeordnet ist, stellt die Themenattribute bereit und bestimmt beispielsweise, ob der Subskribent die Veröffentlichung empfangen soll.

Veröffentlichungsumfang

Der Bereich einer Veröffentlichung steuert, ob Warteschlangenmanager eine Veröffentlichung an ferne Warteschlangenmanager weiterleiten. Verwalten Sie den Bereich von Veröffentlichungen mit dem Themenattribut **PUBSCOPE**.

Wenn eine Veröffentlichung nicht an ferne Warteschlangenmanager weitergeleitet wird, empfangen nur lokale Subskribenten die Veröffentlichung.

Das Themenattribut **PUBSCOPE** wird verwendet, um den Umfang der Veröffentlichungen zu einem bestimmten Thema zu bestimmen. Sie können das Attribut auf einen der folgenden Werte setzen:

QMGR

Die Veröffentlichung wird nur an lokale Subskribenten übermittelt. Diese Veröffentlichungen werden als *lokale Veröffentlichungen* bezeichnet. Lokale Veröffentlichungen werden nicht an ferne Warteschlangenmanager weitergeleitet und deshalb nicht von Subskribenten empfangen, die mit fernen Warteschlangenmanagern verbunden sind.

ALL

Die Veröffentlichung wird an lokale Subskribenten und Subskribenten zugestellt, die mit fernen Warteschlangenmanagern verbunden sind. Diese Veröffentlichungen werden als *globale Veröffentlichungen* bezeichnet.

ASPARENT

Verwenden Sie die Einstellung **PUBSCOPE** des übergeordneten Elements.

Publisher können auch mit der Nachrichteneinreihungsoption MQPMO_SCOPE_QMGR angeben, ob es sich um eine lokale oder globale Veröffentlichung handelt. Wird diese Option verwendet, setzt sie das Verhalten außer Kraft, das mit dem Themenattribut **PUBSCOPE** festgelegt wurde.

Subskriptionsumfang

Der Bereich einer Subskription bestimmt, ob eine Subskription auf einem Warteschlangenmanager nur Veröffentlichungen von lokalen Publishern empfängt oder auch Veröffentlichungen, die auf einem anderen Warteschlangenmanager in einem Publish/Subscribe-Cluster oder einer Publish/Subscribe-Hierarchie erfolgen.

Eine Begrenzung des Subskriptionsbereich auf einen Warteschlangenmanager verhindert, dass Proxy-Subskriptionen an andere Warteschlangenmanager in der Publish/Subscribe-Topologie weitergeleitet werden. Dadurch wird der Publish/Subscribe-Nachrichtenübertragungsverkehr zwischen den Warteschlangenmanagern reduziert.

Das Themenattribut **SUBSCOPE** wird verwendet, um den Umfang der Subskriptionen für ein bestimmtes Thema zu ermitteln. Sie können das Attribut auf einen der folgenden Werte setzen:

QMGR

Eine Subskription empfängt nur lokale Veröffentlichungen und Proxy-Subskriptionen werden nicht an ferne Warteschlangenmanager weitergegeben.

ALL

Proxy-Subskriptionen werden an ferne Warteschlangenmanager weitergeleitet, und der Subskribent empfängt sowohl lokale als auch ferne Veröffentlichungen.

ASPARENT

Verwenden Sie die Einstellung **SUBSCOPE** des übergeordneten Elements.

Einzelne Subskribenten können die **SUBSCOPE** -Einstellung ALL überschreiben, indem sie beim Erstellen einer Subskription die Subskriptionsoption MQSO_SCOPE_QMGR angeben. Eine Subskription kann die **SUBSCOPE** -Einstellung ALL eines Themas überschreiben.

Anmerkung: Einzelne Subskribenten können nur die **SUBSCOPE** des Themas beschränken. Wenn für eine einzelne Subskription **SUBSCOPE** auf ALL gesetzt ist, berücksichtigt die Subskription die Einstellung **SUBSCOPE** der übereinstimmenden Themen.

Kombinieren von Veröffentlichungs- und Subskriptionsbereichen

In WebSphere MQ ab Version 7 arbeiten Veröffentlichungs- und Subskriptionsbereich unabhängig voneinander, um den Fluss von Veröffentlichungen zwischen Warteschlangenmanagern zu bestimmen.

Veröffentlichungen können allen Warteschlangenmanagern, die in einer Publish/Subscribe-Topologie verbunden sind, oder nur dem lokalen WS-Manager fließen. Analog zu Proxy-Subskriptionen. Welche Veröffentlichungen mit einer Subskription übereinstimmen, wird durch die Kombination dieser beiden Datenflüsse gesteuert.

Veröffentlichungen und Subskriptionen können sowohl auf QMGR als auch auf ALL definiert werden. Wenn ein Publisher und ein Subskribent beide mit demselben Warteschlangenmanager verbunden sind, wirken sich die Geltungsbereichseinstellungen nicht auf die Veröffentlichungen aus, die der Subskribent von diesem Bereitsteller erhält.

Wenn der Bereitsteller und der Subskribent mit unterschiedlichen Warteschlangenmanagern verbunden sind, müssen beide Einstellungen ALL sein, um ferne Veröffentlichungen zu empfangen.

Angenommen, Publisher sind mit verschiedenen Warteschlangenmanagern verbunden. Wenn Sie möchten, dass ein Subskribent Veröffentlichungen von einem beliebigen Publisher empfängt, legen Sie den Subskriptionsumfang auf ALL fest. Sie können dann für jeden Bereitsteller entscheiden, ob der Umfang seiner Veröffentlichungen auf Subskribenten beschränkt werden soll, die lokal für den Publisher sind.

Angenommen, die Subskribenten sind mit verschiedenen Warteschlangenmanagern verbunden. Wenn die Veröffentlichungen von einem Publisher an alle Subskribenten gesendet werden sollen, setzen Sie den Veröffentlichungsumfang auf ALL. Wenn Sie möchten, dass ein Subskribent nur Veröffentlichungen von einem Publisher empfängt, der mit demselben Warteschlangenmanager verbunden ist, legen Sie den Subskriptionsumfang auf QMGR fest.

In Version 6 und früher gilt der Veröffentlichungs- und Subskriptionsumfang nicht nur für die Veröffentlichungen, die übertragen wurden. Außerdem musste der Umfang der Veröffentlichung dem Umfang der Subskription entsprechen.

Beispiel: Fußball-Ergebnisdienst

Angenommen, Sie sind ein Mitglied-Team in einer Football-Liga. Jedes Team verfügt über einen WS-Manager, der mit allen anderen Teams in einem Publish/Subscribe-Cluster verbunden ist.

Die Teams veröffentlichen die Ergebnisse aller Spiele, die auf ihrem Heimspielplatz gespielt wurden, unter dem Thema `Football/result/Home team name/Away team name`. Die Zeichenfolgen in Kursivschrift sind variable Themennamen, und die Veröffentlichung ist das Ergebnis der Übereinstimmung.

Jeder Club veröffentlicht die Ergebnisse auch nur für den Club mit Hilfe der Themenzeichenfolge `Football/myteam/Home team name/Away team name`.

Beide Themen werden im gesamten Cluster veröffentlicht.

Die folgenden Abonnements wurden von der Liga eingerichtet, damit die Fans eines jeden Teams die Ergebnisse auf drei interessante Arten abonnieren können.

Beachten Sie, dass Sie Clusterthemen mit `SUBSCOPE(QMGR)` konfigurieren können. Die Themendefinitionen werden an jedes Member des Clusters weitergegeben, aber der Geltungsbereich der Subskription ist nur der lokale Warteschlangenmanager. So empfangen Subskribenten in jedem WS-Manager verschiedene Veröffentlichungen aus derselben Subskription.

Alle Ergebnisse empfangen

```
DEFINE TOPIC(A) TOPICSTR('Football/result/') CLUSTER SUBSCOPE(ALL)
```

Alle Ausgangsergebnisse empfangen

```
DEFINE TOPIC(B) TOPICSTR('Football/result/') CLUSTER SUBSCOPE(QMGR)
```

Da die Subskription den Geltungsbereich `QMGR` hat, werden nur die Ergebnisse verglichen, die auf dem Homerground veröffentlicht werden.

Alle eigenen Teamergebnisse empfangen

```
DEFINE TOPIC(C) TOPICSTR('Football/myteam/') CLUSTER SUBSCOPE(QMGR)
```

Da die Subskription den Geltungsbereich `QMGR` hat, werden nur die Ergebnisse des lokalen Teams, die lokal erneut veröffentlicht werden, abgeglichen.

Themenbereiche

Ein Topicbereich ist die Gruppe von Topics, die Sie subskribieren können. Ein Subskribent, der mit einem Warteschlangenmanager in einer verteilten Publish/Subscribe-Topologie verbunden ist, hat einen Topicbereich, der potenziell Themen enthält, die auf verbundenen Warteschlangenmanagern definiert sind.

Themen werden zunächst administrativ erstellt, wenn Sie ein Themenobjekt oder eine permanente Subskription definieren, oder dynamisch, wenn eine Anwendung eine Veröffentlichung oder Subskription dynamisch erstellt.

Themen werden sowohl über Proxy-Subskriptionen als auch durch die Erstellung administrativer Cluster-Topic-Objekte an andere Warteschlangenmanager weitergegeben. Proxy-Subskriptionen bewirken, dass Veröffentlichungen von dem Warteschlangenmanager, mit dem ein Publisher verbunden ist, an die Warteschlangenmanager von Subskribenten weitergeleitet werden. Proxy-Subskriptionen sind der Mechanismus, mit dem Themen, die auf verschiedenen Warteschlangenmanagern definiert sind, zu einem gemeinsamen Topicbereich kombiniert werden.

Proxy-Subskriptionen werden zwischen allen Warteschlangenmanagern weitergegeben, die durch Beziehungen zwischen übergeordneten und untergeordneten Elementen innerhalb einer Warteschlangenmanagerhierarchie miteinander verbunden sind. Daraus folgt, dass Sie auf jedem der Warteschlangenmanager ein Thema abonnieren können, das auf einem der anderen Warteschlangenmanager in der Hierarchie definiert ist. Solange es einen verbundenen Pfad zwischen den Warteschlangenmanagern gibt, spielt es keine Rolle, wie die Warteschlangenmanager verbunden sind.

Proxy-Subskriptionen werden auch für *Cluster*-Topics zwischen allen Mitgliedern eines Clusters weitergegeben. Ein Cluster-Topic ist ein Thema, das an ein Themenobjekt angehängt ist, das entweder selbst das

Attribut **CLUSTER** besitzt oder das Attribut von seinem übergeordneten Element übernimmt. Themen, die keine Cluster-Topics sind, sind als lokale Themen bekannt und werden nicht im Cluster repliziert. Subskriptionen für lokale Themen geben keine Proxy-Subskriptionen an den Cluster weiter.

Zusammenfassend gesagt, werden also in zwei Situationen Proxy-Subskriptionen für Subskribenten erstellt:

1. Ein Warteschlangenmanager ist Mitglied einer Hierarchie und eine Proxy-Subskription wird an das übergeordnete Element und an die untergeordneten Elemente des Warteschlangenmanagers weitergeleitet.
2. Ein Warteschlangenmanager ist Mitglied eines Clusters und die Subskriptionsthemenzeichenfolge wird in ein Thema aufgelöst, das einem Cluster-Topic-Objekt zugeordnet ist. Proxy-Subskriptionen werden an alle Member des Cluster weitergeleitet. Weitere Informationen zu Komplikationen finden Sie in „Überlappende Themen“ auf Seite 110 .

Wenn ein Warteschlangenmanager Mitglied eines Clusters und einer Hierarchie ist, werden Proxy-Subskriptionen durch beide Mechanismen weitergeleitet, ohne das doppelte Veröffentlichungen an den Subskribenten übergeben werden.

Die Erstellung eines Clusterthemenobjekts hat zwei Auswirkungen. Proxy-Subskriptionen für ein Thema werden an andere Mitglieder des Clusters gesendet, wenn eine Subskription in ein Clusterthema aufgelöst wird. Außerdem wird eine Kopie des Themenobjekts an die anderen Member des Clusters gesendet. Die Weiterleitung von Clusterthemenobjekten vereinfacht die Verwaltung von Themen. Normalerweise werden Clusterthemenobjekte auf einem einzelnen WS-Manager im Cluster definiert, der als Clusterthemenhost bezeichnet wird.

In der folgenden Liste werden die Themenbereiche von drei Publish/Subscribe-Topologien beschrieben:

- „Fall 1. Publish/Subscribe-Cluster“ auf Seite 101.
- „Fall 2. Publish/Subscribe-Hierarchien in Version 7“ auf Seite 103.
- „Fall 3. Publish/Subscribe-Hierarchien und -Datenströme in Version 6“ auf Seite 103.

In separaten Abschnitten beschreiben die folgenden Tasks, wie Themenbereiche kombiniert werden.

- „Einzelnen Topic-Bereich in einem Publish/Subscribe-Cluster erstellen“ auf Seite 104.
- „Warteschlangenmanager der Version 7 zu vorhandenen Topicbereichen der Version 6 hinzufügen“ auf Seite 104.
- „Topicbereiche mehrerer Cluster kombinieren“ auf Seite 105.
- „Themenbereiche in mehreren Clustern kombinieren und isolieren“ auf Seite 107
- „Veröffentlichen und Subskribieren von Topicbereichen in mehreren Clustern“ auf Seite 109

Fall 1. Publish/Subscribe-Cluster

Im Beispiel wird vorausgesetzt, dass der Warteschlangenmanager *nicht* mit einer Publish/Subscribe-Hierarchie verbunden ist.

Wenn ein Warteschlangenmanager Mitglied eines Publish/Subscribe-Clusters ist, besteht sein Themenbereich aus lokalen Themen und Cluster-Topics. Lokale Themen sind Themenobjekten ohne das Attribut **CLUSTER** zugeordnet. Wenn ein Warteschlangenmanager über lokale Themenobjektdefinitionen verfügt, unterscheidet sich sein Themenbereich von dem eines anderen Warteschlangenmanagers im Cluster, der ebenfalls über eigene, lokal definierte Themenobjekte verfügt.

In einem Publish/Subscribe-Cluster können Sie kein Thema abonnieren, das auf einem anderen Warteschlangenmanager definiert ist, außer wenn das Thema, das Sie abonnieren, in ein Cluster-Topic-Objekt aufgelöst wird.

Widersprüchliche Definitionen eines Clusterthemas, das an anderer Stelle im Cluster definiert ist, werden zugunsten der neuesten Definition aufgelöst. Wenn ein Clusterthema mehrfach definiert wurde, kann die Clusterthemendefinition auf verschiedenen Warteschlangenmanagern unterschiedlich sein.

Eine lokale Definition eines Themenobjekts, sei sie nun für ein Cluster-Topic oder ein lokales Thema, hat Vorrang vor demselben Themenobjekt, das anderswo im Cluster definiert ist. Es wird das lokal definierte Thema verwendet, auch wenn das anderswo definierte Thema jünger ist.

Setzen Sie eine der Optionen **PUBSCOPE** und **SUBSCOPE** auf QMGR, um zu verhindern, dass eine Veröffentlichung oder Subskription für ein Clusterthema an verschiedene Warteschlangenmanager im Cluster übertragen wird.

Angenommen, Sie definieren ein Clusterthemenobjekt Alabama mit der Themenzeichenfolge USA/Alabama auf Ihrem Clusterthemenhost. Das würde zu folgenden Ergebnissen führen:

1. Der Topicbereich auf dem Cluster-Topic-Host enthält jetzt das Cluster-Topic-Objekt Alabama und das Topic USA/Alabama.
2. Das Clusterthemenobjekt Alabama wird auf allen Warteschlangenmanagern im Cluster repliziert, wo es mit dem Topicbereich auf jedem Warteschlangenmanager kombiniert wird. Was auf jedem Warteschlangenmanager im Cluster geschieht, hängt davon ab, ob das Themenobjekt Alabama auf einem Warteschlangenmanager vorhanden ist.
 - Wenn Alabama ein neues Themenobjekt ist, fügt der Warteschlangenmanager das Clusterthemenobjekt Alabama und das Thema USA/Alabama zu seinem Themenbereich hinzu.
 - Wenn Alabama eine lokale Definition ist, wird das Clusterthemenobjekt Alabama hinzugefügt. Wenn die lokale Definition nicht gelöscht wird, wird das über Fernzugriff definierte Clusterthemenobjekt ignoriert. Der Warteschlangenmanager behält beide Definitionen bei.
 - Wenn Alabama ein älteres Clusterthemenobjekt ist, das an anderer Stelle definiert ist, wird es durch das neuere Clusterthemenobjekt ersetzt.
3. Eine Anwendung oder ein Administrator kann überall im Cluster eine Subskription für USA/Alabama erstellen, indem er auf das Topic-Objekt Alabama verweist.
4. Eine Anwendung an einer beliebigen Stelle im Cluster, die die Themenzeichenfolge USA/Alabama direkt verwendet, kann eine Subskription erstellen, die die Attribute des Themenobjekts übernimmt Alabama. Das Themenobjekt Alabama wird von einer Subskription übernommen, die aus einer beliebigen Themenzeichenfolge besteht, die mit USA/Alabama beginnt.

Wenn es eine weitere Definition des Alabama-Themenobjekts auf einem der anderen Warteschlangenmanager gibt, hat sie Vorrang vor der Definition auf dem Clusterthemenhost. Das lokale Objekt hat möglicherweise ein Clusterattribut oder nicht. Das Clusterattribut kann auf denselben Cluster oder einen anderen Cluster verweisen. Vermeiden Sie diese mehrfachen Definitionsfälle. Sie führen zu unterschiedlichen Verhaltensweisen.

5. Wenn das Themenobjekt Alabama das **PUBSCOPE**-Attribut ALL hat, werden Subskriptionen, die in Alabama aufgelöst werden, an alle anderen Warteschlangenmanager im Cluster gesendet.

Setzen Sie das Attribut Alabama **PUBSCOPE** auf QMGR, um zu verhindern, dass Veröffentlichungen von Publishern an Subskribenten übertragen werden, die mit verschiedenen Warteschlangenmanagern im Cluster verbunden sind.

Das Alabama-Themenobjekt wird auf jedem Warteschlangenmanager im Cluster repliziert, sodass die Attribute **PUBSCOPE** und **SUBSCOPE** für alle Warteschlangenmanager im Cluster gelten.

Es ist wichtig, dass ein Cluster-Topic-Objekt überall im Cluster derselben Themenzeichenfolge zugeordnet ist. Sie können die Themenzeichenfolge, der ein Themenobjekt zugeordnet ist, nicht ändern. Um dasselbe Themenobjekt einer anderen Themenzeichenfolge zuzuordnen, müssen Sie das Themenobjekt löschen und es mit der neuen Themenzeichenfolge erneut erstellen. Wenn das Thema zu einem Cluster gehört, bedeutet dies, dass die Kopien des Themenobjekts, die auf den anderen Mitgliedern des Clusters gespeichert sind, gelöscht und dann überall im Cluster Kopien des neuen Themenobjekts erstellt werden müssen. Die Kopien des Themenobjekts verweisen alle auf dieselbe Themenzeichenfolge.

Sie können jedoch eine doppelte Definition eines Themenobjekts auf einem anderen Warteschlangenmanager im Cluster mit einer anderen Themenzeichenfolge erstellen. Versuchen Sie immer, Duplikate zu vermeiden, indem Sie Clusterthemenhosts auf einem Warteschlangenmanager verwalten. Weitere Informationen zu diesem wichtigen Punkt finden Sie unter „Mehrere Clusterthemendefinitionen“ auf Seite

77 . Mehrere Definitionen desselben Themenobjekts mit unterschiedlichen Themenzeichenfolgen können abhängig davon, wie und wo das Thema referenziert wird, unterschiedliche Ergebnisse liefern.

Fall 2. Publish/Subscribe-Hierarchien in Version 7

Im Beispiel wird vorausgesetzt, dass der Warteschlangenmanager *nicht* Mitglied eines Publish/Subscribe-Clusters ist.

Wenn in Version 7 ein Warteschlangenmanager Mitglied einer Publish/Subscribe-Hierarchie ist, besteht sein Topicbereich aus allen Themen, die lokal und auf verbundenen Warteschlangenmanagern definiert sind. Der Themenbereich ist bei allen Warteschlangenmanagern in einer Hierarchie derselbe. Es gibt keine Aufteilung von Themen in lokale Themen und Clusterthemen.

Setzen Sie entweder die Option **PUBSCOPE** oder die Option **SUBSCOPE** auf QMGR, um zu verhindern, dass eine Veröffentlichung für ein Thema von einem Publisher an einen Subskribenten, der mit anderen Warteschlangenmanagern in der Hierarchie verbunden ist, übergeben wird.

Angenommen, Sie definieren ein Themenobjekt namens Alabama mit der Themenzeichenfolge USA/Alabama auf Warteschlangenmanager QMA. Das würde zu folgenden Ergebnissen führen:

1. Der Themenbereich auf QMA enthält jetzt auch das Themenobjekt Alabama und die Themenzeichenfolge USA/Alabama.
2. Eine Anwendung oder ein Administrator kann auf QMA eine Subskription mit dem Themenobjektnamen Alabama erstellen.
3. Eine Anwendung kann auf jedem Warteschlangenmanager in der Hierarchie eine Subskription für jedes Thema, einschließlich USA/Alabama, erstellen. Wenn QMA nicht lokal definiert wurde, wird das Thema USA/Alabama in das Themenobjekt SYSTEM.BASE.TOPIC aufgelöst.

Fall 3. Publish/Subscribe-Hierarchien und -Datenströme in Version 6

Vor Version 7 wurde der Themenbereich in separate Datenströme aufgeteilt, zu denen auch der Standarddatenstrom gehörte, der auf allen Warteschlangenmanagern vorhanden war. Veröffentlichungen können nicht zwischen verschiedenen Datenströmen übergeben werden. Bei Verwendung benannter Datenströme können die Themenbereiche auf verschiedenen Warteschlangenmanagern unterschiedlich sein. Themen werden in Themen im Standarddatenstrom und Themen in anderen benannten Datenströmen aufgeteilt.

Anmerkung: Jeder benannte Datenstrom bildet einen separaten Themenbereich. Um eine verbundene Topologie einzurichten, muss jeder benannte Datenstrom auf den verbundenen Warteschlangenmanagern vorhanden sein. Angenommen, Datenstrom X ist auf QMA und QMC definiert, aber nicht auf QMB. Wenn QMA das übergeordnete Element von QMB und QMB das übergeordnete Element von QMC ist, können keine Themen in Datenstrom X zwischen QMA und QMC übergeben werden.

Wenn die beiden Optionen **PUBSCOPE** und **SUBSCOPE** entweder auf QMGR oder auf ALL gesetzt werden, können ein Publisher und ein Subskribent eines Themas nur Veröffentlichungen zur lokalen Nutzung oder nur Veröffentlichungen zur globalen Nutzung austauschen.

Ab Version 7 sind keine Datenströme über die Publish/Subscribe-API verfügbar. Wenn Sie eingereichtes Publish/Subscribe und einen Warteschlangenmanager der Version 7 verwenden, werden Datenströme unterschiedlichen Themenobjekten zugeordnet, die die Wirkung von Datenströmen simulieren können. Ein Datenstrom wird simuliert, indem ein Themenobjekt als Stammthema für alle Themen im Datenstrom erstellt wird. Der Warteschlangenmanager ordnet Veröffentlichungen und Subskriptionen zwischen dem Datenstrom und dem zugehörigen Stammthema jeder Struktur zu.

Themenbereiche kombinieren

Kombinieren Sie den Topic-Bereich eines Warteschlangenmanagers mit anderen WS-Managern in einem Publish/Subscribe-Cluster oder einer Hierarchie. Kombinieren Sie Publish/Subscribe-Cluster und Publish/Subscribe-Cluster mit Hierarchien.

Sie können verschiedene Publish/Subscribe-Themenbereiche erstellen, indem Sie die Bausteine der **CLUSTER-**, **PUBSCOPE** -und **SUBSCOPE** -Attribute, Publish/Subscribe-Cluster und Publish/Subscribe-Hierarchien verwenden.

Ausgehend vom Beispiel der Skalierung von einem einzelnen WS-Manager zu einem Publish/Subscribe-Cluster veranschaulichen die folgenden Szenarios verschiedene Publish/Subscribe-Topologien.

Einzelnen Topic-Bereich in einem Publish/Subscribe-Cluster erstellen

Skalieren Sie ein Publish/Subscribe-System, das auf mehreren Warteschlangenmanagern ausgeführt werden soll. Verwenden Sie einen Publish/Subscribe-Cluster, um jedem Bereitsteller und Subskribenten einen einzigen identischen Topic-Bereich zur Verfügung zu stellen.

Vorbereitende Schritte

Sie haben ein Publish/Subscribe-System in einem WS-Manager der Version 7 implementiert.

Erstellen Sie immer Topic-Bereiche mit eigenen Stammthemen, anstatt sich darauf zu verlassen, dass die Attribute von SYSTEM.BASE.TOPIC übernommen werden. Wenn Sie Ihr Publish/Subscribe-System in einem Cluster skalieren, können Sie Ihre Stammthemen als Clusterthemen auf dem Clusterthemenhost definieren und anschließend alle Themen im gesamten Cluster gemeinsam nutzen.

Informationen zu diesem Vorgang

Sie möchten jetzt das System skalieren, um mehr Publisher und Subskribenten zu unterstützen, und haben alle Themen, die im gesamten Cluster sichtbar sind.

Vorgehensweise

1. Erstellen Sie einen Cluster, der mit dem Publish/Subscribe-System verwendet werden soll.
Wenn Sie einen vorhandenen traditionellen Cluster haben, ist es aus Leistungsgründen besser, einen neuen Cluster für das neue Publish/Subscribe-System einzurichten. Sie können dieselben Server für die Cluster-Repositorys beider Cluster verwenden.
2. Wählen Sie einen Warteschlangenmanager, möglicherweise eines der Repositorys, als Cluster-Topic-Host aus.
3. Stellen Sie sicher, dass alle Themen, die im gesamten Publish/Subscribe-Cluster sichtbar sein sollen, in ein Verwaltungsthemenobjekt aufgelöst werden.
Legen Sie für das Attribut **CLUSTER** den Namen des Publish/Subscribe-Clusters fest.

Nächste Schritte

Verbinden Sie Publisher- und Subskribentenanwendungen mit allen WS-Managern im Cluster.

Erstellen Sie Verwaltungsthemenobjekte, für die das Attribut **CLUSTER** vorhanden ist. Die Themen werden auch im gesamten Cluster weitergegeben. Publisher- und Subskribentenprogramme verwenden die Verwaltungsthemen, so dass ihr Verhalten nicht durch die Verbindung zu verschiedenen Warteschlangenmanagern im Cluster geändert wird.

Wenn Sie SYSTEM.BASE.TOPIC benötigen, um wie ein Clusterthema auf jedem WS-Manager zu agieren, müssen Sie ihn auf jedem WS-Manager ändern.

Warteschlangenmanager der Version 7 zu vorhandenen Topicbereichen der Version 6 hinzufügen

Erweitern Sie ein vorhandenes Publish/Subscribe-System der Version 6, um mit einem Warteschlangenmanager der Version 7 zu interagieren und dieselben Topicbereiche gemeinsam zu nutzen.

Vorbereitende Schritte

Es ist ein Publish/Subscribe-System der Version 6 vorhanden.

Sie haben WebSphere MQ Version 7 auf einem neuen Server installiert und einen Warteschlangenmanager konfiguriert.

Informationen zu diesem Vorgang

Sie möchten Ihr vorhandenes Publish-/Subscribe-System der Version 6 für die Arbeit mit Warteschlangenmanagern der Version 7 erweitern.

Sie haben entschieden, die Entwicklung des Publish/Subscribe-Systems der Version 6 zu stabilisieren, das die Publish/Subscribe-Schnittstelle in der Warteschlange verwendet. Sie möchten dem System Erweiterungen über die MQI der Version 7 hinzufügen. Sie haben jetzt keine Pläne zum Umschreiben der in die Warteschlange gestellten Publish/Subscribe-Anwendungen.

Sie beabsichtigen, die Warteschlangenmanager der Version 6 in Zukunft auf Version 7 zu aktualisieren. Momentan werden die vorhandenen Publish/Subscribe-Anwendungen in der Warteschlange weiterhin auf den Warteschlangenmanagern der Version 7 ausgeführt.

Vorgehensweise

1. Erstellen Sie eine Gruppe von Sender-Empfänger-Kanälen, um den Warteschlangenmanager der Version 7 mit einem der Warteschlangenmanager der Version 6 in beide Richtungen zu verbinden.
2. Erstellen Sie zwei Übertragungswarteschlangen mit den Namen der Ziel-WS-Manager. Verwenden Sie WS-Manager-Aliasnamen, wenn Sie aus irgendeinem Grund den Namen des Zielwarteschlangenmanagers nicht als Name der Übertragungswarteschlange verwenden können.
3. Konfigurieren Sie die Übertragungswarteschlangen, um die Senderkanäle auszulösen.
4. Wenn das Publish/Subscribe-System der Version 6 Datenströme verwendet, fügen Sie die Datenströme dem Warteschlangenmanager der Version 7 hinzu, wie unter [Datenstrom hinzufügen](#) beschrieben.
5. Überprüfen Sie, ob der Warteschlangenmanager **PSMODE** der Version 7 auf **ENABLE** gesetzt ist.
6. Ändern Sie das Attribut **PARENT** so, dass es auf einen der Warteschlangenmanager der Version 6 verweist.
7. Überprüfen Sie den Status der Elternbeziehung zwischen den Warteschlangenmanagern in beide Richtungen.

Nächste Schritte

Nachdem Sie die Aufgabe abgeschlossen haben, nutzen sowohl der Warteschlangenmanager der Version 6 als auch der Warteschlangenmanager der Version 7 dieselben Topic-Bereiche. Sie können zum Beispiel alle folgenden Tasks ausführen.

- Tauschen Sie Veröffentlichungen und Subskriptionen zwischen Warteschlangenmanagern der Versionen 6 und 7 aus.
- Führen Sie Ihre vorhandenen Publish/Subscribe-Programme der Version 6 auf dem Warteschlangenmanager der Version 7 aus.
- Zeigen Sie den Topic-Bereich auf dem Warteschlangenmanager der Version 6 oder Version 7 an und ändern Sie ihn.
- Schreiben Sie Publish/Subscribe-Anwendungen der Version 7 und führen Sie sie auf dem Warteschlangenmanager der Version 7 aus.
- Erstellen Sie neue Veröffentlichungen und Subskriptionen mit den Anwendungen der Version 7 und tauschen Sie sie mit Anwendungen der Version 6 aus.

Topicbereiche mehrerer Cluster kombinieren

Erstellen Sie Topic-Bereiche, die sich über mehrere Cluster erstrecken. Publizieren Sie zu einem Thema in einem Cluster und abonnieren Sie es in einem anderen Cluster.

Vorbereitende Schritte

Sie haben vorhandene Publish/Subscribe-Cluster und möchten einige Clusterthemen an alle Cluster weitergeben.

Informationen zu diesem Vorgang

Um Veröffentlichungen von einem Cluster an einen anderen weiterzugeben, müssen Sie die Cluster in einer Hierarchie miteinander verknüpfen (siehe [Abbildung 35](#) auf Seite 106). Die hierarchischen Verbindungen geben Subskriptionen und Veröffentlichungen zwischen den verbundenen Warteschlangenmanagern weiter und die Cluster propagieren Clusterthemen in jedem Cluster, jedoch nicht zwischen Clustern.

Die Kombination dieser beiden Mechanismen propagiert Clusterthemen zwischen allen Clustern. Sie müssen die Clusterthemendefinitionen in jedem Cluster wiederholen.

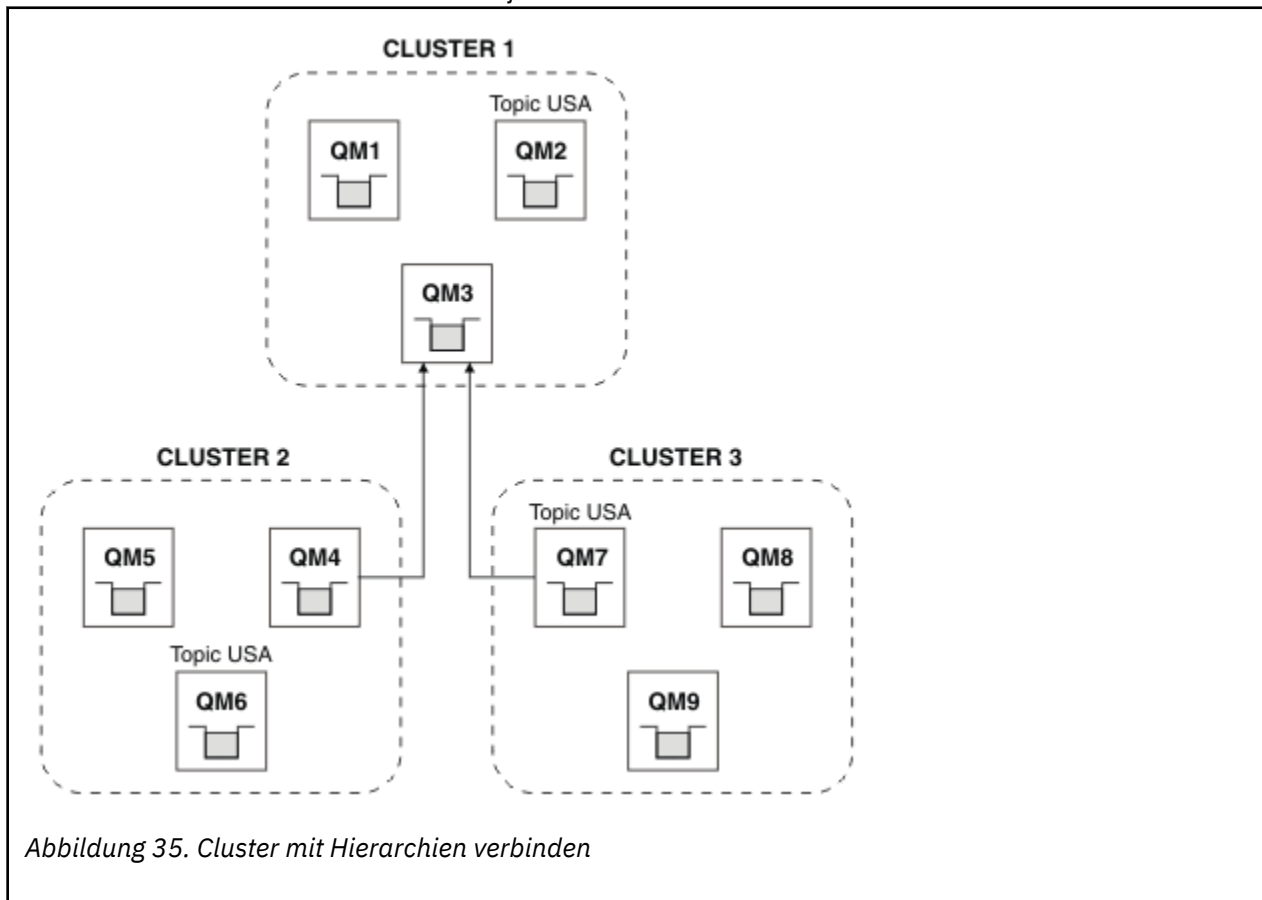


Abbildung 35. Cluster mit Hierarchien verbinden

Mit den folgenden Schritten können Sie die Cluster in einer Hierarchie miteinander verbinden.

Vorgehensweise

1. Erstellen Sie zwei Gruppen von Sender-Empfänger-Kanälen, um QM3 und QM4 sowie QM3 und QM7 in beide Richtungen zu verbinden. Sie müssen die traditionellen Sender-Empfänger-Kanäle und -Übertragungswarteschlangen anstelle eines Clusters verwenden, um eine Hierarchie zu verbinden.
2. Erstellen Sie drei Übertragungswarteschlangen mit den Namen der Ziel-WS-Manager. Verwenden Sie WS-Manager-Aliasnamen, wenn Sie aus irgendeinem Grund den Namen des Zielwarteschlangenmanagers nicht als Name der Übertragungswarteschlange verwenden können.
3. Konfigurieren Sie die Übertragungswarteschlangen, um die Senderkanäle auszulösen.
4. Stellen Sie sicher, dass **PSMODE** von QM3, QM4 und QM7 auf **ENABLE** gesetzt ist.
5. Ändern Sie das **PARENT**-Attribut von QM4 und QM7 in QM3.
6. Überprüfen Sie den Status der Elternbeziehung zwischen den Warteschlangenmanagern in beide Richtungen.
7. Erstellen Sie das Verwaltungsthema USA mit dem Attribut **CLUSTER('CLUSTER 1')**, **CLUSTER('CLUSTER 2')** und **CLUSTER('CLUSTER 3')** auf jedem der drei Clusterthemenhosts in den Clustern 1, 2 und 3. Der Cluster-Topic-Host muss kein hierarchisch verbundener Warteschlangenmanager sein.

Nächste Schritte

Sie können jetzt das Clusterthema USA in [Abbildung 35 auf Seite 106](#) veröffentlichen oder abonnieren. Der Subskriptionsablauf der Veröffentlichungen wird in allen drei Clustern an Publisher und Subskribenten fließen.

Angenommen, Sie haben USA nicht als Clusterthema in den anderen Clustern erstellt. Wenn USA nur in QM7 definiert ist, werden Veröffentlichungen und Subskriptionen für USA zwischen QM7, QM8, QM9 und QM3 ausgetauscht. Publizierer und Subskribenten, die unter QM7, QM8, QM9 ausgeführt werden, übernehmen die Attribute des Verwaltungsthemas USA. Veröffentlichenden und Subskribenten in QM3 übernehmen die Attribute von SYSTEM . BASE . TOPIC auf QM3.

Themenbereiche in mehreren Clustern kombinieren und isolieren

Isolieren Sie einige Topic-Bereiche in einem bestimmten Cluster, und kombinieren Sie andere Topic-Bereiche, um sie in allen verbundenen Clustern zugänglich zu machen.

Vorbereitende Schritte

Untersuchen Sie das Thema „[Topicbereiche mehrerer Cluster kombinieren](#)“ auf Seite 105. Es kann für Ihre Anforderungen ausreichend sein, ohne einen zusätzlichen WS-Manager als Bridge hinzuzufügen.

Informationen zu diesem Vorgang

Eine mögliche Verbesserung bezüglich der unter [Abbildung 35 auf Seite 106](#) im Abschnitt „[Topicbereiche mehrerer Cluster kombinieren](#)“ auf Seite 105 dargestellten Topologie besteht in der Eingrenzung von Clusterthemen, die nicht von allen Clustern gemeinsam genutzt werden. Isolieren Sie Cluster, indem Sie einen Überbrückungswarteschlangenmanager erstellen, der sich nicht in einem der Cluster befindet (siehe [Abbildung 36 auf Seite 108](#)). Verwenden Sie den Brückenwarteschlangenmanager, um zu filtern, welche Veröffentlichungen und Subskriptionen von einem Cluster in einen anderen fließen können.

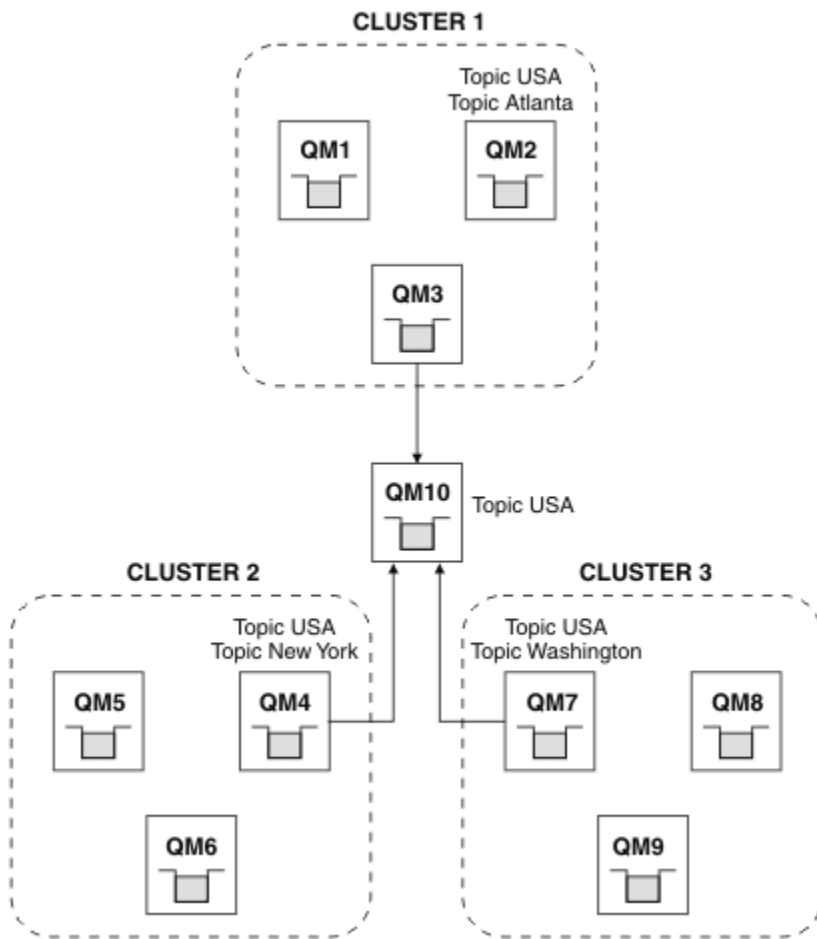


Abbildung 36. Überbrückte Cluster

Verwenden Sie die Brücke, um Clusterthemen zu isolieren, die nicht über die Brücke auf den anderen Clustern zugänglich gemacht werden sollen. In [Abbildung 36](#) auf Seite 108 ist USA ein Clusterthema, das in allen Clustern gemeinsam genutzt wird, und Atlanta, New York und Washington sind Clusterthemen, die jeweils nur in einem Cluster gemeinsam genutzt werden.

Model-Ihre Konfiguration mit der folgenden Prozedur:

Vorgehensweise

1. Ändern Sie alle SYSTEM.BASE.TOPIC-Themenobjekte so, dass **SUBSCOPE(QMGR)** und **PUBSCOPE(QMGR)** auf allen Warteschlangenmanagern vorhanden sind.
Es werden keine Themen (auch Clusterthemen) an andere Warteschlangenmanager weitergegeben, es sei denn, Sie haben explizit **SUBSCOPE(ALL)** und **PUBSCOPE(ALL)** im Stammthema Ihrer Clusterthemen festgelegt.
2. Definieren Sie die Themen auf den drei Clusterthemenhosts, die in jedem Cluster gemeinsam genutzt werden sollen, mit den Attributen **CLUSTER(Clustername)** **SUBSCOPE(ALL)** und **PUBSCOPE(ALL)**.
Wenn einige Clusterthemen von allen Clustern gemeinsam genutzt werden sollen, definieren Sie in jedem der Cluster dasselbe Thema. Verwenden Sie den Clusternamen jedes Cluster als Clusterattribut.
3. Definieren Sie für die Clusterthemen, die von allen Clustern gemeinsam genutzt werden sollen, die Themen erneut auf dem Bridge-Warteschlangenmanager (QM10) mit den Attributen **SUBSCOPE(ALL)** und **PUBSCOPE(ALL)**.

Beispiel

In dem Beispiel in [Abbildung 36 auf Seite 108](#) werden nur Themen, die von USA übernommen werden, zwischen allen drei Clustern weitergegeben.

Nächste Schritte

Subskriptionen für Themen, die auf dem Brückenwarteschlangenmanager mit **SUBSCOPE(ALL)** und **PUBSCOPE(ALL)** definiert sind, werden zwischen den Clustern weitergegeben.

Subskriptionen für Themen, die in jedem Cluster mit den Attributen **CLUSTER(clustername)**, **SUBSCOPE(ALL)** und **PUBSCOPE(ALL)** definiert sind, werden in jedem Cluster weitergegeben.

Alle anderen Subskriptionen sind lokal für einen WS-Manager.

Veröffentlichen und Subskribieren von Topicbereichen in mehreren Clustern

Sie können Themen in mehreren Clustern mit überlappenden Clustern veröffentlichen und subskribieren. Sie können diese Technik verwenden, solange sich die Topic-Bereiche in den Clustern nicht überschneiden.

Vorbereitende Schritte

Erstellen Sie mehrere traditionelle Cluster mit einigen Warteschlangenmanagern in den Schnittbereichen zwischen den Clustern.

Informationen zu diesem Vorgang

Möglicherweise haben Sie die Überlappung von Clustern aus verschiedenen Gründen ausgewählt.

1. Sie verfügen über eine begrenzte Anzahl von Hochverfügbarkeitsservern oder Warteschlangenmanagern. Sie entscheiden, alle Cluster-Repositorys zu implementieren, und Clusterthemenhosts zu ihnen.
2. Sie verfügen über traditionelle WS-Manager-Cluster, die mit Gateway-WS-Managern verbunden sind. Sie möchten Publish/Subscribe-Anwendungen in derselben Clustertopologie implementieren.
3. Sie haben eine Reihe von eigenständigen Publish/Subscribe-Anwendungen. Aus Leistungsgründen ist es besser, Publish/Subscribe-Cluster klein zu halten und sich von herkömmlichen Clustern zu trennen. Sie haben sich entschieden, die Anwendungen in verschiedenen Clustern zu implementieren. Sie möchten jedoch auch alle Publish/Subscribe-Anwendungen in einem WS-Manager überwachen, da Sie nur eine Kopie der Überwachungsanwendung lizenziert haben. Dieser WS-Manager muss über Zugriff auf die Veröffentlichungen zu Clusterthemen in allen Clustern verfügen.

Wenn Sie dafür sorgen, dass Ihre Topics in nicht überlappenden Topic-Bereichen definiert sind, können Sie die Topics in überlappenden Publish/Subscribe-Clustern implementieren; weitere Informationen finden Sie im Abschnitt [Abbildung 37 auf Seite 110](#). Wenn sich die Topic-Bereiche überschneiden, führt die Implementierung in überlappenden Clustern zu Problemen.

Da die Publish/Subscribe-Cluster überlappen, können Sie alle Topic-Bereiche mit den Warteschlangenmanagern in der Überlappung veröffentlichen und subskribieren.

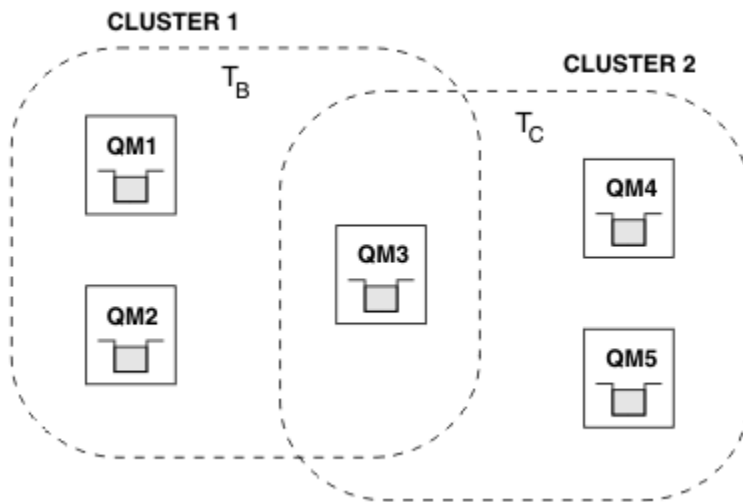


Abbildung 37. Überschneidung von Clustern, nicht überlappenden Themenbereichen

Vorgehensweise

Erstellen Sie eine Möglichkeit, um sicherzustellen, dass die Topic-Bereiche sich nicht überschneiden. Definieren Sie beispielsweise für jeden der Themenbereiche ein eindeutiges Stammthema. Machen Sie die Themen in den Stammthemen zu Themen.

- a) DEFINE TOPIC(B) TOPICSTR('B') CLUSTER('CLUSTER 1') ...
- b) DEFINE TOPIC(C) TOPICSTR('C') CLUSTER('CLUSTER 2') ...

Beispiel

In [Abbildung 37](#) auf Seite 110 können Publisher und Subskribenten, die mit QM3 verbunden sind, T_B oder T_C veröffentlichen oder abonnieren.

Nächste Schritte

Verbinden Sie Publisher und Subskribenten, die Themen in beiden Clustern zu Warteschlangenmanagern in der Überlappung verwenden.

Verbinden Sie Publisher und Subskribenten, die nur Topics in einem bestimmten Cluster zu Warteschlangenmanagern verwenden dürfen, die sich nicht in der Überlappung enthalten.

Überlappende Themen

Überlappende Themen treten auf, wenn eine Veröffentlichung verschiedenen Themenobjekten zugeordnet werden kann, abhängig von der verteilten Publish/Subscribe-Topologie, der Veröffentlichung und den Themenzeichenfolgen der Subskription.

Überlappungen zwischen Themen müssen berücksichtigt werden, wenn ein Thema in mehrere Themenobjekte aufgelöst werden kann.

Lokale Themen in einem Cluster

Ein Thema kann auf jedem WS-Manager in einem Cluster definiert sein. Wenn das Thema lokal definiert ist, hat es Vorrang vor einem Clusterthema, das an anderer Stelle definiert ist und in dieselbe Themenzeichenfolge aufgelöst wird.

Clusterthemen in einem Cluster

Ein Thema kann auf jedem WS-Manager in einem Cluster definiert sein. Wenn das Thema in Gruppen zusammengefasst ist, wird es auf andere Member des Clusters repliziert. Wenn das Thema als Clusterthema auf einem anderen Warteschlangenmanager im Cluster definiert ist, ist dies ein Fehler. Eine Fehlermeldung wird in das Fehlerprotokoll des Warteschlangenmanagers geschrieben, der über eine vorhandene Clusterdefinition verfügt.

Definieren Sie in der Regel Clusterthemen nur auf einem Warteschlangenmanager im Cluster, dem "Clusterthemenhost", um sicherzustellen, dass nur eine Definition eines Clusterthemas vorhanden ist.

Wenn Sie ein Clusterthema neu definieren, nimmt die Änderung Zeit in Anspruch, um jeden Warteschlangenmanager zu erreichen. Schließlich überschreibt die neueste Definition die früheren Clusterthemendefinitionen, die auf Nicht-Cluster-Themenhosts repliziert wurden.

Wenn Sie ein Clusterthema auf mehreren Warteschlangenmanagern im Cluster mit unterschiedlichen Attributen definieren, überschreibt die neueste Definition keine früheren lokalen Definitionen.

Platzhaltersubskriptionen werden in mehrere Themenzeichenfolgen aufgelöst

Wenn eine Subskription Platzhalterzeichen enthält, können potenziell unterschiedliche Themen in einem Topicbereich mit der Subskription übereinstimmen und dazu führen, dass die Subskription in verschiedene Topicobjekte aufgelöst wird.

Betrachten Sie zum Beispiel die folgenden Themendefinitionen im Cluster SPORTS.

```
DEFINE TOPIC(A) TOPICSTR('Football/result/#') SUBSCOPE(QMGR) CLUSTER(SPORTS)
DEFINE TOPIC(B) TOPICSTR('Football/#') SUBSCOPE(ALL) CLUSTER(SPORTS)
DEFINE TOPIC(C) TOPICSTR('Football/result/Newport/Cardiff') PUBSCOPE(ALL) SUBSCOPE(ALL)
CLUSTER(SPORTS)
DEFINE TOPIC(D) TOPICSTR('Football/matches/Newport/Cardiff') PUBSCOPE(ALL) SUBSCOPE(QMGR)
CLUSTER(SPORTS)
```

Angenommen, es gibt zwei Warteschlangenmanager QM1 und QM2 im Cluster. Die Themen C und D werden in QM1 veröffentlicht.

Überlegen Sie, was ein Subskribent unter QM2 empfängt, wenn diese Subskriptionen nicht gruppiert sind.

- Eine Subskription für das Thema A erhält nichts.
 - SUBSCOPE (QMGR) und die Veröffentlichung befindet sich auf dem anderen Warteschlangenmanager.
- Eine Subskription für Thema B empfängt beide Veröffentlichungen.
 - SUBSCOPE (ALL) und PUBSCOPE (ALL) in beiden Fällen.
- Eine Subskription für das Thema C empfängt eine Veröffentlichung.
 - SUBSCOPE (ALL) und PUBSCOPE (ALL) und eine Übereinstimmung mit der Veröffentlichung zum Thema C.
- Eine Subskription für das Thema D erhält nichts.
 - SUBSCOPE (QMGR) und die Veröffentlichung befindet sich auf dem anderen Warteschlangenmanager.

Überlegen Sie, was ein Subskribent unter QM2 empfängt, wenn diese Subskriptionen gruppiert sind.

- Der Subskribent empfängt eine Veröffentlichung zum Thema C.
 - Die übereinstimmende Subskription für Thema A mit SUBSCOPE (QMGR) wird durch die übereinstimmende Subskription für Thema C mit SUBSCOPE (ALL) überschrieben. Die spezifischere Subskription gewinnt und die Veröffentlichung wird empfangen.
 - Die übereinstimmende Subskription für Thema B wird zugunsten der übereinstimmenden Subskription für Thema C zurückgewiesen, da die Subskriptionen gruppiert sind und C spezifischer ist. Die doppelte Veröffentlichung wird gelöscht.
- Der Subskribent erhält keine Veröffentlichung zum Thema D
 - Die übereinstimmende Subskription für Thema B mit SUBSCOPE (ALL) wird durch die übereinstimmende Subskription für Thema D mit SUBSCOPE (QMGR) überschrieben. Die spezifischere Subskription gewinnt und die Veröffentlichung wird verworfen.

Funktionsweise der Schleifenermittlung

In einem verteilten Publish/Subscribe-Netz ist es wichtig, dass Veröffentlichungen und Proxy-Subskriptionen keine Schleife bilden können, da dies dazu führen würde, dass ein Netz mit verbundenen Subskribenten mehrere Kopien derselben ursprünglichen Veröffentlichung empfängt.

Das in „Aggregation von Proxy-Subskriptionen und Aggregation von Veröffentlichungen“ auf Seite 56 beschriebene Aggregationssystem für Proxy-Subskriptionen verhindert nicht die Bildung einer Schleife, obwohl es die zeitlich unbegrenzte Schleife von Proxy-Subskriptionen verhindert. Da die Weitergabe von Veröffentlichungen von den vorhandenen Proxy-Subskriptionen abhängt, können sie in eine zeitlich unbegrenzte Schleife eingereiht werden. In WebSphere MQ V7.0 wird das folgende Verfahren verwendet, um zu verhindern, dass Veröffentlichungen in zeitlich unbegrenzte Schleifen eingereiht werden:

Da die Veröffentlichungen in einer Publish/Subscribe-Topologie zirkulieren, fügt jeder Warteschlangenmanager dem Nachrichtenheader einen eindeutigen elektronischen Fingerabdruck hinzu. Bei jeder von einem anderen Publish/Subscribe-Warteschlangenmanager empfangenen Veröffentlichung überprüft der Publish/Subscribe-Warteschlangenmanager den elektronischen Fingerabdruck im Nachrichtenheader. Falls der Publish/Subscribe-Warteschlangenmanager feststellt, dass sein eigener elektronischer Fingerabdruck bereits vorhanden ist, hat die Veröffentlichung die gesamte Schleife bereits einmal durchlaufen. In diesem Fall löscht der Warteschlangenmanager die Nachricht und fügt dem Fehlerprotokoll einen Eintrag hinzu.

Anmerkung: In einer Schleife werden Veröffentlichungen in beide Richtungen weitergegeben, sodass jeder Warteschlangenmanager in der Schleife beide Veröffentlichungen empfängt, bevor der Warteschlangenmanager, von dem die Veröffentlichungen ausgingen, diese löscht. Dies hat zur Folge, dass abonnie-rende Anwendungen eine Duplikatkopie der Veröffentlichung empfangen, bevor die Schleife unterbrochen wird.

Format des elektronischen Fingerabdrucks für die Schleifenermittlung

Die Fingerabdrücke für die Schleifenermittlung werden als Teil des V7.0 -Protokolls in einen RFH2 -Header oder -Datenfluss eingefügt. Es ist wichtig, dass RFH2-Programmierer den Header verstehen und die Informationen für den elektronischen Fingerabdruck intakt weitergeben. WebSphere MessageBroker verwendet RFH1 -Header, die keine Fingerabdruckinformationen enthalten.

```
<ibm>
  <Rfp>uuid1</Rfp>
  <Rfp>uuid2</Rfp>
  <Rfp>uuid3</Rfp>
</ibm>
```

<ibm> ist der Name des Ordners, in dem die Liste der Routing-Fingerabdrücke gespeichert ist. Die Fingerabdrücke enthalten die eindeutige Benutzer-ID (UUID) jedes einzelnen Warteschlangenmanagers, der beim Routing durchlaufen wurde.

Bei jeder Veröffentlichung einer Nachricht durch einen Warteschlangenmanager fügt er seine UUID mithilfe des Tags <Rfp> (Routing-Fingerabdruck) im Ordner <ibm> hinzu. Wenn eine Veröffentlichung empfangen wird, verwendet WebSphere MQ die API für Nachrichteneigenschaften, um durch die < Rfp > -Tags zu iterieren, um festzustellen, ob dieser bestimmte UUID-Wert vorhanden ist. Aufgrund der Art und Weise, wie die WebSphere Platform Messaging-Komponente von WebSphere MQ eine Verbindung zu WebSphere Message Broker über einen Kanal und eine RFH2 -Subskription herstellt, wenn die eingereihte Publish/Subscribe-Schnittstelle verwendet wird, erstellt WebSphere MQ auch einen Fingerabdruck, wenn eine Veröffentlichung über diese Route empfangen wird.

Ziel ist, die Zustellung einer nicht erwarteten RFH2 an eine Anwendung zu vermeiden, was durch das Hinzufügen des elektronischen Fingerabdrucks geschehen könnte.

Wenn ein RFH2 -Header in Nachrichteneigenschaften konvertiert wird, muss auch der Ordner < ibm > konvertiert werden. Dadurch werden die Fingerabdruckinformationen aus dem RFH2 -Header entfernt, der an Anwendungen übergeben oder zugestellt wird, die die WebSphere MQ V7.0 verwendet haben.

Wenn eine Nachricht, die über Fingerabdruckinformationen verfügt, an einen RFH1 -Subskribenten übermittelt oder an Websphere Message Broker V6.0 übergeben wird, werden die Fingerabdruckinformationen in RFH1 konvertiert.

Wenn WebSphere Message Broker V6.0 diese Nachricht an einen RFH2 -Subskribenten (z. B. SIB) übergibt, müssen die Fingerabdruckinformationen zurück in ein RFH2 -Format konvertiert werden.

JMS-Anwendungen sehen die Fingerabdruckinformationen nicht, weil die JMS-Schnittstelle diese Informationen nicht aus RFH2extrahiert und sie daher nicht an ihre Anwendungen übergibt.

Die Rfp-Nachrichteneigenschaften werden mit `propDesc.CopyOptions = MQCOPY_FORWARD` and `MQCOPY_PUBLISH` erstellt. Dies hat Auswirkungen auf Anwendungen, die die Nachricht empfangen und sie erneut veröffentlichen. Dies bedeutet, dass eine solche Anwendung die Kette der Routing-Fingerabdrücke mit `PutMsgOpts.Action = MQACTP_FORWARD` fortsetzen kann, jedoch entsprechend codiert werden muss, um ihren eigenen Fingerabdruck aus der Kette zu entfernen. Standardmäßig verwendet die Anwendung `PutMsgOpts.Action = MQACTP_NEW` und startet eine neue Kette.

Ständige Veröffentlichungen in einer verteilten Publish/Subscribe-Topologie

Bei Verwendung von ständigen Veröffentlichungen in einer verteilten Publish/Subscribe-Topologie ist es ein bewährtes Verfahren, nur ständige Veröffentlichungen zu demselben Thema von einem einzelnen Warteschlangenmanager in der Topologie zu veröffentlichen.

Andernfalls ist es möglich, dass verschiedene ständige Veröffentlichungen auf verschiedenen Warteschlangenmanagern für dasselbe Topic aktiv sind, was zu einem nicht erwarteten Verhalten führt. Da mehrere Proxy-Subskriptionen verteilt werden, können mehrere ständige Veröffentlichungen empfangen werden.

Publish/Subscribe-Sicherheit zwischen Warteschlangenmanagern

Publish/Subscribe-interne Nachrichten, wie z. B. Proxy-Subskriptionen und Veröffentlichungen, werden mithilfe der normalen Kanalsicherheitsregeln in Warteschlangen für Publish/Subscribe-Systeme gestellt. In den Informationen und Diagrammen in diesem Thema werden die verschiedenen Prozesse und Benutzer-IDs hervorgehoben, die an der Zustellung dieser Nachrichten beteiligt sind.

Lokale Zugriffssteuerung

Der Zugriff auf Themen für Veröffentlichungen und Subskriptionen richtet sich nach lokalen Sicherheitsdefinitionen und -regeln, die in der [Publish/Subscribe-Sicherheit](#) beschrieben sind. Unter z/OS ist kein lokales Themenobjekt erforderlich, um die Zugriffssteuerung einzurichten. Für die Zugriffssteuerung auf anderen Plattformen ist kein lokales Thema erforderlich. Administratoren können die Zugriffssteuerung auf Clusterthemenobjekte anwenden, unabhängig davon, ob sie noch im Cluster vorhanden sind.

Systemadministratoren sind für die Zugriffssteuerung auf ihrem lokalen System verantwortlich. Sie müssen den Administratoren anderer Mitglieder der Hierarchie oder Cluster-Brokerverbänden vertrauen, die für ihre Zugriffssteuerungsrichtlinie verantwortlich sind. Da die Zugriffssteuerung für jede einzelne Maschine definiert ist, ist es wahrscheinlich, dass sie belastet wird, wenn eine Feinsteuerungskontrolle erforderlich ist. Es kann nicht erforderlich sein, eine Zugriffssteuerung zu erzwingen, oder die Zugriffssteuerung kann auf übergeordneten Objekten in der Themenstruktur definiert werden. Die Zugriffssteuerung auf Feinebene kann für jede Unterteilung des Topic-Namespaces definiert werden.

Proxy-Subskription erstellen

Das Vertrauen für eine Organisation, die ihren Warteschlangenmanager mit Ihrem Warteschlangenmanager verbindet, wird durch die normalen Kanalauthentifizierungsmittel bestätigt. Wenn diese vertrauenswürdige Organisation auch verteilte Publish/Subscribe-Verfahren ausführen darf, wird eine Berechtigungs-Prüfung durchgeführt. Die Prüfung wird durchgeführt, wenn der Kanal eine Nachricht in eine verteilte Publish/Subscribe-Warteschlange einreicht. Beispiel: Eine Nachricht wird in die Warteschlange `SYSTEM.INTER.QMGR.CONTROL` gestellt. Die Benutzer-ID für die Warteschlangenberechtigungsüberprüfung hängt von den `PUTAUT` -Werten des empfangenden Kanals ab. Beispiel: Die Benutzer-ID des Kanals `MCAUSER`, der Nachrichtenkontext, abhängig von dem Wert und der Plattform. Weitere Informationen zur Kanalsicherheit finden Sie unter [Kanalsicherheit](#).

Proxy-Subskriptionen werden mit der Benutzer-ID des verteilten Publish/Subscribe-Agenten auf dem fernen WS-Manager erstellt. Beispiel: `QM2` in [Abbildung 38](#) auf Seite 114. Der Benutzer erhält dann

problemlos Zugriff auf lokale Themenobjektprofile, da diese Benutzer-ID im System definiert ist und es daher keine Domänenkonflikte gibt.

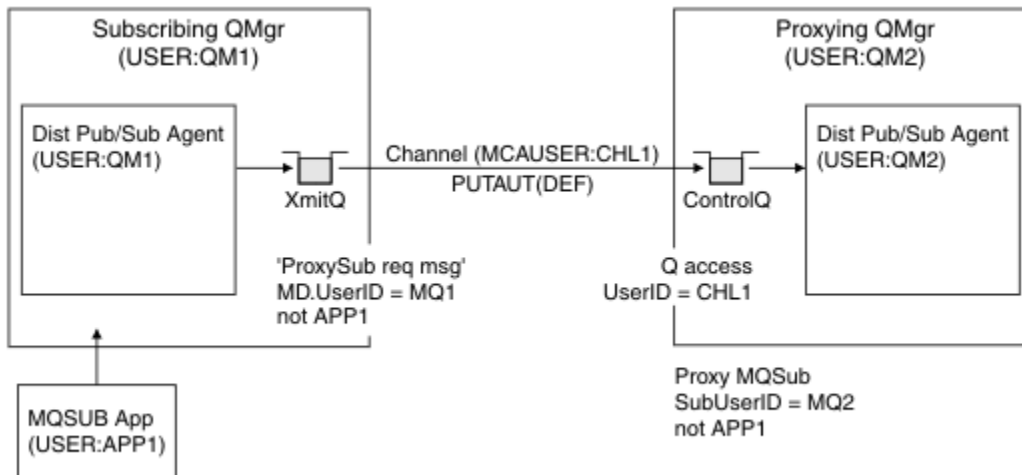


Abbildung 38. Proxy-Abonnementsicherheit, Subskription vornehmen

Zurücksenden von fernen Veröffentlichungen

Wenn eine Veröffentlichung auf dem Veröffentlichungswarteschlangenmanager erstellt wird, wird eine Kopie der Veröffentlichung für jede Proxy-Subskription erstellt. Der Kontext der kopierten Veröffentlichung enthält den Kontext der Benutzer-ID, die die Subskription erstellt hat; QM2 in [Abbildung 39](#) auf Seite 115. Die Proxy-Subskription wird mit einer Zielwarteschlange erstellt, bei der es sich um eine ferne Warteschlange handelt, so dass die Veröffentlichungsnachricht in eine Übertragungswarteschlange aufgelöst wird.

Das Vertrauen für eine Organisation, die ihren Warteschlangenmanager (QM2) mit einem anderen Warteschlangenmanager (QM1) verbinden soll, wird durch die normalen Kanalauthentifizierungsmittel bestätigt. Wenn diese vertrauenswürdige Organisation dann eine verteilte Publish/Subscribe-Datei ausführen darf, wird eine Berechtigungsprüfung durchgeführt, wenn der Kanal die Veröffentlichungsnachricht in die verteilte Publish/Subscribe-Veröffentlichungswarteschlange SYSTEM.INTER.QMGR.PUBS einreicht. Die Benutzer-ID für die Warteschlangenberechtigungsüberprüfung hängt vom Wert PUTAUT des empfangenden Kanals ab (z. B. die Benutzer-ID des Kanals, MCAUSER, Nachrichtenkontext und andere, abhängig von Wert und Plattform). Weitere Informationen zur Kanalsicherheit finden Sie unter [Kanalsicherheit](#).

Wenn die Veröffentlichungsnachricht den Subskribentenwarteschlangenmanager erreicht, wird unter der Berechtigung dieses Warteschlangenmanagers ein weiterer MQPUT-Aufruf ausgeführt, und der Kontext mit der Nachricht wird durch den Kontext jedes lokalen Subskribenten ersetzt, da sie jeweils die Nachricht erhalten.

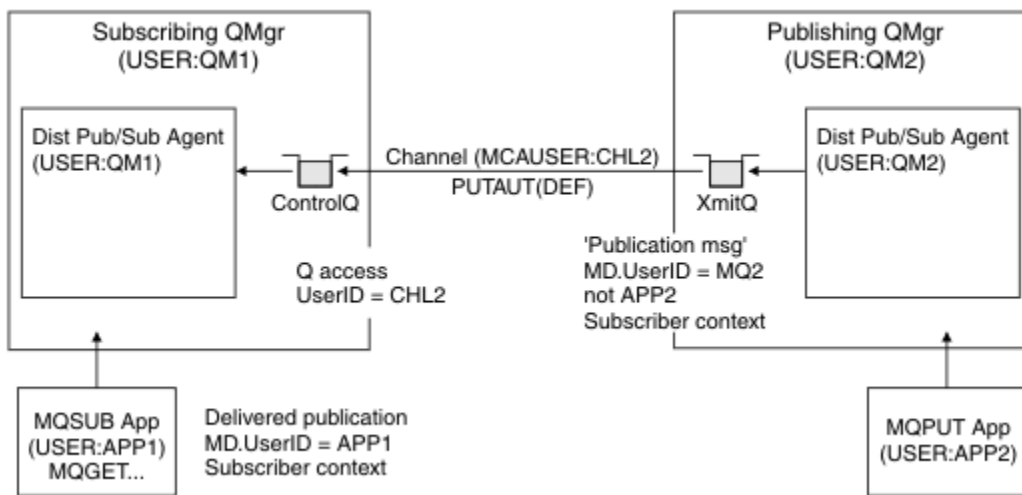


Abbildung 39. Proxy-Abonnementsicherheit, Weiterleitungsveröffentlichung

Auf einem System, auf dem wenig in Bezug auf die Sicherheit berücksichtigt wurde, werden die verteilten Publish/Subscribe-Prozesse wahrscheinlich unter einer Benutzer-ID in der Gruppe mqm ausgeführt, der Parameter MCAUSER in einem Kanal ist leer (der Standardwert), und die Nachrichten werden nach Bedarf an die verschiedenen Systemwarteschlangen zugestellt. Das ungesicherte System macht es einfach, einen Proof-of-Concept einzurichten, um verteiltes Publish/Subscribe zu demonstrieren.

Auf einem System, auf dem die Sicherheit ernsthaft in Betracht gezogen wird, unterliegen diese internen Nachrichten denselben Sicherheitssteuerungen wie jede Nachricht, die über den Kanal gesendet wird.

Wenn der Kanal mit einem nicht leeren MCAUSER und einem PUTAUT -Wert konfiguriert ist, der angibt, dass MCAUSER geprüft werden muss, muss dem Benutzer MCAUSER der Zugriff auf SYSTEM. INTER. QMGR. * -Warteschlangen erteilt werden. Wenn mehrere ferne Warteschlangenmanager vorhanden sind und die Kanäle unter verschiedenen MCAUSER -IDs ausgeführt werden, müssen alle diese Benutzer-IDs Zugriff auf die SYSTEM. INTER. QMGR. * -Warteschlangen erhalten. Kanäle, die unter verschiedenen MCAUSER -IDs ausgeführt werden, können z. B. auftreten, wenn mehrere hierarchische Verbindungen auf einem einzigen Warteschlangenmanager konfiguriert sind.

Wenn der Kanal mit einem PUTAUT -Wert konfiguriert wird, der angibt, dass der Kontext der Nachricht verwendet wird, wird der Zugriff auf die SYSTEM. INTER. QMGR. * -Warteschlangen basierend auf der Benutzer-ID innerhalb der internen Nachricht überprüft. Da alle diese Nachrichten mit der Benutzer-ID des verteilten Publish/Subscribe-Agenten aus dem Warteschlangenmanager, der die interne Nachricht sendet, oder einer Veröffentlichungsnachricht (siehe Abbildung 39 auf Seite 115) gestellt werden, ist es nicht zu groß, dass eine Gruppe von Benutzer-IDs Zugriff auf die verschiedenen Systemwarteschlangen (einen pro fernen Warteschlangenmanager) erteilt, wenn Sie Ihre verteilte Publish/Subscribe-Sicherheit auf diese Weise einrichten möchten. Es weist immer noch dieselben Probleme auf, die die Kanalkontextsicherheit immer hat; die der verschiedenen Benutzer-ID-Domänen und die Tatsache, dass die Benutzer-ID in der Nachricht möglicherweise nicht auf dem empfangenden System definiert ist. Es ist jedoch eine absolut akzeptable Möglichkeit, bei Bedarf auszuführen.

Alle Messaging-Manager-Nachrichten für die Zwecke verteilter Publish/Subscribe-Nachrichten werden unter Verwendung der normalen Kanalsicherheit ausgeführt.

Informationen zur Einschränkung von Veröffentlichungen und Proxy-Subskriptionen auf Themenebene finden Sie unter [Publish/Subscribe-Sicherheit](#).

Standard-Benutzer-IDs mit einer WS-Manager-Hierarchie verwenden

Wenn Sie eine Hierarchie von WS-Managern haben, die auf verschiedenen Plattformen ausgeführt werden und die Standardbenutzer-IDs verwenden, beachten Sie, dass diese Standardbenutzer-IDs von den Plattformen abweichen und auf der Zielplattform möglicherweise nicht bekannt sind. Dies hat zur Folge, dass ein Warteschlangenmanager, der auf einer Plattform ausgeführt wird, Nachrichten zurückweist, die

von Warteschlangenmanagern auf anderen Plattformen mit dem Ursachencode MQRC_NOT_AUTHORIZED empfangen wurden.

Um Nachrichten zu vermeiden, die mindestens zurückgewiesen werden, müssen die folgenden Berechtigungen zu den Standardbenutzer-IDs hinzugefügt werden, die auf anderen Plattformen verwendet werden:

- Berechtigung *PUT *GET für die Warteschlange SYSTEM.BROKER. Warteschlangen
- Berechtigung *PUB *SUB für SYSTEM.BROKER. Themen
- Berechtigung *ADMCR *ADMCLT *ADMCHG in der Warteschlange SYSTEM.BROKER.CONTROL.QUEUE.

Die Standardbenutzer-IDs lauten wie folgt:

Plattform	Standardbenutzer-ID
Windows	MUSR_MQADMIN Anmerkung: MUSR_MQADMIN ist die Standardbenutzer-ID für die erste Installation. Bei nachfolgenden Installationen erstellt der IBM WebSphere MQ -Vorbereitungsassistent ein Benutzerkonto mit dem NamenMUSR_MQADMINx, wobei x die nächste verfügbare Nummer ist, die eine Benutzer-ID darstellt, die nicht vorhanden ist.
UNIX and Linux-Systeme	mqm
IBM i	QMQM
z/OS	Die Benutzer-ID des Kanalinitiatoradressraums

Erstellen Sie die Benutzer-ID 'qmqm' und erteilen Sie sie, wenn sie hierarchisch einem Warteschlangenmanager unter IBM i für Warteschlangenmanager auf Windows-, UNIX-, Linux- und z/OS -Plattformen zugeordnet ist.

Erstellen und erteilen Sie Zugriff auf die Benutzer-ID 'mqm', wenn sie hierarchisch einem Warteschlangenmanager auf Windows-, UNIX- oder Linux für Warteschlangenmanager auf IBM i - und z/OS -Plattformen zugeordnet ist.

Erstellen Sie auf Windows-, UNIX-, Linux- und IBM i -Plattformen Benutzerzugriff auf die Benutzer-ID des z/OS -Kanalinitiatoradressraums, wenn sie hierarchisch einem Warteschlangenmanager unter z/OS zugeordnet ist, und erteilen Sie ihm den Benutzerzugriff.

Bei Benutzer-IDs kann die Groß-/Kleinschreibung beachtet werden. Der ursprüngliche Warteschlangenmanager (bei IBM i-, Windows-, UNIX- oder Linux -Systemen) erzwingt, dass die Benutzer-ID vollständig in Großbuchstaben angegeben wird. Der empfangende Warteschlangenmanager (falls Windows-, UNIX- oder Linux -Systeme) erzwingt, dass die Benutzer-ID in Kleinbuchstaben angegeben wird. Daher müssen alle Benutzer-IDs, die auf UNIX and Linux -Systemen erstellt wurden, in der Kleinbuchstaben erstellt werden. Wenn ein Nachrichtenexit installiert wurde, wird die Benutzer-ID nicht in Großbuchstaben oder in Kleinbuchstaben umgesetzt. Es muss sorgfältig darauf geachtet werden, wie der Nachrichtenexit die Benutzer-ID verarbeitet.

Gehen Sie wie folgt vor, um mögliche Probleme bei der Konvertierung von Benutzer-IDs zu

- Stellen Sie auf UNIX-, Linux - und Windows -Systemen sicher, dass die Benutzer-IDs in Kleinbuchstaben angegeben sind.
- Stellen Sie unter IBM i und z/OS sicher, dass die Benutzer-IDs in Großbuchstaben angegeben sind.

Verteilte Publish/Subscribe-Systemwarteschlangen

Vier Systemwarteschlangen werden von WS-Managern für Publish/Subscribe-Messaging verwendet. Sie müssen ihre Existenz nur zu Zwecken der Problembestimmung oder Kapazitätsplanung kennen.

Systemwarteschlange	Verwendungszweck
SYSTEM.INTER.QMGR.CONTROL	WebSphere MQ Steuerwarteschlange für verteiltes Publish/Subscribe
SYSTEM.INTER.QMGR.FANREQ	WebSphere MQ Eingabewarteschlange für verteilte Publish/Subscribe-interne Proxy-Subskription für Ausgabefächerung
SYSTEM.INTER.QMGR.PUBS	WebSphere MQ Veröffentlichungen für verteiltes Publish/Subscribe
SYSTEM.HIERARCHY.STATE	WebSphere MQ -Status der verteilten Publish/Subscribe-Hierarchiebeziehung

Die Attribute der Publish/Subscribe-Systemwarteschlangen sind in [Tabelle 10 auf Seite 117](#) aufgeführt.

Attribut	Standardwert
DEFPSIST	Ja
DEFSOPT	EXC
MAXMSG	Auf AIX, HP-UX-, Linux-, IBM i-, Solaris- und Windows -Plattformen: Der Wert des Parameters MAXMSG des Befehls ALTER QMGR
MAXDEPTH	999999999
SHARE	nicht zutreffend
STGKLASSE	Dieses Attribut wird nur auf z/OS -Plattformen verwendet.

Publish/Subscribe-Systemwarteschlangenfehler

Fehler können auftreten, wenn verteilte Publish/Subscribe-WS-Manager-Warteschlangen nicht verfügbar sind.

Wenn die Ausgabefächerungsanforderungswarteschlange SYSTEM.INTER.QMGR.FANREQ nicht verfügbar, die MQSUB-API empfängt Ursachencodes und Fehlermeldungen, die in das Fehlerprotokoll geschrieben werden, wenn Proxy-Subskriptionen an direkt verbundene Warteschlangenmanager zugestellt werden müssen.

Wenn die Statuswarteschlange der Hierarchiebeziehung SYSTEM.HIERARCHY.STATE nicht verfügbar, eine Fehlermeldung wird in das Fehlerprotokoll geschrieben und die Publish/Subscribe-Engine wird in den COMPAT-Modus versetzt.

Bei einem anderen SYSTEM.INTER.QMGR -Warteschlangen sind nicht verfügbar, eine Fehlermeldung wird in das Fehlerprotokoll geschrieben und obwohl die Funktion nicht inaktiviert ist, ist es wahrscheinlich, dass Publish/Subscribe-Nachrichten in Warteschlangen auf fernen Warteschlangenmanagern erstellt werden.

Wenn die Übertragungswarteschlange zu einem übergeordneten, untergeordneten oder Publish/Subscribe-Clusterwarteschlangenmanager nicht verfügbar ist:

1. Die MQPUT-API empfängt Ursachencodes und die Veröffentlichungen werden nicht zugestellt.
2. Empfangene Veröffentlichungen zwischen Warteschlangenmanagern werden in die Eingabewarteschlange zurückgesetzt und anschließend erneut versucht, in die Warteschlange für nicht zustellbare Nachrichten eingereiht zu werden, wenn der Rücksetzschwellenwert erreicht wird.
3. Proxy-Subskriptionen werden in die Warteschlange für Ausgabefächerung zurückgesetzt und anschließend erneut versucht, in die Warteschlange für nicht zustellbare Nachrichten eingereiht zu werden,

wenn der Rücksetzschwellenwert erreicht wird. In diesem Fall wird die Proxy-Subskription keinem verbundenen Warteschlangenmanager zugestellt.

4. Nachrichten des Hierarchiebeziehungsprotokolls schlagen fehl und der Verbindungsstatus wird im Befehl PUBSUB als FEHLER markiert.

Behandlung nicht zugestellter Nachrichten mit der Steerroutine der Warteschlange für nicht zustellbare Nachrichten von WebSphere MQ

Was ist eine Warteschlange für nicht zustellbare Nachrichten, wie werden Nachrichten in sie eingereicht und wie verwalten Sie sie?

Eine *Warteschlange für nicht zustellbare Nachrichten* (DLQ), die manchmal auch als *Warteschlange für nicht zugestellte Nachrichten* bezeichnet wird, ist eine Haltewarteschlange für Nachrichten, die nicht an ihre Zielwarteschlangen zugestellt werden können. Jedem Warteschlangenmanager in einem Netz sollte eine DLQ zugeordnet sein.

Nachrichten können von Warteschlangenmanagern, Nachrichtenkanalagenten (MCA; Message Channel Agent) und Anwendungen in die Warteschlange für nicht zustellbare Nachrichten eingereicht werden. Alle Nachrichten in der DLQ müssen mit einer *Headerstruktur einer nicht zustellbaren Nachricht* (MQDLH) als Präfix vorangesetzt werden.

Nachrichten, die von einem Warteschlangenmanager oder einem Nachrichtenkanalagenten in die DLQ eingereicht werden, verfügen immer über einen MQDLH; Anwendungen, die Nachrichten in die DLQ stellen, müssen einen MQDLH bereitstellen. Das Feld *Reason* in der MQDLH-Struktur enthält einen Ursachencode, der angibt, warum die Nachricht in die Warteschlange für nicht zustellbare Nachrichten gestellt wurde.

Alle WebSphere MQ -Umgebungen benötigen eine Routine zur regelmäßigen Verarbeitung von Nachrichten in der DLQ. WebSphere MQ stellt eine Standardroutine bereit, die als *Steerroutine der Warteschlange für nicht zustellbare Nachrichten* (DLQ-Steerroutine) bezeichnet wird, die Sie mit dem Befehl `runmqdlq` aufrufen.

Anweisungen für die Verarbeitung von Nachrichten in der DLQ werden dem DLQ-Handler mit Hilfe einer vom Benutzer geschriebenen *Regeltabelle* bereitgestellt. Das heißt, der DLQ-Handler gleicht Nachrichten in der Warteschlange für DLQ mit Einträgen in der Regeltabelle ab; wenn eine DLQ-Nachricht mit einem Eintrag in der Regeltabelle übereinstimmt, führt der DLQ-Handler die Aktion aus, die diesem Eintrag zugeordnet ist.

DLQ-Handler aufrufen

Der DLQ-Handler wird mit dem Befehl `runmqdlq` aufgerufen. Sie können die DLQ, die Sie verarbeiten möchten, und den Warteschlangenmanager, den Sie auf zwei Arten verwenden möchten, benennen.

Es gibt zwei Möglichkeiten, wie folgt:

- Als Parameter für `runmqdlq` über die Eingabeaufforderung. Beispiel:

```
runmqdlq ABC1.DEAD.LETTER.QUEUE ABC1.QUEUE.MANAGER <qrulc.rul
```

- In der Regeltabelle. Beispiel:

```
INPUTQ(ABC1.DEAD.LETTER.QUEUE) INPUTQM(ABC1.QUEUE.MANAGER)
```

Die Beispiele gelten für die DLQ mit dem Namen ABC1.DEAD.LETTER.QUEUE, deren Eigner der WS-Manager ABC1.QUEUE.MANAGER ist.

Wenn Sie die DLQ oder den Warteschlangenmanager nicht wie gezeigt angeben, wird der Standardwarteschlangenmanager für die Installation zusammen mit der DLQ, die zu diesem Warteschlangenmanager gehört, verwendet.

Der Befehl `runmqdlq` übernimmt seine Eingabe aus `stdin`. Sie ordnen die Regeltabelle `runmqdlq` zu, indem Sie `stdin` aus der Regeltabelle umleiten.

Um den DLQ-Handler ausführen zu können, müssen Sie berechtigt sein, sowohl auf die DLQ selbst als auch auf alle Nachrichtenwarteschlangen zuzugreifen, an die Nachrichten in der DLQ weitergeleitet werden. Damit der DLQ-Handler Nachrichten in Warteschlangen mit der Berechtigung der Benutzer-ID im Nachrichtenkontext einreicht, müssen Sie auch berechtigt sein, die Identität anderer Benutzer zu übernehmen.

Weitere Informationen zum Befehl `runmqdlq` finden Sie im Abschnitt [runmqdlq](#).

Beispiel-DLQ-Handler 'amqsdq'

Neben dem DLQ-Handler, der mit dem Befehl `runmqdlq` aufgerufen wird, stellt WebSphere MQ die Quelle des DLQ-Beispielhandlers 'amqsdq' mit einer Funktion bereit, die der von `runmqdlq` bereitgestellten Funktion ähnelt.

Sie können `amqsdq` anpassen, um einen DLQ-Handler bereitzustellen, der Ihre Anforderungen erfüllt. Sie können beispielsweise entscheiden, dass ein DLQ-Handler Nachrichten ohne Header von nicht zustellbaren Nachrichten verarbeiten kann. (Sowohl der Standard-DLQ-Handler als auch der Mustercode 'amqsdq' verarbeiten nur die Nachrichten in der Warteschlange für nicht zustellbare Nachrichten, die mit einem Header von nicht zustellbaren Nachrichten (MQDLH) beginnen. Nachrichten, die nicht mit einem MQDLH beginnen, werden als fehlerhaft erkannt und verbleiben unbegrenzt auf der DLQ.)

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

In WebSphere MQ für Windows wird die Quelle von `amqsdq` im folgenden Verzeichnis bereitgestellt:

```
MQ_INSTALLATION_PATH\tools\c\samples\dlq
```

und die kompilierte Version wird im folgenden Verzeichnis bereitgestellt:

```
MQ_INSTALLATION_PATH\tools\c\samples\bin
```

In WebSphere MQ für UNIX and Linux -Systeme wird die Quelle von `amqsdq` im folgenden Verzeichnis bereitgestellt:

```
MQ_INSTALLATION_PATH/samp/dlq
```

und die kompilierte Version wird im folgenden Verzeichnis bereitgestellt:

```
MQ_INSTALLATION_PATH/samp/bin
```

Die Regeltabelle für den DLQ-Handler

Die Regeltabelle des DLQ-Handlers definiert, wie der DLQ-Handler Nachrichten verarbeiten soll, die in der DLQ ankommen.

Es gibt zwei Typen von Einträgen in einer Regeltabelle:

- Der erste Eintrag in der Tabelle, der optional ist, enthält *Steuerdaten*.
- Alle anderen Einträge in der Tabelle sind *Regeln*, damit der DLQ-Handler folgen kann. Jede Regel besteht aus einem *Muster* (eine Gruppe von Nachrichtenmerkmalen), mit denen eine Nachricht abgeglichen wird, und eine *Aktion*, die ausgeführt werden soll, wenn eine Nachricht in der DLQ mit dem angegebenen Muster übereinstimmt. Es muss mindestens eine Regel in einer Regeltabelle vorhanden sein.

Jeder Eintrag in der Regeltabelle enthält ein oder mehrere Schlüsselwörter.

Steuerdaten

In diesem Abschnitt werden die Schlüsselwörter beschrieben, die Sie in einen Steuerdateneintrag in eine DLQ-Handler-Regeltabelle aufnehmen können.

Anmerkung:

- Die vertikale Linie (|) trennt Alternativen, von denen nur eine angegeben werden kann.

- Alle Schlüsselwörter sind optional.

INPUTQ (QueueName| " __ ")

Der Name der DLQ, die verarbeitet werden soll:

1. Jeder INPUTQ-Wert, den Sie als Parameter für den Befehl `runmqdlq` angeben, überschreibt jeden INPUTQ-Wert in der Regeltabelle.
2. Wenn Sie keinen INPUTQ-Wert als Parameter für den Befehl `runmqdlq` angeben, aber **einen** Wert in der Regeltabelle angeben, wird der INPUTQ-Wert in der Regeltabelle verwendet.
3. Wenn keine DLQ angegeben wird oder Sie INPUTQ ("") in der Regeltabelle angeben, wird der Name der DLQ verwendet, die zu dem Warteschlangenmanager mit dem Namen gehört, der als Parameter für den Befehl `runmqdlq` angegeben wurde.
4. Wenn Sie keinen INPUTQ-Wert als Parameter für den Befehl `runmqdlq` oder als Wert in der Regeltabelle angeben, wird die DLQ verwendet, die zu dem im Schlüsselwort INPUTQM in der Regeltabelle angegebenen Warteschlangenmanager gehört.

INPUTQM (QueueManagerName| ' __')

Der Name des Warteschlangenmanagers, der Eigner der im Schlüsselwort INPUTQ genannten DLQ ist:

1. Jeder INPUTQM-Wert, den Sie als Parameter für den `runmqdlq` -Befehl angeben, überschreibt jeden INPUTQM-Wert in der Regeltabelle.
2. Wenn Sie keinen INPUTQM-Wert als Parameter des Befehls `runmqdlq` angeben, wird der INPUTQM-Wert in der Regeltabelle verwendet.
3. Wenn kein Warteschlangenmanager angegeben ist oder Sie INPUTQM ("") in der Regeltabelle angeben, wird der Standardwarteschlangenmanager für die Installation verwendet.

RETRYINT (Intervall| 60)

Das Intervall (in Sekunden), in dem der DLQ-Handler Nachrichten in der DLQ erneut verarbeiten soll, die beim ersten Versuch nicht verarbeitet werden konnten und für die wiederholte Versuche angefordert wurden. Das Wiederholungsintervall beträgt standardmäßig 60 Sekunden.

WAIT (YES| NO |nnn)

Gibt an, ob der DLQ-Handler warten soll, bis weitere Nachrichten in der DLQ ankommen, wenn er feststellt, dass keine weiteren Nachrichten mehr verarbeitet werden können.

YES

Der DLQ-Handler wartet unbegrenzt.

Nein

Der DLQ-Handler wird beendet, wenn er feststellt, dass der DLQ entweder leer ist oder keine Nachrichten enthält, die er verarbeiten kann.

nnn

Der DLQ-Handler wartet auf *nnn* Sekunden für neue Arbeit, bevor er beendet wird, nachdem er festgestellt hat, dass die Warteschlange entweder leer ist oder keine Nachrichten enthält, die er verarbeiten kann.

Geben Sie WAIT (YES) für ausgelastete DLQs und WAIT (NO) oder WAIT (*nnn*) für DLQs mit einem niedrigen Aktivitätsgrad an. Wenn der DLQ-Handler beendet werden darf, rufen Sie ihn erneut auf, und verwenden Sie die Triggerung. Weitere Informationen zum Auslösen finden Sie unter [WebSphere MQ -Anwendungen mit Triggern starten](#) .

Eine Alternative zum Einschließen von Steuerdaten in die Regeltabelle besteht darin, die Namen der DLQ und ihres Warteschlangenmanagers als Eingabeparameter für den Befehl `runmqdlq` bereitzustellen. Wenn Sie einen Wert sowohl in der Regeltabelle als auch als Eingabe für den Befehl `runmqdlq` angeben, hat der im Befehl `runmqdlq` angegebene Wert Vorrang.

Wenn Sie einen Steuerdateneintrag in die Regeltabelle aufnehmen, muss er der **erste** Eintrag in der Tabelle sein.

Regeln (Muster und Aktionen)

Eine Beschreibung der Schlüsselwörter für die Mustererkennung (die Schlüsselwörter, mit denen Nachrichten in der DLQ übereinstimmen), und die Aktionsschlüsselwörter (die Schlüsselwörter, die bestimmen, wie der DLQ-Handler eine übereinstimmende Nachricht verarbeiten soll). Es wird auch eine Beispielregel bereitgestellt.

Die Schlüsselwörter für die Mustererkennung

Die Schlüsselwörter für die Mustererkennung, die Sie verwenden, um Werte anzugeben, mit denen Nachrichten in der DLQ übereinstimmen, lauten wie folgt. (Alle Schlüsselwörter für die Mustererkennung sind optional):

APPLIDAT (*ApplIdentityDaten* | *)**

Der *ApplIdentityData* -Wert, der im Nachrichtendeskriptor MQMD der Nachricht in der Warteschlange für nicht zuorden_Nachrichten angegeben ist.

APPLNAME (*PutApplName* | *)**

Der Name der Anwendung, die den MQPUT- oder MQPUT1-Aufruf ausgegeben hat, wie im Feld *PutApplName* des Nachrichtendeskriptors MQMD der Nachricht in der DLQ angegeben.

APPLTYPE (*PutApplTyp* | *)**

Der *PutApplType* -Wert, der im Nachrichtendeskriptor MQMD der Nachricht in der DLQ angegeben ist.

DESTQ (*QueueName* | *)**

Der Name der Nachrichtenwarteschlange, für die die Nachricht bestimmt ist.

DESTQM (*QueueManagerName* | *)**

Der Name des Warteschlangenmanagers für die Nachrichtenwarteschlange, für die die Nachricht bestimmt ist.

FEEDBACK (*Feedback* | *)**

Wenn der Wert *MsgType* den Wert MQFB_REPORT hat, beschreibt *Feedback* die Art des Berichts.

Sie können symbolische Namen verwenden. Sie können beispielsweise den symbolischen Namen MQFB_COA verwenden, um die Nachrichten in der DLQ zu identifizieren, die eine Bestätigung über ihre Ankunft in ihren Zielwarteschlangen benötigen.

FORMAT (*Format* | *)**

Der Name, den der Absender der Nachricht verwendet, um das Format der Nachrichtendaten zu beschreiben.

MSGTYPE (*MsgType* | *)**

Der Nachrichtentyp der Nachricht in der DLQ.

Sie können symbolische Namen verwenden. Sie können beispielsweise den symbolischen Namen MQMT_REQUEST verwenden, um die Nachrichten in der DLQ zu identifizieren, die Antworten benötigen.

PERSIST (*Persistence* | *)**

Der Persistenzwert der Nachricht. (Die Persistenz einer Nachricht bestimmt, ob sie die Neustarts des Warteschlangenmanagers überlebt.)

Sie können symbolische Namen verwenden. Sie können z. B. den symbolischen Namen MQPER_PERSISTENT verwenden, um Nachrichten in der DLQ zu identifizieren, die persistent sind.

REASON (*ReasonCode* | *)**

Der Ursachencode, der beschreibt, warum die Nachricht in die DLQ gestellt wurde.

Sie können symbolische Namen verwenden. Sie können beispielsweise den symbolischen Namen MQRC_Q_FULL verwenden, um die Nachrichten zu identifizieren, die in die Warteschlange für nicht zugebenden_Nachrichten gestellt wurden, weil ihre Zielwarteschlangen voll waren.

REPLYQ (*QueueName* | *)**

Der Name der Warteschlange für Antwortnachrichten, die im Nachrichtendeskriptor MQMD der Nachricht in der Warteschlange für DLQ angegeben ist.

REPLYQM (QueueManagerName|_*)

Der Name des Warteschlangenmanagers der Warteschlange für Antwortnachrichten, wie er im Nachrichtendeskriptor MQMD der Nachricht in der Warteschlange für die DLQ angegeben ist.

USERID (UserIdentifier|_*)

Die Benutzer-ID des Benutzers, der die Nachricht in der Warteschlange für nicht zuorden_Nachrichten erstellt hat, wie im Nachrichtendeskriptor MQMD der Nachricht in der DLQ angegeben ist.

Die Aktionsschlüsselwörter

Die Aktionsschlüsselwörter, die verwendet werden, um zu beschreiben, wie eine übereinstimmende Nachricht verarbeitet werden soll, lauten wie folgt:

ACTION (DISCARD | IGNORE | RETRY | FWD)

Die Aktion, die für alle Nachrichten in der DLQ ausgeführt werden soll, die mit dem in dieser Regel definierten Muster übereinstimmen.

KARTE

Löschen Sie die Nachricht aus der DLQ.

IGNORE

Hinterlassen Sie die Nachricht in der DLQ.

WIEDERHO

Wenn der erste Versuch, die Nachricht in die Zielwarteschlange zu stellen, fehlschlägt, versuchen Sie es erneut. Mit dem Schlüsselwort RETRY wird die Anzahl der Versuche festgelegt, eine Aktion zu implementieren. Das Schlüsselwort RETRYINT in den Steuerdaten steuert das Intervall zwischen den Versuchen.

FWD

Die Nachricht an die Warteschlange weiterleiten, die im Schlüsselwort FWDQ angegeben ist.

Sie müssen das Schlüsselwort ACTION angeben.

FWDQ (QueueName| & DESTQ | & REPLYQ)

Der Name der Nachrichtenwarteschlange, an die die Nachricht weitergeleitet werden soll, wenn ACTION (FWD) angefordert wird.

QueueName

Der Name einer Nachrichtenwarteschlange. FWDQ (") ist nicht gültig.

& DESTQ

Geben Sie den Warteschlangennamen aus dem Feld *DestQName* in der MQDLH-Struktur ein.

& REPLYQ

Der Warteschlangenname wird dem Feld für die Warteschlange für zu beantwortende Nachrichten (*ReplyToQ*) im Nachrichtendeskriptor MQMD entnommen.

Geben Sie im Nachrichtenmuster REPLYQ (? *) an, um Fehlernachrichten zu vermeiden, wenn eine Regel, die FWDQ (& REPLYQ) angibt, mit einer Nachricht mit einem leeren Feld *ReplyToQ* übereinstimmt.

FWDQM (QueueManagerName| & DESTQM | & REPLYQM | '___')

Der WS-Manager der Warteschlange, in die eine Nachricht weitergeleitet werden soll.

QueueManagerName

Der Name des Warteschlangenmanagers für die Warteschlange, an die eine Nachricht weitergeleitet werden soll, wenn ACTION (FWD) angefordert wird.

& DESTQM

Geben Sie den Namen des Warteschlangenmanagers aus dem Feld *DestQMgrName* in der MQDLH-Struktur ein.

& REPLYQM

Geben Sie den Namen des Warteschlangenmanagers aus dem Feld *ReplyToQMgr* im Nachrichtendeskriptor MQMD ein.

..

FWDQM (!), der der Standardwert ist, gibt den lokalen Warteschlangenmanager an.

HEADER (YES| NO)

Gibt an, ob der MQDLH in einer Nachricht verbleiben soll, für die ACTION (FWD) angefordert wird. Standardmäßig bleibt der MQDLH in der Nachricht. Das Schlüsselwort HEADER ist für andere Aktionen als FWD nicht gültig.

PUTAUT (DEF| CTX)

Die Berechtigung, mit der Nachrichten vom DLQ-Handler gestellt werden sollen:

DEF

Stellen Sie Nachrichten mit der Berechtigung des DLQ-Handlers selbst ein.

CTX

Stellen Sie die Nachrichten mit der Berechtigung der Benutzer-ID in den Nachrichtenkontext. Wenn Sie PUTAUT (CTX) angeben, müssen Sie berechtigt sein, die Identität der anderen Benutzer zu übernehmen.

RETRY (RetryCount| 1)

Die Anzahl der Male im Bereich von 1 bis 999.999.999, um eine Aktion zu versuchen (in dem im Schlüsselwort RETRYINT der Steuerdaten angegebenen Intervall). Die Anzahl der Versuche des DLQ-Handlers, eine bestimmte Regel zu implementieren, ist spezifisch für die aktuelle Instanz des DLQ-Steuerroutinen; die Anzahl bleibt nicht über Neustarts hinweg bestehen. Wenn der DLQ-Handler erneut gestartet wird, wird die Anzahl der Versuche, eine Regel anwenden zu lassen, auf Null zurückgesetzt.

Beispielregel

Nachfolgend ist eine Beispielregel aus einer Regeltabelle für DLQ-Handler-Regeln:

```
PERSIST(MQPER_PERSISTENT) REASON (MQRC_PUT_INHIBITED) +  
ACTION (RETRY) RETRY (3)
```

Diese Regel weist den DLQ-Handler an, drei Versuche zu unternehmen, alle persistenten Nachrichten, die in die DLQ eingereicht wurden, an seine Zielwarteschlange zu übermitteln, da MQPUT und MQPUT1 unterdrückt wurden.

Alle Schlüsselwörter, die Sie in einer Regel verwenden können, werden im restlichen Teil dieses Abschnitts beschrieben. Dabei ist Folgendes zu beachten:

- Der Standardwert für ein Schlüsselwort (falls vorhanden) ist unterstrichen. Bei den meisten Schlüsselwörtern ist der Standardwert * (Stern), der mit einem beliebigen Wert übereinstimmt.
- Die vertikale Linie (!) trennt Alternativen, von denen nur eine angegeben werden kann.
- Alle Schlüsselwörter mit Ausnahme von ACTION sind optional.

Konventionen für die Regeltabelle

Die Syntax, die Struktur und der Inhalt der Regeltabelle für den DLQ-Handler müssen diesen Konventionen entsprechen.

Die Regeltabelle muss die folgenden Konventionen einhalten:

- Eine Regeltabelle muss mindestens eine Regel enthalten.
- Schlüsselwörter können in beliebiger Reihenfolge vorkommen.
- Ein Schlüsselwort kann in jeder Regel nur einmal enthalten sein.
- Bei Schlüsselwörtern wird die Groß-/Kleinschreibung nicht beachtet.
- Ein Schlüsselwort und sein Parameterwert müssen durch mindestens ein Leerzeichen oder Komma von anderen Schlüsselwörtern getrennt werden.
- Es kann eine beliebige Anzahl von Leerzeichen am Anfang oder Ende einer Regel sowie zwischen Schlüsselwörtern, Interpunktionszeichen und Werten geben.

- Jede Regel muss in einer neuen Zeile beginnen.
- Auf Windows-Systemen muss die letzte Regel in der Tabelle mit einem Rücklauf-/Zeilenvorschubzeichen enden. Sie können dies erreichen, indem Sie sicherstellen, dass Sie die Eingabetaste am Ende der Regel drücken, damit die letzte Zeile der Tabelle eine Leerzeile ist.
- Aus Gründen der Portierbarkeit darf die signifikante Länge einer Zeile nicht größer als 72 Zeichen sein.
- Verwenden Sie das Pluszeichen (+) als letztes nicht leeres Zeichen in einer Zeile, um anzugeben, dass die Regel aus dem ersten nicht leeren Zeichen in der nächsten Zeile fortgesetzt wird. Verwenden Sie das Minuszeichen (-) als letztes nicht leeres Zeichen in einer Zeile, um anzuzeigen, dass die Regel ab dem Anfang der nächsten Zeile fortgesetzt wird. Fortsetzungszeichen können innerhalb von Schlüsselwörtern und Parametern vorkommen.

Beispiel:

```
APPLNAME(' ABC+
D')
```

führt zu 'ABCD' und

```
APPLNAME(' ABC-
D')
```

ergibt ' ABC D'.

- Kommentarzeilen, die mit einem Stern (*) beginnen, können an einer beliebigen Position in der Regeltabelle vorkommen.
- Leerzeilen werden ignoriert.
- Jeder Eintrag in der Tabelle der DLQ-Handler-Regeln enthält ein oder mehrere Schlüsselwörter und die zugehörigen Parameter. Die Parameter müssen den folgenden Syntaxregeln entsprechen:
 - Jeder Parameterwert muss mindestens ein signifikantes Zeichen enthalten. Die Begrenzungszeichen in Werten, die in Anführungszeichen eingeschlossen sind, werden nicht als bedeutsam betrachtet. Diese Parameter sind z. B. gültig:

FORMAT(' ABC ')	3 signifikante Zeichen
FORMAT(ABC)	3 signifikante Zeichen
FORMAT(' A ')	1 signifikantes Zeichen
FORMAT(A)	1 signifikantes Zeichen
FORMAT(' ')	1 signifikantes Zeichen

Diese Parameter sind ungültig, da sie keine signifikanten Zeichen enthalten:

```
FORMAT(' ')
FORMAT( )
FORMAT()
FORMAT
```

- Platzhalterzeichen werden unterstützt. Sie können das Fragezeichen (?) anstelle eines einzelnen Zeichens verwenden, mit Ausnahme einer abschließenden Leerzeichen; Sie können den Stern (*) anstelle von null oder mehr benachbarte Zeichen verwenden. Der Stern (*) und das Fragezeichen (?) werden als **Immer** in Parameterwerten als Platzhalterzeichen interpretiert.
- Platzhalterzeichen können nicht in die Parameter für die Schlüsselwörter ACTION, HEADER, RETRY, FWDQ, FWDQM und PUTAUT aufgenommen werden.
- Das Abbilden von Leerzeichen in Parameterwerten und in den entsprechenden Feldern in der Nachricht in der Warteschlange für DLQ ist bei der Ausführung von Platzhalterzeichen nicht von Bedeu-

tung. Führende und eingebettete Leerzeichen in Zeichenfolgen, die in einfache Anführungszeichen eingeschlossen sind, sind jedoch von Bedeutung für Platzhalterzeichen.

- Numerische Parameter dürfen das Fragezeichen (?) nicht als Platzhalterzeichen enthalten. Sie können den Stern (*) anstelle eines gesamten numerischen Parameters verwenden, jedoch nicht als Teil eines numerischen Parameters. Dies sind z. B. gültige numerische Parameter:

MSGTYPE (2)	Es können nur Antwortnachrichten ausgewählt werden.
MSGTYPE (*)	Beliebiger Nachrichtentyp ist auswählbar
MSGTYPE ('*')	Beliebiger Nachrichtentyp ist auswählbar

MSGTYPE ('2*') ist jedoch nicht gültig, da er einen Stern (*) als Teil eines numerischen Parameters enthält.

- Numerische Parameter müssen im Bereich von 0 bis 999 999 999 liegen. Wenn der Parameterwert in diesem Bereich liegt, wird er akzeptiert, auch wenn er derzeit in dem Feld, auf das sich das Schlüsselwort bezieht, nicht gültig ist. Sie können symbolische Namen für numerische Parameter verwenden.
- Ist ein Zeichenfolgewert kürzer als das Feld in dem MQDLH oder MQMD, auf das sich das Schlüsselwort bezieht, wird der Wert mit Leerzeichen auf die Länge des Felds aufgefüllt. Wenn der Wert, mit Ausnahme von Sternen, länger als das Feld ist, wird ein Fehler diagnostiziert. Dies sind z. B. alle gültigen Zeichenfolgewerte für ein 8 Zeichen-Feld:

'ABCDEFGH'	8 Zeichen
'A*C*E*G*I'	5 Zeichen mit Ausnahme von Sternen
'*A*C*E*G*I*K*M*O* *'	8 Zeichen mit Ausnahme von Sternen

- Schließen Sie Zeichenfolgen ein, die Leerzeichen, Kleinbuchstaben oder andere Sonderzeichen als Punkt (.), Schrägstrich (/), Unterstreichungszeichen () und Prozentzeichen (%) in einfache Anführungszeichen enthalten. Kleinbuchstaben, die nicht in Hochkommas eingeschlossen sind, werden in Großbuchstaben umgelegt. Wenn die Zeichenfolge ein Angebot enthält, verwenden Sie zwei einfache Anführungszeichen, um sowohl den Anfang als auch das Ende des Angebots zu kennzeichnen. Wenn die Länge der Zeichenfolge berechnet wird, wird jedes Vorkommen von doppelten Anführungszeichen als ein einzelnes Zeichen gezählt.

Informationen zur Verarbeitung der Regeltabelle

Der DLQ-Handler durchsucht die Regeltabelle nach einer Regel, bei der das Muster mit einer Nachricht in der DLQ übereinstimmt.

Die Suche beginnt mit der ersten Regel in der Tabelle und wird sequenziell durch die Tabelle fortgesetzt. Wenn der DLQ-Handler eine Regel mit einem übereinstimmenden Muster findet, führt er die Aktion von dieser Regel aus. Der DLQ-Handler erhöht den Wiederholungszähler für eine Regel immer dann um 1, wenn er diese Regel anwendet. Wenn der erste Versuch fehlschlägt, versucht der DLQ-Handler erneut, bis die Anzahl der Versuche mit der im Schlüsselwort RETRY angegebenen Anzahl übereinstimmt. Wenn alle Versuche fehlschlagen, sucht der DLQ-Handler nach der nächsten übereinstimmenden Regel in der Tabelle.

Dieser Prozess wird für nachfolgende übereinstimmende Regeln wiederholt, bis eine Aktion erfolgreich ausgeführt wird. Wenn für jede übereinstimmende Regel versucht wurde, die Anzahl der im Schlüsselwort RETRY angegebenen Zeiten zu verwenden, und alle Versuche fehlgeschlagen sind, wird ACTION (IGNORE) angenommen. ACTION (IGNORE) wird auch angenommen, wenn keine übereinstimmende Regel gefunden wird.

Anmerkung:

1. Übereinstimmende Regelmuster werden nur für Nachrichten in der DLQ gesucht, die mit einem MQDLH beginnen. Nachrichten, die nicht mit einem MQDLH beginnen, werden in regelmäßigen Abständen als fehlerhaft gemeldet und verbleiben auf unbegrenzte Zeit auf der DLQ.

2. Alle Musterschlüsselwörter können in der Standardeinstellung verwendet werden, so dass eine Regel nur aus einer Aktion bestehen kann. Beachten Sie jedoch, dass nur Regeln für Aktionen auf alle Nachrichten in der Warteschlange angewendet werden, die über MQDLHs verfügen und die noch nicht gemäß anderen Regeln in der Tabelle verarbeitet worden sind.
3. Die Regeltabelle wird geprüft, wenn der DLQ-Handler gestartet wird, und die Fehler werden zu diesem Zeitpunkt markiert. Sie können jederzeit Änderungen an der Regeltabelle vornehmen, diese Änderungen treten jedoch erst in Kraft, wenn der DLQ-Handler erneut gestartet wird.
4. Der DLQ-Handler ändert den Inhalt von Nachrichten, den MQDLH oder den Nachrichtendeskriptor nicht. Der DLQ-Handler reiht Nachrichten immer mit der Nachrichtenoption MQPMO_PASS_ALL_CONTEXT in andere Warteschlangen ein.
5. Aufeinanderfolgende Syntaxfehler in der Regeltabelle werden möglicherweise nicht erkannt, da die Regeltabelle so konzipiert ist, dass die Generierung von Wiederholungsfehlern während der Prüfung nicht mehr generiert wird.
6. Der DLQ-Handler öffnet die DLQ mit der Option MQOO_INPUT_AS_Q_DEF.
7. Es können mehrere Instanzen des DLQ-Handlers gleichzeitig mit derselben Warteschlange ausgeführt werden, wobei dieselbe Regeltabelle verwendet wird. Es ist jedoch eher üblich, dass es eine Eins-zu-eins-Beziehung zwischen einer DLQ und einem DLQ-Handler gibt.

Sicherstellen, dass alle DLQ-Nachrichten verarbeitet werden

Der DLQ-Handler hält fest, welche Nachrichten in der DLQ angesehen, aber nicht entfernt wurden.

Wenn Sie den DLQ-Handler als Filter verwenden, um eine kleine Untergruppe der Nachrichten aus der DLQ zu extrahieren, muss der DLQ-Handler weiterhin einen Satz dieser Nachrichten in der DLQ aufbewahren, die er nicht verarbeitet hat. Außerdem kann der DLQ-Handler nicht garantieren, dass neue Nachrichten, die in der DLQ ankommen, angezeigt werden, selbst wenn die DLQ als First-in-First-out (FIFO) definiert ist. Wenn die Warteschlange nicht leer ist, wird die DLQ in regelmäßigen Abständen erneut durchsucht, um alle Nachrichten zu überprüfen.

Aus diesen Gründen sollte sichergestellt werden, dass die DLQ so wenige Nachrichten wie möglich enthält. Wenn Nachrichten, die nicht gelöscht oder an andere Warteschlangen weitergeleitet werden können (aus welchem Grund auch immer), in der Warteschlange angesammelt werden können, erhöht sich die Auslastung des DLQ-Handlers und die DLQ selbst kann sich füllen.

Sie können bestimmte Maßnahmen ergreifen, um den DLQ-Handler zu aktivieren, um die DLQ zu leeren. Versuchen Sie z. B. nicht, ACTION (IGNORE) zu verwenden, wodurch Nachrichten in der Warteschlange für nicht zulassene Nachrichten hinterlassen werden. (Denken Sie daran, dass ACTION (IGNORE) für Nachrichten angenommen wird, die nicht explizit durch andere Regeln in der Tabelle adressiert werden.) Verwenden Sie stattdessen für die Nachrichten, die Sie ansonsten ignorieren würden, eine Aktion, mit der die Nachrichten in eine andere Warteschlange verschoben werden. Beispiel:

```
ACTION (FWD) FWDQ (IGNORED.DEAD.QUEUE) HEADER (YES)
```

Außerdem müssen Sie die letzte Regel in der Tabelle zu einem Catchall-Prozess machen, um Nachrichten zu verarbeiten, die nicht von früheren Regeln in der Tabelle adressiert wurden. Die letzte Regel in der Tabelle könnte z. B. wie folgt aussehen:

```
ACTION (FWD) FWDQ (REALLY.DEAD.QUEUE) HEADER (YES)
```

Dadurch werden Nachrichten weitergeleitet, die bis zur letzten Regel in der Tabelle in die Warteschlange REALLY.DEAD.QUEUE fallen, wo sie manuell verarbeitet werden können. Wenn Sie nicht über eine solche Regel verfügen, bleiben die Nachrichten wahrscheinlich unbegrenzt auf der DLQ.

Beispieltabelle für DLQ-Handler-Regeln

Eine Beispielregeltabelle für den Befehl 'runmqdlq' mit einem einzigen Steuerdateneintrag und mehreren Regeln.

```

*****
*       An example rules table for the runmqdlq command       *
*****
* Control data entry
* -----
* If no queue manager name is supplied as an explicit parameter to
* runmqdlq, use the default queue manager for the machine.
* If no queue name is supplied as an explicit parameter to runmqdlq,
* use the DLQ defined for the local queue manager.
*
inputqm(' ') inputq(' ')

* Rules
* -----
* We include rules with ACTION (RETRY) first to try to
* deliver the message to the intended destination.
* If a message is placed on the DLQ because its destination
* queue is full, attempt to forward the message to its
* destination queue. Make 5 attempts at approximately
* 60-second intervals (the default value for RETRYINT).

REASON(MQRC_Q_FULL) ACTION(RETRY) RETRY(5)

* If a message is placed on the DLQ because of a put inhibited
* condition, attempt to forward the message to its
* destination queue. Make 5 attempts at approximately
* 60-second intervals (the default value for RETRYINT).

REASON(MQRC_PUT_INHIBITED) ACTION(RETRY) RETRY(5)

* The AAAA corporation are always sending messages with incorrect
* addresses. When we find a request from the AAAA corporation,
* we return it to the DLQ (DEADQ) of the reply-to queue manager
* (&REPLYQM).
* The AAAA DLQ handler attempts to redirect the message.

MSGTYPE(MQMT_REQUEST) REPLYQM(AAAA.*) +
ACTION(FWD) FWDQ(DEADQ) FWDQM(&REPLYQM)

* The BBBB corporation never do things by half measures. If
* the queue manager BBBB.1 is unavailable, try to
* send the message to BBBB.2

DESTQM(bbbb.1) +
action(fwd) fwdq(&DESTQ) fwdqm(bbbb.2) header(no)

* The CCCC corporation considers itself very security
* conscious, and believes that none of its messages
* will ever end up on one of our DLQs.
* Whenever we see a message from a CCCC queue manager on our
* DLQ, we send it to a special destination in the CCCC organization
* where the problem is investigated.

REPLYQM(CCCC.*) +
ACTION(FWD) FWDQ(ALARM) FWDQM(CCCC.SYSTEM)

```

```

* Messages that are not persistent run the risk of being
* lost when a queue manager terminates. If an application
* is sending nonpersistent messages, it should be able
* to cope with the message being lost, so we can afford to
* discard the message. PERSIST(MQPER_NOT_PERSISTENT) ACTION(DISCARD)
* For performance and efficiency reasons, we like to keep
* the number of messages on the DLQ small.
* If we receive a message that has not been processed by
* an earlier rule in the table, we assume that it
* requires manual intervention to resolve the problem.
* Some problems are best solved at the node where the
* problem was detected, and others are best solved where
* the message originated. We don't have the message origin,
* but we can use the REPLYQM to identify a node that has
* some interest in this message.
* Attempt to put the message onto a manual intervention
* queue at the appropriate node. If this fails,
* put the message on the manual intervention queue at
* this node.

```

```
REPLYQM('?*') +  
ACTION(FWD) FWDQ(DEADQ.MANUAL.INTERVENTION) FWDQM(&REPLYQM)  
  
ACTION(FWD) FWDQ(DEADQ.MANUAL.INTERVENTION)
```

Mehrere Installationen

Unter UNIX, Linux, and Windows kann es mehrere Kopien von IBM WebSphere MQ auf einem System geben.

Sie können auswählen, wo die einzelnen Kopien von IBM WebSphere MQ installiert werden sollen, aber jede Kopie muss sich in einem anderen Installationsverzeichnis befinden. Maximal können 128 Installationen von IBM WebSphere MQ gleichzeitig auf einem einzelnen System vorhanden sein. Eine Installation kann eine Installation von IBM WebSphere MQ Version 7.0.1, Fixpack 6 oder höher sein. Sie haben nun folgende Wahl:

- Sie können die Einfachheit der Pflege und Verwaltung einer einzelnen Installation von IBM WebSphere MQ auf einem System beibehalten.
- Nutzen Sie die Flexibilität, die Ihnen durch die Aktivierung mehrerer IBM WebSphere MQ -Installationen geboten wird.

Bevor Sie mehrere Kopien von IBM WebSphere MQ installieren, müssen Sie folgende Punkte bedacht haben:

- Wird eine Kopie von IBM WebSphere MQ Version 7.0.1 auf dem System vorhanden sein?

Wenn IBM WebSphere MQ Version 7.0.1 auf Fixpackstufe 6 oder höher auf dem System installiert wird, müssen mehrere Einschränkungen berücksichtigt werden:

- Auf UNIX and Linux -Systemen muss IBM WebSphere MQ Version 7.0.1 an der Standardposition installiert sein.
- IBM WebSphere MQ Version 7.0.1 muss zuerst auf einem System installiert werden. Sie können IBM WebSphere MQ Version 7.0.1 nach der Installation von Version 7.1 oder höher nicht installieren. Wenn Sie Version 7.0.1 deinstallieren, kann sie nicht mehr erneut installiert werden, wenn eine höhere Version von WebSphere MQ installiert ist.
- IBM WebSphere MQ Version 7.0.1 ist automatisch die primäre Installation. Sie können keine andere Installation als primäre Installation auswählen, wenn IBM WebSphere MQ Version 7.0.1 installiert ist.

- Wo werden Sie die Kopien von IBM WebSphere MQ installieren?

Sie können das Installationsverzeichnis für Ihre Installationen von Version 7.1 oder höher frei auswählen. Weitere Informationen finden Sie unter [Installationsposition auswählen](#).

- Benötigen Sie eine primäre Installation?

Eine primäre Installation ist eine Installation, auf die systemweite Speicherpositionen verweisen. Weitere Informationen finden Sie unter [„Primäre Installation auswählen“](#) auf Seite 129.

- Wie stellen Ihre Anwendungen eine Verbindung her?

Überlegen Sie, wie Ihre Anwendungen nach den geeigneten IBM WebSphere MQ-Bibliotheken suchen. Weitere Informationen finden Sie in den Abschnitten [Anwendungen in einer Umgebung mit mehreren Installationen verbinden](#) und [.NET-Anwendungen in einer Umgebung mit mehreren Installationen verbinden](#).

- Müssen Ihre bestehenden Exits geändert werden?

Wenn IBM WebSphere MQ nicht an der Standardposition installiert wird, müssen Sie Ihre Exits aktualisieren. Weitere Informationen finden Sie im Abschnitt [Exits und installierbare Services schreiben und kompilieren](#).

- Welcher Warteschlangenmanager wird welcher Installation zugeordnet?

Jeder Warteschlangenmanager ist einer bestimmten Installation zugeordnet. 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. Weitere Informationen finden Sie im Abschnitt [Warteschlangenmanager einer Installation zuordnen](#).

- Wie richten Sie Ihre Umgebung so ein, dass sie mit jeder Installation arbeitet?

Bei Mehrfachinstallationen auf einem System müssen Sie berücksichtigen, wie Sie mit bestimmten Installationen arbeiten und wie Sie Befehle in der jeweiligen Installation ausgegeben werden. Sie können entweder den vollständigen Pfad des Befehls angeben oder Umgebungsvariablen mit den Befehlen **setmqenv** bzw. **crtmqenv** festlegen. Durch das Festlegen von Umgebungsvariablen können Sie den Befehlspfad in dieser Umgebung übergehen. Weitere Informationen finden Sie in den Abschnitten [setmqenv](#), und [crtmqenv](#).

Nachdem Sie diese Fragen beantwortet haben, können Sie IBM WebSphere MQ mit den im Abschnitt [IBM WebSphere MQ installieren](#) beschriebenen Schritten installieren.

Wenn Sie über vorhandene Installationen von IBM WebSphere MQ verfügen und die Funktion für mehrere Installationen verwenden möchten, um eine Migration von einer Version von IBM WebSphere MQ auf eine andere Version durchzuführen, lesen Sie den Abschnitt [Koexistenz von Warteschlangenmanagern für mehrere Installationen unter UNIX, Linux und Windows](#).

Support-Pack für IBM Message Service Client for .NET und Mehrfachinstallationen

Zur Unterstützung mehrerer Versionen muss die Komponente *Java- und .NET-Nachrichtenübertragung und Web-Services* mit dem IBM WebSphere MQ-Produkt installiert werden. Diese Komponente enthält alle Funktionen des Support-Pack *IBM Message Service Client for .NET* (IA9H). Wenn das Support-Pack auf dem System installiert ist, werden Mehrfachversionen nicht unterstützt. Deinstallieren Sie das Support-Pack vor der Installation von IBM WebSphere MQ. Weitere Informationen zum Installieren der .NET-Komponente finden Sie unter [WebSphere MQ-Klassen für .NET](#).

Zugehörige Konzepte

[UNIX, Linux und Windows: Parallele Migration von Version 7.0.1 auf Version 7.5](#)

[UNIX, Linux und Windows: Migration in mehreren Stufen von Version 7.0.1 auf Version 7.5](#)

Zugehörige Tasks

[Mehrere Installationen konfigurieren](#)

[WebSphere MQ-Installationen in einem System finden](#)

Primäre Installation auswählen

Auf Systemen, die Mehrfachinstallationen von IBM WebSphere MQ unterstützen (UNIX, Linux, and Windows), ist die primäre Installation diejenige, auf die sich systemweite IBM WebSphere MQ-Positionen beziehen. Die Verwendung einer primären Installation ist optional, jedoch praktisch.

Vor IBM WebSphere MQ Version 7.1 konnte jeweils nur eine Instanz des Produkts installiert werden. Auf Windows -Systemen wurden mehrere globale Umgebungsvariablen gesetzt, die auf diese Installation verweisen. Auf UNIX and Linux -Systemen wurden symbolische Links zu `/usr/lib`, `/usr/bin` und `/usr/include` hinzugefügt, die ebenfalls auf diese einzelne Installation verweisen.

Ab Version 7.1 können Sie mehrere Versionen von IBM WebSphere MQ unter UNIX, Linux, and Windows installieren. Es ist möglich, jederzeit mehrere Installationen von IBM WebSphere MQ auf einem dieser Systeme zu haben und optional eine dieser Installationen als primäre Installation zu konfigurieren. Umgebungsvariablen und symbolische Links, die auf eine einzelne Installation verweisen, sind weniger aussagefähig, wenn mehrere Versionen vorhanden sind. Bei einigen Funktionen ist es jedoch erforderlich, dass diese systemweiten Speicherpositionen funktionieren. Beispielsweise angepasste Benutzerscripts für die Verwaltung von IBM WebSphere MQ und Produkte anderer Anbieter. Diese Funktionen sind nur in der primären Installation verwendbar.

Wenn Sie auf UNIX and Linux -Systemen eine Installation als primäre Installation festlegen, werden symbolische Links zu den externen Bibliotheken und Steuerbefehlen dieser Installation in `/usr/lib` und `/usr/bin` hinzugefügt. Wenn Sie keine primäre Installation einrichten, werden die symbolischen Links nicht erstellt. Eine Liste der symbolischen Links, die zur primären Installation hergestellt werden, finden Sie in [Externe Bibliothek und Steuerbefehlslinks zur primären Installation unter UNIX and Linux](#).

Auf Windows -Systemen verweisen die globalen Umgebungsvariablen auf die Verzeichnisse, in denen die primäre Installation installiert wurde. Diese Umgebungsvariablen werden verwendet, um IBM WebSphere MQ -Bibliotheken, -Steuerbefehle und Headerdateien zu lokalisieren. Außerdem erfordern einige Funktionen des Betriebssystems auf Windows -Systemen die zentrale Registrierung von Schnittstellenbibliotheken, die dann in einen einzigen Prozess geladen werden. Bei mehreren Versionen von IBM WebSphere MQ gibt es widersprüchliche Gruppen von IBM WebSphere MQ -Bibliotheken. Die Komponenten würden versuchen, diese in Konflikt stehenden Bibliothekengruppen in einen Einzelprozess zu laden. Daher können diese Komponenten nur mit der primären Installation verwendet werden. Details zu einigen Features, die auf die Verwendung mit der primären Installation beschränkt sind, finden Sie unter [Features, die nur mit der primären Installation unter Windows](#).

Wenn Sie über eine Installation von IBM WebSphere MQ Version 7.0.1 auf dem System verfügen, ist diese Installation automatisch die primäre Installation. Die primäre Installation kann nicht geändert werden, während Version 7.0.1 installiert ist. Wenn alle Installationen auf dem System Version 7.1 oder höher aufweisen, können Sie auswählen, ob eine primäre Installation verwendet werden soll. Beachten Sie die Optionen in [Tabelle 11 auf Seite 130](#).

Tabelle 11. Optionen für primäre Installation.

Diese Tabelle zeigt die gültigen Installationskonfigurationen für primäre Installationen. Bei einem einzelnen Version 7.1 oder höher kann es sich entweder um einen primären oder einen nicht primären Server handeln. Bei mehreren Installationen muss eine unter Version 7.0.1 und eine oder mehrere unter Version 7.1 oder höher, Version 7.0.1 die primäre und die anderen Installationen die nicht primäre sein. Bei mehreren Installationen unter Version 7.1 oder höher kann eine Installation primär sein oder alle Installationen können nicht primär sein.

Optionen	Gültige Installationskonfigurationen		Zusatzinformationen
	Primär	Nicht primär	
Einzelinstallation von Version 7.1 oder höher.	Version 7.1 oder höher.	--	Wenn Sie weiter wie bei früheren Releases mit einer einzelnen Installation arbeiten möchten, konfigurieren Sie diese Installation als primäre Installation. Informationen zu dieser Option finden Sie im Abschnitt Einzelne Installation von IBM WebSphere MQ Version 7.1 oder höher, konfiguriert als primäre Installation
	--	Version 7.1 oder höher.	Wenn Sie weiter mit einer einzelnen Installation arbeiten möchten, aber keine symbolischen Links oder globalen Umgebungsvariablen erstellt werden sollen, konfigurieren Sie die Installation als nicht primäre Installation. Informationen zu den Auswirkungen dieser Option finden Sie im Abschnitt Einzelne Installation von IBM WebSphere MQ Version 7.1 oder höher, konfiguriert als nicht primäre Installation
Mehrere Installationen: Version 7.0.1 und Version 7.1 oder höher.	Version 7.0.1	Version 7.1 oder höher.	Wenn Sie mehrere Installationen von IBM WebSphere MQ mit einer Installation der Version 7.0.1 verwenden möchten, ist die Installation der Version 7.0.1 automatisch die primäre Installation. Während IBM WebSphere MQ Version 7.0.1 installiert ist, können Sie nicht ändern, welche Installation die primäre Installation ist. Informationen zu dieser Option und ihren Auswirkungen finden Sie unter Mehrere Installationen von IBM WebSphere MQ, eine unter Version 7.0.1 .

Tabelle 11. Optionen für primäre Installation.

Diese Tabelle zeigt die gültigen Installationskonfigurationen für primäre Installationen. Bei einem einzelnen Version 7.1 oder höher kann es sich entweder um einen primären oder einen nicht primären Server handeln. Bei mehreren Installationen muss eine unter Version 7.0.1 und eine oder mehrere unter Version 7.1 oder höher, Version 7.0.1 die primäre und die anderen Installationen die nicht primäre sein. Bei mehreren Installationen unter Version 7.1 oder höher kann eine Installation primär sein oder alle Installationen können nicht primär sein.

(Forts.)

Optionen	Gültige Installationskonfigurationen		Zusatzinformationen
Mehrere Installationen: Version 7.1 oder höher.	Version 7.1 oder höher.	Version 7.1 oder höher.	Wenn Sie mehrere Installationen von WebSphere MQ mit Version 7.1 oder höher verwenden möchten, können Sie wählen, ob eine der Installationen die primäre Installation sein soll. Informationen zu dieser Option finden Sie im Abschnitt <u>Mehrere Installationen von IBM WebSphere MQ Version 7.1 oder höher.</u>
	--	Version 7.1 oder höher.	

Zugehörige Konzepte

[Einzelinstallation von WebSphere MQ Version 7.1 oder höher, konfiguriert als primäre Installation](#)

[Einzelne Installation von WebSphere MQ Version 7.1 oder höher, konfiguriert als nicht primäre Installation](#)

[Mehrere Installationen von WebSphere MQ Version 7.1 oder höher](#)

[Mehrere Installationen von WebSphere MQ, eine mit Version 7.0.1](#)

Zugehörige Tasks

[Primäre Installation ändern](#)

[Installationsverzeichnis auswählen](#)

[Installation planen](#)

[Auswählen eines Installationsnamens](#)

Einzelne Installation von IBM WebSphere MQ Version 7.1 oder höher, konfiguriert als primäre Installation

Durch die Kennzeichnung einer IBM WebSphere MQ-Installation als primäre Installation werden symbolische Links oder globale Umgebungsvariablen zum System hinzugefügt, sodass die von den Anwendungen verwendeten IBM WebSphere MQ-Befehle und -Bibliotheken bei minimaler Anpassung der Systemkonfiguration automatisch verfügbar sind.

Sie entscheiden, wo IBM WebSphere MQ installiert werden soll.

Wo möglich, konfigurieren Sie Anwendungen und Scripts zur Verwendung des Suchpfads des Systems für die Suche nach den IBM WebSphere MQ-Steuerbefehlen oder IBM WebSphere MQ-Bibliotheken. Diese Konfiguration von Anwendungen und Scripts bietet eine hohe Flexibilität bei der Ausführung künftiger Aufgaben wie einer Migration auf das nächste Release von IBM WebSphere MQ oder der Installation einer zweiten Installation. Weitere Informationen zu den Optionen für den Verbindungsaufbau zu Ihren Anwendungen finden Sie unter [Anwendungen in einer Umgebung mit mehreren Installationen verbinden.](#)

Unter Windows wird die erste Installation automatisch als primäre Installation konfiguriert. Auf den Plattformen UNIX and Linux muss die erste Installation auf einem System manuell als primäre Installation konfiguriert werden. Legen Sie die primäre Installation mit dem Befehl **setmqinst** fest. Weitere Informationen finden Sie im Abschnitt [Primäre Installation deinstallieren, aktualisieren und verwalten.](#)

Zugehörige Tasks

[Primäre Installation ändern](#)

[Installationsverzeichnis auswählen](#)

[Installation planen](#)

[Auswählen eines Installationsnamens](#)

Einzelne Installation von IBM WebSphere MQ Version 7.1 oder höher, konfiguriert als nicht primäre Installation

Wenn Sie IBM WebSphere MQ Version 7.1 oder höher installieren, müssen Sie im Fall der nicht primären Installation möglicherweise einen Bibliothekspfad für Anwendungen konfigurieren, in dem IBM WebSphere MQ-Bibliotheken geladen werden können. Unter Windows sind einige Produktfunktionen nur verfügbar, wenn IBM WebSphere MQ als primäre Instanz konfiguriert ist.

UNIX-und Linux -Systeme

Das Ausführen einer nicht primären Installation unter UNIX and Linux hat folgende Auswirkungen:

- Anwendungen, die ihre IBM WebSphere MQ-Bibliotheken über einen eingebetteten Bibliothekspfad suchen, z. B. RPATH, können diese Bibliotheken unter folgenden Bedingungen nicht finden:
 - IBM WebSphere MQ ist in einem anderen Verzeichnis installiert, als in RPATH angegeben.
 - Das Verzeichnis `/usr` enthält keine symbolischen Links.
- Wenn Anwendungen ihre Bibliotheken mithilfe eines externen Bibliothekspfads lokalisieren, z. B. `LD_LIBRARY_PATH`, müssen Sie den externen Bibliothekspfad so konfigurieren, dass er das Verzeichnis `MQ_INSTALLATION_PATH/lib` oder `MQ_INSTALLATION_PATH/lib64` enthält. Die Befehle **setmqenv** und **crtmqenv** können mehrere Umgebungsvariablen in der aktuellen Shell konfigurieren, einschließlich des externen Bibliothekspfads.
- Die meisten IBM WebSphere MQ-Prozesse werden als 'setuid/setgid' ausgeführt. Somit ignorieren sie beim Laden von Benutzerexits den externen Bibliothekspfad. Benutzerexits, die IBM WebSphere MQ-Bibliotheken referenzieren, können diese Bibliotheken nur finden, wenn sie in dem integrierten Bibliothekspfad gefunden werden, der in den Exits integriert ist. Sie würden aufgelöst werden, wenn in `/usr` ein symbolischer Link vorhanden wäre. Benutzerexits, die auf IBM WebSphere MQ Version 7.1 oder höher ausgeführt werden sollen, können jetzt so erstellt werden, dass sie grundsätzlich nicht auf IBM WebSphere MQ-Bibliotheken verweisen. Stattdessen gehen sie davon aus, dass IBM WebSphere MQ Funktionszeiger an die IBM WebSphere MQ-Funktionen übergibt, die der Exit dann verwenden kann. Weitere Informationen finden Sie im Abschnitt [Exits und installierbare Services schreiben und kompilieren](#).

Weitere Informationen zu den Optionen für den Verbindungsaufbau zu Ihren Anwendungen finden Sie unter [Anwendungen in einer Umgebung mit mehreren Installationen verbinden](#).

Auf den Plattformen UNIX and Linux wird die erste Installation auf einem System nicht automatisch als die primäre Installation konfiguriert. Allerdings enthält `/usr/bin` einen einzigen symbolischen Link zur Lokalisierung des Befehls **dspmqr**. Wenn Sie keine symbolischen Links wünschen, können Sie diesen Link mit dem folgenden Befehl entfernen:

```
setmqinst -x -p MQ_INSTALLATION_PATH
```

Windows-Systeme

Das Ausführen einer nicht primären Installation unter Windows hat folgende Auswirkungen:

- Anwendungen suchen ihre Bibliotheken normalerweise unter Verwendung des externen Bibliothekspfads `PATH`. Es gibt kein Konzept eines integrierten Bibliothekspfads oder einen expliziten Bibliotheksstandort. Wenn die Installation nicht primär ist, enthält die globale Umgebungsvariable `PATH` nicht das IBM WebSphere MQ -Installationsverzeichnis. Damit Anwendungen IBM WebSphere MQ-Bibliotheken finden können, aktualisieren Sie die `PATH`-Umgebungsvariable, sodass sie das Installationsverzeichnis von IBM WebSphere MQ referenziert. Die Befehle **setmqenv** und **crtmqenv** können mehrere Umgebungsvariablen in der aktuellen Shell konfigurieren, einschließlich des externen Bibliothekspfads.

- Einige Produktfunktionen sind nur verfügbar, wenn eine Installation als primäre Installation konfiguriert ist (siehe [Features](#), die nur mit der primären Installation unter Windows).

Standardmäßig wird unter Windows die erste Installation automatisch als primäre Installation konfiguriert. Sie müssen sie manuell als die primäre Installation auswählen.

Zugehörige Tasks

[Primäre Installation ändern](#)

[Installationsverzeichnis auswählen](#)

[Installation planen](#)

[Auswählen eines Installationsnamens](#)

Zugehörige Verweise

[setmqenv](#)

[crtmqenv](#)

Mehrere Installationen von IBM WebSphere MQ Version 7.1oder höher

Sie können eine der Installationen von IBM WebSphere MQ Version 7.1 oder höher als primäre Installation konfigurieren. Ihre Entscheidung hängt davon ab, wie die Anwendungen nach Bibliotheken suchen.

Die Schnittstellenbibliotheken von IBM WebSphere MQ, wie z. B. mqm, die zum Lieferumfang von IBM WebSphere MQ Version 7.1 gehören, verwenden automatisch die Installation, die für den Warteschlangenmanager, zu dem sie eine Verbindung herstellen, erforderlich ist. Wenn also eine Anwendung ihre IBM WebSphere MQ-Bibliothek aus einer IBM WebSphere MQ Version 7.1-Installation sucht, kann sie eine Verbindung zu jedem Warteschlangenmanager in diesem System herstellen. Die Konfiguration einer IBM WebSphere MQ Version 7.1-Installation als primäre Installation stellt sicher, dass eine Anwendung, die nach ihrer IBM WebSphere MQ-Schnittstellenbibliothek sucht, eine Verbindung zu jedem Warteschlangenmanager herstellen kann.

Weitere Informationen zur Verbindung von Anwendungen in einer Umgebung mit mehreren Installationen finden Sie unter [Anwendungen in einer Umgebung mit mehreren Installationen verbinden](#).

Die primäre Installation wird nicht automatisch geändert, wenn Sie sie deinstallieren. Wenn Sie eine andere Installation als primäre Installation verwenden möchten, müssen Sie dies manuell mit dem Befehl **setmqinst** festlegen. Weitere Informationen finden Sie im Abschnitt [Primäre Installation deinstallieren, aktualisieren und verwalten](#).

Zugehörige Konzepte

[Mehrere Installationen](#)

Zugehörige Tasks

[Primäre Installation ändern](#)

[Installationsverzeichnis auswählen](#)

[Installation planen](#)

[Auswählen eines Installationsnamens](#)

Mehrere Installationen von IBM WebSphere MQunter Version 7.0.1

IBM WebSphere MQ Version 7.1oder höher kann mit IBM WebSphere MQ Version 7.0.1 mit einigen Einschränkungen koexistieren.

- Auf UNIX and Linux -Systemen kann Version 7.0.1 nur in einer festen Standardposition installiert werden. Daher können Sie Version 7.1oder höher nicht an dieser Standardposition installieren.
- IBM WebSphere MQ Version 7.0.1 wird automatisch als primäre Installation konfiguriert. Auf Systemen mit UNIX and Linux werden symbolische Links automatisch in den entsprechenden Verzeichnissen von IBM WebSphere MQ erstellt. Unter Windows wird alles, was das Produkt bereitstellt, global registriert. IBM WebSphere MQ Version 7.0.1 muss auf diese Weise installiert werden, damit funktioniert. Wenn also IBM WebSphere MQ Version 7.0.1 installiert ist, kann eine Installation von IBM WebSphere MQ Version 7.1oder höher nicht als primäre Installation festgelegt werden.

Die Bibliotheken von IBM WebSphere MQ Version 7.1 können mit jedem Warteschlangenmanager arbeiten, der unter IBM WebSphere MQ Version 7.0.1 (oder höher) ausgeführt wird. Wenn eine Anwendung eine Verbindung zu Warteschlangenmanagern unter Version 7.0.1 oder höheren Versionen herstellen muss, kann sie weiterhin normal funktionieren, wenn die folgenden Bedingungen zutreffen:

- Sie sucht die Bibliotheken von IBM WebSphere MQ Version 7.1 (oder höher) zur Laufzeit.
- Es verwendet nur Funktionen, die in Version 7.0.1 verfügbar sind.

Weitere Informationen zur Verbindung von Anwendungen in einer Umgebung mit mehreren Installationen finden Sie unter [Anwendungen in einer Umgebung mit mehreren Installationen verbinden](#).

Die primäre Installation wird nicht automatisch geändert, wenn Sie IBM WebSphere MQ Version 7.0.1 deinstallieren. Wenn Sie eine andere Installation als primäre Installation verwenden möchten, müssen Sie dies manuell mit dem Befehl **setmqinst** festlegen. Weitere Informationen finden Sie im Abschnitt [Primäre Installation deinstallieren, aktualisieren und verwalten](#).

Zugehörige Konzepte

[Mehrere Installationen](#)

Zugehörige Tasks

[Installationsverzeichnis auswählen](#)

[Installation planen](#)

[Auswählen eines Installationsnamens](#)

Speicher- und Leistungsanforderungen planen

Sie müssen realistische und erreichbare Speicher- und Leistungsziele für Ihr IBM WebSphere MQ-System festlegen. Verwenden Sie die Links, um Informationen zu Faktoren zu finden, die sich auf die Speicherung und Leistung auf Ihrer Plattform auswirken.

Die Anforderungen richten sich nach den Systemen, auf denen Sie IBM WebSphere MQ verwenden und nach den Komponenten, die Sie verwenden möchten.

Aktuelle Informationen zu den unterstützten Hardware- und Softwareumgebungen finden Sie auf der [Systemvoraussetzungen für IBM WebSphere MQ-Website](#) unter:

www.ibm.com/software/integration/wmq/requirements/

IBM WebSphere MQ speichert Warteschlangenmanagerdaten im Dateisystem. Verwenden Sie die folgenden Links, um Informationen zur Planung und Konfiguration von Verzeichnisstrukturen für die Verwendung mit IBM WebSphere MQ zu erhalten:

- [„Dateisystemunterstützung planen“](#) auf Seite 136
- [„Voraussetzungen für gemeinsam genutzte Dateisysteme“](#) auf Seite 137
- [„IBM WebSphere MQ -Dateien teilen“](#) auf Seite 146
- [„Verzeichnisstruktur auf UNIX and Linux -Systemen“](#) auf Seite 149
- [„Verzeichnisstruktur auf Windows -Systemen“](#) auf Seite 159

Verwenden Sie die folgenden Links, um Informationen zu Systemressourcen, gemeinsam genutztem Speicher und Prozesspriorität in UNIX and Linux zu erhalten:

- [„IBM WebSphere MQ -und UNIX System V IPC-Ressourcen“](#) auf Seite 163
- [„Gemeinsam genutzter Speicher unter AIX“](#) auf Seite 163
- [„Prozesspriorität WebSphere MQ und UNIX“](#) auf Seite 164

Zugehörige Konzepte

[„Planung“](#) auf Seite 5

Bei der Planung Ihrer IBM WebSphere MQ -Umgebung müssen Sie die zu konfigurierende IBM WebSphere MQ -Architektur, den Ressourcenbedarf, den Bedarf an Protokollierungs- und Sicherungsfunktionen

berücksichtigen. Verwenden Sie die Links in diesem Abschnitt, um die Umgebung zu planen, in der IBM WebSphere MQ ausgeführt wird.

[„IBM WebSphere MQ -Architektur entwerfen“ auf Seite 15](#)

In diesem Abschnitt finden Sie Informationen zu den verschiedenen Architekturen, die IBM WebSphere MQ für Punkt-zu-Punkt- und Publish/Subscribe-Messaging unterstützt.

[Hardware- und Softwarevoraussetzungen unter UNIX und Linux](#)

[Hardware- und Softwarevoraussetzungen unter Windows](#)

Erforderer Plattenspeicherplatz

Der Speicherbedarf für WebSphere MQ ist davon abhängig, welche Komponenten Sie installieren und wie viel Arbeitsspeicher Sie benötigen.

Der Plattenspeicher ist für die optionalen Komponenten, die Sie installieren möchten, erforderlich, einschließlich aller vorausgesetzten Komponenten, die sie benötigen. Der Gesamtspeicherbedarf hängt auch von der Anzahl der verwendeten Warteschlangen, der Anzahl und Größe der Nachrichten in den Warteschlangen und davon ab, ob die Nachrichten permanent sind. Sie benötigen außerdem die Archivierungskapazität auf Platte, Band oder anderen Medien sowie Speicherplatz für Ihre eigenen Anwendungsprogramme.

Die folgende Tabelle zeigt den ungefähren Plattenspeicherplatz, der erforderlich ist, wenn Sie verschiedene Kombinationen des Produkts auf verschiedenen Plattformen installieren. (Die Werte werden auf die nächsten 5 MB aufgerundet, wobei ein MB 1.048.576 Byte beträgt.)

Plattform	Clientinstallation ¹	Serverinstallation ²	WebSphere MQ-MFT-Installation ³	Vollständige Installation ⁴
AIX	145 MB	190 MB	705 MB	915 MB
HP-UX	225 MB	310 MB	1075 MB	1340 MB
IBM i	215 MB	450 MB	80 MB	655 MB
Linux für System x (32 Bit)	85 MB	N/A	N/A	120 MB
Linux für System x (64 Bit)	125 MB	170 MB	575 MB	935 MB
Linux auf POWER Systems-Big Endian	130 MB	170 MB	565 MB	715 MB
Solaris x86-64, AMD64, EM64T und kompatible Prozessoren	105 MB ⁵	150 MB ⁵	695 MB	860 MB
Solaris SPARC	105 MB ⁵	150 MB ⁵	680 MB	820 MB
Windows (32-Bit-Installation) ⁶	390 MB	N/A	N/A	475 MB
Windows (64-Bit-Installation) ⁶	445 MB	555 MB	710 MB	1005 MB

Hinweise zur Verwendung

1. Eine Clientinstallation umfasst die folgenden Komponenten:

- Laufzeit
- Client

2. Eine Serverinstallation umfasst die folgenden Komponenten:

- Laufzeit
 - Server
3. Eine IBM WebSphere MQ Managed File Transfer-Installation umfasst die folgenden Komponenten:
 - IBM WebSphere MQ Managed File Transfer Service, Logger, Agent, Tools und Base
 - Laufzeit
 - Server
 - Java
 - JRE
 4. Eine vollständige Installation enthält alle verfügbaren Komponenten.
 5.  Auf Solaris-Plattformen müssen Sie die Installation im Hintergrund ausführen, um diese Komponentenkombination zu erhalten.
 6.  Nicht alle der hier aufgelisteten Komponenten können auf Windows-Systemen installiert werden; ihre Funktionalität ist gelegentlich in anderen Komponenten enthalten. Siehe [WebSphere MQ-Features für Windows-Systeme](#).

Zugehörige Tasks

[Auswahl der zu installierenden Komponenten](#)

Dateisystemunterstützung planen

WS-Manager-Daten werden im Dateisystem gespeichert. Ein Warteschlangenmanager verwendet die Sperrung von Dateisystemen, um zu verhindern, dass mehrere Instanzen eines Warteschlangenmanagers mit mehreren Instanzen gleichzeitig aktiv sind.

Gemeinsam genutzte Dateisysteme

Gemeinsam genutzte Dateisysteme ermöglichen es mehreren Systemen, gleichzeitig auf dieselbe physische Speichereinheit zuzugreifen. Eine Unterbrechung würde auftreten, wenn mehrere Systeme direkt auf dieselbe physische Speichereinheit zugegriffen haben, ohne dass die Steuerung von Sperrungen und gemeinsamen Zugriff erzwungen werden muss. Betriebssysteme stellen lokale Dateisysteme mit Sperrung und Steuerung des gemeinsamen Zugriffs für lokale Prozesse bereit. Netzdateisysteme stellen die Steuerung von Sperrungen und die Steuerung des gemeinsamen Zugriffs für verteilte Systeme bereit.

Historische, vernetzte Dateisysteme haben nicht schnell genug ausgeführt oder eine ausreichende Sperrung und Steuerung des gemeinsamen Zugriffs bereitgestellt, um die Anforderungen für die Protokollierung von Nachrichten zu erfüllen. Heute können vernetzte Dateisysteme eine gute Leistung bieten und Implementierungen zuverlässiger Netzdateisystemprotokolle, wie z. B. *RFC 3530, Network File System (NFS) Version 4, Protokoll*, erfüllen die Anforderungen für die zuverlässige Protokollierung von Nachrichten.

Gemeinsam genutzte Dateisysteme und WebSphere MQ

WS-Manager-Daten für einen WS-Manager mit mehreren Instanzen werden in einem gemeinsam genutzten Netzdateisystem gespeichert. Auf Microsoft Windows- UNIX and Linux -Systemen müssen die Daten-dateien und Protokolldateien des Warteschlangenmanagers in das gemeinsam genutzte Netzdateisystem gestellt werden.

Vor Release v7.0.1 unterstützt WebSphere MQ keine Warteschlangenmanagerdaten, die im Netzspeicher gespeichert sind, auf den als gemeinsam genutztes Dateisystem zugegriffen wird. Wenn WS-Manager-Daten in den gemeinsam genutzten Netzspeicher gestellt werden, müssen Sie sicherstellen, dass auf die WS-Manager-Daten nicht von einer anderen Instanz des Warteschlangenmanagers zugegriffen wird, die gleichzeitig ausgeführt wird.

Ab v7.0.1 verwendet WebSphere MQ Sperrungen, um zu verhindern, dass mehrere Instanzen desselben Multi-Instanz-Warteschlangenmanagers gleichzeitig aktiv sind. Dieselbe Sperre stellt auch sicher, dass zwei

separate Warteschlangenmanager nicht versehentlich die gleiche Gruppe von WS-Manager-Datendateien verwenden können. Es kann immer nur eine Instanz eines Warteschlangenmanagers gleichzeitig gesperrt sein. Folglich unterstützt WebSphere MQ Warteschlangenmanagerdaten, die im Netzspeicher gespeichert sind, auf den als gemeinsam genutztes Dateisystem zugegriffen wird.

Da nicht alle Sperrprotokolle von Netzdateisystemen stabil sind und ein Dateisystem möglicherweise für die Leistung und nicht für die Datenintegrität konfiguriert ist, müssen Sie den Befehl **amqmfsc** ausführen, um zu testen, ob ein Netzdateisystem den Zugriff auf Warteschlangenmanagerdaten und -Protokolle ordnungsgemäß steuert. Dieser Befehl gilt nur für UNIX- und IBM i-Systeme. Unter Microsoft Windows gibt es nur ein unterstütztes Netzdateisystem und der Befehl **amqmfsc** ist nicht erforderlich.

Zugehörige Tasks

„Verhalten eines gemeinsam genutzten Dateisystems überprüfen“ auf Seite 139

Führen Sie **amqmfsc** aus, um zu überprüfen, ob ein gemeinsam genutztes Dateisystem auf UNIX -Systemen die Anforderungen zum Speichern der Warteschlangenmanagerdaten eines Multi-Instanz-Warteschlangenmanagers erfüllt. Führen Sie das IBM WebSphere MQ MQI client -Beispielprogramm **amqsfhac** parallel zu **amqmfsc** aus, um zu demonstrieren, dass ein Warteschlangenmanager die Nachrichtenintegrität während eines Fehlers beibehält.

Voraussetzungen für gemeinsam genutzte Dateisysteme

Systeme mit gemeinsam genutzten Dateien müssen Datenschreibintegrität, garantierten exklusiven Zugriff auf Dateien und Release-Sperren für das Fehlschlagen der Arbeit mit IBM WebSphere MQ bereitstellen.

Anforderungen an ein gemeinsam genutztes Dateisystem

Drei grundlegende Anforderungen muss ein gemeinsam genutztes Dateisystem für die zuverlässige Nachrichtenprotokollierung erfüllen:

1. Datenschreibintegrität

Die Datenschreibintegrität wird manchmal auch als *Write through to disk on disk on flush* bezeichnet. Der Warteschlangenmanager muss in der Lage sein, mit Daten zu synchronisieren, die erfolgreich auf der physischen Einheit festgeschrieben wurden. In einem transaktionsorientierten System müssen Sie sicherstellen, dass einige Schreibvorgänge sicher festgeschrieben wurden, bevor Sie mit der anderen Verarbeitung fortfahren können.

Genauer gesagt verwendet IBM WebSphere MQ auf UNIX -Plattformen die Option `O_SYNC` zum Öffnen und den `fsync()` -Systemaufruf, um Schreibvorgänge auf wiederherstellbare Datenträger explizit zu erzwingen, und ist davon abhängig, dass diese Optionen ordnungsgemäß funktionieren.



Achtung: Linux Sie sollten das Dateisystem mit der Option `async` anhängen, die weiterhin die Option synchroner Schreibvorgänge unterstützt und eine bessere Leistung bietet als die Option `sync`.

Beachten Sie jedoch, dass Sie, wenn das Dateisystem aus Linux exportiert wurde, weiterhin das Dateisystem mit der Option `sync` exportieren müssen.

2. Garantiert exklusiver Zugriff auf Dateien

Damit mehrere Warteschlangenmanager synchronisiert werden können, muss ein Mechanismus für einen Warteschlangenmanager vorhanden sein, um eine exklusive Sperre für eine Datei zu erhalten.

3. Release-Sperren bei einem Ausfall

Wenn ein Warteschlangenmanager ausfällt oder wenn ein Kommunikationsfehler mit dem Dateisystem vorliegt, müssen die vom Warteschlangenmanager gesperrten Dateien entsperrt und anderen Prozessen zur Verfügung gestellt werden, ohne zu warten, dass der Warteschlangenmanager erneut mit dem Dateisystem verbunden wird.

Damit IBM WebSphere MQ zuverlässig funktioniert, muss ein gemeinsam genutztes Dateisystem diese Anforderungen erfüllen. Ist dies nicht der Fall, werden die Daten und Protokolle des Warteschlangenma-

nagers beschädigt, wenn das gemeinsam genutzte Dateisystem in einer Multi-Instanz-WS-Manager-Konfiguration verwendet wird.

Für WS-Manager mit mehreren Instanzen unter Microsoft Windows muss auf den Netzspeicher durch das CIFS-Protokoll (Common Internet File System) zugegriffen werden, das von Microsoft Windows -Netzen verwendet wird. Der CIFS-Client (Common Internet File System) erfüllt nicht die Anforderungen von IBM WebSphere MQ für die Sperrsemantik auf anderen Plattformen als Microsoft Windows. Daher dürfen Warteschlangenmanager mit mehreren Instanzen, die auf anderen Plattformen als Microsoft Windows ausgeführt werden, das Common Internet File System (CIFS) nicht als gemeinsam genutztes Dateisystem verwenden.

Für WS-Manager mit mehreren Instanzen auf anderen unterstützten Plattformen muss auf den Speicher durch ein Netzdateisystemprotokoll zugegriffen werden, das mit der Position "Posix-konform" kompatibel ist, und unterstützt die lease-basierte Sperrung. Moderne Dateisysteme wie Network File System (NFS) Version 4 verwenden geleaste Sperren, um Fehler zu erkennen und nach einem Fehler Sperren freizugeben. Ältere Dateisysteme wie Network File System Version 3, die keinen zuverlässigen Mechanismus zum Freigeben von Sperren nach einem Fehler haben, dürfen nicht für Warteschlangenmanager mit mehreren Instanzen verwendet werden.

Überprüfung der Anforderungen an das gemeinsam genutzte Dateisystem

Sie müssen überprüfen, ob das gemeinsam genutzte Dateisystem, das Sie verwenden möchten, diese Anforderungen erfüllt. Außerdem müssen Sie überprüfen, ob das Dateisystem ordnungsgemäß für die Zuverlässigkeit konfiguriert ist. Gemeinsam genutzte Dateisysteme bieten manchmal Konfigurationsoptionen, um die Leistung auf Kosten der Zuverlässigkeit zu verbessern.

Unter normalen Umständen funktioniert IBM WebSphere MQ ordnungsgemäß mit dem Attributcaching, und es ist nicht erforderlich, das Caching zu inaktivieren, z. B. indem Sie NOAC auf einem NFS-Mount festlegen. Das Attributcaching kann Probleme verursachen, wenn mehrere Dateisystemclients für Schreibzugriff auf dieselbe Datei auf dem Dateisystemserver contendieren, da die zwischengespeicherten Attribute, die von den einzelnen Clients verwendet werden, möglicherweise nicht mit den Attributen auf dem Server identisch sind. Ein Beispiel für Dateien, auf die auf diese Weise zugegriffen wird, sind WS-Manager-Fehlerprotokolle für einen Multi-Instanz-Warteschlangenmanager. Die WS-Manager-Fehlerprotokolle können sowohl durch eine aktive als auch durch eine Standby-Warteschlangenmanagerinstanz geschrieben werden, und die Attribute der Cachedatei können dazu führen, dass die Fehlerprotokolle größer werden als erwartet, bevor die Rollover der Dateien auftreten.

Führen Sie die Task „[Verhalten eines gemeinsam genutzten Dateisystems überprüfen](#)“ auf Seite 139 aus, um das Dateisystem zu überprüfen. Diese Task prüft, ob das gemeinsam genutzte Dateisystem die Anforderungen 2 und 3 erfüllt. Sie müssen die Anforderung 1 in der Dokumentation des gemeinsam genutzten Dateisystems prüfen oder indem Sie mit Protokolldaten auf der Platte experimentieren.

Plattenfehler können beim Schreiben auf Platte zu Fehlern führen, die IBM WebSphere MQ als Erfassung von Fehlerdaten beim ersten Auftreten (First Failure Data Capture) meldet. Sie können das Dateisystemprüfprogramm für Ihr Betriebssystem ausführen, um das gemeinsam genutzte Dateisystem auf Plattenfehler zu überprüfen. Auf UNIX -und Linux -Plattformen heißt das Dateisystemprüfprogramm beispielsweise fsck. Auf Windows -Plattformen wird das Dateisystemprüfprogramm als CHKDSK oder SCANDISK bezeichnet.

Sicherheit des NFS-Servers

Anmerkung: Auf einen Network File System (NFS)-Server sollten Sie nur Warteschlangenmanagerdaten einstellen. 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.

Verhalten eines gemeinsam genutzten Dateisystems überprüfen

Führen Sie **amqmfsc** aus, um zu überprüfen, ob ein gemeinsam genutztes Dateisystem auf UNIX -Systemen die Anforderungen zum Speichern der Warteschlangenmanagerdaten eines Multi-Instanz-Warteschlangenmanagers erfüllt. Führen Sie das IBM WebSphere MQ MQI client -Beispielprogramm **amqsfhac** parallel zu **amqmfsc** aus, um zu demonstrieren, dass ein Warteschlangenmanager die Nachrichtenintegrität während eines Fehlers beibehält.

Vorbereitende Schritte

Sie benötigen einen Server mit Netzspeicher und zwei weitere mit ihm verbundene Server, auf denen WebSphere MQ installiert ist. Sie müssen über Administratorberechtigung (Root) verfügen, um das Dateisystem zu konfigurieren, und ein WebSphere MQ -Administrator sein, um **amqmfsc** ausführen zu können.

Informationen zu diesem Vorgang

Unter „[Voraussetzungen für gemeinsam genutzte Dateisysteme](#)“ auf Seite 137 sind die Dateisystemanforderung für die Verwendung eines gemeinsam genutzten Dateisystems mit Multi-Instanz-Warteschlangenmanagern beschrieben. In der IBM WebSphere MQ Technote [Testing and support statement for WebSphere MQ multi-instance queue managers](#) sind die gemeinsam genutzten Dateisysteme aufgelistet, mit denen IBM bereits getestet wurde. Die Prozedur in dieser Task beschreibt, wie Sie ein Dateisystem testen, um zu bewerten, ob ein nicht aufgelistete Dateisysteme die Datenintegrität aufrecht erhalten.

Der Failover eines Multi-Instanz-WS-Managers kann durch Hardware- oder Softwarefehler ausgelöst werden, einschließlich Netzproblemen, die verhindern, dass der WS-Manager in seine Daten oder Protokoll-dateien schreibt. Hauptsächlich sind Sie daran interessiert, Fehler auf dem Dateiserver zu verursachen. Aber Sie müssen auch Fehler auf den IBM WebSphere MQ-Servern verursachen, um erfolgreich freigegebene Sperrungen zu testen. Damit Sie in einem gemeinsam genutzten Dateisystem vertrauen können, testen Sie alle folgenden Fehler und alle anderen Fehler, die für Ihre Umgebung spezifisch sind:

1. Das Betriebssystem auf dem Dateiserver herunterfahren, einschließlich der Synchronisierung der Platten.
2. Das Betriebssystem auf dem Dateiserver anhalten, ohne die Platten zu synchronisieren.
3. Drücken Sie die Grundstellungsschaltfläche auf jedem der Server.
4. Ausziehen des Netzkabels aus jedem der Server.
5. Ziehen Sie das Netzkabel aus jedem der Server heraus.
6. Schalten Sie die einzelnen Server aus.

Erstellen Sie das Verzeichnis im Netzspeicher, den Sie für die gemeinsame Nutzung von WS-Manager-Daten und -Protokollen verwenden werden. Der Verzeichniseigner muss ein WebSphere MQ -Administrator oder ein Mitglied der Gruppe mqm unter UNIX sein. Der Benutzer, der die Tests ausführt, benötigt die WebSphere MQ -Administratorberechtigung.

Verwenden Sie als Unterstützung beim Konfigurieren des Dateisystems das Beispiel zum Exportieren und Anhängen eines Dateisystems in [Multi-Instanz-Warteschlangenmanager unter Linux erstellen](#) . Unterschiedliche Dateisysteme erfordern unterschiedliche Konfigurationsschritte. Lesen Sie die Dokumentation zum Dateisystem.

Vorgehensweise

Bei jedem der Prüfungen führen Sie alle Fehler in der vorherigen Liste durch, während die Dateisystemprüffunktion ausgeführt wird. Wenn Sie **amqsfhac** gleichzeitig mit **amqmfscck** ausführen möchten, müssen Sie die Task „amqsfhac zum Testen der Nachrichtenintegrität ausführen“ auf Seite 144 parallel mit dieser Task ausführen.

1. Hängen Sie das exportierte Verzeichnis auf den beiden IBM WebSphere MQ -Servern an.

Erstellen Sie auf dem Dateisystemserver ein gemeinsam genutztes Verzeichnis `shared` und ein Unterverzeichnis zum Speichern der Daten für Multi-Instanz-Warteschlangenmanager, `qmdata`. Ein Beispiel für die Einrichtung eines gemeinsam genutzten Verzeichnisses für Multi-Instanz-Warteschlangenmanager unter Linux finden Sie unter [Beispiel](#) im Abschnitt [Multi-Instanz-Warteschlangenmanager unter Linux erstellen](#).

2. Überprüfen Sie das Verhalten des Basisdateisystems.

Führen Sie den Dateisystemprüfprogramm auf einem IBM WebSphere MQ -Server ohne Parameter aus.

```
amqmfscck /shared/qmdata
```

Abbildung 40. Auf IBM WebSphere MQ -Server 1

3. Überprüfen Sie gleichzeitig, ob von beiden IBM WebSphere MQ -Servern in dasselbe Verzeichnis geschrieben wird.

Führen Sie auf beiden IBM WebSphere MQ -Servern das Dateisystemprüfprogramm gleichzeitig mit der Option `-c` aus.

```
amqmfscck -c /shared/qmdata
```

Abbildung 41. Auf IBM WebSphere MQ -Server 1

```
amqmfscck -c /shared/qmdata
```

Abbildung 42. Auf IBM WebSphere MQ -Server 2

4. Überprüfen Sie, ob Sperren auf beiden IBM WebSphere MQ -Servern vorhanden sind und freigeben.

Führen Sie auf beiden IBM WebSphere MQ -Servern gleichzeitig das Dateisystemprüfprogramm mit der Option `-w` aus.

```
amqmfscck -w /shared/qmdata
```

Abbildung 43. Auf IBM WebSphere MQ -Server 1

```
amqmfscck -w /shared/qmdata
```

Abbildung 44. Auf IBM WebSphere MQ -Server 2

5. Überprüfen Sie die Datenintegrität.

- a) Formatieren Sie die Testdatei.

Erstellen Sie eine große Datei in dem Verzeichnis, das getestet wird. Die Datei wird so formatiert, dass die nachfolgenden Phasen erfolgreich abgeschlossen werden können. Die Datei muss groß genug sein, dass genügend Zeit vorhanden ist, um die zweite Phase zu unterbrechen, um die Funktionsübernahme zu simulieren. Versuchen Sie, den Standardwert von 262144 Seiten (1 GB) zu verwenden. Das Programm reduziert diese Standardeinstellung bei langsamen Dateisystemen automatisch, so dass die Formatierung in ca. 60 Sekunden abgeschlossen wird.

```
amqmfscck -f /shared/qmdata
```

Der Server antwortet mit den folgenden Nachrichten:

```
Formatting test file for data integrity test.
Test file formatted with 262144 pages of data.
```

Abbildung 45. Auf IBM WebSphere MQ -Server 1

b) Schreiben Sie Daten mit Hilfe des Dateisystemprüfers in die Testdatei, und verursachen Sie einen Fehler.

Führen Sie das Testprogramm auf zwei Servern zur gleichen Zeit aus. Starten Sie das Testprogramm auf dem Server, auf dem der Fehler auftreten wird, und starten Sie dann das Testprogramm auf dem Server, das den Fehler überleben wird. Ursache des Fehlers, den Sie untersuchen.

Das erste Testprogramm stoppt mit einer Fehlermeldung. Das zweite Testprogramm ruft die Sperre für die Testdatei ab und schreibt Daten in die Testdatei, in der das erste Testprogramm abgelassen wurde. Lassen Sie das zweite Testprogramm zum Abschluss führen.

Tabelle 12. Datenintegritätsprüfung auf zwei Servern zur gleichen Zeit ausführen

IBM WebSphere MQ Server 1	IBM WebSphere MQ Server 2
amqmfscck -a /shared/qmdata	
Please start this program on a second machine with the same parameters. File lock acquired. Start a second copy of this program with the same parameters on another server. Writing data into test file. To increase the effectiveness of the test, interrupt the writing by ending the process, temporarily breaking the network connection to the networked storage, rebooting the server or turning off the power. To increase the effectiveness of the test, interrupt the writing by ending the process, temporarily breaking the network connection to the networked storage, rebooting the server or turning off the power.	amqmfscck -a /shared/qmdata Waiting for lock... Waiting for lock... Waiting for lock... Waiting for lock... Waiting for lock... Waiting for lock...
Turn the power off here	

Tabelle 12. Datenintegritätsprüfung auf zwei Servern zur gleichen Zeit ausführen (Forts.)	
IBM WebSphere MQ Server 1	IBM WebSphere MQ Server 2
	<pre>File lock acquired. Reading test file Checking the integrity of the data read. Appending data into the test file after data already found. The test file is full of data. It is ready to be inspected for data integrity.</pre>

Der Zeitpunkt des Tests richtet sich nach dem Verhalten des Dateisystems. Beispielsweise dauert es in der Regel 30 bis 90 Sekunden, wenn ein Dateisystem die Dateisperren freigibt, die durch das erste Programm nach einem Stromausfall erhalten wurden. Wenn Sie zu wenig Zeit haben, um den Fehler einzuführen, bevor das erste Testprogramm die Datei gefüllt hat, verwenden Sie die Option -x von **amqmfscck**, um die Testdatei zu löschen. Testen Sie den Test ab dem Start mit einer größeren Testdatei.

c) Überprüfen Sie die Integrität der Daten in der Testdatei.

```
amqmfscck -i /shared/qmdata
```

Der Server antwortet mit den folgenden Nachrichten:

```
File lock acquired
Reading test file checking the integrity of the data read.
The data read was consistent.
The tests on the directory completed successfully.
```

Abbildung 46. Auf IBM WebSphere MQ -Server 2

6. Löschen Sie die Testdateien.

```
amqmfscck -x /shared/qmdata
Test files deleted.
```

Abbildung 47. Auf IBM WebSphere MQ -Server 2

Der Server antwortet mit der Nachricht:

```
Test files deleted.
```

Ergebnisse

Das Programm gibt den Exit-Code 0 zurück, wenn die Tests erfolgreich abgeschlossen wurden, und andernfalls nicht null.

Beispiele

Die erste Gruppe von drei Beispielen zeigt den Befehl, der die minimale Ausgabe erzeugt.

Erfolgreicher Test der Basisdateispeerrung auf einem Server

```
> amqmfscck /shared/qmdata
The tests on the directory completed successfully.
```

Fehlgeschlagener Test der Basisdateisperrung auf einem Server

```
> amqmfscck /shared/qmdata
AMQ6245: Error Calling 'write()[2]' on file '/shared/qmdata/amqmfscck.lck' error '2'.
```

Erfolgreicher Test der Sperrung auf zwei Servern

Tabelle 13. Erfolgreiches Sperren auf zwei Servern	
IBM WebSphere MQ Server 1	IBM WebSphere MQ Server 2
<pre>> amqmfscck -w /shared/qmdata Please start this program on a second machine with the same parameters. Lock acquired. Press Return or terminate the program to release the lock.</pre>	
	<pre>> amqmfscck -w /shared/qmdata Waiting for lock...</pre>
<pre>[Return pressed] Lock released.</pre>	
	<pre>Lock acquired. The tests on the directory completed successfuly</pre>

In der zweiten Gruppe von drei Beispielen werden dieselben Befehle im ausführlichen Modus angezeigt.

Erfolgreicher Test der Basisdateisperrung auf einem Server

```
> amqmfscck -v /shared/qmdata
System call: stat("/shared/qmdata")'
System call: fd = open("/shared/qmdata/amqmfscck.lck", O_RDWR, 0666)
System call: fchmod(fd, 0666)
System call: fstat(fd)
System call: fcntl(fd, F_SETLK, F_WRLCK)
System call: write(fd)
System call: close(fd)
System call: fd = open("/shared/qmdata/amqmfscck.lck", O_RDWR, 0666)
System call: fcntl(fd, F_SETLK, F_WRLCK)
System call: close(fd)
System call: fd1 = open("/shared/qmdata/amqmfscck.lck", O_RDWR, 0666)
System call: fcntl(fd1, F_SETLK, F_RDLCK)
System call: fd2 = open("/shared/qmdata/amqmfscck.lck", O_RDWR, 0666)
System call: fcntl(fd2, F_SETLK, F_RDLCK)
System call: close(fd2)
System call: write(fd1)
System call: close(fd1)
The tests on the directory completed successfully.
```

Fehlgeschlagener Test der Basisdateisperrung auf einem Server

```
> amqmfscck -v /shared/qmdata
System call: stat("/shared/qmdata")
System call: fd = open("/shared/qmdata/amqmfscck.lck", O_RDWR, 0666)
System call: fchmod(fd, 0666)
System call: fstat(fd)
System call: fcntl(fd, F_SETLK, F_WRLCK)
System call: write(fd)
System call: close(fd)
System call: fd = open("/shared/qmdata/amqmfscck.lck", O_RDWR, 0666)
System call: fcntl(fd, F_SETLK, F_WRLCK)
System call: close(fd)
System call: fd = open("/shared/qmdata/amqmfscck.lck", O_RDWR, 0666)
System call: fcntl(fd, F_SETLK, F_RDLCK)
System call: fdSameFile = open("/shared/qmdata/amqmfscck.lck", O_RDWR, 0666)
System call: fcntl(fdSameFile, F_SETLK, F_RDLCK)
System call: close(fdSameFile)
System call: write(fd)
```

```
AMQxxxx: Error calling 'write()[2]' on file '/shared/qmdata/amqmfsc.lck', errno 2
(Permission denied).
```

Erfolgreicher Test der Sperrung auf zwei Servern

Tabelle 14. Erfolgreiches Sperren auf zwei Servern-Modus 'verbose'	
IBM WebSphere MQ Server 1	IBM WebSphere MQ Server 2
<pre>> amqmfsc -wv /shared/qmdata Calling 'stat("/shared/qmdata")' Calling 'fd = open("/shared/qmda ta/amqmfsc.lkw", O_EXCL O_CREAT O_RDWR, 0666)' Calling 'fchmod(fd, 0666)' Calling 'fstat(fd)' Please start this program on a second machine with the same parameters. Calling 'fcntl(fd, F_SETLK, F_WRLCK)' Lock acquired. Press Return or terminate the program to release the lock.</pre>	
	<pre>> amqmfsc -wv /shared/qmdata Calling 'stat("/shared/qmdata")' Calling 'fd = open("/shared/qmda ta/amqmfsc.lkw", O_EXCL O_CREAT O_RDWR,0666)' Calling 'fd = open("/shared/qmdata/amqmfsc.lkw, O_RDWR, 0666)' Calling 'fcntl(fd, F_SETLK, F_WRLCK)' 'Waiting for lock...</pre>
<pre>[Return pressed] Calling 'close(fd)' Lock released.</pre>	
	<pre>Calling 'fcntl(fd, F_SETLK, F_WRLCK)' Lock acquired. The tests on the directory completed successfuly</pre>

Zugehörige Verweise

amqmfsc (Dateisystemprüfung)

amqsfhac zum Testen der Nachrichtenintegrität ausführen

amqsfhac prüft, ob ein Warteschlangenmanager, der einen vernetzten Speicher verwendet, nach einem Fehler die Nachrichtenintegrität beibehält.

Vorbereitende Schritte

Für diesen Test benötigen Sie vier Server. Zwei Server für den Warteschlangenmanager mit mehreren Instanzen, einer für das Dateisystem und einer für die Ausführung von **amqsfhac** als IBM WebSphere MQ MQI client -Anwendung.

Führen Sie Schritt „1“ auf Seite 140 unter Vorgehensweise aus, um das Dateisystem für einen Multi-Instanz-Warteschlangenmanager einzurichten.

Informationen zu diesem Vorgang

Vorgehensweise

1. Erstellen Sie auf einem anderen Server einen Multi-Instanz-Warteschlangenmanager mit dem Namen QM1. Verwenden Sie dabei das in Schritt „1“ auf Seite 140 unter [Vorgehensweise](#) erstellte Dateisystem.

Siehe [Multi-Instanz-WS-Manager erstellen](#) .

2. Starten Sie den Warteschlangenmanager auf beiden Servern, die ihn hoch verfügbar machen.

Auf Server 1:

```
strmqm -x QM1
```

Auf Server 2:

```
strmqm -x QM1
```

3. Richten Sie die Clientverbindung für die Ausführung von **amqsfhac** ein.
 - a) Verwenden Sie die Prozedur im Abschnitt [Clientinstallation überprüfen](#) , um eine Clientverbindung einzurichten, oder die Beispielscripts im Abschnitt [Clientbeispiele für Wiederherstellung der Verbindung](#) .
 - b) Ändern Sie den Clientkanal so, dass zwei IP-Adressen vorhanden sind, die den beiden Servern entsprechen, auf der QM1 ausgeführt wird.

Ändern Sie im Beispielscript Folgendes:

```
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +  
CONNAME('LOCALHOST(2345)') QMNAME(QM1) REPLACE
```

In:

```
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +  
CONNAME('server1(2345),server2(2345)') QMNAME(QM1) REPLACE
```

Dabei sind `server1` und `server2` die Hostnamen der beiden Server, und 2345 ist der Port, an dem der Kanal-Listener empfangsbereit ist. Gewöhnlich ist dies der Standardwert 1414. Sie können 1414 mit der Standard-Listener-Konfiguration verwenden.

4. Erstellen Sie zwei lokale Warteschlangen unter QM1 für den Test.
Führen Sie das folgende MQSC-Script aus:

```
DEFINE QLOCAL(TARGETQ) REPLACE  
DEFINE QLOCAL(SIDEQ) REPLACE
```

5. Testen Sie die Konfiguration mit **amqsfhac**.

```
amqsfhac QM1 TARGETQ SIDEQ 2 2 2
```

6. Testen Sie die Nachrichtenintegrität, während Sie die Integrität des Dateisystems testen.

Führen Sie **amqsfhac** während Schritt „5“ auf Seite 140 der [Prozeduraus](#).

```
amqsfhac QM1 TARGETQ SIDEQ 10 20 0
```

Wenn Sie die aktive Warteschlangenmanagerinstanz stoppen, stellt **amqsfhac** die Verbindung zur anderen Warteschlangenmanagerinstanz wieder her, sobald sie aktiv geworden ist. Starten Sie die gestoppte WS-Manager-Instanz erneut, so dass Sie den Fehler beim nächsten Test rückgängig machen können. Sie müssen wahrscheinlich die Anzahl der Iterationen auf der Basis des Experiments mit Ihrer Umgebung erhöhen, damit das Testprogramm genügend Zeit für die Übernahme von Failover ausgeführt wird.

Ergebnisse

Ein Beispiel für die Ausführung von **amqsfhac** in Schritt „6“ auf Seite 145 ist in [Abbildung 48](#) auf Seite 146 dargestellt. Der Test ist ein Erfolg.

Wenn der Test ein Problem festgestellt hat, würde die Ausgabe den Fehler melden. In einigen Testläufen kann MQRC_CALL_INTERRUPTED möglicherweise "Resolving to backed out" melden. Es macht keinen Unterschied zum Ergebnis. Das Ergebnis hängt davon ab, ob der Schreibzugriff auf die Platte durch den Netzdateispeicher vor oder nach dem Fehlschlagen der Platte festgeschrieben wurde.

```
Sample AMQSFHAC start
qmname = QM1
qname = TARGETQ
sidename = SIDEQ
transize = 10
iterations = 20
verbose = 0
Iteration 0
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Iteration 6
Resolving MQRC_CALL_INTERRUPTED
MQGET browse side tranid=14 pSideinfo->tranid=14
Resolving to committed
Iteration 7
Iteration 8
Iteration 9
Iteration 10
Iteration 11
Iteration 12
Iteration 13
Iteration 14
Iteration 15
Iteration 16
Iteration 17
Iteration 18
Iteration 19
Sample AMQSFHAC end
```

*Abbildung 48. Ausgabe einer erfolgreichen Ausführung von **amqsfhac***

Zugehörige Verweise

[Beispielprogramme zur Hochverfügbarkeit](#)

IBM WebSphere MQ -Dateien teilen

Auf einige IBM WebSphere MQ-Dateien wird ausschließlich über einen aktiven Warteschlangenmanager zugegriffen, während andere Dateien gemeinsam genutzt werden.

WebSphere MQ -Dateien werden in Programm- und Datendateien aufgeteilt. Programmdateien werden in der Regel lokal auf jedem Server installiert, auf dem WebSphere MQ ausgeführt wird. Warteschlangenmanager nutzen den Zugriff auf Datendateien und Verzeichnisse im Standarddatenverzeichnis gemeinsam. Sie benötigen exklusiven Zugriff auf die Verzeichnisstrukturen ihres eigenen Warteschlangenmanagers, die sich jeweils in den Verzeichnissen qmgrs und log, die in [Abbildung 49](#) auf Seite 147 dargestellt sind.

[Abbildung 49](#) auf Seite 147 ist eine übergeordnete Ansicht der Verzeichnisstruktur von WebSphere MQ. Sie zeigt die Verzeichnisse an, die von den WS-Managern gemeinsam genutzt werden können und die fern ausgeführt werden können. Die Details variieren je nach Plattform. Die gepunkteten Linien geben konfigurierbare Pfade an.

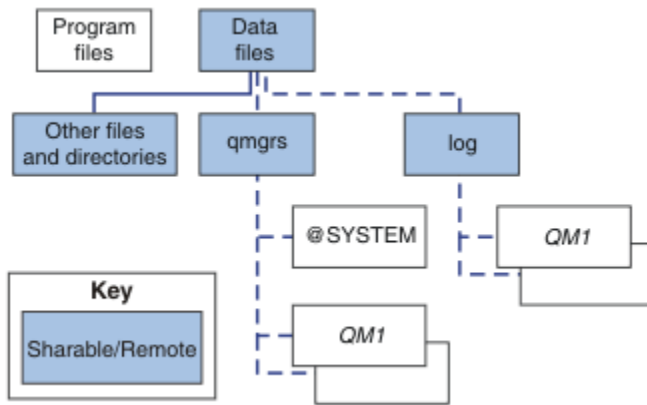


Abbildung 49. Gesamtansicht der Verzeichnisstruktur von WebSphere MQ

Programm- dateien

Das Verzeichnis der Programmdateien wird in der Regel an der Standardposition belassen, ist lokal und wird von allen WS-Managern auf dem Server gemeinsam genutzt.

Daten- dateien

Das Verzeichnis für Datendateien befindet sich normalerweise lokal an der Standardposition /var/mqm auf UNIX and Linux -Systemen und kann bei der Installation unter Windows konfiguriert werden. Sie wird von WS-Managern gemeinsam genutzt. Sie können die Standardposition als ferne Position definieren, sie jedoch nicht von verschiedenen Installationen von WebSphere MQ gemeinsam nutzen. Das Attribut `DefaultPrefix` in der Konfiguration von WebSphere MQ verweist auf diesen Pfad.

qmgrs

Ab Version 7.0.1 gibt es zwei alternative Möglichkeiten, die Position der WS-Manager-Daten anzugeben.

Prefix verwenden

Das Attribut `Prefix` gibt die Position des `qmgrs`-Verzeichnisses an. WebSphere MQ erstellt den Namen des Warteschlangenmanagerverzeichnisses aus dem Namen des Warteschlangenmanagers und erstellt ihn als Unterverzeichnis des Verzeichnisses `qmgrs`.

Das Attribut `Prefix` befindet sich in der Zeilengruppe 'QueueManager' und wird von dem Wert im Attribut `DefaultPrefix` übernommen. Standardmäßig verwenden die Warteschlangenmanager standardmäßig dasselbe `qmgrs`-Verzeichnis, weil sie sich auf die Verwaltung von Verwaltungsaufgaben nicht teilen.

Die Zeilengruppe `QueueManager` befindet sich in der Datei `mq.ini`.

Wenn Sie die Position des Verzeichnisses `qmgrs` für einen WS-Manager ändern, müssen Sie den Wert des Attributs `Prefix` ändern.

Das Attribut `Prefix` für das `QM1`-Verzeichnis in [Abbildung 49 auf Seite 147](#) für eine UNIX and Linux -Plattform ist,

```
Prefix=/var/mqm
```

DataPath verwenden

Das Attribut `DataPath` gibt die Position des Datenverzeichnisses des Warteschlangenmanagers an.

Das Attribut `DataPath` gibt den vollständigen Pfad an, einschließlich des Namens des Datenverzeichnisses des Warteschlangenmanagers. Das Attribut `DataPath` entspricht nicht dem Attribut `Prefix`, das einen unvollständigen Pfad zum Datenverzeichnis des Warteschlangenmanagers angibt.

Das Attribut `DataPath` befindet sich in der Zeilengruppe 'QueueManager', wenn es angegeben ist. Wenn sie angegeben wurde, hat sie Vorrang vor jedem Wert im Attribut `Prefix`.

Die Zeilengruppe `QueueManager` befindet sich in der Datei `mqs.ini`.

Wenn Sie die Position des Datenverzeichnisses des Warteschlangenmanagers für einen WS-Manager ändern, müssen Sie den Wert des Attributs `DataPath` ändern.

Das Attribut `DataPath` für das Verzeichnis QM1 in [Abbildung 49 auf Seite 147](#) für eine Plattform von UNIX oder Linux ist,

```
DataPath=/var/mqm/qmgrs/QM1
```

log

Das Protokollverzeichnis wird für jeden Warteschlangenmanager in der Zeilengruppe `Log` in der WS-Manager-Konfiguration separat angegeben. Die WS-Manager-Konfiguration befindet sich in `qm.ini`.

DataPath/QmgrName/@IPCC -Unterverzeichnisse

Die Unterverzeichnisse von `DataPath/QmgrName/@IPCC` befinden sich im gemeinsam genutzten Verzeichnispfad. Sie werden verwendet, um den Verzeichnispfad für IPC-Dateisystemobjekte zu erstellen. Sie müssen den Namensbereich eines Warteschlangenmanagers unterscheiden, wenn ein Warteschlangenmanager von mehreren Systemen gemeinsam genutzt wird. Vor Version 7.0.1 wurde ein WS-Manager nur auf einem System verwendet. Eine Gruppe von Unterverzeichnissen war ausreichend, um den Verzeichnispfad für IPC-Dateisystemobjekte zu definieren (siehe [Abbildung 50 auf Seite 148](#)).

```
DataPath/QmgrName/@IPCC/esem
```

Abbildung 50. Beispiel für das IPC-Unterverzeichnis vor Version 7.0.1

In Version 7.0.1 und höher müssen die IPC-Dateisystemobjekte vom System unterschieden werden. Ein Unterverzeichnis für jedes System, auf dem der Warteschlangenmanager ausgeführt wird, wird dem Verzeichnispfad hinzugefügt (siehe [Abbildung 51 auf Seite 148](#)).

```
DataPath/QmgrName/@IPCC/esem/myHostName/
```

Abbildung 51. Beispiel für das IPC-Unterverzeichnis V7.0.1 und nachfolgende Releases

`myHostName` ist bis zu den ersten 20 Zeichen des Hostnamens, der vom Betriebssystem zurückgegeben wird. Auf einigen Systemen kann der Hostname bis zu 64 Zeichen lang sein, bevor er abgeschnitten wird. Der generierte Wert von `myHostName` kann aus zwei Gründen ein Problem verursachen:

1. Die ersten 20 Zeichen sind nicht eindeutig.
2. Der Hostname wird von einem DHCP-Algorithmus generiert, der nicht immer denselben Hostnamen einem System zuordnet.

Legen Sie in diesen Fällen `myHostName` mithilfe der Umgebungsvariablen `MQC_IPC_HOST` fest (siehe [Abbildung 52 auf Seite 148](#)).

```
export MQC_IPC_HOST=myHostName
```

Abbildung 52. Beispiel: MQC_IPC_HOST festlegen

Andere Dateien und Verzeichnisse

Andere Dateien und Verzeichnisse, wie z. B. das Verzeichnis mit den Tracedateien und das allgemeine Fehlerprotokoll, werden normalerweise gemeinsam genutzt und auf dem lokalen Dateisystem gespeichert.

Bis v7.0.1 stützte sich WebSphere MQ auf die externe Verwaltung, um Warteschlangenmanagern exklusiven Zugriff auf die Warteschlangenmanager-Daten und -Protokolldateien zu garantieren. Ab v7.0.1 verwaltet WebSphere MQ mit Unterstützung gemeinsam genutzter Dateisysteme den exklusiven Zugriff

auf diese Dateien über Dateisystemsperrern. Eine Dateisystemsperrere erlaubt es nur einer Instanz eines bestimmten Warteschlangenmanagers, aktiv zu sein.

Wenn Sie die erste Instanz eines bestimmten Warteschlangenmanagers starten, wird das Eigentumsrecht an dem Warteschlangenmanager-Verzeichnis des Warteschlangenmanagers angezeigt. Wenn Sie eine zweite Instanz starten, kann sie nur dann das Eigentumsrecht übernehmen, wenn die erste Instanz gestoppt wurde. Wenn der erste Warteschlangenmanager noch aktiv ist, kann die zweite Instanz nicht gestartet werden, und es wird gemeldet, dass der Warteschlangenmanager an anderer Stelle ausgeführt wird. Wenn der erste Warteschlangenmanager gestoppt wurde, übernimmt der zweite Warteschlangenmanager das Eigentumsrecht an den WS-Manager-Dateien und wird zum aktiven Warteschlangenmanager.

Sie können die Prozedur des zweiten Warteschlangenmanagers, der von der ersten übernommen wird, automatisieren. Starten Sie den ersten Warteschlangenmanager mit der Option `strmqm -x`, die es einem anderen Warteschlangenmanager ermöglicht, von ihm zu übernehmen. Der zweite WS-Manager wartet dann, bis die WS-Manager-Dateien entsperrt sind, bevor er versucht, das Eigentumsrecht an den WS-Manager-Dateien zu übernehmen, und startet.

Verzeichnisstruktur auf UNIX and Linux -Systemen

Die Verzeichnisstruktur WebSphere MQ auf UNIX and Linux -Systemen kann verschiedenen Dateisystemen zugeordnet werden, um die Verwaltung zu vereinfachen, die Leistung zu verbessern und die Zuverlässigkeit zu verbessern.

Verwenden Sie die flexible Verzeichnisstruktur von WebSphere MQ, um gemeinsam genutzte Dateisysteme für die Ausführung von Warteschlangenmanagern mit mehreren Instanzen zu nutzen.

Verwenden Sie den Befehl `crtmqm QM1`, um die in [Abbildung 53 auf Seite 150](#) gezeigte Verzeichnisstruktur zu erstellen, wobei R das Release des Produkts ist. Dies ist eine typische Verzeichnisstruktur für einen Warteschlangenmanager, der auf einem WebSphere MQ -System ab v7.0.1 erstellt wurde. Einige Verzeichnisse, Dateien und .ini-Attributeinstellungen werden aus Gründen der Übersichtlichkeit weggelassen, und ein anderer Name des WS-Managers kann durch das Mangeln geändert werden. Die Namen der Dateisysteme hängen von unterschiedlichen Systemen ab.

Bei einer Standardinstallation verweist jeder Warteschlangenmanager, den Sie erstellen, auf die allgemeinen `log` -und `qmgrs` -Verzeichnisse im lokalen Dateisystem. In einer Konfiguration mit mehreren Instanzen befinden sich die Verzeichnisse `log` und `qmgrs` in einem Netzdateisystem, das mit einer anderen Installation von WebSphere MQ gemeinsam genutzt wird.

[Abbildung 53 auf Seite 150](#) zeigt die Standardkonfiguration für WebSphere MQ v7.R unter AIX, wobei R das Release des Produkts ist. Beispiele für alternative Konfigurationen mit mehreren Instanzen finden Sie in [„Beispielverzeichnisstrukturen auf UNIX and Linux -Systemen“](#) auf Seite 154.

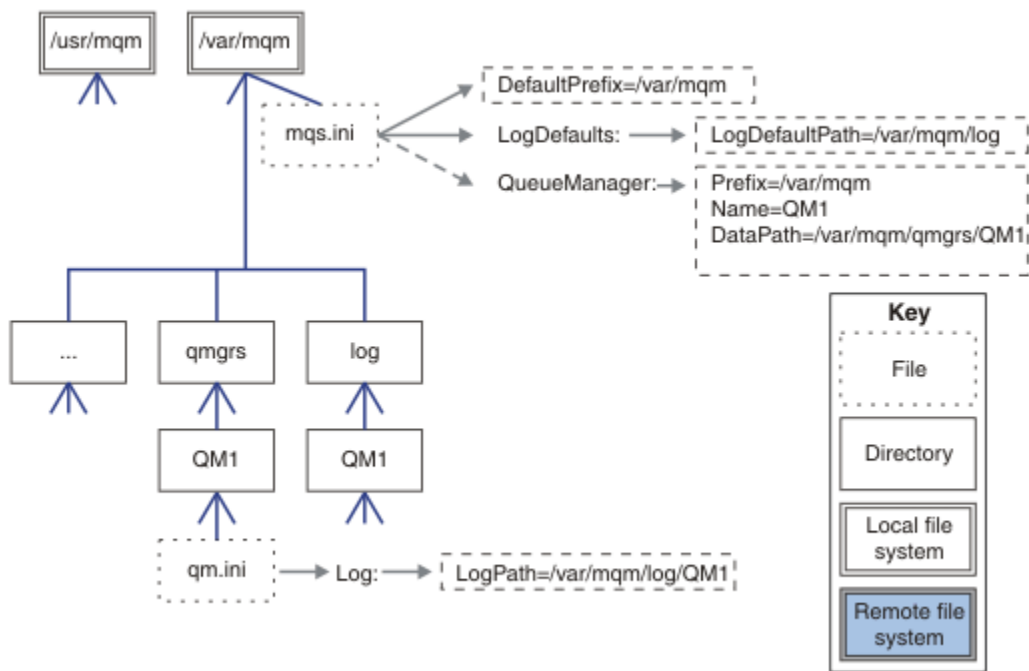


Abbildung 53. Beispiel für WebSphere MQ v7.R -Verzeichnisstruktur für UNIX and Linux -Systeme

Das Produkt wird standardmäßig in `/usr/mqm` unter AIX und `/opt/mqm` auf den anderen Systemen installiert. Die Arbeitsverzeichnisse werden in das Verzeichnis `/var/mqm` installiert.

Anmerkung: Wenn Sie das Dateisystem `/var/mqm` vor der Installation von IBM WebSphere MQ erstellt hatten, stellen Sie sicher, dass der Benutzer "mqm" über die vollständige Verzeichnisberechtigung verfügt, z. B. Dateimodus 755.

Die Verzeichnisse `log` und `qmgrs` werden in den Standardpositionen angezeigt, die durch die Standardwerte der Attribute `LogDefaultPath` und `DefaultPrefix` in der Datei `mqs.ini` definiert sind. Wenn ein Warteschlangenmanager erstellt wird, wird standardmäßig das Datenverzeichnis des Warteschlangenmanagers in `DefaultPrefix/qmgrs` und das Protokolldateiverzeichnis in `LogDefaultPath / log` erstellt. `LogDefaultPath` und `DefaultPrefix` wirken sich nur auf die Erstellung von Warteschlangenmanagern und Protokolldateien aus. Die tatsächliche Position eines WS-Manager-Verzeichnisses wird in der Datei `mqs.ini` gespeichert, die Position des Protokolldateiverzeichnisses wird in der Datei `qm.ini` gespeichert.

Das Protokolldateiverzeichnis für einen Warteschlangenmanager ist in der Datei `qm.ini` im Attribut `LogPath` definiert. Verwenden Sie die Option `-ld` im Befehl `crtmqm`, um das Attribut `LogPath` für einen Warteschlangenmanager festzulegen, z. B. `crtmqm -ld LogPath QM1`. Wenn Sie den Parameter `ld` nicht angeben, wird stattdessen der Wert von `LogDefaultPath` verwendet.

Das Datenverzeichnis des Warteschlangenmanagers wird im Attribut `DataPath` in der Zeilengruppe `QueueManager` in der Datei `mqs.ini` definiert. Verwenden Sie die Option `-md` im Befehl `crtmqm`, um `DataPath` für einen Warteschlangenmanager festzulegen, z. B. `crtmqm -md DataPath QM1`. Wenn Sie den Parameter `md` nicht angeben, wird stattdessen der Wert des Attributs `DefaultPrefix` oder `Prefix` verwendet. Präfix hat Vorrang vor `DefaultPrefix`.

In der Regel erstellen Sie `QM1`, indem Sie sowohl die Protokoll- als auch die Datenverzeichnisse in einem einzigen Befehl angeben.

```
crtmqm
-md DataPath -ld
LogPath QM1
```

Sie können die Position eines Warteschlangenmanagerprotokolls und Datenverzeichnisses eines vorhandenen Warteschlangenmanagers ändern, indem Sie die Attribute `DataPath` und `LogPath` in der `qm.ini`-Datei bearbeiten, wenn der Warteschlangenmanager gestoppt wird.

Der Pfad zum Verzeichnis `errors` ist wie die Pfade zu allen anderen Verzeichnissen in `/var/mqm` nicht änderbar. Die Verzeichnisse können jedoch auf verschiedenen Dateisystemen angehängt werden oder symbolisch mit verschiedenen Verzeichnissen verknüpft sein.

Verzeichnisinhalt auf UNIX and Linux -Systemen

Inhalt der Verzeichnisse, die einem WS-Manager zugeordnet sind.

Informationen zur Position der Produktdateien finden Sie unter Installationsposition auswählen.

Informationen zu alternativen Verzeichniskonfigurationen finden Sie unter „Dateisystemunterstützung planen“ auf Seite 136.

In Abbildung 54 auf Seite 152 ist das Layout repräsentativ für WebSphere MQ, nachdem ein WS-Manager seit einiger Zeit verwendet wurde. Die tatsächliche Struktur, von der Sie abhängig sind, hängt davon ab, welche Operationen auf dem Warteschlangenmanager ausgeführt wurden.

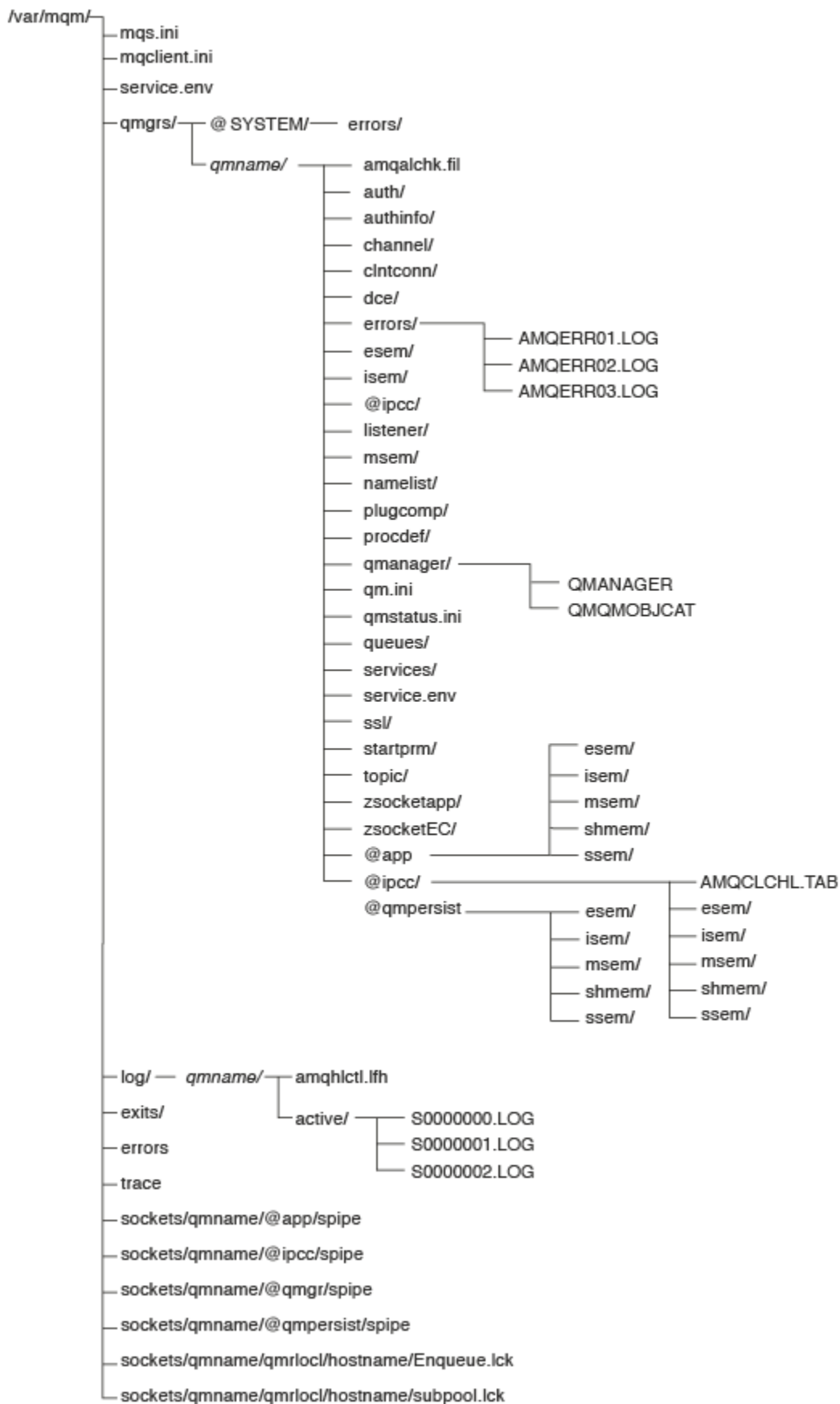


Abbildung 54. Standardverzeichnisstruktur (UNIX -Systeme), nachdem ein Warteschlangenmanager gestartet wurde

/var/mqm/

Das Verzeichnis `/var/mqm` enthält Konfigurationsdateien und Ausgabeverzeichnisse, die für eine WebSphere MQ -Installation als Ganzes und nicht für einen einzelnen Warteschlangenmanager gelten.

<u>mqs.ini</u>	WebSphere MQ -Konfigurationsdatei für die gesamte Installation; wird gelesen, wenn ein Warteschlangenmanager gestartet wird Der Dateipfad kann mit der Umgebungsvariablen AMQ_MQS_INI_LOCATION geändert werden. Stellen Sie sicher, dass dies in der Shell festgelegt und exportiert wird, in der der strmqm -Befehl ausgeführt wird.
<u>mqclient.ini</u>	Standardclientkonfigurationsdatei, die von WebSphere MQ MQI-Clientprogrammen gelesen wird. Der Dateipfad kann mit der Umgebungsvariablen MQCLNTCF geändert werden.
<u>service.env</u>	Enthält Umgebungsvariablen des Maschinenbereichs für einen Serviceprozess. Dateipfad wurde korrigiert.
<u>Fehler/</u>	Fehlerprotokolle für den Maschinengeltungsbereich und FFST-Dateien Verzeichnispfad wurde korrigiert. Siehe auch FFST: IBM WebSphere MQ für UNIX and Linux -Systeme.
<u>Sockets/</u>	Enthält nur Informationen zu jedem Warteschlangenmanager für die Systemverwendung.
<u>Trace/</u>	Tracedateien. Verzeichnispfad wurde korrigiert.
<u>exits/</u>	Standardverzeichnis, das Benutzerkanalexitprogramme enthält.
<u>exits64/</u>	Die Position kann in ApiExit-Zeilengruppen in der Datei 'mqs.ini' geändert werden.

/var/mqm/qmgrs/qmname/

`/var/mqm/qmgrs/qmname/` enthält Verzeichnisse und Dateien für einen Warteschlangenmanager. Das Verzeichnis ist für exklusiven Zugriff durch die aktive WS-Manager-Instanz gesperrt. Der Verzeichnispfad kann direkt in der Datei `mqs.ini` oder mithilfe der Option **md** des Befehls **crtmqm** geändert werden.

<u>qm.ini</u>	Warteschlangenmanagerkonfigurationsdatei, lesen, wenn ein Warteschlangenmanager gestartet wird.
<u>Fehler/</u>	Fehlerprotokolle des Warteschlangenmanagers. <code>qmname = @system</code> enthält kanalbezogene Nachrichten für einen unbekannt oder nicht verfügbaren Warteschlangenmanager.
<u>@ipcc/</u> <u>AMQCLCHL.TAB</u>	Standardsteuertabelle für Clientkanäle, die vom WebSphere MQ -Server erstellt und von WebSphere MQ MQI-Clientprogrammen gelesen wird. Der Dateipfad kann mit den Umgebungsvariablen MQCHLLIB und MQCHLTAB geändert werden.

Tabelle 16. Dokumentierter Inhalt des Verzeichnisses `/var/mqm/qmgrs/qmname` auf UNIX-Systemen (Forts.)

qmanager	WS-Manager-Objektdatei: QMANAGER Objektkatalog des WS-Managers: QMQMOBJCAT
authinfo/	Jedem Objekt, das im Warteschlangenmanager definiert ist, wird eine Datei in diesen Verzeichnissen zugeordnet. Der Dateiname entspricht ungefähr dem Definitionsnamen; siehe Informationen zu WebSphere MQ-Dateinamen .
Kanal/	
clntconn/	
Empfangsprogramm/	
namelist/	
procdef/	
Warteschlangen/	
Dienstleistungen/	
Themen/	
...	Andere Verzeichnisse, die von WebSphere MQ verwendet werden, wie z. B. @ipcc, die nur von WebSphere MQ geändert werden sollen

`/var/mqm/log/qmname/`

`/var/mqm/log/qmname/` enthält die WS-Manager-Protokolldateien. Das Verzeichnis ist für exklusiven Zugriff durch die aktive WS-Manager-Instanz gesperrt. Der Verzeichnispfad kann in der Datei `qm.ini` oder mithilfe der Option **ld** des Befehls **crtmqm** geändert werden.

Tabelle 17. Dokumentierter Inhalt des Verzeichnisses `/var/mqm/log/qmname` auf UNIX-Systemen

amqhlctl.lfh	Protokollsteuerdatei.
Aktiv/	Dieses Verzeichnis enthält die Protokolldateien S0000000.LOG, S0000001.LOG, S0000002.LOG und so weiter.

Beispielverzeichnisstrukturen auf UNIX and Linux-Systemen

Beispiele für alternative Dateisystemkonfigurationen auf UNIX and Linux-Systemen.

Sie können die Verzeichnisstruktur WebSphere MQ auf verschiedene Arten anpassen, um eine Reihe unterschiedlicher Ziele zu erreichen.

- Platzieren Sie die Verzeichnisse `qmgrs` und `log` auf fernen gemeinsam genutzten Dateisystemen, um einen WS-Manager mit mehreren Instanzen zu konfigurieren.
- Verwenden Sie separate Dateisysteme für die Daten- und Protokollverzeichnisse und ordnen Sie die Verzeichnisse verschiedenen Platten zu, um die Leistung zu verbessern, indem Sie die E/A-Konkurrenzsituationen verringern.
- Verwenden Sie schnellere Speichereinheiten für Verzeichnisse, die sich stärker auf die Leistung auswirken. Die Latenzzeit der physischen Einheit ist häufig ein wichtiger Faktor bei der Leistung des persistenten Messaging, als ob eine Einheit lokal oder über Remotezugriff angehängt ist. Die folgende Liste zeigt, welche Verzeichnisse die meisten und die leistungsfähigsten Verzeichnisse sind.

1. `log`
2. `qmgrs`
3. Andere Verzeichnisse, einschließlich `/usr/mqm`

- Erstellen Sie die Verzeichnisse `qmgrs` und `log` auf Dateisystemen, die einem Speicher mit guter Ausfallsicherheit zugeordnet sind, z. B. eine redundante Platteneinheit.
- Es ist besser, die allgemeinen Fehlerprotokolle lokal in `var/mqm/errors` und nicht in einem Netzdateisystem zu speichern, damit Fehler im Zusammenhang mit dem Netzdateisystem protokolliert werden können.

Abbildung 55 auf Seite 155 ist eine Vorlage, von der alternative WebSphere MQ -Verzeichnisstruktur abgeleitet werden. In der Schablone stellen gepunktete Linien Pfade dar, die konfiguriert werden können. In den Beispielen werden die gepunkteten Linien durch durchgezogene Linien ersetzt, die den Konfigurationsdaten entsprechen, die in der Umgebungsvariablen `AMQ_MQS_INI_LOCATION` und in den Dateien `mqs.ini` und `qm.ini` gespeichert sind.

Anmerkung: Die Pfadinformationen werden angezeigt, wenn sie in den Dateien `mqs.ini` oder `qm.ini` angezeigt werden. Wenn Sie im Befehl `crtmqm` Pfadparameter angeben, lassen Sie den Namen des Warteschlangenmanagerverzeichnisses weg: Der Name des Warteschlangenmanagers wird dem Pfad von WebSphere MQ hinzugefügt, nachdem er durch Mangled geändert wurde.

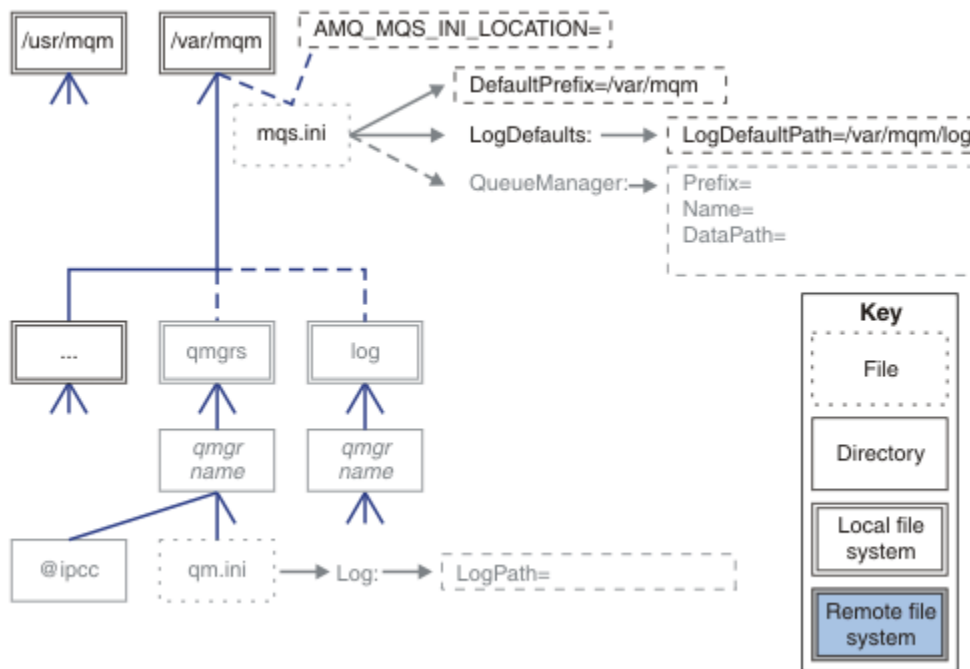


Abbildung 55. Vorlage für Verzeichnisstrukturmuster

Es folgen Beispiele für konfigurierte Verzeichnisstrukturen. Das erste Beispiel zeigt eine typische Standardverzeichnisstruktur für WebSphere MQ v7.0.1, die mit dem Befehl `crtmqm QM1` erstellt wurde. Das zweite Beispiel zeigt, wie eine typische Verzeichnisstruktur für einen Warteschlangenmanager dargestellt wird, der mit einem WebSphere MQ -Release vor v7.0.1 erstellt wurde. Die Verzeichnisstruktur ändert sich nicht.

Warteschlangenmanager, die in Version 7.0.1 neu erstellt wurden, haben eine andere Konfigurationsdatei als frühere Releases von v7. Wenn Sie das Fixpack für v7.0.1 entfernen müssen, um die Installation auf v7.0.0.2 zurückzusetzen, müssen Sie die Konfigurationsdateien erneut erstellen. Sie müssen möglicherweise nur das Attribut "Prefix" verwenden, um den Pfad zum neuen Datenverzeichnis des Warteschlangenmanagers zu definieren, oder Sie müssen das Datenverzeichnis des Warteschlangenmanagers und die Protokollverzeichnisse möglicherweise an eine andere Position verschieben. Der sicherste Weg, den Warteschlangenmanager neu zu konfigurieren, besteht darin, die WS-Manager-Daten und -Protokollverzeichnisse zu speichern, den Warteschlangenmanager zu löschen und neu zu erstellen und anschließend die Daten- und Protokollverzeichnisse an ihrer neuen Position zu ersetzen, wobei die Daten gespeichert wurden.

Typische Verzeichnisstruktur für Releases ab v7.0.1

Abbildung 56 auf Seite 156 zeigt die in v7.0.1 mit dem Befehl **crtmqm QM1** erstellte Standardverzeichnisstruktur.

Die Datei `mqs.ini` enthält eine Zeilengruppe für den Warteschlangenmanager QM1, die anhand des Werts von `DefaultPrefix` erstellt wird. Die Zeilengruppe `Log` in der Datei `qm.ini` hat einen Wert für `LogPath`, der durch Verweis auf `LogDefaultPath` in `mqs.ini` festgelegt wird.

Verwenden Sie die optionalen Parameter **crtmqm**, um die Standardwerte von `DataPath` und `LogPath` zu überschreiben.

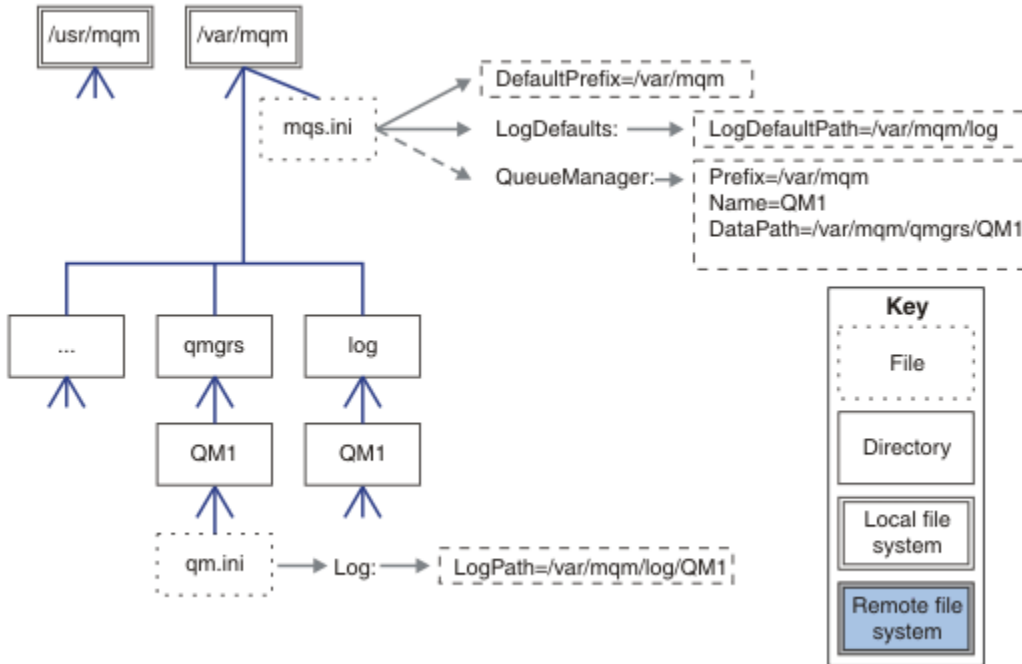


Abbildung 56. Beispiel für WebSphere MQ v7.R -Verzeichnisstruktur für UNIX and Linux -Systeme

Typische Verzeichnisstruktur für Releases vor v7.0.1

Das Attribut `DataPath` war vor WebSphere MQ v7.0.1; das Attribut ist in der Datei `mqs.ini` nicht vorhanden. Die Position des `qmgrs`-Verzeichnisses wurde mit dem Attribut `Prefix` konfiguriert. Die Position der einzelnen Verzeichnisse kann durch Verwendung symbolischer Verbindungen konfiguriert werden, um auf verschiedene Dateisystempositionen zu verweisen.

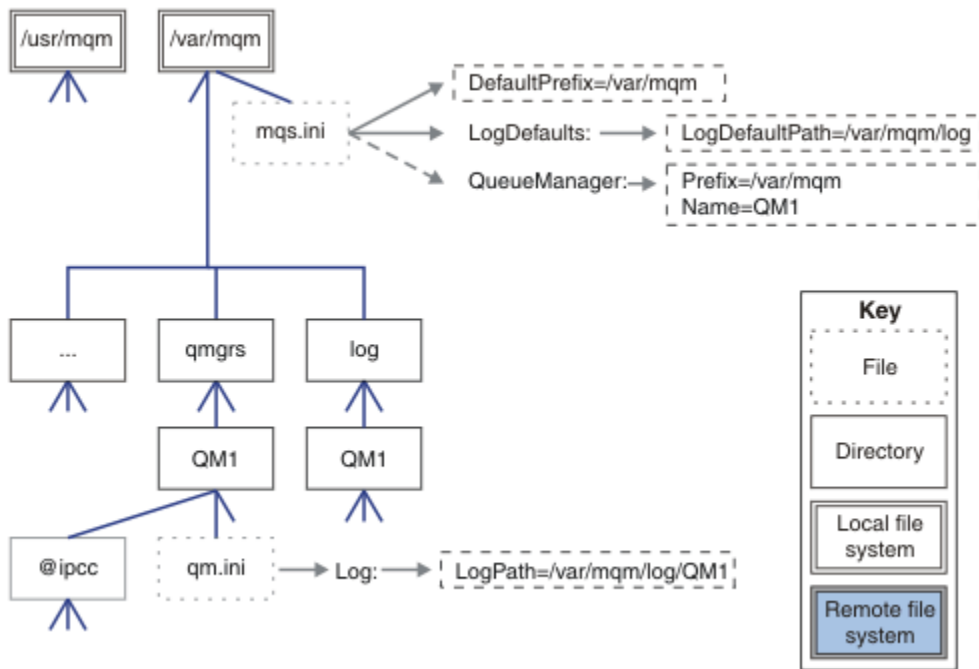


Abbildung 57. Typische Verzeichnisstruktur für Releases vor v7.0.1

Gemeinsame Nutzung der Standardverzeichnisse `qmgrs` und `log` (ab Release v7.0.1)

Eine Alternative zu „Gesamtes Dateisystem gemeinsam nutzen (ab Release v7.0.1)“ auf Seite 158 ist die separate gemeinsame Nutzung der Verzeichnisse `qmgrs` und `log` (Abbildung 58 auf Seite 157). In dieser Konfiguration muss `AMQ_MQS_INI_LOCATION` nicht festgelegt werden, da die Standarddatei `mqs.ini` im lokalen `/var/mqm`-Dateisystem gespeichert wird. Die Dateien und Verzeichnisse, wie z. B. `mqclient.ini` und `mqserver.ini`, werden ebenfalls nicht gemeinsam genutzt.

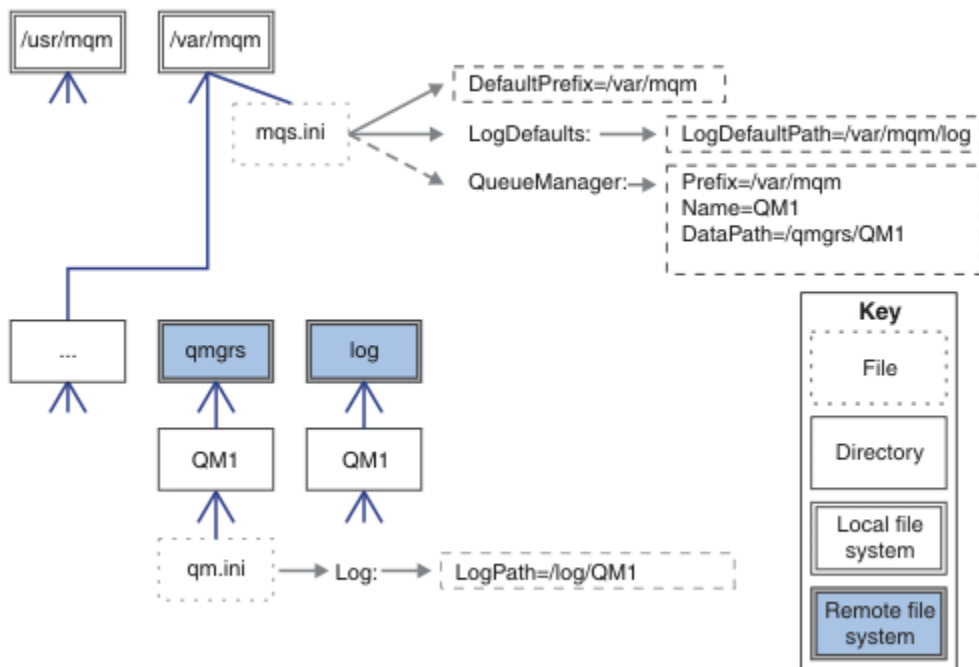


Abbildung 58. Verzeichnisse `qmgrs` und `log` gemeinsam nutzen

Gemeinsam genutzte Verzeichnisse qmgrs und log (ab Release v7.0.1)

Bei der Konfiguration in [Abbildung 59](#) auf Seite 158 werden log und qmgrs in einem gemeinsam genutzten fernem Dateisystem mit dem Namen /ha gespeichert. Dieselbe physische Konfiguration kann auf zwei verschiedene Arten erstellt werden.

1. Legen Sie `LogDefaultPath=/ha` fest und führen Sie dann den Befehl `crtmqm -md /ha/qmgrs QM1aus`. Das Ergebnis ist genau wie in [Abbildung 59](#) auf Seite 158 dargestellt.
2. Lassen Sie die Standardpfade unverändert und führen Sie dann den Befehl `crtmqm -ld /ha/log -md /ha/qmgrs QM1aus`.

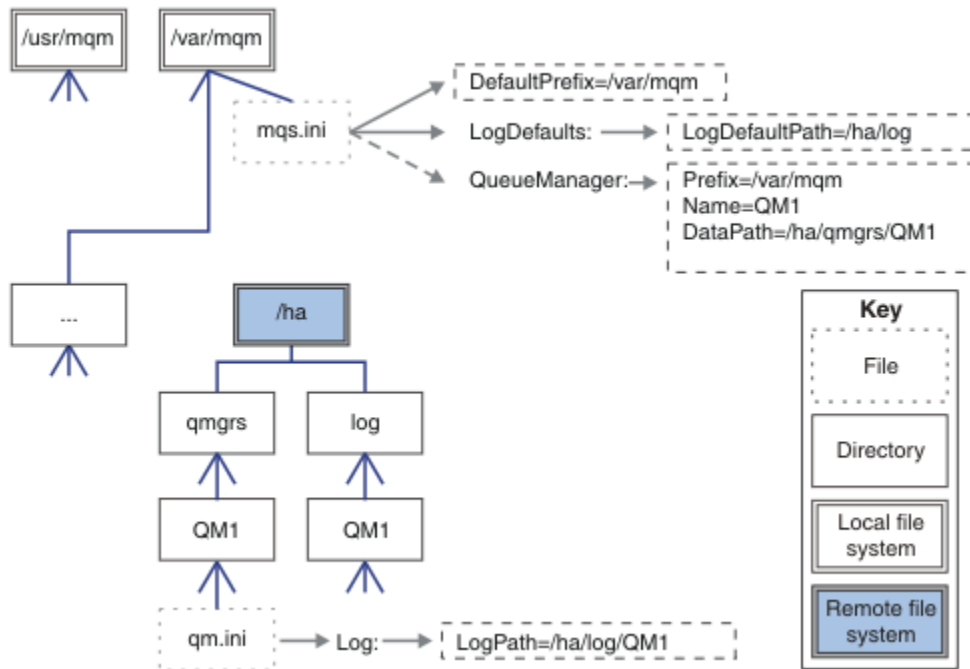


Abbildung 59. Benannte Verzeichnisse qmgrs und log gemeinsam nutzen

Gesamtes Dateisystem gemeinsam nutzen (ab Release v7.0.1)

[Abbildung 60](#) auf Seite 159 ist eine einfache Konfiguration für System mit schnell vernetztem Dateispeicher.

Hängen Sie `/var/mqm` als fernes gemeinsam genutzter Dateisystem an. Wenn Sie QM1 starten, sucht es standardmäßig nach `/var/mqm`, findet es im gemeinsam genutzten Dateisystem und liest die Datei `mqs.ini` in `/var/mqm`. Anstatt die einzelne Datei `/var/mqm/mqs.ini` für Warteschlangenmanager auf allen Servern zu verwenden, können Sie die Umgebungsvariable `AMQ_MQS_INI_LOCATION` auf jedem Server so festlegen, dass sie auf unterschiedliche `mqs.ini`-Dateien verweist.

Anmerkung: Der Inhalt der generischen Fehlerdatei in `/var/mqm/errors/` wird von Warteschlangenmanagern auf verschiedenen Servern gemeinsam genutzt.

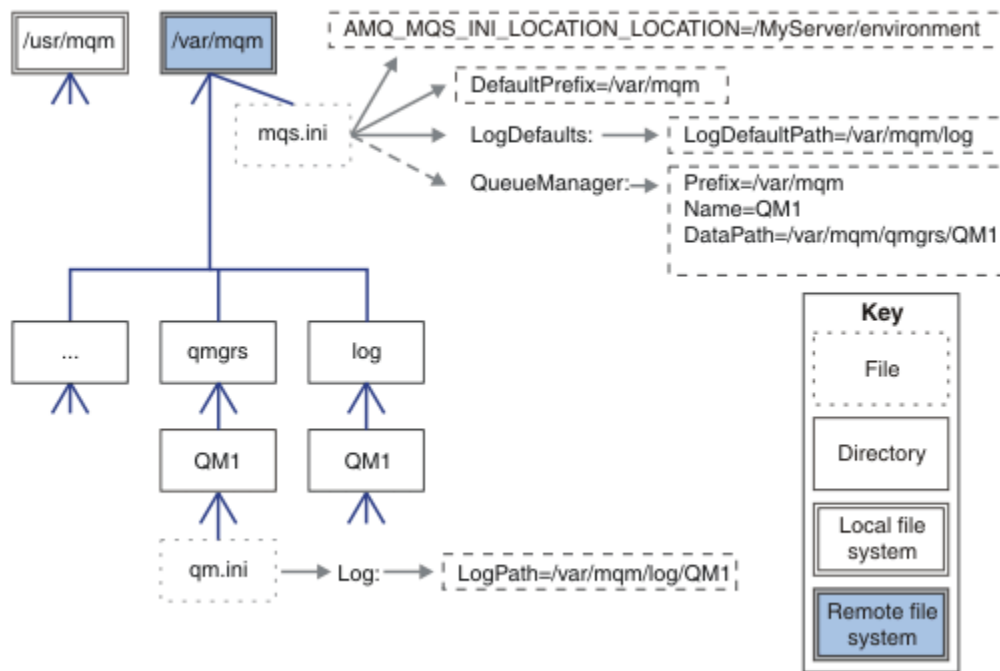


Abbildung 60. Alles teilen

Beachten Sie, dass Sie dies nicht für WS-Manager mit mehreren Instanzen verwenden können. Der Grund hierfür ist, dass für jeden Host in einem Warteschlangenmanager mit mehreren Instanzen eine eigene lokale Kopie von `/var/mqm` vorhanden ist, um die lokalen Daten, wie z. B. Semaphore und gemeinsam genutzten Speicher, zu überwachen. Diese Entitäten können nicht über Hosts hinweg gemeinsam genutzt werden.

Verzeichnisstruktur auf Windows -Systemen

Vorgehensweise zum Auffinden von Warteschlangenmanager-Konfigurationsinformationen und von Verzeichnissen unter Windows.

Das Standardverzeichnis für die Installation von IBM WebSphere MQ for Windows ist:

32-Bit

C:\Program Files\IBM\WebSphere MQ

64-Bit

C:\Program Files (x86)\IBM\WebSphere MQ

Die Installationsinformationen werden in der Windows -Registry gespeichert. Der Registrierungsschlüssel, in dem IBM WebSphere MQ -Informationen gespeichert werden, lautet wie folgt:

32-Bit

My Computer\HKEY_LOCAL_MACHINE\SOFTWARE\IBM\WebSphere MQ\

64-Bit

My Computer\HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\IBM\WebSphere MQ\

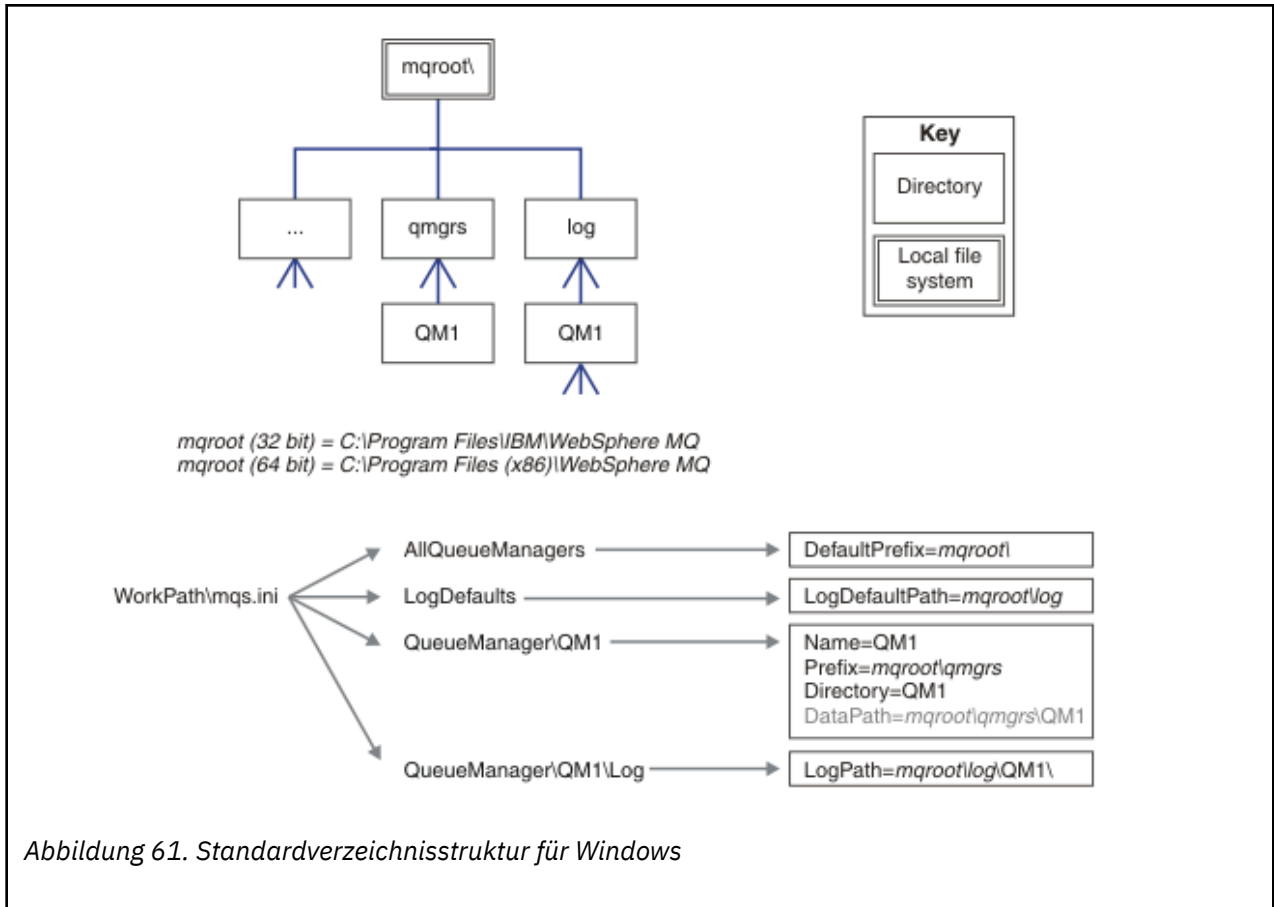
Jede Installation hat einen bestimmten Unterschlüssel:

Installation\<InstallationName>\

Der Pfad, der auf das IBM WebSphere MQ -Datenverzeichnis verweist, wird in einem Zeichenfolgewert namens `WorkPath` gespeichert und das Standardverzeichnis für Protokolle wird in `LogDefaultPath` gespeichert. Warteschlangenmanagerdatenverzeichnisse werden in `WorkPath\qmgrs\Qmgrname` erstellt. Warteschlangenmanagerprotokolle werden in `LogDefaultPath\QmgrName` erstellt. (siehe [Abbildung 61](#) auf Seite 160).

Wenn Sie die Warteschlangenmanager-Daten und Protokollverzeichnisse bei der Installation von IBM WebSphere MQ definieren, werden *WorkPath* und *LogDefaultPath* mit den angepassten Pfadinformationen aktualisiert.

WorkPath und *LogDefaultPath* werden nur zum Erstellen eines Warteschlangenmanagers verwendet.



Warteschlangenmanager mit mehreren Instanzen

Zum Konfigurieren eines Warteschlangenmanagers mit mehreren Instanzen müssen die Protokoll- und Datenverzeichnisse in den Netzspeicher gestellt werden, vorzugsweise auf einem anderen Server auf einem der Server, auf denen Instanzen des Warteschlangenmanagers ausgeführt werden.

Im Befehl **crtmqm** werden zwei Parameter bereitgestellt: **-md** und **-ld**, um die Angabe der Position der Warteschlangenmanagerdaten und Protokollverzeichnisse zu vereinfachen. Die Angabe des **-md**-Parameters hat eine vierfache Auswirkung:

1. Die Zeilengruppe `mq5.ini QueueManager\QmgrName` enthält eine neue Variable `DataPath`, die auf das Datenverzeichnis des Warteschlangenmanagers verweist. Im Gegensatz zur Variablen `Prefix` enthält der Pfad den Namen des WS-Manager-Verzeichnisses.
2. Die Konfigurationsdaten des Warteschlangenmanagers, die in der Datei `mq5.ini` gespeichert sind, werden auf `Name`, `Prefix`, `Directory` und `DataPath` reduziert.

Verzeichnisinhalt

Listet die Position und den Inhalt von WebSphere MQ -Verzeichnissen auf

Eine WebSphere MQ -Konfiguration verfügt über drei Hauptgruppen von Dateien und Verzeichnissen:

1. Ausführbare Dateien und andere schreibgeschützte Dateien wie die Readme-Datei, das WebSphere MQ Explorer-Plug-in und die Hilfedateien sowie Lizenzdateien, die nur aktualisiert werden, wenn die Wartung angewendet wird. Diese Dateien werden in [Tabelle 18 auf Seite 161](#) beschrieben.

2. Potenziell änderbare Dateien und Verzeichnisse, die für einen bestimmten WS-Manager nicht spezifisch sind. Diese Dateien und Verzeichnisse werden in [Tabelle 19 auf Seite 161](#) beschrieben.
3. Dateien und Verzeichnisse, die für die einzelnen WS-Manager auf einem Server spezifisch sind. Diese Dateien und Verzeichnisse werden in [Tabelle 18 auf Seite 161](#) beschrieben.

Ressourcenverzeichnisse und -dateien

Die Ressourcenverzeichnisse und -dateien enthalten den gesamten ausführbaren Code und die Ressourcen für die Ausführung eines Warteschlangenmanagers. Die Variable *FilePath* im installationsspezifischen Registrierungsschlüssel für die IBM WebSphere MQ-Konfiguration enthält den Pfad zu den Ressourcenverzeichnissen.

<i>Tabelle 18. Verzeichnisse und Dateien im Verzeichnis FilePath</i>	
Dateipfad	Inhalt
<i>FilePath\bin</i>	Befehle und DLLs
<i>FilePath\bin64</i>	Befehle und DLLs (64 Bit)
<i>FilePath\conv</i>	Datenkonvertierungstabellen
<i>FilePath\doc</i>	Hilfdateien für
<i>FilePath\MQExplorer</i>	Eclipse-Plug-ins für Explorer und Explorer
<i>FilePath\gskit8</i>	Globaler Sicherheitssatz
<i>FilePath\java</i>	Java-Ressourcen, einschließlich JRE
<i>FilePath\licenses</i>	Lizenzinformation
<i>FilePath\Non_IBM_License</i>	Lizenzinformation
<i>FilePath\properties</i>	Wird intern verwendet
<i>FilePath\Tivoli</i>	
<i>FilePath\tools</i>	Entwicklungsressourcen und -beispiele
<i>FilePath\Uninst</i>	Wird intern verwendet
<i>FilePath\README.TXT</i>	Readme-Datei

Verzeichnisse, die nicht für einen Warteschlangenmanager spezifisch sind

Einige Verzeichnisse enthalten Dateien, wie z. B. Tracedateien und Fehlerprotokolle, die nicht spezifisch für einen bestimmten Warteschlangenmanager sind. Die Variable *DefaultPrefix* enthält den Pfad zu diesen Verzeichnissen. *DefaultPrefix* ist Teil der Zeilengruppe *AllQueueManagers*.

<i>Tabelle 19. Verzeichnisse und Dateien im Verzeichnis DefaultPrefix</i>	
Dateipfad	Inhalt
<i>DefaultPrefix\Config</i>	Wird intern verwendet
<i>DefaultPrefix\conv</i>	Steuerdatei für die Konvertierung von ccsid.tbl-Daten, beschrieben in Datenkonvertierung
<i>DefaultPrefix\errors</i>	Fehlerprotokolle des Nicht-WS-Managers, AMQERRnn . LOG
<i>DefaultPrefix\exits</i>	Kanalexitprogramme
<i>DefaultPrefix\exits64</i>	Kanalexitprogramme (64 Bit)
<i>DefaultPrefix\ipc</i>	Nicht verwendet

Tabelle 19. Verzeichnisse und Dateien im Verzeichnis *DefaultPrefix* (Forts.)

Dateipfad	Inhalt
<i>DefaultPrefix</i> \Qmgrs	In Tabelle 20 auf Seite 162 beschrieben
<i>DefaultPrefix</i> \trace	Tracedateien
<i>DefaultPrefix</i> \amqmjpse.txt	Wird intern verwendet

WS-Manager-Verzeichnisse

Wenn Sie einen WS-Manager erstellen, wird eine neue Gruppe von Verzeichnissen erstellt, die für den Warteschlangenmanager spezifisch sind.

Wenn Sie einen Warteschlangenmanager mit dem Parameter **-md** *filepath* erstellen, wird der Pfad in der Variable *DataPath* in der Zeilengruppe des Warteschlangenmanagers der Datei *mqs.ini* gespeichert. Wenn Sie einen Warteschlangenmanager erstellen, ohne den Parameter **-md** *filepath* festzulegen, werden die Warteschlangenmanagerverzeichnisse in dem Pfad erstellt, der in *DefaultPrefix* gespeichert ist, und der Pfad wird in die Variable *Prefix* in der Zeilengruppe des Warteschlangenmanagers in der Datei *mqs.ini* kopiert.

Tabelle 20. Verzeichnisse und Dateien in den Verzeichnissen *DataPath* und *Prefix/Qmgrs/QmgrName*

Dateipfad	Inhalt
<i>DataPath</i> \@ipcc	Standardposition für AMQCLCHL . TAB, die Clientverbindungstabelle.
<i>DataPath</i> \authinfo	Wird intern verwendet.
<i>DataPath</i> \channel	
<i>DataPath</i> \clntconn	
<i>DataPath</i> \errors	Fehlerprotokolle, AMQERRnn . LOG
<i>DataPath</i> \listener	Wird intern verwendet.
<i>DataPath</i> \namelist	
<i>DataPath</i> \plugcomp	
<i>DataPath</i> \procdef	
<i>DataPath</i> \qmanager	
<i>DataPath</i> \queues	
<i>DataPath</i> \services	
<i>DataPath</i> \ssl	
<i>DataPath</i> \startprm	
<i>DataPath</i> \topic	
<i>DataPath</i> \active	
<i>DataPath</i> \active.dat	
<i>DataPath</i> \amqalchk.fil	
<i>DataPath</i> \master	
<i>DataPath</i> \master.dat	
<i>DataPath</i> \qm.ini	WS-Manager-Konfiguration

Tabelle 20. Verzeichnisse und Dateien in den Verzeichnissen *DataPath* und *Prefix/Qmgrs/QmgrName* (Forts.)

Dateipfad	Inhalt
<i>DataPath</i> \qmstatus.ini	Status des Warteschlangenmanagers
<i>Prefix</i> \Qmgrs\QmgrName	Wird intern verwendet
<i>Prefix</i> \Qmgrs\@SYSTEM	Nicht verwendet
<i>Prefix</i> \Qmgrs\@SYSTEM\errors	

IBM WebSphere MQ -und UNIX System V IPC-Ressourcen

Ein WS-Manager verwendet einige IPC-Ressourcen. Verwenden Sie **ipcs -a**, um herauszufinden, welche Ressourcen verwendet werden.

Diese Informationen gelten nur für IBM WebSphere MQ auf Systemen mit UNIX and Linux.

IBM WebSphere MQ verwendet System V-IPC-Ressourcen (*Semaphore* und *gemeinsam genutzte Speichersegmente*), um Daten zwischen Systemkomponenten zu speichern und zu übergeben. Diese Ressourcen werden von WS-Managerprozessen und -Anwendungen verwendet, die eine Verbindung zum Warteschlangenmanager herstellen. IBM WebSphere MQ MQI-Clients verwenden keine IPC-Ressourcen, mit Ausnahme der IBM WebSphere MQ -Tracesteuerung. Verwenden Sie den UNIX -Befehl **ipcs -a**, um vollständige Informationen zur Anzahl und Größe der momentan auf der Maschine verwendeten IPC-Ressourcen abzurufen.

Gemeinsam genutzter Speicher unter AIX

Wenn bestimmte Anwendungstypen aufgrund einer AIX -Speicherbegrenzung keine Verbindung herstellen können, kann dies in den meisten Fällen durch Festlegen der Umgebungsvariable EXTSHM=ON behoben werden.

Einige 32-Bit-Prozesse unter AIX können auf eine Betriebssystemeinschränkung stoßen, die sich auf ihre Fähigkeit auswirkt, eine Verbindung zu WebSphere MQ -Warteschlangenmanagern herzustellen. Jede Standardverbindung zu WebSphere MQ verwendet gemeinsam genutzten Speicher, aber im Gegensatz zu anderen UNIX and Linux -Plattformen ermöglicht AIX 32-Bit-Prozessen, nur 11 gemeinsam genutzte Speichergruppen zuzuordnen.

Bei den meisten 32-Bit-Prozessen tritt dieser Grenzwert nicht auf, aber Anwendungen mit hohem Speicherbedarf können möglicherweise mit dem Ursachencode 2102: MQRC_RESOURCE_PROBLEM keine Verbindung zu WebSphere MQ herstellen. In den folgenden Anwendungstypen wird möglicherweise dieser Fehler angezeigt:

- Programme, die in 32-Bit-Java Virtual Machines ausgeführt werden
- Programme, die die großen oder sehr großen Speichermodelle verwenden
- Programme, die Verbindungen zu vielen Warteschlangenmanagern oder Datenbanken herstellen
- Programme, die an gemeinsam genutzte Speichergruppen angehängt sind

AIX bietet eine erweiterte gemeinsam genutzte Speicherfunktion für 32-Bit-Prozesse, mit der sie mehr gemeinsam genutzten Speicher zuordnen können. Wenn Sie eine Anwendung mit dieser Funktion ausführen möchten, exportieren Sie die Umgebungsvariable EXTSHM=ON, bevor Sie Ihre Warteschlangenmanager und Ihr Programm starten. Die Funktion EXTSHM=ON verhindert in den meisten Fällen diesen Fehler, ist aber mit Programmen, die die Option SHM_SIZE der Funktion shmctl verwenden, nicht kompatibel.

WebSphere MQ MQI-Cliantwendungen und alle 64-Bit-Prozesse sind von dieser Einschränkung nicht betroffen. Sie können unabhängig davon, ob EXTSHM festgelegt wurde, eine Verbindung zu WebSphere MQ -Warteschlangenmanagern herstellen.

Prozesspriorität WebSphere MQ und UNIX

Good Practices beim Festlegen der Werte für die Prozesspriorität *nice* .

Diese Informationen gelten nur für WebSphere MQ auf UNIX and Linux -Systemen.

Wenn Sie einen Prozess im Hintergrund ausführen, kann diesem Prozess durch die aufrufende Shell ein höherer *nice* -Wert (und damit eine niedrigere Priorität) erteilt werden. Dies kann allgemeine Auswirkungen auf die Leistung von WebSphere MQ haben. Wenn es in stark beanspruchten Situationen viele gebrauchsfertige Threads mit einer höheren Priorität und einigen mit einer niedrigeren Priorität gibt, können die Merkmale der Betriebssystem-Zeitplanung die Threads mit der niedrigeren Priorität der Prozessorzeit vorenthalten.

Es empfiehlt sich, dass unabhängig gestartete Prozesse, die Warteschlangenmanagern zugeordnet sind, wie z. B. `runmq1sr`, dieselben *nice* -Werte haben wie der Warteschlangenmanager, dem sie zugeordnet sind. Stellen Sie sicher, dass die Shell diesen Hintergrundprozessen keinen höheren *nice* -Wert zuordnet. Verwenden Sie beispielsweise in ksh die Einstellung `"set +o bgnice"` , um zu verhindern, dass ksh den Wert *nice* für Hintergrundprozesse erhöht. Sie können die *nice* -Werte von aktiven Prozessen überprüfen, indem Sie die Spalte *NI* einer Liste `" ps -efl "` prüfen.

Starten Sie außerdem WebSphere MQ -Anwendungsprozesse mit demselben Wert für *nice* wie der Warteschlangenmanager. Wenn sie mit unterschiedlichen *nice* -Werten ausgeführt werden, blockiert ein Anwendungsthread möglicherweise einen Warteschlangenmanager-Thread, oder umgekehrt, wodurch sich die Leistung absetzt.

IBM WebSphere MQ -Clientumgebung unter HP Integrity NonStop Server planen

Bei der Planung Ihrer IBM WebSphere MQ-Umgebung müssen Sie die HP Integrity NonStop Server-Umgebung und die Transaction Manager Facility (TMF) von HP NonStop berücksichtigen. Planen Sie mithilfe der Informationen die Umgebung, in der der IBM WebSphere MQ-Client für HP Integrity NonStop Server ausgeführt wird.

Bevor Sie die Architektur Ihres IBM WebSphere MQ-Clients für HP Integrity NonStop Server planen, sollten Sie sich mit den Basiskonzepten des IBM WebSphere MQ-Clients für HP Integrity NonStop Server vertraut machen. Beachten Sie hierzu die Themen im Abschnitt [IBM WebSphere MQ-Client für HP Integrity NonStop Server - technische Übersicht](#).

HP Integrity NonStop Server -Umgebung vorbereiten

Vor der Installation muss die Umgebung vorbereitet werden, je nachdem, ob die Installation sofort überprüft werden soll oder nicht.

Für die Installation benötigen Sie die folgenden Elemente:

- Eine Benutzer-ID, die die Anforderungen erfüllt. Weitere Informationen zu den Benutzer-ID-Anforderungen finden Sie unter [Benutzer und Gruppen in HP Integrity NonStop Server einrichten](#).
- Verifizierte Positionen in den OSS- und Guardian-Dateisystemen, die für die Installationsdateien verwendet werden können.
- Ein betriebsbereit OSS-Shell- und OSS-Dateisystem. Sie können das Dateisystem überprüfen, indem Sie die folgenden Tasks ausführen:
 - Melden Sie sich an der OSS-Umgebung (Shell) an. Stellen Sie sicher, dass Sie Schreibzugriff auf das Stammverzeichnis der OSS-Installation haben, das Sie verwenden möchten.
 - Melden Sie sich mit der Benutzer-ID in der MQM-Gruppe bei der TAACL-Umgebung an. Stellen Sie sicher, dass der Datenträger, den Sie verwenden möchten, die Anforderungen erfüllt und für Sie zugänglich ist, und dass der Unterdatenträger nicht vorhanden ist.

Sie können sich sowohl mit einem Alias als auch mit einem Alias an OSS oder TAACL anmelden, wenn Sie einen Alias haben oder wenn Sie den vollständigen Principal verwenden.

Wenn Sie fortfahren möchten, um zu überprüfen, ob die Installation verwendbar ist, benötigen Sie möglicherweise auch die folgenden optionalen Elemente:

- Ein funktionales und zugängliches lokales Sockets-Subsystem in der OSS-Umgebung.
- Ein betriebes TCP/IP-Subsystem.

Wenn Sie beabsichtigen, TMF-koordinierte globale Arbeitseinheiten zu verwenden, benötigen Sie die folgenden Elemente:

- Ein betriebsbereit-TMF-Subsystem.
- Ein Subsystem für Betriebspfadbetrieb (TS/MP).

Arbeiten Sie mit Ihrem Systemadministrator, wenn Sie Zweifel am Status dieser kritischen Subsysteme haben.

IBM WebSphere MQ und HP NonStop TMF

Der IBM WebSphere MQ-Client für HP Integrity NonStop Server kann in Arbeitseinheiten mitwirken, die von HP NonStop Transaction Management Facility (HP NonStop TMF) koordiniert werden. Die Koordination von Transaktionen über HP NonStop TMF wird nur unterstützt, wenn der Warteschlangenmanager IBM WebSphere MQ Version 7.1 oder höher aufweist.

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. Das durch IBM WebSphere MQ bereitgestellte TMF-Gateway, das die durch HP NonStop TMF zur Verfügung gestellten Services nutzt, fungiert als Brücke zwischen TMF und Warteschlangenmanagertransaktionen und ist zur Ausführung in einer Pathway-Umgebung vorgesehen.

HP NonStop TMF-Software bietet Transaktionsschutz und Datenbankkonsistenz in anspruchsvollen Umgebungen. Weitere Informationen zu HP NonStop TMF finden Sie unter [HP NonStop TMF Introduction](#).

Informationen zum Konfigurieren des durch IBM WebSphere MQ bereitgestellten TMF-Gateway finden Sie unter [HP Integrity NonStop Server konfigurieren](#).

HP NonStop TMF verwenden

HP NonStop Transaction Management Facility (TMF) ist der native Transaktionsmanager auf HP Integrity NonStop Server, der in das Dateisystem und die Manager für relationale Datenbanken, SQL/MP und SQL/MX, integriert ist.

Der IBM WebSphere MQ-Client für HP Integrity NonStop Server kann TMF zur Koordination globaler Arbeitseinheiten verwenden.

Um globale Arbeitseinheiten zu koordinieren, fungiert TMF als Transaktionsmanager, und eine Anwendung muss die von TMF zur Verfügung gestellten APIs verwenden, um globale Arbeitseinheiten zu starten, festzuschreiben und zurück zu sichern. Globale Arbeitseinheiten werden von einer Anwendung über den Aufruf BEGINTRANSACTION gestartet; die anschließende Aktualisierung der IBM WebSphere MQ-Ressourcen in der globalen Arbeitseinheit erfolgt durch Ausgabe der Aufrufe MQPUT, MQPUT1 und MQGET in der Synchronisationspunktsteuerung. Die Anwendung kann dann die globale Arbeitseinheit durch Aufrufen von ENDTRANSACTION oder durch Aufrufen von ABORTTRANSACTION festschreiben.

Eine Anwendung, die TMF-Transaktionen verwendet, kann nur zu einem beliebigen Zeitpunkt aktiv an einer Transaktion arbeiten. Mit RESUMETRANSACTION kann jedoch eine Anwendung von einer aktiven Transaktion zu einer anderen wechseln oder einer TMF-Transaktion zugeordnet werden, ohne die zuvor aktive Transaktion zu beenden oder abzubrechen. Alle Aufrufe von MQPUT, MQPUT1 oder MQGET werden unter der derzeit aktiven TMF-Transaktion, falls vorhanden, oder einer lokalen UOWs (falls nicht vorhanden) ausgeführt. Daher muss innerhalb der Anwendung darauf geachtet werden, dass diese Aufrufe innerhalb der richtigen Arbeitseinheit erfolgen.

In einer globalen Arbeitseinheit und bei der Aktualisierung von IBM WebSphere MQ-Ressourcen kann eine Anwendung Enscribe-Dateien, SQL/MP-Datenbanken oder SQL/MX-Datenbanken aktualisieren.

Globale Arbeitseinheiten verwenden

Eine globale Arbeitseinheit wird als TMF-Transaktion implementiert. Eine Anwendung startet eine globale Arbeitseinheit mit dem Aufruf von BEGINTRANSACTION und schreibt die Arbeitseinheit fest, indem sie ENDTRANSACTION aufruft oder die Arbeitseinheit durch Aufrufen von ABORTTRANSACTION zurückgesetzt wird. Eine Anwendung kann auch andere TMF-API-Aufrufe verwenden.

Eine Anwendung kann eine TMF-Transaktion von einer anderen Anwendung übernehmen. Eine Anwendung (die erste Anwendung) kann beispielsweise in der Transaktion Arbeiten ausführen, bevor sie die Transaktion zurückgibt und an eine zweite Anwendung zurückgibt, um die Verarbeitung weiter zu verarbeiten. So können beide Anwendungen an derselben globalen Arbeitseinheit mitwirken und beispielsweise IBM WebSphere MQ-Warteschlangen oder Dateien und Datenbanken aktualisieren. Die Möglichkeit, eine TMF-Transaktion zwischen Anwendungen hin- und herzuschicken bedeutet, dass mehrere IBM WebSphere MQ-Anwendungen Messaging-Operationen an derselben globalen Arbeitseinheit durchführen können.

Eine Anwendung kann mehrere aktive TMF-Transaktionen zur gleichen Zeit verwalten und steuern. Die Transaktionen können von der Anwendung selbst gestartet oder von anderen Anwendungen übernommen werden. Dies bedeutet, dass eine Anwendung an mehreren globalen Arbeitseinheiten gleichzeitig teilnehmen kann.

Die maximale Anzahl gleichzeitiger aktiver TMF-Transaktionen pro Prozess beträgt 1000, was eine architektonische Grenze darstellt. Wenn eine Anwendung mehrere TMF-Transaktionen verwaltet, kann nur eine Transaktion zu einem beliebigen Zeitpunkt vorhanden sein. Alternativ kann keine der Transaktionen aktuell sein. Die Anwendung kann TMF-API-Aufrufe wie RESUMETRANSACTION, ACTIVATERECEIVETRANSID und TMF_SET_TX_ID verwenden, um den Status von einer Transaktion in eine andere zu versetzen, oder um zu bestimmen, dass keine Transaktion aktuell ist. Die Anwendung verwendet diese Steuerstufe, um zu bestimmen, ob eine Messaging-Operation innerhalb einer lokalen Arbeitseinheit, einer globalen Arbeitseinheit oder außerhalb der Synchronisationspunktsteuerung ausgeführt wird:

- Wenn eine Anwendung MQPUT, MQPUT1 oder MQGET innerhalb der Synchronisationspunktsteuerung aufruft und keine TMF-Transaktion aktuell ist, verarbeitet IBM WebSphere MQ den Aufruf in einer lokalen Arbeitseinheit.
- Wenn eine Anwendung MQPUT, MQPUT1 oder MQGET innerhalb der Synchronisationspunktsteuerung aufruft und die Anwendung über eine aktuelle TMF-Transaktion verfügt, verarbeitet IBM WebSphere MQ den Aufruf in der durch die aktuelle TMF-Transaktion implementierten globalen Arbeitseinheit.
- Wenn eine Anwendung MQPUT, MQPUT1 oder MQGET außerhalb der Synchronisationspunktsteuerung aufruft, verarbeitet IBM WebSphere MQ den Aufruf außerhalb der Synchronisationspunktsteuerung, unabhängig davon, ob die Anwendung zum Zeitpunkt des Aufrufs über eine aktuelle TMF-Transaktion verfügt.

IBM WebSphere MQ ändert nie den Status einer TMF-Transaktion einer Anwendung während eines MQI-Aufrufs, es sei denn, bei der Verarbeitung tritt ein Software- oder Hardwarefehler auf und IBM WebSphere MQ oder das Betriebssystem bestimmt, dass die Transaktion zurückgesetzt werden muss, um die Datenintegrität zu wahren. Jeder MQI-Aufruf stellt den Transaktionsstatus der Anwendung wieder her, kurz bevor die Steuerung an die Anwendung zurückgegeben wird.

Vermeiden von Transaktionen mit langer Laufzeit

Vermeiden Sie das Entwerfen von Anwendungen, in denen TMF-Transaktionen für mehr als ein paar Dutzend Sekunden aktiv bleiben. Transaktionen mit langer Laufzeit können dazu führen, dass die Umlaufprotokollierungsspur von TMF gefüllt wird. Da TMF eine kritische systemweite Ressource ist, schützt TMF sich selbst, indem es Anwendungs-Transaktionen sichert, die zu lange aktiv sind.

Angenommen, die Verarbeitung in einer Anwendung wird durch das Abrufen von Nachrichten aus einer Warteschlange gesteuert, und die Anwendung ruft eine Nachricht aus der Warteschlange ab und verarbeitet die Nachricht innerhalb einer UO-Unit. Gewöhnlich ruft eine Anwendung MQGET mit der Option 'wait' und innerhalb der Synchronisationspunktsteuerung auf, um eine Nachricht aus der Warteschlange abzurufen.

Wenn die Anwendung stattdessen eine globale Arbeitseinheit verwendet, muss das angegebene Warteintervall für den MQGET-Aufruf kurz sein, um eine lange laufende Transaktion zu vermeiden. Dies bedeutet, dass die Anwendung möglicherweise mehr als einmal den MQGET-Aufruf absetzen muss, bevor sie eine Nachricht abrufen.

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. 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: