

9.0

Administrowanie programem IBM MQ

IBM

Uwaga

Przed skorzystaniem z niniejszych informacji oraz produktu, którego one dotyczą, należy zapoznać się z informacjami zamieszczonymi w sekcji [“Uwagi” na stronie 489](#).

To wydanie dotyczy wersji 9 wydania 0 produktu IBM® MQ oraz wszystkich kolejnych wydań i modyfikacji, o ile nie zostanie to określone inaczej w nowych edycjach.

Wysyłając informacje do IBM, użytkownik przyznaje IBM niewyłączne prawo do używania i rozpowszechniania informacji w dowolny sposób, jaki uzna za właściwy, bez żadnych zobowiązań wobec ich autora.

© **Copyright International Business Machines Corporation 2015, 2023.**

Spis treści

Administrowanie.....	5
Administrowanie lokalne i zdalne.....	8
Administrowanie programem IBM MQ za pomocą komend sterujących.....	9
Administrowanie programem MQ za pomocą komend MQSC.....	10
Komendy skryptowe (MQSC).....	11
Nazwy obiektów produktu IBM MQ w komendach MQSC.....	13
Standardowe wejście i wyjście.....	13
Interaktywna obsługa komend MQSC.....	13
Uruchamianie komend MQSC z plików tekstowych.....	16
Uruchamianie komend MQSC z plików wsadowych.....	19
Rozwiązywanie problemów za pomocą komend MQSC.....	20
Automatyzacja administrowania produktem IBM MQ przy użyciu komend PCF.....	21
Wprowadzenie do formatów komend programowalnych IBM MQ.....	22
Korzystanie z interfejsu MQAI w celu uproszczenia korzystania z systemów PCF.....	34
Administrowanie przy użyciu REST API.....	72
Pierwsze kroki w produkcie administrative REST API.....	73
Korzystanie z administrative REST API.....	79
Zdalne administrowanie przy użyciu REST API.....	85
REST API datowniki.....	89
REST API obsługa błędów.....	89
REST API Wykrywanie.....	92
REST API Obsługa języków narodowych.....	93
Administrowanie przy użyciu IBM MQ Console.....	95
Pierwsze kroki w produkcie IBM MQ Console.....	95
Praca z menedżerami kolejek lokalnych.....	99
Praca z obiektami IBM MQ.....	101
Praca z rekordami uprawnień.....	116
Monitorowanie wykorzystania zasobów systemu.....	119
Konfigurowanie układów panelu kontrolnego.....	131
Elementy sterujące panelu kontrolnego.....	133
Skróty klawiaturowe.....	134
Administrowanie przy użyciu IBM MQ Explorer.....	134
Co można zrobić za pomocą IBM MQ Explorer.....	134
Konfigurowanie produktu IBM MQ Explorer.....	136
Korzystanie z aplikacji paska zadań IBM MQ (tylko w wersji Windows).....	142
Aplikacja monitora alertów IBM MQ (tylko Windows).....	143
Administrowanie lokalnymi obiektami IBM MQ.....	143
Uruchamianie i zatrzymywanie menedżera kolejek.....	143
Zatrzymywanie kanałów MQI.....	147
Praca z menedżerami kolejek.....	147
Praca z kolejkami lokalnymi.....	149
Praca z kolejkami aliasami.....	154
Praca z kolejkami niedostarczonych komunikatów.....	156
Praca z kolejkami modelami.....	175
Praca z tematami administracyjnymi.....	176
Praca z subskrypcjami.....	179
Praca z usługami.....	183
Zarządzanie obiektami w celu wyzwolenia.....	191
Korzystanie z programu narzędziowego dmpmqmsg między dwoma systemami.....	193
Administrowanie zdalnymi obiektami IBM MQ.....	196
Kanały i kolejkowanie zdalne.....	196
Zdalne administrowanie z lokalnego menedżera kolejek.....	198

Tworzenie lokalnej definicji kolejki zdalnej.....	204
Sprawdzanie, czy komendy asynchroniczne dla sieci rozproszonych zostały zakończone.....	206
Używanie definicji kolejek zdalnych jako aliasów.....	209
Konwersja danych.....	209
administrowanieMQ Telemetry.....	214
Konfigurowanie menedżera kolejek dla telemetrii w systemach Linux i AIX.....	215
Konfigurowanie menedżera kolejek dla telemetrii w systemie Windows.....	217
Konfigurowanie rozproszonego kolejkowania w celu wysyłania komunikatów do klientów MQTT.....	218
MQTT -identyfikacja, autoryzacja i uwierzytelnianie klienta.....	221
Uwierzytelnianie kanału telemetrycznego przy użyciu protokołu TLS.....	227
Prywatność publikacji na kanałach telemetrycznych.....	229
Konfiguracja TLS dla klientów MQTT Java i kanałów telemetrycznych.....	230
Konfiguracja kanału telemetrycznego JAAS.....	235
administrowanieIBM MQ Light.....	237
Wyświetlanie obiektów IBM MQ w użyciu przez klienty MQ Light.....	237
MQ Light -identyfikacja, autoryzacja i uwierzytelnianie klienta.....	239
Prywatność publikacji w kanałach.....	241
Konfigurowanie klientów MQ Light za pomocą protokołu TLS.....	242
Rozłączanie klientów MQ Light z menedżera kolejek.....	243
Administrowanie rozgłaszanie.....	243
Pierwsze kroki w trybie rozsyłania grupowego.....	243
IBM MQ Topologia tematu rozsyłania grupowego.....	244
Kontrolowanie wielkości komunikatów rozsyłania grupowego.....	245
Włączanie konwersji danych na potrzeby przesyłania komunikatów w trybie Multicast.....	247
Monitorowanie aplikacji rozsyłania grupowego.....	248
Niezawodność komunikatów rozsyłania grupowego.....	249
Zaawansowane zadania rozsyłania grupowego.....	250
administrowanieIBM MQ for IBM i.....	253
Zarządzanie produktem IBM MQ for IBM i za pomocą komend CL.....	253
Alternatywne sposoby administrowania produktem IBM MQ for IBM i.....	267
Zarządzanie pracą dla produktu IBM i.....	272
Dostępność, tworzenie kopii zapasowych, odtwarzanie i restartowanie w systemie IBM i.....	280
WYCISZANIEIBM MQ for IBM i.....	325
administrowanieIBM MQ for z/OS.....	329
Wysyłanie komend do programu IBM MQ for z/OS.....	330
Programy narzędziowe serwera IBM MQ for z/OS.....	338
Operacyjny IBM MQ for z/OS.....	340
Pisanie programów do administrowania programem IBM MQ for z/OS.....	362
Zarządzanie zasobami IBM MQ w systemie z/OS.....	374
Odtwarzanie i restartowanie w systemie z/OS.....	415
IBM MQandIMS.....	437
Operacyjne Advanced Message Security w systemie z/OS.....	450
Dostawca usług IBM MQ for z/OS dla produktu z/OS Connect.....	451
Uwagi.....	489
Informacje dotyczące interfejsu programistycznego.....	490
Znaki towarowe.....	491



administrowanie IBM MQ

Aby administrować menedżerami kolejek produktu IBM MQ i powiązаныmi zasobami, należy wybrać preferowaną metodę z zestawu zadań, które mogą być używane do aktywowania i zarządzania tymi zasobami.






Obiekty produktu IBM MQ można administrować lokalnie lub zdalnie, patrz [“Administrowanie lokalne i zdalne”](#) na stronie 8.


Istnieje wiele różnych metod tworzenia i administrowania menedżerami kolejek i ich zasobami pokrewnymi w produkcie IBM MQ. Do metod tych należą interfejsy wiersza komend, graficzne interfejsy użytkownika oraz interfejs API administrowania.

Istnieją różne zestawy komend, których można używać do administrowania produktem IBM MQ w zależności od platformy:

- [“Komendy sterujące produktu IBM MQ”](#) na stronie 5
- [“Komendy produktu IBM MQ Script \(MQSC\)”](#) na stronie 6
- [“Programowalne formaty komend \(PCFs\)”](#) na stronie 6
-  [administrative REST API](#)
-  [“Język CL \(IBM i Control Language\)”](#) na stronie 7

Istnieją również inne następujące opcje tworzenia obiektów IBM MQ i zarządzania nimi:

-   [“IBM MQ Explorer”](#) na stronie 7
-  [“IBM MQ Console”](#) na stronie 8
-  [“Aplikacja Konfiguracja domyślna produktu Windows”](#) na stronie 8
-  [“Usługa klastrowa produktu Microsoft \(MSCS\)”](#) na stronie 8

 Informacje na temat interfejsów i opcji administracyjnych w programie IBM MQ for z/OS można znaleźć w sekcji [“administrowanie IBM MQ for z/OS”](#) na stronie 329.

Niektóre zadania administrowania i monitorowania można zautomatyzować zarówno dla lokalnych, jak i zdalnych menedżerów kolejek za pomocą komend PCF. Komendy te można również uprościć, korzystając z interfejsu administracyjnego produktu IBM MQ (MQAI) na niektórych platformach. Więcej informacji na temat automatyzowania zadań administracyjnych zawiera sekcja [“Automatyzacja administrowania produktem IBM MQ przy użyciu komend PCF”](#) na stronie 21.

Komendy sterujące produktu IBM MQ



Za pomocą komend sterujących można wykonywać zadania administracyjne dotyczące samych menedżerów kolejek.

IBM MQ for Windows, systemy UNIX and Linux® udostępniają *komendy sterujące*, które są uruchamiane w wierszu komend systemu.

Komendy sterujące są opisane w sekcji [Tworzenie i zarządzanie menedżerami kolejek na wielu platformach](#). Informacje na temat komend sterujących można znaleźć w sekcji [Komendy sterujące programu IBM MQ](#).

Komendy produktu IBM MQ Script (MQSC)

Za pomocą komend MQSC służą do zarządzania obiektami menedżera kolejek, w tym samym menedżerem kolejek, kolejkami, definicjami procesów, listami nazw, kanałami, kanałami połączeń klientów, obiektami nastuchiwania, usługami i obiektami informacji uwierzytelniających.

Komendy MQSC są wystawiane na potrzeby menedżera kolejek za pomocą komendy `runmqsc`. Można to zrobić interaktywnie, wydając komendy z klawiatury lub można przekierować standardowe urządzenie wejściowe (stdin), aby uruchomić sekwencję komend z pliku tekstowego ASCII. W obu przypadkach format komend jest taki sam.

Komendę `runmqsc` można uruchomić w trzech trybach, w zależności od opcji ustawionych w komendzie:

- *Tryb weryfikacji*, w którym komendy MQSC są weryfikowane w lokalnym menedżerze kolejek, ale nie są uruchamiane.
- *Tryb bezpośredni*, w którym komendy MQSC są uruchamiane w lokalnym menedżerze kolejek
- *Tryb pośredni*, w którym komendy MQSC są uruchamiane w zdalnym menedżerze kolejek

Komendy MQSC są dostępne na wszystkich platformach, w tym IBM i i z/OS. Komendy MQSC są podsumowywane w sekcji [Porównywanie zestawów komend](#).

ULW W systemie UNIX, Linux, and Windowsza pomocą komend MQSC można używać pojedynczych komend wydawanych w wierszu komend systemu. Aby można było wydać bardziej skomplikowane lub wiele komend, MQSC można utworzyć w pliku, który jest uruchamiany z poziomu wiersza komend. Komendy MQSC mogą być wysyłane do zdalnego menedżera kolejek. Szczegółowe informacje na ten temat zawiera sekcja [Budowanie skryptów komend](#).

IBM i Aby wydać komendy na serwerze IBM i, należy utworzyć listę komend w pliku skryptowych, a następnie uruchomić ten plik za pomocą komendy STRMQMMQSC.

Uwagi: **IBM i**

1. Nie należy używać biblioteki QTEMP jako biblioteki wejściowej do komendy STRMQMMQSC, ponieważ użycie biblioteki QTEMP jest ograniczone. Do komendy należy użyć innej biblioteki jako pliku wejściowego.
2. W systemie IBM i odpowiedzi MQSC dla komend, które są wysyłane z pliku skryptowego, są zwracane w pliku buforowy-buforu.

“Komendy skryptowe (MQSC)” na stronie 11 zawiera opis każdej komendy MQSC i jej składni.

Więcej informacji na temat używania komend MQSC zawiera sekcja [“Administrowanie programem MQ za pomocą komend MQSC”](#) na stronie 10.

Programowalne formaty komend (PCFs)

Programowalne formaty komend (PCFs) definiują komunikaty komend i odpowiedzi, które mogą być wymieniane między programem a dowolnym menedżerem kolejek (który obsługuje systemy PCF) w sieci. Komendy PCF można używać w aplikacji do zarządzania systemami w celu administrowania obiektami produktu IBM MQ: obiekty informacji uwierzytelniającej, kanały, obiekty nastuchiwania kanałów, listy nazw, definicje procesów, menedżery kolejek, kolejki, usługi i klasy pamięci masowej. Aplikacja może działać z jednego punktu w sieci, aby komunikować informacje o komendach i odpowiedziach z dowolnym menedżerem kolejek, lokalnym lub zdalnym za pomocą lokalnego menedżera kolejek.

Więcej informacji na temat systemów PCFs zawiera sekcja [“Wprowadzenie do formatów komend programowalnych IBM MQ”](#) na stronie 22.

Definicje PCF i struktur dla komend i odpowiedzi zawiera sekcja [Programmable command formats reference](#)(Skorowidz formatów komend programowalnych).

administrative REST API

V 9.0.1

Produkt administrative REST API udostępnia interfejs zgodny ze specyfikacją REST, który może być używany do administrowania produktem IBM MQ. Podczas korzystania z produktu administrative REST API wywoływana jest metoda HTTP w adresie URL, która reprezentuje obiekt IBM MQ. Na przykład można zażądać informacji na temat instalacji produktu IBM MQ przy użyciu metody HTTP GET dla następującego adresu URL:

V 9.0.4 IBM MQ 9.0.4 i nowsze:

```
https://localhost:9443/ibmmq/rest/v1/admin/installation
```

IBM MQ 9.0.3 i wcześniejsze:

```
https://localhost:9443/ibmmq/rest/v1/installation
```

Produktu administrative REST API można używać z implementacją HTTP/REST języka programowania lub za pomocą narzędzi, takich jak cURL, lub z dodaniem przeglądarki klienta REST.

Więcej informacji na ten temat zawiera sekcja [administrative REST API](#).

Język CL (IBM i Control Language)

IBM i

Ten język może być używany do wydawania komend administracyjnych do produktu IBM MQ for IBM i. Komendy mogą być wydawane w wierszu komend lub przez napisanie programu CL. Te komendy pełnią podobne funkcje w komendach PCF, ale format jest inny. Komendy CL są przeznaczone wyłącznie dla serwerów i odpowiedzi CL są czytelne dla użytkownika, podczas gdy komendy PCF są niezależne od platformy, a oba formaty komend i odpowiedzi są przeznaczone do użycia przez program.

Szczegółowe informacje na temat języka CL (IBM i Control Language) zawiera sekcja [Komendy CL programu IBM MQ for IBM i](#).

IBM MQ Explorer

Windows Linux

Za pomocą IBM MQ Explorer można wykonywać następujące czynności:

- Definiowanie i kontrolowanie różnych zasobów, takich jak menedżery kolejek, kolejki, definicje procesów, listy nazw, kanały, kanały połączeń klientów, obiekty nastuchiwania, usługi i klastry.
- Uruchamianie lub zatrzymywanie lokalnego menedżera kolejek i powiązanych z nim procesów.
- Wyświetlanie menedżerów kolejek i powiązanych z nimi obiektów na stacji roboczej lub na innych stacjach roboczych.
- Sprawdź status menedżerów kolejek, klastrów i kanałów.
- Sprawdź, które aplikacje, użytkownicy lub kanały mają otwartą określoną kolejkę, z poziomu statusu kolejki.

W systemach Windows i Linux można uruchomić program IBM MQ Explorer, korzystając z menu systemowego, pliku wykonywalnego MQExplorer lub komendy **strmqcfig**.

Linux

W systemie Linux, aby pomyślnie uruchomić serwer IBM MQ Explorer, należy mieć możliwość zapisu pliku w katalogu osobistym, a katalog osobisty musi istnieć.

Więcej informacji na ten temat zawiera sekcja [“Administrowanie przy użyciu IBM MQ Explorer” na stronie 134](#).

Za pomocą produktu IBM MQ Explorer można administrować zdalnymi menedżerami kolejek na innych platformach, w tym z/OS, aby uzyskać szczegółowe informacje i pobrać pakiet SupportPac MS0T. Więcej informacji na ten temat zawiera sekcja <https://www.ibm.com/support/docview.wss?uid=swg24021041>.

IBM MQ Console

V 9.0.1

Za pomocą programu IBM MQ Console można administrować produktem IBM MQ z poziomu przeglądarki WWW.

Więcej informacji na ten temat zawiera sekcja [“Administrowanie przy użyciu IBM MQ Console”](#) na stronie 95.

Aplikacja Konfiguracja domyślna produktu Windows

Windows

Do utworzenia *uruchamiającego* (lub domyślnego) zestawu obiektów produktu IBM MQ można użyć domyślnego programu konfiguracyjnego Windows . Podsumowanie tworzonych obiektów domyślnych znajduje się w tabeli [Tabela 1: Obiekty utworzone przez domyślną aplikację konfiguracyjną produktu Windows](#).

Usługa klastrowa produktu Microsoft (MSCS)

Windows

Program Microsoft Cluster Service (MSCS) umożliwia połączenie serwerów z *klastrem*, co zapewnia wyższą dostępność danych i aplikacji, a także ułatwia zarządzanie systemem. MSCS może automatycznie wykrywać i odzyskiwać awarie serwera lub aplikacji.

Ważne jest, aby nie mylić klastrów w sensie MSCS z klastrami produktu IBM MQ . Rozróżnienie to jest następujące:

IBM MQKlastry

Są to grupy dwóch lub większej liczby menedżerów kolejek na jednym lub większej liczby komputerów, zapewniające automatyczne połączenie wzajemne i umożliwiające współużytkowanie kolejek między nimi w celu równoważenia obciążenia i nadmiarowości.

Klastry MSCS

Są to grupy komputerów, które są połączone ze sobą i skonfigurowane w taki sposób, że jeśli jedno się nie powiedzie, MSCS wykonuje *przełączenie awaryjne*, przesyłając dane stanu aplikacji z uszkodzonego komputera na inny komputer w klastrze i ponownie inicjując ich działanie.

Obsługa usługi Microsoft Cluster Service (MSCS) zawiera szczegółowe informacje na temat konfigurowania systemu IBM MQ for Windows w taki sposób, aby można było używać usługi MSCS.

Informacje pokrewne

[Porównanie zestawów komend](#)

[IBM MQ Przegląd techniczny](#)

[Planowanie](#)

[Konfigurowanie](#)

Administrowanie lokalne i zdalne

Obiekty produktu IBM MQ można administrować lokalnie lub zdalnie.

administracja lokalna

Administrowanie lokalne oznacza wykonywanie zadań administracyjnych na wszystkich menedżerach kolejek, które zostały zdefiniowane w systemie lokalnym. Dostęp do innych systemów można uzyskać, na przykład za pomocą programu emulacji terminalu TCP/IP **telnet**, i przeprowadzając tam administrację. W programie IBM MQ można uznać, że jest to administracja lokalna, ponieważ nie są zaangażowane żadne kanały, to znaczy komunikacja jest zarządzana przez system operacyjny.

Zdalne administrowanie

Produkt IBM MQ obsługuje administrowanie z pojedynczego punktu kontaktowego przy użyciu zdalnego administrowania. Zdalne administrowanie umożliwia wydawanie komend z systemu lokalnego, które są przetwarzane w innym systemie i ma zastosowanie także do IBM MQ Explorer. Na przykład można wydać komendę zdalną w celu zmiany definicji kolejki w zdalnym menedżerze kolejek. Użytkownik nie musi logować się do tego systemu, chociaż trzeba mieć zdefiniowane odpowiednie kanały. Musi być uruchomiony menedżer kolejek i serwer komend w systemie docelowym.

Niektóre komendy nie mogą być wydawane w ten sposób, w szczególności w przypadku tworzenia lub uruchamiania menedżerów kolejek i uruchamiania serwerów komend. Aby wykonać ten typ zadania, należy zalogować się do systemu zdalnego i wydać z niego komendy lub utworzyć proces, który będzie mógł wydawać komendy dla użytkownika. To ograniczenie ma zastosowanie również do IBM MQ Explorer.

Produkt [“Administrowanie zdalnymi obiektami IBM MQ”](#) na stronie 196 zawiera szczegółowy opis tematu zdalnego administrowania.

ULW Administrowanie programem IBM MQ za pomocą komend sterujących

Komendy sterujące umożliwiają wykonywanie wielu zadań administracyjnych serwera IBM MQ w systemie UNIX, Linux, and Windows.

Jeśli wymagane jest wydanie komend sterujących, ID użytkownika musi należeć do grupy mqm dla większości komend sterujących. Więcej informacji na ten temat zawiera sekcja [Uprawnienia do administrowania produktem IBM MQ w systemie UNIX, Linux, and Windows](#). Ponadto należy zwrócić uwagę na informacje specyficzne dla środowiska, dla platformy lub platform, używane są przedsiębiorstwa.

Podczas korzystania z komend sterujących, które działają w menedżerze kolejek, należy użyć komendy z instalacji powiązanej z menedżerem kolejek, z którym pracuje użytkownik.

Podczas korzystania z komend sterujących, które działają w menedżerze kolejek skonfigurowanym do używania uwierzytelniania połączenia z CHCKLOCL (REQUIRED), i obserwuje się niepowodzenie połączenia,

- Podaj ID użytkownika i hasło, jeśli ta komenda sterująca pozwala na to.
- Użyj MQSC równoważników komend sterujących, w których istnieją.
- Uruchom menedżer kolejek przy użyciu opcji -ns, podczas gdy komendy sterujące, które nie mogą nawiązać połączenia, muszą zostać uruchomione.

Pełna lista komend sterujących znajduje się w sekcji [Komendy sterujące programu IBM MQ](#).

Używanie komend sterujących w systemach Windows

Windows

W programie IBM MQ for Windows komendy sterujące są wprowadzane w wierszu komend.

W komendach sterujących i ich flagach nie jest rozróżniana wielkość liter, ale argumenty tych komend, takie jak nazwy kolejek i nazwy menedżerów kolejek, są rozróżniane wielkości liter.

Na przykład w komendzie:

```
crtmqm /u SYSTEM.DEAD.LETTER.QUEUE jupiter.queue.manager
```

- Nazwę komendy można wprowadzić wielkimi lub małymi literami lub mieszaniną tych dwóch znaków. Są to wszystkie poprawne wartości: crtmqm, CRTMQM i CRTmqm.
- Flagę można wprowadzić jako -u, -U, /ulub /U.
- SYSTEM.DEAD.LETTER.QUEUE i jupiter.queue.manager muszą być wprowadzane dokładnie tak, jak pokazano.

Korzystanie z komend sterujących w systemach UNIX i Linux

Linux → UNIX

W systemach IBM MQ for UNIX i Linux komendy sterujące są wprowadzane w oknie powłoki.

W środowiskach UNIX komendy sterujące, w tym samą nazwę komendy, opcje i wszystkie argumenty, są rozróżniane wielkośćmi liter. Na przykład w komendzie:

```
crtmqm -u SYSTEM.DEAD.LETTER.QUEUE jupiter.queue.manager
```

- Nazwa komendy musi mieć wartość `crtmqm`, a nie `CRTMQM`.
- Flaga musi mieć wartość `-u`, a nie `-U`.
- Kolejka niedostarczonych komunikatów nosi nazwę `SYSTEM.DEAD.LETTER.QUEUE`.
- Argument jest określany jako `jupiter.queue.manager`, który różni się od `JUPITER.queue.manager`.

Zadbaj o to, aby wpisać komendy dokładnie tak, jak w nich znajdują się przykłady.

Informacje pokrewne

[Skorowidz komend sterujących programu IBM MQ](#)

Administrowanie programem MQ za pomocą komend MQSC

Jak używać komend MQSC do wykonywania typowych zadań.

Komendy MQSC są dostępne na wszystkich platformach, w tym IBM i z/OS.

Za pomocą komend MQSC można zarządzać obiektami menedżera kolejek, w tym samym menedżerem kolejek, kolejkami, definicjami procesów, kanałami, kanałami połączeń klientów, obiektami nastuchiwania, usługami, listami nazw, klastrami i obiektami informacji uwierzytelniających. Ta sekcja dotyczy menedżerów kolejek, kolejek i definicji procesów. W celu zapoznania się z przeglądem kanału, kanału połączenia klienta i obiektów nastuchiwania, należy zapoznać się z sekcją [Obiekty](#). Więcej informacji na temat wszystkich komend MQSC służących do zarządzania obiektami menedżera kolejek zawiera sekcja [“Komendy skryptowe \(MQSC\)”](#) na stronie 11.

Komendy MQSC są emitowane do menedżera kolejek za pomocą komendy `runmqsc`. Szczegółowe informacje na temat tej komendy można znaleźć w sekcji `runmqsc`. Można to zrobić interaktywnie, wydając komendy za pomocą klawiatury lub można przekierować standardowe urządzenie wejściowe (`stdin`), aby uruchomić sekwencję komend z pliku tekstowego ASCII. W obu przypadkach format komend jest taki sam. (Informacje na temat uruchamiania komend z pliku tekstowego można znaleźć w sekcji [“Uruchamianie komend MQSC z plików tekstowych”](#) na stronie 16).

Komendę `runmqsc` można uruchomić na trzy sposoby, w zależności od opcji ustawionych w komendzie:

- Sprawdź, czy komenda nie została uruchomiona, gdzie komendy MQSC są weryfikowane w lokalnym menedżerze kolejek, ale nie są wykonywane.
- Uruchom komendę w lokalnym menedżerze kolejek, w którym komendy MQSC są uruchamiane w lokalnym menedżerze kolejek.
- Uruchom komendę w zdalnym menedżerze kolejek, w którym komendy MQSC są uruchamiane w zdalnym menedżerze kolejek.

Można również uruchomić komendę, po której następuje znak zapytania, aby wyświetlić składnię.

Atrybuty obiektów określone w komendach MQSC są wyświetlane w tej sekcji wielkimi literami (na przykład `RQMNAME`), chociaż nie są one rozróżniane wielkośćmi liter. Nazwy atrybutów komend MQSC są ograniczone do ośmiu znaków.

V 9.0.1

W przypadku produktu Continuous Deliveryz poziomu produktu IBM MQ 9.0.1można ustawić pytanie o wybór przy użyciu zmiennej środowiskowej `MQPROMPT`. Oprócz zwykłego tekstu, zmienna `MQPROMPT` umożliwia również wstawianie zmiennych środowiskowych za pomocą programu

+VARNAME+ notation w taki sam sposób, jak definicje obiektów usług produktu IBM MQ (patrz “Definiowanie obiektu usługi” na stronie 184). Na przykład:

```
sh> export MQPROMPT="+USER+ @ +QMNAME+ @ +MQ_HOST_NAME+> "  
sh> runmqsc MY.QMGR  
5724-H72 (C) Copyright IBM Corp. 1994, 2023.  
Starting MQSC for queue manager MY.QMGR.  
username @ MY.QMGR @ aix1> DISPLAY QMSTATUS
```

V9.0.0.1 W produkcie IBM MQ 9.0.0 Fix Pack 1 zmienna środowiskowa **MQPROMPT** jest również dostępna w wersji Long Term Support .

Komendy MQSC są szczegółowo opisane w sekcji [Komendy MQSC](#) .

Windows **Linux** Na platformach Windows lub Linux (platformy x86 i x86-64) można również wykonywać operacje opisane w tej sekcji za pomocą konsoli IBM MQ Explorer. Więcej informacji na ten temat zawiera sekcja “Administrowanie przy użyciu IBM MQ Explorer” na stronie 134.

Informacje pokrewne

[runmqsc \(uruchamianie komend MQSC\)](#)

[Skorowidz komend MQSC](#)

Komendy skryptowe (MQSC)

Komendy MQSC udostępniają jednolitą metodę wydawania czytelnych komend na platformach IBM MQ .

Ogólny format komend jest opisany w sekcji [Komendy MQSC](#).

Podczas korzystania z komend MQSC należy przestrzegać następujących reguł:

- Każda komenda zaczyna się od parametru podstawowego (czasownika), a po nim następuje dodatkowy parametr (rzeczownik). Następnie następuje po nim nazwa lub ogólna nazwa obiektu (w nawiasach), jeśli istnieje jedna, która znajduje się w większości komend. W związku z tym, parametry mogą zwykle występować w dowolnej kolejności; jeśli parametr ma odpowiednią wartość, wartość musi być wykonywana bezpośrednio po parametrze, do którego się odnosi.

Uwaga: **z/OS** W systemie z/OS parametr dodatkowy nie musi być drugi.

- Słowa kluczowe, nawiasy i wartości mogą być rozdzielane dowolną liczbą odstępów i przecinków. Przecinek wyświetlany na diagramach składniowych zawsze może być zastąpiony przez co najmniej jeden znak odstępu. Musi istnieć co najmniej jedno puste pole bezpośrednio poprzedzające każdy parametr (po parametrze podstawowym) z wyjątkiem z/OS .
- Na początku lub na końcu komendy może wystąpić dowolna liczba odstępów oraz między parametrami, interpunkcją i wartościami. Na przykład poprawna jest następująca komenda:

```
ALTER QLOCAL ('Account' ) TRIGDPTH ( 1)
```

Odstępy w obrębie pary znaków cudzysłowu są znaczące.

- Dodatkowe przecinki mogą pojawiać się w dowolnym miejscu, gdzie dozwolone są odstępy i są traktowane tak, jakby były odstępami (chyba, że są one oczywiście wewnątrz łańcuchów ujętych w cudzysłow).
- Powtarzające się parametry nie są dozwolone. Powtarzanie parametru z jego wersją "NO", tak jak w REPLACE NOREPLACE, również nie jest dozwolone.
- Łańcuchy, które zawierają spacje, małe litery lub znaki specjalne inne niż:
 - Kropka (.)
 - Prawy ukośnik (/)
 - Podkreślenie (_)
 - Znak procentu (%)

muszą być ujęte w apostrofach, o ile nie są:

- **z/OS** Wydane z poziomu paneli sterujących i sterujących IBM MQ for z/OS
- Wartości ogólne kończące się gwiazdką (na IBM i te muszą być ujęte w pojedynczy cudzysłów)
- Pojedyncza gwiazdka, na przykład TRACE (*) (na IBM i te muszą być ujęte w apostrofach)
- Specyfikacja zakresu zawierająca dwukropek, na przykład: KLASA (01:03)

Jeśli sam łańcuch zawiera pojedynczy cudzysłów, pojedynczy cudzysłów jest reprezentowany przez dwa apostrofowe znaki cudzysłowu. Małe litery, które nie są ujęte w cudzysłów, są składane wielkimi literami.

- **Multi** W systemie Multiplatform łańcuch zawierający nie znaki (czyli dwa pojedyncze cudzysłowy bez spacji między) jest interpretowany jako spacja ujęta w znaki pojedynczego cudzysłowu, czyli interpretowana w ten sam sposób co ("). Wyjątkiem jest ten wyjątek, jeśli używany atrybut ma jedną z następujących wartości:

- TOPICSTR
- SUB
- USERDATA
- SELECTOR

wtedy dwa pojedyncze cudzysłowy bez spacji są interpretowane jako łańcuch o zerowej długości.

z/OS W systemie z/OS, jeśli spacja ma być ujęta w znaki pojedynczego cudzysłowu, należy ją wprowadzić jako taką ("). Łańcuch zawierający nie znaki (") jest taki sam, jak wprowadzanie ().

- W programie IBM WebSphere MQ 7.0 wszystkie odstępy końcowe w tych atrybutach łańcuchowych, które są oparte na typach MQCHARV, takie jak SELECTOR, dane użytkownika podrzędne, są traktowane jako znaczące, co oznacza, że 'abc' nie równa się "abc".
- Lewy nawias, po którym następuje prawy nawias, bez istotnych informacji pomiędzy, na przykład

```
NAME ( )
```

jest niepoprawna, z wyjątkiem przypadków, w których wyraźnie zaznaczono.

- W słowach kluczowych nie jest rozróżniana wielkość liter: ALTER, alter, and ALTER są akceptowalne. Wszystko, co nie jest zawarte w cudzysłowie, jest składane na wielkie litery.
- Synonimy są zdefiniowane dla niektórych parametrów. Na przykład, DEF jest zawsze synonimem dla DEFINE, więc DEF QLOCAL jest ważne. Synonimy nie są jednak tylko łańcuchami minimalnymi; DEF1 nie jest poprawnym synonimem dla definicji DEFINE.

Uwaga: Dla parametru DELETE nie ma synonimu. Ma to na celu uniknięcie przypadkowego usunięcia obiektów podczas korzystania z DEF, czyli synonimu dla DEFINE.

Przegląd informacji na temat używania komend MQSC do administrowania produktem IBM MQ zawiera sekcja [“Administrowanie programem MQ za pomocą komend MQSC”](#) na stronie 10.

Komendy MQSC używają określonych znaków specjalnych do określenia określonych znaczeń. Więcej informacji na temat tych znaków specjalnych oraz sposobu ich używania zawiera sekcja [Znaki ze specjalnymi znaczeniami](#).

Aby dowiedzieć się, w jaki sposób można budować skrypty za pomocą komend MQSC, należy zapoznać się z [Budowanie skryptów komend](#).

Pełną listę komend MQSC można znaleźć w sekcji [Komendy MQSC](#).

Tę komendę można wydać z poziomu źródeł 2CR. Wyjaśnienie symboli źródła można znaleźć w sekcji [Używanie komend w systemie z/OS](#).

Pojęcia pokrewne

[“Wprowadzenie do formatów komend programowalnych IBM MQ”](#) na stronie 22

Programowalne formaty komend (PCFs) definiują komunikaty komend i odpowiedzi, które mogą być wymieniane między programem a dowolnym menedżerem kolejek (który obsługuje systemy PCF) w sieci. PCF upraszczają administrację menedżera kolejek i inne administrowanie siecią. Mogą one być wykorzystywane do rozwiązywania problemu skomplikowanej administracji rozproszonych sieci, zwłaszcza, gdy sieci rosną w wielkości i złożoności.

Informacje pokrewne

[Budowanie skryptów komend](#)

Nazwy obiektów produktu IBM MQ w komendach MQSC

Informacje na temat używania nazw obiektów w komendach MQSC.

W przykładach używamy niektórych długich nazw obiektów. Jest to pomocne w określeniu typu obiektu, z którym użytkownik ma do czynienia.

Podczas wydawania komend MQSC konieczne jest podanie tylko nazwy lokalnej kolejki. W naszych przykładach używamy nazw kolejek takich jak:

```
ORANGE . LOCAL . QUEUE
```

Część nazwy LOCAL . QUEUE ma na celu zilustrować, że ta kolejka jest kolejką lokalną. W przypadku nazw kolejek lokalnych **nie** jest wymagane w przypadku nazw kolejek lokalnych.

Nazwa saturn . queue . manager jest również używana jako nazwa menedżera kolejek. Część nazwy queue . manager ma na celu zilustrować, że ten obiekt jest menedżerem kolejek. Nie jest on wymagany w przypadku nazw menedżerów kolejek w ogóle.

Rozróżnianie wielkości liter w komendach MQSC

Komendy MQSC, w tym ich atrybuty, mogą być zapisywane wielkimi lub małymi literami. Nazwy obiektów w komendach MQSC są składane na wielkie litery (to znaczy w kolejce i w kolejce nie są rozróżniane), chyba że nazwy są ujęte w znaki pojedynczego cudzysłowu. Jeśli znaki cudzysłowu nie są używane, obiekt jest przetwarzany z nazwą zapisanej wielkimi literami. Więcej informacji na ten temat zawiera sekcja [Znaki ze specjalnymi znaczeniami](#).

Wywołanie komendy runmqsc, wspólne dla wszystkich komend sterujących IBM MQ, rozróżnia wielkość liter w niektórych środowiskach IBM MQ. Więcej informacji na ten temat zawiera sekcja ["Administrowanie programem IBM MQ za pomocą komend sterujących"](#) na stronie 9.

Standardowe wejście i wyjście

Standardowe urządzenie wejściowe, zwane również stdin, jest urządzeniem, z którego pobierane są dane wejściowe do systemu. Zwykle jest to klawiatura, ale można określić, że dane wejściowe mają pochodzić z portu szeregowego lub z pliku na dysku, np. *Standardowe urządzenie wyjściowe*, zwane również stdout, jest urządzeniem, do którego wysyłane są dane wyjściowe z systemu. Zwykle jest to ekran, ale można przekierować dane wyjściowe do portu szeregowego lub pliku.

W komendach systemu operacyjnego i IBM MQ komend sterujących < operator przekierowuje dane wejściowe. Jeśli po tym operatorze występuje nazwa pliku, dane wejściowe są pobierane z pliku. Podobnie, > operator przekierowuje dane wyjściowe; jeśli po nim następuje nazwa pliku, dane wyjściowe są kierowane do tego pliku.

Interaktywna obsługa komend MQSC

Komendy MQSC można używać interaktywnie, korzystając z okna komend lub powłoki.

Aby używać komend MQSC interaktywnie, otwórz okno komend lub powłokę i wpisz:

```
runmqsc
```

W tej komendzie nie określono nazwy menedżera kolejek, dlatego komendy MQSC są przetwarzane przez domyślny menedżer kolejek. Jeśli ma być używany inny menedżer kolejek, należy podać nazwę menedżera kolejek w komendzie **runmqsc** . Na przykład, aby uruchomić komendy MQSC dla menedżera kolejek `jupiter.queue.manager`, należy użyć komendy:

```
runmqsc jupiter.queue.manager
```

Po wykonaniu tej czynności wszystkie typy komend MQSC używane przez użytkownika są przetwarzane przez ten menedżer kolejek, zakładając, że jest on w tym samym węźle i jest już uruchomiony.

Teraz można wpisać dowolne komendy MQSC, zgodnie z wymaganiami. Na przykład spróbuj wykonać następujące czynności:

```
DEFINE QLOCAL (ORANGE.LOCAL.QUEUE)
```

W przypadku komend, które mają zbyt wiele parametrów, aby zmieścić się w jednym wierszu, należy użyć znaków kontynuacji w celu wskazania, że komenda jest kontynuowana w następującym wierszu:

- Znak minus (-) wskazuje, że komenda ma być kontynuowana od początku następnego wiersza.
- Znak plus (+) wskazuje, że komenda ma być kontynuowana od pierwszego niepustego znaku w następnym wierszu.

Wejście komendy kończy się znakiem finałowym niepustej linii, która nie jest znakiem kontynuacji. Dane wejściowe komendy można również zakończyć jawnie, wprowadzając średnik (;). (Jest to szczególnie przydatne w przypadku przypadkowego wprowadzenia znaku kontynuacji na końcu ostatniego wiersza wprowadzania komendy).

Informacje zwrotne z komend MQSC

Po wydaniu komend MQSC menedżer kolejek zwraca komunikaty operatora, które potwierdzają działania użytkownika, lub informacje o błędach, które zostały wykonane. Na przykład:

```
AMQ8006: IBM MQ queue created.
```

Ten komunikat potwierdza, że kolejka została utworzona.

```
AMQ8405: Syntax error detected at or near end of command segment below:-
```

```
AMQ8426: Valid MQSC commands are:
```

```
ALTER  
CLEAR  
DEFINE  
DELETE  
DISPLAY  
END  
PING  
REFRESH  
RESET  
RESOLVE  
RESUME  
START  
STOP  
SUSPEND  
4 : end
```

Ten komunikat wskazuje, że wystąpił błąd składniowy.

Komunikaty te są wysyłane do standardowego urządzenia wyjściowego. Jeśli komenda nie została poprawnie wprowadzona, należy zapoznać się z poprawnymi składnikami w sekcji [Komendy MQSC](#) .

Zakończenie interaktywnego wprowadzania komend MQSC

Aby zakończyć pracę z komendami MQSC, należy wprowadzić komendę END.

Alternatywnie można użyć znaku EOF dla danego systemu operacyjnego.

Pojęcia pokrewne

“Uruchamianie komend MQSC z plików tekstowych” na stronie 16

Uruchamianie komend MQSC interaktywnie jest odpowiednie do szybkich testów, ale jeśli użytkownik ma bardzo długie komendy lub wielokrotnie używa określonej sekwencji komend, należy rozważyć przekierowanie programu stdin z pliku tekstowego.

Informacje pokrewne

[runmqsc](#)



Komenda runmqsc w systemach UNIX i Linux

Wiersz komend **runmqsc** w systemach UNIX i Linux obsługuje przywracanie komend, zakończenie komend oraz klawisze komend Emacs.

Dostępne są następujące funkcje edytora wiersza komend:

- Przypomnij poprzednio wprowadzone komendy za pomocą klawisza strzałki w górę i klawisza strzałki w dół
- Automatyczne uzupełnianie dla następnego słowa kluczowego komendy przy użyciu klawisza Tab i klawisza spacji
- Klawisze komend Emacs lub podobne funkcje klawiszy funkcji.

Aby można było korzystać z tych funkcji, należy zainstalować bibliotekę curses. Jeśli biblioteka curses nie jest zainstalowana w systemie, program **runmqsc** nie będzie miał funkcji edytora wiersza komend, a po uruchomieniu wiersza komend **runmqsc** zostanie wyświetlony komunikat. Nazwa biblioteki curses do zainstalowania zależy od platformy UNIX :

-  W systemie AIX zainstaluj produkt curses .
-  W systemie HP-UX zainstaluj produkt Xcurses .
- Na wszystkich innych platformach UNIX i w systemie Linux zainstaluj produkt ncurses .

Dostosowywanie przypisań klawiszy Emacs

Istnieje możliwość dostosowania kluczy, które są powiązane z komendami. Na przykład można powiązać klucze z powiązaniem edytora vi zamiast domyślnych powiązań kluczy Emacs.

Klucze są dostosowywane, edytując plik `.editrc` zapisany w katalogu osobistym. Więcej informacji na ten temat zawiera sekcja [editrc](#) na stronach podręcznika FreeBSD .

Wyłączanie opcji przywracania komend, uzupełniania komend i klawiszy komend Emacs

Ustawiając zmienną środowiskową, można wyłączyć opcję przywracania komend, zakończenia komend i klawiszy komend Emacs. Ustaw zmienną środowiskową `MQ_OVERRIDE_LIBEDIT_LOAD` na TRUE.

Ta zmienna środowiskowa może być używana jako obejście, gdy program **runmqsc** wyświetla następujący komunikat informacyjny:

```
AMQ8521I: Command completion and history unavailable
```

Uruchamianie komend MQSC z plików tekstowych

Uruchamianie komend MQSC interaktywnie jest odpowiednie do szybkich testów, ale jeśli użytkownik ma bardzo długie komendy lub wielokrotnie używa określonej sekwencji komend, należy rozważyć przekierowanie programu `stdIn` z pliku tekstowego.

Aby przekierować program `stdIn` z pliku tekstowego, należy najpierw utworzyć plik tekstowy zawierający komendy MQSC, korzystając z zwykłego edytora tekstu, a następnie uruchomić komendę `runmqsc`.

Uwaga: Jeśli komenda `runmqsc` została uruchomiona w trybie klienta przez przekierowanie programu `stdIn` z pliku tekstowego, program IBM MQ oczekuje, że pierwszy wiersz pliku wejściowego będzie hasłem.

Jeśli używana jest komenda `runmqsc`, należy użyć operatorów przekierowania. Na przykład następująca komenda uruchamia sekwencję komend zawartych w pliku tekstowym `myprog.in`:

```
runmqsc < myprog.in
```

Podobnie, można również przekierować dane wyjściowe do pliku. Plik zawierający komendy MQSC dla danych wejściowych nosi nazwę pliku komend MQSC. Plik wyjściowy zawierający odpowiedzi z menedżera kolejek jest nazywany plikiem wyjściowym.

Aby przekierować zarówno `stdIn`, jak i `stdOut` w komendzie `runmqsc`, należy użyć następującej formy komendy:

```
runmqsc < myprog.in > myprog.out
```

Ta komenda wywołuje komendy MQSC zawarte w pliku komend MQSC `myprog.in`. Ponieważ nie określiliśmy nazwy menedżera kolejek, komendy MQSC są uruchamiane dla domyślnego menedżera kolejek. Dane wyjściowe są wysyłane do pliku tekstowego `myprog.out`. [Rysunek 1 na stronie 17](#) przedstawia ekstrakt z pliku komend MQSC `myprog.in` i [Rysunek 2 na stronie 18](#) przedstawia odpowiedni wyciąg danych wyjściowych w programie `myprog.out`.

Aby przekierować `stdIn` i `stdOut` w komendzie `runmqsc`, dla menedżera kolejek (`saturn.queue.manager`), który nie jest domyślny, należy użyć następującej postaci komendy:

```
runmqsc saturn.queue.manager < myprog.in > myprog.out
```

Pliki komend MQSC

Komendy MQSC są zapisywane w postaci czytelnej dla użytkownika, to znaczy w tekście ASCII. [Rysunek 1 na stronie 17](#) to wyciąg z pliku komend MQSC, który przedstawia komendę MQSC (**DEFINE QLOCAL**) z jej atrybutami. [Komendy MQSC](#) zawierają opis każdej komendy MQSC i jej składni.


```

.
.
.
DEFINE QLOCAL(ORANGE.LOCAL.QUEUE) REPLACE +
DESCR(' ') +
PUT(ENABLED) +
DEFPRTY(0) +
DEFPSIST(NO) +
GET(ENABLED) +
MAXDEPTH(5000) +
MAXMSGL(1024) +
DEFSOPT(SHARED) +
NOHARDENBO +
USAGE(NORMAL) +
NOTRIGGER;
.
.
.

```

Rysunek 1. Wyodrębnij z pliku komend MQSC

W celu zapewnienia przenośności w środowiskach IBM MQ należy ograniczyć długość wiersza w plikach komend MQSC do 72 znaków. Znak plus wskazuje, że komenda jest kontynuowana w następnym wierszu.

Raporty komend MQSC

Komenda **runmqsc** zwraca raport, który jest wysyłany do programu stdout. Raport zawiera:

- Nagłówek identyfikujący komendy MQSC jako źródło raportu:

```
Starting MQSC for queue manager jupiter.queue.manager.
```

Gdzie `jupiter.queue.manager` to nazwa menedżera kolejek.

- Opcjonalna numerowana lista wydanych komend MQSC. Domyślnie tekst danych wejściowych jest odbity na wyjściu. W tym wyniku każda komenda jest poprzedzona numerem kolejnym, jak pokazano na [Rysunek 2 na stronie 18](#). Można jednak użyć opcji **-e** w komendzie **runmqsc** w celu zablokowania danych wyjściowych.
- Komunikat o błędzie składni dla wszystkich komend, które mogą być błędnie używane.
- Komunikat operatora wskazujący wynik działania każdej komendy. Na przykład komunikat operatora dla pomyślnego zakończenia komendy **DEFINE QLOCAL** jest następujący:

```
AMQ8006: IBM MQ queue created.
```

- Inne komunikaty wynikające z błędów ogólnych podczas uruchamiania pliku skryptowego.
- Krótkie statystyczne podsumowanie raportu, wskazujące liczbę odczytanych komend, liczbę komend z błędami składniową oraz liczbę komend, które nie mogły zostać przetworzone.

Uwaga: Menedżer kolejek próbuje przetworzyć tylko te komendy, które nie zawierają błędów składniowych.

```

Starting MQSC for queue manager jupiter.queue.manager.
.
.
12:  DEFINE QLOCAL('ORANGE.LOCAL.QUEUE') REPLACE +
:    DESCR(' ') +
:    PUT(ENABLED) +
:    DEFPRTY(0) +
:    DEFPSIST(NO) +
:    GET(ENABLED) +
:    MAXDEPTH(5000) +
:    MAXMSGL(1024) +
:    DEFSOPT(SHARED) +
:    NOHARDENBO +
:    USAGE(NORMAL) +
:    NOTRIGGER;
AMQ8006: IBM MQ queue created.
.
.
.

```

Rysunek 2. Wyodrębnij z pliku raportu komend MQSC

Uruchamianie dostarczonych plików komend MQSC

Z produktem IBM MQ dostarczane są następujące pliki komend MQSC:

amqscos0.tst

Definicje obiektów używanych przez programy przykładowe.

amqscic0.tst

Definicje kolejek dla transakcji CICS .

Windows W programie IBM MQ for Windows pliki te znajdują się w katalogu `MQ_INSTALLATION_PATH\tools\mqsc\samples`. `MQ_INSTALLATION_PATH` reprezentuje katalog najwyższego poziomu, w którym zainstalowany jest produkt IBM MQ .

Linux **UNIX** W systemach UNIX and Linux te pliki znajdują się w katalogu `MQ_INSTALLATION_PATH/samp`. `MQ_INSTALLATION_PATH` reprezentuje katalog najwyższego poziomu, w którym zainstalowany jest produkt IBM MQ .

Komenda, która je uruchamia, jest następująca:

```
runmqsc < amqscos0.tst >test.out
```

Używanie komendy runmqsc do weryfikowania komend

Komendy **runmqsc** można użyć do sprawdzenia komend MQSC w lokalnym menedżerze kolejek bez ich rzeczywistego uruchamiania. Aby to zrobić, należy ustawić flagę **-v** w komendzie **runmqsc** , na przykład:

```
runmqsc -v < myprog.in > myprog.out
```

Po wywołaniu programu **runmqsc** w odniesieniu do pliku komend MQSC menedżer kolejek weryfikuje każdą komendę i zwraca raport bez wykonywania komend MQSC. Pozwala to na sprawdzenie składni komend w pliku komend. Jest to szczególnie ważne w przypadku, gdy:

- Uruchamianie dużej liczby komend z pliku komend.
- Korzystanie z pliku komend MQSC wiele razy.

Zwrócony raport jest podobny do przedstawionego w sekcji [Rysunek 2 na stronie 18](#).

Tej metody nie można używać do zdalnego sprawdzania komend MQSC. Na przykład, jeśli podejmowana jest próba wykonania tej komendy:

```
runmqsc -w 30 -v jupiter.queue.manager < myprog.in > myprog.out
```

Opcja **-w**, używana do wskazywania, że menedżer kolejek jest zdalna, jest ignorowana, a komenda jest uruchamiana lokalnie w trybie weryfikacji. Wartość 30 to liczba sekund, przez które program IBM MQ oczekuje na odpowiedzi ze zdalnego menedżera kolejek.

Pojęcia pokrewne

[“Standardowe wejście i wyjście” na stronie 13](#)

Standardowe urządzenie wejściowe, zwane również `stdin`, jest urządzeniem, z którego pobierane są dane wejściowe do systemu. Zwykle jest to klawiatura, ale można określić, że dane wejściowe mają pochodzić z portu szeregowego lub z pliku na dysku, np. *Standardowe urządzenie wyjściowe*, zwane również `stdout`, jest urządzeniem, do którego wysyłane są dane wyjściowe z systemu. Zwykle jest to ekran, ale można przekierować dane wyjściowe do portu szeregowego lub pliku.

[“Interaktywna obsługa komend MQSC” na stronie 13](#)

Komendy MQSC można używać interaktywnie, korzystając z okna komend lub powłoki.

Informacje pokrewne

[runmqsc](#)

Uruchamianie komend MQSC z plików wsadowych

Jeśli użytkownik ma bardzo długie komendy lub wielokrotnie używa określonej sekwencji komend, należy rozważyć przekierowanie produktu `stdin` z pliku wsadowego.

Aby przekierować program `stdin` z pliku wsadowego, należy najpierw utworzyć plik wsadowy zawierający komendy MQSC, korzystając z zwykłego edytora tekstu. Jeśli używana jest komenda `runmqsc`, należy użyć operatorów przekierowania. Poniższy przykład:

1. Tworzy menedżera kolejek testowych, TESTQM
2. Tworzy zgodny zestaw CLNTCONN i zestaw nasłuchujący, aby użyć portu TCP/IP 1600
3. Tworzy kolejkę testową TESTQ
4. Umieszcza komunikat w kolejce przy użyciu przykładowego programu `amqspc`.

```
export MYTEMPQM=TESTQM
export MYPOR=1600
export MQCHLLIB=/var/mqm/qmgrs/$MQTEMPQM/@ipcc

crtmqm $MYTEMPQM
stimqm $MYTEMPQM
runmqslsr -m $MYTEMPQM -t TCP -p $MYPOR &

runmqsc $MYTEMPQM << EOF
DEFINE CHANNEL(NLTM) CHLTYPE(SVRCONN) TRPTYPE(TCP)
DEFINE CHANNEL(NLTM) CHLTYPE(CLNTCONN) QMNAME('$MYTEMPQM') CONNAME('hostname($MYPOR)')
ALTER CHANNEL(NLTM) CHLTYPE(CLNTCONN)
DEFINE QLOCAL(TESTQ)
EOF

amqspc TESTQ $MYTEMPQM << EOF
hello world
EOF

endmqm -i $MYTEMPQM
```

Rysunek 3. Przykładowy skrypt do uruchamiania komend MQSC z pliku wsadowego

Rozwiązywanie problemów za pomocą komend MQSC

Jeśli nie można pobrać komend MQSC do uruchomienia, należy zapoznać się z informacjami w tym temacie, aby sprawdzić, czy jakiegokolwiek z tych typowych problemów dotyczy użytkownika. Nie zawsze jest oczywiste, czym jest problem, gdy czytasz błąd generowany przez komendę.

Jeśli używana jest komenda `runmqsc`, należy pamiętać o następujących przypadkach:

- Użyj operatora `<`, aby przekierować dane wejściowe z pliku. Jeśli ten operator zostanie pominięty, menedżer kolejek interpretuje nazwę pliku jako nazwę menedżera kolejek i wysyła następujący komunikat o błędzie:

```
AMQ8118E: IBM MQ queue manager does not exist.
```

- Jeśli dane wyjściowe są przekierowane do pliku, należy użyć operatora przekierowania `>`. Domyślnie plik jest umieszczany w bieżącym katalogu roboczym w momencie wywołania komendy `runmqsc`. Podaj pełną nazwę pliku, aby wysłać dane wyjściowe do określonego pliku i katalogu.
- Sprawdź, czy został utworzony menedżer kolejek, który ma uruchamiać te komendy, za pomocą następującej komendy w celu wyświetlenia wszystkich menedżerów kolejek:

```
dspmq
```

- Menedżer kolejek musi być uruchomiony. Jeśli tak nie jest, uruchom go; (patrz [Uruchamianie menedżera kolejek](#)). W przypadku próby uruchomienia menedżera kolejek, który jest już uruchomiony, wyświetlany jest komunikat o błędzie.
- Podaj nazwę menedżera kolejek w komendzie `runmqsc`, jeśli nie zdefiniowano domyślnego menedżera kolejek, lub jeśli wystąpi ten błąd:

```
AMQ8146E: IBM MQ queue manager not available.
```

- Nie można określić komendy MQSC jako parametru komendy `runmqsc`. Na przykład nie jest to poprawne:

```
runmqsc DEFINE QLOCAL(FRED)
```

- Nie można wprowadzić komend MQSC przed wydaniem komendy **runmqsc**.
- Nie można uruchamiać komend sterujących z programu **runmqsc**. Na przykład nie można wydać komendy `strmqm` w celu uruchomienia menedżera kolejek w czasie, gdy komendy MQSC są uruchamiane interaktywnie. W takim przypadku zostaną wyświetlone komunikaty o błędach podobne do poniższych:

```
runmqsc
.
.
Starting MQSC for queue manager jupiter.queue.manager.

1 : strmqm saturn.queue.manager
AMQ8405: Syntax error detected at or near end of cmd segment below:-s

AMQ8426: Valid MQSC commands are:
ALTER
CLEAR
DEFINE
DELETE
DISPLAY
END
PING
REFRESH
RESET
RESOLVE
RESUME
START
STOP
```

Automatyzacja administrowania produktem IBM MQ przy użyciu komend PCF

Użytkownik może zdecydować, że będzie on korzystny dla danej instalacji w celu zautomatyzowania niektórych zadań administracyjnych i monitorowania. Można zautomatyzować zadania administracyjne zarówno dla lokalnych, jak i zdalnych menedżerów kolejek za pomocą komend PCF (programmable command format). W tej sekcji założono, że użytkownik posiada doświadczenie związane z administrowaniem obiektami produktu IBM MQ .

Komendy PCF

Komendy programu IBM MQ programmable command format (PCF) mogą być używane do uruchamiania zadań administracyjnych w programie administracyjnym. W ten sposób z poziomu programu można manipulować obiektami menedżera kolejek (kolejkami, definicjami procesów, listami nazw, kanałami, kanałami połączeń klientów, obiektami nasłuchiwania, usługami i obiektami informacji uwierzytelniających), a nawet samodzielnie operować menedżerami kolejek.

Komendy PCF obejmują ten sam zakres funkcji udostępnianych przez komendy MQSC. Można napisać program, aby wydać komendy PCF do dowolnego menedżera kolejek w sieci z jednego węzła. W ten sposób można zarówno scentralizować, jak i automatyzować zadania administracyjne.

Każda komenda PCF jest strukturą danych osadzoną w części danych aplikacji komunikatu produktu IBM MQ . Każda komenda jest wysyłana do docelowego menedżera kolejek za pomocą funkcji MQI MQPUT w taki sam sposób, jak w przypadku innych komunikatów. Jeśli serwer komend jest uruchomiony w menedżerze kolejek odbierającym komunikat, serwer komend interpretuje go jako komunikat komendy i uruchamia komendę. Aby uzyskać odpowiedzi, aplikacja wydaje wywołanie MQGET , a dane odpowiedzi są zwracane w innej strukturze danych. Następnie aplikacja może przetworzyć odpowiedź i działać odpowiednio.

Uwaga: W przeciwieństwie do komend MQSC, komendy PCF i ich odpowiedzi nie są w formacie tekstowym, który można odczytać.

Krótko mówiąc, są to niektóre z elementów potrzebnych do utworzenia komunikatu komendy PCF:

deskryptor komunikatu

Jest to standardowy deskryptor komunikatu produktu IBM MQ , w którym:

- Typ komunikatu (*MsgType*) to MQMT_REQUEST.
- Format komunikatu (*Format*) to MQFMT_ADMIN.

Dane aplikacji

Zawiera komunikat PCF zawierający nagłówek PCF, w którym:

- Typ komunikatu PCF (*Type*) określa komendę MQCFT_COMMAND.
- Identyfikator komendy określa komendę, na przykład *Change Queue* (MQCMD_CHANGE_Q).

Pełny opis struktur danych PCF i sposób ich implementowania zawiera sekcja [“Wprowadzenie do formatów komend programowalnych IBM MQ” na stronie 22.](#)

Atrybuty obiektu PCF

Atrybuty obiektów w programie PCF nie są ograniczone do ośmiu znaków, ponieważ są przeznaczone dla komend MQSC. Są one przedstawione w tym przewodniku kursywą. Na przykład odpowiednikiem PCF w produkcie RQMNAME jest *RemoteQMgrName*.

Zmiany w systemach PCF

Zmiany PCF to komendy PCF, które zawierają komendy MQSC w tekście komunikatu. Komendy PCF można używać do wysyłania komend do zdalnego menedżera kolejek. Więcej informacji na temat systemów PCF o zmienionym znaczeniu zawiera sekcja [Escape\(Escape\)](#).

Wprowadzenie do formatów komend programowalnych IBM MQ

Programowalne formaty komend (PCFs) definiują komunikaty komend i odpowiedzi, które mogą być wymieniane między programem a dowolnym menedżerem kolejek (który obsługuje systemy PCF) w sieci. PCF upraszczają administrację menedżera kolejek i inne administrowanie siecią. Mogą one być wykorzystywane do rozwiązywania problemu skomplikowanej administracji rozproszonych sieci, zwłaszcza, gdy sieci rosną w wielkości i złożoności.

Formaty programowalnych komend są obsługiwane przez:

- ▶ **AIX** IBM MQ for AIX
- ▶ **HP-UX** IBM MQ for HP-UX
- ▶ **IBM i** IBM MQ for IBM i
- ▶ **Linux** IBM MQ for Linux
- ▶ **Solaris** IBM MQ for Solaris
- ▶ **Windows** IBM MQ for Windows
- ▶ **z/OS** IBM MQ for z/OS

Problem z komendami PCF

Administrowanie sieciami rozproszonymi może stać się skomplikowane. Problemy administracji w dalszym ciągu rosną w miarę wzrostu i złożoności sieci.

Przykłady administrowania specyficzne dla przesyłania komunikatów i kolejkowania obejmują:

- Zarządzanie zasobami.

Na przykład tworzenie kolejek i usuwanie.

- Monitorowanie wydajności.

Na przykład: maksymalna głębokość kolejki lub szybkość komunikatów.

- .

Na przykład: strojenie parametrów kolejki, takich jak maksymalna głębokość kolejki, maksymalna długość komunikatu oraz włączanie i wyłączenie kolejek.

- Kierowanie komunikatów.

Definicja tras alternatywnych poprzez sieć.

Komendy IBM MQ PCF mogą być używane do uproszczenia administrowania menedżerem kolejek i innej administracji sieciowej. Komendy PCF umożliwiają korzystanie z jednej aplikacji w celu wykonania administrowania siecią z jednego menedżera kolejek w sieci.

Co to są PCFs?

Systemy PCF definiują komunikaty komend i odpowiedzi, które mogą być wymieniane między programem a dowolnym menedżerem kolejek (który obsługuje systemy PCF) w sieci. Komendy PCF można używać w aplikacji do zarządzania systemami w celu administrowania obiektami produktu IBM MQ : obiekty informacji uwierzytelniającej, kanały, obiekty nasłuchiwanie kanałów, listy nazw, definicje procesów, menedżery kolejek, kolejki, usługi i klasy pamięci masowej. Aplikacja może działać z jednego punktu

w sieci, aby komunikować informacje o komendach i odpowiedziach z dowolnym menedżerem kolejek, lokalnym lub zdalnym za pomocą lokalnego menedżera kolejek.


Każdy menedżer kolejek ma kolejkę administracyjną o standardowej nazwie kolejki, a aplikacja może wysyłać komunikaty komend PCF do tej kolejki. Każdy menedżer kolejek posiada również serwer komend do obsługi komunikatów komend z kolejki administracyjnej. Komunikaty komend PCF mogą więc być przetwarzane przez dowolny menedżer kolejek w sieci, a dane odpowiedzi mogą być zwracane do aplikacji przy użyciu określonej kolejki odpowiedzi. Komendy PCF i komunikaty odpowiedzi są wysyłane i odbierane przy użyciu normalnego interfejsu kolejki komunikatów (Message Queue Interface-MQI).

Listę dostępnych komend PCF, w tym ich parametrów, można znaleźć w sekcji [Definicje formatów komend programowalnych](#).

Korzystanie z formatów komend programowalnych IBM MQ

Systemów PCF można używać w programie do zarządzania systemami w celu zdalnego administrowania programem IBM MQ .

W tej sekcji zawarto następujące informacje:


- [“Komunikaty komend PCF” na stronie 23](#)
- [“Odpowiedzi PCF w produkcie IBM MQ” na stronie 26](#)
-  [“Rozszerzone odpowiedzi” na stronie 28](#)
- [Reguły dotyczące nazewnictwa obiektów produktu IBM MQ](#)
- [“Sprawdzanie uprawnień dla komend PCF w produkcie IBM MQ” na stronie 29](#)


Komunikaty komend PCF

Komunikaty komend PCF składają się z nagłówka PCF, parametrów zidentyfikowanych w tym nagłówku oraz danych komunikatu zdefiniowanych przez użytkownika. Komunikaty są wysyłane za pomocą wywołań interfejsu kolejki komunikatów.

Każda komenda i jej parametry są wysyłane jako osobny komunikat komendy zawierający nagłówek PCF, po którym następuje pewna liczba struktur parametrów. Szczegółowe informacje na temat nagłówka PCF można znaleźć w sekcji [MQCFH-PCF header](#), a na przykład w strukturze parametrów można znaleźć w sekcji [MQCFST-PCF string parameter](#)(MQCFST-parametr łańcucha PCF). Nagłówek PCF identyfikuje komendę i liczbę struktur parametrów, które są zgodne w tym samym komunikacie. Każda struktura parametru udostępnia parametr komendy.

Odpowiedzi na komendy wygenerowane przez serwer komend mają podobną strukturę. Istnieje nagłówek PCF, po którym następuje liczba struktur parametrów. Odpowiedzi mogą składać się z więcej niż jednego komunikatu, ale komendy zawsze składają się tylko z jednego komunikatu.

 W systemie [Multiplatformskolejka](#), do której wysyłane są komendy PCF, jest zawsze nazywana `SYSTEM.ADMIN.COMMAND.QUEUE`.

 W systemie [z/OS](#) komendy są wysyłane do systemu `SYSTEM.COMMAND.INPUT`, chociaż `SYSTEM.ADMIN.COMMAND.QUEUE` może być aliasem. Serwer komend, który obsługuje tę kolejkę, wysyła odpowiedzi do kolejki zdefiniowanej przez pola `ReplyToQ` i `ReplyToQMgr` w deskrypcji komunikatu komendy.

Jak wystawiać komunikaty komend PCF

Użyj zwykłych wywołań interfejsu MQI (Message Queue Interface), `MQPUT`, `MQGET` i tak dalej, aby umieścić i pobrać komunikaty komend i odpowiedzi PCF do kolejek i ich z nich.

Uwaga:

Upewnij się, że serwer komend jest uruchomiony w docelowym menedżerze kolejek dla komendy PCF, która ma zostać wykonana w tym menedżerze kolejek.

Lista dostarczonych plików nagłówkowych znajduje się w sekcji IBM MQ COPY, header, include and module files.

Deskryptor komunikatu dla komendy PCF

Deskryptor komunikatu produktu IBM MQ jest w pełni udokumentowany w pliku MQMD-deskryptor komunikatu.

Komunikat komendy PCF zawiera następujące pola w deskrytorze komunikatu:

Raport

Poprawna wartość, jeśli jest to wymagane.

MsgType

To pole musi mieć wartość MQMT_REQUEST, aby wskazać komunikat wymagający odpowiedzi.

Utrata ważności

Poprawna wartość, jeśli jest to wymagane.

Opinie

Ustaw na wartość MQFB_NONE

Kodowanie

W przypadku wysyłania do jednego z poniższych systemów należy ustawić to pole na kodowanie używane dla danych komunikatu. W razie potrzeby konwersja jest wykonywana:

-  IBM i
-  Linux
-  UNIX
-  Windows

CodedCharSetId

Jeśli wysyłasz do jednego z poniższych systemów, ustaw to pole na identyfikator kodowanego zestawu znaków używany dla danych komunikatu; w razie potrzeby konwersja jest wykonywana:

-  IBM i
-  Linux
-  UNIX
-  Windows

Format

Ustaw wartość MQFMT_ADMIN.

Priorytet

Poprawna wartość, jeśli jest to wymagane.

Trwałość

Poprawna wartość, jeśli jest to wymagane.

MsgId

Aplikacja wysyłający może określić dowolną wartość lub można określić wartość MQMI_NONE w celu żądania menedżera kolejek w celu wygenerowania unikalnego identyfikatora komunikatu.

CorrelId

Aplikacja wysyłający może określić dowolną wartość, a wartość MQCI_NONE można określić, aby wskazać brak identyfikatora korelacji.

ReplyToQ

Nazwa kolejki, w której ma zostać odebranie odpowiedzi.

ReplyToQMGr

Nazwa menedżera kolejek dla odpowiedzi (lub pusta).

Pola kontekstu komunikatu

W razie potrzeby te pola mogą być ustawione na dowolne poprawne wartości. Zwykle opcja komunikatu umieszczanie MQPMO_DEFAULT_CONTEXT jest używana do ustawiania pól kontekstu komunikatu na wartości domyślne.

Jeśli używana jest struktura MQMD w wersji version-2 , należy ustawić następujące dodatkowe pola:

GroupId

Ustaw na wartość MQGI_NONE

Numer_kolejny_komunikatu

Ustaw na 1

Depozycja

Ustaw na 0

MsgFlags

Ustaw na MQMF_NONE

OriginalLength

Ustaw na wartość MQOL_UNDEFINED

Wysyłanie danych użytkownika

Struktury PCF mogą być również używane do wysyłania danych komunikatów zdefiniowanych przez użytkownika. W takim przypadku pole deskryptora komunikatu *Format* musi być ustawione na wartość MQFMT_PCF.

Wysyłanie i odbieranie komunikatów PCF w określonej kolejce

Wysyłanie komunikatów PCF do określonej kolejki

Aby wysłać komunikat do określonej kolejki, wywołanie komendy mqPutBag przekształca treść określonej torby w komunikat PCF i wysyła komunikat do określonej kolejki. Zawartość worka jest niezmieniona po wywołaniu.

Jako dane wejściowe dla tego wywołania należy podać:

- Uchwyt połączenia MQI.
- Uchwyt obiektu dla kolejki, w której ma być umieszczony komunikat.
- Deskryptor komunikatu. Więcej informacji na temat deskryptora komunikatu zawiera sekcja [MQMD-deskryptor komunikatu](#).
- Umieść opcje komunikatów przy użyciu struktury MQPMO. Więcej informacji na temat struktury MQPMO zawiera sekcja [Opcje komunikatu MQPMO-Put-message](#).
- Uchwyt worka, który ma zostać przekształcony w komunikat.

Uwaga: Jeśli w torbie znajduje się komunikat administracyjny, a do wstawiania wartości do worka użyto wywołania mqAddInquiry, wartość elementu danych MQIASY_COMMAND musi być komendą INQUIRE rozpoznawaną przez interfejs MQAI.

Pełny opis wywołania funkcji mqPutBag znajduje się w sekcji [mqPutBag](#) (Wtorka dla produktu MQ).

Odbieranie komunikatów PCF z określonej kolejki

Aby odebrać komunikat z określonej kolejki, wywołanie funkcji Bag mqGetpobiera komunikat PCF z określonej kolejki i przekształca dane komunikatu w worek danych.

Jako dane wejściowe dla tego wywołania należy podać:

- Uchwyt połączenia MQI.
- Uchwyt obiektu kolejki, z której ma być odczytany komunikat.
- Deskryptor komunikatu. W strukturze MQMD parametr **Format** musi mieć wartość MQFMT_ADMIN, MQFMT_EVENT, lub MQFMT_PCF.

Uwaga: Jeśli komunikat jest odbierany w ramach jednostki pracy (czyli z opcją MQGMO_SYNCPOINT), a komunikat ma nieobsługiwany format, to można utworzyć kopię zapasową jednostki pracy. Następnie komunikat zostanie przywrócony do kolejki i można go pobrać za pomocą wywołania MQGET zamiast wywołania komendy mqGet. Więcej informacji na temat deskryptora komunikatu można znaleźć w sekcji [Opcje MQGMO-Get-message](#).

- Pobierz opcje komunikatu przy użyciu struktury MQGMO. Więcej informacji na temat struktury MQGMO zawiera sekcja [MQMD-deskryptor komunikatu](#).
- Uchwyt worka, który ma zawierać przekształcony komunikat.

Pełny opis wywołania funkcji mqGetBag znajduje się w sekcji [mqGetBag\(Wtorek mqGet\)](#).

Odpowiedzi PCF w produkcji IBM MQ

W odpowiedzi na każdą komendę serwer komend generuje jeden lub większą liczbę komunikatów odpowiedzi. Komunikat odpowiedzi ma podobny format do komunikatu komendy.

Nagłówek PCF ma taką samą wartość identyfikatora komendy, jak komenda, do której jest ona odpowiedzią (szczegółowe informacje zawiera sekcja [MQCFH-PCF header](#)). Identyfikator komunikatu i identyfikator korelacji są ustawiane zgodnie z opcjami raportu dla żądania.

Jeśli typem nagłówek PCF komunikatu komendy jest MQCFT_COMMAND, generowane są tylko standardowe odpowiedzi. Takie komendy są obsługiwane na wszystkich platformach z wyjątkiem produktu z/OS. Starsze aplikacje nie obsługują PCF w systemie z/OS; IBM MQ Explorer w systemie Windows to jedna taka aplikacja (jednak IBM WebSphere MQ 6.0 lub nowsza IBM MQ Explorer obsługuje PCF w systemie z/OS).

Jeśli typem nagłówek PCF komunikatu komendy jest MQCFT_COMMAND_XR, wówczas generowane są rozszerzone lub standardowe odpowiedzi. Takie komendy są obsługiwane w systemie z/OS i na niektórych innych platformach. Komendy wydane w systemie z/OS generują tylko rozszerzone odpowiedzi. Na innych platformach może zostać wygenerowany dowolny typ odpowiedzi.

Jeśli pojedyncza komenda określa ogólną nazwę obiektu, w komunikacie dla każdego zgodnego obiektu zwracana jest osobna odpowiedź. W przypadku generowania odpowiedzi pojedyncza komenda z nazwą ogólną jest traktowana jako wiele pojedynczych komend (z wyjątkiem pola sterującego MQCFC_LAST lub MQCFC_NOT_LAST). W przeciwnym razie jeden komunikat komendy generuje jeden komunikat odpowiedzi.

Niektóre odpowiedzi PCF mogą zwrócić strukturę nawet wtedy, gdy nie jest ona żądana. Ta struktura jest wyświetlana w definicji odpowiedzi ([Definicje formatów komend programowalnych](#)), jako *zawsze zwracana*. Przyczyna, dla której w przypadku tych odpowiedzi należy podać nazwę obiektów w odpowiedzi, aby zidentyfikować obiekt, którego dotyczą dane.

Deskryptor komunikatu dla odpowiedzi

W komunikacie odpowiedzi znajdują się następujące pola w deskrypcji komunikatu:

MsgType

To pole ma wartość MQMT_REPLY.

MsgId

To pole jest generowane przez menedżer kolejek.

CorrelId

To pole jest generowane zgodnie z opcjami raportu w komunikacie komendy.

Format

To pole to MQFMT_ADMIN.

Kodowanie

Ustaw wartość MQENC_NATIVE.

CodedCharSetId

Ustaw wartość MQCCSI_Q_MGR.

Trwałość

To samo, co w komunikacie komendy.

Priorytet

To samo, co w komunikacie komendy.

Odpowiedź jest generowana z opcją MQPMO_PASS_IDENTITY_CONTEXT.

Odpowiedzi standardowe

Komunikaty komend z typem nagłówka MQCFT_COMMAND, generowane są standardowe odpowiedzi. Takie komendy są obsługiwane na wszystkich platformach z wyjątkiem produktu z/OS.

Istnieją trzy typy standardowej odpowiedzi:

- Odpowiedź OK
- Odpowiedź na błąd
- Odpowiedź na dane

Odpowiedź OK

Ta odpowiedź składa się z komunikatu rozpoczynający się od nagłówka formatu komendy, w polu *CompCode* (MQCC_OK) lub MQCC_WARNING (MQCC_WARNING).

Dla komendy MQCC_OK *Reason* ma wartość MQRC_NONE.

W przypadku wartości MQCC_WARNING *Reason* identyfikuje rodzaj ostrzeżenia. W tym przypadku po nagłówku formatu komendy może nastąpić jedna lub więcej struktur parametrów ostrzegawczych, które są odpowiednie dla tego kodu przyczyny.

W obu przypadkach w przypadku komendy inquire dalsze struktury parametrów mogą być zgodne z opisem w poniższych sekcjach.

Odpowiedź na błąd

Jeśli komenda ma błąd, wysyłany jest jeden lub więcej komunikatów o odpowiedzi na błędy (więcej niż jeden może zostać wysłany nawet dla komendy, która normalnie ma tylko jeden komunikat odpowiedzi). Komunikaty odpowiedzi o błędach mają odpowiednio ustawione MQCFC_LAST lub MQCFC_NOT_LAST.

Każdy taki komunikat rozpoczyna się od nagłówka w formacie odpowiedzi, którego wartość *CompCode* ma wartość MQCC_FAILED i pole *Reason*, które identyfikuje konkretny błąd. W ogólnym przypadku każdy komunikat opisuje inny błąd. Ponadto każdy komunikat ma zero lub jeden (nigdy więcej niż jeden) strukturę parametrów błędu po nagłówku. Ta struktura parametrów, jeśli istnieje, jest strukturą MQCFIN, z polem *Parameter* zawierającym jedną z następujących wartości:

- MQIACF_PARAMETER_ID,

Pole *Value* w strukturze jest identyfikatorem parametru, który wystąpił w błędzie (na przykład MQCA_Q_NAME).

- MQIACF_ERROR_ID

Ta wartość jest używana z wartością *Reason* (w nagłówku formatu komendy) błędu MQRC_UNEXPECTED_ERROR. Pole *Value* w strukturze MQCFIN jest nieoczekiwanym kodem przyczyny odebranych przez serwer komend.

- MQIACF_SELECTOR

Ta wartość występuje wtedy, gdy struktura listy (MQCFIL) wysłana z komendą zawiera zduplikowany selektor lub jeden, który jest niepoprawny. Pole *Reason* w nagłówku formatu komendy identyfikuje

błąd, a pole *Value* w strukturze MQCFIN jest wartością parametru w strukturze MQCFIL komendy, która była błędna.

- MQIACF_ERROR_OFFSET

Ta wartość występuje wtedy, gdy w komendzie Ping Channel wystąpił błąd porównywania danych. Pole *Value* w strukturze jest przesuniętym błędem porównania kanału Ping.

- MQIA_CODED_CHAR_SET_ID

Ta wartość występuje wtedy, gdy identyfikator kodowanego zestawu znaków w deskrytorze komunikatu przychodzącego komunikatu komendy PCF nie jest zgodny z identyfikatorem docelowego menedżera kolejek. Pole *Value* w strukturze jest kodowanym identyfikatorem zestawu znaków menedżera kolejek.

Ostatni (lub tylko) komunikat odpowiedzi o błędzie jest odpowiedzią podsumowową z polem *CompCode* o wartości MQCC_FAILED i polem *Reason* o nazwie MQRCCF_COMMAND_FAILED. Ten komunikat nie ma struktury parametru po nagłówku.

Odpowiedź na dane

Ta odpowiedź składa się z odpowiedzi OK (zgodnie z wcześniejszym opisem) do komendy inquire. Po kliknięciu przycisku OK zostaną wykonane dodatkowe struktury zawierające żądane dane zgodnie z opisem w sekcji Definicje formatów komend programowalnych.

Aplikacje nie mogą zależeć od tych dodatkowych struktur parametrów, które są zwracane w dowolnej kolejności.

Rozszerzone odpowiedzi

Komendy wydawane w systemie z/OS generują rozszerzone odpowiedzi.

Istnieją trzy typy odpowiedzi rozszerzonej:

- Odpowiedź komunikatu z typem MQCFT_XR_MSG
- Odpowiedź elementu, o typie MQCFT_XR_ITEM
- Odpowiedź podsumowania, o typie MQCFT_XR_SUMMARY

Każda komenda może generować jeden lub więcej zestawów odpowiedzi. Każdy zestaw odpowiedzi składa się z jednego lub większej liczby komunikatów, ponumerowanych sekwencyjnie od 1 w polu *MsgSeqNumber* nagłówka PCF. Pole *Control* ostatniej odpowiedzi (lub tylko) w każdym zestawie ma wartość MQCFC_LAST. Dla wszystkich innych odpowiedzi w zestawie ta wartość to MQCFC_NOT_LAST.

Każda odpowiedź może zawierać jedną lub większą liczbę opcjonalnych struktur MQCFBS, w których pole *Parameter* jest ustawione na wartość MQBACF_RESPONSE_SET, a wartość jest identyfikatorem zestawu odpowiedzi. Identyfikatory są unikalne i identyfikują zestaw odpowiedzi, które zawierają odpowiedź. Dla każdego zestawu odpowiedzi znajduje się struktura MQCFBS, która go identyfikuje.

Rozszerzone odpowiedzi mają co najmniej dwie struktury parametrów:

- Struktura MQCFBS z polem *Parameter* ustawionym na wartość MQBACF_RESPONSE_ID. Wartość w tym polu jest identyfikatorem zestawu odpowiedzi, do którego należy odpowiedź. Identyfikator w pierwszym zestawie jest dowolny. W kolejnych zestawach identyfikator jest wcześniej powiadamiany w strukturze MQBACF_RESPONSE_SET.
- Struktura MQCFST z polem *Parameter* ustawionym na wartość MQCACF_RESPONSE_Q_MGR_NAME (wartość to nazwa menedżera kolejek, z którego pochodzi zestaw odpowiedzi).

Wiele odpowiedzi ma dodatkowe struktury parametrów, a struktury te są opisane w poniższych sekcjach.

Nie można z wyprzedzeniem określić, ile odpowiedzi ma miejsce w zestawie innym niż przez uzyskanie odpowiedzi do czasu znalezienia jednego z nich za pomocą komendy MQCFC_LAST. Nie można wcześniej określić liczby zestawów odpowiedzi, które mogą zawierać struktury MQBACF_RESPONSE_SET, aby wskazać, że generowane są dodatkowe zestawy.

Rozszerzone odpowiedzi na komendy Inquire

Komendy inquire zwykle generują odpowiedź elementu (typ MQCFT_XR_ITEM) dla każdego znalezionej elementu, który jest zgodny z podanymi kryteriami wyszukiwania. Odpowiedź na element zawiera pole *CompCode* w nagłówku z wartością MQCC_OK oraz pole *Reason* z wartością MQRC_NONE. Zawiera także inne struktury parametrów opisujące element i jego żądane atrybuty, zgodnie z opisem w sekcji [Definicje formatów komend programowalnych](#).

Jeśli element jest błędny, pole *CompCode* w nagłówku ma wartość MQCC_FAILED, a pole *Reason* identyfikuje konkretny błąd. Do identyfikowania elementu dołączane są dodatkowe struktury parametrów.

Niektóre komendy Inquire mogą zwracać odpowiedzi na komunikaty ogólne (nie specyficzne dla nazwy), a także odpowiedzi na pytania. Te odpowiedzi mają charakter informacyjny lub błąd, odpowiedzi typu MQCFT_XR_MSG.

Jeśli wykonanie komendy Inquire powiedzie się, może to być opcjonalnie odpowiedź podsumowująca (typ MQCFT_XR_SUMMARY) z wartością *CompCode* MQCC_OK i wartością pola *Reason* MQRC_NONE.

Jeśli wykonanie komendy Inquire nie powiedzie się, mogą zostać zwrócone odpowiedzi pozycji, a opcjonalnie może być odpowiedź podsumowująca (typ MQCFT_XR_SUMMARY) z wartością *CompCode* MQCC_FAILED i wartością pola *Reason* MQRCCF_COMMAND_FAILED.

Rozszerzone odpowiedzi na komendy inne niż Inquire

Pomyślne komendy generują odpowiedzi komunikatów, w których pole *CompCode* w nagłówku ma wartość MQCC_OK, a pole *Reason* ma wartość MQRC_NONE. Zawsze istnieje co najmniej jeden komunikat. Może to być komunikat informacyjny (MQCFT_XR_MSG) lub podsumowanie (MQCFT_XR_SUMMARY). Opcjonalnie mogą być wyświetlane dodatkowe komunikaty informacyjne (typ MQCFT_XR_MSG). Każdy komunikat informacyjny może zawierać pewną liczbę dodatkowych struktur parametrów z informacjami o komendzie; należy zapoznać się z opisami poszczególnych komend dla struktur, które mogą wystąpić.

Komendy, które nie powiodły się, generują odpowiedzi na komunikaty o błędach (typ MQCFT_XR_MSG), w których pole *CompCode* w nagłówku ma wartość MQCC_FAILED, a pole *Reason* identyfikuje konkretny błąd. Każdy komunikat może zawierać pewną liczbę dodatkowych struktur parametrów z informacjami o błędzie: patrz poszczególne opisy błędów dla struktur, które mogą wystąpić. Mogą zostać wygenerowane komunikaty informacyjne. Możliwe jest, opcjonalnie, podsumowanie odpowiedzi (MQCFT_XR_SUMMARY) z wartością *CompCode* MQCC_FAILED i wartością pola *Reason* MQRCCF_COMMAND_FAILED.

Rozszerzone odpowiedzi na komendy za pomocą funkcji CommandScope

Jeśli komenda używa parametru **CommandScope** lub powoduje wygenerowanie komendy przy użyciu parametru **CommandScope**, to istnieje początkowy zestaw odpowiedzi z menedżera kolejek, w którym została odebrana komenda. Następnie dla każdego menedżera kolejek, do którego kierowane jest polecenie, generowany jest osobny zestaw lub zestawy odpowiedzi (tak jak w przypadku wydawania wielu pojedynczych komend). Na koniec istnieje zestaw odpowiedzi z odbierającego menedżera kolejek, który zawiera ogólną odpowiedź podsumowującą (typ MQCFT_XR_SUMMARY). Struktura parametru MQCACF_RESPONSE_Q_MGR_NAME identyfikuje menedżera kolejek, który generuje każdy zestaw.


Początkowy zestaw odpowiedzi ma następujące dodatkowe struktury parametrów:

- MQIACF_COMMAND_INFO (MQCFIN). Możliwe wartości w tej strukturze to MQCMDI_CMDSCOPE_ACCEPTED lub MQCMDI_CMDSCOPE_GENERATED.
- MQIACF_CMDSCOPE_Q_MGR_COUNT (MQCFIN). Ta struktura wskazuje liczbę menedżerów kolejek, do których wysyłana jest komenda.

Sprawdzanie uprawnień dla komend PCF w produkcie IBM MQ

Po przetworzeniu komendy PCF *UserIdentifier* z deskryptora komunikatu w komunikacie komendy jest używany dla wymaganych sprawdzeń uprawnień do obiektu w produkcie IBM MQ. Sprawdzanie uprawnień jest implementowane w inny sposób na każdej platformie zgodnie z opisem w tym temacie.

Sprawdzenia są wykonywane w systemie, w którym komenda jest przetwarzana; dlatego ten identyfikator użytkownika musi istnieć w systemie docelowym i musi mieć wymagane uprawnienia do przetwarzania komendy. Jeśli komunikat pochodzi z systemu zdalnego, jednym ze sposobów na osiągnięcie identyfikatora istniejącego w systemie docelowym jest posiadanie zgodnego identyfikatora użytkownika zarówno w systemie lokalnym, jak i zdalnym.

Uwaga:  Informacje na temat sprawdzania uprawnień w systemie z/OS znajdują się w temacie [Czynność 1: Identyfikowanie parametrów systemu z/OS](#).

IBM MQ for IBM i



Aby możliwe było przetworzenie dowolnej komendy PCF, użytkownik musi mieć uprawnienie *dsp* dla obiektu IBM MQ w systemie docelowym.

Dodatkowo, dla niektórych komend PCF wykonywane są sprawdzenia uprawnień do obiektu IBM MQ, jak to pokazano na [Tabela 1 na stronie 31](#).

W większości przypadków te kontrole są takie same, jak kontrole przeprowadzane przez równoważną komendę CL systemu IBM MQ, która została wydana w systemie lokalnym. Więcej informacji na temat odwzorowania uprawnień IBM MQ uprawnień do systemu IBM i oraz wymagań dotyczących uprawnień dla komend CL produktu IBM MQ zawiera sekcja [Konfigurowanie zabezpieczeń w systemie IBM i](#). Szczegółowe informacje dotyczące zabezpieczeń procedur zewnętrznych są podane w dokumentacji [Zabezpieczenia na poziomie łącza przy użyciu wyjścia zabezpieczeń](#).

Aby przetworzyć dowolne z następujących komend ID użytkownika musi należeć do profilu grupowego QMQMADM:

- Kanał ping
- Zmień kanał
- Kopiuj kanał
- Utwórz kanał
- Usuń kanał
- Resetowanie kanału
- Rozstrzygnięcie kanału
- Uruchom kanał
- Zamknij kanał
- Uruchom inicjator kanału
- Uruchom program nastuchujący kanału

IBM MQ for UNIX, Linux, and Windows



Aby możliwe było przetworzenie dowolnej komendy PCF, identyfikator użytkownika musi mieć uprawnienie *dsp* dla obiektu menedżera kolejek w systemie docelowym. Dodatkowo, dla niektórych komend PCF wykonywane są sprawdzenia uprawnień do obiektu IBM MQ, jak to pokazano na [Tabela 1 na stronie 31](#).

Aby przetworzyć jedną z następujących komend, identyfikator użytkownika musi należeć do grupy *mqm*.

Uwaga: W przypadku serwera Windows **tylko** identyfikator użytkownika może należeć do grupy *Administratorzy* lub grupy *mqm*.

- Zmień kanał
- Kopiuj kanał

- Utwórz kanał
- Usuń kanał
- Kanał ping
- Resetowanie kanału
- Uruchom kanał
- Zamknij kanał
- Uruchom inicjator kanału
- Uruchom program nasłuchujący kanału
- Rozstrzygnięcie kanału
- Resetowanie klastra
- Odśwież klaster
- Menedżer kolejki - SUSPEND
- Wznów menedżer kolejek

Uprawnienia do obiektów produktu IBM MQ dla wielu platform



Tabela 1. Uprawnienia do obiektu

Komenda	Uprawnienie do obiektu IBM MQ	Uprawnienie klasy (dla typu obiektu)
Zmień informacje uwierzytelniające	dsp i chg	nie dotyczy
Zmień kanał	dsp i chg	nie dotyczy
Zmiana obiektu nasłuchiwanie kanału	dsp i chg	nie dotyczy
Zmień kanał połączenia klienta	dsp i chg	nie dotyczy
Zmień listę nazw	dsp i chg	nie dotyczy
Zmień proces	dsp i chg	nie dotyczy
Zmiana kolejki	dsp i chg	nie dotyczy
Zmiana menedżera kolejek	chg <i>patrz uwaga 3 i uwaga 5</i>	nie dotyczy
Zmień usługę	dsp i chg	nie dotyczy
Wyczyść kolejkę	clr	nie dotyczy
Kopiowanie informacji uwierzytelniających	dsp	crt
Kopiowanie informacji uwierzytelniających (zastępowanie) <i>patrz uwaga 1</i>	<i>od: dsp do: chg</i>	crt
Kopiuj kanał	dsp	crt
Kopiuj kanał (zastępowanie) <i>patrz uwaga 1</i>	<i>od: dsp do: chg</i>	crt
Kopiuj program nasłuchujący kanału	dsp	crt

<i>Tabela 1. Uprawnienia do obiektu (kontynuacja)</i>		
Komenda	Upewnienie do obiektu IBM MQ	Upewnienie klasy (dla typu obiektu)
Kopiowanie procesu nastuchiwania kanału (zastępowanie) <i>patrz uwaga 1</i>	<i>od: dsp do: chg</i>	crt
Kopiuj kanał połączenia klienta	dsp	crt
Kopiuj kanał połączenia klienta (zastępowanie) <i>patrz uwaga 1</i>	<i>od: dsp do: chg</i>	crt
Kopiuj listę nazw	dsp	crt
Kopiowanie listy nazw (Replace) <i>patrz Uwaga 1</i>	<i>from: dsp to: dsp i chg</i>	crt
Kopiuj proces	dsp	crt
Kopiowanie procesu (zastępowanie) <i>patrz uwaga 1</i>	<i>od: dsp do: chg</i>	crt
Kopiuj kolejkę	dsp	crt
Kopiowanie kolejki (zastępowanie) <i>patrz uwaga 1</i>	<i>from: dsp to: dsp i chg</i>	crt
Tworzenie informacji uwierzytelniających	<i>(systemowe domyślne informacje o uwierzytelnianiu) dsp</i>	crt
Tworzenie informacji uwierzytelniających (zastępowanie) <i>patrz uwaga 1</i>	<i>(domyślne informacje o uwierzytelnianiu systemu) dsp to: chg</i>	crt
Utwórz kanał	<i>(domyślny kanał systemowy) dsp</i>	crt
Tworzenie kanału (zastępowanie) <i>patrz uwaga 1</i>	<i>(domyślny kanał systemowy) dsp to: chg</i>	crt
Tworzenie nastuchiwania kanału	<i>(domyślny program nastuchujący systemu) dsp</i>	crt
Tworzenie nastuchiwania kanału (zastępowanie) <i>patrz uwaga 1</i>	<i>(domyślny program nastuchujący systemu) dsp to: chg</i>	crt
Utwórz kanał połączenia klienta	<i>(domyślny kanał systemowy) dsp</i>	crt
Utwórz kanał połączenia klienta (zastępowanie) <i>patrz uwaga 1</i>	<i>(domyślny kanał systemowy) dsp to: chg</i>	crt
Utwórz listę nazw	<i>(systemowa domyślna lista nazw) dsp</i>	crt
Tworzenie listy nazw (Replace) <i>patrz Uwaga 1</i>	<i>(systemowa domyślna lista nazw) dsp to: dsp i chg</i>	crt
Utwórz proces	<i>(domyślny proces systemowy) dsp</i>	crt
Utwórz proces (zastępowanie) <i>patrz uwaga 1</i>	<i>(domyślny proces systemowy) dsp to: chg</i>	crt
Tworzenie kolejki	<i>(systemowa kolejka domyślna) dsp</i>	crt
Tworzenie kolejki (zastępowanie) <i>patrz uwaga 1</i>	<i>(systemowa kolejka domyślna) dsp to: dsp i chg</i>	crt

<i>Tabela 1. Uprawnienia do obiektu (kontynuacja)</i>		
Komenda	Uprawnienie do obiektu IBM MQ	Uprawnienie klasy (dla typu obiektu)
Utwórz usługę	<i>(systemowa kolejka domyślna)</i> dsp	crt
Utwórz usługę (zastępowanie) <i>patrz uwaga 1</i>	<i>(systemowa kolejka domyślna)</i> dsp to: chg	crt
Usuń informacje uwierzytelniające	dsp i dlt	nie dotyczy
Usuń rekord uprawnień	<i>(obiekt menedżera kolejek)</i> chg <i>patrz Uwaga 4</i>	<i>patrz Uwaga 4</i>
Usuń kanał	dsp i dlt	nie dotyczy
Usuń program nasłuchujący kanału	dsp i dlt	nie dotyczy
Usuń kanał połączenia klienta	dsp i dlt	nie dotyczy
Usuń listę nazw	dsp i dlt	nie dotyczy
Usuń proces	dsp i dlt	nie dotyczy
Usuń kolejkę	dsp i dlt	nie dotyczy
Usuwanie usługi	dsp i dlt	nie dotyczy
Sprawdzanie informacji uwierzytelniających	dsp	nie dotyczy
Sprawdź rekordy uprawnień	<i>patrz Uwaga 4</i>	<i>patrz Uwaga 4</i>
Sprawdź kanał	dsp	nie dotyczy
Sprawdź program nasłuchujący kanału	dsp	nie dotyczy
Sprawdzanie statusu kanału (dla ChannelType MQCHT_CLSSDR)	inq	nie dotyczy
Sprawdź kanał połączenia klienta	dsp	nie dotyczy
Sprawdź listę nazw	dsp	nie dotyczy
Sprawdź proces	dsp	nie dotyczy
Sprawdź kolejkę	dsp	nie dotyczy
Zapytaj menedżera kolejek	<i>patrz uwaga 3</i>	nie dotyczy
Sprawdź status kolejki	dsp	nie dotyczy
Sprawdź usługę	dsp	nie dotyczy
Kanał ping	ctrl	nie dotyczy
Menedżer kolejek ping	<i>patrz uwaga 3</i>	nie dotyczy
Odśwież menedżera kolejek	(obiekt menedżera kolejek) chg	nie dotyczy
Odśwież zabezpieczenia (dla SecurityType MQSECTYPE_SSL)	(obiekt menedżera kolejek) chg	nie dotyczy
Resetowanie kanału	ctrlx	nie dotyczy

Tabela 1. Uprawnienia do obiektu (kontynuacja)		
Komenda	Uprawnienie do obiektu IBM MQ	Uprawnienie klasy (dla typu obiektu)
Resetowanie menedżera kolejek	(obiekt menedżera kolejek) chg	nie dotyczy
Resetuj statystyki kolejki	dsp i chg	nie dotyczy
Rozstrzygnięcie kanału	ctrlx	nie dotyczy
Ustaw rekord uprawnień	(obiekt menedżera kolejek) chg patrz Uwaga 4	patrz Uwaga 4
Uruchom kanał	ctrl	nie dotyczy
Zamknij kanał	ctrl	nie dotyczy
Zatrzymaj połączenie	(obiekt menedżera kolejek) chg	nie dotyczy
Uruchom proces nasłuchujący	ctrl	nie dotyczy
Zatrzymaj proces nasłuchujący	ctrl	nie dotyczy
Uruchom usługę	ctrl	nie dotyczy
Zatrzymaj usługę	ctrl	nie dotyczy
Esc	patrz Uwaga 2	patrz Uwaga 2

Uwagi:

1. Ta komenda ma zastosowanie, jeśli obiekt, który ma zostać zastąpiony, istnieje, w przeciwnym razie kontrola uprawnień jest taka, jak dla operacji Create (tworzenie) lub Copy bez zastępowania (Replace).
2. Wymagane uprawnienia są określone przez komendę MQSC zdefiniowaną przez tekst zmiany znaczenia, która jest równoważna jednej z poprzednich komend.
3. Aby możliwe było przetworzenie dowolnej komendy PCF, identyfikator użytkownika musi mieć uprawnienie dsp dla obiektu menedżera kolejek w systemie docelowym.
4. Ta komenda PCF jest autoryzowana, jeśli serwer komend nie został uruchomiony z parametrem -a. Domyślnie serwer komend jest uruchamiany, gdy menedżer kolejek jest uruchomiony i bez parametru -a. Więcej informacji na ten temat zawiera sekcja [Skorowidz formatów komend programowalnych](#).
5. Nadawanie ID użytkownika chg uprawnień dla menedżera kolejek daje możliwość ustawiania rekordów uprawnień dla wszystkich grup i użytkowników. Nie należy przyznawać tego uprawnienia zwykłym użytkownikom lub aplikacjom.

Produkt IBM MQ udostępnia również niektóre punkty wyjścia zabezpieczeń kanału, dzięki czemu można będzie dostarczać własne programy obsługi wyjścia dla sprawdzania zabezpieczeń. Więcej informacji na ten temat zawiera sekcja [Wyświetlanie kanału](#).

Korzystanie z interfejsu MQAI w celu uproszczenia korzystania z systemów PCF

Interfejs administracyjny produktu IBM MQ (MQAI) jest interfejsem programistycznym dla produktu IBM MQ dostępnym w systemie AIX HP-UX IBM i Linux Solaris i Windows. Wykonuje zadania administracyjne w menedżerze kolejek produktu IBM MQ przy użyciu worków danych w celu obsługi właściwości (lub parametrów) obiektów w sposób łatwiejszy niż przy użyciu formatów komend programowalnych (Programmable Command Formats-PCFs).

Interfejs MQAI wykonuje zadania administracyjne w menedżerze kolejek przy użyciu *worków danych*. Worki danych pozwalają na obsługę właściwości (lub parametrów) obiektów w sposób łatwiejszy niż przy użyciu PCFs.

Zalety korzystania z interfejsu MQAI są następujące:

Uproszczenie korzystania z komunikatów PCF

Interfejs MQAI jest łatwiejszym sposobem administrowania produktem IBM MQ. Jeśli używana jest usługa MQAI, nie ma potrzeby zapisywania własnych komunikatów PCF. Pozwala to uniknąć problemów związanych ze złożonymi strukturami danych.

Aby przekazać parametry w programach napisanych za pomocą wywołań MQI, komunikat PCF musi zawierać komendę oraz szczegółowe informacje na temat danych łańcucha lub liczb całkowitych. Aby utworzyć tę konfigurację ręcznie, należy dodać kilka instrukcji w programie dla każdej struktury i przydzielić miejsce w pamięci. To zadanie może być długie i pracochłonne.

Programy napisane przy użyciu parametru MQAI przekazują parametry do odpowiedniego worka danych, a dla każdej struktury potrzebne jest tylko jedno oświadczenie. Użycie toreb danych MQAI usuwa potrzebę obsługi tablic i przydzielania pamięci masowej, a także zapewnia pewien stopień izolacji od szczegółów PCF.

Łatwiejsza obsługa warunków błędu

Trudno jest uzyskać kody powrotu z komend PCF. Interfejs MQAI ułatwia obsługę warunków błędów przez program.

Wymiana danych między aplikacjami

Dane aplikacji są wysyłane w formacie PCF, a następnie spakowane i rozpakowane przez interfejs MQAI. Jeśli dane komunikatu składają się z liczb całkowitych i łańcuchów znaków, można użyć interfejsu MQAI, aby skorzystać z wbudowanej konwersji danych produktu IBM MQ na potrzeby danych PCF. Pozwala to uniknąć konieczności pisania wyjść konwersji danych.

Po utworzeniu i wypełnieniu worka danych można wysłać komunikat komendy administracyjnej do serwera komend menedżera kolejek przy użyciu wywołania `mqExecute`. To wywołanie oczekuje na komunikaty odpowiedzi. Wywołanie metody `mqExecute` obsługuje wymianę z serwerem komend i zwraca odpowiedzi w *worku odpowiedzi*.

Przykłady użycia interfejsu MQAI

Przedstawiona lista zawiera przykładowe programy, które demonstrują użycie interfejsu MQAI. Przykłady wykonują następujące zadania:

1. Utwórz kolejkę lokalną. [“Przykładowy program C do tworzenia kolejki lokalnej \(amqsaicq.c\)” na stronie 42](#)
2. Wyświetlaj zdarzenia na ekranie za pomocą prostego monitora zdarzeń. [“Przykładowy program C do wyświetlania zdarzeń przy użyciu monitora zdarzeń \(amqsaiem.c\)” na stronie 45](#)
3. Służ do drukowania listy wszystkich kolejek lokalnych i ich bieżących głębokości. [“Przykładowy program C do uzyskiwania informacji o kolejkach i drukowaniu informacji \(amqsailq.c\)” na stronie 58](#)
4. Wydrukuj listę wszystkich kanałów i ich typów. [“Przykładowy program w języku C dla zapytań o obiekty kanału \(amqsaicl.c\)” na stronie 52](#)

Budowanie aplikacji MQAI

Aby zbudować aplikację przy użyciu interfejsu MQAI, należy połączyć się z tymi samymi bibliotekami, co w przypadku produktu IBM MQ. Informacje na temat budowania aplikacji produktu IBM MQ można znaleźć w sekcji [Budowanie aplikacji proceduralnej](#).

Porady i wskazówki dotyczące konfigurowania produktu IBM MQ przy użyciu interfejsu MQAI

Interfejs MQAI używa komunikatów PCF do wysyłania komend administracyjnych do serwera komend, a nie do bezpośredniego zajmowania się tym serwerem komend. Wskazówki dotyczące konfigurowania produktu IBM MQ przy użyciu interfejsu MQAI można znaleźć w sekcji [“Wskazówki dotyczące korzystania z interfejsu MQAI do konfigurowania produktu IBM MQ” na stronie 36](#).

Informacje pokrewne

[Skorowidz interfejsu administracyjnego produktu IBM MQ](#)

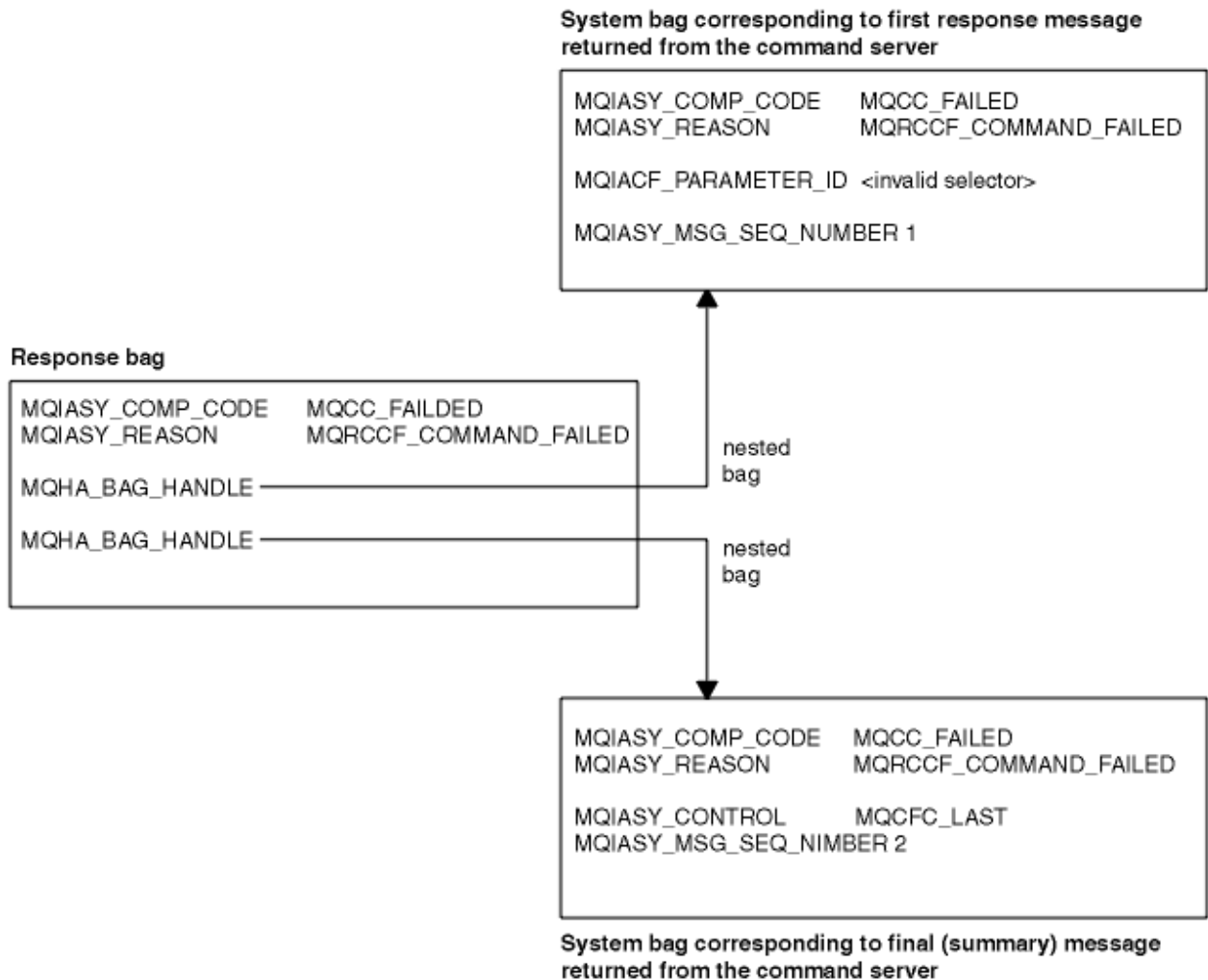
Wskazówki dotyczące korzystania z interfejsu MQAI do konfigurowania produktu IBM MQ

Interfejs administracyjny produktu IBM MQ (MQAI) używa komunikatów PCF do wysyłania komend administracyjnych do serwera komend, a nie do bezpośredniego zajmowania się tym serwerem komend. Poniżej przedstawiono kilka wskazówek dotyczących konfigurowania produktu IBM MQ przy użyciu interfejsu MQAI.

- Łańcuchy znaków w IBM MQ są puste dopełnione do stałej długości. Za pomocą języka C łańcuchy zakończone znakiem o kodzie zero mogą być zwykle dostarczane jako parametry wejściowe do interfejsów programistycznych produktu IBM MQ .
- Aby wyczyścić wartość atrybutu łańcuchowego, należy ustawić go jako pojedynczy pusty łańcuch, a nie pusty łańcuch.
- Należy rozważyć z wyprzedzeniem atrybuty, które mają zostać zmienione, a także zapytania dotyczące tylko tych atrybutów.
- Niektóre atrybuty nie mogą być zmieniane, na przykład nazwa kolejki lub typ kanału. Upewnij się, że podjęto próbę zmiany tylko tych atrybutów, które mogą być modyfikowane. Zapoznaj się z listą wymaganych i opcjonalnych parametrów dla konkretnego obiektu zmiany PCF. Więcej informacji zawiera sekcja [Definicje formatów komend programowalnych](#).
- Jeśli wywołanie MQAI nie powiedzie się, w worku odpowiedzi zostaną zwrócone szczegółowe informacje o niepowodzeniu. Dalsze szczegóły można następnie znaleźć w zagnieżdżonej torbie, do której można uzyskać dostęp za pomocą selektora MQHA_BAG_HANDLE. Na przykład, jeśli wywołanie mqExecute nie powiedzie się z kodem przyczyny MQRCCF_COMMAND_FAILED, informacje te są zwracane w worku odpowiedzi. Możliwe przyczyny tego kodu przyczyny: określony selektor nie jest poprawny dla typu komunikatu komendy, a ten szczegół informacji znajduje się w zagnieżdżonej torbie, do której można uzyskać dostęp za pomocą uchwytu torby.

Więcej informacji na temat komendy MQExecute można znaleźć w sekcji [“Wysyłanie komend administracyjnych do serwera komend qm przy użyciu wywołania mqExecute” na stronie 71](#) .

Poniższy diagram przedstawia ten scenariusz:



Zaawansowane tematy MQAI

Informacje na temat indeksowania, konwersji danych i używania deskryptora komunikatu

- Indeksowanie

Indeksy są używane przy zastępowaniu lub usuwaniu istniejących elementów danych z torby w celu zachowania kolejności wstawiania. Szczegółowe informacje na temat indeksowania można znaleźć w podręczniku [“Indeksowanie w interfejsie MQAI”](#) na stronie 37.

- Konwersja danych

Łańcuchy zawarte w torbie danych MQAI mogą znajdować się w różnych kodowanych zestawach znaków i można je przekształcić za pomocą wywołania `mqSetInteger`. Szczegółowe informacje na temat konwersji danych można znaleźć w podręczniku [“Przetwarzanie konwersji danych w interfejsie MQAI”](#) na stronie 38.

- Użycie deskryptora komunikatu

Interfejs MQAI generuje deskryptor komunikatu, który jest ustawiany na wartość początkową podczas tworzenia worka danych. Szczegółowe informacje na temat używania deskryptora komunikatu można znaleźć w sekcji [“Użycie deskryptora komunikatu w interfejsie MQAI”](#) na stronie 40.

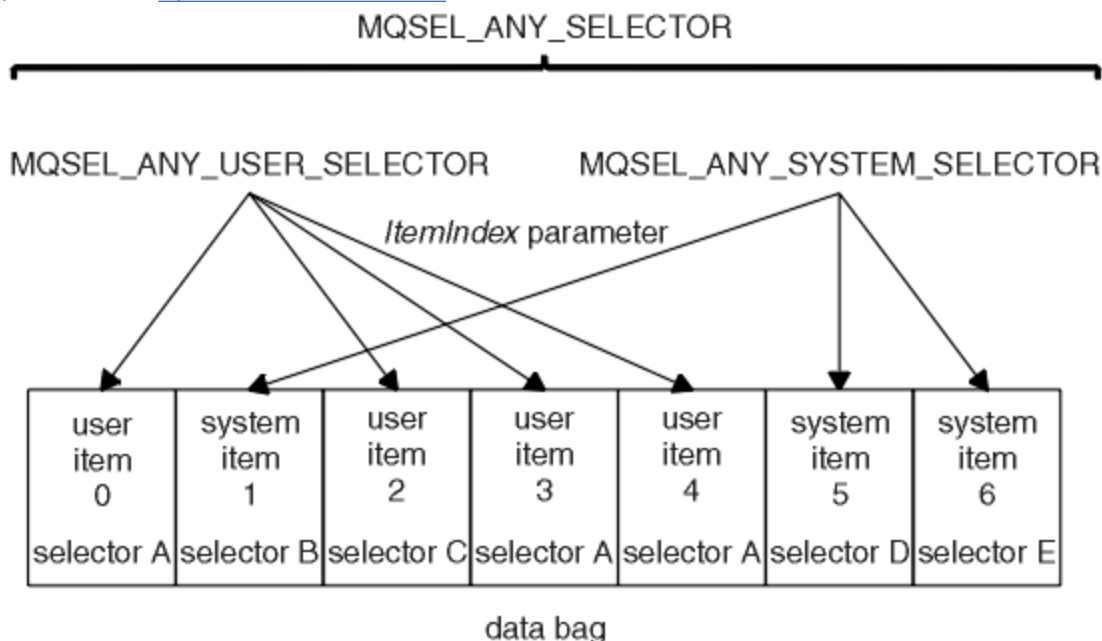
Indeksowanie w interfejsie MQAI

Indeksy są używane przy zastępowaniu lub usuwaniu istniejących elementów danych z torby. Istnieją trzy typy indeksowania, dzięki którym elementy danych mogą być łatwo pobrane.

Każdy selektor i wartość w obrębie elementu danych w torbie mają trzy powiązane numery indeksów:

- Indeks względem innych elementów, które mają ten sam selektor.
- Indeks w odniesieniu do kategorii selektora (użytkownika lub systemu), do którego należy element.
- Indeks względem wszystkich elementów danych w torbie (użytkownik i system).

Umożliwia to indeksowanie przez selektory użytkownika, selektory systemowe lub oba te elementy, jak to pokazano na [Rysunek 4 na stronie 38](#).



Rysunek 4. Indeksowanie

Na rysunku [Rysunek 4 na stronie 38](#) element użytkownika 3 (selektor A) może być przywołany przez następujące pary indeksów:

Selector	ItemIndex
selektor A	1
MQSEL_ANY_USER_SELECTOR	2
MQSEL_ANY_SELECTOR	3

Indeks jest oparty na zerach, podobnie jak w przypadku tablicy w języku C. Jeśli występują wystąpienia 'n' wystąpień, indeks waha się od zera do wartości 'n-1', bez przerw.

Indeksy są używane przy zastępowaniu lub usuwaniu istniejących elementów danych z torby.

W przypadku użycia w ten sposób kolejność wstawiania jest zachowywana, ale może mieć to wpływ na indeksy innych elementów danych. Przykłady tego można znaleźć w sekcji [Zmiana informacji w torbie i Usuwanie elementów danych](#).

Trzy typy indeksowania umożliwiają łatwe pobieranie elementów danych. Na przykład, jeśli w worku znajdują się trzy instancje konkretnego selektora, wywołanie `mqCountElementów` może zliczyć liczbę instancji tego selektora, a wywołania `mqInquire*` mogą określać zarówno selektor, jak i indeks, aby uzyskać dostęp tylko do tych wartości. Jest to przydatne w przypadku atrybutów, które mogą mieć listę wartości, takich jak niektóre z wyjść w kanałach.

Przetwarzanie konwersji danych w interfejsie MQAI

Łańcuchy zawarte w torbie danych MQAI mogą znajdować się w różnych kodowanych zestawach znaków. Łańcuchy te można przekształcić za pomocą wywołania `mqSetInteger`.

Podobnie jak w przypadku komunikatów PCF, łańcuchy zawarte w worku danych MQAI mogą znajdować się w różnych kodowanych zestawach znaków. Zwykle wszystkie łańcuchy w komunikacie PCF znajdują się w tym samym kodowanym zestawie znaków; jest to ten sam zestaw, co menedżer kolejek.

Każdy element łańcucha w worku danych zawiera dwie wartości: sam łańcuch i identyfikator CCSID. Łańcuch dodawany do tego pliku jest uzyskiwany z parametru **Buffer** w łańcuchu mqAddString lub mqSetString. Identyfikator CCSID jest uzyskiwany z elementu systemu zawierającego selektor tabeli MQIASY_CODED_CHAR_SET_ID. Jest on znany jako *identyfikator CCSID torby* i może zostać zmieniony przy użyciu wywołania mqSetInteger.

Podczas uzyskiwania informacji o wartości łańcucha zawartego w worku danych, identyfikator CCSID jest parametrem wyjściowym z wywołania.

Tabela 2 na stronie 39 przedstawia reguły stosowane podczas przekształcania worków danych w komunikaty i odwrotnie:

<i>Tabela 2. Przetwarzanie CCSID</i>			
Wywołanie MQAI	CCSID	Dane wejściowe do wywołania	Dane wyjściowe do wywołania
mqBagToBuffer	Identyfikator CCSID torby (<u>1</u>)	Ignorowany	Niezmienione
mqBagToBuffer	Identyfikatory CCSID łańcuchów w torbie	Wykorzystane	Niezmienione
mqBagToBuffer	Identyfikatory CCSID łańcuchów w buforze	Nie dotyczy	Skopiowane z identyfikatorów CCSID łańcuchów w torbie
mqBufferToBag	Identyfikator CCSID torby (<u>1</u>)	Ignorowany	Niezmienione
mqBufferToBag	Identyfikatory CCSID łańcuchów w buforze	Wykorzystane	Niezmienione
mqBufferToBag	Identyfikatory CCSID łańcuchów w torbie	Nie dotyczy	Skopiowano z łańcuchowych identyfikatorów CCSID w buforze
TorbamqPut	CCSID MQMD	Wykorzystane	Niezmienione (<u>2</u>)
TorbamqPut	Identyfikator CCSID torby (<u>1</u>)	Ignorowany	Niezmienione
TorbamqPut	Identyfikatory CCSID łańcuchów w torbie	Wykorzystane	Niezmienione
TorbamqPut	Identyfikatory CCSID łańcuchów w wystanym komunikacie	Nie dotyczy	Skopiowane z identyfikatorów CCSID łańcuchów w torbie
TorbamqGet	CCSID MQMD	Służy do przetwarzania danych komunikatu.	Ustawiono identyfikator CCSID zwracanych danych (<u>3</u>)
TorbamqGet	Identyfikator CCSID torby (<u>1</u>)	Ignorowany	Niezmienione
TorbamqGet	Identyfikatory CCSID łańcuchów w komunikacie	Wykorzystane	Niezmienione

<i>Tabela 2. Przetwarzanie CCSID (kontynuacja)</i>			
Wywołanie MQAI	CCSID	Dane wejściowe do wywołania	Dane wyjściowe do wywołania
TorbamqGet	Identyfikatory CCSID łańcuchów w torbie	Nie dotyczy	Skopiowano z identyfikatorów CCSID łańcucha w komunikacie
mqExecute	Identyfikator CCSID torby żądanie-worek	Używane w przypadku żądania MQMD komunikatu żądania (4)	Niezmienione
mqExecute	Identyfikator CCSID rekordu odpowiedzi	Używane do konwersji danych komunikatu odpowiedzi (4)	Ustawiono identyfikator CCSID zwracanych danych (3)
mqExecute	Identyfikatory CCSID łańcuchów w rekordzie żądania	Używane dla komunikatu żądania	Niezmienione
mqExecute	Identyfikatory CCSID łańcuchów w rekordzie odpowiedzi	Nie dotyczy	Skopiowano z identyfikatorów CCSID łańcucha w komunikacie odpowiedzi

Uwagi:

1. Identyfikator CCSID torby jest elementem systemowym z selektorem MQIASY_CODED_CHAR_SET_ID.
2. Wartość MQCCSI_Q_MGR jest zmieniana na rzeczywisty identyfikator CCSID menedżera kolejek.
3. Jeśli zażądano konwersji danych, identyfikator CCSID zwracanych danych jest taki sam jak wartość wyjściowa. Jeśli konwersja danych nie jest żądana, zwracany identyfikator CCSID danych jest taki sam jak wartość komunikatu. Należy pamiętać, że jeśli żądana jest konwersja danych, nie jest zwracany żaden komunikat.
4. Jeśli identyfikator CCSID ma wartość MQCCSI_DEFAULT, używany jest identyfikator CCSID menedżera kolejek.

Informacje pokrewne

Konwersja danych

[Plik ccsid_part2.tbl](#)

Użycie deskryptora komunikatu w interfejsie MQAI

Deskryptor komunikatu generowanego przez interfejs MQAI jest ustawiany na wartość początkową podczas tworzenia worka danych.

Typ komendy PCF jest uzyskiwany z elementu systemu z selektorem MQIASY_TYPE. Podczas tworzenia worka danych początkowa wartość tego elementu jest ustawiana w zależności od typu tworzonej przez użytkownika torby:

<i>Tabela 3. Typ komendy PCF</i>	
Rodzaj torby	Początkowa wartość elementu MQIASY_TYPE
MQCBO_ADMIN_BAG	MQCFT_COMMAND
MQCBO_COMMAND_BAG	MQCFT_COMMAND
MQCBO_*	MQCFT_USER,

Gdy interfejs MQAI generuje deskryptor komunikatu, wartości używane w parametrach **Format** i **MsgType** zależą od wartości elementu systemu z selektorem MQIASY_TYPE, co pokazano w sekcji Tabela 3 na stronie 40.

Tabela 4. Parametry formatu i MsgType deskryptora MQMD

Typ komendy PCF	Format	MsgType
MQCFT_COMMAND	ADMINISTRATOR MQFMT_ADMIN	MQMT_REQUEST
RAPORT MQCFT_REPORT	ADMINISTRATOR MQFMT_ADMIN	Raport_menedżera_mQMT
MQCFT_RESPONSE	ADMINISTRATOR MQFMT_ADMIN	MQMT_REPLY
MQCFT_TRACE_ROUTE	ADMINISTRATOR MQFMT_ADMIN	MQMT_DATAGRAM
MQCFT_EVENT,	Zdarzenie MQFMT_EVENT	MQMT_DATAGRAM
MQCFT_*	MQFMT_PCF	MQMT_DATAGRAM

Tabela 4 na stronie 41 pokazuje, że jeśli zostanie utworzony worek administracyjny lub worek komend, *Format* deskryptora komunikatu to MQFMT_ADMIN, a *MsgType* to MQMT_REQUEST. Jest to odpowiednie dla komunikatu żądania PCF wysłanego do serwera komend, gdy oczekiwana jest odpowiedź z powrotem.

Inne parametry w deskrypcji komunikatu przyjmują wartości wyświetlane w programie Tabela 5 na stronie 41.

Tabela 5. Wartości deskryptora komunikatu

Parametr	Wartość
<i>StrucId</i>	MQMD_STRUC_ID
<i>Version</i>	MQMD_VERSION_1
<i>Report</i>	MQRO_NONE
<i>MsgType</i>	Patrz sekcja Tabela 4 na stronie 41
<i>Expiry</i>	30 sekund (uwaga "1" na stronie 42)
<i>Feedback</i>	MQFB_NONE
<i>Encoding</i>	MQENC_NATIVE
<i>CodedCharSetId</i>	zależy od identyfikatora CCSID torby (uwaga "2" na stronie 42)
<i>Format</i>	Patrz sekcja Tabela 4 na stronie 41
<i>Priority</i>	MQPRI_PRIORITY_AS_Q_DEF
<i>Persistence</i>	MQPER_NOT_PERSISTENT
<i>MsgId</i>	MQMI_NONE
<i>CorrelId</i>	MQCI_NONE
<i>BackoutCount</i>	0
<i>ReplyToQ</i>	patrz uwaga "3" na stronie 42
<i>ReplyToQMGr</i>	puste

Uwagi:

1. Tę wartość można przestonić w wywołaniu funkcji `mqExecute` przy użyciu parametru **OptionsBag**. Więcej informacji na ten temat zawiera sekcja `mqExecute`.
2. Więcej informacji zawiera sekcja [“Przetwarzanie konwersji danych w interfejsie MQAI”](#) na stronie 38.
3. Nazwa kolejki odpowiedzi określonej przez użytkownika lub kolejki tymczasowej wygenerowanej przez produkt MQAI dla komunikatów typu `MQMT_REQUEST`. W przeciwnym razie puste.

Przykładowy program C do tworzenia kolejki lokalnej (amqsaicq.c)

Przykładowy program C `amqsaicq.c` tworzy kolejkę lokalną za pomocą interfejsu MQAI.

```
/*
*****
*/
/* Program name: AMQSAICQ.C
*/
/* Description: Sample C program to create a local queue using the
IBM MQ Administration Interface (MQAI).
*/
/* Statement: Licensed Materials - Property of IBM
*/
/* 84H2000, 5765-B73
*/
/* 84H2001, 5639-B42
*/
/* 84H2002, 5765-B74
*/
/* 84H2003, 5765-B75
*/
/* 84H2004, 5639-B43
*/
/* (C) Copyright IBM Corp. 1999, 2023.
*/
*****
*/
/* Function:
*/
/* AMQSAICQ is a sample C program that creates a local queue and is an
example of the use of the mqExecute call.
*/
/* - The name of the queue to be created is a parameter to the program.
*/
/* - A PCF command is built by placing items into an MQAI bag.
These are:-
*/
/* - The name of the queue
*/
/* - The type of queue required, which, in this case, is local.
*/
/* - The mqExecute call is executed with the command MQCMD_CREATE_Q.
The call generates the correct PCF structure.
*/
/* The call receives the reply from the command server and formats into
the response bag.
*/
/* - The completion code from the mqExecute call is checked and if there
is a failure from the command server then the code returned by the
command server is retrieved from the system bag that is
embedded in the response bag to the mqExecute call.
*/
/* Note: The command server must be running.
*/
*/
*****
*/
/* AMQSAICQ has 2 parameters - the name of the local queue to be created
*/
/* - the queue manager name (optional)
*/
*/
*****
*/
/* Includes
*/
*****
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>

#include <cmqc.h> /* MQI */
#include <cmqcfh.h> /* PCF */
#include <cmqbc.h> /* MQAI */

void CheckCallResult(MQCHAR *, MQLONG , MQLONG );
void CreateLocalQueue(MQHCONN, MQCHAR *);
```

```

int main(int argc, char *argv[])
{
    MQHCONN hConn; /* handle to IBM MQ connection */
    MQCHAR QMName[MQ_Q_MGR_NAME_LENGTH+1]=""; /* default QMgr name */
    MQLONG connReason; /* MQCONN reason code */
    MQLONG compCode; /* completion code */
    MQLONG reason; /* reason code */

    /*****
    /* First check the required parameters */
    /*****
    printf("Sample Program to Create a Local Queue\n");
    if (argc < 2)
    {
        printf("Required parameter missing - local queue name\n");
        exit(99);
    }

    /*****
    /* Connect to the queue manager */
    /*****
    if (argc > 2)
        strncpy(QMName, argv[2], (size_t)MQ_Q_MGR_NAME_LENGTH);
        MQCONN(QMName, &hConn, &compCode, &connReason);

    /*****
    /* Report reason and stop if connection failed */
    /*****
    if (compCode == MQCC_FAILED)
    {
        CheckCallResult("MQCONN", compCode, connReason);
        exit( (int)connReason);
    }

    /*****
    /* Call the routine to create a local queue, passing the handle to the
    /* queue manager and also passing the name of the queue to be created.
    /*****
    CreateLocalQueue(hConn, argv[1]);

    /*****
    /* Disconnect from the queue manager if not already connected */
    /*****
    if (connReason != MQRC_ALREADY_CONNECTED)
    {
        MQDISC(&hConn, &compCode, &reason);
        CheckCallResult("MQDISC", compCode, reason);
    }
    return 0;
}

/*****
/*
/* Function: CreateLocalQueue
/* Description: Create a local queue by sending a PCF command to the command
/* server.
/*
/*****
/*
/* Input Parameters: Handle to the queue manager
/* Name of the queue to be created
/*
/* Output Parameters: None
/*
/* Logic: The mqExecute call is executed with the command MQCMD_CREATE_Q.
/* The call generates the correct PCF structure.
/* The default options to the call are used so that the command is sent
/* to the SYSTEM.ADMIN.COMMAND.QUEUE.
/* The reply from the command server is placed on a temporary dynamic
/* queue.
/* The reply is read from the temporary queue and formatted into the
/* response bag.
/*
/* The completion code from the mqExecute call is checked and if there
/* is a failure from the command server then the code returned by the
/* command server is retrieved from the system bag that is
/* embedded in the response bag to the mqExecute call.
/*****
void CreateLocalQueue(MQHCONN hConn, MQCHAR *qName)

```

```

{
MQLONG reason;                /* reason code                */
MQLONG compCode;              /* completion code            */
MQHBAG commandBag = MQHB_UNUSABLE_HBAG; /* command bag for mqExecute */
MQHBAG responseBag = MQHB_UNUSABLE_HBAG; /* response bag for mqExecute */
MQHBAG resultBag;            /* result bag from mqExecute */
MQLONG mqExecuteCC;          /* mqExecute completion code */
MQLONG mqExecuteRC;          /* mqExecute reason code     */

printf("\nCreating Local Queue %s\n\n", qName);

/*****
/* Create a command Bag for the mqExecute call. Exit the function if the
/* create fails.
*****/
mqCreateBag(MQCBO_ADMIN_BAG, &commandBag, &compCode, &reason);
CheckCallResult("Create the command bag", compCode, reason);
if (compCode !=MQCC_OK)
    return;

/*****
/* Create a response Bag for the mqExecute call, exit the function if the
/* create fails.
*****/
mqCreateBag(MQCBO_ADMIN_BAG, &responseBag, &compCode, &reason);
CheckCallResult("Create the response bag", compCode, reason);
if (compCode !=MQCC_OK)
    return;

/*****
/* Put the name of the queue to be created into the command bag. This will
/* be used by the mqExecute call.
*****/
mqAddString(commandBag, MQCA_Q_NAME, MQBL_NULL_TERMINATED, qName, &compCode,
            &reason);
CheckCallResult("Add q name to command bag", compCode, reason);

/*****
/* Put queue type of local into the command bag. This will be used by the
/* mqExecute call.
*****/
mqAddInteger(commandBag, MQIA_Q_TYPE, MQQT_LOCAL, &compCode, &reason);
CheckCallResult("Add q type to command bag", compCode, reason);

/*****
/* Send the command to create the required local queue.
/* The mqExecute call will create the PCF structure required, send it to
/* the command server and receive the reply from the command server into
/* the response bag.
*****/
mqExecute(hConn,                /* IBM MQ connection handle */
          MQCMD_CREATE_Q,        /* Command to be executed   */
          MQHB_NONE,            /* No options bag            */
          commandBag,           /* Handle to bag containing commands */
          responseBag,          /* Handle to bag to receive the response*/
          MQHO_NONE,            /* Put msg on SYSTEM.ADMIN.COMMAND.QUEUE*/
          MQHO_NONE,            /* Create a dynamic q for the response */
          &compCode,            /* Completion code from the mqExecute */
          &reason);            /* Reason code from mqExecute call */

if (reason == MQRC_CMD_SERVER_NOT_AVAILABLE)
{
    printf("Please start the command server: <strmqcsv QMgrName>\n")
    MQDISC(&hConn, &compCode, &reason);
    CheckCallResult("MQDISC", compCode, reason);
    exit(98);
}

/*****
/* Check the result from mqExecute call and find the error if it failed.
*****/
if ( compCode == MQCC_OK )
    printf("Local queue %s successfully created\n", qName);
else
{
    printf("Creation of local queue %s failed: Completion Code = %d
           qName, compCode, reason);
    if (reason == MQRCCF_COMMAND_FAILED)
    {
        /*****

```

```

        /* Get the system bag handle out of the mqExecute response bag. */
        /* This bag contains the reason from the command server why the */
        /* command failed. */
        /*******/
        mqInquireBag(responseBag, MQHA_BAG_HANDLE, 0, &resultBag, &compCode,
                    &reason);
        CheckCallResult("Get the result bag handle", compCode, reason);

        /*******/
        /* Get the completion code and reason code, returned by the command */
        /* server, from the embedded error bag. */
        /*******/
        mqInquireInteger(resultBag, MQIASY_COMP_CODE, MQIND_NONE, &mqExecuteCC,
                        &compCode, &reason);
        CheckCallResult("Get the completion code from the result bag",
                        compCode, reason);
        mqInquireInteger(resultBag, MQIASY_REASON, MQIND_NONE, &mqExecuteRC,
                        &compCode, &reason);
        CheckCallResult("Get the reason code from the result bag", compCode,
                        reason);
        printf("Error returned by the command server: Completion code = %d :
                Reason = %d\n", mqExecuteCC, mqExecuteRC);
    }
}
/*******/
/* Delete the command bag if successfully created. */
/*******/
if (commandBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&commandBag, &compCode, &reason);
    CheckCallResult("Delete the command bag", compCode, reason);
}

/*******/
/* Delete the response bag if successfully created. */
/*******/
if (responseBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&responseBag, &compCode, &reason);
    CheckCallResult("Delete the response bag", compCode, reason);
}
} /* end of CreateLocalQueue */

/*******/
/*
/* Function: CheckCallResult
/*
/*******/
/*
/* Input Parameters: Description of call
/* Completion code
/* Reason code
/*
/* Output Parameters: None
/*
/* Logic: Display the description of the call, the completion code and the
/* reason code if the completion code is not successful
/*
/*******/
void CheckCallResult(char *callText, MQLONG cc, MQLONG rc)
{
    if (cc != MQCC_OK)
        printf("%s failed: Completion Code = %d :
                Reason = %d\n", callText, cc, rc);
}
}

```

Przykładowy program C do wyświetlania zdarzeń przy użyciu monitora zdarzeń (amqsaiem.c)

Przykładowy program C amqsaiem.c demonstruje podstawowy monitor zdarzeń przy użyciu interfejsu MQAI.

```

/*******/
/*
/* Program name: AMQSAIEM.C
/*

```

```

/*
/* Description: Sample C program to demonstrate a basic event monitor
/* using the IBM MQ Admin Interface (MQAI).
/* Licensed Materials - Property of IBM
/*
/* 63H9336
/* (c) Copyright IBM Corp. 1999, 2023. All Rights Reserved.
/*
/* US Government Users Restricted Rights - Use, duplication or
/* disclosure restricted by GSA ADP Schedule Contract with
/* IBM Corp.
/*****
/*
/* Function:
/* AMQSAIEM is a sample C program that demonstrates how to write a simple
/* event monitor using the mqGetBag call and other MQAI calls.
/*
/* The name of the event queue to be monitored is passed as a parameter
/* to the program. This would usually be one of the system event queues:-
/* SYSTEM.ADMIN.QMGR.EVENT Queue Manager events
/* SYSTEM.ADMIN.PERFM.EVENT Performance events
/* SYSTEM.ADMIN.CHANNEL.EVENT Channel events
/* SYSTEM.ADMIN.LOGGER.EVENT Logger events
/*
/* To monitor the queue manager event queue or the performance event queue,
/* the attributes of the queue manager need to be changed to enable
/* these events. For more information about this, see Part 1 of the
/* Programmable System Management book. The queue manager attributes can
/* be changed using either MQSC commands or the MQAI interface.
/* Channel events are enabled by default.
/*
/* Program logic
/* Connect to the Queue Manager.
/* Open the requested event queue with a wait interval of 30 seconds.
/* Wait for a message, and when it arrives get the message from the queue
/* and format it into an MQAI bag using the mqGetBag call.
/* There are many types of event messages and it is beyond the scope of
/* this sample to program for all event messages. Instead the program
/* prints out the contents of the formatted bag.
/* Loop around to wait for another message until either there is an error
/* or the wait interval of 30 seconds is reached.
/*
/*****
/*
/* AMQSAIEM has 2 parameters - the name of the event queue to be monitored
/* - the queue manager name (optional)
/*
/*****

/*****
/* Includes
/*****
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>

#include <cmqc.h> /* MQI
#include <cmqcfh.h> /* PCF
#include <cmqbc.h> /* MQAI

/*****
/* Macros
/*****
#if MQAT_DEFAULT == MQAT_WINDOWS_NT
#define Int64 "I64"
#elif defined(MQ_64_BIT)
#define Int64 "l"
#else
#define Int64 "ll"
#endif

/*****
/* Function prototypes
/*****
void CheckCallResult(MQCHAR *, MQLONG , MQLONG);
void GetQEvents(MQHCONN, MQCHAR *);
int PrintBag(MQHBAG);
int PrintBagContents(MQHBAG, int);

/*****
/* Function: main
*/

```

```

/*****
int main(int argc, char *argv[])
{
    MQHCONN hConn;          /* handle to connection          */
    MQCHAR QMName[MQ_Q_MGR_NAME_LENGTH+1]=""; /* default QM name          */
    MQLONG reason;         /* reason code              */
    MQLONG connReason;     /* MQCONN reason code      */
    MQLONG compCode;       /* completion code         */

    /*****
    /* First check the required parameters          */
    /*****
    printf("Sample Event Monitor (times out after 30 secs)\n");
    if (argc < 2)
    {
        printf("Required parameter missing - event queue to be monitored\n");
        exit(99);
    }

    /*****
    /* Connect to the queue manager                */
    /*****
    if (argc > 2)
        stncpy(QMName, argv[2], (size_t)MQ_Q_MGR_NAME_LENGTH);
    MQCONN(QMName, &hConn, &compCode, &connReason);
    /*****
    /* Report the reason and stop if the connection failed */
    /*****
    if (compCode == MQCC_FAILED)
    {
        CheckCallResult("MQCONN", compCode, connReason);
        exit( (int)connReason);
    }

    /*****
    /* Call the routine to open the event queue and format any event messages */
    /* read from the queue.          */
    /*****
    GetQEvents(hConn, argv[1]);

    /*****
    /* Disconnect from the queue manager if not already connected */
    /*****
    if (connReason != MQRC_ALREADY_CONNECTED)
    {
        MQDISC(&hConn, &compCode, &reason);
        CheckCallResult("MQDISC", compCode, reason);
    }

    return 0;
}

/*****
/*
/* Function: CheckCallResult          */
/*
/*
/*****
/*
/* Input Parameters:  Description of call          */
/*                   Completion code              */
/*                   Reason code                  */
/*
/* Output Parameters: None          */
/*
/* Logic: Display the description of the call, the completion code and the
/*        reason code if the completion code is not successful
/*
/*****
void CheckCallResult(char *callText, MQLONG cc, MQLONG rc)
{
    if (cc != MQCC_OK)
        printf("%s failed: Completion Code = %d : Reason = %d\n",
            callText, cc, rc);
}

/*****
/*
/* Function: GetQEvents          */
/*
/*****

```

```

/*                                                                    */
/* Input Parameters:  Handle to the queue manager                      */
/*                   Name of the event queue to be monitored          */
/*                                                                    */
/* Output Parameters: None                                           */
/*                                                                    */
/* Logic:  Open the event queue.                                     */
/*         Get a message off the event queue and format the message into */
/*         a bag.                                                  */
/*         A real event monitor would need to be programmed to deal with */
/*         each type of event that it receives from the queue. This is */
/*         outside the scope of this sample, so instead, the contents of */
/*         the bag are printed.                                     */
/*         The program waits for 30 seconds for an event message and then */
/*         terminates if no more messages are available.          */
/*                                                                    */
/*****
void GetQEvents(MQHCONN hConn, MQCHAR *qName)
{
    MQLONG openReason;                /* MQOPEN reason code          */
    MQLONG reason;                    /* reason code                  */
    MQLONG compCode;                  /* completion code             */
    MQHOBJ eventQueue;                /* handle to event queue       */

    MQHBAG eventBag = MQHB_UNUSABLE_HBAG; /* event bag to receive event msg */
    MQOD od = {MQOD_DEFAULT};          /* Object Descriptor           */
    MQMD md = {MQMD_DEFAULT};          /* Message Descriptor          */
    MQGMO gmo = {MQGMO_DEFAULT};       /* get message options         */
    MQLONG bQueueOK = 1;                /* keep reading msgs while true */

    /*****
    /* Create an Event Bag in which to receive the event.             */
    /* Exit the function if the create fails.                         */
    /*****
    mqCreateBag(MQCBO_USER_BAG, &eventBag, &compCode, &reason);
    CheckCallResult("Create event bag", compCode, reason);
    if (compCode != MQCC_OK)
        return;

    /*****
    /* Open the event queue chosen by the user                         */
    /*****
    strncpy(od.ObjectName, qName, (size_t)MQ_Q_NAME_LENGTH);
    MQOPEN(hConn, &od, MQOO_INPUT_AS_Q_DEF+MQOO_FAIL_IF QUIESCING, &eventQueue,
            &compCode, &openReason);
    CheckCallResult("Open event queue", compCode, openReason);

    /*****
    /* Set the GMO options to control the action of the get message from the */
    /* queue.                                                                    */
    /*****
    gmo.WaitInterval = 30000;          /* 30 second wait for message */
    gmo.Options = MQGMO_WAIT + MQGMO_FAIL_IF QUIESCING + MQGMO_CONVERT;
    gmo.Version = MQGMO_VERSION_2;    /* Avoid need to reset Message ID */
    gmo.MatchOptions = MQMO_NONE;     /* and Correlation ID after every */
                                     /* mqGetBag

    /*****
    /* If open fails, we cannot access the queue and must stop the monitor. */
    /*****
    if (compCode != MQCC_OK)
        bQueueOK = 0;

    /*****
    /* Main loop to get an event message when it arrives                */
    /*****
    while (bQueueOK)
    {
        printf("\nWaiting for an event\n");

        /*****
        /* Get the message from the event queue and convert it into the event */
        /* bag.                                                                    */
        /*****
        mqGetBag(hConn, eventQueue, &md, &gmo, eventBag, &compCode, &reason);

        /*****
        /* If get fails, we cannot access the queue and must stop the monitor. */
        /*****
        if (compCode != MQCC_OK)
            {
                bQueueOK = 0;

```



```

/*****
/* If get fails because no message available then we have timed out, */
/* so report this, otherwise report an error. */
/*****
if (reason == MQRC_NO_MSG_AVAILABLE)
{
    printf("No more messages\n");
}
else
{
    CheckCallResult("Get bag", compCode, reason);
}
}

/*****
/* Event message read - Print the contents of the event bag */
/*****
else
{
    if ( PrintBag(eventBag) )
        printf("\nError found while printing bag contents\n");

} /* end of msg found */
} /* end of main loop */
/*****
/* Close the event queue if successfully opened */
/*****
if (openReason == MQRC_NONE)
{
    MQCLOSE(hConn, &eventQueue, MQCO_NONE, &compCode, &reason);
    CheckCallResult("Close event queue", compCode, reason);
}

/*****
/* Delete the event bag if successfully created. */
/*****
if (eventBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&eventBag, &compCode, &reason);
    CheckCallResult("Delete the event bag", compCode, reason);
}

} /* end of GetQEvents */

/*****
/*
/* Function: PrintBag
/*
/*****
/*
/* Input Parameters: Bag Handle
/*
/*
/* Output Parameters: None
/*
/*
/* Returns: Number of errors found
/*
/*
/* Logic: Calls PrintBagContents to display the contents of the bag.
/*
/*
/*****

int PrintBag(MQHBAG dataBag)
{
    int errors;

    printf("\n");
    errors = PrintBagContents(dataBag, 0);
    printf("\n");

    return errors;
}

/*****
/*
/* Function: PrintBagContents
/*
/*****
/*
/* Input Parameters: Bag Handle
/*
/* Indentation level of bag
/*
/*
/* Output Parameters: None
/*
/*
/*****

```

```

/* Returns:          Number of errors found          */
/*                                                         */
/* Logic: Count the number of items in the bag          */
/* Obtain selector and item type for each item in the bag. */
/* Obtain the value of the item depending on item type and display the */
/* index of the item, the selector and the value.      */
/* If the item is an embedded bag handle then call this function again */
/* to print the contents of the embedded bag increasing the */
/* indentation level.                                  */
/*                                                         */
/******
int PrintBagContents(MQHBAG dataBag, int indent)
{
    /******
    /* Definitions
    /******
    #define LENGTH 500          /* Max length of string to be read*/
    #define INDENT 4           /* Number of spaces to indent   */
    /* embedded bag display    */

    /******
    /* Variables
    /******
    MQLONG  itemCount;          /* Number of items in the bag */
    MQLONG  itemType;          /* Type of the item           */
    int     i;                 /* Index of item in the bag   */
    MQCHAR  stringVal[LENGTH+1]; /* Value if item is a string  */
    MQBYTE  byteStringVal[LENGTH]; /* Value if item is a byte string */
    MQLONG  stringLength;      /* Length of string value     */
    MQLONG  ccSID;             /* CCSID of string value     */
    MQINT32 iValue;            /* Value if item is an integer */
    MQINT64 i64Value;          /* Value if item is a 64-bit  */
    /* integer                  */
    MQLONG  selector;          /* Selector of item           */
    MQHBAG  bagHandle;         /* Value if item is a bag handle */
    MQLONG  reason;            /* reason code                 */
    MQLONG  compCode;          /* completion code            */
    MQLONG  trimLength;        /* Length of string to be trimmed */
    int     errors = 0;        /* Count of errors found      */
    char    blanks[] = "     "; /* Blank string used to      */
    /* indent display          */

    /******
    /* Count the number of items in the bag
    /******
    mqCountItems(dataBag, MQSEL_ALL_SELECTORS, &iItemCount, &compCode, &reason);

    if (compCode != MQCC_OK)
        errors++;
    else
    {
        printf("
        printf("
        printf("
    }

    /******
    /* If no errors found, display each item in the bag
    /******
    if (!errors)
    {
        for (i = 0; i < itemCount; i++)
        {
            /******
            /* First inquire the type of the item for each item in the bag
            /******
            mqInquireItemInfo(dataBag,          /* Bag handle
                               MQSEL_ANY_SELECTOR, /* Item can have any selector*/
                               i,                /* Index position in the bag */
                               &selector,        /* Actual value of selector */
                                               /* returned by call
                               &iItemType,        /* Actual type of item
                                               /* returned by call
                               &compCode,        /* Completion code
                               &reason);        /* Reason Code

            if (compCode != MQCC_OK)
                errors++;

            switch(itemType)

```

```

{
case MQITEM_INTEGER:
/*****
/* Item is an integer. Find its value and display its index, */
/* selector and value. */
/*****
mqInquireInteger(dataBag, /* Bag handle */
MQSEL_ANY_SELECTOR, /* Allow any selector */
i, /* Index position in the bag */
&iValue, /* Returned integer value */
&compCode, /* Completion code */
&reason); /* Reason Code */

if (compCode != MQCC_OK)
errors++;
else
printf("%.s %-2d %-4d (%d)\n",
indent, blanks, i, selector, iValue);
break

case MQITEM_INTEGER64:
/*****
/* Item is a 64-bit integer. Find its value and display its */
/* index, selector and value. */
/*****
mqInquireInteger64(dataBag, /* Bag handle */
MQSEL_ANY_SELECTOR, /* Allow any selector */
i, /* Index position in the bag */
&i64Value, /* Returned integer value */
&compCode, /* Completion code */
&reason); /* Reason Code */

if (compCode != MQCC_OK)
errors++;
else
printf("%.s %-2d %-4d (%"Int64"d)\n",
indent, blanks, i, selector, i64Value);
break;

case MQITEM_STRING:
/*****
/* Item is a string. Obtain the string in a buffer, prepare */
/* the string for displaying and display the index, selector, */
/* string and Character Set ID. */
/*****
mqInquireString(dataBag, /* Bag handle */
MQSEL_ANY_SELECTOR, /* Allow any selector */
i, /* Index position in the bag */
LENGTH, /* Maximum length of buffer */
stringVal, /* Buffer to receive string */
&stringLength, /* Actual length of string */
&ccsid, /* Coded character set ID */
&compCode, /* Completion code */
&reason); /* Reason Code */

/*****
/* The call can return a warning if the string is too long for */
/* the output buffer and has been truncated, so only check */
/* explicitly for call failure. */
/*****
if (compCode == MQCC_FAILED)
errors++;
else
{
/*****
/* Remove trailing blanks from the string and terminate with*/
/* a null. First check that the string should not have been */
/* longer than the maximum buffer size allowed. */
/*****
if (stringLength > LENGTH)
trimLength = LENGTH;
else
trimLength = stringLength;
mqTrim(trimLength, stringVal, stringVal, &compCode, &reason);
printf("%.s %-2d %-4d '%S' %d\n",
indent, blanks, i, selector, stringVal, ccsid);
}
break;

case MQITEM_BYTE_STRING:
/*****

```

```

/* Item is a byte string. Obtain the byte string in a buffer, */
/* prepare the byte string for displaying and display the */
/* index, selector and string. */
/*******/
mqInquireByteString(dataBag, /* Bag handle */
                    MQSEL_ANY_SELECTOR, /* Allow any selector */
                    i, /* Index position in the bag */
                    LENGTH, /* Maximum length of buffer */
                    byteStringVal, /* Buffer to receive string */
                    &stringLength, /* Actual length of string */
                    &compCode, /* Completion code */
                    &reason); /* Reason Code

/*******/
/* The call can return a warning if the string is too long for */
/* the output buffer and has been truncated, so only check */
/* explicitly for call failure. */
/*******/
if (compCode == MQCC_FAILED)
    errors++;
else
{
    printf("%.s %-2d %-4d X'",
           indent, blanks, i, selector);

    for (i = 0 ; i < stringLength ; i++)
        printf("

    printf("'\\n");
}
break;

case MQITEM_BAG:
/*******/
/* Item is an embedded bag handle, so call the PrintBagContents*/
/* function again to display the contents. */
/*******/
mqInquireBag(dataBag, /* Bag handle */
             MQSEL_ANY_SELECTOR, /* Allow any selector */
             i, /* Index position in the bag */
             &bagHandle, /* Returned embedded bag hdl*/
             &compCode, /* Completion code */
             &reason); /* Reason Code

if (compCode != MQCC_OK)
    errors++;
else
{
    printf("%.s %-2d %-4d (%d)\\n", indent, blanks, i,
           selector, bagHandle);
    if (selector == MQHA_BAG_HANDLE)
        printf("
    else
        printf("
        PrintBagContents(bagHandle, indent+INDENT);
}
break;

default:
    printf("
}
}
}
return errors;
}

```

Przykładowy program w języku C dla zapytań o obiekty kanału (amqsaicl.c)

Przykładowy program C amqsaicl.c zapytania o obiekty kanału przy użyciu interfejsu MQAI.

```

/*******/
/*
/* Program name: AMQSAICL.C */
/*
/* Description: Sample C program to inquire channel objects */
/* using the IBM MQ Administration Interface (MQAI) */
/*
/* <N_OCO_COPYRIGHT> */

```

```

/* Licensed Materials - Property of IBM */
/* */
/* 63H9336 */
/* (c) Copyright IBM Corp. 2008, 2023. All Rights Reserved. */
/* */
/* US Government Users Restricted Rights - Use, duplication or */
/* disclosure restricted by GSA ADP Schedule Contract with */
/* IBM Corp. */
/* <NOC_COPYRIGHT> */
/***** */
/* */
/* Function: */
/* AMQSAICL is a sample C program that demonstrates how to inquire */
/* attributes of the local queue manager using the MQAI interface. In */
/* particular, it inquires all channels and their types. */
/* */
/* - A PCF command is built from items placed into an MQAI administration */
/* bag. */
/* These are:- */
/* - The generic channel name "*" */
/* - The attributes to be inquired. In this sample we just want */
/* name and type attributes */
/* */
/* - The mqExecute MQCMD_INQUIRE_CHANNEL call is executed. */
/* The call generates the correct PCF structure. */
/* The default options to the call are used so that the command is sent */
/* to the SYSTEM.ADMIN.COMMAND.QUEUE. */
/* The reply from the command server is placed on a temporary dynamic */
/* queue. */
/* The reply from the MQCMD_INQUIRE_CHANNEL is read from the */
/* temporary queue and formatted into the response bag. */
/* */
/* - The completion code from the mqExecute call is checked and if there */
/* is a failure from the command server, then the code returned by the */
/* command server is retrieved from the system bag that has been */
/* embedded in the response bag to the mqExecute call. */
/* */
/* Note: The command server must be running. */
/* */
/***** */
/* AMQSAICL has 2 parameter - the queue manager name (optional) */
/* - output file (optional) default varies */
/***** */

/***** */
/* Includes */
/***** */
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>
#if (MQAT_DEFAULT == MQAT_OS400)
#include <recio.h>
#endif

#include <cmqc.h> /* MQI */
#include <cmqcf.h> /* PCF */
#include <cmqbc.h> /* MQAI */
#include <cmqxc.h> /* MQCD */

/***** */
/* Function prototypes */
/***** */
void CheckCallResult(MQCHAR *, MQLONG , MQLONG);

/***** */
/* DataTypes */
/***** */
#if (MQAT_DEFAULT == MQAT_OS400)
typedef _RFIL FILE;
#else
typedef FILE FILE;
#endif

/***** */
/* Constants */
/***** */
#if (MQAT_DEFAULT == MQAT_OS400)
const struct
{
char name[9];

```

```

} ChlTypeMap[9] =
{
  "SDR      ", /* MQCHT_SENDER */
  "SVR      ", /* MQCHT_SERVER */
  "RCVR     ", /* MQCHT_RECEIVER */
  "RQSTR    ", /* MQCHT_REQUESTER */
  "ALL      ", /* MQCHT_ALL */
  "CLTCN    ", /* MQCHT_CLNTCONN */
  "SVRCONN  ", /* MQCHT_SVRCONN */
  "CLUSRCVR", /* MQCHT_CLUSRCVR */
  "CLUSSDR  " /* MQCHT_CLUSSDR */
};
#else
const struct
{
  char name[9];
} ChlTypeMap[9] =
{
  "sdr      ", /* MQCHT_SENDER */
  "svr      ", /* MQCHT_SERVER */
  "rcvr     ", /* MQCHT_RECEIVER */
  "rqstr    ", /* MQCHT_REQUESTER */
  "all      ", /* MQCHT_ALL */
  "cltconn  ", /* MQCHT_CLNTCONN */
  "svrcn    ", /* MQCHT_SVRCONN */
  "clusrcvr", /* MQCHT_CLUSRCVR */
  "clussdr  " /* MQCHT_CLUSSDR */
};
#endif

/*****
/* Macros */
/*****
/*****
*/
#define OUTFILE "QTEMP/AMQSAICL(AMQSAICL)"
#define OPENOUTFILE(hdl, fname) \
  (hdl) = _Ropen((fname), "wr", rtncode=Y);
#define CLOSEOUTFILE(hdl) \
  _Rclose((hdl));
#define WRITEOUTFILE(hdl, buf, buflen) \
  _Rwrite((hdl), (buf), (buflen));

#elif (MQAT_DEFAULT == MQAT_UNIX)
#define OUTFILE "/tmp/amqsaicl.txt"
#define OPENOUTFILE(hdl, fname) \
  (hdl) = fopen((fname), "w");
#define CLOSEOUTFILE(hdl) \
  fclose((hdl));
#define WRITEOUTFILE(hdl, buf, buflen) \
  fwrite((buf), (buflen), 1, (hdl)); fflush((hdl));

#else
#define OUTFILE "amqsaicl.txt"
#define OPENOUTFILE(fname) \
  fopen((fname), "w");
#define CLOSEOUTFILE(hdl) \
  fclose((hdl));
#define WRITEOUTFILE(hdl, buf, buflen) \
  fwrite((buf), (buflen), 1, (hdl)); fflush((hdl));

#endif

#define ChlType2String(t) ChlTypeMap[(t)-1].name

/*****
/* Function: main */
/*****
int main(int argc, char *argv[])
{
  /*****
  /* MQAI variables */
  /*****
  MQHCONN hConn; /* handle to MQ connection */
  MQCHAR qmName[MQ_Q_MGR_NAME_LENGTH+1]=""; /* default QMgr name */
  MQLONG reason; /* reason code */
  MQLONG connReason; /* MQCONN reason code */
  MQLONG compCode; /* completion code */
  MQHBAG adminBag = MQHB_UNUSABLE_HBAG; /* admin bag for mqExecute */
  MQHBAG responseBag = MQHB_UNUSABLE_HBAG; /* response bag for mqExecute */
  MQHBAG cAttrsBag; /* bag containing chl attributes */
  MQHBAG errorBag; /* bag containing cmd server error */
  MQLONG mqExecuteCC; /* mqExecute completion code */

```

```

MQLONG mqExecuteRC; /* mqExecute reason code */
MQLONG chlNameLength; /* Actual length of chl name */
MQLONG chlType; /* Channel type */
MQLONG i; /* loop counter */
MQLONG numberOfBags; /* number of bags in response bag */
MQCHAR chlName[MQ_OBJECT_NAME_LENGTH+1]; /* name of chl extracted from bag */
MQCHAR OutputBuffer[100]; /* output data buffer */
OUTFILEHDL *outfp = NULL; /* output file handle */

/*****
/* Connect to the queue manager */
/*****
if (argc > 1)
    strncpy(qmName, argv[1], (size_t)MQ_Q_MGR_NAME_LENGTH);
MQCONN(qmName, &hConn, &compCode, &connReason);

/*****
/* Report the reason and stop if the connection failed. */
/*****
if (compCode == MQCC_FAILED)
{
    CheckCallResult("Queue Manager connection", compCode, connReason);
    exit( (int)connReason);
}

/*****
/* Open the output file */
/*****
if (argc > 2)
{
    OPENOUTFILE(outfp, argv[2]);
}
else
{
    OPENOUTFILE(outfp, OUTFILE);
}

if(outfp == NULL)
{
    printf("Could not open output file.\n");
    goto MOD_EXIT;
}

/*****
/* Create an admin bag for the mqExecute call */
/*****
mqCreateBag(MQCBO_ADMIN_BAG, &adminBag, &compCode, &reason);
CheckCallResult("Create admin bag", compCode, reason);

/*****
/* Create a response bag for the mqExecute call */
/*****
mqCreateBag(MQCBO_ADMIN_BAG, &responseBag, &compCode, &reason);
CheckCallResult("Create response bag", compCode, reason);

/*****
/* Put the generic channel name into the admin bag */
/*****
mqAddString(adminBag, MQCACH_CHANNEL_NAME, MQBL_NULL_TERMINATED, "*",
            &compCode, &reason);
CheckCallResult("Add channel name", compCode, reason);

/*****
/* Put the channel type into the admin bag */
/*****
mqAddInteger(adminBag, MQIACH_CHANNEL_TYPE, MQCHT_ALL, &compCode, &reason);
CheckCallResult("Add channel type", compCode, reason);

/*****
/* Add an inquiry for various attributes */
/*****
mqAddInquiry(adminBag, MQIACH_CHANNEL_TYPE, &compCode, &reason);
CheckCallResult("Add inquiry", compCode, reason);

/*****
/* Send the command to find all the channel names and channel types. */
/* The mqExecute call creates the PCF structure required, sends it to */
/* the command server, and receives the reply from the command server into */
/* the response bag. The attributes are contained in system bags that are */
/* embedded in the response bag, one set of attributes per bag. */
/*****
mqExecute(hConn, /* MQ connection handle */
          MQCMD_INQUIRE_CHANNEL, /* Command to be executed */

```

```

MQHB_NONE,          /* No options bag          */
adminBag,           /* Handle to bag containing commands */
responseBag,       /* Handle to bag to receive the response*/
MQHO_NONE,         /* Put msg on SYSTEM.ADMIN.COMMAND.QUEUE*/
MQHO_NONE,         /* Create a dynamic q for the response */
&compCode;,        /* Completion code from the mqexecute */
&reason;);         /* Reason code from mqexecute call   */

/*****
/* Check the command server is started. If not exit.
*****/
if (reason == MQRC_CMD_SERVER_NOT_AVAILABLE)
{
    printf("Please start the command server: <strmqcsv QMgrName="">\n");
    goto MOD_EXIT;
}

/*****
/* Check the result from mqExecute call. If successful find the channel
/* types for all the channels. If failed find the error.
*****/
if ( compCode == MQCC_OK )          /* Successful mqExecute */
{
    /*****
    /* Count the number of system bags embedded in the response bag from the
    /* mqExecute call. The attributes for each channel are in separate bags.
    *****/
    mqCountItems(responseBag, MQHA_BAG_HANDLE, &numberOfBags;,
                &compCode;, &reason;);
    CheckCallResult("Count number of bag handles", compCode, reason);

    for ( i=0; i<numberOfBags; i++)
    {
        /*****
        /* Get the next system bag handle out of the mqExecute response bag.
        /* This bag contains the channel attributes
        *****/
        mqInquireBag(responseBag, MQHA_BAG_HANDLE, i, &AttrsBag,
                    &compCode, &reason);
        CheckCallResult("Get the result bag handle", compCode, reason);

        /*****
        /* Get the channel name out of the channel attributes bag
        *****/
        mqInquireString(cAttrsBag, MQCACH_CHANNEL_NAME, 0, MQ_OBJECT_NAME_LENGTH,
                        chlName, &chlNameLength, NULL, &compCode, &reason);
        CheckCallResult("Get channel name", compCode, reason);

        /*****
        /* Get the channel type out of the channel attributes bag
        *****/
        mqInquireInteger(cAttrsBag, MQIACH_CHANNEL_TYPE, MQIND_NONE, &chlType,
                        &compCode, &reason);
        CheckCallResult("Get type", compCode, reason);

        /*****
        /* Use mqTrim to prepare the channel name for printing.
        /* Print the result.
        *****/
        mqTrim(MQ_CHANNEL_NAME_LENGTH, chlName, chlName, &compCode, &reason);
        sprintf(OutputBuffer, "%-20s%-9s", chlName, ChlType2String(chlType));
        WRITEOUTFILE(outfp,OutputBuffer,29)
    }
}

else
{
    printf("Call to get channel attributes failed: Cc = %ld : Rc = %ld\n",
          compCode, reason);
    /*****
    /* If the command fails get the system bag handle out of the mqexecute
    /* response bag.This bag contains the reason from the command server
    /* why the command failed.
    *****/
    if (reason == MQRCCF_COMMAND_FAILED)
    {
        mqInquireBag(responseBag, MQHA_BAG_HANDLE, 0, &errorBag,
                    &compCode, &reason);
        CheckCallResult("Get the result bag handle", compCode, reason);

        /*****

```



```

        /* Get the completion code and reason code, returned by the command */
        /* server, from the embedded error bag. */
        /******
mqInquireInteger(errorBag, MQIASY_COMP_CODE, MQIND_NONE, &mqExecuteCC,
                &compCode, &reason );
        CheckCallResult("Get the completion code from the result bag",
                compCode, reason);
mqInquireInteger(errorBag, MQIASY_REASON, MQIND_NONE, &mqExecuteRC,
                &compCode, &reason);
        CheckCallResult("Get the reason code from the result bag",
                compCode, reason);
        printf("Error returned by the command server: Cc = %ld : Rc = %ld\n",
                mqExecuteCC, mqExecuteRC);
    }
}

MOD_EXIT:
/******
/* Delete the admin bag if successfully created. */
/******
if (adminBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&adminBag, &compCode, &reason);
    CheckCallResult("Delete the admin bag", compCode, reason);
}

/******
/* Delete the response bag if successfully created. */
/******
if (responseBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&responseBag, &compCode, &reason);
    CheckCallResult("Delete the response bag", compCode, reason);
}

/******
/* Disconnect from the queue manager if not already connected */
/******
if (connReason != MQRC_ALREADY_CONNECTED)
{
    MQDISC(&hConn, &compCode, &reason);
    CheckCallResult("Disconnect from Queue Manager", compCode, reason);
}

/******
/* Close the output file if open */
/******
if(outfp != NULL)
    CLOSEOUTFILE(outfp);

return 0;
}

/******
/*
/* Function: CheckCallResult */
/*
/******
/*
/* Input Parameters:  Description of call */
/*                   Completion code */
/*                   Reason code */
/*
/* Output Parameters: None */
/*
/* Logic: Display the description of the call, the completion code and the */
/*        reason code if the completion code is not successful */
/*
/******
void CheckCallResult(char *callText, MQLONG cc, MQLONG rc)
{
    if (cc != MQCC_OK)
        printf("%s failed: Completion Code = %ld : Reason = %ld\n", callText,
                cc, rc);
}

```

Przykładowy program C do uzyskiwania informacji o kolejkach i drukowaniu informacji (amqsailq.c)

Przykładowy program C amqsailq.c wyświetla informacje o bieżącej głębokości kolejek lokalnych za pomocą interfejsu MQAI.

```
/*
/*
/* Program name: AMQSAILQ.C
/*
/*
/* Description: Sample C program to inquire the current depth of the local
/* queues using the IBM MQ Administration Interface (MQAI)
/*
/*
/* Statement: Licensed Materials - Property of IBM
/*
/*
/* 84H2000, 5765-B73
/* 84H2001, 5639-B42
/* 84H2002, 5765-B74
/* 84H2003, 5765-B75
/* 84H2004, 5639-B43
/*
/* (C) Copyright IBM Corp. 1999, 2023.
/*
/*
/*
/* Function:
/* AMQSAILQ is a sample C program that demonstrates how to inquire
/* attributes of the local queue manager using the MQAI interface. In
/* particular, it inquires the current depths of all the local queues.
/*
/* - A PCF command is built by placing items into an MQAI administration
/* bag.
/* These are:-
/* - The generic queue name "*"
/* - The type of queue required. In this sample we want to
/* inquire local queues.
/* - The attribute to be inquired. In this sample we want the
/* current depths.
/*
/* - The mqExecute call is executed with the command MQCMD_INQUIRE_Q.
/* The call generates the correct PCF structure.
/* The default options to the call are used so that the command is sent
/* to the SYSTEM.ADMIN.COMMAND.QUEUE.
/* The reply from the command server is placed on a temporary dynamic
/* queue.
/* The reply from the MQCMD_INQUIRE_Q command is read from the
/* temporary queue and formatted into the response bag.
/*
/* - The completion code from the mqExecute call is checked and if there
/* is a failure from the command server, then the code returned by
/* command server is retrieved from the system bag that has been
/* embedded in the response bag to the mqExecute call.
/*
/* - If the call is successful, the depth of each local queue is placed
/* in system bags embedded in the response bag of the mqExecute call.
/* The name and depth of each queue is obtained from each of the bags
/* and the result displayed on the screen.
/*
/* Note: The command server must be running.
/*
/*
/*
/* AMQSAILQ has 1 parameter - the queue manager name (optional)
/*
/*
/*
/*
/* Includes
/*
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>

#include <cmqc.h> /* MQI
#include <cmqcfh.h> /* PCF
#include <cmqbc.h> /* MQAI
```

```

/*****
/* Function prototypes
*****/
void CheckCallResult(MQCHAR *, MQLONG , MQLONG);

/*****
/* Function: main
*****/
int main(int argc, char *argv[])
{
    /*****
    /* MQAI variables
    *****/
    MQHCONN hConn; /* handle to IBM MQ connection */
    MQCHAR qmName[MQ_Q_MGR_NAME_LENGTH+1]=""; /* default QMgr name */
    MQLONG reason; /* reason code */
    MQLONG connReason; /* MQCONN reason code */
    MQLONG compCode; /* completion code */
    MQHBAG adminBag = MQHB_UNUSABLE_HBAG; /* admin bag for mqExecute */
    MQHBAG responseBag = MQHB_UNUSABLE_HBAG; /* response bag for mqExecute */
    MQHBAG qAttrsBag; /* bag containing q attributes */
    MQHBAG errorBag; /* bag containing cmd server error */
    MQLONG mqExecuteCC; /* mqExecute completion code */
    MQLONG mqExecuteRC; /* mqExecute reason code */
    MQLONG qNameLength; /* Actual length of q name */
    MQLONG qDepth; /* depth of queue */
    MQLONG i; /* loop counter */
    MQLONG numberOfBags; /* number of bags in response bag */
    MQCHAR qName[MQ_Q_NAME_LENGTH+1]; /* name of queue extracted from bag*/

    printf("Display current depths of local queues\n\n");

    /*****
    /* Connect to the queue manager
    *****/
    if (argc > 1)
        strncpy(qmName, argv[1], (size_t)MQ_Q_MGR_NAME_LENGTH);
    MQCONN(qmName, &hConn, &compCode, &connReason);

    /*****
    /* Report the reason and stop if the connection failed.
    *****/
    if (compCode == MQCC_FAILED)
    {
        CheckCallResult("Queue Manager connection", compCode, connReason);
        exit( (int)connReason);
    }

    /*****
    /* Create an admin bag for the mqExecute call
    *****/
    mqCreateBag(MQCBO_ADMIN_BAG, &adminBag, &compCode, &reason);
    CheckCallResult("Create admin bag", compCode, reason);
    /*****
    /* Create a response bag for the mqExecute call
    *****/
    mqCreateBag(MQCBO_ADMIN_BAG, &responseBag, &compCode, &reason);
    CheckCallResult("Create response bag", compCode, reason);

    /*****
    /* Put the generic queue name into the admin bag
    *****/
    mqAddString(adminBag, MQCA_Q_NAME, MQBL_NULL_TERMINATED, "*",
        &compCode, &reason);
    CheckCallResult("Add q name", compCode, reason);

    /*****
    /* Put the local queue type into the admin bag
    *****/
    mqAddInteger(adminBag, MQIA_Q_TYPE, MQQT_LOCAL, &compCode, &reason);
    CheckCallResult("Add q type", compCode, reason);

    /*****
    /* Add an inquiry for current queue depths
    *****/
    mqAddInquiry(adminBag, MQIA_CURRENT_Q_DEPTH, &compCode, &reason);
    CheckCallResult("Add inquiry", compCode, reason);

    /*****
    /* Send the command to find all the local queue names and queue
    /* The mqExecute call creates the PCF structure required, sends it to

```

```

/* the command server, and receives the reply from the command server into */
/* the response bag. The attributes are contained in system bags that are */
/* embedded in the response bag, one set of attributes per bag. */
/*****
mqExecute(hConn, /* IBM MQ connection handle */
          MQCMD_INQUIRE_Q, /* Command to be executed */
          MQHB_NONE, /* No options bag */
          adminBag, /* Handle to bag containing commands */
          responseBag, /* Handle to bag to receive the response*/
          MQHO_NONE, /* Put msg on SYSTEM.ADMIN.COMMAND.QUEUE*/
          MQHO_NONE, /* Create a dynamic q for the response */
          &compCode, /* Completion code from the mqExecute */
          &reason); /* Reason code from mqExecute call */

/*****
/* Check the command server is started. If not exit. */
/*****
if (reason == MQRC_CMD_SERVER_NOT_AVAILABLE)
{
    printf("Please start the command server: <strmqcsv QMgrName>\n");
    MQDISC(&hConn, &compCode, &reason);
    CheckCallResult("Disconnect from Queue Manager", compCode, reason);
    exit(98);
}

/*****
/* Check the result from mqExecute call. If successful find the current */
/* depths of all the local queues. If failed find the error. */
/*****
if ( compCode == MQCC_OK ) /* Successful mqExecute */
{
    /*****
    /* Count the number of system bags embedded in the response bag from the */
    /* mqExecute call. The attributes for each queue are in a separate bag. */
    /*****
    mqCountItems(responseBag, MQHA_BAG_HANDLE, &numberOfBags, &compCode,
                &reason);
    CheckCallResult("Count number of bag handles", compCode, reason);

    for ( i=0; i<numberOfBags; i++)
    {
        /*****
        /* Get the next system bag handle out of the mqExecute response bag. */
        /* This bag contains the queue attributes */
        /*****
        mqInquireBag(responseBag, MQHA_BAG_HANDLE, i, &qAttrsBag, &compCode,
                    &reason);
        CheckCallResult("Get the result bag handle", compCode, reason);

        /*****
        /* Get the queue name out of the queue attributes bag */
        /*****
        mqInquireString(qAttrsBag, MQCA_Q_NAME, 0, MQ_Q_NAME_LENGTH, qName,
                       &qNameLength, NULL, &compCode, &reason);
        CheckCallResult("Get queue name", compCode, reason);

        /*****
        /* Get the depth out of the queue attributes bag */
        /*****
        mqInquireInteger(qAttrsBag, MQIA_CURRENT_Q_DEPTH, MQIND_NONE, &qDepth,
                        &compCode, &reason);
        CheckCallResult("Get depth", compCode, reason);

        /*****
        /* Use mqTrim to prepare the queue name for printing. */
        /* Print the result. */
        /*****
        mqTrim(MQ_Q_NAME_LENGTH, qName, qName, &compCode, &reason)
        printf("%4d %-48s\n", qDepth, qName);
    }
}

else /* Failed mqExecute */
{
    printf("Call to get queue attributes failed: Completion Code = %d :
          Reason = %d\n", compCode, reason);

    /*****
    /* If the command fails get the system bag handle out of the mqExecute */
    /* response bag. This bag contains the reason from the command server */
    /* why the command failed. */

```

```

/*****
if (reason == MQRCCF_COMMAND_FAILED)
{
    mqInquireBag(responseBag, MQHA_BAG_HANDLE, 0, &errorBag, &compCode,
                &reason);
    CheckCallResult("Get the result bag handle", compCode, reason);

    /*****
    /* Get the completion code and reason code, returned by the command */
    /* server, from the embedded error bag. */
    /*****
    mqInquireInteger(errorBag, MQIASY_COMP_CODE, MQIND_NONE, &mqExecuteCC,
                    &compCode, &reason );
    CheckCallResult("Get the completion code from the result bag",
                    compCode, reason);
    mqInquireInteger(errorBag, MQIASY_REASON, MQIND_NONE, &mqExecuteRC,
                    &compCode, &reason);
    CheckCallResult("Get the reason code from the result bag",
                    compCode, reason);
    printf("Error returned by the command server: Completion Code = %d :
           Reason = %d\n", mqExecuteCC, mqExecuteRC);
}
}

/*****
/* Delete the admin bag if successfully created. */
/*****
if (adminBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&adminBag, &compCode, &reason);
    CheckCallResult("Delete the admin bag", compCode, reason);
}

/*****
/* Delete the response bag if successfully created. */
/*****
if (responseBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&responseBag, &compCode, &reason);
    CheckCallResult("Delete the response bag", compCode, reason);
}

/*****
/* Disconnect from the queue manager if not already connected */
/*****
if (connReason != MQRC_ALREADY_CONNECTED)
{
    MQDISC(&hConn, &compCode, &reason);
    CheckCallResult("Disconnect from queue manager", compCode, reason);
}
return 0;
}

*****/
* Function: CheckCallResult */
* */
*****/
* Input Parameters: Description of call */
* Completion code */
* Reason code */
* */
* Output Parameters: None */
* */
* Logic: Display the description of the call, the completion code and the */
* reason code if the completion code is not successful */
* */
*****/
void CheckCallResult(char *callText, MQLONG cc, MQLONG rc)
{
    if (cc != MQCC_OK)
        printf("%s failed: Completion Code = %d : Reason = %d\n",
              callText, cc, rc);
}

```

Torby z danymi i MQAI

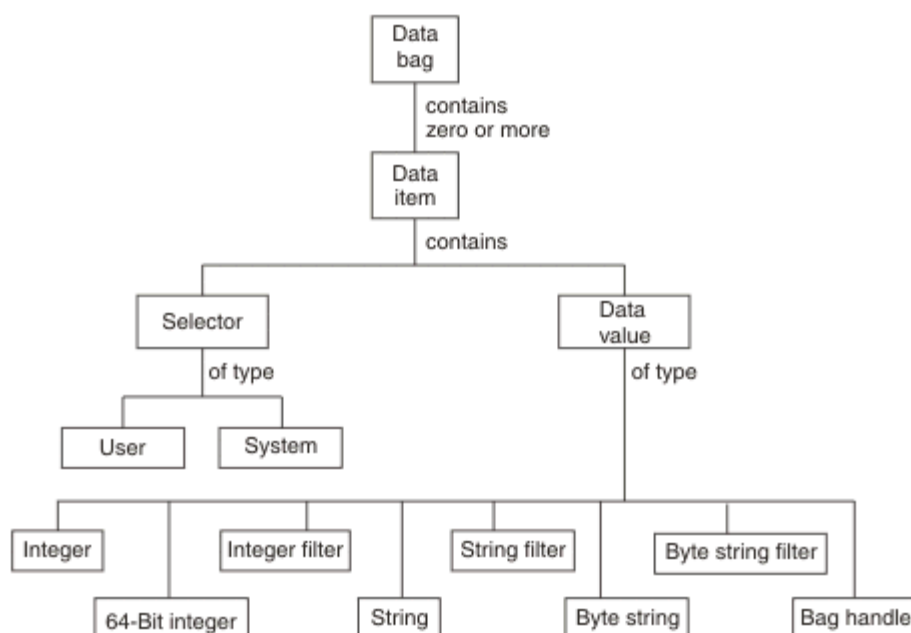
Worek danych jest środkiem do obsługi właściwości lub parametrów obiektów za pomocą interfejsu administracyjnego IBM MQ (MQAI).

Torby danych

- Worek danych zawiera zero lub więcej *elementów danych*. Te elementy danych są uporządkowane w worku, ponieważ są one umieszczane w torbie. Jest to nazywane *kolejką wstawiania*. Każdy element danych zawiera *selektor*, który identyfikuje element danych i *wartość* tego elementu danych, który może być liczbą całkowitą, 64-bitową liczbą całkowitą, filtrem całkowitoliczbowym, łańcuchem, filtrem łańcuchowym, łańcuchem bajtowym, filtrem łańcucha bajtowego lub uchwyttem innej torby. Elementy danych są opisane w szczegółach w sekcji [“Typy elementów danych dostępne w interfejsie MQAI”](#) na stronie 64

Istnieją dwa typy selektorów: *selektory użytkownika* i *selektory systemowe*. Są one opisane w sekcji [Selektory MQAI](#). Selektory są zwykle unikalne, ale możliwe jest, że istnieje wiele wartości dla tego samego selektora. W tym przypadku *indeks* identyfikuje konkretne wystąpienie selektora, który jest wymagany. Indeksy są opisane w sekcji [“Indeksowanie w interfejsie MQAI”](#) na stronie 37.

Hierarchia tych pojęć jest przedstawiona na [Rysunku 1](#).



Rysunek 5. Hierarchia pojęć związanych z MQAI

Hierarchia została wyjaśniona w poprzednim akapicie.

Typy worka danych

W zależności od zadania, które ma zostać wykonane, można wybrać typ torby danych, która ma zostać utworzona:

zbiór użytkownika

Prosta torba używana na potrzeby danych użytkownika.

zbiór administracyjny

Worek utworzony dla danych wykorzystywanych do administrowania obiektami produktu IBM MQ przez wysyłanie komunikatów administracyjnych do serwera komend. Torba administracyjna

automatycznie implikuje pewne opcje opisane w sekcji [“Tworzenie i usuwanie worków danych”](#) na stronie 63.

zbiór komend

W przypadku komend do administrowania obiektami produktu IBM MQ tworzona jest również torba. Jednak w przeciwieństwie do worka administracyjnego, worek komend nie oznacza automatycznie niektórych opcji, mimo że opcje te są dostępne. Więcej informacji na temat opcji zawiera sekcja [“Tworzenie i usuwanie worków danych”](#) na stronie 63.

worek grupowy

Worek używany do przechowywania zestawu zgrupowanych elementów danych. Worki grupowe nie mogą być używane do administrowania obiektami IBM MQ .

Dodatkowo **torba systemowa** jest tworzona przez interfejs MQAI, gdy komunikat odpowiedzi jest zwracany z serwera komend i umieszczony w worku wyjściowym użytkownika. Użytkownik nie może modyfikować torby systemowej.

W tym temacie wyszczególnia się różne sposoby używania toreb danych:

Korzystanie z worków danych

Różne sposoby korzystania z toreb danych są przedstawione na poniższej liście:

- Istnieje możliwość tworzenia i usuwania worków danych [“Tworzenie i usuwanie worków danych”](#) na stronie 63.
- Istnieje możliwość wysyłania danych między aplikacjami przy użyciu worków danych [“Umieszczanie i odbieranie worków danych przy użyciu interfejsu MQAI”](#) na stronie 64.
- Elementy danych można dodawać do toreb danych [“Dodawanie elementów danych do worków za pomocą interfejsu MQAI”](#) na stronie 65.
- Komendę uzyskiwania informacji można dodać w worku danych [“Dodawanie komendy uzyskiwania informacji do torby”](#) na stronie 66.
- Istnieje możliwość uzyskania informacji o danych w obrębie toreb danych [“Sprawdzanie w obrębie toreb danych”](#) na stronie 67.
- Istnieje możliwość zliczania elementów danych w obrębie worka danych [“Zliczanie elementów danych”](#) na stronie 70.
- Istnieje możliwość zmiany informacji w obszarze danych [“Zmiana informacji w torbie”](#) na stronie 67.
- Użytkownik może wyczyścić worek danych [“Kasowanie torby przy użyciu wywołania mqClearBag”](#) na stronie 68.
- Można obciąć worek danych [“Obcinanie torby przy użyciu wywołania funkcji mqTruncate”](#) na stronie 69.
- Istnieje możliwość przekształcenia worków i buforów [“Przekształcanie worków i buforów”](#) na stronie 69.

Tworzenie i usuwanie worków danych

Tworzenie worków danych

Aby korzystać z interfejsu MQAI, należy najpierw utworzyć worek danych przy użyciu wywołania mqCreate. Jako dane wejściowe dla tego wywołania należy podać jedną lub więcej opcji w celu sterowania tworzeniem torby.

Parametr **Options** wywołania MQCreateBag umożliwia wybranie, czy ma zostać utworzony worek użytkownika, worek komend, worek grupowy czy też worek administracyjny.

Aby utworzyć torbę użytkownika, worek komend lub worek grupowy, można wybrać jedną lub więcej dodatkowych opcji do:

- Użyj formularza listy, gdy w worku znajdują się dwa lub więcej sąsiadujących wystąpień tego samego selektora.

- Zmień kolejność elementów danych, ponieważ są one dodawane do komunikatu PCF, aby upewnić się, że parametry są w ich poprawnej kolejności. Więcej informacji na temat elementów danych zawiera sekcja [“Typy elementów danych dostępne w interfejsie MQAI”](#) na stronie 64.
- Sprawdź wartości selektorów użytkownika dla elementów, które są dodawane do torby.

Worki administracyjne automatycznie oznaczają te opcje.

Worek danych jest identyfikowany przez uchwyt. Uchwyt worka jest zwracany przez komendę `mqCreateBag` i musi być podany we wszystkich innych wywołaniach, które korzystają z tego worka danych.

Pełny opis wywołania funkcji `mqCreateBag` znajduje się w sekcji [mqCreateBag\(Worek mqCreate\)](#).

Usuwanie worków danych

Wszystkie elementy, które są tworzone przez użytkownika, również muszą zostać usunięte przy użyciu wywołania `mqDelete`. Na przykład, jeśli w kodzie użytkownika jest tworzona torba, należy ją również usunąć w kodzie użytkownika.

Torby systemowe są tworzone i usuwane automatycznie przez interfejs MQAI. Więcej informacji na ten temat zawiera sekcja [“Wysyłanie komend administracyjnych do serwera komend qm przy użyciu wywołania mqExecute”](#) na stronie 71. Kod użytkownika nie może usunąć torby systemowej.

Pełny opis wywołania funkcji `mqDeleteBag` znajduje się w sekcji [mqDeleteBag\(Wtorka mqDelete\)](#).

Umieszczanie i odbieranie worków danych przy użyciu interfejsu MQAI

Dane mogą być również wysyłane między aplikacjami przez wprowadzanie i pobieranie danych za pomocą wywołania `mqPutBag` i wywołania `mqGetBag`. Dzięki temu interfejs administracyjny produktu IBM MQ (MQAI) obsługuje bufor, a nie aplikację.

Wywołanie `mqPutBag` przekształca treść podanej torby w komunikat PCF i wysyła komunikat do określonej kolejki, a wywołanie funkcji `mqGetBag` usuwa komunikat z określonej kolejki i przekształca je z powrotem w worek danych. Dlatego wywołanie metody `mqPutBag` jest odpowiednikiem wywołania `mqBagToBuffer`, po którym następuje wywołanie `MQPUT`, a element `mqGetBag` jest odpowiednikiem wywołania `MQGET`, po którym następuje wywołanie metody `mqBufferToBag`.

Więcej informacji na temat wysyłania i odbierania komunikatów PCF w określonej kolejce można znaleźć w sekcji [“Wysyłanie i odbieranie komunikatów PCF w określonej kolejce”](#) na stronie 25

Uwaga: Jeśli zostanie wybrana opcja użycia wywołania `mqGetBag`, szczegóły PCF w komunikacie muszą być poprawne; jeśli nie są, odpowiednie wyniki błędów i komunikat PCF nie jest zwracany.

Typy elementów danych dostępne w interfejsie MQAI

Elementy danych są używane przez interfejs administracyjny produktu IBM MQ (MQAI) do zapewnienia worków danych podczas ich tworzenia. Tymi elementami danych mogą być elementy użytkownika lub systemowe.

Te elementy użytkownika zawierają dane użytkownika, takie jak atrybuty obiektów, które są administrowane. Elementy systemowe powinny być używane do większej kontroli nad wygenerowanymi komunikatami: na przykład generowanie nagłówków komunikatów. Więcej informacji na temat elementów systemowych zawiera sekcja [“Elementy systemowe i MQAI”](#) na stronie 65.

Typy elementów danych

Po utworzeniu worka danych można je wypełnić elementami całkowitymi lub łańcuchami znakowymi. Można zapytać o wszystkie trzy typy pozycji.

Element danych może być liczbą całkowitą lub pozycją łańcucha znaków. Poniżej przedstawiono typy elementów danych dostępne w obrębie interfejsu MQAI:

- Liczba całkowita

- 64-bitowa liczba całkowita
- Filtr liczb całkowitych
- Znakowy-łańcuch
- Filtr łańcuchowy
- Łańcuch bajtów
- Filtr łańcucha bajtowego
- Uchwyt worka

Korzystanie z elementów danych

Są to następujące sposoby korzystania z elementów danych:

- [“Zliczanie elementów danych” na stronie 70.](#)
- [“Usuwanie elementów danych” na stronie 70.](#)
- [“Dodawanie elementów danych do worków za pomocą interfejsu MQAI” na stronie 65.](#)
- [“Filtrowanie i odpytywanie elementów danych” na stronie 66.](#)

Elementy systemowe i MQAI

Elementy systemowe mogą być używane przez interfejs administracyjny produktu IBM MQ (MQAI) dla:

- Generowanie nagłówek PCF. Elementy systemu mogą sterować identyfikatorem komendy PCF, opcjami sterowania, numerem kolejnym komunikatu i typem komendy.
- Konwersja danych. Elementy systemu obsługują identyfikator zestawu znaków dla pozycji łańcucha znaków w torbie.

Podobnie jak wszystkie elementy danych, elementy systemowe składają się z selektora i wartości. Więcej informacji na temat tych selektorów i ich wartości zawiera sekcja [Selektory MQAI](#).

Elementy systemowe są unikalne. Jeden lub więcej elementów systemu może być identyfikowany przez selektor systemu. Istnieje tylko jedno wystąpienie każdego selektora systemu.

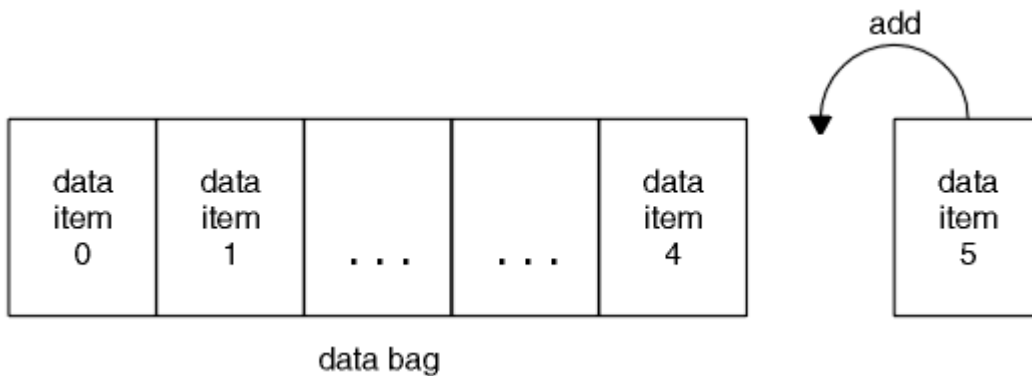
Większość elementów systemu może być modyfikowana (patrz [“Zmiana informacji w torbie” na stronie 67](#)), ale użytkownik nie może zmienić opcji tworzenia worka. Nie można usuwać elementów systemowych. (Patrz [“Usuwanie elementów danych” na stronie 70](#)).

Dodawanie elementów danych do worków za pomocą interfejsu MQAI

Gdy worek danych jest tworzony przy użyciu interfejsu administracyjnego IBM MQ (MQAI), można je zapęnić elementami danych. Tymi elementami danych mogą być elementy użytkownika lub systemowe.

Więcej informacji na temat elementów danych zawiera sekcja [“Typy elementów danych dostępne w interfejsie MQAI” na stronie 64](#).

Zmaterializowana tabela zapytania (MQAI) umożliwia dodawanie pozycji całkowitych, 64-bitowych elementów całkowitych, elementów filtru liczb całkowitych, elementów łańcucha znakowego, filtru łańcuchów, elementów łańcucha bajtowego i elementów filtru łańcucha bajtów do worków, co jest wyświetlane w składkach [Rysunek 6 na stronie 66](#). Elementy są identyfikowane przez selektor. Zwykle jeden selektor identyfikuje tylko jeden element, ale nie zawsze jest to przypadek. Jeśli element danych o określonym selektorze znajduje się już w torbie, dodatkowa instancja tego selektora jest dodawana do końca torby.



Rysunek 6. Dodawanie elementów danych

Dodaj elementy danych do torby przy użyciu wywołań mqAdd*:

- Aby dodać elementy całkowite, należy użyć wywołania mqAddInteger w sposób opisany w sekcji [mqAddInteger](#) .
- Aby dodać 64-bitowe elementy całkowite, należy użyć wywołania mqAddInteger64 zgodnie z opisem w sekcji [mqAddInteger64](#) .
- Aby dodać elementy filtru liczby całkowitej, należy użyć wywołania mqAddIntegerFilter zgodnie z opisem w sekcji [mqAddIntegerFilter](#) .
- Aby dodać elementy znakowe, należy użyć wywołania mqAddString zgodnie z opisem w sekcji [mqAddString](#) .
- Aby dodać elementy filtru łańcucha, należy użyć wywołania mqAddStringFilter zgodnie z opisem w sekcji [mqAddStringFilter](#) .
- Aby dodać elementy łańcucha bajtowego, należy użyć wywołania mqAddByteString zgodnie z opisem w sekcji [mqAddByteString](#) .
- Aby dodać elementy filtru łańcuchów bajtów, należy użyć wywołania filtru mqAddByteStringzgodnie z opisem w sekcji [mqAddByteStringFilter](#) .

Więcej informacji na temat dodawania elementów danych do torby zawiera sekcja [“Elementy systemowe i MQAI”](#) na stronie 65.

Dodawanie komendy uzyskiwania informacji do torby

Wywołanie zapytania mqAddsłuży do dodawania komendy uzyskiwania informacji do torby. Połączenie jest specjalnie przeznaczone do celów administracyjnych, dlatego może być używane tylko z workami administracyjnymi. Umożliwia on określenie selektorów atrybutów, dla których użytkownik chce uzyskać informacje z produktu IBM MQ.

Pełny opis wywołania uzyskiwania informacji o produkcie mqAddmożna znaleźć w sekcji [ZapytaniemqAdd](#).

Filtrowanie i odpytywanie elementów danych

Korzystając z interfejsu MQAI do uzyskiwania informacji o atrybutach obiektów produktu IBM MQ , można sterować danymi zwracaną do programu na dwa sposoby.

- Można **filtrować** danych zwracanych za pomocą wywołań mqAddInteger i mqAddString. To podejście umożliwia określenie pary *Selector* i *ItemValue* , na przykład:

```
mqAddInteger(inputbag, MQIA_Q_TYPE, MQQT_LOCAL)
```

Ten przykład określa, że typ kolejki (*Selector*) musi być lokalny (*ItemValue*), a ta specyfikacja musi być zgodna z atrybutami obiektu (w tym przypadku kolejka), o której użytkownik jest pytany.

Inne atrybuty, które mogą być filtrowane, odpowiadają komendom PCF Inquire *, które można znaleźć w programie [“Wprowadzenie do formatów komend programowalnych IBM MQ”](#) na stronie 22. Na

przykład, aby dowiedzieć się więcej o atrybutach kanału, należy zapoznać się z informacjami na temat komendy Inquire Channel w tej dokumentacji produktu. Parametry "Wymagane parametry" i "Opcjonalne" w komendzie Inquire Channel identyfikują selektory, których można użyć do filtrowania.

- Istnieje możliwość **zapytania** poszczególnych atrybutów obiektu za pomocą wywołania zapytania mqAdd. Określa selektor, w którym użytkownik jest zainteresowany. Jeśli selektor nie zostanie określony, zwrócone zostaną wszystkie atrybuty obiektu.

Poniżej znajduje się przykład filtrowania i wysyłania zapytań do atrybutów kolejki:

```
/* Request information about all queues */
mqAddString(adminbag, MQCA_Q_NAME, "*")

/* Filter attributes so that local queues only are returned */
mqAddInteger(adminbag, MQIA_Q_TYPE, MQQT_LOCAL)

/* Query the names and current depths of the local queues */
mqAddInquiry(adminbag, MQCA_Q_NAME)
mqAddInquiry(adminbag, MQIA_CURRENT_Q_DEPTH)

/* Send inquiry to the command server and wait for reply */
mqExecute(MQCMD_INQUIRE_Q, ...)
```

Sprawdzanie w obrębie toreb danych

Informacje na temat:

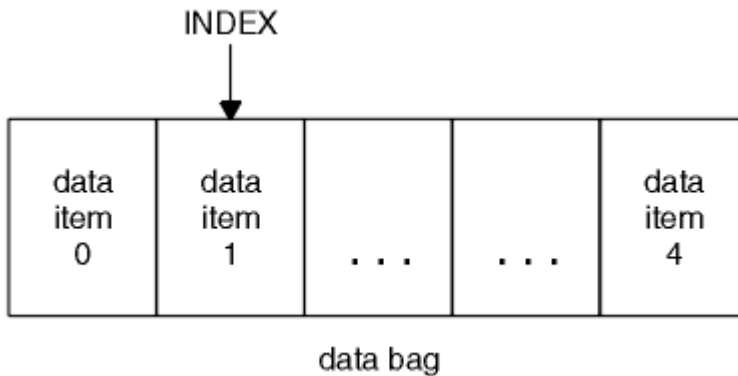
- Wartość elementu całkowitoliczbowego przy użyciu wywołania mqInquireInteger. Patrz [mqInquireInteger](#).
- Wartość 64-bitowego elementu całkowitoliczbowego przy użyciu wywołania mqInquireInteger64. Patrz [mqInquireInteger64](#).
- Wartość elementu filtru całkowitoliczbowego za pomocą wywołania mqInquireIntegerFilter. Patrz [mqInquireIntegerFilter](#).
- Wartość elementu łańcucha znakowego przy użyciu wywołania łańcucha mqInquire. Patrz [mqInquireString](#).
- Wartość elementu filtru łańcucha przy użyciu wywołania mqInquireStringFilter. Patrz [mqInquireStringFilter](#).
- Wartość elementu łańcucha bajtowego przy użyciu wywołania mqInquireByteString. Patrz [mqInquireByteString](#).
- Wartość elementu filtru łańcucha bajtowego przy użyciu wywołania filtru mqInquireByteString. Patrz [mqInquireByteStringFilter](#).
- Wartość uchwytu torby przy użyciu wywołania funkcji mqInquire. Patrz [mqInquireBag](#).

Można również zapytać o typ (liczba całkowita, 64-bitowa liczba całkowita, filtr liczb całkowitych, łańcuch znaków, filtr łańcuchowy, łańcuch bajtowy, filtr łańcucha bajtowego lub uchwyt worka) konkretnego elementu przy użyciu wywołania mqInquireItemInfo. Patrz [mqInquireItemInfo](#).

Zmiana informacji w torbie

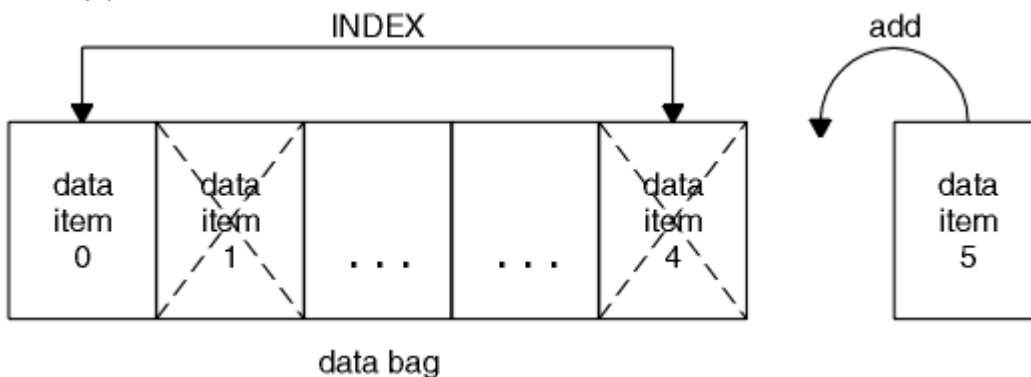
Interfejs MQAI umożliwia zmianę informacji w obrębie torby przy użyciu wywołań mqSet*. Można wykonać następujące czynności:

1. Modyfikowanie elementów danych w torbie. Indeks umożliwia zastąpienie pojedynczej instancji parametru identyfikacją wystąpienia elementu, który ma zostać zmodyfikowany (patrz [Rysunek 7 na stronie 68](#)).



Rysunek 7. Modyfikowanie pojedynczego elementu danych

2. Usuń wszystkie istniejące wystąpienia określonego selektora i dodaj nowe wystąpienie do końca torby. (Patrz Rysunek 8 na stronie 68.) Specjalna wartość indeksu umożliwia zastępowanie **wszystkich** instancji parametru.



Rysunek 8. Modyfikowanie wszystkich elementów danych

Uwaga: Indeks zachowuje kolejność wstawiania w worku, ale może wpływać na indeksy innych elementów danych.

Wywołanie `mqSetLiczba całkowita` umożliwia modyfikowanie elementów liczby całkowitej w obrębie torby. Wywołanie `mqSetInteger64` pozwala na zmodyfikowanie 64-bitowych elementów całkowitych. Wywołanie `mqSetIntegerFilter` umożliwia modyfikowanie elementów filtra liczby całkowitej. Wywołanie łańcucha `mqSet` umożliwia modyfikowanie elementów łańcucha znaków. Wywołanie metody `mqSetStringFilter` umożliwia modyfikowanie elementów filtra łańcuchów. Wywołanie metody `mqSetByteString` umożliwia modyfikowanie elementów łańcucha bajtowego. Wywołanie filtra `mqSetByteString` umożliwia modyfikowanie elementów filtra łańcuchów bajtów. Można również użyć tych wywołań, aby usunąć wszystkie istniejące wystąpienia określonego selektora i dodać nowe wystąpienie na końcu tej torby. Element danych może być elementem użytkownika lub elementem systemowym.

Pełny opis tych wywołań znajduje się w:

- [mqSetLiczba całkowita](#)
- [mqSetInteger64](#)
- [mqSetIntegerFilter](#)
- [mqSetłańcuch](#)
- [mqSetStringFilter](#)
- [mqSetByteString](#)
- [Filtr mqSetByteString](#)

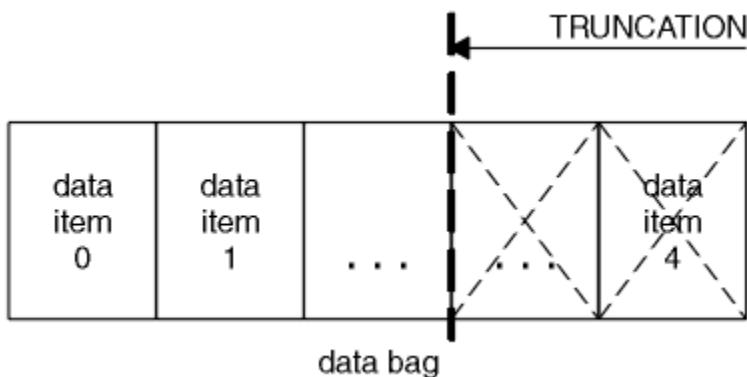
Kasowanie torby przy użyciu wywołania `mqClearBag`

Wywołanie funkcji mqClearBag usuwa wszystkie elementy użytkownika z worka użytkownika i resetuje elementy systemowe do ich wartości początkowych. Worki systemowe zawarte w worku są również usuwane.

Pełny opis wywołania funkcji mqClearBag znajduje się w sekcji [mqClearBag](#)(Torba mqClear).

Obcinanie torby przy użyciu wywołania funkcji mqTruncate

Wywołanie funkcji mqTruncate umożliwia zmniejszenie liczby elementów użytkownika w torbie użytkownika przez usunięcie elementów z końca torby, zaczynając od ostatnio dodanego elementu. Na przykład można użyć tego samego nagłówka w celu wygenerowania więcej niż jednego komunikatu.

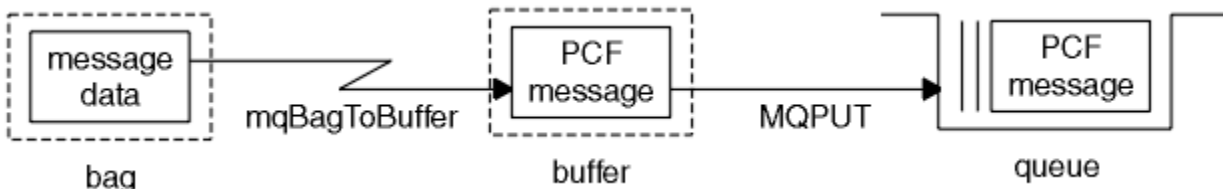


Rysunek 9. Obcinanie torby

Pełny opis wywołania funkcji mqTruncateBag znajduje się w sekcji [mqTruncateBag](#).

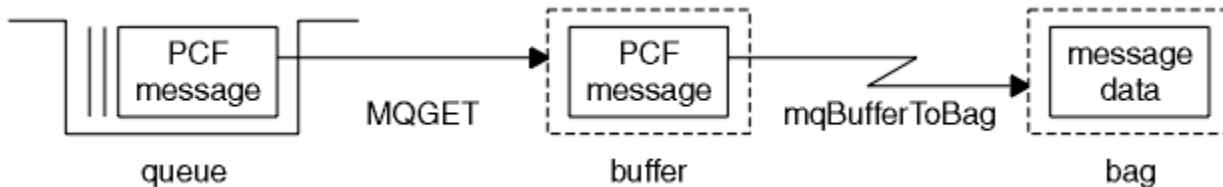
Przekształcanie worków i buforów

Aby wysłać dane między aplikacjami, po pierwsze dane komunikatu są umieszczane w torbie. Następnie dane znajdujące się w torbie są przekształcane w komunikat PCF za pomocą wywołania mqBagToBuffer. Komunikat PCF jest wysyłany do wymaganej kolejki przy użyciu wywołania MQPUT. Pokazano to na rysunku Rysunek 10 na stronie 69. Pełny opis wywołania mqBagToBuffer można znaleźć w sekcji [mqBagToBuffer](#).



Rysunek 10. Przekształcanie torby w komunikaty PCF

Aby odbierać dane, komunikat jest odbierany do buforu przy użyciu wywołania MQGET. Dane w buforze są następnie przekształcane w torbę za pomocą wywołania mqBufferToBag, udostępniając bufor, który zawiera poprawny komunikat PCF. Pokazano to na rysunku Rysunek 11 na stronie 69. Pełny opis wywołania mqBufferToBag można znaleźć w sekcji [mqBufferToBag](#).



Rysunek 11. Przekształcanie komunikatów PCF w formularz torby

Zliczanie elementów danych

Wywołanie `mqCountPozycje` zlicza liczbę elementów użytkownika, elementów systemowych lub obu tych elementów, które są przechowywane w torbie danych, i zwraca ten numer. Na przykład, `mqCountItems(Bag, 7, ...)`, zwraca liczbę elementów w torbie z selektorem 7. Może on zliczać pozycje według indywidualnego selektora, selektory użytkownika, selektory systemowe lub wszystkie selektory.

Uwaga: To wywołanie zlicza liczbę elementów danych, a nie liczbę unikalnych selektorów w torbie. Selektor może występować wiele razy, więc w worku może być mniej unikalnych selektorów niż elementy danych.

Pełny opis wywołania `mqCountItems` zawiera sekcja [mqCountItems](#).

Usuwanie elementów danych

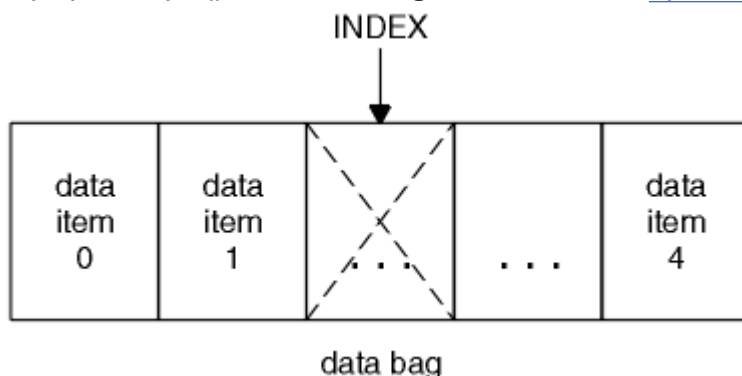
Elementy z worków można usuwać na wiele sposobów. Można wykonać następujące czynności:

- Usuń jeden lub więcej elementów użytkownika z torby. Więcej informacji można znaleźć w sekcji [“Usuwanie elementów danych z torby przy użyciu wywołania elementu `mqDelete`”](#) na stronie 70.
- Usuń **wszystkie** elementy użytkownika z torby, to znaczy *Wyczyść* torbę. Szczegółowe informacje na ten temat zawiera sekcja [“Kasowanie torby przy użyciu wywołania `mqClearBag`”](#) na stronie 68.
- Usuń elementy użytkownika z końca torby, to znaczy *obetnij* worek. Więcej informacji można znaleźć w sekcji [“Obcinanie torby przy użyciu wywołania funkcji `mqTruncate`”](#) na stronie 69.

Usuwanie elementów danych z torby przy użyciu wywołania elementu `mqDelete`

Wywołanie elementu `mqDelete` powoduje usunięcie jednego lub większej liczby elementów użytkownika z torby. Indeks jest używany do usuwania:

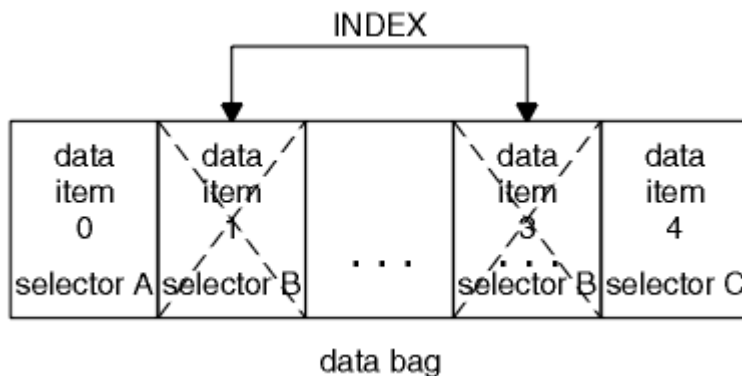
1. Pojedyncze wystąpienie określonego selektora. (Patrz [Rysunek 12](#) na stronie 70.)



Rysunek 12. Usuwanie pojedynczego elementu danych

lub wersji

2. Wszystkie wystąpienia określonego selektora. (Patrz [Rysunek 13](#) na stronie 71.)



Rysunek 13. Usuwanie wszystkich elementów danych

Uwaga: Indeks zachowuje kolejność wstawiania w worku, ale może wpływać na indeksy innych elementów danych. Na przykład wywołanie elementu `mqDelete` zachowuje wartości indeksu dla elementów danych, które są zgodne z usuniętym elementem, ponieważ indeksy są reorganizowane w celu wypełnienia luki, która pozostaje z usuniętego elementu.

Pełny opis wywołania elementu `mqDelete` można znaleźć w sekcji [mqDeleteItem](#) (Element `mqDelete`).

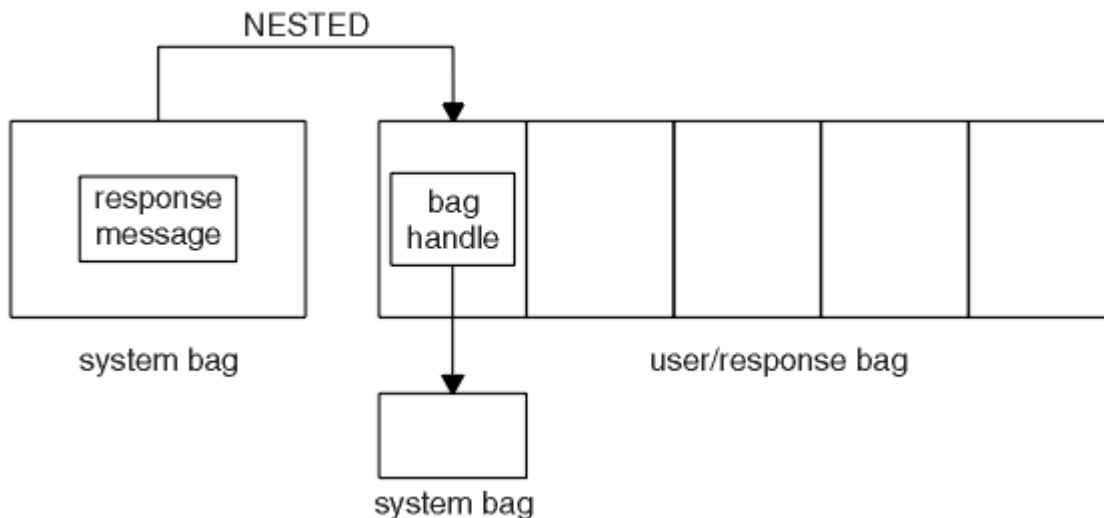
Wysyłanie komend administracyjnych do serwera komend `qm` przy użyciu wywołania `mqExecute`

Po utworzeniu i wypełnieniu worka danych do serwera komend menedżera kolejek za pomocą wywołania `mqExecute` może zostać wysłany komunikat komendy administracyjnej. Ta opcja obsługuje wymianę z serwerem komend i zwraca odpowiedzi w torbie.

Po utworzeniu i wypełnieniu worka danych można wysłać komunikat komendy administracyjnej do serwera komend menedżera kolejek. Najprostszym sposobem wykonania tego działania jest użycie wywołania `mqExecute`. Wywołanie funkcji `mqExecute` wysyła komunikat komendy administracyjnej jako komunikat nietrwały i oczekuje na wszystkie odpowiedzi. Odpowiedzi są zwracane w worku odpowiedzi. Mogą one zawierać informacje o atrybutach związanych z kilkoma obiektami IBM MQ lub seriami komunikatów odpowiedzi błędów PCF, na przykład. W związku z tym worek odpowiedzi może zawierać tylko kod powrotu lub może zawierać *worki zagnieżdżone*.

Komunikaty odpowiedzi są umieszczane w workach systemowych, które są tworzone przez system. Na przykład w przypadku zapytań dotyczących nazw obiektów tworzona jest torba systemowa, w której przechowywane są te nazwy obiektów, a torbka jest wstawiana do worka użytkownika. Uchwyty do tych worków są następnie wstawiane do worka odpowiedzi, a do zagnieżdżonej torby można uzyskać dostęp za pomocą selektora `MQHA_BAG_HANDLE`. Torba systemowa pozostaje w pamięci masowej, jeśli nie została usunięta, dopóki nie zostanie usunięta torba odpowiedzi.

Pojęcie *zagnieżdżanie* jest wyświetlane w produkcie [Rysunek 14 na stronie 72](#).



Rysunek 14. Zagnieżdżenie

Jako dane wejściowe dla wywołania `mqExecute` należy podać:

- Uchwyt połączenia MQI.
- Komenda do wykonania. Powinna to być jedna z wartości `MQCMD_*`.

Uwaga: Jeśli ta wartość nie jest rozpoznawana przez interfejs MQAI, wartość jest nadal akceptowana. Jeśli jednak do wstawiania wartości do worka użyto wywołania `mqAddInquiry`, ten parametr musi być komendą `INQUIRE` rozpoznawaną przez interfejs MQAI. Oznacza to, że parametr powinien mieć postać `MQCMD_INQUIRE_*`.

- Opcjonalnie uchwyt worka zawierający opcje sterujące przetwarzaniem wywołania. W tym miejscu można również określić maksymalny czas (w milisekundach), przez który program MQAI powinien czekać na każdy komunikat odpowiedzi.
- Uchwyt worka administracyjnego, który zawiera szczegóły komendy administracyjnej, która ma zostać wydana.
- Uchwyt worka odpowiedzi, który odbiera komunikaty odpowiedzi.

Następujące uchwyty są opcjonalne:

- Uchwyt obiektu kolejki, w której ma zostać umieszczona komenda administracyjna.

Jeśli nie zostanie podany żaden uchwyt obiektu, komenda administracyjna zostanie umieszczona w systemie `SYSTEM.ADMIN.COMMAND.QUEUE` należącej do aktualnie połączonego menedżera kolejek. Jest to opcja domyślna.

- Uchwyt obiektu kolejki, w którym mają być umieszczane komunikaty odpowiedzi.

Komunikaty odpowiedzi można umieścić w kolejce dynamicznej, która jest tworzona automatycznie przez interfejs MQAI. Utworzona kolejka istnieje tylko przez czas trwania wywołania i jest usuwana przez interfejs MQAI przy wyjściu z wywołania `mqExecute`.

Przykłady użycia wywołania komendy `mqExecute` można znaleźć w sekcji [Przykładowy kod](#).

V 9.0.1 Administrowanie przy użyciu REST API

Za pomocą programu administracyjnego REST API można administrować obiektami produktu IBM MQ, takimi jak menedżery kolejek i kolejki **V 9.0.5**, oraz usługi Managed File Transfer GET usługi REST. Informacje są wysyłane do administracyjnego REST API i odbierane z niego w formacie JSON. Te interfejsy API zgodne ze specyfikacją REST mogą pomóc w osadzonym administrowaniu produktem IBM MQ w popularnych narzędziach DevOps i automatyzacji.

Zanim rozpocznesz

Informacje uzupełniające na temat dostępnych zasobów REST znajdują się w sekcji [Skorowidz produktu administrative REST API](#).

Uwaga:

W produkcie IBM MQ 9.0.1 produkt administrative REST API nie jest zintegrowany z zabezpieczeniami produktu IBM MQ. Oznacza to, że produkt administrative REST API jest domyślnie wyłączony. Należy ręcznie włączyć administrative REST API, aby można było z niego korzystać. Więcej informacji na temat włączania składnika administrative REST API zawiera sekcja [“Pierwsze kroki w produkcie administrative REST API”](#) na stronie 73.

V 9.0.2 W produkcie IBM MQ 9.0.2 administrative REST API jest zintegrowany z zabezpieczeniami produktu IBM MQ. Opcja administrative REST API jest domyślnie włączona. Należy jednak skonfigurować zabezpieczenia przed użyciem produktu administrative REST API. Więcej informacji na ten temat zawiera sekcja [Bezpieczeństwo konsoli IBM MQ i REST API](#).

Procedura

- [“Pierwsze kroki w produkcie administrative REST API”](#) na stronie 73
- [“Korzystanie z administrative REST API”](#) na stronie 79
- [“Zdalne administrowanie przy użyciu REST API”](#) na stronie 85
- [“REST API datowniki”](#) na stronie 89
- [“REST API obsługa błędów”](#) na stronie 89
- [“REST API Wykrywanie”](#) na stronie 92
- [“REST API Obsługa języków narodowych”](#) na stronie 93

V 9.0.1 Pierwsze kroki w produkcie administrative REST API

Przed uruchomieniem produktu administrative REST API należy zainstalować odpowiednie komponenty, włączyć opcję REST API, skonfigurować zabezpieczenia i uruchomić serwer mqweb.

Zanim rozpocznesz

IBM i W systemie IBM i komendy powinny być uruchomione w systemie QSHHELL.

O tym zadaniu

Procedura dotycząca tego zadania koncentruje się na szybkim rozpoczęciu pracy z administrative REST API. Kroki konfigurowania zabezpieczeń opisują sposób konfigurowania podstawowego rejestru użytkowników, ale istnieją inne opcje dotyczące konfigurowania użytkowników i ról. Więcej informacji na temat konfigurowania zabezpieczeń dla produktu administrative REST API zawiera sekcja [IBM MQ Console and REST API security](#) (Zabezpieczenia konsoli i konsoli produktu IBM MQ).

Uwaga: Aby uzyskać dostęp do pliku mqwebuser.xml, należy być [użytkownikiem uprzywilejowanym](#).

Procedura

1. Zainstaluj komponent IBM MQ Console i REST API :

- **V 9.0.4** **AIX** W systemie AIX zainstaluj zestaw plików mqm.web.rte.
- **Linux** W systemie Linux zainstaluj komponent MQSeriesWeb. Więcej informacji na temat instalowania komponentów i składników w systemie Linux można znaleźć w sekcji [Zadania instalacyjne produktu Linux](#).

- **Windows** W systemie Windowszainstaluj składnik Web Administration .Więcej informacji na temat instalowania komponentów i składników w systemie Windowsmożna znaleźć w sekcji [Zadania instalacyjne produktuWindows](#).
 - **z/OS** W systemie z/OSzainstaluj składnik IBM MQ for z/OS Unix System Services Web Components . Więcej informacji na temat instalowania komponentów i składników w systemie z/OSmożna znaleźć w sekcji [Zadania instalacyjne produktuz/OS](#).
2. Opcjonalne: W produkcie IBM MQ 9.0.2domyślnie włączona jest opcja administrative REST API . However, in IBM MQ 9.0.1, the administrative REST API is disabled by default. Jeśli używany jest produkt IBM MQ 9.0.1, włącz REST API , dodając następujący kod XML do pliku mqwebuser .xml , w znacznikach <server> :

```
<variable name="mqRestAutostart" value="true"/>
```

mqwebuser .xml można znaleźć w jednym z następujących katalogów:

- **ULW** `MQ_DATA_DIRECTORY/web/installations/installationName/servers/mqweb`
- **z/OS** `WLP_user_directory/servers/mqweb`

gdzie `WLP_user_directory` jest katalogiem, który został określony podczas tworzenia skryptu produktu `crtmqweb.sh` w celu utworzenia definicji serwera WWW mqweb.

3. **V 9.0.2**

W produkcie IBM MQ 9.0.2zabezpieczenia są włączone dla administrative REST API. Przed użyciem produktu administrative REST API należy skonfigurować użytkowników i role:

- a) Skopiuj plik `basic_registry.xml` z katalogu `MQ_INSTALLATION_PATH/web/mq/samp/configuration` .
- b) Umieść przykładowy plik XML w odpowiednim katalogu:

- **ULW** W systemie UNIX, Linux, and Windows: `MQ_DATA_DIRECTORY/web/installations/installationName/servers/mqweb`
- **z/OS** W systemie z/OS: `WLP_user_directory/servers/mqweb`

gdzie `katalog_użytkownika_WLP_użytkownika` jest katalogiem, który został określony podczas uruchamiania skryptu `crtmqweb.sh` w celu utworzenia definicji serwera WWW mqweb.

- c) Zmień nazwę przykładowego pliku XML na `mqwebuser.xml`.

Uwaga: Ten plik o zmienionej nazwie zastępuje istniejący plik, który jest również używany dla IBM MQ Console. Dlatego, jeśli zmieniono plik `mqwebuser.xml` dla IBM MQ Console, należy skopiować zmiany do nowego pliku XML przed zmianą nazwy.

- d) Opcjonalne: Aby dodać użytkowników i grupy, należy zmodyfikować plik `mqwebuser.xml` . Przypisz tych użytkowników i grupy odpowiednie role, aby były autoryzowane do korzystania z produktu administrative REST API. Można również zmienić hasła dla użytkowników, którzy są domyślnie zdefiniowani, i zakodować nowe hasła. Więcej informacji na ten temat zawiera sekcja [Konfigurowanie użytkowników i ról](#).

4. Włącz zdalne połączenia z serwerem mqweb:

- Z wersji 9.0.4użyj komendy **setmqweb** :
- ```
setmqweb properties -k httpHost -v hostname
```
- W przypadku wersji 9.0.1dodaj następujący kod XML do pliku mqwebuser .xml w znacznikach <server> :

```
<variable name="httpHost" value="hostname"/>
```

gdzie `nazwa_hosta` określa adres IP, nazwę hosta serwera nazw domen (DNS) z przyrostkiem nazwy domeny lub nazwę DNS hosta serwera, na którym zainstalowany jest produkt IBM MQ . Aby określić wszystkie dostępne interfejsy sieciowe, należy użyć gwiazdki (\*).



**Ostrzeżenie:** z/OS V 9.0.4

Przed wydaniem komendy **setmqweb** lub **dspmqweb** w systemie z/OS należy ustawić zmienną środowiskową `WLP_USER_DIR` tak, aby wskazywała konfigurację serwera mqweb.

W tym celu należy wydać następującą komendę:

```
export WLP_USER_DIR=WLP_user_directory
```

gdzie `WLP_user_directory` to nazwa katalogu, który jest przekazywany do produktu `crtmqweb.sh`. Na przykład:

```
export WLP_USER_DIR=/var/mqm/web/installation1
```

Więcej informacji na ten temat zawiera sekcja [Tworzenie definicji serwera Liberty](#).

#### 5. Uruchom serwer mqweb, który obsługuje REST API:

- **Windows Linux** W systemach Windows i Linux jako użytkownik uprzywilejowany wprowadź w wierszu komend następującą komendę:  
`stmqweb`
- **z/OS** W systemie z/OS uruchom procedurę utworzoną w polu Czynność 29: Tworzenie procedury dla serwera IBM WLP.

## Co dalej

1. **V 9.0.2** Wybierz sposób uwierzytelniania użytkowników produktu administrative REST API na serwerze mqweb. Nie ma potrzeby używania tej samej metody dla wszystkich użytkowników. Dostępne opcje opisano w następujących tematach:
  - Pozwól użytkownikom na uwierzytelnianie za pomocą podstawowego uwierzytelniania HTTP. W takim przypadku nazwa użytkownika i hasło są kodowane, ale nie są szyfrowane i wysyłane wraz z każdym żądaniem REST API do uwierzytelniania i autoryzowania użytkownika dla tego żądania. Aby to uwierzytelnianie było bezpieczne, należy użyć bezpiecznego połączenia. Oznacza to, że należy używać protokołu HTTPS. Więcej informacji zawiera temat [Korzystanie z podstawowego uwierzytelniania HTTP przy użyciu interfejsu REST API](#).
  - Pozwól użytkownikom na uwierzytelnianie za pomocą uwierzytelniania za pomocą tokenu. W tym przypadku użytkownik udostępnia ID użytkownika i hasło do zasobu REST API `login` za pomocą metody HTTP POST. Generowany jest token LTPA, który umożliwia użytkownikowi pozostawienie się i autoryzowanie przez określony czas. Aby to uwierzytelnianie było bezpieczne, należy użyć bezpiecznego połączenia. Oznacza to, że należy używać protokołu HTTPS. Więcej informacji na ten temat zawiera sekcja [Korzystanie z uwierzytelniania opartego na tokenie za pomocą REST API](#).
  - Umożliwia użytkownikom uwierzytelnianie za pomocą certyfikatów klienta. W takim przypadku użytkownik nie korzysta z identyfikatora użytkownika ani hasła w celu zalogowania się do administrative REST API, ale zamiast niego korzysta z certyfikatu klienta. Więcej informacji na ten temat zawiera sekcja [Korzystanie z uwierzytelniania przy użyciu certyfikatu klienta przy użyciu produktu REST API](#).
2. **V 9.0.1** Konfigurowanie ustawień produktu REST API, w tym włączanie połączeń HTTP, a także zmiana numeru portu. Więcej informacji na ten temat zawiera sekcja [Konfigurowanie konsoli IBM MQ Console i produktu REST API](#).
3. **V 9.0.5** Opcjonalnie skonfiguruj REST API dla produktu MFT. Więcej informacji na ten temat zawiera sekcja ["Wymagana konfiguracja dla MFT REST API" na stronie 76](#).
4. **V 9.0.2** Opcjonalnie skonfiguruj opcję Cross Origin Resource Sharing dla REST API. Domyślnie nie można uzyskać dostępu do REST API z zasobów WWW, które nie są udostępniane w tej samej domenie, co REST API. Oznacza to, że żądania o różnym pochodzeniu nie są włączone. Istnieje

możliwość skonfigurowania funkcji CORS (Cross Origin Resource Sharing) w taki sposób, aby zezwalać na żądania między pochodzeniem z określonych adresów URL. Więcej informacji na ten temat zawiera sekcja [Konfigurowanie mechanizmu CORS dla produktu REST API](#).

5. Należy używać komponentu REST API. Aby uzyskać więcej informacji, patrz [“Korzystanie z administrative REST API”](#) na stronie 79, [“Zdalne administrowanie przy użyciu REST API”](#) na stronie 85 i [Administracyjne odwołanie do REST API](#).

**Uwaga:** Serwer mqweb można zatrzymać w dowolnym momencie za pomocą komendy **endmqweb**. Jeśli jednak serwer mqweb nie jest uruchomiony, nie można użyć REST API ani IBM MQ Console.

## V 9.0.5 Wymagana konfiguracja dla MFT REST API

Kroki, które należy wykonać, aby skonfigurować produkt Managed File Transfer do korzystania z produktu REST API.

W pliku `mqwebuser.xml` ustaw:

- Właściwość **mqRestMftEnabled** musi mieć wartość *true*

**Uwaga:** Jeśli wartość tej właściwości zostanie zmieniona, należy zrestartować serwer mqweb.

- Właściwość **mqRestMftCoordinationQmgr** do odpowiedniego menedżera kolejek koordynacji produktu MFT działającego lokalnie na komputerze, na którym działa serwer mqweb.

Aby menedżer kolejek działał jako menedżer kolejek koordynacji produktu MFT dla serwera REST API, należy uruchomić następujące czynności:

- Komenda **fteSetupCoordination** dla tego samego menedżera kolejek lokalnych, który został ustawiony dla właściwości **mqRestMftCoordinationQmgr** w pliku `mqwebuser.xml`.

Ta komenda generuje plik MQSC, który zawiera definicje obiektów IBM MQ. Należy pamiętać, że komenda może być uruchomiona na dowolnym komputerze, na którym zainstalowano produkt MFT.

- Plik MQSC wygenerowany w poprzednim kroku powinien zostać uruchomiony dla menedżera kolejek koordynacji w celu utworzenia wymaganych obiektów produktu IBM MQ.

### Informacje pokrewne

[Konfigurowanie zabezpieczeń produktu MFTREST API](#)

[Administrowanie MFT REST API](#)

[GET-lista transferów](#)

[GET-status transferu](#)

## V 9.0.1 Określanie adresu URL produktu REST API

Domyślny adres URL w celu uzyskania dostępu do produktu administrative REST API w systemie IBM MQ 9.0.4 i jego nowszych wersjach to: `https://localhost:9443/ibmmq/rest/v1/admin`. W systemie IBM MQ 9.0.3 i wcześniejszym domyślnym adresem URL jest: `https://localhost:9443/ibmmq/rest/v1`. Domyślnym adresem URL do uzyskania dostępu do produktu messaging REST API jest: `https://localhost:9443/ibmmq/rest/v1/messaging`. Jeśli host lub port zostaną zmienione z wartości domyślnej lub jeśli włączony jest protokół HTTP, można określić adres URL za pomocą komendy **dspmqweb**.

### O tym zadaniu

V 9.0.4 Za pomocą komendy IBM MQ 9.0.4 można użyć komendy **dspmqweb status** w celu określenia adresu URL produktu REST API w systemach Windows, Linux i z/OS.

W przypadku produktu IBM MQ 9.0.3 i wcześniejszych wersji produktu Windows i Linux można użyć komendy **dspmqweb**. W systemie z/OS należy przeszukać plik `messages.log`, aby określić adres URL.

## Procedura

### V 9.0.4

Aby określić adres URL, należy użyć jednej z następujących metod:

- W produkcie IBM MQ 9.0.4 jako użytkownik uprzywilejowany należy użyć komendy **dspmweb status**:
  1. Upewnij się, że serwer mqweb jest uruchomiony, wprowadzając komendę **stzmqweb** w wierszu komend.
  2. Wyświetl adres URL, wprowadzając komendę **dspmweb status** w wierszu komend.
- W przypadku produktu IBM MQ 9.0.3 i wcześniejszych wersji należy użyć jednej z następujących metod:
  - **Windows** / **Linux** W systemie Windows lub Linux należy użyć komendy **dspmweb** jako użytkownika uprzywilejowanego:
    1. Upewnij się, że serwer mqweb jest uruchomiony, wprowadzając komendę **stzmqweb** w wierszu komend.
    2. Wyświetl adres URL, wprowadzając komendę **dspmweb** w wierszu komend.
  - **z/OS** W systemie z/OS użyj pliku `messages.log`, aby znaleźć adres URL:
    1. Otwórz plik `messages.log`.

Plik `messages.log` znajduje się w ścieżce `WLP_user_directory/servers/mqweb/logs`, gdzie `WLP_user_directory` jest katalogiem, który został określony podczas uruchamiania skryptu **crtmqweb.sh** w celu utworzenia definicji serwera WWW mqweb.
    2. Wyszukaj najnowszy kod komunikatu CWWKT0016I, który kończy się na `ibmmq/rest/v1`. Adres URL jest zawarty w tym komunikacie.

## Przykład

W poniższym przykładzie przedstawiono kod komunikatu CWWKT0016I z adresem URL jako domyślny adres URL:

```
A CWWKT0016I: Web application available (default_host): https://localhost:9443/ibmmq/rest/v1
```

### z/OS V 9.0.2 Sprawdzenie, czy serwer mqweb jest poprawnie skonfigurowany w systemie z/OS

Kroki, które należy wykonać, aby sprawdzić, czy serwer mqweb jest poprawnie skonfigurowany w systemie z/OS, oraz aby naprawić typowe problemy związane z konfiguracją.

## Procedura

1. Sprawdź, czy uruchomione zadanie serwera mqweb zostało uruchomione.  
Komunikaty protokołu zadania powinny być podobne do następujących:

```
+CWWKE0001I: The server mqweb has been launched.
+CWWKZ0001I: Application com.ibm.mq.rest started in 1.493 seconds.
+CWWKZ0001I: Application com.ibm.mq.console started in 0.885 seconds.
+CWWKF0011I: The server mqweb is ready to run a smarter planet.
```

W STDERR nie powinno być żadnych komunikatów

W STDOUT powinny znajdować się komunikaty podobne do następujących:

```
Launching mqweb (WebSphere Application Server 17.0.0.2, WAS FOR Z/OS 17.0.0.2,
```

```

MQM MVS/ESA V9 R0.4/wlp-1.0.17.c1170220170523-1818) on IBM J9 VM,
version pmz6480sr4fp7-20170627_02 (SR4 FP7) (en_US)
[AUDIT] CWWKE0001I: The server mqweb has been
launched.
[AUDIT] CWWKG0028A: Processing included configuration resource: /mqm/V9R0M4/web/mq/etc/
mqweb.xml
[AUDIT] CWWKG0028A: Processing included configuration resource:
var/mqm/mqweb904/servers/mqweb/mqwebuser.xml
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport/api/docs/
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport/api/explorer/
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport/ibm/api/
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport/ibm/api/explorer/
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport/ibm/api/docs/subscription/websocket/
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport:9080/ibmmq/rest/v1/
[AUDIT] CWWKZ0001I: Application com.ibm.mq.rest started in 1.493 seconds.
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport/ibmmq/console/
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport/ibmmq/console/internal/
[AUDIT] CWWKZ0001I: Application com.ibm.mq.console started in 1.459 seconds.
[AUDIT] CWWKF0012I: The server installed the following features:
[jsp-2.2, servlet-3.1, ssl-1.0, jndi-1.0, basicAuthenticationMQ-1.0,
apiDiscovery-1.0, localConnector-1.0, appSecurity-2.0, jaxrs-1.1,
concurrent-1.0, json-1.0, websocket-1.0, distributedMap-1.0,
applicationMonitorMQ-1.0].
[AUDIT] CWWKF0011I: The server mqweb is ready to run a smarter planet.
[AUDIT] MQWB2019I: MQ Console level: 9.0.4 - V904-GA904-L171016
[AUDIT] MQWB0023I: MQ REST API level: 9.0.4 - V904-GA904-L171016

```

### Uwagi:

- Jeśli uruchomienie procedury nie powiedzie się, rozwiąż problemy.
- Jeśli komunikaty są podobne do następujących:

```

Web application available (default_host):
http://localhost:portnumber/ibmmq/console/

```

Nie ustawiono właściwości **httpHost** , aby zezwolić na połączenia zdalne z serwerem mqweb .

```
<variable name="httpHost" value="*" />
```

- Jeśli została zmieniona właściwość **httpHost** , zostanie wyświetlony komunikat podobny do następującego:

```

Web application available (default_host):
yourhost:portnumber/ibmmq/console/

```

W zależności od konfiguracji TCP/IP może nie być możliwe użycie adresu wyświetlanego jako adres URL.

- Zmiany w plikach konfiguracyjnych serwera XML są pobierane w kilka sekund później. Nie ma potrzeby restartowania serwera mqweb .
- Połącz się z IBM MQ Console.
 

Użytkownik powinien wyświetlić okno z tytułem IBM MQ Console.

### Uwagi:

- Jeśli zostanie wyświetlone okno z programem This site can't be reached lub Context Root Not Found, IBM MQ Console nie jest aktywne. Należy poczekać.
- Jeśli nie zostanie wyświetlone okno z tytułem IBM MQ Console, mogą istnieć dodatkowe informacje diagnostyczne w programie WLP\_USER\_DIRECTORY/servers/mqweb/logs/messages.log, gdzie WLP\_USER\_DIRECTORY jest katalogiem podanym przez użytkownika, podczas uruchamiania skryptu **crtmqweb.sh** w celu utworzenia definicji serwera mqweb .

Należy zauważyć, że ten plik znajduje się w kodzie ASCII i aby wyświetlić ten plik, można użyć komendy **odit** z wiersza komend USS lub użyć opcji ISPF 3.17 , a następnie użyć komendy wiersza komend **va** (view ASCII).

c. Komunikaty w STDOUT mogą być podobne do następujących:

```
[WARNING] SRVE0190E: File not found: /nls/en_GB/labels.json
[WARNING] SRVE0190E: File not found: /nls/en_GB/errors.json
[WARNING] SRVE0190E: File not found: /nls/en_GB/strings.json
[WARNING] SRVE0190E: File not found: /nls/en_GB/
pcf.json
```

Komunikaty te można zignorować.

d. Jeśli menedżery kolejek są zdefiniowane w systemie, w którym działa serwer mqweb , i działają na tym samym poziomie co serwer mqweb , te menedżery kolejek będą wyświetlane w widżecie **Lokalne menedżery kolejek** w MQ Console.

Jeśli nie są wyświetlane żadne menedżery kolejek, nie ma żadnych menedżerów kolejek na tym samym poziomie, co serwer mqweb , które zostały uruchomione od ostatniego IPL.

3. Jeśli pojawi się komunikat:

```
Lost communication with the server Could not establish communication with
the server.
```

a) W procedurze użytej do uruchomienia serwera mqweb :

- i) Sprawdź, czy biblioteki STEPLIB są na poprawnym poziomie i czy są autoryzowane przez APF.
- ii) Sprawdź ścieżkę PATH i LIBPATH, aby uzyskać poprawną ścieżkę. Zmienna PATH i LIBPATH są zdefiniowane w ramach procedury uruchomionej zadania serwera WWW, która została oparta na przykładowej ścieżce CSQ4WEBS.

b) W USS należy użyć komendy `ls -Eltr PathPrefix/web/bin/dspmq`, gdzie PathPrefix jest ścieżką instalacyjną IBM MQ Unix System Services Components.

Powinno być wyświetlane dane wyjściowe podobne do `-rwxr-xr-t a-s- ... /mqm/V9R0M4/web/bin/dspmq`

Sprawdź, czy ustawione są opcje t i a.

Użyj komendy:

- `chmod +t PathPrefix/web/bin/dspmq`, aby ustawić bit sticky (t)
- `extattr +a PathPrefix/web/bin/dspmq`, aby ustawić atrybut autoryzowany APF (a)

## V 9.0.1 Korzystanie z administrative REST API

Podczas korzystania z produktu administrative REST API można wywoływać metody HTTP w adresach URL, które reprezentują różne obiekty produktu IBM MQ , takie jak menedżery kolejek lub kolejki. Metoda HTTP, na przykład POST, reprezentuje typ działania, które ma zostać wykonane dla obiektu, który jest reprezentowany przez adres URL. Dalsze informacje na temat działania mogą zostać udostępnione w formacie JSON jako część ładunku metody HTTP lub zakodowane w parametrach zapytania. Informacje na temat wyniku wykonania działania mogą zostać zwrócone jako treść odpowiedzi HTTP.

### Zanim rozpoczniesz

Przed użyciem produktu administrative REST API należy rozważyć następujące kwestie:

- **V 9.0.2** Aby można było korzystać z produktu administrative REST API, należy uwierzytelnić się na serwerze mqweb. Uwierzytelnianie można uwierzytelnić za pomocą podstawowego uwierzytelniania HTTP, uwierzytelniania certyfikatu klienta lub uwierzytelniania opartego na tokenie. Więcej informacji na temat korzystania z tych metod uwierzytelniania zawiera sekcja [Zabezpieczenia konsoli IBM MQ i zabezpieczenia produktu REST API](#).
- W programie REST API jest rozróżniana wielkość liter. Na przykład żądanie HTTP GET dla następującego adresu URL nie wyświetla informacji, jeśli menedżer kolejek ma nazwę `qmgr1`.

```
/ibmmq/rest/v1/admin/qmgr/QMGR1
```

- Nie wszystkie znaki, które mogą być używane w nazwach obiektów IBM MQ, mogą być bezpośrednio zakodowane w adresie URL. Aby zakodować te znaki poprawnie, należy użyć odpowiedniego kodowania adresu URL:
  - Ukośnik (/) musi być zakodowany jako %2F.
  - Znak procentu,%, musi być zakodowany jako %25.
- Ze względu na zachowanie niektórych przeglądarek nie należy używać nazw obiektów przy użyciu tylko kropki lub znaków ukośnika.

## O tym zadaniu

Jeśli do wykonania działania na obiekcie używany jest produkt REST API, należy najpierw utworzyć adres URL reprezentujący ten obiekt. Każdy adres URL rozpoczyna się od przedrostka, który opisuje nazwę hosta i port, do którego ma zostać wysłane żądanie. Pozostała część adresu URL opisuje konkretny obiekt lub zestaw obiektów, znany jako zasób.

Działanie, które ma zostać wykonane na zasobie, określa, czy adres URL wymaga parametrów zapytania. Definiuje ona także metodę HTTP, która jest używana, oraz określa, czy do adresu URL są wysyłane dodatkowe informacje, czy też są one zwracane z niego w formularzu JSON. Dodatkowe informacje mogą stanowić część żądania HTTP lub być zwracane jako część odpowiedzi HTTP.

Po zbudowaniu adresu URL i utworzeniu opcjonalnego ładunku JSON na potrzeby wysyłania w żądaniu HTTP można wysłać żądanie HTTP do produktu IBM MQ. Żądanie można wysłać, korzystając z implementacji HTTP wbudowanej w wybrany język programowania. Żądania można również wysłać za pomocą narzędzi wiersza komend, takich jak cURL, przeglądarki WWW lub dodania przeglądarki WWW.

**Ważne:** Należy wykonać co najmniej kroki [“1.a” na stronie 80](#) i [“1.b” na stronie 80](#).

## Procedura

### 1. Utwórz adres URL:

#### a) Zaczynij od następującego przedrostka adresu URL:

**V 9.0.4** IBM MQ 9.0.4 i nowsze:

```
https://host:port/ibmmq/rest/v1/admin
```

IBM MQ 9.0.3 i wcześniejsze:

```
https://host:port/ibmmq/rest/v1
```

#### host

Określa nazwę hosta lub adres IP, na którym jest dostępny administrative REST API.

Wartością domyślną jest localhost.

#### Port

Określa numer portu HTTPS używany przez produkt administrative REST API.

Wartością domyślną jest 9443.

**V 9.0.1** Jeśli połączenia HTTP zostaną włączone, można użyć protokołu HTTP zamiast protokołu HTTPS. Więcej informacji na temat włączania protokołu HTTP zawiera sekcja [Konfigurowanie portów HTTP i HTTPS](#).

Więcej informacji na temat określania przedrostka adresu URL zawiera sekcja [“Określanie adresu URL produktu REST API” na stronie 76](#).

#### b) Dodaj zasób do ścieżki adresu URL.

Dostępne są następujące zasoby:



- instalacja
- QMGR
- **V 9.0.2** kolejka
- **V 9.0.4** subskrypcja
- **V 9.0.4** kanał

Na przykład w celu interakcji z menedżerami kolejek należy dodać łańcuch /qmgr do przedrostka adresu URL w celu utworzenia następującego adresu URL:

**V 9.0.4** IBM MQ 9.0.4 i nowsze:

```
https://localhost:9443/ibmmq/rest/v1/admin/qmgr
```

IBM MQ 9.0.3 i wcześniejsze:

```
https://localhost:9443/ibmmq/rest/v1/qmgr
```

c) Opcjonalne: Dodaj do adresu URL dodatkowe opcjonalne segmenty ścieżki.

W informacjach dodatkowych dla każdego typu obiektu segmenty opcjonalne można zidentyfikować w adresie URL za pomocą nawiasów klamrowych, które otaczają go { }.

Na przykład dodaj nazwę menedżera kolejek QM1 do adresu URL, aby utworzyć następujący adres URL:

**V 9.0.4** IBM MQ 9.0.4 i nowsze:

```
https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1
```

IBM MQ 9.0.3 i wcześniejsze:

```
https://localhost:9443/ibmmq/rest/v1/qmgr/QM1
```

d) Opcjonalne: Dodaj opcjonalny parametr zapytania do adresu URL.

Czy dodać znak zapytania?, nazwa zmiennej, znak równości =, a wartość lub lista wartości do adresu URL.

Na przykład, aby zażądać wszystkich atrybutów menedżera kolejek QM1, należy utworzyć następujący adres URL:

**V 9.0.4** IBM MQ 9.0.4 i nowsze:

```
https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1?attributes=*
```

IBM MQ 9.0.3 i wcześniejsze:

```
https://localhost:9443/ibmmq/rest/v1/qmgr/QM1?attributes=*
```

e) Dodaj kolejne opcjonalne parametry zapytania do adresu URL.

Dodaj znak ampersand &, do adresu URL, a następnie powtórz [krok d.](#)

2. Wywołaj odpowiednią metodę HTTP w adresie URL. Określ dowolny opcjonalny ładunek JSONi podaj odpowiednie referencje zabezpieczeń do uwierzytelnienia. Na przykład:

- Użyj implementacji HTTP/REST wybranego języka programowania.
- Należy użyć narzędzia, takiego jak dodatek przeglądarki klienta REST lub cURL.

## **V 9.0.5** Uzyskiwanie listy MFT transferów lub statusu przesyłania

Jak uzyskać listę Managed File Transfer transferów za pomocą REST API

Więcej informacji na ten temat zawiera sekcja [Administrowanie za pomocą konsoli REST API](#).

Aby uzyskać listę przelewów:

1. Wyślij żądanie [GET](#) do serwera, na którym znajduje się aplikacja WWW MFT iMFT . Podczas wysyłania żądania należy wziąć pod uwagę format adresu URL.

Więcej informacji na ten temat zawiera sekcja [admin/mft/transfer](#) .

2. Po odebraniu żądania, na żądanie jest wykonywane sprawdzenie, czy żądanie jest poprawne.

Więcej informacji na ten temat zawiera sekcja [Konfigurowanie zabezpieczeń produktu MFT REST API](#) .

3. Jeśli żądanie jest poprawne, odpowiedź zostanie zakończona pomyślnie wraz z treścią odpowiedzi. W przeciwnym razie zostanie wygenerowany odpowiedni kod błędu i kod odpowiedzi.

Listę odpowiedzi można znaleźć w sekcji [Kody statusu odpowiedzi](#) .

Aby uzyskać status przeniesienia:

1. Wyślij żądanie [GET](#) do serwera, na którym znajduje się aplikacja WWW MFT iMFT . Podczas wysyłania żądania należy wziąć pod uwagę format adresu URL.

Więcej informacji na ten temat zawiera sekcja [admin/mft/transfer](#) .

2. Po odebraniu żądania, na żądanie jest wykonywane sprawdzenie, czy żądanie jest poprawne.

Więcej informacji na ten temat zawiera sekcja [Konfigurowanie zabezpieczeń produktu MFT REST API](#) .

3. Jeśli żądanie jest poprawne, odpowiedź zostanie zakończona pomyślnie wraz z treścią odpowiedzi. W przeciwnym razie zostanie wygenerowany odpowiedni kod błędu i kod odpowiedzi.

Listę odpowiedzi można znaleźć w sekcji [Kody statusu odpowiedzi](#) .

## Przykładowy treść odpowiedzi

Lista przesyłania jest pobierana z wewnętrznej pamięci masowej, a treść odpowiedzi jest generowana w formacie JSON dla każdego z przesunięć. Treść odpowiedzi dla operacji przesyłania wygląda następująco:

```
{
 "transfer": [
 {
 "destinationAgent": {
 "name": "AGENT.TRI.BANK"
 },
 "originator": {
 "host": "192.168.99.1",
 "userId": "johndoe"
 },
 "sourceAgent": {
 "name": "TESTAGENT"
 },
 "statistics": {
 "endTime": "2018-01-08T16:22:15.569Z",
 "numberOfFileFailures": 0,
 "numberOfFileSuccesses": 2,
 "numberOfFileWarnings": 0,
 "numberOfFiles": 2,
 "startTime": "2018-01-08T16:22:15.242Z"
 },
 "status": {
 "state": "successful"
 },
 "id": "414D51204D465444454D4F3320202020513E525A21109908"
 },
 {
 "destinationAgent": {
 "name": "AGENT.TRI.BANK"
 },
 "originator": {
 "host": "192.168.99.1",
 "userId": "ramsubbarao"
 },
 "sourceAgent": {
 "name": "TESTAGENT"
 },
 "statistics": {
 "endTime": "2018-01-08T16:22:13.573Z",
 "numberOfFileFailures": 0,
 "numberOfFileSuccesses": 2,
 "numberOfFileWarnings": 0,
 "numberOfFiles": 2,
 "startTime": "2018-01-08T16:22:13.167Z"
 }
 }
]
}
```

```

 },
 "status": {
 "state": "successful"
 },
 "id": "414D51204D465444454D4F3320202020513E525A21109702"
 }
}

```

Więcej informacji na temat atrybutów można znaleźć w sekcji [Atrybuty treści odpowiedzi JSON produktu REST API](#).

## V 9.0.5 Uzyskiwanie statusu agenta MFT

Sposób uzyskiwania statusu agenta Managed File Transfer przy użyciu REST API

Więcej informacji na ten temat zawiera sekcja [Administrowanie za pomocą konsoli REST API](#).

Aby uzyskać status agenta, wykonaj następujące czynności:

1. Wyślij żądanie [GET](#) do agenta w domyślnym menedżerze kolejek koordynacji, który korzysta z aplikacji WWW MFT iMFT. Podczas wysyłania żądania należy wziąć pod uwagę format adresu URL.

Więcej informacji na ten temat zawiera sekcja [admin/mft/agent](#).

2. Po odebraniu żądania, na żądanie jest wykonywane sprawdzenie, czy żądanie jest poprawne.

Więcej informacji na ten temat zawiera sekcja [Konfigurowanie zabezpieczeń produktu MFT REST API](#).

3. Jeśli żądanie jest poprawne, odpowiedź zostanie zakończona pomyślnie wraz z treścią odpowiedzi. W przeciwnym razie zostanie wygenerowany odpowiedni kod błędu i kod odpowiedzi.

Listę odpowiedzi można znaleźć w sekcji [Kody statusu odpowiedzi](#).

## Przykładowe adresy URL

### **{baseURI}/ibmmq/rest/{version}/admin/mft/agent**

Wyświetla listę serwerów **name**, **state** i **type** wszystkich agentów znajdujących się w menedżerze kolejek koordynacji.

Więcej informacji na temat tych atrybutów zawiera sekcja [Status agenta REST API - atrybuty](#).

Jeśli wartość `?attributes=*` zostanie dodana na końcu łańcucha adresu URL, dane wyjściowe będą zawierać listę wszystkich atrybutów dla każdego agenta, który znajduje się w menedżerze kolejek koordynacji.

Jeśli wartość `/ {agentName}` zostanie dodana na końcu łańcucha adresu URL, dane wyjściowe będą zawierać **name**, **state** i **type** agenta, który jest zgodny z `agentName`.

Analogicznie, jeśli do końca łańcucha adresu URL zostanie dodany łańcuch `/ {agentName}? attributes=*` , dane wyjściowe będą zawierać listę wszystkich atrybutów agenta, które są zgodne z `agentName`.

### **{baseURI}/ibmmq/rest/{version}/admin/mft/agent?name={<prefix>\*<suffix> }**

Na przykład `{baseURI}/ibmmq/rest/{version}/admin/mft/agent?name=AGENT*TEST` wyświetla domyślne podsumowanie atrybutów dla agentów rozpoczynających się od nazwy `AGENT` i kończące się nazwą `TEST`.

Jeśli wartość `*TEST` zostanie pominięta z adresu URL, dane wyjściowe będą zawierać podsumowanie domyślnego atrybutu dla wszystkich agentów, których nazwa rozpoczyna się od nazwy `AGENT`.

### **{baseURI}/ibmmq/rest/{version}/admin/mft/agent?type={agentType}**

Na przykład `{baseURI}/ibmmq/rest/{version}/admin/mft/agent?type=standard` wyświetla domyślne podsumowanie atrybutów dla agentów typu `standard`.

Podobnie, jeśli zostanie zastąpiony `type={agentType} (state={agentState})`, dane wyjściowe będą wyświetlane jako domyślne podsumowanie atrybutów dla agentów określonego stanu.

**{baseURI}/ibmmq/rest/{version}/admin/mft/agent?type={agentType}&attributes=\***

Na przykład {baseURI}/ibmmq/rest/{version}/admin/mft/agent?

type=standard&attributes=\* generuje wszystkie atrybuty dla agenta, którego typem jest *standard*.

Możliwe jest również filtrowanie według kombinacji atrybutów. Na przykład:

- {baseURI}/ibmmq/rest/{version}/admin/mft/agent?name=\*bob&type=standard&state=ready&attributes=\*
- {baseURI}/ibmmq/rest/{version}/admin/mft/agent?type=standard&state=ready&attributes=\*
- {baseURI}/ibmmq/rest/{version}/admin/mft/agent?name=agent\*&state=ready&attributes=\*

## Przykładowy treść odpowiedzi

Więcej informacji na temat opisanych atrybutów można znaleźć w sekcji [Status agenta REST API - atrybuty i Status odpowiedzi agenta REST API atrybuty- atrybuty](#).

Wydając następującą komendę /ibmmq/rest/v1/admin/mft/agent/, wyświetla podstawowy status agenta o nazwie, typie i stanie agenta. Na przykład:

```
{
 "agent": [{ "name": "AGENT1",
 "state": "ready",
 "type": "standard" },
 { "name": "AGENT2",
 "state": "ready",
 "type": "standard" },
 { "name": "BRIDGE_AGENT3",
 "type": "protocolBridge",
 "state": "ready" },
 { "name": "CD_AGENT",
 "type": "connectDirectBridge",
 "state": "ready" }]
}
```

Wydawanie każdej z następujących komend:

```
/ibmmq/rest/v1/admin/mft/agent?type=standard
/ibmmq/rest/v1/admin/mft/agent?state=stopped
/ibmmq/rest/v1/admin/mft/agent?name=AGENT*
```

zawiera podstawowe informacje o agencie, które są zgodne z konkretną nazwą, stanem i typem w każdej komendzie. Na przykład:

```
{ "agent": [{ "name": "AGENT1",
 "state": "ready",
 "type": "standard" },
 { "name": "AGENT2",
 "state": "ready",
 "type": "standard" }]
}
{ "agent": [{ "name": "AGENT1",
 "state": "stopped",
 "type": "standard" },
 { "name": "AGENT2",
 "state": "stopped",
 "type": "standard" }]
}
{ "agent": [{ "name": "AGENT1",
 "state": "ready",
 "type": "standard" },
 { "name": "AGENT2",
 "state": "ready",
 "type": "standard" }]
}
```

Wydając następującą komendę, /ibmq/rest/v1/admin/mft/agent? attributes=general&type=standard wyświetla wszystkie komunikaty o typie *standard* wraz z atrybutami general . Na przykład:

```
{
 "agent": [
 {
 "name": "AGENT1",
 "state": "ready",
 "type": "standard",
 "general": {
 "description": "Standard connected to the qmgr in client mode",
 "statusAge": "06:31:00",
 "version": "9.0.3.0",
 "level": "p903-L170513",
 "statusPublicationRate": 300,
 "statusPublishTime": "2017-10-31T06:57:07.000Z",
 "maximumQueuedTransfers": 1000,
 "maximumDestinationTransfers": 25,
 "maximumSourceTransfers": 25,
 "operatingSystem": "Windows7"
 }
 },
 {
 "name": "AGENT2",
 "state": "ready",
 "type": "standard",
 "general": {
 "description": "Standard connected to qmgr in Binding mode",
 "statusAge": "05:00:00",
 "version": "9.0.3.0",
 "level": "p903-L170513",
 "statusPublicationRate": 300,
 "statusPublishTime": "2017-09-13T09:10:09.000Z",
 "maximumQueuedTransfers": 1000,
 "maximumDestinationTransfers": 25,
 "maximumSourceTransfers": 25,
 "operatingSystem": "Windows7"
 }
 }
]
}
```

## V 9.0.4 Zdalne administrowanie przy użyciu REST API

Za pomocą programu REST API można administrować zdalnymi menedżerami kolejek oraz obiektami produktu IBM MQ, które są powiązane z tymi menedżerami kolejek. Ta zdalna administracja obejmuje menedżery kolejek, które znajdują się w tym samym systemie, ale nie w tej samej instalacji produktu IBM MQ, co serwer mqweb. Z tego względu produkt REST API może być używany do administrowania całą siecią produktu IBM MQ za pomocą tylko jednej instalacji, która uruchamia serwer mqweb. Aby administrować zdalnymi menedżerami kolejek, należy skonfigurować bramę produktu administrative REST API w taki sposób, aby co najmniej jeden menedżer kolejek w tej samej instalacji, co serwer WWW mqweb, był menedżerem kolejek bramy. Następnie można określić zdalny menedżer kolejek w adresie URL zasobu REST API, aby wykonać określone działanie administracyjne.

### Zanim rozpoczniesz

Aby zapobiec zdalnemu administrowaniu, należy wyłączyć bramę administrative REST API. Więcej informacji na ten temat zawiera sekcja [Konfigurowanie bramy produktu administrative REST API](#).

Aby można było używać bramy administrative REST API, muszą być spełnione następujące warunki:

- Serwer mqweb musi być skonfigurowany i uruchomiony. Więcej informacji na temat konfigurowania i uruchamiania serwera mqweb zawiera sekcja ["Pierwsze kroki w produkcie administrative REST API" na stronie 73](#).
- Menedżer kolejek, który ma zostać skonfigurowany jako menedżer kolejek bramy, musi znajdować się w tej samej instalacji, co serwer mqweb.
- Menedżer kolejek zdalnych, który ma być administrowany, musi być w wersji IBM MQ 8.0 lub nowszej.
- Należy upewnić się, że wszystkie atrybuty określone w żądaniu są poprawne dla systemu, do którego wysyłane jest żądanie. Na przykład, jeśli menedżer kolejek bramy znajduje się w systemie Windows, a zdalny menedżer kolejek znajduje się w systemie z/OS, nie można zażądać, aby atrybut `dataCollection.statistics` został zwrócony dla żądania HTTP GET w zasobie `queue`.

- Należy upewnić się, że wszystkie atrybuty określone w żądaniu są poprawne dla poziomu IBM MQ , do którego wysyłane jest żądanie. Na przykład, jeśli zdalny menedżer kolejek jest uruchomiony IBM MQ 8.0, nie można zażądać, aby atrybut `extended.enableMediaImageOperations` został zwrócony dla żądania HTTP GET w zasobie `queue`.
- Należy użyć jednego z następujących obsługiwanych zasobów REST:
  - `/queue`
  - `/subscription`
  - `V 9.0.5 /channel`
  - `V 9.0.5 /mqsc`
  - `/qmgr`

Zasób `/qmgr` zwraca tylko podzbiór atrybutów podczas wysyłania zapytań do menedżera kolejek zdalnych: `name`, `status.started`, `status.channelInitiatorState`, `status.ldapConnectionState`, `status.connectionCount` i `status.publishSubscribeState`.

## O tym zadaniu

Aby można było używać bramy administrative REST API do administrowania zdalnymi menedżerami kolejek, należy przygotować menedżery kolejek do zdalnego administrowania. Oznacza to, że konieczne jest skonfigurowanie kolejek transmisji, programów nasłuchujących oraz kanałów nadawczych i odbiorczych między menedżerem kolejek bramy a zdalnym menedżerem kolejek. Następnie można wysłać żądanie REST do zdalnego menedżera kolejek, określając menedżer kolejek w adresie URL zasobu. Menedżer kolejek bramy jest określany przez ustawienie atrybutu `mqRestGatewayQmgr` za pomocą komendy `setmqweb` na nazwę menedżera kolejek gatewaya lub wysłanie nazwy menedżera kolejek bramy w nagłówku, który jest wysyłany z żądaniem. Żądanie jest wysyłane za pośrednictwem menedżera kolejek bramy do zdalnego menedżera kolejek. Odpowiedź jest zwracana z nagłówkiem, który wskazuje menedżer kolejek, który był używany jako menedżer kolejek bramy.

## Procedura

1. Skonfiguruj komunikację między menedżerem kolejek bramy a menedżerami kolejek zdalnych, które mają być administrować. Te kroki konfiguracji są tymi samymi krokami, które są wymagane do skonfigurowania zdalnego administrowania za pomocą komendy `runmqsc` i PCF.
 

Więcej informacji na temat tych kroków zawiera sekcja [“Zdalne administrowanie z lokalnego menedżera kolejek”](#) na stronie 198.
2. Skonfiguruj zabezpieczenia na zdalnych menedżerach kolejek:
  - a) Upewnij się, że odpowiednie identyfikatory użytkowników istnieją w systemie, na którym działa zdalny menedżer kolejek. Identyfikator użytkownika, który musi istnieć w systemie zdalnym, zależy od roli użytkownika produktu REST API :
    - Jeśli użytkownik produktu REST API znajduje się w grupie `MQWebAdmin` lub `MQWebAdminRO`, to identyfikator użytkownika, który uruchomił serwer `mqweb`, musi istnieć w systemie zdalnym. Na serwerze IBM MQ Appliance użytkownik, który uruchamia serwer `mqweb`, ma wartość `mqsystem`.
    - Jeśli użytkownik produktu REST API znajduje się w grupie `MQWebUser`, ten identyfikator użytkownika produktu REST API musi istnieć w systemie zdalnym.
  - b) Upewnij się, że dla odpowiednich identyfikatorów użytkowników nadano niezbędne uprawnienia dostępu do odpowiednich zasobów produktu REST API w zdalnym menedżerze kolejek:
    - Uprawnienie do umieszczania komunikatów w `SYSTEM.ADMIN.COMMAND.QUEUE`.
    - Uprawnienie do umieszczania komunikatów w `SYSTEM.REST.REPLY.QUEUE`.
    - Uprawnienie dostępu do kolejek transmisyjnych zdefiniowanych dla zdalnego administrowania.
    - Uprawnienie do wyświetlania atrybutów menedżera kolejek.

- Uprawnienie do wykonywania żądań REST. Więcej informacji na ten temat można znaleźć w sekcji Wymagania dotyczące zabezpieczeń w tematach dotyczących zasobów produktu REST API.
3. Skonfiguruj, który menedżer kolejek lokalnych jest używany jako brama. Użytkownik może skonfigurować domyślny menedżer kolejek bramy, określić menedżer kolejek bramy w nagłówku HTTP lub użyć kombinacji obu metod:

- Skonfiguruj domyślny menedżer kolejek bramy za pomocą komendy **setmqweb** :

```
setmqweb properties -k mqRestGatewayQmgr -v qmgrName
```

gdzie *qmgrName* jest nazwą menedżera kolejek bramy.

Ten menedżer kolejek bramy jest używany, gdy spełnione są oba poniższe instrukcje:

- Menedżer kolejek nie jest określony w nagłówku `ibm-mq-rest-gateway-qmgr` żądania REST.
  - Menedżer kolejek określony w adresie URL zasobu REST API nie jest lokalnym menedżerem kolejek.
- Skonfiguruj menedżer kolejek bramy dla każdego żądania REST, ustawiając nagłówek HTTP `ibm-mq-rest-gateway-qmgr` na nazwę menedżera kolejek bramy.
4. Podaj nazwę zdalnego menedżera kolejek, który ma być administrowany w adresie URL zasobu. Na przykład, aby uzyskać listę kolejek ze zdalnego menedżera kolejek `remoteQM`, należy użyć następującego adresu URL:

```
https://localhost:9443/ibmmq/rest/v1/admin/qmgr/remoteQM/queue
```

## Wyniki

Nagłówek `ibm-mq-rest-gateway-qmgr` jest zwracany z odpowiedzią REST. Ten nagłówek określa, który menedżer kolejek był używany jako menedżer kolejek bramy.

## Przykład

W poniższym przykładzie istnieją trzy instalacje IBM MQ na dwóch komputerach. W systemie Machine 1 znajduje się Installation 1 i Installation 2. W systemie Machine 2 znajduje się Installation 3. Serwer mqweb jest skonfigurowany dla produktu Installation 1. W każdej instalacji jest pojedynczy menedżer kolejek, a te menedżery kolejek są skonfigurowane do zdalnego administrowania. Oznacza to, że następujące obiekty nasłuchiwanie, kanały i kolejki są skonfigurowane i uruchomione:

- W menedżerze kolejek QM1, w programie Installation 1, w systemie Machine 1:
  - Kanał nadawczy QM1.to.QM2
  - Kanał odbiorczy QM2.to.QM1
  - Kanał nadawczy QM1.to.QM3
  - Kanał odbiorczy QM3.to.QM1
  - Kolejka transmisji QM2
  - Kolejka transmisji QM3
  - Nasłuchiwanie skonfigurowane na porcie 1414
- W menedżerze kolejek QM2, w programie Installation 2, w systemie Machine 1:
  - Kanał nadawczy QM2.to.QM1
  - Kanał odbiorczy QM1.to.QM2
  - Kolejka transmisji QM1
  - Nasłuchiwanie skonfigurowane na porcie 1415
- W menedżerze kolejek QM3, w programie Installation 3, w systemie Machine 2:
  - Kanał nadawczy QM3.to.QM1

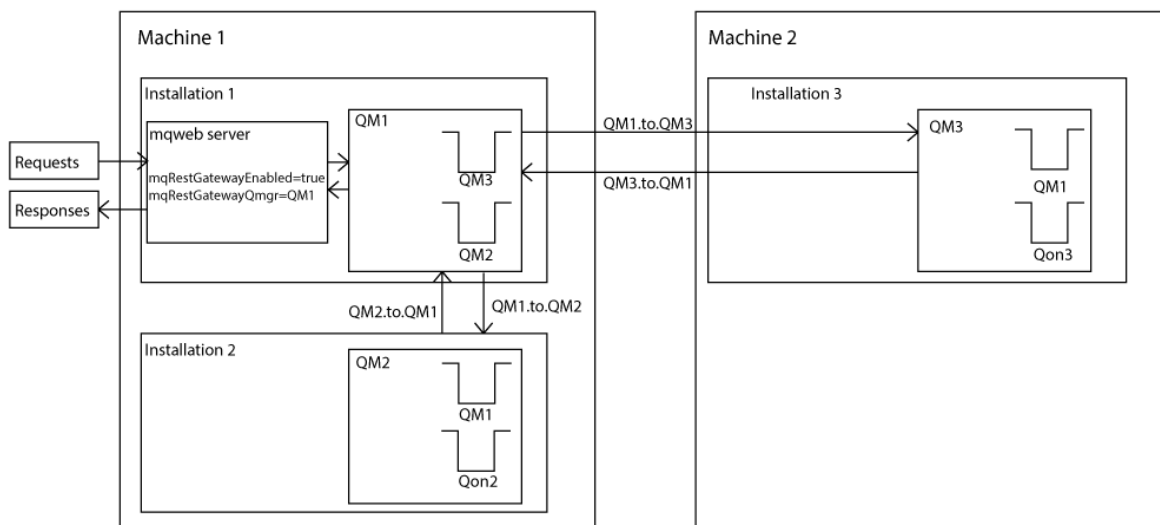
- Kanał odbiorczy QM1.to.QM3
- Kolejka transmisji QM1
- Domyślny program nasłuchujący

Kolejka Qon2 jest zdefiniowana w systemie QM2, a kolejka Qon3 jest zdefiniowana w systemie QM3.

Użytkownik mquser jest zdefiniowany na obu komputerach, ma przypisaną rolę MQWebAdmin w REST APIi ma nadane uprawnienia do uzyskiwania dostępu do odpowiednich kolejek w każdym menedżerze kolejek.

Komenda setmqweb służy do konfigurowania menedżera kolejek QM1 jako domyślnego menedżera kolejek bramy.

Poniższy diagram przedstawia tę konfigurację:



Rysunek 15. Diagram przykładowej konfiguracji zdalnego administrowania za pomocą konsoli REST API.

Na serwer mqweb wysyłane są następujące żądania REST:

```
GET https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM2/queue?
attributes=general.isTransmissionQueue
```

Otrzymano następującą odpowiedź:

```
{
 "queue" :
 [{
 "general": {
 "isTransmissionQueue": true
 },
 "name": "QM1",
 "type": "local"
 },
 {
 "general": {
 "isTransmissionQueue": false
 },
 "name": "Qon2",
 "type": "local"
 }
]
}
```

Na serwer mqweb wysyłane są następujące żądania REST:



```
GET https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM3/queue?
attributes=general.isTransmissionQueue,general.description
```

Otrzymano następującą odpowiedź:

```
{
 "queue" :
 [{
 "general": {
 "isTransmissionQueue": true,
 "description": "Transmission queue for remote admin."
 },
 "name": "QM1",
 "type": "local"
 },
 {
 "general": {
 "isTransmissionQueue": false,
 "description": "A queue on QM3."
 },
 "name" : "Qon3",
 "type" : "local"
 }
]
```

## V 9.0.2 REST API datowniki

Gdy informacje o dacie i godzinie są zwracane przez administrative REST API, jest ono zwracane w formacie czasu uniwersalnego (Coordinated Universal Time-UTC) i w formacie zestawu.

Data i godzina są zwracane w następującym formacie datownika:

```
YYYY-MM-DDTHH:mm:ss:sssZ
```

Na przykład 2012-04-23T18:25:43.000Z, gdzie Z wskazuje, że strefa czasowa jest koordynowana czasem uniwersalnym (UTC).

W programie IBM MQ 9.0.2 dokładność tego znacznika czasu nie jest gwarantowana. Na przykład, jeśli serwer mqweb nie jest uruchomiony w tej samej strefie czasowej co menedżer kolejek określony w adresie URL zasobu, znacznik czasu może nie być dokładny. Ponadto, jeśli konieczne jest dostosowanie czasu letniego, znacznik czasu może nie być dokładny.

## REST API obsługa błędów

Produkt REST API zgłasza błędy, zwracając odpowiedni kod odpowiedzi HTTP, na przykład 404 (Nie znaleziono) i odpowiedź JSON. Każdy kod odpowiedzi HTTP, który nie znajduje się w zakresie od 200 do 299, jest uznawany za błąd.

### Format odpowiedzi błędu

Odpowiedź jest w formacie JSON w kodowaniu UTF-8 . Zawiera zagnieżdżone obiekty JSON:

- Zewnętrzny obiekt JSON, który zawiera pojedynczą tablicę JSON o nazwie `error`.
- Każdy element w tablicy jest obiektem JSON, który reprezentuje informacje o błędzie. Każdy obiekt JSON zawiera następujące właściwości:

#### **typ**

Łańcuch.

Typ błędu.

#### **messageId**

Łańcuch.

Unikalny identyfikator komunikatu w postaci MQWBnnnnX. Ten identyfikator składa się z następujących elementów:

## MQWB

Przedrostek wskazujący, że komunikat pochodzi z interfejsu API REST produktu MQ .

## nnnn

Unikalny numer identyfikujący komunikat.

## X

Pojedynczy list oznaczający istotność komunikatu:

- I , jeśli komunikat ma charakter wyłącznie informacyjny.
- W , jeśli komunikat jest ostrzegany przed problemem.
- E , jeśli komunikat wskazuje, że wystąpił błąd.
- S , jeśli komunikat wskazuje, że wystąpił poważny błąd.

## message (komunikat)

Łańcuch.

Opis błędu.

## objaśnienie

Łańcuch.


Wyjaśnienie błędu.

## działanie

Łańcuch.


Opis kroków, które można podjąć w celu rozwiązania tego błędu.

## qmgrName

 To pole jest dostępne tylko w przypadku produktu z/OS , w którym menedżer kolejek jest elementem grupy współużytkowania kolejki. Należy podać opcjonalny parametr zapytania **commandScope** lub atrybut **queueSharingGroupDisposition** .

Łańcuch.

Nazwa menedżera kolejek, który napotkał błąd.

 To pole nie ma zastosowania w przypadku serwera messaging REST API.

This field is only available when **type** is pcf, java, or rest.

Liczba.

Kod zakończenia MQ powiązany z niepowodzeniem.

## reasonCode

This field is only available when **type** is pcf, java, or rest.

Liczba.

Kod przyczyny MQ powiązany z awarią.

## wyjątki

To pole jest dostępne tylko wtedy, gdy **type** jest java.

Tablica.

Tablica łańcuchów języka Java lub wyjątków JMS. Każdy element tablicy wyjątków zawiera tablicę łańcuchów **stackTrace** .

Tablica łańcuchów **stackTrace** zawiera szczegółowe informacje o każdym wyjątku w podziale na wiersze.

## Błędy w grupach współużytkowania kolejek

### 

W grupie współużytkowania kolejek możliwe jest określenie opcjonalnego parametru zapytania **commandScope** dla niektórych komend. Ten parametr umożliwia propagację komendy do innych menedżerów kolejek w grupie współużytkowania kolejek. Każda z tych komend może zakończyć się


niepowodzeniem niezależnie, co spowoduje pomyślne wykonanie niektórych komend i wykonanie niektórych komend, które nie powiedzą się dla grupy współużytkownika kolejek.

W przypadkach, w których komenda częściowo nie powiedzie się, zwracany jest kod błędu HTTP o wartości 500. Dla każdego menedżera kolejek, który wygenerował błąd, informacje na temat tego niepowodzenia są zwracane jako element w tablicy JSON produktu `error`. Dla każdego menedżera kolejek, który pomyślnie uruchomił komendę, nazwa tego menedżera kolejek jest zwracana jako element w tablicy JSON produktu `success`.

## Przykłady

- W poniższym przykładzie przedstawiono odpowiedź na błąd w celu uzyskania informacji o menedżerze kolejek, który nie istnieje:

```
"error": [
 {
 "type": "rest",
 "messageId": "MQWB0009E",
 "message": "MQWB0009E: Could not query the queue manager 'QM1'",
 "explanation": "The MQ REST API was invoked specifying a queue manager name which cannot be located.",
 "action": "Resubmit the request with a valid queue manager name or no queue manager name, to retrieve a list of queue managers."
 }
]
```

-  `z/OS` W poniższym przykładzie przedstawiono odpowiedź na błąd w celu usunięcia kolejki w grupie współużytkowania kolejek, która nie istnieje dla niektórych menedżerów kolejek:

```
"error" : [
 {
 "type": "rest",
 "messageId": "MQWB0037E",
 "message": "MQWB0037E: Could not find the queue 'missingQueue' - the queue manager reason code is 3312 : 'MQRCCF_UNKNOWN_OBJECT_NAME'",
 "explanation": "The MQ REST API was invoked specifying a queue name which cannot be located.",
 "action": "Resubmit the request with the name of an existing queue, or with no queue name to retrieve a list of queues.",
 "qmgrName": "QM1"
 },
 {
 "type": "rest",
 "messageId": "MQWB0037E",
 "message": "MQWB0037E: Could not find the queue 'missingQueue' - the queue manager reason code is 3312 : 'MQRCCF_UNKNOWN_OBJECT_NAME'",
 "explanation": "The MQ REST API was invoked specifying a queue name which cannot be located.",
 "action": "Resubmit the request with the name of an existing queue, or with no queue name to retrieve a list of queues.",
 "qmgrName": "QM2"
 }
],
"success" : [{"qmgrName": "QM3"}, {"qmgrName": "QM4"}]
```

## Błędy w żądaniach MFT

Jeśli usługi MFT REST API nie są włączone, a użytkownik wywoła MFT REST API, zostanie zgłoszony następujący wyjątek:

```
{"error": [{
 "action": "Enable the Managed File Transfer REST API and resubmit the request.",
 "completionCode": 0,
 "explanation": "Managed File Transfer REST calls are not permitted as the service is disabled.",
 "message": "MQWB0400E: Managed File Transfer REST API is not enabled.",
 "msgId": "MQWB0400E",
 "reasonCode": 0,
```

```
"type": "rest"
}]}
```

Jeśli usługi MFT REST API są włączone, a menedżer kolejek koordynacji nie jest ustawiony w pliku `mqwebuser.xml`, zostanie wyświetlony następujący wyjątek:

```
{"error": [{
 "action": "Set the coordination queue manager name and restart the mqweb server.",
 "completionCode": 0,
 "explanation": "Coordination queue manager name must be set before using Managed File
Transfer REST services.",
 "message": "MQWB0402E: Coordination queue manager name is not set.",
 "msgId": "MQWB0402E",
 "reasonCode": 0,
 "type": "rest"
}]}
```

## V 9.0.1 REST API Wykrywanie

Sekcja Documentation dla produktu REST API jest dostępna w produkcie IBM Documentation i w formacie Swagger. Program Swagger jest powszechnie używanym podejściem do dokumentowania interfejsów REST API. Dokumentację Swagger dla produktu REST API można wyświetlić, włączając funkcję wykrywania interfejsu API na serwerze mqweb.

### Zanim rozpoczniesz

Należy włączyć zabezpieczenia serwera mqweb, aby wyświetlić dokumentację Swagger za pomocą wykrywania interfejsu API. Więcej informacji o krokach, które są wymagane do włączenia zabezpieczeń, zawiera sekcja [Konfigurowanie zabezpieczeń produktu IBM MQ Console](#).

### Procedura

1. Znajdź plik `mqwebuser.xml` w jednym z następujących katalogów:

- **ULW** `MQ_DATA_DIRECTORY/web/installations/installationName/servers/mqweb`
- **Z/OS** `WLP_user_directory/servers/mqweb`

Gdzie `WLP_user_directory` to katalog, który został określony podczas uruchamiania skryptu `crtmqweb.sh` w celu utworzenia definicji serwera WWW mqweb.

2. Dodaj odpowiedni kod XML do pliku `mqwebuser.xml`:

- Jeśli znaczniki `<featureManager>` istnieją w pliku `mqwebuser.xml`, należy dodać następujący kod XML w znacznikach produktu `<featureManager>`:
- ```
<feature>apiDiscovery-1.0</feature>
```
- Jeśli znaczniki produktu `<featureManager>` nie istnieją w pliku `mqwebuser.xml`, należy dodać następujący kod XML w znacznikach produktu `<server>`:

```
<featureManager>
  <feature>apiDiscovery-1.0</feature>
</featureManager>
```

3. Zapoznaj się z dokumentacją Swagger, używając jednej z następujących metod:

- Wyświetl stronę WWW, którą można przeglądać i wypróbować REST API, wprowadzając w przeglądarce następujący adres URL:

```
https://host:port/ibm/api/explorer
```

Oprócz uwierzytelniania każdego żądania, należy dołączyć nagłówek `ibm-mq-rest-csrf-token` dla każdego żądania POST, PATCH lub DELETE.

V 9.0.5

Wymagana zawartość tego nagłówka różni się w zależności od wersji produktu IBM MQ:

- W przypadku produktu IBM MQ 9.0.5 nagłówek HTTP `ibm-mq-rest-csrf-token` musi być obecny w żądaniu; jego wartość może być dowolna, w tym wartość pusta.
- W wersjach wcześniejszych niż IBM MQ 9.0.5 wartością nagłówka jest treść informacji cookie produktu `csrfToken`. `csrfToken` jest generowana, gdy z REST API używana jest metoda HTTP GET. Zawartość plików cookies można przeglądać, wprowadzając następujący tekst na pasku adresu przeglądarki:

```
javascript:alert(document.cookie)
```

Ten nagłówek żądania jest używany do potwierdzenia, że informacje autoryzacyjne używane do uwierzytelniania żądania są używane przez właściciela referencji. Oznacza to, że znacznik jest używany w celu zapobiegania atakom typu cross-site request forgery.

- Pobierz pojedynczy dokument Swagger 2, który opisuje całe REST API, wysyłając żądanie HTTP GET do następującego adresu URL:

```
https://host:port/ibm/api/docs
```

Ten dokument może być używany w aplikacjach, w których wymagane jest programowo nawigowanie w dostępnych interfejsach API.

host

Określa nazwę hosta lub adres IP, na którym jest dostępny REST API.

Wartością domyślną jest `localhost`.

Port

Określa numer portu HTTPS używany przez produkt administrative REST API.

Wartością domyślną jest `9443`.

Jeśli nazwa hosta lub numer portu jest zmieniany z wartości domyślnej, można określić poprawne wartości z adresu URL produktu REST API. Więcej informacji na temat określania adresu URL zawiera sekcja [“Określanie adresu URL produktu REST API”](#) na stronie 76.

V 9.0.4

REST API Obsługa języków narodowych

Produkt REST API obsługuje, z pewnymi kwalifikacjami, możliwość określania języków narodowych jako części żądania HTTP.

Tłó

Nagłówki HTTP umożliwiają określenie konkretnych zachowań w żądaniach i dodatkowych informacjach, które mają być udostępniane w odpowiedziach.

Zawarte w nagłówkach HTTP jest możliwość żądania zwrócenia informacji w języku narodowym. REST API honoruje ten nagłówek tam, gdzie jest to możliwe.

Określanie języka narodowego

W nagłówku `ACCEPT-LANGUAGE` HTTP można podać jeden lub więcej znaczników języka. Opcjonalnie można powiązać klasyfikację z oznakacjami, co pozwala na określenie listy uporządkowanej według preferencji. [Ta strona](#) zawiera pomocną dyskusję na temat zasady.

REST API honoruje ten nagłówek, wybierając język z nagłówka `ACCEPT-LANGUAGE` i zwracając komunikaty w tym języku. Jeśli nagłówek `ACCEPT-LANGUAGE` nie zawiera języka, który może być używany przez produkt REST API, komunikaty są zwracane w języku domyślnym. Ten język domyślny odpowiada domyślnym ustawieniom narodowych serwera WWW REST API.

Sekcja [“Jakie dane są tłumaczone?”](#) na stronie 94 wyjaśnia, jakie dane są tłumaczone.

Wskazywania odpowiedniego języka na odpowiedziach

Nagłówek CONTENT-LANGUAGE HTTP w odpowiedziach z REST API wskazuje język, w którym zwracane są komunikaty.

Jakie dane są tłumaczone?

Komunikaty o błędach i komunikaty informacyjne są tłumaczone, a inny tekst nie jest.

- Dane zwracane przez menedżera kolejek nie są tłumaczone-na przykład w przypadku wykonywania komendy MQSC za pomocą REST API odpowiedzi menedżera kolejek są określone w ustawieniach narodowych menedżera kolejek.
- Wygenerowana (Swagger) dokumentacja produktu REST API, która jest prezentowana za pomocą funkcji apiDiscovery , jest w języku angielskim.

Jakie języki są obsługiwane?

Oprócz języka angielskiego komunikaty o błędach i komunikaty informacyjne produktu REST API są tłumaczone na następujące języki.

Chiński (uproszczony)

Oznaczone przez znacznik języka zh_CN

Chiński (tradycyjny)

Oznaczone przez znacznik języka zh_TW

czeski

Oznaczone przez znacznik języka cs

francuski

Oznaczone przez znacznik języka fr

węgierski

Oznaczone przez znacznik języka hu

włoski

Oznaczone przez znacznik języka it

japoński

Oznaczone przez znacznik języka ja

koreański

Oznaczone przez znacznik języka ko

polski

Oznaczone przez znacznik języka pl

(brazylijski) portugalski

Oznaczone przez znacznik języka pt_BR

rosyjski

Oznaczone przez znacznik języka ru

hiszpański

Oznaczone przez znacznik języka es

Przykłady

W przykładach serwer WWW ma domyślne ustawienia narodowe w języku angielskim.

Określanie pojedynczego obsługiwanego języka

W nagłówkach żądań program ACCEPT-LANGUAGE jest ustawiony na wartość fr. To ustawienie określa, że język francuski jest preferowanym językiem dla przetłumaczonego tekstu.

W nagłówkach odpowiedzi produkt CONTENT-LANGUAGE jest ustawiony na wartość fr. To ustawienie wskazuje, że komunikaty o błędach i komunikaty informacyjne w odpowiedzi są w języku francuskim.

Określanie listy języków

W nagłówkach żądań program ACCEPT - LANGUAGE jest ustawiony na wartość `am`, `fr`. To ustawienie określa, że Amharic i francuski są akceptowalnymi językami do przetłumaczonego tekstu i że Amharic jest preferowanym językiem do przetłumaczonego tekstu.

W nagłówkach odpowiedzi produkt CONTENT - LANGUAGE jest ustawiony na wartość `fr`. To ustawienie wskazuje, że komunikaty o błędach i komunikaty informacyjne w odpowiedzi są w języku francuskim, ponieważ produkt REST API nie obsługuje systemu Amharic.

Określanie pojedynczego nieobsługiwanego języka

W nagłówkach żądań program ACCEPT - LANGUAGE jest ustawiony na wartość `am`. To ustawienie określa, że język Amharic jest preferowanym językiem do przetłumaczonego tekstu.

W nagłówkach odpowiedzi produkt CONTENT - LANGUAGE jest ustawiony na wartość `en`. To ustawienie wskazuje, że komunikaty o błędach i komunikaty informacyjne w odpowiedzi są w języku angielskim, ponieważ produkt REST API nie obsługuje systemu Amharic.

V 9.0.1 Administrowanie przy użyciu IBM MQ Console

IBM MQ Console to internetowy interfejs użytkownika, który może być używany do wykonywania typowych zadań administracyjnych.

Zanim rozpocznie

Uwaga:

- Nie należy wyłączać serwera komend na żadnym z menedżerów kolejek, gdy używany jest produkt IBM MQ Console. Jeśli serwer komend jest wyłączony dla menedżera kolejek, IBM MQ Console nie reaguje z długimi opóźnieniami w przetwarzaniu komend. Wszystkie komendy, które są wysyłane do menedżera kolejek, dla którego serwer komend jest wyłączony.
- Po nawiązaniu połączenia z serwerem IBM MQ Console, przeglądarka próbuje wyświetlić IBM MQ Console w ustawieniach narodowych ustawionych przez przeglądarkę. W celu sprawdzenia, czy produkt IBM MQ Console obsługuje określony język, należy sprawdzić. Jeśli pliki językowe nie zostaną znalezione, używany jest język angielski (Stany Zjednoczone), a serwer mqweb rejestruje ten plik, ponieważ nie znaleziono błędów. Dlatego następujące komunikaty są oczekiwane, gdy przeglądarka jest ustawiona na język, który nie jest obsługiwany przez produkt IBM MQ Console:

```
SRVE0190E: File not found: /nls/en_GB/labels.json
SRVE0190E: File not found: /nls/en_GB/pcf.json
SRVE0190E: File not found: /nls/en_GB/errors.json
SRVE0190E: File not found: /nls/en_GB/strings.json
```

Procedura

- [Praca z menedżerami kolejek lokalnych](#)
- [Praca z obiektami IBM MQ](#)
- [Praca z rekordami uprawnień](#)
- [Monitorowanie wykorzystania zasobów systemowych](#)
- [Konfigurowanie układów panelu kontrolnego](#)

V 9.0.1 Pierwsze kroki w produkcie IBM MQ Console

Przed uruchomieniem produktu IBM MQ Console należy zainstalować odpowiednie komponenty i uruchomić serwer mqweb. Następnie produkt IBM MQ Console można uruchomić w przeglądarce.

Zanim rozpocznie

IBM i

W systemie IBM i komendy powinny być uruchomione w systemie QSHLL.

O tym zadaniu

Procedura dotycząca tego zadania koncentruje się na podstawowych krokach, aby szybko rozpocząć działanie z produktem IBM MQ Console. Więcej informacji na temat dostosowywania konfiguracji można znaleźć w sekcji odsyłaczy wyświetlanych pod tym, co należy zrobić dalej.

Uwaga: Opcja ta jest dostępna podczas instalowania produktu IBM MQ Console bez zabezpieczeń.

Procedura

1. Zainstaluj komponent IBM MQ Console i REST API :

- **V 9.0.4** **AIX** W systemie AIX zainstaluj zestaw plików `mqm.web.rte`.
- **Linux** W systemie Linux zainstaluj komponent `MQSeriesWeb`. Więcej informacji na temat instalowania komponentów i składników w systemie Linux można znaleźć w sekcji [Zadania instalacyjne produktu Linux](#).
- **Windows** W systemie Windows zainstaluj składnik `Web Administration`. Więcej informacji na temat instalowania komponentów i składników w systemie Windows można znaleźć w sekcji [Zadania instalacyjne produktu Windows](#).
- **z/OS** W systemie z/OS zainstaluj składnik `IBM MQ for z/OS Unix System Web Services Components`. Więcej informacji na temat instalowania komponentów i składników w systemie z/OS można znaleźć w sekcji [Zadania instalacyjne produktu z/OS](#).

2. Skonfiguruj podstawowe zabezpieczenia, aby umożliwić użytkownikom i grupom uzyskanie dostępu do produktu IBM MQ Console.

a) Skopiuj przykładowy plik `basic_registry.xml` z katalogu `MQ_INSTALLATION_PATH/web/mq/samp/configuration` do katalogu:

- **ULW** W systemie UNIX, Linux, and Windows: `MQ_DATA_DIRECTORY/web/installations/installationName/servers/mqweb`
- **z/OS** W systemie z/OS: `WLP_user_directory/servers/mqweb`

gdzie `katalog_uzytkownika_WLP_uzytkownika` jest katalogiem, który został określony podczas uruchamiania skryptu `crtmqweb.sh` w celu utworzenia definicji serwera WWW `mqweb`.

b) Zmień nazwę przykładowego pliku XML na `mqwebuser.xml`.

Uwaga: Ten plik o zmienionej nazwie zastępuje istniejący plik, który jest również używany dla administracyjnej REST API. Dlatego, jeśli zmieniono plik `mqwebuser.xml` dla administracyjnej REST API, należy skopiować zmiany do nowego pliku XML przed zmianą nazwy.

3. Zależne od platformy, włącz połączenia zdalne z serwerem `mqweb`:

- **Linux** Opcjonalnie w systemie Linux
- **Windows** Opcjonalnie w systemie Windows
- **z/OS** `wł.z/OS`
- W programie IBM MQ 9.0.4 należy użyć komendy **setmqweb** :
`setmqweb properties -k httpHost -v hostname`
- W produkcie IBM MQ 9.0.1 dodaj następujący kod XML do pliku `mqwebuser.xml` w znacznikach produktu `<server>` :
`<variable name="httpHost" value="hostname"/>`

gdzie `nazwa_hosta` określa adres IP, nazwę hosta serwera nazw domen (DNS) z przyrostkiem nazwy domeny lub nazwę DNS hosta serwera, na którym zainstalowany jest produkt IBM MQ. Aby określić wszystkie dostępne interfejsy sieciowe, należy użyć gwiazdki (*).

**Ostrzeżenie:** z/OS V 9.0.4

Przed wydaniem komendy **setmqweb** lub **dspmqweb** w systemie z/OS należy ustawić zmienną środowiskową WLP_USER_DIR tak, aby wskazywała konfigurację serwera mqweb.

W tym celu należy wydać następującą komendę:

```
export WLP_USER_DIR=WLP_user_directory
```

gdzie *WLP_user_directory* to nazwa katalogu, który jest przekazywany do produktu *crtmqweb.sh*. Na przykład:

```
export WLP_USER_DIR=/var/mqm/web/installation1
```

Więcej informacji na ten temat zawiera sekcja [Tworzenie definicji serwera Liberty](#).

4. Uruchom serwer mqweb, który obsługuje IBM MQ Console:

- **Windows** **Linux** W systemach Windows i Linux jako użytkownik uprzywilejowany wprowadź w wierszu komend następującą komendę:
stmqweb
- **z/OS** W systemie z/OS uruchom procedurę utworzoną w polu Czynność 29: Tworzenie procedury dla serwera IBM WLP.

5. Połącz się z serwerem IBM MQ Console, wpisując w przeglądarce następujący adres URL:

```
https://hostname:9443/ibmmq/console
```

gdzie *nazwa_hosta* określa adres IP lub nazwę hosta serwera nazw domen (DNS) z przyrostkiem nazwy domeny lub nazwę DNS hosta serwera, na którym zainstalowano produkt IBM MQ. Jeśli w kroku 3 nie skonfigurowano połączeń zdalnych, wartość *nazwa_hosta* to localhost.

6. Zaloguj się do konsoli IBM MQ Console. W produkcie IBM MQ 9.0.2 należy użyć nazwy użytkownika *mqadmini* hasła *mqadmin*. W systemie IBM MQ 9.0.1 należy użyć nazwy użytkownika *admini* hasła *admin*.

Co dalej

- Więcej informacji na temat konfigurowania zabezpieczeń produktu IBM MQ Console, w tym konfigurowania użytkowników i grup, protokołu LDAP i certyfikatów klienta, zawiera sekcja [Konfigurowanie zabezpieczeń produktu IBM MQ Console](#).
- Więcej informacji na temat konfigurowania ustawień produktu IBM MQ Console, w tym włączania połączeń HTTP, zawiera sekcja [Konfigurowanie konsoli produktu IBM MQ](#).
- Więcej informacji na temat określania adresu URL, jeśli nie jest to domyślny adres URL, zawiera sekcja ["Określanie adresu URL produktu IBM MQ Console"](#) na stronie 97.

V 9.0.1

Określanie adresu URL produktu IBM MQ Console

Domyślnym adresem URL do uzyskania dostępu do produktu IBM MQ Console jest `https://localhost:9443/ibmmq/console`. Jeśli host lub port HTTP zostanie zmieniony z wartości domyślnej lub jeśli port HTTP jest włączony, można określić adres URL za pomocą komendy **dspmqweb**.

O tym zadaniu

V 9.0.4

Za pomocą komendy IBM MQ 9.0.4 można użyć komendy **dspmqweb status** w celu określenia adresu URL produktu IBM MQ Console w systemach Windows, Linux i z/OS. W przypadku systemu IBM MQ 9.0.3 i wcześniejszych wersji w systemach Windows i Linux można użyć komendy **dspmqweb**. W systemie z/OS należy przeszukać plik `messages.log`, aby określić adres URL.

Procedura

V 9.0.4

Aby określić adres URL, należy użyć jednej z następujących metod:

- W produkcie IBM MQ 9.0.4 jako użytkownik uprzywilejowany należy użyć komendy **dspmqrweb status** :
 1. Upewnij się, że serwer mqweb jest uruchomiony, wprowadzając komendę **strmqweb** w wierszu komend.
 2. Wyświetl adres URL, wprowadzając komendę **dspmqrweb status** w wierszu komend.
- W przypadku produktu IBM MQ 9.0.3 i wcześniejszych wersji należy użyć jednej z następujących metod:

- **Windows** / **Linux** W systemie Windows lub Linux należy użyć komendy **dspmqrweb** jako użytkownika uprzywilejowanego:

1. Upewnij się, że serwer mqweb jest uruchomiony, wprowadzając komendę **strmqweb** w wierszu komend.
2. Wyświetl adres URL, wprowadzając komendę **dspmqrweb** w wierszu komend.

- **z/OS** W systemie z/OS użyj pliku `messages.log`, aby znaleźć adres URL:

1. Otwórz plik `messages.log`.

Plik `messages.log` znajduje się w ścieżce `WLP_user_directory/servers/mqweb/logs`, gdzie `WLP_user_directory` jest katalogiem, który został określony podczas uruchamiania skryptu `crtmqrweb.sh` w celu utworzenia definicji serwera WWW mqweb.

2. Wyszukaj najnowszy kod komunikatu CWWKT0016I, który kończy się na `ibmmq/console`. Adres URL jest zawarty w tym komunikacie.

Uwaga: Produkt IBM MQ Console korzysta z wewnętrznego adresu URL produktu / `ibmmq/console/internal`. Ten adres URL jest wyświetlany w niektórych komunikatach CWWKT0016I, ale może być używany tylko przez IBM MQ Console.

Przykład

W poniższym przykładzie przedstawiono kod komunikatu CWWKT0016I z adresem URL jako domyślny adres URL:

A CWWKT0016I: Web application available (default_host): <https://localhost:9443/ibmmq/console>

z/OS V 9.0.1 Ograniczenia w systemie z/OS

W przypadku korzystania z produktu IBM MQ Console do zarządzania menedżerami kolejek w systemie z/OS obowiązują następujące ograniczenia.

- Menedżery kolejek w systemie z/OS nie mogą być tworzone, usuwane, uruchamiane ani zatrzymane.
- Inicjatory kanału w systemie z/OS nie mogą być uruchamiane ani zatrzymane, a status inicjatora kanału nie jest wyświetlany.
- Programy nasłuchujące nie mogą być wyświetlane ani administrowane.
- Komendy `start`, `ping`, `resolve` i `reset` kanału mogą być wydawane tylko z opcją `CHLDISP (DEFAULT)`.
- Nowe obiekty mogą być tworzone tylko z `QSGDISP (QMGR)`.
- Obiekty zdefiniowane za pomocą `QSGDISP (GROUP)` nie mogą być wyświetlane ani zarządzane.
- Zabezpieczenia menedżera kolejek nie mogą być zarządzane.
- Użycie zasobów systemowych nie może być monitorowane.

Zadania pokrewne

“Administrowanie przy użyciu IBM MQ Console” na stronie 95

IBM MQ Console to internetowy interfejs użytkownika, który może być używany do wykonywania typowych zadań administracyjnych.

Informacje pokrewne

Praca z menedżerami kolejek lokalnych

V 9.0.1 Praca z menedżerami kolejek lokalnych


Za pomocą widgetu menedżera kolejek lokalnych w produkcie IBM MQ Console można tworzyć, konfigurować i kontrolować lokalne menedżery kolejek.

O tym zadaniu

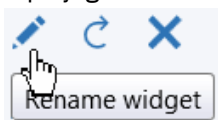
Widget menedżera kolejek lokalnych zawiera listę menedżerów kolejek lokalnych, które są dodawane do instalacji produktu IBM MQ, z której uruchomiony jest produkt IBM MQ Console. Menedżery kolejek, które są powiązane z różnymi instalacjami produktu IBM MQ w tym samym systemie, nie są wyświetlane na liście. Z listy można wybrać pojedyncze menedżery kolejek, z którymi ma pracować praca.

Uwaga: V 9.0.4 Produkt IBM MQ Console nie obsługuje replikowanych menedżerów kolejek danych (RDQMs).


Istnieje możliwość dodania widgetu lokalnego menedżera kolejek do panelu kontrolnego, klikając opcję

Dodaj widget V 9.0.5 . Następnie wybierz opcję **Lokalne menedżery kolejek**.



V 9.0.5 Nazwę widgetu można zmienić po jego utworzeniu. Umieść wskaźnik myszy na pasku tytułu,






a następnie pojawi się ikona zmiany nazwy . Kliknij ikonę, wprowadź nową nazwę w oknie Zmiana nazwy widgetu i kliknij przycisk **Zmień nazwę**.

 Nie można tworzyć, uruchamiać, zatrzymować ani usuwać menedżerów kolejek w systemie z/OS.

Procedura

- Aby utworzyć nowy menedżer kolejek lokalnych:
 - a) Kliknij ikonę tworzenia V 9.0.5  na pasku narzędzi widgetu menedżera kolejek lokalnych.
 - b) Wprowadź nazwę nowego menedżera kolejek. Nazwa może zawierać maksymalnie 48 znaków. Poprawnymi znakami są litery i cyfry oraz znaki ":", "/", "_i \"%".
 - c) Opcjonalne: Wprowadź dostępny port TCP/IP dla menedżera kolejek, na którym ma być nasłuchiwać. Numer portu nie może być większy niż 65535.
 - d) Kliknij opcję **Create**. Nowy menedżer kolejek zostanie utworzony i uruchomiony.
- Aby uruchomić lokalny menedżer kolejek:
 - a) Wybierz z listy menedżer kolejek, który ma zostać uruchomiony, w widżecie menedżera kolejek lokalnych.
 - b) Kliknij ikonę uruchamiania V 9.0.5  na pasku narzędzi widgetu menedżera kolejek lokalnych.
- Aby zatrzymać lokalny menedżer kolejek:

- a) Wybierz z listy menedżer kolejek, który ma zostać zatrzymany, w widżecie menedżera kolejek lokalnych.
- b) Kliknij ikonę zatrzymania  na pasku narzędzi widżetu menedżera kolejek lokalnych.
- c) Upewnij się, że menedżer kolejek ma zostać zatrzymany, klikając przycisk **Zatrzymaj**.
- Aby usunąć lokalny menedżer kolejek:
 - a) Wybierz z listy menedżer kolejek, który chcesz usunąć z listy w lokalnym widżecie menedżera kolejek.
 - b) Jeśli menedżer kolejek jest uruchomiony, zatrzymaj go.
 - c) Kliknij ikonę usuwania  na pasku narzędzi widżetu menedżera kolejek lokalnych.
 - d) Potwierdź, że menedżer kolejek ma zostać usunięty, klikając przycisk **Usuń**. Menedżer kolejek i wszystkie powiązane obiekty są usuwane.
 - Aby wyświetlić i edytować właściwości lokalnego menedżera kolejek:
 - a) Upewnij się, że menedżer kolejek jest uruchomiony, a następnie wybierz go na liście menedżerów kolejek.
 - b) Kliknij ikonę właściwości  na pasku narzędzi widżetu menedżera kolejek lokalnych. Alternatywnie kliknij dwukrotnie menedżer kolejek.
 - c) Wyświetl właściwości i edytuj je zgodnie z wymaganiami. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend. Informacje na temat właściwości można znaleźć w sekcji [Właściwości menedżera kolejek](#) w dokumentacji programu MQ Explorer.
 - Aby odświeżyć zabezpieczenia lokalnego menedżera kolejek:
 - a) Upewnij się, że lokalny menedżer kolejek jest uruchomiony, a następnie wybierz go na liście menedżerów kolejek.
 - b) Wybierz opcję ... > **Odśwież zabezpieczenia**
 - c) Wybierz zabezpieczenia menedżera kolejek do odświeżenia:
 - Wybierz opcję **Usługa autoryzacji**, aby odświeżyć listę autoryzacji, które są przechowywane wewnątrz przez komponent usług autoryzacji.
 - Wybierz opcję **Uwierzytelnianie połączenia**, aby odświeżyć buforowany widok konfiguracji na potrzeby uwierzytelniania połączenia.
 - Wybierz opcję **SSL**, aby odświeżyć buforowany widok repozytorium kluczy SSL lub TLS. Ta opcja odświeża również połączenia serwerów LDAP, które są używane na potrzeby certyfikowanych list odwołań, a także wszystkie parametry sprzętu szyfrującego.
 - Aby pracować z rekordami uprawnień dla lokalnego menedżera kolejek:
 - a) Upewnij się, że lokalny menedżer kolejek jest uruchomiony, a następnie wybierz go na liście menedżerów kolejek.
 - b) Wybierz jedną z następujących opcji:
 - Wybierz opcję ... > **Zarządzaj rekordami uprawnień**, aby pracować z rekordami uprawnień dla menedżera kolejek, a także określić, jakie grupy działań mogą być wykonywane przez użytkowników.
 - Wybierz opcję ... > **Zarządzaj rekordami uprawnień do tworzenia**, aby pracować z rekordami uprawnień do tworzenia dla menedżera kolejek oraz określić, które grupy obiektów mogą być tworzone przez użytkowników w tym menedżerze kolejek.
 - Aby automatycznie utworzyć kartę panelu kontrolnego dla lokalnego menedżera kolejek:

- a) Wybierz menedżer kolejek w widżecie menedżera kolejek lokalnych.
- b) Wybierz opcję ... > **Dodaj nową kartę panelu kontrolnego**
Zostanie utworzona nowa karta panelu kontrolnego. Na karcie znajduje się nazwa menedżera kolejek.
- Aby filtrować listę menedżerów kolejek lokalnych:
 - a) Wpisz tekst filtru w polu wyszukiwania.
 - b) Aby zatrzymać filtrowanie, należy usunąć tekst z pola wyszukiwania.

V 9.0.1 Praca z obiektami IBM MQ



Widżety obiektów produktu IBM MQ w produkcie IBM MQ Console mogą być używane do pracy z różnymi typami obiektów produktu IBM MQ .

O tym zadaniu

Każdy widżet obiektu IBM MQ zawiera obiekty, które są powiązane z określonym menedżerem kolejek. Do panelu kontrolnego można dodać następujące typy widżetów obiektów produktu IBM MQ :

- Kolejki, widżet
- widżet tematu
- Widżet Obiekty nastuchiwania
- widżet kanałów
- Kanał połączenia klienckiego-widżet
- widżet informacji uwierzytelniających
- Widżet Subskrypcje

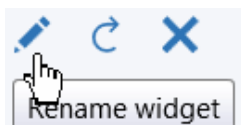
Procedura

- Aby utworzyć widżet obiektu IBM MQ :
 - a) Kliknij opcję **Dodaj widżet**   .
 - b) Wybierz z listy odpowiedni menedżer kolejek.
 - c) Kliknij nazwę typu widżetu obiektu, który ma zostać utworzony.

V 9.0.5

Aby zmienić nazwę widżetu obiektu:

- a) Umieść wskaźnik myszy na pasku tytułu, a następnie pojawi się ikona zmiany nazwy






. Kliknij ikonę.

- b) Wprowadź nową nazwę w oknie Zmiana nazwy widżetu i kliknij przycisk **Zmień nazwę**.

V 9.0.5

Aby skonfigurować widżet obiektu IBM MQ :

- a) Kliknij ikonę konfiguracji  na pasku tytułu widżetu.
- b) Opcjonalne: Określ menedżer kolejek, dla którego są wyświetlane obiekty produktu IBM MQ .
- c) Opcjonalne: Określ, czy obiekty systemowe są wyświetlane, czy ukryte.
- d) Kliknij przycisk **Save** (Zapisz).
- Aby filtrować obiekty, które są wyświetlane w widżecie:

- a) Wpisz tekst filtru w polu wyszukiwania.
- b) Aby zatrzymać filtrowanie, należy usunąć tekst z pola wyszukiwania.
- Aby odświeżyć zawartość widgetu, kliknij ikonę odświeżania  na pasku tytułu widgetu.
- Aby usunąć widget, kliknij ikonę usuwania  znajdującą się na pasku tytułu widgetu.

Praca z kolejkami


Za pomocą widgetu Kolejki w produkcie IBM MQ Console można wyświetlić kolejki, które istnieją dla konkretnego menedżera kolejek. Następnie można dodawać i usuwać kolejki, dodawać i usuwać komunikaty w kolejce, przeglądać komunikaty, wyświetlać i ustawiać właściwości kolejki, a także zarządzać rekordami uprawnień kolejki.

Zanim rozpoczniesz




Przed użyciem widgetu kolejki należy utworzyć widget kolejki. Więcej informacji na temat tworzenia widgetów obiektów IBM MQ zawiera sekcja [“Praca z obiektami IBM MQ” na stronie 101](#).





O tym zadaniu

Widget kolejki zawiera listę kolejek, które istnieją dla konkretnego menedżera kolejek. Z listy można wybrać poszczególne kolejki, z którymi ma pracować praca.

 Nie można wyświetlać ani edytować rekordów uprawnień dla kolejek w systemie z/OS.

Procedura

- Aby dodać kolejkę:
 - a) Kliknij ikonę tworzenia  na pasku narzędzi widgetu kolejki.
 - b) Wprowadź nazwę kolejki. Poprawnymi znakami są litery i cyfry oraz znaki ".", "/", "_i" "%".
 - c) Wybierz typ kolejki do dodania.
 - d) Kliknij opcję **Create**. Zostanie utworzona nowa kolejka.
- Aby umieścić komunikaty w kolejce:
 - a) Wybierz z listy w widżecie kolejki kolejkę, do której mają zostać dodane komunikaty. Nie można wybrać kolejki modelowej.
 - b) Kliknij ikonę umieszczonego komunikatu  na pasku narzędzi widgetu kolejki.
 - c) Wprowadź komunikat, który ma zostać umieszczony w kolejce.
 - d) Kliknij przycisk **Umieść**.
- Aby wyczyścić komunikaty z kolejki:
 - a) Wybierz z listy w widżecie kolejki kolejkę lokalną, która ma zostać skasowana.
 - b) Wybierz opcję  **Wyczyść kolejkę**.
 - c) Potwierdź, że chcesz wyczyścić kolejkę, klikając opcję **Wyczyść kolejkę**.
- Aby przeglądać komunikaty w kolejce:
 - a) Z listy w widżecie kolejek wybierz kolejkę lokalną lub kolejkę aliasową, która ma być przeglądany.
 - b) Kliknij ikonę przeglądania  na pasku narzędzi widgetu kolejki. Zostanie otwarte okno przeglądania komunikatów, w którym wyświetlane są komunikaty w kolejce.
- Aby usunąć kolejkę:

- a) Wybierz z listy w widżecie kolejki kolejkę, która ma zostać usunięta.
 - b) Kliknij ikonę usuwania  na pasku narzędzi widżetu kolejki.
 - c) Opcjonalnie: Jeśli w kolejce znajdują się komunikaty, należy sprawdzić, czy kolejka może zostać wyczyszczona, klikając opcję **Wyczyść kolejkę**.
 - d) Potwierdź, że chcesz usunąć kolejkę, klikając przycisk **Usuń**. Kolejka została usunięta.
- Aby wyświetlić i edytować właściwości kolejki:
 - a) Wybierz kolejkę w widżecie kolejki.
 - b) Kliknij ikonę właściwości  na pasku narzędzi widżetu kolejki. Alternatywnie kliknij dwukrotnie kolejkę.
 - c)  Wyświetl właściwości i edytuj je zgodnie z wymaganiami. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend. Informacje na temat właściwości można znaleźć w sekcji [Właściwości kolejki](#) w dokumentacji programu MQ Explorer.
 - Aby wyświetlić i edytować rekordy uprawnień dla kolejki:
 - a) Wybierz kolejkę w widżecie.
 - b) Kliknij opcję  ... > **Zarządzanie rekordami uprawnień**.
Rekordy uprawnień przedstawiają uprawnienia, które użytkownicy i administratorzy mają w wybranej kolejce.

Praca z wątkami


Za pomocą widżetu Tematy w produkcie IBM MQ Console można dodawać i usuwać tematy oraz wyświetlać i ustawiać właściwości tematu.

Zanim rozpoczniesz


Aby można było używać widżetu, należy utworzyć widżet tematy. Więcej informacji na temat tworzenia widżetów obiektów IBM MQ zawiera sekcja ["Praca z obiektami IBM MQ"](#) na stronie 101.



O tym zadaniu

W widżecie tematy są wyświetlane tematy, które istnieją dla konkretnego menedżera kolejek. Poszczególne tematy można wybierać z listy, z którą można pracować.

 Nie można wyświetlać ani edytować rekordów uprawnień dla tematu w systemie z/OS.

Procedura

- Aby dodać temat:
 - a) Kliknij ikonę tworzenia  na pasku narzędzi widżetu tematów.
 - b) Wprowadź nazwę nowego tematu. Poprawnymi znakami są litery i cyfry oraz znaki ".", "/", "_i" "%".
 - c) Określ łańcuch tematu, który będzie publikował komunikaty dla tematu. Więcej informacji na ten temat zawiera sekcja [Właściwości tematu](#).
 - d) Kliknij opcję **Create**. Zostanie utworzony nowy temat.
- Aby usunąć temat:
 - a) Wybierz temat, który ma zostać usunięty z listy w widżecie tematów.

- b) Kliknij ikonę usuwania  na pasku narzędzi widgetu tematów.
- c) Potwierdź, że chcesz usunąć temat, klikając przycisk **Usuń**. Temat zostanie usunięty.
- Aby wyświetlić i edytować właściwości tematu, wykonaj następujące czynności:
 - a) Wybierz temat w widżecie Tematy.
 - b) Kliknij ikonę właściwości  na pasku narzędzi widgetu tematów. Alternatywnie kliknij dwukrotnie temat.
 - c)  Wyświetl właściwości i edytuj je zgodnie z wymaganiami. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend. Informacje na temat właściwości można znaleźć w sekcji [Właściwości tematu](#) w dokumentacji programu MQ Explorer.
- Aby opublikować komunikat w temacie:
 - a) Kliknij ikonę umieszczonego komunikatu  na pasku narzędzi widgetu tematów.
 - b) Wprowadź komunikat w polu **Komunikat**.
 - c) Wprowadź łańcuch tematu w celu opublikowania komunikatu w polu **Łańcuch tematu**.
 - d) Kliknij opcję **Publikuj**.
- Aby zasubskrybować temat:
 - a) Kliknij ikonę subskrybowania  na pasku narzędzi widgetu tematów.
 - b) Wprowadź łańcuch tematu, który ma zostać zasubskrybowany w polu **łańcuch tematu**.
 - c) Kliknij opcję **Subscribe**.
- Aby wyświetlić i edytować rekordy uprawnień dla tematu, wykonaj następujące czynności:
 - a) Wybierz temat w widżecie Tematy.
 - b) Kliknij opcję  ... > **Zarządzanie rekordami uprawnień**.
Rekordy uprawnień przedstawiają uprawnienia, które użytkownicy i administratorzy mają w wybranym temacie.

Praca ze nastuchiwaczami

Za pomocą widgetu nastuchiwania w produkcie IBM MQ Console można dodawać i usuwać obiekty nastuchiwania, uruchamiać i zatrzymują obiekty nastuchiwania, wyświetlać i ustawiać właściwości nastuchiwania oraz zarządzać rekordami uprawnień dla obiektu nastuchiwania.

Zanim rozpoczniesz








Aby można było używać widgetu nastuchiwania, należy utworzyć widget nastuchiwania. Więcej informacji na temat tworzenia widgetów obiektów IBM MQ zawiera sekcja [“Praca z obiektami IBM MQ”](#) na stronie 101.

O tym zadaniu

Widget Programy nastuchujące zawiera listę obiektów nastuchiwania, które istnieją dla konkretnego menedżera kolejek. Z listy można wybrać poszczególne obiekty nastuchiwania, z którymi ma pracować praca.

 Widgetu nastuchiwania nie można używać w systemie z/OS.

Procedura

- Aby dodać obiekt nasłuchiwania TCP/IP:
 - a) Kliknij ikonę tworzenia  na pasku narzędzi widgetu Programy nasłuchujące.
 - b) Wprowadź nazwę obiektu nasłuchiwania. Poprawnymi znakami są litery i cyfry oraz znaki ".", "/", "_" i "%".
 - c) Wprowadź dostępny port TCP/IP dla programu nasłuchującego. Numer portu nie może być większy niż 65535.
 - d) Kliknij opcję **Create**. Zostanie utworzony nowy obiekt nasłuchiwania.
- Aby usunąć obiekt nasłuchiwania:
 - a) Wybierz obiekt nasłuchiwania, który ma zostać usunięty z listy w widżecie Obiekty nasłuchiwania.
 - b) Kliknij ikonę usuwania  na pasku narzędzi widgetu Programy nasłuchujące.
 - c) Potwierdź, że chcesz usunąć program nasłuchujący, klikając przycisk **Usuń**. Obiekt nasłuchiwania został usunięty.
- Aby uruchomić program nasłuchujący:
 - a) Wybierz obiekt nasłuchiwania, który ma zostać uruchomiony z listy w widżecie Obiekty nasłuchiwania.
 - b) Kliknij ikonę uruchamiania  na pasku narzędzi widgetu Programy nasłuchujące.
- Aby zatrzymać proces nasłuchujący:
 - a) Wybierz obiekt nasłuchiwania, który ma zostać zatrzymany, z listy w widżecie Obiekty nasłuchiwania.
 - b) Kliknij ikonę zatrzymania  na pasku narzędzi widgetu Programy nasłuchujące.
 - c) Kliknij przycisk **Zatrzymaj**, aby potwierdzić, że nasłuchiwanie ma zostać zatrzymane.
- Aby wyświetlić i edytować właściwości obiektu nasłuchiwania:
 - a) Wybierz obiekt nasłuchiwania w widżecie Obiekty nasłuchiwania.
 - b) Kliknij ikonę właściwości  na pasku narzędzi widgetu Programy nasłuchujące. Alternatywnie kliknij dwukrotnie obiekt nasłuchiwania.
 - c)  Wyświetl właściwości i edytuj je zgodnie z wymaganiami. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend. Informacje na temat właściwości można znaleźć w sekcji [Właściwości nasłuchiwania](#) w dokumentacji programu MQ Explorer.
- Aby wyświetlić i edytować rekordy uprawnień dla obiektu nasłuchiwania:
 - a) Wybierz obiekt nasłuchiwania w widżecie Obiekty nasłuchiwania.
 - b) Kliknij opcję  ... > **Zarządzanie rekordami uprawnień**. Rekordy uprawnień przedstawiają uprawnienia, jakie mają użytkownicy i administratorzy w wybranym obiekcie nasłuchiwania.

Praca z kanałami


Widget kanałów w produkcie IBM MQ Console umożliwia dodawanie i usuwanie kanałów, uruchamianie i zatrzymywanie kanałów, resetowanie i rozstrzyganie kanałów oraz kanały ping. Użytkownik może również wyświetlać i ustawiać właściwości kanału oraz zarządzać rekordami uprawnień dla kanału.

Zanim rozpoczniesz



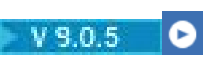
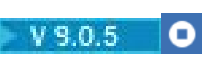


Należy utworzyć widget kanałów, zanim będzie można go użyć. Więcej informacji na temat tworzenia widgetów obiektów IBM MQ zawiera sekcja [“Praca z obiektami IBM MQ”](#) na stronie 101.





O tym zadaniu

Widget kanały wyświetla listę kanałów, które istnieją dla konkretnego menedżera kolejek. Z listy można wybrać poszczególne kanały, z którymi ma pracować praca.

 Nie można wyświetlać ani edytować rekordów uprawnień dla kanału w systemie z/OS.

Procedura

- Aby dodać kanał:
 - a) Kliknij ikonę tworzenia  na pasku narzędzi widgetu kanałów.
 - b) Wprowadź nazwę kanału. Poprawnymi znakami są litery i cyfry oraz znaki ".", "/", "_i" "%".
 - c) Wybierz typ kanału, który ma zostać dodany.
 - d) Jeśli tworzysz kanał nadawczy, nadawca klastra lub kanał requestera, podaj nazwę połączenia. Nazwa połączenia to nazwa komputera, na którym znajduje się docelowy menedżer kolejek. Format nazwy to *nazwa_komputera(numer_portu)*. *nazwa_komputera* to nazwa lub adres IP komputera, na którym znajduje się docelowy menedżer kolejek, a *numer_portu* to port używany przez proces nasłuchujący menedżera kolejek docelowych.
 - e) W przypadku tworzenia kanału nadawczego lub kanału serwera należy określić kolejkę transmisji, która odpowiada menedżerowi kolejek na końcu odbiornika kanału.
 - f) Kliknij opcję **Create**. Zostanie utworzony nowy kanał.
- Aby usunąć kanał:
 - a) Wybierz kanał, który ma zostać usunięty z listy w widżecie kanałów.
 - b) Kliknij ikonę usuwania  na pasku narzędzi widgetu.
 - c) Potwierdź, że chcesz usunąć kanał, klikając przycisk **Usuń**. Kanał zostanie usunięty.
- Aby uruchomić kanał:
 - a) Wybierz kanał, który ma zostać uruchomiony z listy w widżecie kanałów.
 - b) Kliknij ikonę uruchamiania  na pasku narzędzi widgetu.
- Aby zatrzymać kanał:
 - a) Wybierz kanał, który ma zostać zatrzymany z listy w widżecie kanałów.
 - b) Kliknij ikonę zatrzymania  na pasku narzędzi widgetu.
 - c) Potwierdź, że chcesz zatrzymać kanał, klikając przycisk **Zatrzymaj**.
- Aby wyświetlić właściwości kanału:
 - a) Wybierz kanał w widżecie kanałów.
 - b) Kliknij ikonę właściwości  na pasku narzędzi widgetu kanały. Alternatywnie kliknij dwukrotnie kanał.
 - c) 
Wyświetl właściwości i edytuj je zgodnie z wymaganiami. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend. Informacje na temat właściwości można znaleźć w sekcji [Właściwości kanału](#) w dokumentacji programu MQ Explorer.

- Aby zresetować kanał:
 - a) Wybierz kanał w widżecie kanałów.
 - b) Kliknij opcję  ... > **Resetuj**.
 - c) Określ numer kolejny komunikatu.
Zresetuj kanał, jeśli nie zostanie on uruchomiony, ponieważ dwa końce nie zgadzają się co do numeru kolejnego następnego komunikatu do wysłania. Numer kolejny komunikatu określa ten numer.
 - d) Kliknij opcję **Resetuj kanał**.
- Aby rozstrzygnąć kanał:
 - a) Wybierz kanał w widżecie kanałów.
 - b) Kliknij opcję  ... > **Rozstrzygnij**.
 - c) Wybierz, czy chcesz zatwierdzić, czy wycofać bieżącą partię komunikatów, klikając opcję **Zatwierdź** lub **Cofnij wycofanie**.
- Aby wykonać komendę ping dla kanału:
 - a) Wybierz kanał w widżecie kanałów.
 - b) Kliknij opcję  ... > **Ping**.
- Aby wyświetlić lub edytować rekordy uprawnień dla kanału:
 - a) Wybierz kanał w widżecie.
 - b) Kliknij opcję  ... > **Zarządzanie rekordami uprawnień**.
Rekordy uprawnień przedstawiają uprawnienia, które użytkownicy i administratorzy mają w wybranym kanale.

Praca z kanałami połączeń klientów


Za pomocą widżetu kanałów połączenia klienckiego w produkcie IBM MQ Console można dodawać i usuwać kanały połączenia z klientem w menedżerze kolejek, wyświetlać i ustawiać właściwości oraz zarządzać rekordami uprawnień dla kanału.

Zanim rozpocznie



Przed użyciem widżetu kanały połączenia klienckiego należy utworzyć widżet kanały połączenia klienckiego. Więcej informacji na temat tworzenia widżetów obiektów IBM MQ zawiera sekcja [“Praca z obiektami IBM MQ”](#) na stronie 101.

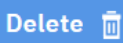
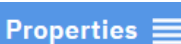


O tym zadaniu

Widżet kanały połączenia klienckiego zawiera listę kanałów połączenia klienckiego, które istnieją dla konkretnego menedżera kolejek. Z listy można wybrać pojedyncze kanały połączenia klienckiego, z którymi ma pracować praca.

 Nie można wyświetlać ani edytować rekordów uprawnień dla kanału połączenia klienckiego w systemie z/OS.

Procedura

- Aby dodać kanał połączenia klienckiego:
 - a) Kliknij ikonę tworzenia   na pasku narzędzi widżetu kanały połączenia klienckiego.
 - b) Wprowadź nazwę kanału połączenia klienckiego. Poprawnymi znakami są litery i cyfry oraz znaki " ", "/", "_i "%".

- c) Podaj nazwę połączenia. Nazwa połączenia to nazwa komputera, na którym znajduje się docelowy menedżer kolejek. Format to *nazwa_komputera(numer_portu)*, gdzie *nazwa_komputera* jest nazwą lub adresem IP komputera udostępniającego docelowy menedżer kolejek, a *numer_portu* jest portem, z którego korzysta proces nasłuchujący menedżera kolejek docelowych.
- d) Kliknij opcję **Create**. Zostanie utworzony nowy kanał połączenia klienckiego.
- Aby usunąć kanał połączenia z klientem:
 - a) Wybierz z listy kanał połączenia klienckiego, który ma zostać usunięty, w widżecie kanałów połączenia klienckiego.
 - b) Kliknij ikonę usuwania  na pasku narzędzi widżetu.
 - c) Potwierdź, że chcesz usunąć kanał połączenia klienckiego, klikając przycisk **Usuń**. Kanał połączenia klienckiego został usunięty.
- Aby wyświetlić i edytować właściwości kanału połączenia klienckiego:
 - a) Wybierz kanał połączenia klienckiego w widżecie kanałów połączenia klienckiego.
 - b) Kliknij ikonę właściwości  na pasku narzędzi widżetu kanału połączenia klienckiego. Alternatywnie kliknij dwukrotnie kanał połączenia klienckiego.
 - c)  Wyświetl właściwości i edytuj je zgodnie z wymaganiami. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend. Informacje na temat właściwości można znaleźć w sekcji [Właściwości kanału](#) w dokumentacji programu MQ Explorer.
- Aby wyświetlić i edytować rekordy uprawnień dla kanału połączenia klienckiego:
 - a) Wybierz kanał połączenia klienckiego w widżecie kanałów połączenia klienckiego.
 - b) Kliknij opcję  ... > **Zarządzaj rekordami uprawnień**. Rekordy uprawnień przedstawiają uprawnienia, które użytkownicy i administratorzy mają na wybranym kanale połączenia klienckiego.

Praca z informacjami uwierzytelniających

Za pomocą widżetu Informacje o uwierzytelnianiu w produkcie IBM MQ Console można dodawać i usuwać obiekty informacji uwierzytelniających w menedżerze kolejek. Możliwe jest również wyświetlanie i ustawianie właściwości oraz zarządzanie rekordami uprawnień dla obiektów.


Zanim rozpoczniesz

Przed użyciem widżetu Informacje o uwierzytelnianiu należy utworzyć widżet informacji uwierzytelniających. Więcej informacji na temat tworzenia widżetów obiektów IBM MQ zawiera sekcja [“Praca z obiektami IBM MQ” na stronie 101](#).





O tym zadaniu

Widżet Informacje o uwierzytelnianiu zawiera listę informacji uwierzytelniających, które istnieją dla konkretnego menedżera kolejek. Z listy można wybrać pojedyncze informacje uwierzytelniania, z którymi ma pracować praca.

Informacje o uwierzytelnianiu menedżera kolejek są częścią obsługi protokołu TLS (Transport Layer Security-TLS) produktu IBM MQ. Obiekty te zawierają definicje wymagane do sprawdzania odwołań certyfikatów przy użyciu protokołu OCSP lub listy odwołań certyfikatów (CRL) na serwerach LDAP, a także definicje wymagane do włączenia sprawdzania ID użytkownika i hasła.

 Nie można używać protokołu LDAP IDPW ani wyświetlać ani edytować rekordów uprawnień dla obiektu informacji uwierzytelniających w systemie z/OS.

Procedura

- Aby dodać obiekt informacji uwierzytelniającej:
 - a) Kliknij ikonę tworzenia  na pasku narzędzi widgetu Informacje o uwierzytelnianiu.
 - b) Określ nazwę obiektu informacji uwierzytelniającej. Poprawnymi znakami są litery i cyfry oraz znaki ".", "/", "_", "%".
 - c) Określ typ obiektu informacji uwierzytelniającej.
 - d) Określ dodatkowe informacje odpowiednie dla typu obiektu:
 - W polu **CRL LDAP** (Nazwa serwera LDAP) podaj **nazwę serwera LDAP**. Ta nazwa to nazwa hosta, adres IPv4 w postaci dziesiętnej z kropkami lub zapis szesnastkowy IPv6 hosta, na którym działa serwer LDAP, z opcjonalnym numerem portu.
 - W przypadku produktu **OCSP** należy określić adres URL **OCSP responder URL**. Ten adres URL jest adresem URL respondera, który jest używany do sprawdzania odwołania certyfikatu. Ta wartość musi być adresem URL HTTP zawierającym nazwę hosta i numer portu respondera OCSP. Jeśli moduł odpowiadający OCSP używa portu 80, który jest wartością domyślną dla protokołu HTTP, numer portu może zostać pominięty. Adresy URL HTTP są zdefiniowane w dokumencie RFC 1738.
 - W przypadku systemu **IDPW OS** nie ma dodatkowych wymagań.
 - W polu **IDPW LDAP** podaj **nazwę serwera LDAP** i nazwę **Skrócona nazwa użytkownika**. Nazwa serwera LDAP to nazwa hosta, adres IPv4 w postaci dziesiętnej z kropkami lub zapis szesnastkowy IPv6 hosta, na którym działa serwer LDAP, z opcjonalnym numerem portu. Skrócona nazwa użytkownika to pole w rekordzie użytkownika LDAP, które jest używane jako skrócona nazwa połączenia.
 - e) Kliknij opcję **Create**.
- Aby usunąć obiekt informacji uwierzytelniającej:
 - a) Z listy w widżecie wybierz obiekt informacji uwierzytelniającej, który ma zostać usunięty.
 - b) Kliknij ikonę usuwania  na pasku narzędzi widgetu.
 - c) Potwierdź, że chcesz usunąć obiekt informacji uwierzytelniającej, klikając przycisk **Usuń**. Obiekt został usunięty.
- Aby wyświetlić i edytować właściwości obiektu informacji uwierzytelniającej:
 - a) Wybierz obiekt informacji uwierzytelniającej w widżecie.
 - b) Kliknij ikonę właściwości  na pasku narzędzi widgetu. Alternatywnie kliknij dwukrotnie obiekt informacji uwierzytelniającej.
 - c) Wyświetl właściwości i edytuj je zgodnie z wymaganiami. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend.
- Aby wyświetlić i edytować rekordy uprawnień dla obiektu informacji uwierzytelniającej:
 - a) Wybierz obiekt informacji uwierzytelniającej w widżecie Informacje o uwierzytelnianiu.
 - b) Kliknij opcję  **> Zarządzaj rekordami uprawnień**. Rekordy uprawnień przedstawiają uprawnienia, które użytkownicy i administratorzy mają na wybranym obiekcie informacji uwierzytelniających.

V 9.0.1 Praca z subskrypcjami

Za pomocą widgetu subskrypcje w produkcie IBM MQ Console można dodawać i usuwać subskrypcje menedżera kolejek, wyświetlać i ustawiać właściwości oraz zarządzać rekordami uprawnień dla subskrypcji.

Zanim rozpoczniesz

Przed użyciem widgetu subskrypcji należy utworzyć widget subskrypcji. Więcej informacji na temat tworzenia widgetów obiektów IBM MQ zawiera sekcja [“Praca z obiektami IBM MQ” na stronie 101](#).



O tym zadaniu

Subskrypcje są wysyłane do menedżera kolejek i zawierają informacje o publikacjach, które subskrybent chce otrzymywać:

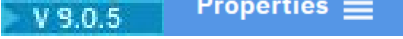
- Łańcuch tematu, który jest zainteresowany subskrybentem. Ten temat może zostać przetłumaczony na wiele łańcuchów tematów, jeśli używane są znaki wieloznaczne.
- Opcjonalny łańcuch wyboru, który ma być stosowany do opublikowanych komunikatów.
- Nazwa kolejki, w której umieszczane są wybrane publikacje.

Więcej informacji na temat subskrypcji zawiera sekcja [Subskrybenci i subskrypcje](#) oraz sekcja [DEFINIOWANIE SUBSKRYPCJI PODRZĘDNEJ](#).

Procedura

- Aby dodać obiekt subskrypcji:
 - a) Kliknij ikonę tworzenia  na pasku narzędzi widgetu subskrypcji.
 - b) Określ nazwę obiektu. Poprawnymi znakami są litery i cyfry oraz znaki ".", "/", "_" i "%".
 - c) Wybierz opcję **Klasa docelowa** w obszarze **Zarządzane** lub **Udostępnione**. Jeśli zostanie wybrana opcja **Zarządzane**, miejsce docelowe zostanie utworzone w lokalnym menedżerze kolejek.
 - d) Jeśli w polu **Miejsce docelowe** zostanie wybrana klasa docelowa **Opatrzona**, należy określić nazwę kolejki, do której przekazywane są komunikaty dla tej subskrypcji.
 - e) W polu **Łańcuch tematu** określ łańcuch tematu, który ma zostać zasubskrybowany.
 - f) Wybierz ustawienie **Użycie znaku wieloznacznego**. Wybierz opcję **Znak wieloznaczny na poziomie znaku**, aby określić, że znaki wieloznaczne reprezentują części łańcuchów. Wybierz opcję **Znak wieloznaczny poziomu tematu**, aby określić, że znaki wieloznaczne reprezentują części hierarchii tematów.
 - g) Wybierz opcję **Zasięg**. Wybierz opcję **Wszystkie**, aby subskrypcja była przekazywana do wszystkich menedżerów kolejek bezpośrednio połączonych za pośrednictwem kolektywu lub hierarchii publikowania/subskrypcji. Wybierz opcję **Menedżer kolejek** w taki sposób, aby subskrypcje komunikatów były publikowane w tym menedżerze kolejek tylko w tym menedżerze kolejek.
 - h) Opcjonalne: Określ **Selektor**. Łańcuch wyboru jest wyrażeniem, które jest stosowane do publikacji w celu określenia, czy jest ona zgodna z subskrypcją.
 - i) Kliknij opcję **Create**.
- Aby usunąć obiekt subskrypcji:
 - a) Wybierz z listy obiekt subskrypcji, który ma zostać usunięty z listy w widżecie subskrypcji.
 - b) Kliknij ikonę usuwania  na pasku narzędzi widgetu.
 - c) Potwierdź, że chcesz usunąć obiekt subskrypcji, klikając przycisk **Usuń**. Obiekt został usunięty.
- Aby wyświetlić i edytować właściwości obiektu subskrypcji:

a) Wybierz obiekt subskrypcji w widżecie.

b) Kliknij ikonę właściwości  na pasku narzędzi widżetu. Alternatywnie kliknij dwukrotnie obiekt subskrypcji.

c) Wyświetl właściwości i edytuj je zgodnie z wymaganiami. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend.

Praca z rekordami uwierzytelniania kanału

Za pomocą widżetu Rekordy uwierzytelniania kanału w produkcie IBM MQ Console można dodawać i usuwać rekordy uwierzytelniania kanału w menedżerze kolejek. Istnieje również możliwość wyświetlenia i ustawienia właściwości dla rekordów uwierzytelniania kanału.

Zanim rozpocznieš


Aby można było używać widżetu, należy utworzyć widżet rekordów uwierzytelniania kanału. Więcej informacji na temat tworzenia widżetów obiektów IBM MQ zawiera sekcja [“Praca z obiektami IBM MQ”](#) na stronie 101.


O tym zadaniu

Aby wykonać bardziej precyzyjną kontrolę nad dostępem, który jest nadawany do łączenia systemów na poziomie kanału, można użyć rekordów uwierzytelniania kanału.

Aby wymusić ochronę, można użyć blokujących rekordów uwierzytelniania kanału w celu zablokowania dostępu do kanałów. Można również użyć rekordów uwierzytelniania kanału odwzorowania adresów, aby umożliwić dostęp do określonych użytkowników. Więcej informacji na temat rekordów uwierzytelniania kanału znajduje się w sekcji [Rekordy uwierzytelniania kanału](#).

Procedura

- Aby dodać rekord uwierzytelniania kanału z tożsamością nazwy wyróżniającej SSL/TLS, należy zapoznać się z [“Tworzenie rekordów uwierzytelniania kanału przy użyciu tożsamości nazwy wyróżniającej SSL/TLS”](#) na stronie 112.
- Aby dodać rekord uwierzytelniania kanału z tożsamością identyfikatora użytkownika aplikacji klienckiej, należy zapoznać się z [“Tworzenie rekordów uwierzytelniania kanału z tożsamością identyfikatora użytkownika aplikacji klienckiej”](#) na stronie 113.
- Aby dodać rekord uwierzytelniania kanału ze zdalną tożsamością nazwy menedżera kolejek, należy zapoznać się z [“Tworzenie rekordów uwierzytelniania kanału ze zdalną tożsamością nazwy menedżera kolejek”](#) na stronie 114.
- Informacje na temat dodawania rekordu uwierzytelniania kanału z tożsamością adresu zawiera sekcja [“Tworzenie rekordów uwierzytelniania kanału z tożsamością adresu IP”](#) na stronie 115.
- Aby usunąć rekord uwierzytelniania kanału:
 - a) Wybierz z listy rekord uwierzytelniania kanału, który ma zostać usunięty, w widżecie Rekordy uwierzytelniania kanału.
 - b) Kliknij ikonę usuwania  na pasku narzędzi widżetu.
 - c) Potwierdź, że chcesz usunąć rekord uwierzytelniania kanału, klikając przycisk **Usuń**. Rekord uwierzytelniania kanału zostanie usunięty.
- Aby wyświetlić i edytować właściwości rekordu uwierzytelniania kanału:
 - a) Wybierz z listy rekord uwierzytelniania kanału, który ma być edytowany, w widżecie rekordu uwierzytelniania kanału.

- b) Kliknij ikonę właściwości  na pasku narzędzi widgetu. Alternatywnie kliknij dwukrotnie rekord uwierzytelniania kanału.
- c) Wyświetl właściwości i edytuj je zgodnie z wymaganiami. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend.


Tworzenie rekordów uwierzytelniania kanału przy użyciu tożsamości nazwy wyróżniającej SSL/TLS

Widget rekordów uwierzytelniania kanału umożliwia tworzenie, blokowanie i ostrzeganie rekordów uwierzytelniania kanału z tożsamością nazwy wyróżniającej SSL/TLS. Tożsamość wyróżniająca SSL/TLS jest zgodna z tożsamościami użytkowników, którzy prezentują certyfikat osobisty SSL lub TLS, który zawiera określoną nazwę wyróżniającą.

Zanim rozpocznie

Aby można było używać widgetu, należy utworzyć widget rekordów uwierzytelniania kanału. Więcej informacji na temat tworzenia widgetów obiektów IBM MQ zawiera sekcja [“Praca z obiektami IBM MQ” na stronie 101](#).

Procedura

- Aby dodać rekord uwierzytelniania kanału:
 - a) Kliknij ikonę tworzenia  na pasku narzędzi widgetu rekordu uwierzytelniania kanału.
 - b) Wybierz opcję **Typ reguły** , aby wskazać, jaki typ reguły ma być używany w rekordzie uwierzytelniania kanału:
 - Wybierz opcję **Zezwól** , aby zezwolić na dostęp do połączeń przychodzących.
 - Wybierz opcję **Blok** , aby zablokować dostęp do połączeń przychodzących.
 - Wybierz opcję **Warn** (Warn), aby ostrzec o dostępie do połączeń przychodzących, które zostałyby zablokowane. Połączenie jest dozwolone i zgłaszany jest komunikat o błędzie. Jeśli zdarzenia są skonfigurowane, tworzony jest komunikat o zdarzeniu, który przedstawia szczegółowe informacje na temat tego, co jest zablokowane. Zgłaszane są tylko zgodne reguły.
 - c) Wybierz z listy typ tożsamości **Nazwa wyróżniająca SSL/TLS** .
 - d) Kliknij przycisk **Dalej**.
 - e) Określ **Profil kanału**.

Profil kanału to nazwa kanału lub zestawu kanałów, dla których jest ustawiane uwierzytelnianie kanału. Profil może zawierać znaki wieloznaczne, aby można było zablokować zakres kanałów. Na przykład profil `a1phade1ta*` blokuje kanały o nazwach `alphadelta1`, `alphadelta2`, `alphdelta3` i tak dalej.
 - f) Podaj **nazwę węzła sieci**. Na przykład: `CN=John Smith, O=IBM ,OU=Test , C=GB`. Więcej informacji na temat nazw węzłów sieci można znaleźć w sekcji [Reguły WebSphere MQ dla wartości SSLPEER](#) .
 - g) Opcjonalne: Określ używany filtr **Adres** . Adres IP jest adresem IP, który jest oczekiwany na drugim końcu kanału.
 - h) Opcjonalne: Podaj **nazwę wystawcy certyfikatów SSL**. Nazwa wystawcy certyfikatu SSL to nazwa ośrodka certyfikacji, który musi być wystawiony przez certyfikat SSL/TLS.
 - i) Opcjonalne: Kliknij przycisk **Dalej**.
 - j) Opcjonalne: W przypadku typu reguły **Zezwalaj** można opcjonalnie określić **Źródło użytkownika** dla rekordu uwierzytelniania kanału. Źródło użytkownika określa źródło identyfikatora użytkownika, który jest używany, gdy połączenie przychodzące jest zgodne z nazwą wyróżniającą SSL/TLS.

- Opcja **Kanał** określa, że połączenia przychodzące, które są zgodne z odwzorowaniem, używają ID użytkownika z przepływem lub dowolnego użytkownika zdefiniowanego na obiekcie kanału.
 - Opcja **Mapa** określa, że połączenia przychodzące, które są zgodne z odwzorowaniem, używają identyfikatora użytkownika określonego w polu **ID użytkownika MCA**.
- k) Opcjonalne: Kliknij przycisk **Dalej**.
- l) Opcjonalne: W polu **Opis** podaj opis dla rekordu uwierzytelniania kanału.
- m) Kliknij opcję **Create**. Zostanie utworzony nowy rekord uwierzytelniania kanału.

V 9.0.1 Tworzenie rekordów uwierzytelniania kanału z tożsamością identyfikatora użytkownika aplikacji klienckiej


Widget rekordów uwierzytelniania kanału umożliwia tworzenie, blokowanie i ostrzeganie rekordów uwierzytelniania kanału z tożsamością identyfikatora użytkownika aplikacji klienckiej. Tożsamość identyfikatora użytkownika aplikacji klienckiej jest zgodna z identyfikatorami aplikacji klienckich z kanału połączenia klienckiego.

Zanim rozpocznie

Aby można było używać widgetu, należy utworzyć widget rekordów uwierzytelniania kanału. Więcej informacji na temat tworzenia widgetów obiektów IBM MQ zawiera sekcja [“Praca z obiektami IBM MQ”](#) na stronie 101.

Procedura

- Aby dodać rekord uwierzytelniania kanału:

- a) Kliknij ikonę tworzenia  na pasku narzędzi widgetu rekordu uwierzytelniania kanału.
- b) Wybierz opcję **Typ reguły**, aby wskazać, jaki typ reguły ma być używany w rekordzie uwierzytelniania kanału:
 - Wybierz opcję **Zezwól**, aby zezwolić na dostęp do połączeń przychodzących.
 - Wybierz opcję **Blok**, aby zablokować dostęp do połączeń przychodzących.
 - Wybierz opcję **Warn** (Warn), aby ostrzec o dostępie do połączeń przychodzących, które zostałyby zablokowane. Połączenie jest dozwolone i zgłaszany jest komunikat o błędzie. Jeśli zdarzenia są skonfigurowane, tworzony jest komunikat o zdarzeniu, który przedstawia szczegółowe informacje na temat tego, co jest zablokowane. Zgłaszane są tylko zgodne reguły.
- c) Wybierz z listy typ tożsamości **ID użytkownika aplikacji klienckiej**.
- d) Kliknij przycisk **Dalej**.
- e) Określ **Profil kanału**.
Nazwa profilu kanału to nazwa kanału lub zestawu kanałów, dla których ustawiane jest uwierzytelnianie kanału. Profil może zawierać znaki wieloznaczne, aby można było zablokować zakres kanałów. Na przykład profil `a1phade1ta*` blokuje kanały o nazwach `alphadelta1`, `alphadelta2`, `alphadelta3` i tak dalej.
- f) Podaj wartość w polu **Identyfikator użytkownika klienta**. Identyfikator użytkownika klienta to identyfikator użytkownika klienta, który ma być dopuszczony, blokowany lub ostrzegany o tym identyfikatorze.
- g) Opcjonalne: Określ używany filtr **Adres**. Adres IP jest adresem IP, który jest oczekiwany na drugim końcu kanału.
- h) Opcjonalne: Kliknij przycisk **Dalej**.
- i) Opcjonalne: W przypadku typu reguły **Zezwalaj** można opcjonalnie określić **Źródło użytkownika** dla rekordu uwierzytelniania kanału. Źródło użytkownika określa źródło identyfikatora użytkownika, który jest używany, gdy połączenie przychodzące jest zgodne z identyfikatorem użytkownika klienta.

- Opcja **Kanał** określa, że połączenia przychodzące, które są zgodne z odwzorowaniem, używają ID użytkownika z przepływem lub dowolnego użytkownika zdefiniowanego na obiekcie kanału.
 - Opcja **Mapa** określa, że połączenia przychodzące, które są zgodne z odwzorowaniem, używają identyfikatora użytkownika określonego w polu **ID użytkownika MCA**.
- j) Opcjonalne: Kliknij przycisk **Dalej**.
- k) Opcjonalne: W polu **Opis** podaj opis dla rekordu uwierzytelniania kanału.
- l) Kliknij opcję **Create**. Zostanie utworzony nowy rekord uwierzytelniania kanału.


V 9.0.1 Tworzenie rekordów uwierzytelniania kanału ze zdalną tożsamością nazwy menedżera kolejek

Widget rekordów uwierzytelniania kanału umożliwia tworzenie, blokowanie i ostrzeganie rekordów uwierzytelniania kanału ze zdalną tożsamością nazwy menedżera kolejek. Tożsamość nazwy menedżera kolejek zdalnych jest zgodna z określonym menedżerem kolejek.

Zanim rozpocznieš

Aby można było używać widgetu, należy utworzyć widget rekordów uwierzytelniania kanału. Więcej informacji na temat tworzenia widgetów obiektów IBM MQ zawiera sekcja [“Praca z obiektami IBM MQ”](#) na stronie 101.

Procedura

- Aby dodać rekord uwierzytelniania kanału:
 - a) Kliknij ikonę tworzenia  na pasku narzędzi widgetu rekordu uwierzytelniania kanału.
 - b) Wybierz opcję **Typ reguły**, aby wskazać, jaki typ reguły ma być używany w rekordzie uwierzytelniania kanału:
 - Wybierz opcję **Zezwól**, aby zezwolić na dostęp do połączeń przychodzących.
 - Wybierz opcję **Blok**, aby zablokować dostęp do połączeń przychodzących.
 - Wybierz opcję **Warn** (Warn), aby ostrzec o dostępie do połączeń przychodzących, które zostałyby zablokowane. Połączenie jest dozwolone i zgłaszany jest komunikat o błędzie. Jeśli zdarzenia są skonfigurowane, tworzony jest komunikat o zdarzeniu, który przedstawia szczegółowe informacje na temat tego, co jest zablokowane. Zgłaszane są tylko zgodne reguły.
 - c) Z listy wybierz typ tożsamości **Nazwa zdalnego menedżera kolejek**.
 - d) Kliknij przycisk **Dalej**.
 - e) Podaj wartość w polu **Nazwa profilu**.
Nazwa profilu to nazwa kanału lub zestawu kanałów, dla których jest ustawiane uwierzytelnianie kanału. Profil może zawierać znaki wieloznaczne, aby można było zablokować zakres kanałów. Na przykład profil `a1phade1ta*` blokuje kanały o nazwach `alphadelta1`, `alphadelta2`, `alphdelta3` i tak dalej.
 - f) Podaj wartość w polu **Nazwa menedżera kolejek**. Nazwa menedżera kolejek określa nazwę zdalnego menedżera kolejek, który ma być dopuszczony, blokowany lub ostrzegany.
 - g) Opcjonalne: Określ używany filtr **Adres**. Adres IP jest adresem IP, który jest oczekiwany na drugim końcu kanału.
 - h) Opcjonalne: Kliknij przycisk **Dalej**.
 - i) Opcjonalne: W przypadku typu reguły **Zezwalaj** można opcjonalnie określić **Źródło użytkownika** dla rekordu uwierzytelniania kanału. Źródło użytkownika określa źródło identyfikatora użytkownika, który jest używany, gdy połączenie przychodzące jest zgodne z nazwą zdalnego menedżera kolejek.
 - Opcja **Kanał** określa, że połączenia przychodzące, które są zgodne z odwzorowaniem, używają ID użytkownika z przepływem lub dowolnego użytkownika zdefiniowanego na obiekcie kanału.

- Opcja **Mapa** określa, że połączenia przychodzące, które są zgodne z odwzorowaniem, używają identyfikatora użytkownika określonego w polu **ID użytkownika MCA**.
- j) Opcjonalne: Kliknij przycisk **Dalej**.
- k) Opcjonalne: W polu **Opis** podaj opis dla rekordu uwierzytelniania kanału.
- l) Kliknij opcję **Create**. Zostanie utworzony nowy rekord uwierzytelniania kanału.


V 9.0.1 Tworzenie rekordów uwierzytelniania kanału z tożsamością adresu IP

Widget rekordów uwierzytelniania kanału umożliwia tworzenie, blokowanie i ostrzeganie rekordów uwierzytelniania kanału z tożsamością adresu. Tożsamość adresu jest zgodna z konkretnymi adresami IP.

Zanim rozpocznieš

Aby można było używać widgetu, należy utworzyć widget rekordów uwierzytelniania kanału. Więcej informacji na temat tworzenia widgetów obiektów IBM MQ zawiera sekcja [“Praca z obiektami IBM MQ”](#) na stronie 101.

Procedura

- Aby dodać rekord uwierzytelniania kanału:
 - a) Kliknij ikonę tworzenia  na pasku narzędzi widgetu rekordu uwierzytelniania kanału.
 - b) Wybierz opcję **Typ reguły**, aby wskazać, jaki typ reguły ma być używany w rekordzie uwierzytelniania kanału:
 - Wybierz opcję **Zezwól**, aby zezwolić na dostęp do połączeń przychodzących.
 - Wybierz opcję **Blok**, aby zablokować dostęp do połączeń przychodzących.
 - Wybierz opcję **Warn** (Warn), aby ostrzec o dostępie do połączeń przychodzących, które zostałyby zablokowane. Połączenie jest dozwolone i zgłaszany jest komunikat o błędzie. Jeśli zdarzenia są skonfigurowane, tworzony jest komunikat o zdarzeniu, który przedstawia szczegółowe informacje na temat tego, co jest zablokowane. Zgłaszane są tylko zgodne reguły.
 - c) Wybierz z listy typ tożsamości **Adres**.
 - d) Kliknij przycisk **Dalej**.
 - e) Opcjonalne: W przypadku reguły typu **Blok** lub **Warn** określ wartość **Kiedy dopasować**. Można wybrać jedną z następujących opcji:
 - **W programie nasłuchującym**. Ta opcja próbuje dopasować regułę do obiektu nasłuchiwanego.
 - **W kanale**. Ta opcja próbuje dopasować regułę w kanale.
 - f) Podaj wartość w polu **Nazwa profilu**.
Nazwa profilu to nazwa kanału lub zestawu kanałów, dla których jest ustawiane uwierzytelnianie kanału. Profil może zawierać znaki wieloznaczne, aby można było zablokować zakres kanałów. Na przykład profil `a1phade1ta*` blokuje kanały o nazwach `alphadelta1`, `alphadelta2`, `alphdelta3` i tak dalej.
 - g) Podaj wartość w polu **Adres**. Adres jest adresem IP lub listą rozdzielonych przecinkami adresów IP, które są dozwolone lub zablokowane.
 - h) Opcjonalne: Kliknij przycisk **Dalej**.
 - i) Opcjonalne: W przypadku typu reguły **Zezwalaj** można opcjonalnie określić **Źródło użytkownika** dla rekordu uwierzytelniania kanału. Źródło użytkownika określa źródło identyfikatora użytkownika, który jest używany, gdy połączenie przychodzące jest zgodne z nazwą zdalnego menedżera kolejek.
 - Opcja **Kanał** określa, że połączenia przychodzące, które są zgodne z odwzorowaniem, używają ID użytkownika z przepływem lub dowolnego użytkownika zdefiniowanego na obiekcie kanału.

- Opcja **Mapa** określa, że połączenia przychodzące, które są zgodne z odwzorowaniem, używają identyfikatora użytkownika określonego w polu **ID użytkownika MCA** .
- j) Opcjonalne: Kliknij przycisk **Dalej**.
- k) Opcjonalne: W polu **Opis** podaj opis dla rekordu uwierzytelniania kanału.
- l) Kliknij opcję **Create**. Zostanie utworzony nowy rekord uwierzytelniania kanału.


V 9.0.1 Tworzenie rekordów uwierzytelniania kanału przy użyciu końcowego identyfikatora przypisanego identyfikatora użytkownika

Za pomocą widgetu rekordy uwierzytelniania kanału można tworzyć blokujące i ostrzegawcze rekordy uwierzytelniania kanału z identyfikatorem przypisanym do końcowego identyfikatora użytkownika. Końcowy przypisany identyfikator ID użytkownika jest zgodny z listą określonych identyfikatorów użytkowników z kanału serwera.

Zanim rozpoczniesz

Aby można było używać widgetu, należy utworzyć widget rekordów uwierzytelniania kanału. Więcej informacji na temat tworzenia widgetów obiektów IBM MQ zawiera sekcja [“Praca z obiektami IBM MQ”](#) na stronie 101.

Procedura

- Aby dodać rekord uwierzytelniania kanału:
 - a) Kliknij ikonę tworzenia  na pasku narzędzi widgetu rekordu uwierzytelniania kanału.
 - b) Wybierz opcję **Typ reguły** , aby wskazać, jaki typ reguły ma być używany w rekordzie uwierzytelniania kanału:
 - Wybierz opcję **Blok** , aby zablokować dostęp do połączeń przychodzących.
 - Wybierz opcję **Warn** (Warn), aby ostrzec o dostępie do połączeń przychodzących, które zostałyby zablokowane. Połączenie jest dozwolone i zgłaszany jest komunikat o błędzie. Jeśli zdarzenia są skonfigurowane, tworzony jest komunikat o zdarzeniu, który przedstawia szczegółowe informacje na temat tego, co jest zablokowane. Zgłaszane są tylko zgodne reguły.
 - c) Wybierz z listy typ tożsamości **Identyfikator przypisany do końcowego użytkownika** .
 - d) Kliknij przycisk **Dalej**.
 - e) Podaj wartość w polu **Nazwa profilu**.
Nazwa profilu to nazwa kanału lub zestawu kanałów, dla których jest ustawiane uwierzytelnianie kanału. Profil może zawierać znaki wieloznaczne, aby można było zablokować zakres kanałów. Na przykład profil `a1phade1ta*` blokuje kanały o nazwach `alphadelta1`, `alphadelta2`, `alphdelta3` i tak dalej.
 - f) Wybierz opcję **Lista użytkowników**. Lista użytkowników to rozdzielona przecinkami lista identyfikatorów użytkowników, które mają być blokowane z kanału.
 - g) Opcjonalne: Kliknij przycisk **Dalej**.
 - h) Opcjonalne: W polu **Opis** podaj opis dla rekordu uwierzytelniania kanału.
 - i) Kliknij opcję **Create**. Zostanie utworzony nowy rekord uwierzytelniania kanału.

V 9.0.1 Praca z rekordami uprawnień

Użytkownik może kontrolować dostęp grup do menedżerów kolejek i obiektów IBM MQ , określając rekord uprawnień dla tej grupy.

O tym zadaniu

Użytkownik może precyzyjnie dostosować dostęp, jaki grupa użytkowników przesyłania komunikatów ma do konkretnego menedżera kolejek lub obiektu IBM MQ, korzystając z rekordów uprawnień. Rekord uprawnień należy skonfigurować w ten sam sposób dla wszystkich typów obiektów przy użyciu tej samej procedury, chociaż rzeczywiste uprawnienia, które zostały skonfigurowane, zależą od typu obiektu.

Na przykład można porównać różne uprawnienia, które są dostępne dla menedżera kolejek i kolejki, tak jak przedstawiono to w następujących obrazach: **V 9.0.5**

Authority records for 'qm3'

Entity name	Entity type
mqm	Group
mqsystem	User


Total: 2 Last updated: 3:40:18 PM

Administration	Context	MQI
<input checked="" type="checkbox"/> Change	<input checked="" type="checkbox"/> Set all context	<input checked="" type="checkbox"/> Alternate user authority
<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Set identity context	<input checked="" type="checkbox"/> Connect
<input checked="" type="checkbox"/> Display		<input checked="" type="checkbox"/> Inquire
<input checked="" type="checkbox"/> Ctrl		<input checked="" type="checkbox"/> Set
		<input checked="" type="checkbox"/> System

Check all Uncheck all Close Save

V 9.0.5

Authority records for 'q1' on qm3

Delete  1 item selected Cancel	
▲ Entity name	Entity type
mqm	Group
mqsystem	User

Administration	Context	MQI
<input checked="" type="checkbox"/> Change	<input checked="" type="checkbox"/> Pass all context	<input checked="" type="checkbox"/> Browse
<input checked="" type="checkbox"/> Clear	<input checked="" type="checkbox"/> Pass identity context	<input checked="" type="checkbox"/> Inquire
<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Set all context	<input checked="" type="checkbox"/> Get
<input checked="" type="checkbox"/> Display	<input checked="" type="checkbox"/> Set identity context	<input checked="" type="checkbox"/> Put
		<input checked="" type="checkbox"/> Set

[Check all](#) [Uncheck all](#) [Close](#) [Save](#)



Nie można pracować z rekordami uprawnień w systemie z/OS.

Procedura

- Aby wyświetlić lub edytować rekord uprawnień dla obiektu IBM MQ , wykonaj następujące czynności:
 - a) Wybierz obiekt w widżecie na panelu kontrolnym. Powiązany menedżer kolejek musi być uruchomiony.
 - b) Z poziomu odpowiedniego paska narzędzi widżetu wybierz opcję ... > **Zarządzaj rekordami uprawnień**.
 - c) Wybierz grupę, dla której ma zostać wyświetlony rekord uprawnień. Zostaną wyświetlone uprawnienia dla tej grupy.
 - d) W razie potrzeby wybierz odpowiednie uprawnienia. W zależności od typu obiektu, dla którego tworzony jest rekord uprawnień, dostępne są różne uprawnienia.
 - e) Kliknij przycisk **Save** (Zapisz).
- Aby wyświetlić lub edytować rekord uprawnień do tworzenia dla menedżera kolejek:
 - a) Wybierz menedżer kolejek w widżecie menedżera kolejek na panelu kontrolnym. Menedżer kolejek musi być uruchomiony.
 - b) Z paska narzędzi widżetu wybierz opcję ... > **Zarządzaj rekordami uprawnień do tworzenia**.
 - c) Wybierz grupę, dla której chcesz wyświetlić rekord tworzenia uprawnień. Zostaną wyświetlone uprawnienia dla tej grupy.
 - d) Wybierz lub usuń zaznaczenie wymaganych uprawnień.

- e) Kliknij przycisk **Save** (Zapisz).
- Aby utworzyć rekord uprawnień dla obiektu IBM MQ :
 - a) Wybierz obiekt IBM MQ w widżecie na panelu kontrolnym. Powiązany menedżer kolejek musi być uruchomiony.
 - b) Z paska narzędzi widżetu wybierz opcję ... > **Zarządzaj rekordami uprawnień**.
- c) Kliknij ikonę plusa .
- d) Należy określić nazwę użytkownika lub grupy, dla której tworzony jest rekord uprawnień. Użytkownik lub grupa musi istnieć.
- e) Wybierz opcję **Typ jednostki** , aby określić, czy obiekt jest użytkownikiem, czy grupą.
- f) Kliknij opcję **Create**.
- g) Wybierz lub anuluj wybór uprawnień, które mają być używane przez użytkownika lub grupę. Dla każdego typu obiektu dostępne są różne uprawnienia.
- h) Kliknij przycisk **Save** (Zapisz).
- Aby utworzyć rekord uprawnień do tworzenia obiektów w menedżerze kolejek:
 - a) Wybierz menedżer kolejek w widżecie na panelu kontrolnym. Menedżer kolejek musi być uruchomiony.
 - b) Z paska narzędzi widżetu wybierz opcję ... > **Zarządzaj rekordami uprawnień do tworzenia**.
- c) Kliknij ikonę tworzenia .
- d) Należy określić nazwę użytkownika lub grupy, dla której tworzony jest rekord uprawnień. Użytkownik lub grupa musi istnieć.
- e) Wybierz opcję **Typ jednostki** , aby określić, czy obiekt jest użytkownikiem, czy grupą.
- f) Kliknij opcję **Create**.
- g) Wybierz lub anuluj wybór uprawnień do tworzenia, które mają być używane przez użytkownika lub grupę.
- h) Kliknij przycisk **Save** (Zapisz).

V 9.0.1

Monitorowanie wykorzystania zasobów systemu

Za pomocą widżetu Wykresy w produkcie IBM MQ Console można wyświetlać dane monitorowania dla menedżerów kolejek.

O tym zadaniu

Do panelu kontrolnego należy dodać widżet Wykresy, a następnie skonfigurować go w celu monitorowania konkretnego aspektu użycia zasobów. Istnieje możliwość utworzenia wielu instancji widżetu Wykresy w celu wyświetlenia różnych danych. Dane są wyświetlane w formacie wykresu.

Dane są gromadzone w odstępach 10-sekundynowych. Na osi X wykresu wyświetlana jest oś czasu. Na osi Y wyświetlane są jednostki odpowiednie dla wyświetlanego zasobu. Oś Y jest dynamicznie zmieniana, aby pomieścić zwracane dane.



Aby można było skonfigurować widżet wykresu, należy mieć co najmniej jeden uruchomiony menedżer kolejek.





Nie można monitorować wykorzystania zasobów systemowych w systemie z/OS.

Procedura

1. Dodaj widżet Wykresy do panelu kontrolnego:

- a) Kliknij ikonę Dodaj widget   .
- b) Wybierz opcję **Wykresy**.
2. Skonfiguruj widget Wykresy, aby wyświetlić dane:

- a) Kliknij ikonę konfiguracji   na pasku tytułu widgetu Wykresy.
- b) Opcjonalne: Wprowadź wartość w polu **Tytuł widgetu**. Ten tytuł jest wyświetlany na pasku tytułu widgetu.
- c) Wybierz opcję **Klasa zasobów** , która ma być monitorowana:

Procesory platformy

Monitoruj użycie procesorów.

Składnice danych trwałych platformy

Monitorowanie użycia zasobu dyskowego.

Statystyka użycia interfejsu API

Monitoruj wywołania funkcji API.

Statystyka użycia interfejsu API na kolejkę

Monitorowanie wywołań interfejsu API przez poszczególne kolejki. Po wybraniu tej klasy należy określić nazwę kolejki, która ma być monitorowana w polu **Obiekt** .

- d) Wybierz opcję **Typ zasobu** do monitorowania.

Typy zasobów, które są dostępne do wyboru, zależą od wybranej klasy zasobów. W poniższej tabeli przedstawiono typy zasobów:

<i>Tabela 6. Typy zasobów</i>		
Klasa	Typ	Opis
Procesory platformy	Wydajność procesora - cała platforma	Wybierz ten typ, aby wyświetlić dane o wydajności procesorów i pamięci.
	Wydajność procesora - uruchomiony menedżer kolejek	Wybierz ten typ, aby wyświetlić dane o wydajności procesorów i pamięci, które są powiązane z monitorowaną przez użytkownika menedżerami kolejek. Menedżer kolejek musi być uruchomiony, aby można było go monitorować. W przypadku monitorowania wyników z więcej niż jednego menedżera kolejek używane są różne kolory w celu odróżnienia danych o wydajności na wykresie.
Składnice danych trwałych platformy	Wykorzystanie dysku - cała platforma	Wybierz ten typ, aby wyświetlić dane o wydajności dla globalnego wykorzystania dysku.

<i>Tabela 6. Typy zasobów (kontynuacja)</i>		
Klasa	Typ	Opis
	Wykorzystanie dysku - uruchomione menedżery kolejek	Wybierz ten typ, aby wyświetlić dane o wydajności dotyczące użycia dysku powiązanego z monitorowanymi przez użytkownika menedżerami kolejek. Menedżer kolejek musi być uruchomiony, aby można było go monitorować. W przypadku monitorowania wyników z więcej niż jednego menedżera kolejek używane są różne kolory w celu odróżnienia danych o wydajności na wykresie.
	Wykorzystanie dysku - dziennik odtwarzania menedżera kolejek	Należy wybrać ten typ, aby wyświetlić dane dotyczące sposobu, w jaki pamięć dyskowa jest używana dla dziennika odtwarzania każdego monitorowanego menedżera kolejek.
Statystyka użycia interfejsu API	MQCONN i MQDISC	Wybierz ten typ, aby wyświetlić dane w wywołaniach MQCONN i MQDISC.
	MQOPEN i MQCLOSE	Wybierz ten typ, aby wyświetlić dane w wywołaniach MQOPEN i MQCLOSE.
	MQINQ i MQSET	Wybierz ten typ, aby wyświetlić dane w wywołaniach MQINQ i MQSET.
	MQPUT	Wybierz ten typ, aby wyświetlić dane na temat wywołań związanych z MQPUT.
	MQGET	Wybierz ten typ, aby wyświetlić dane na temat wywołań związanych z MQGET.
	Zatwierdzenie i wycofanie zmian	Wybierz ten typ, aby wyświetlić informacje na temat wykorzystania punktów synchronizacji przez menedżera kolejek.
	Subskrybowanie	Wybierz ten typ, aby wyświetlić dane powiązane z wywołaniami MQSUB.
	Publikowanie	Wybierz ten typ, aby wyświetlić dane o opublikowanych komunikatach.
Statystyka użycia interfejsu API na kolejkę	MQOPEN i MQCLOSE	Wybierz ten typ, aby wyświetlić dane w wywołaniach MQOPEN i MQCLOSE dla podanej kolejki.
	MQINQ i MQSET	Wybierz ten typ, aby wyświetlić dane w wywołaniach MQINQ i MQSET dla podanej kolejki.
	MQPUT i MQPUT1	Wybierz ten typ, aby wyświetlić dane w wywołaniach MQPUT i MQPUT1-related dla podanej kolejki.

<i>Tabela 6. Typy zasobów (kontynuacja)</i>		
Klasa	Typ	Opis
	MQGET	Wybierz ten typ, aby wyświetlić dane na temat wywołań MQGET dla określonej kolejki.

e) Wybierz opcję **Element zasobu** do monitorowania:

Elementy zasobów, które są dostępne do wyboru, zależą od wybranej klasy zasobów i typu zasobu. W poniższych tabelach przedstawiono elementy zasobów:

<i>Tabela 7. Elementy zasobów jednostek przetwarzania centralnego platformy</i>		
Typ	Element	Opis
Wydajność procesora - cała platforma	Procent czasu pracy procesora dla użytkownika	Wyświetla procent zajętości procesora w stanie użytkownika.
	Procent czasu pracy procesora dla systemu	Wyświetla procent zajętości procesora w stanie systemowym.
	Obciążenie procesora-średnia jednon минутowa	Pokazuje średnie obciążenie w ciągu 1 minuty.
	Obciążenie procesora-średnia pięciominutowa	Wyświetla średnie obciążenie w ciągu 5 minut.
	Obciążenie procesora-średnia piętnastominutowa	Pokazuje średnie obciążenie w ciągu piętnastu minut.
	Procent wolnej pamięci RAM	Wyświetla procent wolnej pamięci RAM.
	Łączna liczba bajtów pamięci RAM	Wyświetla łączną liczbę skonfigurowanych bajtów pamięci RAM.
Wydajność procesora - uruchomiony menedżer kolejek	Czas pracy procesora dla użytkownika - szacunkowy procent dla menedżera kolejek	Szacuje procent wykorzystania procesora w stanie użytkownika dla procesów, które są powiązane z monitorowanymi menedżerami kolejek.
	Czas pracy procesora dla systemu - szacunkowy procent dla menedżera kolejek	Szacuje wartość procentową wykorzystania procesora w stanie systemowym dla procesów powiązanych z monitorowanymi menedżerami kolejek.
	Łączna liczba bajtów pamięci RAM-szacowana liczba menedżerów kolejek	Szacuje łączną liczbę bajtów pamięci RAM, które są używane przez monitorowane menedżery kolejek.

<i>Tabela 8. Elementy dla zasobów danych trwałych platformy</i>		
Typ	Element	Opis
Wykorzystanie dysku - cała platforma	System plików śledzenia MQ - bajty w użyciu	Wyświetla liczbę bajtów pamięci dyskowej, które są używane przez system plików śledzenia.
	System plików śledzenia MQ - wolne miejsce	Wyświetla pamięć dyskową, która jest zarezerwowana dla systemu plików śledzenia, który jest wolny.
	System plików błędów MQ - bajty w użyciu	Wyświetla liczbę bajtów pamięci dyskowej, która jest używana przez dane błędu.
	System plików błędów MQ - wolne miejsce	Wyświetla pamięć dyskową, która jest zarezerwowana dla danych o błędach, które są wolne.
	Liczba plików FDC produktu MQ	Wyświetla bieżącą liczbę plików FDC.
Wykorzystanie dysku - uruchomione menedżery kolejek	System plików menedżera kolejek - bajty w użyciu	Wyświetla liczbę bajtów pamięci dyskowej, która jest używana przez pliki menedżera kolejek dla monitorowanych menedżerów kolejek.
	System plików menedżera kolejek - wolne miejsce	Wyświetla pamięć dyskową, która jest zarezerwowana dla plików menedżera kolejek, które są wolne.
Wykorzystanie dysku - dziennik odtwarzania menedżera kolejek	Dziennik - bajty w użyciu	Wyświetla liczbę bajtów pamięci dyskowej, która jest używana dla dzienników odtwarzania menedżerów kolejek, które są monitorowane.
	Dziennik - maksymalna liczba bajtów	Wyświetla maksymalną liczbę bajtów pamięci dyskowej, która jest skonfigurowana do użycia dla dzienników odtwarzania menedżera kolejek.
	System plików dzienników - bajty w użyciu	Wyświetla łączną liczbę bajtów na dysku, które są używane w systemie plików dziennika.
	System plików dzienników - maksymalna liczba bajtów	Wyświetla liczbę bajtów na dysku, które zostały skonfigurowane dla systemu plików dziennika.
	Dziennik - liczba zapisanych bajtów fizycznych	Wyświetla liczbę bajtów zapisanych w dziennikach odtwarzania.
	Dziennik - liczba zapisanych bajtów logicznych	Wyświetla logiczną liczbę bajtów zapisanych w dziennikach odtwarzania.

<i>Tabela 8. Elementy dla zasobów danych trwałych platformy (kontynuacja)</i>		
Typ	Element	Opis
	Dziennik - opóźnienie zapisu	Wyświetla miara opóźnienia podczas synchronicznego zapisu do dziennika odtwarzania menedżera kolejek.

<i>Tabela 9. Elementy dla zasobów statystyki użycia interfejsu API</i>		
Typ	Element	Opis
MQCONN i MQDISC	Liczba operacji MQCONN lub MQCONNX	Wyświetla liczbę wywołań MQCONN i MQCONNX.
	Liczba operacji MQCONN lub MQCONNX zakończonych niepowodzeniem	Wyświetla liczbę nieudanych połączeń z MQCONN i MQCONNX.
	Połączenia współbieżne - górny wskaźnik poziomu	Wyświetla maksymalną liczbę współbieżnych połączeń w bieżącym przedziale czasu statystyk.
	Liczba operacji MQDISC	Wyświetla liczbę wywołań MQDISC.
MQOPEN i MQCLOSE	Liczba operacji MQOPEN	Wyświetla liczbę wywołań MQOPEN.
	Liczba operacji MQOPEN zakończonych niepowodzeniem	Wyświetla liczbę zakończonych niepowodzeniem wywołań MQOPEN.
	Liczba operacji MQCLOSE	Wyświetla liczbę wywołań MQCLOSE.
	Liczba operacji MQCLOSE zakończonych niepowodzeniem	Wyświetla liczbę nieudanych połączeń z MQCLOSE.
MQINQ i MQSET	Liczba operacji MQINQ	Wyświetla liczbę wywołań MQINQ.
	Liczba operacji MQINQ zakończonych niepowodzeniem	Wyświetla liczbę nieudanych wywołań MQINQ.
	Liczba operacji MQSET	Wyświetla liczbę wywołań MQSET.
	Liczba operacji MQSET zakończonych niepowodzeniem	Wyświetla liczbę nieudanych wywołań MQSET.
MQPUT	Łączna liczba operacji MQPUT lub MQPUT1 w okresie	Wyświetla liczbę wywołań MQPUT i MQPUT1.
	Łączna liczba bajtów operacji MQPUT lub MQPUT1 w okresie	Wyświetla łączną liczbę bajtów danych, które są umieszczane przez wywołania MQPUT i MQPUT1.

<i>Tabela 9. Elementy dla zasobów statystyki użycia interfejsu API (kontynuacja)</i>		
Typ	Element	Opis
	Liczba operacji MQPUT dla komunikatów nietrwałych	Wyświetla liczbę nietrwałych komunikatów, które są umieszczane przez komendę MQPUT.
	Liczba operacji MQPUT dla komunikatów trwałych	Wyświetla liczbę trwałych komunikatów, które są umieszczane przez komendę MQPUT.
	Liczba operacji MQPUT zakończonych niepowodzeniem	Wyświetla liczbę nieudanych wywołań operacji MQPUT.
	Liczba operacji MQPUT1 dla komunikatów nietrwałych	Wyświetla liczbę nietrwałych komunikatów, które są umieszczane przez MQPUT1.
	Liczba operacji MQPUT1 dla komunikatów trwałych	Wyświetla liczbę trwałych komunikatów, które są umieszczane przez MQPUT1.
	Liczba operacji MQPUT1 zakończonych niepowodzeniem	Wyświetla liczbę wywołań zakończonych niepowodzeniem do MQPUT1.
	Umieszczanie nietrwałej liczby bajtów komunikatu	Wyświetla liczbę bajtów umieszczonych w nietrwałych komunikatach.
	Umieszczanie trwałych komunikatów-liczba bajtów	Wyświetla liczbę bajtów umieszczonych w trwałych komunikatach.
	Liczba operacji MQSTAT	Wyświetla liczbę wywołań MQSTAT.
	Liczba operacji MQSTAT zakończonych niepowodzeniem	Wyświetla liczbę nieudanych wywołań MQSTAT.
MQGET	Łączna liczba destruktywnych operacji get-count	Liczba komunikatów, które zostały usunięte z kolejek przez komendę MQGET.
	Łączna liczba bajtów niszczących operacji get w okresie	Liczba bajtów danych, które są usuwane z kolejek przez komendę MQGET.
	Liczba niszczących operacji get dla komunikatów nietrwałych	Liczba nietrwałych komunikatów, które są usuwane z kolejek przez komendę MQGET.
	Liczba niszczących operacji get dla komunikatów trwałych	Liczba trwałych komunikatów, które są usuwane z kolejek przez komendę MQGET.

<i>Tabela 9. Elementy dla zasobów statystyki użycia interfejsu API (kontynuacja)</i>		
Typ	Element	Opis
	Liczba operacji MQGET zakończonych niepowodzeniem	Wyświetla liczbę nieudanych wywołań MQGET.
	Otrzymane komunikaty nietrwałe - liczba bajtów	Wyświetla liczbę bajtów nietrwałych komunikatów, które są zwracane do wywołania MQGET.
	Otrzymane komunikaty trwałe - liczba bajtów	Wyświetla liczbę bajtów trwałych komunikatów, które są zwracane do wywołania MQGET.
	Przeglądanie komunikatów nietrwałych - liczba	Wyświetla liczbę komunikatów nietrwałych, które zostały przejrane.
	Przeglądanie komunikatów trwałych - liczba	Wyświetla liczbę komunikatów trwałych, które zostały przejrane.
	Liczba operacji przeglądania zakończonych niepowodzeniem	Wyświetla liczbę przeglądanych komunikatów zakończonych niepowodzeniem.
	Przeglądanie komunikatów nietrwałych - liczba bajtów	Wyświetla liczbę bajtów nietrwałych komunikatów, które zostały przejrane.
	Przeglądanie komunikatów trwałych - liczba bajtów	Wyświetla liczbę bajtów trwałych komunikatów, które zostały przejrane.
	Liczba komunikatów, które utraciły ważność	Wyświetla liczbę przedawnionych komunikatów.
	Liczba wyczyszczonych kolejek	Wyświetla liczbę kolejek, które zostały wyczyszczone.
	Liczba operacji MQCB	Wyświetla liczbę wywołań MQCB.
	Liczba operacji MQCB zakończonych niepowodzeniem	Wyświetla liczbę wywołań zakończonych niepowodzeniem dla obiektu MQCB.
	Liczba operacji MQCTL	Wyświetla liczbę wywołań MQCTL.
	Liczba operacji MQCTL zakończonych niepowodzeniem	Wyświetla liczbę nieudanych połączeń z MQCTL.

<i>Tabela 9. Elementy dla zasobów statystyki użycia interfejsu API (kontynuacja)</i>		
Typ	Element	Opis
Zatwierdzenie i wycofanie zmian	Liczba zatwierdzeń	Wyświetla liczbę wywołań MQCMIT.
	Liczba zatwierdzeń zakończonych niepowodzeniem	Wyświetla liczbę nieudanych połączeń z MQCMIT.
	Liczba wycofań zmian	Wyświetla liczbę wywołań MQBACK.
Subskrybowanie	Liczba operacji tworzenia trwałej subskrypcji	Wyświetla liczbę wywołań programu MQSUB w celu utworzenia trwałych subskrypcji.
	Liczba operacji zmiany trwałej subskrypcji	Wyświetla liczbę wywołań MQSUB w celu zmodyfikowania trwałych subskrypcji.
	Liczba operacji wznowienia trwałej subskrypcji	Wyświetla liczbę wywołań programu MQSUB w celu wznowienia trwałych subskrypcji.
	Liczba operacji tworzenia nietrwałej subskrypcji	Wyświetla liczbę wywołań programu MQSUB w celu utworzenia nietrwałych subskrypcji.
	Liczba operacji zmiany nietrwałej subskrypcji	Wyświetla liczbę wywołań MQSUB w celu zmodyfikowania nietrwałych subskrypcji.
	Liczba operacji wznowienia nietrwałej subskrypcji	Wyświetla liczbę wywołań programu MQSUB w celu wznowienia nietrwałych subskrypcji.
	Liczba operacji tworzenia/zmiany/wznowienia subskrypcji zakończonych niepowodzeniem	Wyświetla liczbę nieudanych połączeń z subskrypcjami MQSUBRQ w celu utworzenia, zmiany lub wznowienia subskrypcji.
	Liczba operacji usuwania trwałej subskrypcji	Wyświetla liczbę wywołań programu MQSUB w celu usunięcia trwałych subskrypcji.
	Liczba operacji usuwania nietrwałej subskrypcji	Wyświetla liczbę wywołań programu MQSUB w celu usunięcia nietrwałych subskrypcji.

<i>Tabela 9. Elementy dla zasobów statystyki użycia interfejsu API (kontynuacja)</i>		
Typ	Element	Opis
	Liczba operacji usuwania subskrypcji zakończonych niepowodzeniem	Wyświetla liczbę wywołań programu MQSUB w celu usunięcia subskrypcji.
	Liczba operacji MQSUBRQ	Wyświetla liczbę wywołań MQSUBRQ
	Liczba operacji MQSUBRQ zakończonych niepowodzeniem	Wyświetla liczbę nieudanych wywołań MQSUBRQ
	Trwały subskrybent - górny wskaźnik poziomu	Wyświetla maksymalną liczbę trwałych subskrypcji w bieżącym przedziale czasu statystyk.
	Trwały subskrybent - dolny wskaźnik poziomu	Wyświetla minimalną liczbę trwałych subskrypcji w bieżącym przedziale czasu statystyk.
	Nietrwały subskrybent - górny wskaźnik poziomu	Wyświetla maksymalną liczbę nietrwałych subskrypcji w bieżącym przedziale czasu statystyk.
	Nietrwały subskrybent - dolny wskaźnik poziomu	Wyświetla minimalną liczbę nietrwałych subskrypcji w bieżącym przedziale czasu statystyk.
Publikowanie	Łączna liczba operacji MQPUT lub MQPUT1 dla tematów w okresie	Liczba komunikatów, które są umieszczane w tematach.
	Łączna liczba bajtów operacji put w okresie	Liczba bajtów wiadomości umieszczonych w tematach.
	Liczba komunikatów opublikowanych do subskrybentów	Wyświetla liczbę komunikatów, które są publikowane dla subskrybentów.
	Liczba bajtów opublikowanych do subskrybentów	Wyświetla liczbę bajtów wiadomości, które są publikowane dla subskrybentów.
	Nietrwałe - liczba operacji MQPUT lub MQPUT1 dla tematów	Wyświetla liczbę nietrwałych komunikatów, które są umieszczane w tematach.
	Trwałe - liczba operacji MQPUT lub MQPUT1 dla tematów	Wyświetla liczbę komunikatów trwałych, które są umieszczane w tematach.

<i>Tabela 9. Elementy dla zasobów statystyki użycia interfejsu API (kontynuacja)</i>		
Typ	Element	Opis
	Liczba operacji MQPUT lub MQPUT1 dla tematów zakończonych niepowodzeniem	Wyświetla liczbę nieudanych prób umieszczenia w temacie.

<i>Tabela 10. Elementy dla zasobów statystyki wykorzystania interfejsu API na kolejkę</i>		
Typ	Element	Opis
MQOPEN i MQCLOSE	Liczba operacji MQOPEN	Wyświetla liczbę wywołań MQOPEN.
	Liczba operacji MQCLOSE	Wyświetla liczbę wywołań MQCLOSE.
MQINQ i MQSET	Liczba operacji MQINQ	Wyświetla liczbę wywołań MQINQ.
	Liczba operacji MQSET	Wyświetla liczbę wywołań MQSET.
MQPUT i MQPUT1	Liczba operacji MQPUT lub MQPUT1	Wyświetla liczbę wywołań MQPUT i MQPUT1.
	Liczba bajtów operacji MQPUT	Wyświetla łączną liczbę bajtów danych, które są umieszczane przez wywołania MQPUT i MQPUT1.
	Liczba operacji MQPUT dla komunikatów nietrwałych	Wyświetla liczbę nietrwałych komunikatów, które są umieszczane przez komendę MQPUT.
	Liczba operacji MQPUT dla komunikatów trwałych	Wyświetla liczbę trwałych komunikatów, które są umieszczane przez komendę MQPUT.
	Liczba operacji MQPUT1 dla komunikatów nietrwałych	Wyświetla liczbę nietrwałych komunikatów, które są umieszczane przez MQPUT1.
	Liczba operacji MQPUT1 dla komunikatów trwałych	Wyświetla liczbę trwałych komunikatów, które są umieszczane przez MQPUT1.
	nietrwałe - liczba bajtów	Wyświetla liczbę bajtów umieszczonych w nietrwałych komunikatach.
	trwałe - liczba bajtów	Wyświetla liczbę bajtów umieszczonych w trwałych komunikatach.

Tabela 10. Elementy dla zasobów statystyki wykorzystania interfejsu API na kolejkę (kontynuacja)

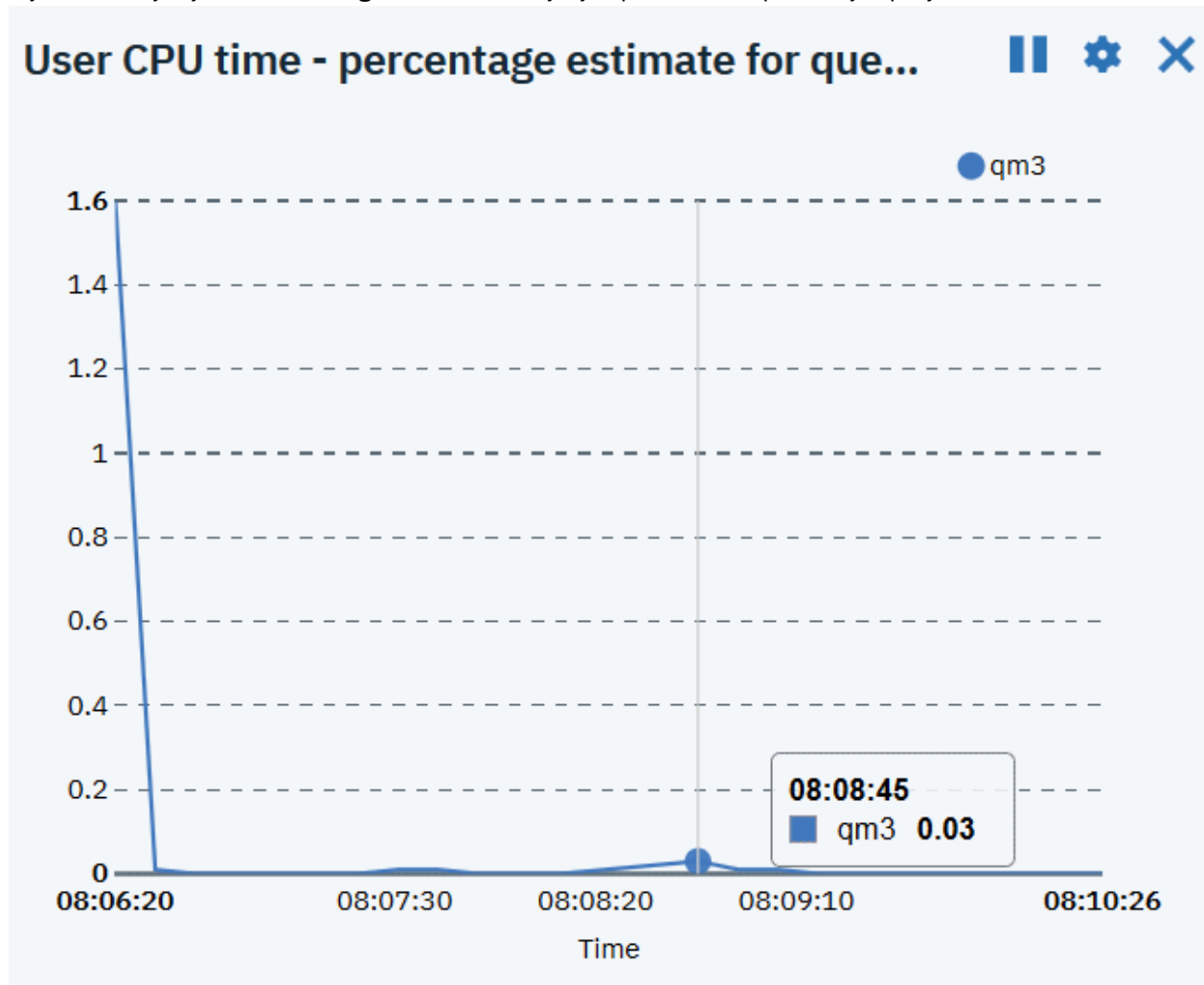
Typ	Element	Opis
	Liczba operacji put - unikanie kolejki	
	Liczba bajtów - unikanie kolejki	
	rywalizacja między blokadami	
MQGET	Liczba operacji MQGET	
	Liczba bajtów operacji MQGET	
	Liczba niszczących operacji MQGET dla komunikatów nietrwałych	Liczba nietrwałych komunikatów, które zostały usunięte z kolejki za pomocą komendy MQGET.
	Liczba niszczących operacji MQGET dla komunikatów trwałych	Liczba trwałych komunikatów, które są usuwane z kolejki przez komendę MQGET.
	Liczba bajtów niszczących operacji MQGET dla komunikatów nietrwałych	Wyświetla liczbę bajtów nietrwałych komunikatów, które są zwracane do wywołania MQGET.
	Liczba bajtów niszczących operacji MQGET dla komunikatów trwałych	Wyświetla liczbę bajtów trwałych komunikatów, które są zwracane do wywołania MQGET.
	Liczba operacji MQGET przeglądania komunikatów nietrwałych	Wyświetla liczbę komunikatów nietrwałych, które zostały przejrane.
	Liczba operacji MQGET przeglądania komunikatów trwałych	Wyświetla liczbę komunikatów trwałych, które zostały przejrane.
	Liczba bajtów operacji MQGET przeglądania komunikatów nietrwałych	Wyświetla liczbę bajtów nietrwałych komunikatów, które zostały przejrane.
	Liczba bajtów operacji MQGET przeglądania komunikatów trwałych	Wyświetla liczbę bajtów trwałych komunikatów, które zostały przejrane.
	komunikaty, które utraciły ważność	Wyświetla liczbę przedawnionych komunikatów.
	liczba wyczyszczonych kolejek	Wyświetla liczbę kolejek, które zostały wyczyszczone.
	średni czas w kolejce	
	Czas kolejki	

f) Wybierz menedżera kolejek, który ma być monitorowany, a następnie określ kolor, w którym mają być wyświetlane informacje dla tego menedżera kolejek. Kliknij przycisk **Dodaj**, aby dodać więcej menedżerów kolejek. Możliwe jest określenie do pięciu menedżerów kolejek.

g) Kliknij przycisk **Save** (Zapisz).

Wyniki

Po skonfigurowaniu widgetu istnieje krótkie opóźnienie, zanim dane zostaną wyświetlone na wykresie. Dane są wyświetlane wzdłuż osi czasu. Każdy punkt danych reprezentuje koniec 10-sekundowego okresu, w którym gromadzone są dane. Można umieścić wskaźnik myszy nad punktami danych na wykresie, aby wyświetlić szczegółowe informacje, jak pokazano w poniższym przykładzie: V 9.0.5



V 9.0.1

Konfigurowanie układów panelu kontrolnego

Panel kontrolny jest kontenerem w IBM MQ Console, w którym wyświetlane są widgety. Można utworzyć wiele kart panelu kontrolnego, aby wyświetlić różne wybory informacji.

O tym zadaniu

Każdą kartę panelu kontrolnego można skonfigurować, klikając strzałkę znajdującą się obok nazwy karty

▼ . Można zmienić nazwę karty i dodać opis dla karty. Można również skonfigurować liczbę kolumn, jaką ma karta.

Układ widgetów można skonfigurować w obrębie karty panelu kontrolnego, przeciągając i upuszczając widgety.

Procedura

- [“Tworzenie i usuwanie kart panelu kontrolnego” na stronie 132](#)
- [“Importowanie i eksportowanie układów panelu kontrolnego” na stronie 132](#)

V 9.0.1 Tworzenie i usuwanie kart panelu kontrolnego

Użytkownik może automatycznie utworzyć kartę panelu kontrolnego, która wyświetla informacje o konkretnym menedżerze kolejek lokalnych. Możliwe jest ręczne tworzenie i usuwanie kart panelu kontrolnego.

O tym zadaniu

W przypadku automatycznego tworzenia karty panelu kontrolnego w celu wyświetlenia informacji o konkretnym lokalnym menedżerze kolejek, automatycznie dodawane są następujące widżety:

- Kolejki, widżet
- Kanał połączenia klienckiego-widżet
- widżet kanałów
- Widżet Obiekty nastuchiwania
- Widżet Subskrypcje
- widżet tematu
- widżet informacji uwierzytelniających

Procedura

- Aby utworzyć kartę panelu kontrolnego:
 - a) Kliknij ikonę ze znakiem plus znajdującą się obok istniejących kart panelu kontrolnego



- b) Wprowadź nazwę nowej karty.
 - c) Opcjonalne: Wprowadź opis nowej karty.
 - d) Kliknij przycisk **Add** (Dodaj).
- Aby automatycznie utworzyć kartę panelu kontrolnego dla konkretnego menedżera kolejek:
 - a) Wybierz menedżer kolejek w widżecie menedżera kolejek lokalnych.
 - b) Wybierz opcję ... > **Dodaj nową kartę panelu kontrolnego**
Zostanie utworzona nowa karta panelu kontrolnego. Na karcie znajduje się nazwa menedżera kolejek.
 - Aby usunąć kartę panelu kontrolnego:

- a) Kliknij strzałkę znajdującą się obok nazwy karty panelu kontrolnego
- b) Wybierz opcję **Usuń kartę**.
- c) Potwierdź, że chcesz usunąć kartę panelu kontrolnego, klikając przycisk **Usuń**. Karta zostanie usunięta.

V 9.0.1 Importowanie i eksportowanie układów panelu kontrolnego



Układ panelu kontrolnego można zapisać, eksportując go z serwera IBM MQ Console. Zapisany układ panelu kontrolnego można zaimportować do programu IBM MQ Console.

O tym zadaniu

Podczas eksportowania panelu kontrolnego tworzony jest plik .json na dysku lokalnym. Następnie można zaimportować plik .json do panelu kontrolnego, aby ponownie utworzyć układ. W przypadku importowania układu panelu kontrolnego można wybrać opcję dodania zaimportowanych kart do istniejącego układu panelu kontrolnego. Alternatywnie można zastąpić istniejący układ panelu kontrolnego importowanym układem.

Procedura

- Aby wyeksportować układ panelu kontrolnego:

- a) Kliknij ikonę menu panelu kontrolnego   .
- b) Wybierz opcję **Eksportuj panel kontrolny**.
Plik zostanie zapisany w folderze pobierania przeglądarki.

- Aby zaimportować układ panelu kontrolnego:

- a) Kliknij ikonę menu panelu kontrolnego   .
- b) Wybierz opcję **Importuj panel kontrolny**.
Zostanie otwarte okno Importowanie konfiguracji panelu kontrolnego.
- c) Kliknij przycisk **Przełóżaj** i przejdź do położenia pliku zawierającego konfigurację.
- d) Wybierz sposób importowania kart panelu kontrolnego:
Do wyboru dostępne są następujące opcje:
 - **Dodaj zaimportowane karty panelu kontrolnego do istniejącego panelu kontrolnego**
 - **Zastąp istniejący panel kontrolny kartami z zaimportowanego panelu kontrolnego**
- e) Kliknij przycisk **Import** (Importuj).
Karty panelu kontrolnego są importowane.







Elementy sterujące panelu kontrolnego

Za pomocą elementów sterujących w górnej części panelu kontrolnego można włączyć śledzenie produktu IBM MQ Console, uzyskać dostęp do pomocy elektronicznej, wyświetlić informacje o IBM MQ Consolei wylogować się z produktu IBM MQ Console.

O tym zadaniu

Elementy sterujące panelu kontrolnego mają zastosowanie do ogólnych funkcji produktu IBM MQ Console.

Procedura

- Uzyskaj dostęp do IBM Documentation dla IBM MQ, klikając ikonę pomocy  .
- Kliknij ikonę menu  , aby zaimportować, wyeksportować lub zresetować panel kontrolny. Więcej informacji na ten temat zawiera sekcja [“Konfigurowanie układów panelu kontrolnego”](#) na stronie 131.
- Kliknij ikonę ustawień  , aby włączyć lub wyłączyć śledzenie diagnostyczne dla IBM MQ Console.
- Aby wyświetlić informacje o IBM MQ Console, w tym informacje o zalogowanym użytkowniku, należy kliknąć ikonę osoby  i wybrać opcję **Informacje**.

- Wyloguj się z IBM MQ Console , klikając opcję **Wyloguj**.
Ta opcja nie jest dostępna, jeśli użytkownik jest zalogowany do bazy danych IBM MQ Console za pomocą certyfikatu klienta lub jeśli używana jest przykładowa konfiguracja zabezpieczeń no_security.xml . Więcej informacji na temat konfiguracji zabezpieczeń zawiera sekcja [Konfigurowanie zabezpieczeń produktu IBM MQ Console](#).

Skróty klawiaturowe

Do pracy z produktem IBM MQ Console można używać skrótów klawiaturowych.

Poniższa tabela zawiera listę dostępnych skrótów. Wszystkie skróty są używane z poziomu widgetu.

<i>Tabela 11. Skróty klawiaturowe w konsoli</i>	
Klucz	Działanie
p	Wyświetlanie właściwości wybranego obiektu
c	Utwórz nowy obiekt
shift-d	Usuń wybrany obiekt
a	Wybierz wszystkie obiekty w widżecie
shift-a	Anuluj wybór obiektów

Windows

Linux

Administrowanie przy użyciu IBM MQ Explorer

IBM MQ Explorer umożliwia lokalne lub zdalne administrowanie siecią z poziomu komputera, na którym działa produkt Windows, lub tylko w systemie Linux x86-64 .

Produkty IBM MQ for Windows i IBM MQ for Linux x86-64 udostępniają interfejs administracyjny o nazwie IBM MQ Explorer , który umożliwia wykonywanie zadań administracyjnych jako alternatywa dla używania komend sterujących lub MQSC. Sekcja [Porównywanie zestawów komend](#) zawiera informacje o tym, co można zrobić za pomocą konsoli IBM MQ Explorer.

IBM MQ Explorer umożliwia lokalne lub zdalne administrowanie siecią z komputera, na którym działa produkt Windows, lub Linux x86-64, poprzez wskazanie IBM MQ Explorer w menedżerach kolejek i klastrach, które są zainteresowane. Może on łączyć się zdalnie z menedżerami kolejek, które działają na dowolnej obsługiwanej platformie, w tym w systemie z/OS, co umożliwia przeglądanie, eksplorowanie i modyfikowanie całej szyny komunikacyjnej z poziomu konsoli.

Aby skonfigurować zdalne menedżery kolejek produktu IBM MQ , tak aby produkt IBM MQ Explorer mógł administrować nimi, należy zapoznać się z [“Wstępnie wymagane oprogramowanie i definicje dla produktu IBM MQ Explorer”](#) na stronie 136.

Umożliwia on wykonywanie zadań, zwykle związanych z konfigurowaniem i dostrajaniem środowiska pracy dla produktu IBM MQ, lokalnie lub zdalnie w domenie systemowej Windows lub Linux x86-64 .

W systemie Linux uruchomienie produktu IBM MQ Explorer może się nie powieść, jeśli istnieje więcej niż jedna instalacja produktu Eclipse . W takim przypadku należy uruchomić produkt IBM MQ Explorer przy użyciu innego identyfikatora użytkownika, który zostanie użyty dla innej instalacji środowiska Eclipse .

W systemie Linux, aby pomyślnie uruchomić serwer IBM MQ Explorer , należy mieć możliwość zapisu pliku w katalogu osobistym, a katalog osobisty musi istnieć.

Windows

Linux

Co można zrobić za pomocą IBM MQ Explorer

Za pomocą programu IBM MQ Explorer można wykonywać zadania administracyjne przy użyciu serii okien dialogowych Widoki treści i właściwości. Produkt IBM MQ Explorer można również rozszerzyć, pisząc jedną lub więcej wtyczek Eclipse .

Zadania dotyczące serwera IBM MQ Explorer

Za pomocą konsoli IBM MQ Explorer można wykonywać następujące zadania:

- Tworzenie i usuwanie menedżera kolejek (tylko na komputerze lokalnym).
- Uruchamianie i zatrzymywanie menedżera kolejek (tylko na komputerze lokalnym).
- Zdefiniuj, wyświetl i zmień definicje obiektów produktu IBM MQ , takich jak kolejki i kanały.
- Przeglądaj komunikaty w kolejce.
- Uruchamianie i zatrzymywanie kanału.
- Wyświetlanie informacji o statusie kanału, obiektu nastuchiwania, kolejki lub obiektów usług.
- Wyświetlanie menedżerów kolejek w klastrze.
- Sprawdź, które aplikacje, użytkownicy lub kanały mają otwartą określoną kolejkę.
- Utwórz nowy klaster menedżera kolejek przy użyciu kreatora tworzenia nowego klastra.
- Dodawanie menedżera kolejek do klastra przy użyciu kreatora dodawania menedżera kolejek do klastra.
- Zarządzanie obiektem informacji uwierzytelniającej, używanym z zabezpieczeniami kanału Transport Layer Security (TLS).
- Tworzenie i usuwanie inicjatorów kanałów, monitorów wyzwalaczy i programów nastuchujących.
- Uruchamianie lub zatrzymywanie serwerów komend, inicjatorów kanałów, monitorów wyzwalaczy i programów nastuchujących.
- Ustaw konkretne usługi, które mają być uruchamiane automatycznie podczas uruchamiania menedżera kolejek.
- Zmodyfikuj właściwości menedżerów kolejek.
- Zmień lokalny domyślny menedżer kolejek.
- Za pomocą interfejsu GUI programu **strmqikm** (ikeyMan) można zarządzać certyfikatami TLS, powiązać certyfikaty z menedżerami kolejek oraz skonfigurować i skonfigurować składnice certyfikatów (tylko na komputerze lokalnym).
- Utwórz obiekty JMS z obiektów IBM MQ , a obiekty IBM MQ z obiektów JMS .
- Utwórz fabrykę połączeń produktu JMS dla dowolnego z obecnie obsługiwanych typów.
- Zmodyfikuj parametry dla dowolnej usługi, takie jak numer portu TCP dla obiektu nastuchiwania lub nazwę kolejki inicjatora kanału.
- Uruchom lub zatrzymaj śledzenie usługi.

Okna dialogowe Widoki treści i właściwości

Zadania administracyjne można wykonywać przy użyciu serii okien dialogowych Widoki treści i właściwości.

Widok zawartości

Widok treści to panel, który może wyświetlać następujące elementy:

- Atrybuty i opcje administracyjne odnoszące się do samego IBM MQ .
- Atrybuty i opcje administracyjne odnoszące się do jednego lub większej liczby powiązanych obiektów.
- Atrybuty i opcje administracyjne dla klastra.

Okna dialogowe właściwości

Okno dialogowe właściwości to panel wyświetlający atrybuty odnoszące się do obiektu z serii pól, z których niektóre można edytować.

Za pomocą widoku Navigator można poruszać się za pomocą konsoli IBM MQ Explorer . Navigator umożliwia wybranie widoku treści, który jest wymagany.

Rozszerzanie serwera IBM MQ Explorer

Produkt IBM MQ Explorer prezentuje informacje w stylu zgodnym ze strukturą środowiska Eclipse i innymi aplikacjami we wtyczkach, które są obsługiwane przez Eclipse .

Dzięki rozszerzaniu partycji IBM MQ Explorer administratorzy systemu mają możliwość dostosowania IBM MQ Explorer w celu usprawnienia sposobu administrowania produktem IBM MQ.

Więcej informacji na ten temat zawiera sekcja [Rozszerzanie programu MQ Explorer](#).

Rozstrzygnięcie, czy ma być używany IBM MQ Explorer

Podczas określania, czy podczas instalacji ma być używany produkt IBM MQ Explorer , należy wziąć pod uwagę informacje wymienione w tym temacie.

Należy zdawać sobie sprawę z następujących kwestii:

nazwy obiektów

Jeśli podczas pracy z obiektami za pomocą komend MQSC używane są małe nazwy dla menedżerów kolejek i innych obiektów za pomocą IBM MQ Explorer, nazwy obiektów należy ująć w pojedynczy cudzysłów lub IBM MQ nie rozpoznaje ich.

Duże menedżery kolejek

IBM MQ Explorer najlepiej współpracuje z małymi menedżerami kolejek. Jeśli istnieje duża liczba obiektów w pojedynczym menedżerze kolejek, mogą wystąpić opóźnienia, gdy program IBM MQ Explorer wyodrębnia wymagane informacje w celu zaprezentowania go w widoku.

Klastry

Klastry produktu IBM MQ mogą potencjalnie zawierać setki lub tysiące menedżerów kolejek. Produkt IBM MQ Explorer przedstawia menedżery kolejek w klastrze przy użyciu struktury drzewa. Fizyczna wielkość klastra nie wpływa na szybkość IBM MQ Explorer dramatycznie, ponieważ serwer IBM MQ Explorer nie łączy się z menedżerami kolejek w klastrze, dopóki nie zostaną wybrane.

Konfigurowanie produktu IBM MQ Explorer

W tej sekcji opisano kroki, które należy wykonać, aby skonfigurować produkt IBM MQ Explorer.

- [“Wstępnie wymagane oprogramowanie i definicje dla produktu IBM MQ Explorer” na stronie 136](#)
- [“zabezpieczenia dla IBM MQ Explorer” na stronie 137](#)
- [“Wyświetlanie i ukrywanie menedżerów kolejek i klastrów w produkcie IBM MQ Explorer” na stronie 140](#)
- [“Przypisanie do klastra i IBM MQ Explorer” na stronie 141](#)
- [“Konwersja danych dla IBM MQ Explorer” na stronie 142](#)

Wstępnie wymagane oprogramowanie i definicje dla produktu IBM MQ Explorer

Przed podjęciem próby użycia IBM MQ Explorer należy upewnić się, że spełnione są następujące wymagania.

Program IBM MQ Explorer może łączyć się ze zdalnymi menedżerami kolejek tylko przy użyciu protokołu komunikacyjnego TCP/IP.

Sprawdź, czy:

1. Serwer komend jest uruchomiony dla każdego zdalnie administrowanego menedżera kolejek.
2. Odpowiedni obiekt nasłuchiwania TCP/IP musi być uruchomiony w każdym zdalnym menedżerze kolejek. Ten obiekt może być programem nasłuchującym IBM MQ lub, w systemach UNIX and Linux , demonem inetd.
3. Kanał połączenia z serwerem, domyślnie nazwany SYSTEM.ADMIN.SVRCONN, istnieje we wszystkich menedżerach kolejek zdalnych.

Kanał można utworzyć za pomocą następującej komendy MQSC:

```
DEFINE CHANNEL(SYSTEM.ADMIN.SVRCONN) CHLTYPE(SVRCONN)
```

Ta komenda tworzy podstawową definicję kanału. Jeśli potrzebna jest bardziej zaawansowana definicja (na przykład w celu skonfigurowania zabezpieczeń), potrzebne są dodatkowe parametry. Więcej informacji na ten temat zawiera sekcja [DEFINE CHANNEL](#).

4. Kolejka systemowa SYSTEM.MQEXPLORER.REPLY.MODEL musi istnieć.

zabezpieczenia dla IBM MQ Explorer

Jeśli produkt IBM MQ jest używany w środowisku, w którym ważne jest, aby kontrolować dostęp użytkowników do konkretnych obiektów, może być konieczne rozważenie aspektów związanych z bezpieczeństwem korzystania z produktu IBM MQ Explorer.

Autoryzacja do używania IBM MQ Explorer

Każdy użytkownik może korzystać z IBM MQ Explorer, ale niektóre uprawnienia są wymagane do nawiązywania połączeń, uzyskiwania dostępu do menedżerów kolejek i zarządzania nimi.

Do wykonywania lokalnych czynności administracyjnych przy użyciu IBM MQ Explorer wymagane jest posiadanie przez użytkownika niezbędnych uprawnień do wykonywania zadań administracyjnych. Jeśli użytkownik jest członkiem grupy mqm, użytkownik ma uprawnienia do wykonywania wszystkich lokalnych zadań administracyjnych.

Aby połączyć się ze zdalnym menedżerem kolejek i wykonać zdalne zadania administracyjne za pomocą IBM MQ Explorer, użytkownik wykonujący IBM MQ Explorer musi mieć następujące uprawnienia:

- Uprawnienie CONNECT do obiektu docelowego menedżera kolejek
- Uprawnienie INQUIRE do obiektu docelowego menedżera kolejek
- Wyświetlaj uprawnienia do docelowego obiektu menedżera kolejek
- INQUIRE do kolejki, SYSTEM.MQEXPLORER.REPLY.MODEL
- Wyświetlaj uprawnienia do kolejki SYSTEM.MQEXPLORER.REPLY.MODEL
- INPUT (get) uprawnienie do kolejki, SYSTEM.MQEXPLORER.REPLY.MODEL
- Uprawnienie OUTPUT (put) do kolejki SYSTEM.MQEXPLORER.REPLY.MODEL
- Uprawnienie OUTPUT (put) do kolejki SYSTEM.ADMIN.COMMAND.QUEUE
- INQUIRE w kolejce, SYSTEM.ADMIN.COMMAND.QUEUE
- Uprawnienie do wykonania wybranego działania

Uwaga: Uprawnienie INPUT odnosi się do danych wejściowych dla użytkownika z kolejki (operacja get). Uprawnienia OUTPUT odnoszą się do danych wyjściowych użytkownika do kolejki (operacja put).

Aby połączyć się ze zdalnym menedżerem kolejek w systemie IBM MQ for z/OS i wykonać zdalne zadania administracyjne za pomocą IBM MQ Explorer, należy podać następujące informacje:

- Profil RACF dla kolejki systemowej SYSTEM.MQEXPLORER.REPLY.MODEL
- Profil RACF dla kolejek, AMQ.MQEXPLORER.*

Ponadto użytkownik wykonujący IBM MQ Explorer jest zobowiązany do posiadania następujących uprawnień:

- RACF Uprawnienie UPDATE do kolejki systemowej SYSTEM.MQEXPLORER.REPLY.MODEL
- RACF Uprawnienia UPDATE do kolejek AMQ.MQEXPLORER.*
- Uprawnienie CONNECT do obiektu docelowego menedżera kolejek
- Uprawnienie do wykonania wybranego działania
- Uprawnienie READ do wszystkich profili hlq.DISPLAY.object w klasie MQCMD5

Informacje na temat nadawania uprawnień do obiektów produktu IBM MQ zawiera sekcja [Udzielanie dostępu do obiektu IBM MQ w systemach UNIX lub Linux oraz w produkcie Windows](#).

Jeśli użytkownik podejmie próbę wykonania operacji, której nie ma do wykonania, docelowy menedżer kolejek wywołuje procedury dotyczące niepowodzenia autoryzacji, a operacja nie powiedzie się.

Domyślny filtr w IBM MQ Explorer polega na wyświetlaniu wszystkich obiektów IBM MQ. Jeśli istnieją obiekty IBM MQ, do których użytkownik nie ma uprawnień DISPLAY, generowane są błędy autoryzacji. Jeśli rejestrowane są zdarzenia uprawnień, należy ograniczyć zakres obiektów wyświetlanych do tych obiektów, do których użytkownik ma uprawnienie DISPLAY.

Zabezpieczenia dotyczące nawiązywania połączenia ze zdalnymi menedżerami kolejek z programu IBM MQ Explorer

Należy zabezpieczyć kanał między serwerem IBM MQ Explorer i każdym zdalnym menedżerem kolejek.

Program IBM MQ Explorer łączy się ze zdalnymi menedżerami kolejek jako aplikacją kliencką MQI. Oznacza to, że każdy zdalny menedżer kolejek musi mieć definicję kanału połączenia z serwerem i odpowiedni program nasłuchujący TCP/IP. Jeśli kanał połączenia z serwerem nie jest bezpieczny, istnieje możliwość nawiązania przez złośliwą aplikację połączenia z tym samym kanałem połączenia z serwerem i uzyskania dostępu do obiektów menedżera kolejek z nieograniczonymi uprawnieniami. W celu zabezpieczenia kanału połączenia z serwerem należy określić niepustą wartość atrybutu MCAUSER kanału, użyć rekordów uwierzytelniania kanału lub użyć wyjścia zabezpieczeń.

Domyślna wartość atrybutu MCAUSER to identyfikator użytkownika lokalnego. Jeśli jako atrybut MCAUSER kanału połączenia z serwerem zostanie podana niepusta nazwa użytkownika, wszystkie programy łączące się z menedżerem kolejek przy użyciu tego kanału będą uruchamiane z tożsamością nazwanego użytkownika i mają ten sam poziom uprawnień. Nie jest to możliwe, jeśli używane są rekordy uwierzytelniania kanału.

Korzystanie z wyjścia zabezpieczeń z IBM MQ Explorer

Za pomocą IBM MQ Explorer można określić domyślne wyjścia zabezpieczeń dla wyjścia zabezpieczeń i menedżera kolejek.

Użytkownik może zdefiniować domyślne wyjście zabezpieczeń, które może być używane dla wszystkich nowych połączeń klientów z IBM MQ Explorer. To domyślne wyjście może zostać przestonięte w momencie nawiązania połączenia. Istnieje również możliwość zdefiniowania wyjścia zabezpieczeń dla pojedynczego menedżera kolejek lub zestawu menedżerów kolejek, które to wyjście jest wykonywane po nawiązaniu połączenia. Procedury zewnętrzne są określane za pomocą IBM MQ Explorer. Więcej informacji zawiera pomoc programu IBM MQ Explorer.

Korzystanie z IBM MQ Explorer w celu nawiązania połączenia ze zdalnym menedżerem kolejek przy użyciu kanałów MQI z włączoną obsługą protokołu TLS

IBM MQ Explorer łączy się ze zdalnymi menedżerami kolejek przy użyciu kanału MQI. Aby zabezpieczyć kanał MQI przy użyciu zabezpieczeń TLS, należy utworzyć kanał przy użyciu tabeli definicji kanału klienta.

Informacje na temat tworzenia kanału MQI przy użyciu tabeli definicji kanału klienta zawiera sekcja [Przegląd produktu IBM MQ MQI clients](#).

Gdy kanał jest ustanawiany przy użyciu tabeli definicji kanału klienta, można użyć programu IBM MQ Explorer w celu nawiązania połączenia ze zdalnym menedżerem kolejek za pomocą kanału MQI z włączoną obsługą protokołu TLS, zgodnie z opisem w sekcji [“Zadania w systemie, w którym znajduje się zdalny menedżer kolejek”](#) na stronie 138 i [“Zadania w systemie, w którym znajduje się IBM MQ Explorer”](#) na stronie 139.

Zadania w systemie, w którym znajduje się zdalny menedżer kolejek

W systemie, w którym znajduje się zdalny menedżer kolejek, wykonaj następujące czynności:

1. Zdefiniuj parę kanałów połączenia z serwerem i połączenia z klientem, a następnie określ odpowiednią wartość atrybutu *SSLCIPH* w połączeniu z serwerem w obu kanałach. Więcej informacji na temat atrybutu *SSLCIPH* zawiera sekcja [Zabezpieczanie kanałów za pomocą protokołu TLS](#).

2. Wyślij tabelę definicji kanału AMQCLCHL . TAB, która znajduje się w katalogu @ipcc menedżera kolejek, do systemu, w którym znajduje się IBM MQ Explorer.
3. Uruchom program nasłuchujący TCP/IP na wyznaczonym porcie.
4. Umieść zarówno certyfikaty CA, jak i osobiste TLS w katalogu SSL menedżera kolejek:
 - /var/mqm/qmgrs/+QMNAME+/SSL dla systemów UNIX and Linux
 - C:\Program Files\IBM\MQ\qmgrs\+QMNAME+\SSL dla systemów Windows
 Gdzie +QMNAME+ to znacznik reprezentujący nazwę menedżera kolejek.
5. Utwórz plik bazy danych kluczy typu CMS o nazwie key . kdb. Ukryj hasło w pliku, sprawdzając opcję w pliku **strmqikm** (iKeyman). Interfejs GUI lub za pomocą opcji -stsh z komendami **runmqckm**.
6. Dodaj certyfikaty ośrodka CA do bazy danych kluczy utworzonej w poprzednim kroku.
7. Zaimportuj certyfikat osobisty dla menedżera kolejek do bazy danych kluczy.

Więcej szczegółowych informacji na temat pracy z protokołem TLS w systemach Windows zawiera sekcja [Praca z protokołem TLS w systemie UNIX, Linux, and Windows](#).

Zadania w systemie, w którym znajduje się IBM MQ Explorer

W systemie, w którym znajduje się IBM MQ Explorer, wykonaj następujące czynności:

1. Utwórz plik bazy danych kluczy typu JKS o nazwie key . jks. Ustaw hasło dla tego pliku bazy danych kluczy.

Produkt IBM MQ Explorer używa plików kluczy Java (JKS) na potrzeby zabezpieczeń TLS, a więc tworzony plik kluczy na potrzeby konfigurowania protokołu TLS dla IBM MQ Explorer musi być zgodny z tym plikiem.
2. Dodaj certyfikaty ośrodka CA do bazy danych kluczy utworzonej w poprzednim kroku.
3. Zaimportuj certyfikat osobisty dla menedżera kolejek do bazy danych kluczy.
4. W systemach Windows i Linux uruchom program IBM MQ Explorer , korzystając z menu systemowego, pliku wykonywalnego MQExp1orer lub komendy **strmqcfig**.
5. Na pasku narzędzi produktu IBM MQ Explorer kliknij opcję **Okna-> Preferencje**, a następnie rozwiń pozycję **IBM MQ Explorer**, a następnie kliknij opcję **Sklepy certyfikatów klienta SSL**. Wprowadź nazwę i hasło dla pliku JKS utworzonego w kroku 1 produktu [“Zadania w systemie, w którym znajduje się IBM MQ Explorer”](#) na stronie 139, zarówno w magazynie zaufanych certyfikatów, jak i w składnicy certyfikatów osobistych, a następnie kliknij przycisk **OK**.
6. Zamknij okno **Preferencje**, a następnie kliknij prawym przyciskiem myszy opcję **Menedżery kolejek**. Kliknij opcję **Pokaż/ukryj menedżery kolejek**, a następnie kliknij przycisk **Dodaj** na ekranie **Pokaż/ukryj menedżery kolejek**.
7. Wpisz nazwę menedżera kolejek i wybierz opcję **Połącz bezpośrednio**. Kliknij przycisk Dalej.
8. Wybierz opcję **Użyj tabeli definicji kanału klienta (CCDT)** i określ położenie pliku tabeli kanału, który został przesłany ze zdalnego menedżera kolejek w kroku 2 w [“Zadania w systemie, w którym znajduje się zdalny menedżer kolejek”](#) na stronie 138 w systemie, w którym znajduje się zdalny menedżer kolejek.
9. Kliknij przycisk **Zakończ**. Teraz można uzyskać dostęp do menedżera kolejek zdalnych z poziomu IBM MQ Explorer.

Nawiązanie połączenia przez inny menedżer kolejek z produktem IBM MQ Explorer

IBM MQ Explorer umożliwia nawiązanie połączenia z menedżerem kolejek za pośrednictwem pośredniego menedżera kolejek, do którego IBM MQ Explorer jest już połączony.

W tym przypadku program IBM MQ Explorer umieszcza komunikaty komend PCF w pośrednim menedżerze kolejek, podając następujące informacje:

- Parametr *ObjectQMgrName* w deskrypcji obiektu (MQOD) jako nazwa docelowego menedżera kolejek. Więcej informacji na temat rozstrzygania nazw kolejek można znaleźć w sekcji [Uchwata nazwy](#).

- Parametr *UserIdentifier* w deskrytorze komunikatu (MQMD) jako lokalny userId.

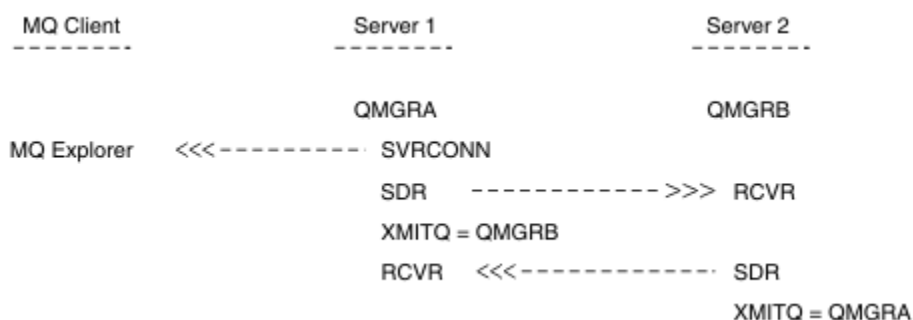
Jeśli połączenie jest następnie używane do łączenia się z docelowym menedżerem kolejek za pośrednictwem pośredniego menedżera kolejek, identyfikator userId jest ponownie generowany w parametrze *UserIdentifier* deskryptora komunikatu (MQMD). Aby program nasłuchujący MCA w docelowym menedżerze kolejek akceptować ten komunikat, musi być ustawiony atrybut MCAUSER lub parametr userId musi już istnieć z uprawnieniem do umieszczania.

Serwer komend w docelowym menedżerze kolejek umieszcza komunikaty w kolejce transmisji, określając parametr userId w parametrze *UserIdentifier* w deskrytorze komunikatu (MQMD). Aby ta operacja została pomyślnie wykonana, użytkownik userId musi już istnieć w docelowym menedżerze kolejek z uprawnieniem do umieszczania.

W poniższym przykładzie przedstawiono sposób łączenia menedżera kolejek za pośrednictwem pośredniego menedżera kolejek z produktem IBM MQ Explorer.

Nawiąże zdalne połączenie administracyjne z menedżerem kolejek. Sprawdź, czy:

- Menedżer kolejek na serwerze jest aktywny i ma zdefiniowany kanał połączenia z serwerem (SVRCONN).
- Program nasłuchujący jest aktywny.
- Serwer komend jest aktywny.
- SYSTEM.MQ EXPLORER.REPLY.MODEL została utworzona i użytkownik ma wystarczające uprawnienia.
- Programy nasłuchujące menedżera kolejek, serwery komend i kanały nadawcze są uruchomione.



W tym przykładzie:

- Program IBM MQ Explorer jest połączony z menedżerem kolejek QMGRA (działającym na serwerze Server1) przy użyciu połączenia klienckiego.
- Menedżer kolejek QMGRB na serwerze Server2 może być teraz połączony z produktem IBM MQ Explorer za pomocą pośredniego menedżera kolejek (QMGRA)
- Podczas nawiązywania połączenia z produktem QMGRB z produktem IBM MQ Explorer wybierz opcję QMGRA jako pośredni menedżer kolejek.

W tej sytuacji nie ma bezpośredniego połączenia z QMGRB z IBM MQ Explorer; połączenie z QMGRB jest za pośrednictwem QMGRA.

Menedżer kolejek QMGRB na serwerze Server2 jest połączony z serwerem QMGRA na serwerze Server1 za pomocą kanałów nadawczych. Kanał między QMGRA i QMGRB musi być skonfigurowany w taki sposób, aby zdalne administrowanie było możliwe. Patrz [“Przygotowywanie kanałów i kolejek transmisji do zdalnego administrowania” na stronie 199.](#)

Wyświetlanie i ukrywanie menedżerów kolejek i klastrów w produkcie IBM MQ Explorer

Program IBM MQ Explorer może wyświetlać więcej niż jeden menedżer kolejek jednocześnie. Na panelu Pokaż/ukryj menedżera kolejek (wybierany z menu dla węzła drzewa Menedżery kolejek) można wybrać, czy wyświetlane są informacje o innej (zdalnej) maszynie. Lokalne menedżery kolejek są wykrywane automatycznie.

Aby wyświetlić zdalny menedżer kolejek:

1. Kliknij prawym przyciskiem myszy węzeł drzewa **Menedżery kolejek** , a następnie wybierz opcję **Pokaż/ukryj menedżery kolejek**.
2. Kliknij przycisk **Add** (Dodaj). Zostanie wyświetlony panel Pokaż/ukryj menedżery kolejek.
3. Wprowadź nazwę zdalnego menedżera kolejek oraz nazwę hosta lub adres IP w udostępnionych polach.

Nazwa hosta lub adres IP są używane do nawiązania połączenia klienckiego ze zdalnym menedżerem kolejek przy użyciu domyślnego kanału połączenia z serwerem SYSTEM.ADMIN.SVRCONN lub kanału połączenia z serwerem zdefiniowanym przez użytkownika.

4. Kliknij opcję **Zakończ**.

Na panelu Pokaż/ukryj menedżery kolejek również wyświetlana jest lista wszystkich widocznych menedżerów kolejek. Panel ten służy do ukrywania menedżerów kolejek w widoku nawigacyjnym.

Jeśli w programie IBM MQ Explorer jest wyświetlany menedżer kolejek, który jest elementem klastra, klastr jest wykrywany i wyświetlany automatycznie.

Aby wyeksportować listę menedżerów kolejek zdalnych z tego panelu, wykonaj następujące czynności:

1. Zamknij panel Pokaż/ukryj menedżery kolejek.
2. Kliknij prawym przyciskiem myszy najwyższy węzeł drzewa **IBM MQ** na panelu nawigacyjnym IBM MQ Explorer, a następnie wybierz opcję **Eksportuj ustawienia produktu IBM MQ Explorer** .
3. Kliknij opcję **IBM MQ Explorer > Ustawienia IBM MQ Explorer** .
4. Wybierz opcję **Informacje o połączeniu > Menedżery kolejek zdalnych**.
5. Wybierz plik, w którym mają zostać zapisane wyeksportowane ustawienia.
6. Na koniec należy kliknąć przycisk **Zakończ** , aby wyeksportować informacje o połączeniu zdalnego menedżera kolejek do określonego pliku.

Aby zaimportować listę menedżerów kolejek zdalnych:

1. Kliknij prawym przyciskiem myszy najwyższy węzeł drzewa **IBM MQ** na panelu nawigacyjnym IBM MQ Explorer, a następnie wybierz opcję **Importuj ustawienia produktu IBM MQ Explorer** .
2. Kliknij opcję **IBM MQ Explorer > Ustawienia IBM MQ Explorer** .
3. Kliknij przycisk **Przełóżnik** przejdź do ścieżki do pliku zawierającego informacje o połączeniu z menedżerem kolejek zdalnych.
4. Kliknij przycisk **Otwórz**. Jeśli plik zawiera listę menedżerów kolejek zdalnych, to pole **Informacje o połączeniu > Menedżery kolejek zdalnych** jest zaznaczone.
5. Na koniec należy kliknąć przycisk **Zakończ** , aby zaimportować informacje o połączeniu zdalnego menedżera kolejek do serwera IBM MQ Explorer.

Przypisanie do klastra i IBM MQ Explorer

Produkt IBM MQ Explorer wymaga informacji o menedżerach kolejek, które są elementami klastra.

Jeśli menedżer kolejek jest elementem klastra, to węzeł drzewa klastra zostanie zapełniony automatycznie.

Jeśli menedżery kolejek stają się elementami klastrów podczas działania serwera IBM MQ Explorer , należy zachować IBM MQ Explorer aktualne dane administracyjne dotyczące klastrów, tak aby mógł on skutecznie komunikować się z nimi i wyświetlać poprawne informacje o klastrze w razie potrzeby. Aby to zrobić, program IBM MQ Explorer musi mieć następujące informacje:

- Nazwa menedżera kolejek repozytorium
- Nazwa połączenia menedżera kolejek repozytorium, jeśli znajduje się on w zdalnym menedżerze kolejek

Dzięki tym informacjom produkt IBM MQ Explorer może:

- Za pomocą menedżera kolejek repozytorium można uzyskać listę menedżerów kolejek w klastrze.

- Administruj menedżerami kolejek, które są elementami klastra, i znajdują się na obsługiwanych platformach i poziomach komend.

Podawanie nie jest możliwe, jeśli:

- Wybrane repozytorium stanie się niedostępne. Produkt IBM MQ Explorer nie przetacza się automatycznie na alternatywne repozytorium.
- Nie można skontaktować się z wybranym repozytorium za pośrednictwem protokołu TCP/IP.
- Wybrane repozytorium działa w menedżerze kolejek działającym na platformie i na poziomie komend, które nie są obsługiwane przez produkt IBM MQ Explorer.

Elementy klastra, które mogą być administrowane, mogą być lokalne lub mogą być zdalne, jeśli można się z nim skontaktować za pomocą protokołu TCP/IP. IBM MQ Explorer łączy się z menedżerami kolejek lokalnych, które są elementami klastra bezpośrednio, bez użycia połączenia klienckiego.

Konwersja danych dla IBM MQ Explorer

IBM MQ Explorer działa w CCSID 1208 (UTF-8). Dzięki temu program IBM MQ Explorer może poprawnie wyświetlać dane ze zdalnych menedżerów kolejek. Niezależnie od tego, czy połączenie z menedżerem kolejek jest nawiązane bezpośrednio, czy przy użyciu pośredniego menedżera kolejek, produkt IBM MQ Explorer wymaga, aby wszystkie komunikaty przychodzące były przekształcane w identyfikator CCSID 1208 (UTF-8).

Jeśli użytkownik spróbuje nawiązać połączenie między serwerem IBM MQ Explorer i menedżerem kolejek z identyfikatorem CCSID, którego nie rozpoznaje IBM MQ Explorer, zostanie wyświetlony komunikat o błędzie.

Obsługiwane konwersje są opisane w sekcji [Konwersja stron kodowych](#).

Windows Korzystanie z aplikacji paska zadań IBM MQ (tylko w wersji Windows)

Aplikacja paska zadań programu IBM MQ wyświetla ikonę na pasku zadań serwera Windows na serwerze. Ikona udostępnia bieżący status produktu IBM MQ oraz menu, z którego można wykonywać proste czynności.

W systemie Windows ikona IBM MQ znajduje się na podajniku systemowym na serwerze i jest nakładana na symbol o kodzie oznaczonym kolorami, który może mieć jeden z następujących znaczeń:

Zielony

Praca poprawnie; brak alertów w chwili obecnej

Niebieski

Nieokreślone; IBM MQ jest uruchamiany lub zamykany

Żółty

Alert; jedna lub więcej usług kończy się niepowodzeniem lub nie powiodło się

Aby wyświetlić menu, kliknij prawym przyciskiem myszy ikonę IBM MQ. Z poziomu menu można wykonać następujące czynności:

- Kliknij przycisk **Otwórz**, aby otworzyć monitor alertów programu IBM MQ.
- Kliknij przycisk **Wyjdź**, aby wyjść z aplikacji IBM MQ Taskbar.
- Kliknij przycisk **IBM MQ Explorer**, aby uruchomić IBM MQ Explorer.
- Kliknij przycisk **Zatrzymaj IBM MQ**, aby zatrzymać produkt IBM MQ.
- Kliknij **Informacje o produkcie IBM MQ**, aby wyświetlić informacje o monitorze alertów programu IBM MQ.

Monitor alertów IBM MQ to narzędzie do wykrywania błędów, które identyfikuje i rejestruje problemy z produktem IBM MQ na komputerze lokalnym.

Monitor alertów wyświetla informacje na temat bieżącego statusu instalacji lokalnej serwera IBM MQ. Monitoruje on również interfejs ACPI (Windows Advanced Configuration and Power Interface) i zapewnia wymuszanie ustawień ACPI.

Z poziomu monitora alertów programu IBM MQ można:

- Uzyskaj bezpośredni dostęp do IBM MQ Explorer
- Wyświetlanie informacji dotyczących wszystkich zaległych alertów
- Wyłącz usługę IBM MQ na komputerze lokalnym
- Kierowanie komunikatów o alertach za pośrednictwem sieci do konfigurowalnego konta użytkownika lub do stacji roboczej lub serwera Windows

Administrowanie lokalnymi obiektami IBM MQ

Istnieje możliwość administrowania lokalnymi obiektami produktu IBM MQ w celu obsługi aplikacji, które używają interfejsu kolejki komunikatów (Message Queue Interface-MQI).

O tym zadaniu

W tym kontekście lokalne administrowanie oznacza tworzenie, wyświetlanie, zmienianie, kopiowanie i usuwanie obiektów IBM MQ.

Oprócz metod opisanych w tej sekcji można użyć programu IBM MQ Explorer do administrowania lokalnymi obiektami produktu IBM MQ. Więcej informacji na ten temat zawiera sekcja [“Administrowanie przy użyciu IBM MQ Explorer”](#) na stronie 134.

Procedura

- Informacje zawarte w poniższych tematach ułatwiają administrowanie lokalnymi obiektami produktu IBM MQ.
 - [Programy aplikacji korzystające z interfejsu MQI](#)
 - [“Administrowanie programem MQ za pomocą komend MQSC”](#) na stronie 10
 - [“Praca z menedżerami kolejek”](#) na stronie 147
 - [“Praca z kolejkami lokalnymi”](#) na stronie 149
 - [“Praca z kolejkami aliasami”](#) na stronie 154
 - [“Praca z kolejkami modelami”](#) na stronie 175
 - [“Praca z usługami”](#) na stronie 183
 - [“Zarządzanie obiektami w celu wyzwolenia”](#) na stronie 191

Uruchamianie i zatrzymywanie menedżera kolejek

Wprowadzenie do zatrzymywania i uruchamiania menedżera kolejek z odsyłaczami do dalszych informacji.

O tym zadaniu

Menedżer kolejek można uruchamiać i zatrzymując za pomocą komend:

- Aby uruchomić menedżer kolejek, należy użyć komendy **strmqm**.

- Aby zatrzymać menedżer kolejek, należy użyć komendy **endmqm**, która udostępnia trzy sposoby zatrzymywania menedżera kolejek: kontrolowane lub wyciszone, zamknięcie systemu, natychmiastowe zamknięcie systemu i zamknięcie z góry.

Windows **Linux** Alternatywnie w systemach Windows i Linux można użyć IBM MQ Explorer, aby uruchomić i zatrzymać menedżer kolejek.

Windows W systemie Windows można skonfigurować menedżer kolejek, który będzie uruchamiany automatycznie podczas uruchamiania systemu za pomocą konsoli IBM MQ Explorer.

Procedura

1. Więcej informacji na temat uruchamiania menedżera kolejek zawiera sekcja [Uruchamianie menedżera kolejek](#).
2. Więcej informacji na temat zatrzymywania menedżera kolejek zawiera sekcja [Zatrzymywanie menedżera kolejek](#).

ULW Ręczne zatrzymywanie menedżera kolejek

Jeśli standardowe metody zatrzymywania i usuwania menedżera kolejek nie powiodą się, można spróbować zatrzymać menedżer kolejek ręcznie.

O tym zadaniu

Standardowym sposobem zatrzymywania menedżerów kolejek jest użycie komendy **endmqm** zgodnie z opisem w sekcji [Zatrzymywanie menedżera kolejek](#). Jeśli nie jest możliwe zatrzymanie menedżera kolejek w standardowy sposób, można spróbować zatrzymać menedżer kolejek ręcznie. Sposób, w jaki to robisz, zależy od platformy, którą używasz.

Procedura

- **Windows**
Informacje na temat zatrzymywania menedżera kolejek w systemie Windows zawiera sekcja [“Ręczne zatrzymywanie menedżera kolejek w systemie Windows”](#) na stronie 144.
- **Linux** **UNIX**
Aby zatrzymać menedżera kolejek w systemie UNIX lub Linux, należy zapoznać się z [“Ręczne zatrzymywanie menedżera kolejek w systemie UNIX i Linux”](#) na stronie 145.

Informacje pokrewne

[Tworzenie i zarządzanie menedżerami kolejek na wielu platformach](#)
[endmqm](#)

Windows Ręczne zatrzymywanie menedżera kolejek w systemie Windows

Jeśli nie można zatrzymać menedżera kolejek w systemie Windows za pomocą komendy **endmqm**, można spróbować ręcznie zatrzymać menedżer kolejek, kończąc wszystkie działające procesy i zatrzymując usługę IBM MQ.

O tym zadaniu

Wskazówka: Menedżer zadań produktu Windows i komenda **tasklist** udzieli ograniczonych informacji na temat zadań. Więcej informacji na temat określania, które procesy dotyczą konkretnego menedżera kolejek, można znaleźć przy użyciu narzędzia takiego jak *Process Explorer* (`procexp.exe`), które jest dostępne do pobrania z serwisu WWW produktu Microsoft pod adresem <https://www.microsoft.com>.

Aby zatrzymać menedżer kolejek w systemie Windows, wykonaj następujące kroki.

Procedura

1. Lista nazw (identyfikatorów) procesów, które są uruchomione, za pomocą menedżera zadań produktu Windows .
2. Zakończ procesy za pomocą programu Windows Task Manager lub komendy **taskkill** w następującej kolejności (jeśli są uruchomione):

Tabela 12. Procesy Windows , które mają zostać zatrzymane, jeśli są uruchomione

Nazwa procesu	Opis
AMQZMUC0	Menedżer procesu krytycznego
AMQZXMA0	Kontroler wykonywania
AMQZFUMA	Proces OAM
AMQZLAA0	Agenty LQM
AMQZLSA0	Agenty LQM
AMQZMUFO	Menedżer narzędzi
AMQZMGRO	Kontroler procesów
AMQZMUR0	Restartowalny menedżer procesu
AMQFQ PUB	Proces publikowania subskrybowania
AMQFCXBA	Proces roboczy brokera
AMQRMPPA	Proces zestawiania procesów
AMQCRSTA	Proces zadania, który nie jest wątkiem
AMQCRS6B	Kanał odbiorczy LU62 i połączenie klienta
AMQRRMFA	Proces repozytorium (dla klastrów)
AMQPCSEA	Serwer komend
RUNMQTRM	Wywołaj monitor wyzwalacza dla serwera
RUNMQDLQ	Wywołaj procedurę obsługi kolejki niedostarczonych komunikatów
RUNMQCHI	Proces inicjatora kanału
RUNMQLSR	Proces nastuchiwania kanału
AMQXSSVN	Współużytkowane serwery pamięci

3. Zatrzymaj usługę IBM MQ z poziomu opcji **Narzędzia administracyjne > Usługi** na panelu sterowania programu Windows .
4. Jeśli próbowano wszystkich metod, a menedżer kolejek nie został zatrzymany, zrestartuj system.

Linux > UNIX **Ręczne zatrzymywanie menedżera kolejek w systemie UNIX**

i Linux

Jeśli nie można zatrzymać menedżera kolejek w systemie UNIX i Linux za pomocą komendy **endmqm** , można spróbować ręcznie zatrzymać menedżer kolejek, kończąc wszystkie uruchomione procesy i zatrzymując usługę IBM MQ .

O tym zadaniu

Aby zatrzymać menedżer kolejek w systemie UNIX i Linux, wykonaj następujące kroki.

Jeśli menedżer kolejek zostanie zatrzymany ręcznie, to mogą zostać pobrane FFST i pliki FDC umieszczone w produkcie `/var/mqm/errors`. Nie należy go traktować jako defektu w menedżerze kolejek.

Menedżer kolejek zostanie zrestartowany normalnie, nawet po jego zatrzymaniu przy użyciu tej metody zatrzymania ręcznie.

Procedura

1. Znajdź identyfikatory procesów dla programów menedżera kolejek, które nadal działają, za pomocą komendy **ps**.

Na przykład, jeśli menedżer kolejek ma nazwę QMNAME, użyj następującej komendy:

```
ps -ef | grep QMNAME
```

2. Zakończ wszystkie procesy menedżera kolejek, które nadal działają, za pomocą komendy **kill**, określając identyfikatory PID wykryte za pomocą komendy **ps**.

Aby zakończyć proces, należy użyć komendy **kill -KILL <pid>** lub równoważnej komendy **kill -9 <pid>**.

Musisz pracować przez PID, które chcesz zabić, po jednym, wydając tę komendę za każdym razem.

Ważne: Jeśli używany jest dowolny sygnał inny niż **9 (SIGKILL)**, proces prawdopodobnie nie zostanie zatrzymany, a wyniki będą nieprzewidywalne.

Zakończ procesy w następującej kolejności:

<i>Tabela 13. Procesy UNIX i Linux, które mają zostać zatrzymane, jeśli są uruchomione</i>	
Nazwa procesu	Opis
amqzmuc0	Menedżer procesu krytycznego
amqzma0	Kontroler wykonywania
amqzfuma	Proces OAM
amqzlaa0	Agenty LQM
amqzlsa0	Agenty LQM
amqzmuf0	Menedżer narzędzi
amqzmur0	Restartowalny menedżer procesu
amqzmgr0	Kontroler procesów
amqfqpub	Proces publikowania subskrybowania
amqfcxba	Proces roboczy brokera
amqrmppa	Proces zestawiania procesów
amqcrsta	Proces zadania, który nie jest wątkiem
amqcrs6b	Kanał odbiorczy LU62 i połączenie klienta
amqrrmfa	Proces repozytorium (dla klastrów)
amqpcsea	Serwer komend
runmqtrm	Wywołaj monitor wyzwalacza dla serwera
runmqdlq	Wywołaj procedurę obsługi kolejki niedostarczonych komunikatów
runmqchi	Proces inicjatora kanału

Tabela 13. Procesy UNIX i Linux , które mają zostać zatrzymane, jeśli są uruchomione (kontynuacja)	
Nazwa procesu	Opis
runmqtsr	Proces nastuchiwania kanału

Uwaga: Komendy **kill -9** można użyć do zakończenia procesów, które nie zostały zatrzymane.

Zatrzymywanie kanałów MQI

Po wydaniu komendy STOP CHANNEL dla kanału połączenia z serwerem można wybrać metodę, która ma zostać użyta do zatrzymania kanału połączenia klienckiego. Oznacza to, że kanał klienta, który wysyła wywołanie funkcji MQGET, może być sterowany, a użytkownik może zdecydować, w jaki sposób i kiedy zatrzymać kanał.

Komendę STOP CHANNEL można wydać w trzech trybach, wskazując, w jaki sposób kanał ma być zatrzymany:

Wyciszenie

Zatrzymuje kanał po przetworzeniu wszystkich bieżących komunikatów.

Jeśli współużytkowanie konwersacji jest włączone, program IBM MQ MQI client w odpowiednim czasie uzyska informacje o żądaniu zatrzymania. Ten czas jest zależny od szybkości sieci. Aplikacja kliencka zapoznaje się z żądaniem zatrzymania w wyniku wydania kolejnego wywołania do produktu IBM MQ.

Wymuszenie

Zatrzymuje kanał natychmiast.

Przerwij

Zatrzymuje kanał natychmiast. Jeśli kanał działa jako proces, może zakończyć proces kanału lub, jeśli kanał działa jako wątek, jego wątek.

Jest to proces wieloetapowy. Jeśli używany jest tryb zakończenia, podejmowana jest próba zatrzymania kanału połączenia z serwerem, najpierw w trybie wygaszania trybu, a następnie z wymuszonym działaniem trybu, a w razie potrzeby z trybem zakończenia. Klient może odbierać różne kody powrotu podczas różnych etapów kończenia pracy. Jeśli proces lub wątek zostanie zakończony, klient otrzymuje błąd komunikacji.

Kody powrotu zwracane do aplikacji różnią się w zależności od wywołanego wywołania MQI i komendy STOP CHANNEL. Klient otrzyma albo komendę MQRC_CONNECTION_QUIESCING, albo kod powrotu MQRC_CONNECTION_BROKEN. Jeśli klient wykryje opcję MQRC_CONNECTION_QUIESCING, powinien on spróbować zakończyć bieżącą transakcję i zakończyć działanie. Nie jest to możliwe przy użyciu komendy MQRC_CONNECTION_BROKEN. Jeśli klient nie zrealizuje transakcji i zakończy działanie wystarczająco szybko, po kilku sekundach zostanie zerwany z CONNECTION_BROKEN. Komenda STOP CHANNEL z trybem MODE (FORCE) lub MODE (TERMINATE) prawdopodobnie spowoduje ustawienie CONNECTION_BROKEN niż w trybie MODE (QUIESCE).

Informacje pokrewne

[Kanały](#)

Praca z menedżerami kolejek

Przykłady komend MQSC, których można używać do wyświetlania lub modyfikowania atrybutów menedżera kolejek.

Wyświetlanie atrybutów menedżera kolejek

Aby wyświetlić atrybuty menedżera kolejek określonego w komendzie **runmqsc** , należy użyć komendy MQSC **DISPLAY QMGR** :

```
DISPLAY QMGR
```

```

DISPLAY QMGR
  1 : DISPLAY QMGR
AMQ8408: Display Queue Manager details.
QMNAME(QM1)
ACCTINT(1800)
ACCTQ(OFF)
ACTVCONO (DISABLED)
ALTDATE(2012-05-27)
AUTHOREV(DISABLED)
CHAD(DISABLED)
CHADEXIT( )
CLWLDATA( )
CLWLLEN(100)
CLWLUSEQ(LOCAL)
CMDLEVEL(800)
CONFIGEV(DISABLED)
CRTIME(16.14.01)
DEFXMITQ( )
DISTL(YES)
IPADDRV(IPV4)
LOGGEREV(DISABLED)
MAXHANDS(256)
MAXPROPL(NOLIMIT)
MAXUMSGS(10000)
MONCHL(OFF)
PARENT( )
PLATFORM(WINDOWSNT)
PSNPMMSG(DISCARD)
PSSYNCP( IFFER)
PSMODE(ENABLED)
REPOS( )
ROUTEREC(MSG)
SCMDSERV(QMGR)
SSLCRYP( )
SSLFIPS(NO)
MQ\Data\qmgrs\QM1\ssl\key)
SSLKEYC(0)
STATCHL(OFF)
STATMQI(OFF)
STRSTPEV(ENABLED)
TREELIFE(1800)
ACCTCONO(DISABLED)
ACCTMQI(OFF)
ACTIVREC(MSG)
ACTVTRC(OFF)
ALTTIME(16.14.01)
CCSID(850)
CHADEV(DISABLED)
CHLEV(DISABLED)
CLWLEXIT( )
CLWLMRUC(999999999)
CMDEV(DISABLED)
COMMANDQ(SYSTEM.ADMIN.COMMAND.QUEUE)
CRDATE(2011-05-27)
DEADQ( )
DESCR( )
INHIBTEV(DISABLED)
LOCALEV(DISABLED)
MARKINT(5000)
MAXMSGL(4194304)
MAXPRTY(9)
MONACLS(QMGR)
MONQ(OFF)
PERFMEV(DISABLED)
  PSRTYCNT(5)
PSNPRES(NORMAL)
QMID(QM1_2011-05-27_16.14.01)
REMOETEVEV(DISABLED)
REPOSNL( )
SCHINIT(QMGR)
SSLCRLNL( )
SSLEV(DISABLED)
SSLKEYR(C:\Program Files\IBM\WebSphere
STATACLS(QMGR)
STATINT(1800)
STATQ(OFF)
SYNCP
TRIGINT(999999999)

```

Rysunek 16. Typowe dane wyjściowe komendy DISPLAY QMGR

Uwaga: SYNCP jest atrybutem menedżera kolejek tylko do odczytu.

Parametr **ALL** jest wartością domyślną w komendzie **DISPLAY QMGR**. Wyświetla on wszystkie atrybuty menedżera kolejek. W szczególności dane wyjściowe informują o domyślnej nazwie menedżera kolejek, nazwie kolejki niedostarczonych komunikatów oraz o nazwie kolejki komend.

Można sprawdzić, czy te kolejki istnieją, wprowadzając komendę:

```
DISPLAY QUEUE (SYSTEM.*)
```

Spowoduje to wyświetlenie listy kolejek zgodnych z łodyg SYSTEM.*. Nawiasy są wymagane.

Zmiana atrybutów menedżera kolejek

Aby zmienić atrybuty menedżera kolejek określonego w komendzie **runmqsc**, należy użyć komendy **MQSC ALTER QMGR**, określając atrybuty i wartości, które mają zostać zmienione. Na przykład można użyć następujących komend, aby zmienić atrybuty produktu `jupiter.queue.manager`:

```
runmqsc jupiter.queue.manager
ALTER QMGR DEADQ (ANOTHERDLQ) INHIBTEV (ENABLED)
```

Komenda **ALTER QMGR** służy do zmiany używanej kolejki niedostarczonych komunikatów, a także umożliwia zahamowanie zdarzeń.

Informacje pokrewne

[Tworzenie i zarządzanie menedżerami kolejek na wielu platformach](#)

[Atrybuty dla menedżera kolejek](#)

[runmqsc \(uruchamianie komend MQSC\)](#)

[WYŚWIETL QMGR](#)

[ALTER QMGR](#)

Praca z kolejkami lokalnymi

Ta sekcja zawiera przykłady niektórych komend MQSC, których można używać do zarządzania kolejkami lokalnymi, modelowymi i aliasowymi.

Szczegółowe informacje na temat tych komend zawiera sekcja [Komendy MQSC](#).

Informacje pokrewne

[Ograniczenia nazewnictwa dla kolejek](#)

[Ograniczenia nazewnictwa dla innych obiektów](#)

Definiowanie kolejki lokalnej

W przypadku aplikacji lokalny menedżer kolejek jest menedżerem kolejek, z którym połączona jest aplikacja. Kolejki zarządzane przez lokalny menedżer kolejek są lokalne względem tego menedżera kolejek.

Aby utworzyć kolejkę lokalną, należy użyć komendy MQSC **DEFINE QLOCAL**. Można również użyć wartości domyślnej zdefiniowanej w domyślnej definicji kolejki lokalnej lub można zmodyfikować charakterystykę kolejki z domyślnej lokalnej kolejki.

Uwaga: Domyślna kolejka lokalna ma nazwę SYSTEM.DEFAULT.LOCAL.QUEUE i została utworzona w instalacji systemu.

Na przykład komenda **DEFINE QLOCAL**, która jest następująca, definiuje kolejkę o nazwie ORANGE.LOCAL.QUEUE o następujących parametrach:

- Jest on włączony dla operacji pobierania, włączony dla operacji put i działa w oparciu o kolejność priorytetów.
- Jest to kolejka *normalna*. Nie jest to kolejka inicjujący ani kolejka transmisji, a nie generuje komunikatów wyzwacza.
- Maksymalna głębokość kolejki to 5000 komunikatów. Maksymalna długość komunikatu to 4194304 bajtów.

```
DEFINE QLOCAL(ORANGE.LOCAL.QUEUE) +
DESCR('Queue for messages from other systems') +
PUT(ENABLED) +
GET(ENABLED) +
NOTRIGGER +
MSGDLVSQ(PRIORITY) +
MAXDEPTH(5000) +
MAXMSGL(4194304) +
USAGE(NORMAL)
```

Uwaga:

1. Z wyjątkiem wartości dla opisu, wszystkie wartości atrybutów wyświetlane w przykładzie są wartościami domyślnymi. Przykłady te są dołączane do ilustracji purposesyes zdecydowanie. Można je pominąć, jeśli użytkownik ma pewność, że wartości domyślne są takie, które mają zostać zmienione lub które nie zostały zmienione. Patrz także ["Wyświetlanie domyślnych atrybutów obiektów"](#) na stronie 150.
2. **USAGE(NORMAL)** wskazuje, że ta kolejka nie jest kolejką transmisji.

3. Jeśli w tym samym menedżerze kolejek znajduje się już kolejka lokalna o nazwie ORANGE.LOCAL.QUEUE. Ta komenda nie powiedzie się. Użyj atrybutu **REPLACE** , jeśli chcesz nadpisać istniejącą definicję kolejki, ale zobacz także [“Zmiana atrybutów kolejki lokalnej”](#) na stronie 151.

Informacje pokrewne

[DEFINE QLOCAL](#)

Wyświetlanie domyślnych atrybutów obiektów

Komendy **DISPLAY QUEUE** można użyć do wyświetlenia atrybutów, które zostały pobrane z obiektu domyślnego podczas definiowania obiektu IBM MQ .

Podczas definiowania obiektu IBM MQ przyjmuje on wszystkie atrybuty, które nie zostały określone na podstawie obiektu domyślnego. Na przykład podczas definiowania kolejki lokalnej kolejka dziedziczy wszystkie atrybuty, które są pomijane w definicji z domyślnej kolejki lokalnej, która jest nazywana SYSTEM.DEFAULT.LOCAL.QUEUE. Aby dokładnie sprawdzić, jakie są te atrybuty, należy użyć następującej komendy:

```
DISPLAY QUEUE (SYSTEM.DEFAULT.LOCAL.QUEUE)
```

Składnia tej komendy różni się od składni odpowiadającej jej komendy **DEFINE** . W komendzie **DISPLAY** można podać tylko nazwę kolejki, podczas gdy w komendzie **DEFINE** należy określić typ kolejki, czyli QLOCAL, QALIAS, QMODEL lub QREMOTE.

Można selektywnie wyświetlać atrybuty, określając je pojedynczo. Na przykład:

```
DISPLAY QUEUE (ORANGE.LOCAL.QUEUE) +  
MAXDEPTH +  
MAXMSGL +  
CURDEPTH;
```

Ta komenda wyświetla trzy określone atrybuty w następujący sposób:

```
AMQ8409: Display Queue details.  
QUEUE (ORANGE.LOCAL.QUEUE)      TYPE (QLOCAL)  
CURDEPTH (0)                     MAXDEPTH (5000)  
MAXMSGL (4194304)
```

CURDEPTH to bieżąca głębokość kolejki, to znaczy liczba komunikatów w kolejce. Jest to przydatny atrybut do wyświetlenia, ponieważ monitorując głębokość kolejki, można się upewnić, że kolejka nie jest pełna.

Informacje pokrewne

[WYŚWIETL KOLEJKĘ](#)

[Kolejki DEFINE](#)

Kopiowanie definicji kolejki lokalnej

Definicję kolejki można skopiować za pomocą atrybutu **LIKE** w komendzie **DEFINE** .

Na przykład:

```
DEFINE QLOCAL (MAGENTA.QUEUE) +  
LIKE (ORANGE.LOCAL.QUEUE)
```

Ta komenda tworzy kolejkę o takich samych atrybutach, jak w przypadku oryginalnej kolejki ORANGE.LOCAL.QUEUE, a nie w domyślnej lokalnej kolejce systemowej. Wpisz nazwę kolejki, która ma zostać skopiowana dokładnie tak, jak została wprowadzona podczas tworzenia kolejki. Jeśli nazwa zawiera małe litery, należy ująć ją w znaki pojedynczego cudzysłowu.

Można również użyć tego formularza komendy **DEFINE** , aby skopiować definicję kolejki, ale zastąpić jedną lub więcej zmian w atrybutach oryginału. Na przykład:

```
DEFINE QLOCAL (THIRD.QUEUE) +  
LIKE (ORANGE.LOCAL.QUEUE) +  
MAXMSGL(1024) ;
```

Ta komenda kopiuje atrybuty kolejki ORANGE.LOCAL.QUEUE do kolejki THIRD.QUEUE, ale określa, że maksymalna długość komunikatu w nowej kolejce ma wynosić 1024 bajty, a nie 4194304.

Uwaga:

1. Jeśli w komendzie **DEFINE** używany jest atrybut **LIKE** , kopiowane są tylko atrybuty kolejki. Nie kopujesz komunikatów w kolejce.
2. W przypadku zdefiniowania kolejki lokalnej, bez określania parametru **LIKE**, jest ona taka sama, jak zdefiniowana w definicji **DEFINE LIKE (SYSTEM.DEFAULT.LOCAL.QUEUE)**.

Informacje pokrewne

[Kolejki DEFINE](#)

Zmiana atrybutów kolejki lokalnej

Atrybuty kolejki można zmieniać na dwa sposoby, używając komendy **ALTER QLOCAL** lub komendy **DEFINE QLOCAL** z atrybutem **REPLACE** .

W programie [“Definiowanie kolejki lokalnej”](#) na stronie 149 jest to kolejka o nazwie ORANGE.LOCAL.QUEUE została zdefiniowana. Przypuśćmy na przykład, że chcesz zmniejszyć maksymalną długość komunikatu w tej kolejce do 10 000 bajtów.

- Za pomocą komendy **ALTER** :

```
ALTER QLOCAL (ORANGE.LOCAL.QUEUE) MAXMSGL(10000)
```

Ta komenda zmienia jeden atrybut o maksymalną długość komunikatu. Wszystkie pozostałe atrybuty pozostają takie same.

- Za pomocą komendy **DEFINE** z opcją **REPLACE** , na przykład:

```
DEFINE QLOCAL (ORANGE.LOCAL.QUEUE) MAXMSGL(10000) REPLACE
```

Ta komenda zmienia nie tylko maksymalną długość komunikatu, ale także wszystkie inne atrybuty, które mają podane wartości domyślne. Kolejka jest teraz włączona, podczas gdy poprzednio została ona zablokowana. Włączona opcja put jest wartością domyślną, określoną w kolejce SYSTEM.DEFAULT.LOCAL.QUEUE.

Zmniejszenie maksymalnej długości komunikatu w istniejącej kolejce powoduje, że nie ma to wpływu na istniejące komunikaty. Wszystkie nowe komunikaty muszą jednak spełniać nowe kryteria.

Informacje pokrewne

[Kolejki ALTER](#)

[ALTER QLOCAL](#)

[Kolejki DEFINE](#)

[DEFINE QLOCAL](#)

Czyszczenie kolejki lokalnej

Aby wyczyścić kolejkę lokalną, można użyć komendy **CLEAR** .

Na przykład, aby usunąć wszystkie komunikaty z kolejki lokalnej o nazwie MAGENTA.QUEUE, użyj następującej komendy:

```
CLEAR QLOCAL (MAGENTA.QUEUE)
```

Uwaga: Nie ma podpowiedzi, która pozwala Ci zmienić zdanie; po naciśnięciu klawisza Enter wiadomości zostaną utracone.

Nie można wyczyścić kolejki, jeśli:

- Istnieją niezatwierdzone komunikaty, które zostały umieszczone w kolejce w punkcie synchronizacji.
- Aplikacja aktualnie ma otwartą kolejkę.

Informacje pokrewne

[CLEAR QLOCAL](#)

Usuwanie kolejki lokalnej

Aby usunąć kolejkę lokalną, można użyć komendy MQSC **DELETE QLOCAL** .

Nie można usunąć kolejki, jeśli na niej nie ma zatwierdzonych komunikatów. Jeśli jednak kolejka ma jeden lub więcej zatwierdzonych komunikatów i nie ma żadnych niezatwierdzonych komunikatów, można ją usunąć tylko wtedy, gdy zostanie podana opcja **PURGE** . Na przykład:

```
DELETE QLOCAL (PINK.QUEUE) PURGE
```

Określenie **NOPURGE** zamiast **PURGE** zapewnia, że kolejka nie zostanie usunięta, jeśli zawiera jakiegokolwiek zatwierdzone komunikaty.

Informacje pokrewne

[USUŃ QLOCAL](#)

Przeglądanie kolejek

Produkt IBM MQ udostępnia przykładową przeglądarkę kolejek, której można używać do przeglądania treści komunikatów w kolejce. Przeglądarka jest dostarczana zarówno w formacie źródłowym, jak i w formacie wykonywalnym.

MQ_INSTALLATION_PATH reprezentuje katalog najwyższego poziomu, w którym zainstalowany jest produkt IBM MQ .

Windows W systemie Windows nazwy plików i ścieżki dla przeglądarki kolejki przykładowej są następujące:

Źródło

```
MQ_INSTALLATION_PATH\tools\c\samples\
```

Wykonywalny

```
MQ_INSTALLATION_PATH\tools\c\samples\bin\amqsbcg.exe
```

Linux **UNIX** W systemach UNIX i Linux nazwy plików i ścieżki są następujące:

Źródło

```
MQ_INSTALLATION_PATH/samp/amqsbcg0.c
```

Wykonywalny

```
MQ_INSTALLATION_PATH/samp/bin/amqsbcg
```

Przykład wymaga dwóch parametrów wejściowych, nazwy kolejki i nazwy menedżera kolejek. Na przykład:

```
amqsbcg SYSTEM.ADMIN.QMGREVENT.tpp01 saturn.queue.manager
```

Typowe wyniki tej komendy są przedstawione w sekcji [Rysunek 17 na stronie 153](#).


```

AMQSBCG0 - starts here
*****

MQOPEN - 'SYSTEM.ADMIN.QMGR.EVENT'

MQGET of message number 1
****Message descriptor****

  StrucId : 'MD ' Version : 2
  Report  : 0 MsgType : 8
  Expiry  : -1 Feedback : 0
  Encoding : 546 CodedCharSetId : 850
  Format   : 'MQEVENT '
  Priority : 0 Persistence : 0
  MsgId   : X'414D512073617475726E2E71756575650005D30033563DB8'
  CorrelId : X'0000000000000000000000000000000000000000000000000000'
  BackoutCount : 0
  ReplyToQ      : '
  ReplyToQMgr   : 'saturn.queue.manager'
  ** Identity Context
  UserIdentifier : '
  AccountingToken :
  X'0000000000000000000000000000000000000000000000000000000000000000'
  ApplIdentityData : '
  ** Origin Context
  PutApplType : '7'
  PutApplName : 'saturn.queue.manager'
  PutDate : '19970417' PutTime : '15115208'
  ApplOriginData : '

  GroupId : X'00000000000000000000000000000000000000000000000000000'
  MsgSeqNumber : '1'
  Offset : '0'
  MsgFlags : '0'
  OriginalLength : '104'

**** Message ****

length - 104 bytes

00000000: 0700 0000 2400 0000 0100 0000 2C00 0000 '.....→.....'
00000010: 0100 0000 0100 0000 0100 0000 AE08 0000 '.....'
00000020: 0100 0000 0400 0000 4400 0000 DF07 0000 '.....D.....'
00000030: 0000 0000 3000 0000 7361 7475 726E 2E71 '...0...saturn.q'
00000040: 7565 7565 2E6D 616E 6167 6572 2020 2020 'ueue.manager'
00000050: 2020 2020 2020 2020 2020 2020 2020 2020 '
00000060: 2020 2020 2020 2020 '

No more messages
MQCLOSE
MQDISC

```

Rysunek 17. Typowe wyniki w przeglądarce kolejek

Informacje pokrewne

[Przykładowy program przeglądarki](#)

Włączanie dużych kolejek

IBM MQ obsługuje kolejki większe niż 2 GB.

Windows

W systemach Windows obsługa dużych plików jest dostępna bez dodatkowego włączenia.

Linux

UNIX

W systemach UNIX i Linux należy jawnie włączyć obsługę dużych plików, zanim możliwe będzie utworzenie plików kolejki większych niż 2 GB. Informacje na temat sposobu wykonania tej czynności można znaleźć w dokumentacji systemu operacyjnego.

Niektóre programy narzędziowe, takie jak tar, nie mogą sobie poradzić z plikami większymi niż 2 GB. Przed włączeniem obsługi dużych plików należy zapoznać się z dokumentacją systemu operacyjnego, aby uzyskać informacje na temat ograniczeń w użyciu programów narzędziowych.

Informacje na temat planowania ilości pamięci masowej dla kolejek można znaleźć w sekcji [IBM MQ Family - Performance Reports](#) dla raportów dotyczących wydajności dla poszczególnych platform.

Praca z kolejkami aliasami

Kolejkę aliasową można zdefiniować w taki sposób, aby odwoływały się pośrednio do innej kolejki lub tematu.

V9.0.0.1 V9.0.1



Ostrzeżenie: Listy dystrybucyjne nie obsługują korzystania z kolejek aliasowych, które wskazują na obiekty tematów. W przypadku serwerów IBM MQ 9.0.1 i IBM MQ 9.0.0 Fix Pack 1, jeśli kolejka aliasowa wskazuje obiekt tematu na liście dystrybucyjnej, program IBM MQ zwraca wartość MQRC_ALIAS_BASE_Q_TYPE_ERROR.

Kolejka, do której odnosi się kolejka aliasowa, może być dowolna z następujących:

- Kolejka lokalna (patrz [“Definiowanie kolejki lokalnej”](#) na stronie 149).
- Lokalna definicja kolejki zdalnej (patrz [“Tworzenie lokalnej definicji kolejki zdalnej”](#) na stronie 204).
- Temat.

Kolejka aliasowa nie jest kolejką rzeczywistą, ale definicją, która jest tłumaczona na rzeczywistą (lub docelową) kolejkę w czasie wykonywania. Definicja kolejki aliasowej określa kolejkę docelową. Gdy aplikacja tworzy wywołanie komendy MQOPEN z kolejką aliasową, menedżer kolejek rozstrzygnie alias do docelowej nazwy kolejki.

Kolejka aliasowa nie może być przetłumaczona na inną lokalnie zdefiniowaną kolejkę aliasową. Jednak kolejka aliasowa może być tłumaczona na kolejki aliasowe, które są zdefiniowane w innych miejscach klastrów, do których należy lokalny menedżer kolejek. Więcej informacji na ten temat zawiera sekcja [Rozdzielczość nazw](#).

Kolejki aliasowe są przydatne w następujących celach:

- Nadawanie różnym aplikacjom różnych poziomów uprawnień dostępu do kolejki docelowej.
- Dopuszczenie różnych aplikacji do pracy z tą samą kolejką na różne sposoby. (Być może należy przypisać różne domyślne priorytety lub inne domyślne wartości trwałości).
- Uproszczone zarządzanie konserwacją, migracją i równoważeniem obciążenia. (Być może konieczna jest zmiana nazwy docelowej kolejki bez konieczności zmiany aplikacji, która nadal używa aliasu).

Na przykład założmy, że aplikacja została opracowana w celu umieszczenia komunikatów w kolejce o nazwie MY.ALIAS.QUEUE. Określa ona nazwę tej kolejki, gdy tworzy żądanie MQOPEN i pośrednio, jeśli umieszcza komunikat w tej kolejce. Aplikacja nie ma informacji o tym, że kolejka jest kolejką aliasową. Dla każdego wywołania MQI za pomocą tego aliasu menedżer kolejek rozstrzygnie rzeczywistą nazwę kolejki, która może być kolejką lokalną lub zdalną zdefiniowaną w tym menedżerze kolejek.

Zmieniając wartość atrybutu TARGQ, można przekierować wywołania MQI do innej kolejki, prawdopodobnie w innym menedżerze kolejek. Jest to przydatne w przypadku konserwacji, migracji i równoważenia obciążenia.

Definiowanie kolejki aliasowej

Następująca komenda tworzy kolejkę aliasową:

```
DEFINE QALIAS (MY.ALIAS.QUEUE) TARGET (YELLOW.QUEUE)
```

Ta komenda przekierowuje wywołania MQI, które określają MY.ALIAS.QUEUE do kolejki YELLOW.QUEUE. Komenda nie tworzy kolejki docelowej. Wywołanie MQI nie powiedzie się, jeśli kolejka YELLOW.QUEUE nie istnieje w czasie wykonywania.

Jeśli zmienisz definicję aliasu, możesz przekierować wywołania MQI do innej kolejki. Na przykład:

```
ALTER QALIAS (MY.ALIAS.QUEUE) TARGET (MAGENTA.QUEUE)
```

Ta komenda przekierowuje wywołania MQI do innej kolejki, MAGENTA.QUEUE.

Można również użyć kolejek aliasowych, aby utworzyć pojedynczą kolejkę (kolejka docelowa), która ma różne atrybuty dla różnych aplikacji. W tym celu należy zdefiniować dwa aliasy, po jednym dla każdej aplikacji. Załóżmy, że istnieją dwie aplikacje:

- Aplikacja ALPHA może umieszczać komunikaty w systemie YELLOW.QUEUE, ale nie można pobrać z niego komunikatów.
- Aplikacja BETA może pobrać komunikaty z obszaru YELLOW.QUEUE, ale nie można umieszczać na nim komunikatów.

Poniższa komenda definiuje alias, który jest włączony i został wyłączony dla aplikacji ALPHA:

```
DEFINE QALIAS (ALPHAS.ALIAS.QUEUE) +  
TARGET (YELLOW.QUEUE) +  
PUT (ENABLED) +  
GET (DISABLED)
```

Poniższa komenda definiuje alias, który jest wyłączony i został włączony dla aplikacji BETA:

```
DEFINE QALIAS (BETAS.ALIAS.QUEUE) +  
TARGET (YELLOW.QUEUE) +  
PUT (DISABLED) +  
GET (ENABLED)
```

ALPHA korzysta z nazwy kolejki ALPHAS.ALIAS.QUEUE w swoich wywołaniach MQI; BETA używa nazwy kolejki BETAS.ALIAS.QUEUE. Oba mają dostęp do tej samej kolejki, ale na różne sposoby.

Atrybutów LIKE i REPLACE można używać podczas definiowania aliasów kolejek w taki sam sposób, w jaki używane są atrybuty z kolejkami lokalnymi.

Korzystanie z innych komend z kolejkami aliasami

Za pomocą odpowiednich komend MQSC można wyświetlać lub zmieniać atrybuty kolejki aliasowej, a także usuwać obiekty kolejki aliasowej. Na przykład:

Użyj komendy **DISPLAY QALIAS**, aby wyświetlić atrybuty kolejki aliasowej:

```
DISPLAY QUEUE (ALPHAS.ALIAS.QUEUE)
```

Użyj komendy **ALTER QALIAS**, aby zmienić nazwę kolejki podstawowej, do której alias jest tłumaczona, w przypadku gdy opcja **FORCE** wymusza zmianę, nawet jeśli kolejka jest otwarta:

```
ALTER QALIAS (ALPHAS.ALIAS.QUEUE) TARGET(ORANGE.LOCAL.QUEUE) FORCE
```

Aby usunąć alias kolejki, należy użyć komendy **DELETE QALIAS**:

```
DELETE QALIAS (ALPHAS.ALIAS.QUEUE)
```

Nie można usunąć kolejki aliasowej, jeśli aplikacja aktualnie ma otwartą kolejkę.

Informacje pokrewne

[ALTER QALIAS](#)

[ZDEFINIUJ ALIAS QALIAS](#)

[USUŃ QALIAS](#)

[Lista dystrybucyjna](#)

Praca z kolejkami niedostarczonych komunikatów

Każdy menedżer kolejek zwykle ma kolejkę lokalną, która ma być używana jako kolejka niedostarczonych komunikatów, dzięki czemu komunikaty, których nie można dostarczyć do ich poprawnego miejsca docelowego, mogą zostać zapisane w celu późniejszego pobrania. Należy poinformować menedżera kolejek o kolejce niedostarczonych komunikatów, a także określić, w jaki sposób mają być przetwarzane komunikaty znalezione w kolejce niedostarczonych komunikatów. Użycie kolejek niedostarczonych komunikatów może mieć wpływ na kolejność dostarczania komunikatów, dlatego użytkownik może zdecydować, że nie będzie ich używać.

Aby poinformować menedżera kolejek o kolejce niedostarczonych komunikatów, należy podać nazwę kolejki niedostarczonych komunikatów w komendzie **crtmqm** (na przykład **crtmqm -u DEAD.LETTER.QUEUE**) lub za pomocą atrybutu **DEADQ** w komendzie **ALTER QMGR**, aby określić jedną z nich później. Przed użyciem kolejki niedostarczonych komunikatów należy zdefiniować kolejkę niedostarczonych komunikatów.

Przykładowa kolejka niedostarczonych komunikatów o nazwie SYSTEM.DEAD.LETTER.QUEUE jest dostępna razem z produktem. Ta kolejka jest tworzona automatycznie podczas tworzenia menedżera kolejek. Jeśli jest to wymagane, można zmodyfikować tę definicję i zmienić jej nazwę.

Kolejka niedostarczonych komunikatów nie ma specjalnych wymagań, z wyjątkiem sytuacji, gdy:

- Musi to być kolejka lokalna
- Atrybut MAXMSGL (maksymalna długość komunikatu) musi włączyć kolejkę w celu uwzględnienia największych komunikatów, które menedżer kolejek musi obsłużyć **plus** wielkość nagłówka niedostarczonych komunikatów (MQDLH)

Użycie kolejek niedostarczonych komunikatów może mieć wpływ na kolejność dostarczania komunikatów, dlatego użytkownik może zdecydować, że nie będzie ich używać. Atrybut kanału USEDQL ustawia się w celu określenia, czy kolejka niedostarczonych komunikatów jest używana, gdy komunikaty nie mogą być dostarczane. Ten atrybut można skonfigurować w taki sposób, aby niektóre funkcje menedżera kolejek używały kolejki niedostarczonych komunikatów, podczas gdy inne funkcje nie są dostępne. Więcej informacji na temat używania atrybutu kanału USEDQL w różnych komendach MQSC można znaleźć w sekcji [DEFINE CHANNEL\(DEFINE CHANNEL\)](#), [DISPLAY CHANNEL\(WYŚWIETL KANAŁ\)](#), [ALTER CHANNEL\(ALTER CHANNEL\)](#) i [DISPLAY CLUSQMGR\(DISPLAY CLUSQMGR\)](#).

Produkt IBM MQ udostępnia procedurę obsługi kolejki niedostarczonych komunikatów, która umożliwia określenie sposobu, w jaki komunikaty znalezione w kolejce niedostarczonych komunikatów mają być przetwarzane lub usuwane. Patrz sekcja [“Przetwarzanie komunikatów w kolejce niedostarczonych komunikatów programu IBM MQ”](#) na stronie 156.

Informacje pokrewne

[Kolejki niedostarczane](#)

[Rozwiązywanie problemów z niedostarczonym](#)

[ALTER QMGR](#)

[crtmqm \(tworzenie menedżera kolejek\)](#)

Przetwarzanie komunikatów w kolejce niedostarczonych komunikatów programu IBM MQ

Aby przetwarzać komunikaty w kolejce niedostarczonych komunikatów (DLQ), produkt IBM MQ dostarcza domyślną procedurę obsługi DLQ. Procedura obsługi dopasowuje komunikaty w DLQ do pozycji w tabeli reguł, która jest definiowana.

Komunikaty mogą być umieszczane w kolejce DLQ przez menedżery kolejek, agenty kanałów komunikatów (MCAs) i aplikacje. Wszystkie komunikaty w DLQ muszą być poprzedzane strukturą *dead-letter header*, MQDLH. Komunikaty umieszczone w kolejce DLQ przez menedżera kolejek lub agenta kanału komunikatów zawsze mają ten nagłówek; aplikacje umieszczające komunikaty w DLQ muszą dostarczyć ten nagłówek. Pole *Przyczyna* struktury MQDLH zawiera kod przyczyny, który określa, dlaczego komunikat znajduje się w kolejce DLQ.

Wszystkie środowiska produktu IBM MQ muszą regularnie przetwarzać komunikaty w DLQ. Produkt IBM MQ udostępnia procedurę domyślną o nazwie *procedura obsługi kolejki niedostarczonych komunikatów* (procedura obsługi kolejki DLQ), która jest wywoływana przy użyciu komendy `runmqdlq`.

Instrukcje dotyczące przetwarzania komunikatów na DLQ są dostarczane do procedury obsługi DLQ za pomocą napisanej przez użytkownika *tabeli reguł*. Oznacza to, że procedura obsługi DLQ jest zgodna z komunikatami w DLQ w odniesieniu do pozycji w tabeli reguł. Gdy komunikat DLQ jest zgodny z pozycją w tabeli reguł, procedura obsługi DLQ wykonuje działanie powiązane z tym wpisem.

Informacje pokrewne

[Kolejki niedostarczane](#)

[Rozwiązywanie problemów z niedostarczonym](#)

Procedura obsługi kolejki niedostarczonych komunikatów w systemie IBM i

Co to jest procedura obsługi kolejki niedostarczonych komunikatów programu IBM i i jak ją wywołać?

Kolejka niedostarczonych komunikatów (DLQ), nazywana czasem *kolejką niedostarczonych komunikatów*, jest kolejką wstrzymywania komunikatów, które nie mogą być dostarczane do kolejek docelowych. Każdy menedżer kolejek w sieci powinien mieć przypisany kod DLQ.

Uwaga: Często preferowane jest unikanie umieszczania komunikatów w kolejce DLQ. Informacje na temat używania i unikania plików DLQ znajdują się w sekcji [“Praca z kolejkami niedostarczonych komunikatów”](#) na stronie 156.

Menedżery kolejek, agenty kanałów komunikatów i aplikacje mogą umieszczać komunikaty w DLQ. Wszystkie komunikaty w DLQ muszą być poprzedzane strukturą *dead-letter header*, MQDLH. Komunikaty umieszczone w kolejce DLQ przez menedżera kolejek lub przez agenta kanału komunikatów zawsze mają wartość MQDLH. Zawsze dostarczaj MQDLH do aplikacji umieszczających komunikaty w DLQ. Pole *Przyczyna* struktury MQDLH zawiera kod przyczyny, który określa, dlaczego komunikat znajduje się w kolejce DLQ.

We wszystkich środowiskach IBM MQ musi istnieć procedura, która regularnie działa w celu przetwarzania komunikatów w DLQ. Produkt IBM MQ udostępnia procedurę domyślną o nazwie *procedura obsługi kolejki niedostarczonych komunikatów* (procedura obsługi kolejki DLQ), która jest wywoływana przy użyciu komendy `STRMQMDLQ`. Napisana przez użytkownika *tabela reguł* zawiera instrukcje do procedury obsługi DLQ, służące do przetwarzania komunikatów w kolejce DLQ. Oznacza to, że procedura obsługi DLQ jest zgodna z komunikatami w DLQ w odniesieniu do pozycji w tabeli reguł. Gdy komunikat DLQ jest zgodny z wpisem w tabeli reguł, procedura obsługi DLQ wykonuje działanie powiązane z tym wpisem.

Wywoływanie procedury obsługi DLQ

Użyj komendy `STRMQMDLQ`, aby wywołać procedurę obsługi DLQ. Można nazwać DLQ, który ma być przetwarzany, oraz menedżera kolejek, który ma być używany na dwa sposoby:

- Jako parametry komendy `STRMQMDLQ` z wiersza komend. Na przykład:

```
STRMQMDLQ UDLMSGQ(ABC1.DEAD.LETTER.QUEUE) SRCMBR(QRULE) SRCFILE(library/QXTSRC)
MQMNAME(MY.QUEUE.MANAGER)
```

- W tabeli reguł. Na przykład:

```
INPUTQ(ABC1.DEAD.LETTER.QUEUE)
```

Uwaga: Tabela reguł jest elementem w źródłowym zbiorze fizycznym, który może przyjmować dowolną nazwę.

Przykłady mają zastosowanie do DLQ o nazwie ABC1 . DEAD . LETTER . QUEUE, którego właścicielem jest domyślny menedżer kolejek.

Jeśli kolejka DLQ lub menedżer kolejek nie zostanie określona jako wyświetlona, zostanie użyty domyślny menedżer kolejek dla instalacji wraz z DLQ należącym do tego menedżera kolejek.

Komenda STRMQMDLQ pobiera dane wejściowe z tabeli reguł.

Aby można było uruchomić procedurę obsługi DLQ, użytkownik musi mieć uprawnienia dostępu do samego DLQ, a także do wszystkich kolejek komunikatów, do których są przekazywane komunikaty w DLQ. Użytkownik musi mieć również uprawnienia do przyjmowania tożsamości innych użytkowników, aby DLQ umieć umieszczać komunikaty w kolejkach z uprawnieniami identyfikatora użytkownika w kontekście komunikatu.

Informacje pokrewne

[Kolejki niedostarczane](#)

[Rozwiązywanie problemów z niedostarczonym](#)

Tabela reguł procedury obsługi DLQ w systemie IBM i

Tabela reguł programu obsługi kolejki niedostarczonych komunikatów definiuje, w jaki sposób procedura obsługi DLQ przetwarza komunikaty docierające do kolejki DLQ produktu IBM i .

Tabela reguł procedury obsługi DLQ definiuje, w jaki sposób procedura obsługi DLQ ma przetwarzać komunikaty, które docierają do DLQ. W tabeli reguł istnieją dwa typy pozycji:

- Pierwsza pozycja w tabeli, która jest opcjonalna, zawiera *dane sterujące*.
- Wszystkie pozostałe pozycje w tabeli to *reguły* dla procedury obsługi DLQ, która ma być śledzona. Każda reguła składa się z *wzorca* (zestawu charakterystyk komunikatu), z którym jest dopasowywany komunikat, oraz *działania* , które ma zostać podjęte, gdy komunikat w DLQ jest zgodny z określonym wzorcem. W tabeli reguł musi istnieć co najmniej jedna reguła.

Każda pozycja w tabeli reguł składa się z jednego lub większej liczby słów kluczowych.

Dane kontrolne

W tej sekcji opisano słowa kluczowe, które można uwzględnić w pozycji danych kontrolnych w tabeli reguł procedury obsługi DLQ. Na co zwrócić uwagę:

- Wartość domyślna dla słowa kluczowego, jeśli istnieje, jest podkreślona.
- Linia pionowa (|) oddziela alternatywy. Można określić tylko jedną z nich.
- Wszystkie słowa kluczowe są opcjonalne.

INPUTQ (QueueName| ' ')

Nazwa kolejki DLQ, która ma być przetwarzana:

1. Każda wartość UDLMMSGQ (lub *DFT) podana jako parametr komendy **STRMQMDLQ** przestania dowolną wartość INPUTQ w tabeli reguł.
2. Jeśli jako parametr komendy **STRMQMDLQ** zostanie podana pusta wartość UDLMMSGQ, zostanie użyta wartość INPUTQ w tabeli reguł.
3. Jeśli jako parametr komendy **STRMQMDLQ** zostanie podana pusta wartość UDLMMSGQ, a w tabeli reguł zostanie podana pusta wartość INPUTQ, zostanie użyta systemowa domyślna kolejka niedostarczonych komunikatów.

INPUTQM (QueueManagerNazwa| ' ')

Nazwa menedżera kolejek, do którego należy DLQ, o nazwie określonej w słowie kluczowym INPUTQ.

Jeśli menedżer kolejek nie zostanie określony lub zostanie określony parametr INPUTQM (") w tabeli reguł, system użyje domyślnego menedżera kolejek dla instalacji.

RETRYINT (Interwał|60)

Odstęp czasu (w sekundach), w którym procedura obsługi DLQ powinna podejmować próby ponownego przetwarzania komunikatów w DLQ, które nie mogły zostać przetworzone przy pierwszej próbie, i dla których zażądano powtórzonych prób. Domyślnie odstęp czasu między ponownymi próbami wynosi 60 sekund.

WAIT (TAK|NIE|nnn)

Określa, czy procedura obsługi DLQ powinna czekać na dotarcie kolejnych komunikatów do kolejki DLQ, gdy wykryje, że nie ma dalszych komunikatów, które mogą być przetwarzane.

YES

Powoduje, że procedura obsługi DLQ czeka bezterminowo.

NO

Powoduje zakończenie procedury obsługi DLQ, gdy wykryje, że kolejka DLQ jest pusta lub nie zawiera żadnych komunikatów, które może przetworzyć.

nnn

Powoduje, że procedura obsługi DLQ oczekuje na *nnn* sekund, aby nowa praca dotarła przed zakończeniem, po wykryciu, że kolejka jest pusta lub nie zawiera żadnych komunikatów, które może przetworzyć.

Określ wartość WAIT (YES) dla zajętych DLQ, a WAIT (NO) lub WAIT (*nnn*) dla DLQs, które mają niski poziom aktywności. Jeśli procedura obsługi DLQ może zakończyć działanie, ponownie wywołaj ją przy użyciu wyzwalania.

Nazwę DLQ można podać jako parametr wejściowy dla komendy **STRMQMDLQ**, jako alternatywę dla włączenia danych sterujących do tabeli reguł. Jeśli dowolna wartość jest określona zarówno w tabeli reguł, jak i w danych wejściowych komendy **STRMQMDLQ**, pierwszeństwo ma wartość podana w komendzie **STRMQMDLQ**.

Uwaga: Jeśli pozycja kontroli danych jest uwzględniona w tabeli reguł, musi to być pierwsza pozycja w tabeli.

IBM i Reguły DLQ (wzorce i działania) w systemie IBM i

Opis wzorców i działań dla każdej z reguł w kolejce niedostarczonych komunikatów produktu IBM i.

Poniżej znajduje się przykładowa reguła z tabeli reguł procedury obsługi DLQ:

```
PERSIST(MQPER_PERSISTENT) REASON (MQRC_PUT_INHIBITED) +  
ACTION (RETRY) RETRY (3)
```

Ta reguła nakazuje procedurze obsługi DLQ, aby 3 próby dostarczenia do kolejki docelowej były komunikaty trwałe, które zostały umieszczone w DLQ, ponieważ MQPUT i MQPUT1 zostały wstrzymane.

W tej sekcji opisano słowa kluczowe, które można uwzględnić w regule. Na co zwrócić uwagę:

- Wartość domyślna dla słowa kluczowego, jeśli istnieje, jest podkreślona. W przypadku większości słów kluczowych wartością domyślną jest * (gwiazdka), która jest zgodna z dowolną wartością.
- Linia pionowa (|) oddziela alternatywy. Można określić tylko jedną z nich.
- Wszystkie słowa kluczowe oprócz ACTION są opcjonalne.

Ta sekcja zaczyna się od opisu zgodnych ze wzorcem słów kluczowych (tych, dla których dopasowywane są komunikaty w DLQ). Następnie opisano słowa kluczowe działania (te, które określają, w jaki sposób procedura obsługi DLQ ma przetwarzać pasujący komunikat).

IBM i Słowa kluczowe dopasowywania wzorca DLQ w systemie IBM i

Słowa kluczowe dopasowywania wzorca są opisane w przykładzie. Te słowa kluczowe służą do określania wartości, dla których są dopasowywane komunikaty w kolejce niedostarczonych komunikatów serwera IBM i. Wszystkie słowa kluczowe dopasowywania wzorca są opcjonalne.

APPLIDAT (ApplIdentityDane|*)

Wartość *ApplIdentityData* komunikatu w DLQ, określona w deskrypcji komunikatu, MQMD.

APPLNAME (PutApplNazwa|*)

Nazwa aplikacji, która wywołała wywołanie MQPUT lub MQPUT1 , zgodnie z wartością określoną w polu *PutApplNazwa* deskryptora komunikatu, MQMD, komunikatu w DLQ.

APPLTYPE (PutApplTyp|*)

Wartość *PutApplType* określona w deskrypcji komunikatu, MQMD, komunikatu w DLQ.

DESTQ (QueueName|*)

Nazwa kolejki komunikatów, dla której jest przeznaczony komunikat.

DESTQM (QueueManagerNazwa|*)

Nazwa menedżera kolejek dla kolejki komunikatów, dla której jest przeznaczony komunikat.

FEEDBACK (Opinia|*)

Jeśli wartością parametru *MsgType* jest MQMT_REPORT, to *Opinia* opisuje charakter raportu.

Można używać nazw symbolicznych. Na przykład można użyć nazwy symbolicznej MQFB_COA, aby zidentyfikować te komunikaty w kolejce DLQ, które wymagają potwierdzenia ich przybycia do kolejek docelowych.

FORMAT (Format|*)

Nazwa, która jest używana przez nadawcę wiadomości do opisu formatu danych komunikatu.

MSGTYPE (MsgType|*)

Typ komunikatu w DLQ.

Można używać nazw symbolicznych. Na przykład można użyć nazwy symbolicznej MQMT_REQUEST, aby zidentyfikować te komunikaty w DLQ, które wymagają odpowiedzi.

PERSIST (Trwałość|*)

Wartość trwałości komunikatu. (Trwałość komunikatu określa, czy jest on zachowywanym restartami menedżera kolejek).

Można używać nazw symbolicznych. Na przykład można użyć nazwy symbolicznej MQPER_PERSISTENT w celu zidentyfikowania tych komunikatów w pliku DLQ, które są trwałe.

PRZYCZYNA (ReasonCode|*)

Kod przyczyny opisujący, dlaczego komunikat został umieszczony w kolejce DLQ.

Można używać nazw symbolicznych. Na przykład można użyć nazwy symbolicznej MQRC_Q_FULL, aby zidentyfikować te komunikaty umieszczone w kolejce DLQ, ponieważ ich kolejki docelowe były pełne.

REPLYQ (QueueName|*)

Nazwa kolejki odpowiedzi podana w deskrypcji komunikatu, MQMD, komunikatu w DLQ.

REPLYQM (QueueManagerNazwa|*)

Nazwa menedżera kolejek dla kolejki odpowiedzi określonej w słowie kluczowym REPLYQ.

USERID (UserIdentifier|*)

Identyfikator użytkownika, który zainicjował komunikat w DLQ, zgodnie z określonym w deskrypcji komunikatu, MQMD.

IBM i Słowa kluczowe działania DLQ w systemie IBM i

Użyj tych słów kluczowych działań w kolejce niedostarczonych komunikatów, aby określić, w jaki sposób zostanie przetworzony zgodny komunikat w kolejce niedostarczonych komunikatów programu IBM i .

ACTION (DISCARD | IGNORE | RETRY | FWD)

Działanie podjęte dla dowolnego komunikatu w DLQ, które jest zgodne ze wzorcem zdefiniowanym w tej regule.

ODRZUĆ

Powoduje, że komunikat zostanie usunięty z kolejki DLQ.

IGNORE

Powoduje, że komunikat ma być przechowywany w DLQ.

Ponów

Powoduje, że procedura obsługi DLQ próbuje ponownie umieścić komunikat w jego kolejce docelowej.

FWD

Powoduje, że procedura obsługi DLQ przekazuje komunikat do kolejki o nazwie określonej w słowie kluczowym FWDQ.

Należy określić słowo kluczowe ACTION. Liczba podjętych prób wykonania działania jest określana przez słowo kluczowe RETRY. Słowo kluczowe RETRYINT danych sterujących określa odstęp czasu między kolejnymi próbami.

FWDQ (*QueueName* | & DESTQ | & REPLYQ)

Nazwa kolejki komunikatów, do której przekazywany jest komunikat po wybraniu słowa kluczowego ACTION.

QueueName

Nazwa kolejki komunikatów. Wartość FWDQ (") jest niepoprawna.

& DESTQ

Wprowadź nazwę kolejki z pola *DestQName* w strukturze MQDLH.

& REPLYQ

Wprowadź nazwę kolejki z pola *ReplyToQ* w deskrytorze komunikatu, MQMD.

Można podać wartość REPLYQ (? *) we wzorcu komunikatów, aby uniknąć komunikatów o błędach, gdy reguła określająca wartość FWDQ (& REPLYQ) jest zgodna z komunikatem o pustym polu *ReplyToQ* .

FWDQM (*QueueManagerNazwa* | & DESTQM | & REPLYQM | ' ')

Menedżer kolejek kolejki, do której przekazywany jest komunikat.

QueueManagerName

Nazwa menedżera kolejek dla kolejki, do której przekazywany jest komunikat po wybraniu słowa kluczowego ACTION (FWD).

& DESTQM

Należy podać nazwę menedżera kolejek z pola *DestQMGrNazwa* w strukturze MQDLH.

& REPLYQM

Wprowadź nazwę menedżera kolejek z pola *ReplyToQMGr* w deskrytorze komunikatu, MQMD.

..

FWDQM (") , która jest wartością domyślną, identyfikuje lokalny menedżer kolejek.

HEADER (TAK|NIE)

Określa, czy kod MQDLH powinien pozostać w komunikacie, dla którego żądana jest ACTION (FWD). Domyślnie MQDLH pozostaje w komunikacie. Słowo kluczowe HEADER nie jest poprawne dla działań innych niż FWD.

PUTAUT (DEF| CTX)

Uprawnienie, z którym komunikaty powinny być umieszczane przez procedurę obsługi DLQ:

DEF

Umieszcza komunikaty z uprawnieniami samej procedury obsługi DLQ.

CTX

Powoduje, że komunikaty mają być umieszczane z uprawnieniami identyfikatora użytkownika w kontekście komunikatu. Jeśli zostanie określona wartość PUTAUT (CTX), użytkownik musi mieć uprawnienia do przyjmowania tożsamości innych użytkowników.

PONÓW (*RetryCount*|1)

Liczba prób wykonania działania z zakresu od 1 do 999.999,999 (w odstępie czasu określonym w słowie kluczowym RETRYINT danych sterujących).

Uwaga: Liczba prób podejmowanych przez procedurę obsługi DLQ w celu zaimplementowania konkretnej reguły jest specyficzna dla bieżącej instancji procedury obsługi DLQ. Liczba ta nie jest utrwalana po restarcie. Jeśli procedura obsługi DLQ zostanie zrestartowany, liczba podjętych prób zastosowania reguły zostanie zresetowana do zera.

IBM i Konwencje tabeli reguł DLQ w systemie IBM i

Tabela reguł w kolejce niedostarczonych komunikatów IBM i musi stosować się do konkretnych konwencji dotyczących jej składni, struktury i treści.

- Tabela reguł musi zawierać co najmniej jedną regułę.
- Słowa kluczowe mogą wystąpić w dowolnej kolejności.
- Słowo kluczowe może zostać włączone tylko raz w dowolnej regule.
- W słowach kluczowych rozróżniana jest wielkość liter.
- Słowo kluczowe i jego wartość parametru muszą być oddzielone od innych słów kluczowych za pomocą co najmniej jednego znaku odstępu lub przecinka.
- Na początku lub na końcu reguły może wystąpić dowolna liczba odstępów, a także między słowami kluczowymi, interpunkcją i wartościami.
- Każda reguła musi zaczynać się od nowej linii.
- W przypadku przenośności długość wiersza nie może być większa niż 72 znaki.
- Znak plus (+) jest używany jako ostatni niepusty znak w wierszu, aby wskazać, że reguła jest kontynuowana od pierwszego niepułstego znaku w następnym wierszu. Znak minus (-) oznacza ostatni niepusty znak w wierszu, który wskazuje, że reguła jest kontynuowana od początku następnego wiersza. Znaki kontynuacji mogą występować w słowach kluczowych i parametrach.

Na przykład:

```
APPLNAME (' ABC+  
D')
```

wyniki w "ABCD".

```
APPLNAME (' ABC -  
D')
```

wyniki w " ABC D".

- Wiersze komentarza rozpoczynające się gwiazdką (*) mogą występować w dowolnym miejscu w tabeli reguł.
- Puste wiersze są ignorowane.
- Każda pozycja w tabeli reguł procedury obsługi DLQ składa się z jednego lub większej liczby słów kluczowych i powiązanych z nimi parametrów. Parametry muszą być zgodne z następującymi regułami składni:
 - Każda wartość parametru musi zawierać co najmniej jeden znaczący znak. Znaki cudzysłowu ograniczające wartości ujęte w znaki cudzysłowu nie są uznawane za znaczące. Na przykład te parametry są poprawne:

FORMAT (' ABC ')	3 znaczące znaki
FORMAT (ABC)	3 znaczące znaki
FORMAT (' A ')	1 znak znaczący
FORMAT (A)	1 znak znaczący
FORMAT (' ')	1 znak znaczący

Te parametry są niepoprawne, ponieważ nie zawierają znaczących znaków:

FORMAT(' ')

FORMAT()

FORMAT()

FORMAT

- Obsługiwane są znaki wieloznaczne. Znak zapytania (?) można użyć w miejsce dowolnego pojedynczego znaku, z wyjątkiem odstępów kończącego. Można użyć znaku gwiazdki (*) w miejscu zero lub więcej sąsiednich znaków. Gwiazdka (*) i znak zapytania (?) są **zawsze** interpretowani jako znaki wieloznaczne w wartościach parametrów.
- W parametrach tych słów kluczowych nie można używać znaków wieloznacznych: ACTION, HEADER, RETRY, FWDQ, FWDQM i PUTAUT.
- Odstępy końcowe w wartościach parametrów i odpowiadające im pola w komunikacie w DLQ nie są znaczące podczas wykonywania znaków wieloznacznych. Jednak początkowe i osadzone odstępy w łańcuchach w cudzysłowach są znaczące w przypadku zgodności ze znakami wieloznacznymi.
- Parametry liczbowe nie mogą zawierać znaku wieloznacznego znaku zapytania (?). Można dołączyć gwiazdkę (*) w miejsce całego parametru liczbowego, ale gwiazdka nie może być dołączona jako część parametru liczbowego. Na przykład są to poprawne parametry liczbowe:

MSGTYPE(2) Zakwalifikowane są tylko komunikaty odpowiedzi

MSGTYPE(*) Dowolny typ komunikatu jest zakwalifikowany

MSGTYPE(' * ') Dowolny typ komunikatu jest zakwalifikowany

Jednak wartość MSGTYPE(' 2* ') nie jest poprawna, ponieważ zawiera gwiazdkę (*) jako część parametru liczbowego.

- Parametry liczbowe muszą mieścić się w zakresie od 0 do 999 999 999. Jeśli wartość parametru znajduje się w tym zakresie, jest ona akceptowana, nawet jeśli nie jest ona obecnie poprawna w polu, do którego odnosi się słowo kluczowe. Dla parametrów liczbowych można użyć nazw symbolicznych.
- Jeśli wartość łańcuchowa jest krótsza niż wartość pola w MQDLH lub MQMD, do którego odnosi się słowo kluczowe, to wartość jest dopełniona spacjami do długości pola. Jeśli wartość, z wyjątkiem gwiazdek, jest dłuższa niż pole, zostanie zdiagnozowany błąd. Na przykład są to wszystkie poprawne wartości łańcuchowe dla 8-znakowego pola:

'ABCDEFGH' 8 znaków

'A*C*E*G*I' 5 znaków z wyłączeniem gwiazdek

'*A*C*E*G*I*K*M*O*' 8 znaków z wyłączeniem gwiazdek

- Łańcuchy zawierające spacje, małe litery lub znaki specjalne inne niż kropka (.), ukośnik (?), znak podkreślenia (_) i znak procentu (%) muszą być ujęte w pojedynczy cudzysłów. Małe litery, które nie są ujęte w znaki cudzysłowu, są składane wielkimi literami. Jeśli łańcuch zawiera znak cudzysłowu, należy użyć dwóch cudzysłowów, aby określić zarówno początek, jak i koniec cytowania. Jeśli długość łańcucha jest obliczana, każde wystąpienie podwójnych cudzysłowów jest traktowane jako pojedynczy znak.

IBM i Przetwarzanie tabeli reguł DLQ w systemie IBM i

Procedura obsługi kolejki niedostarczonych komunikatów przeszukuje tabelę reguł dla reguły z wzorcem, który jest zgodny z komunikatem w kolejce niedostarczonych komunikatów serwera IBM i .

Wyszukiwanie rozpoczyna się od pierwszej reguły w tabeli i jest kontynuowane sekwencyjnie przez tabelę. Jeśli zostanie znaleziona reguła z dopasowaniem wzorca, tabela reguł podejmie próbę wykonania działania z tej reguły. Procedura obsługi DLQ zwiększa liczbę ponowień dla reguły o 1, gdy próbuje zastosować tę regułę. Jeśli pierwsza próba nie powiedzie się, próba zostanie powtórzona do momentu,

gdy liczba wykonanych prób zostanie dopasowana do liczby określonej w słowie kluczowym RETRY. Jeśli wszystkie próby nie powiodą się, procedura obsługi DLQ wyszuka następną zgodną regułę w tabeli.

Ten proces jest powtarzany dla kolejnych dopasowanych reguł, dopóki działanie nie powiedzie się. Jeśli dla każdej reguły dopasowania została podjęta próba, ile razy określono w słowie kluczowym RETRY, a wszystkie próby nie powiodły się, przyjmowana jest wartość ACTION (IGNORE). Przyjęto również działanie ACTION (IGNORE), jeśli nie zostanie znaleziona żadna zgodna reguła.

Uwaga:

1. Wzorce reguł dopasowywania są wyszukiwane tylko dla komunikatów w DLQ, które rozpoczynają się od MQDLH. Komunikaty, które nie rozpoczynają się od wywołania MQDLH, są okresowo zgłaszane jako błędne i pozostają w kolejce DLQ w nieskończoność.
2. Wszystkie słowa kluczowe wzorca mogą być domyślne, tak aby reguła mogła się składać tylko z działania. Należy jednak pamiętać o tym, że reguły tylko do działania są stosowane do wszystkich komunikatów w kolejce, które mają MQDLHs i które nie zostały jeszcze przetworzone zgodnie z innymi regułami w tabeli.
3. Poprawność tabeli reguł jest sprawdzana podczas uruchamiania procedury obsługi DLQ, a błędy w tym czasie oznaczone są flagą. Komunikaty o błędach wystawione przez procedurę obsługi DLQ są opisane w sekcji *Komunikaty i kody przyczyny*. W dowolnym momencie można wprowadzić zmiany w tabeli reguł, ale te zmiany nie wchodzi w życie, dopóki procedura obsługi DLQ nie zostanie zrestartowana.
4. Procedura obsługi DLQ nie zmienia treści komunikatów, MQDLH ani deskryptora komunikatu. Procedura obsługi DLQ zawsze umieszcza komunikaty w innych kolejkach za pomocą opcji komunikatu MQPMO_PASS_ALL_CONTEXT.
5. Kolejne błędy składniowe w tabeli reguł mogą nie zostać rozpoznane, ponieważ sprawdzanie poprawności tabeli reguł eliminuje generowanie powtarzających się błędów.
6. Procedura obsługi DLQ otwiera DLQ z opcją MQOO_INPUT_AS_Q_DEF.
7. Wiele instancji procedury obsługi DLQ może działać współbieżnie z tą samą kolejką, korzystając z tej samej tabeli reguł. Jednak bardziej zwykle jest to relacja jeden do jednego między DLQ a procedurą obsługi DLQ.

IBM i

Sprawdzanie, czy wszystkie komunikaty DLQ są przetwarzane w systemie IBM i

Procedura obsługi kolejki niedostarczonych komunikatów przechowuje zapis wszystkich komunikatów w kolejce DLQ produktu IBM i , które zostały wyświetlone, ale nie zostały usunięte.

Jeśli procedura obsługi DLQ jest używana jako filtr w celu wyodrębnienia niewielkiego podzbioru komunikatów z kolejki DLQ, procedura obsługi DLQ nadal zapisuje te komunikaty w DLQ, które nie zostały przetworzone. Ponadto procedura obsługi DLQ nie może zagwarantować, że nowe komunikaty przychodzące do kolejki DLQ będą widoczne, nawet jeśli DLQ jest zdefiniowane jako pierwszy przyszedł-pierwszy wyszedł (FIFO). Jeśli kolejka nie jest pusta, kolejka DLQ jest okresowo ponownie skanowana w celu sprawdzenia wszystkich komunikatów.

Z tych powodów należy spróbować upewnić się, że DLQ zawiera możliwie jak najmniejszej liczby komunikatów. Jeśli komunikaty, które nie mogą zostać usunięte lub przekazane do innych kolejek (z jakiegokolwiek powodu), mogą gromadzić się w kolejce, obciążenie procedury obsługi DLQ zwiększa się, a samo DLQ jest w niebezpieczeństwie zapelnienia.

Istnieje możliwość podjęcia określonych działań, aby umożliwić obsługę opróżnienia DLQ przez procedurę obsługi DLQ. Na przykład, spróbuj nie używać ACTION (IGNORE), który pozostawia komunikaty w DLQ. (Pamiętaj, że dla komunikatów, które nie są jawnie adresowane przez inne reguły w tabeli, przyjmowana jest wartość ACTION (IGNORE)). Zamiast tego w przypadku komunikatów, które w przeciwnym razie należy zignorować, należy użyć działania, które przenosi komunikaty do innej kolejki. Na przykład:

```
ACTION (FWD) FWDQ (IGNORED.DEAD.QUEUE) HEADER (YES)
```

Analogicznie należy wprowadzić regułę końcową w tabeli a catchall w celu przetwarzania komunikatów, które nie zostały uwzględnione we wcześniejszych regułach w tabeli. Na przykład ostateczna reguła w tabeli może być taka, jak ta:

```
ACTION (FWD) FWDQ (REALLY.DEAD.QUEUE) HEADER (YES)
```

Powoduje to, że komunikaty, które są przekazywane do ostatecznej reguły w tabeli, są przekazywane do kolejki REALLYY . DEAD . QUEUE, gdzie mogą być przetwarzane ręcznie. Jeśli użytkownik nie ma takiej reguły, komunikaty prawdopodobnie pozostaną w kolejce DLQ w nieskończoność.

IBM i Przykładowa tabela reguł procedury obsługi DLQ w systemie IBM i

Przykładowy kod dla tabeli reguł procedury obsługi kolejki niedostarczonych komunikatów w systemie IBM i. W tej przykładowej tabeli reguł znajduje się pojedynczy element sterujący-wprowadzanie danych i kilka reguł.

```
*****
*   An example rules table for the STRMQMDLQ command           *
*****
* Control data entry
* -----
* If no queue manager name is supplied as an explicit parameter to
* STRMQMDLQ, use the default queue manager for the machine.
* If no queue name is supplied as an explicit parameter to STRMQMDLQ,
* 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 is 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 does 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 must 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 do not 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)
```

Wywoływanie procedury obsługi DLQ

Wywołaj procedurę obsługi kolejki niedostarczonych komunikatów przy użyciu komendy `runmqdlq`. Można nazwać DLQ, który ma być przetwarzany, oraz menedżera kolejek, który ma być używany na dwa sposoby.

Są to następujące dwa sposoby:

- Jako parametry dla komendy `runmqdlq` z wiersza komend. Na przykład:

```
runmqdlq ABC1.DEAD.LETTER.QUEUE ABC1.QUEUE.MANAGER <qrule.rul
```

- W tabeli reguł. Na przykład:

```
INPUTQ(ABC1.DEAD.LETTER.QUEUE) INPUTQM(ABC1.QUEUE.MANAGER)
```

Przykłady mają zastosowanie do DLQ o nazwie `ABC1.DEAD.LETTER.QUEUE`, której właścicielem jest menedżer kolejek `ABC1.QUEUE.MANAGER`.

Jeśli kolejka DLQ lub menedżer kolejek nie zostanie określona jako wyświetlona, zostanie użyty domyślny menedżer kolejek dla instalacji wraz z DLQ należącym do tego menedżera kolejek.

Komenda `runmqdlq` pobiera dane wejściowe z programu `stdin`. Tabela reguł zostanie powiązana z komendą `runmqdlq`, przekierując `stdin` z tabeli reguł.

Aby uruchomić procedurę obsługi DLQ, użytkownik musi mieć uprawnienia dostępu zarówno do samego DLQ, jak i do wszystkich kolejek komunikatów, do których przekazywane są komunikaty w DLQ. Aby procedura obsługi DLQ umiała umieszczać komunikaty w kolejkach z uprawnieniami identyfikatora użytkownika w kontekście komunikatu, należy mieć również uprawnienia do przyjmowania tożsamości innych użytkowników.

Więcej informacji na temat komendy `runmqdlq` zawiera sekcja [runmqdlq](#).

Informacje pokrewne

[Kolejki niedostarczane](#)

[Rozwiązywanie problemów z niedostarczonym](#)

Przykładowa procedura obsługi DLQ `amqsd1q`

Oprócz procedury obsługi kolejki niedostarczonych komunikatów wywołanej za pomocą komendy **`runmqdlq`**, program IBM MQ udostępnia źródło przykładowej procedury obsługi DLQ **`amqsd1q`** z funkcją podobną do funkcji dostarczonej przez produkt **`runmqdlq`**.

Użytkownik może dostosować program `amqsd1q` w celu udostępnienia procedury obsługi DLQ, która spełnia wymagania użytkownika. Na przykład użytkownik może zdecydować, że procedura obsługi DLQ

może przetwarzać komunikaty bez nagłówek niewysłanych wiadomości. (Zarówno domyślna procedura obsługi DLQ, jak i przykład, amqsd1q, przetwarzają tylko komunikaty w DLQ, które rozpoczynają się od nagłówka niedostarczonych komunikatów, MQDLH. Komunikaty, które nie rozpoczynają się od wywołania MQDLH, są identyfikowane jako błędne i pozostają w kolejce DLQ w nieskończoność.)

`MQ_INSTALLATION_PATH` reprezentuje katalog najwyższego poziomu, w którym zainstalowany jest produkt IBM MQ.

W katalogu IBM MQ for Windows źródło `amqsd1q` jest dostarczane w katalogu:

```
MQ_INSTALLATION_PATH\tools\c\samples\d1q
```

a skompilowana wersja jest dostarczana w katalogu:

```
MQ_INSTALLATION_PATH\tools\c\samples\bin
```

W systemach IBM MQ for UNIX i Linux źródło `amqsd1q` jest dostarczane w katalogu:

```
MQ_INSTALLATION_PATH/samp/d1q
```

a skompilowana wersja jest dostarczana w katalogu:

```
MQ_INSTALLATION_PATH/samp/bin
```

Można również skompilować produkt **amqsd1q** w trybie klienta. Więcej informacji na ten temat zawiera sekcja [Zapisywanie aplikacji proceduralnych klienta, Budowanie aplikacji dla produktu IBM MQ MQI clientsi Uruchamianie aplikacji w środowisku produktu IBM MQ MQI client.](#)

Tabela reguł procedury obsługi DLQ

Tabela reguł programu obsługi kolejki niedostarczonych komunikatów definiuje, w jaki sposób procedura obsługi DLQ przetwarza komunikaty docierające do kolejki DLQ.

W tabeli reguł istnieją dwa typy pozycji:

- Pierwsza pozycja w tabeli, która jest opcjonalna, zawiera *dane sterujące*.
- Wszystkie pozostałe pozycje w tabeli to *reguły* dla procedury obsługi DLQ, która ma być śledzona. Każda reguła składa się z *wzorca* (zestawu charakterystyk komunikatu), z którym jest dopasowywany komunikat, oraz *działania*, które ma zostać podjęte, gdy komunikat w DLQ jest zgodny z określonym wzorcem. W tabeli reguł musi istnieć co najmniej jedna reguła.

Każda pozycja w tabeli reguł składa się z jednego lub większej liczby słów kluczowych.

Informacje pokrewne

[Kolejki niedostarczane](#)

[Rozwiązywanie problemów z niedostarczonym](#)

Dane sterujące DLQ

Do pozycji danych sterujących można dołączyć słowa kluczowe w tabeli reguł procedur obsługi kolejki niedostarczonych komunikatów.

Uwaga:

- Linia pionowa (|) oddziela alternatywy, tylko jedna z nich może być określona.
- Wszystkie słowa kluczowe są opcjonalne.

INPUTQ (*QueueName* | ' ')

Nazwa kolejki DLQ, która ma być przetwarzana:

1. Każda wartość `INPUTQ` podana jako parametr komendy `runmqd1q` nadpisuje dowolną wartość `INPUTQ` w tabeli reguł.
2. Jeśli wartość `INPUTQ` nie zostanie określona jako parametr dla komendy `runmqd1q`, ale zostanie podana wartość **do** w tabeli reguł, zostanie użyta wartość `INPUTQ` w tabeli reguł.

3. Jeśli nie podano parametru DLQ lub określono wartość INPUTQ (") w tabeli reguł, używana jest nazwa DLQ należącego do menedżera kolejek o nazwie podanej jako parametr komendy `runmqdlq`.
4. Jeśli wartość INPUTQ nie zostanie określona jako parametr dla komendy `runmqdlq` lub jako wartość w tabeli reguł, zostanie użyta wartość DLQ należącej do menedżera kolejek o nazwie określonej w słowie kluczowym INPUTQM w tabeli reguł.

INPUTQM (*QueueManagerNazwa* | ')

Nazwa menedżera kolejek, do którego należy DLQ, o nazwie określonej w słowie kluczowym INPUTQ:

1. Każda wartość INPUTQM podana jako parametr komendy `runmqdlq` przestania dowolną wartość INPUTQM w tabeli reguł.
2. Jeśli wartość INPUTQM nie zostanie określona jako parametr dla komendy `runmqdlq`, zostanie użyta wartość INPUTQM w tabeli reguł.
3. Jeśli nie określono menedżera kolejek lub określono INPUTQM (") w tabeli reguł, używany jest domyślny menedżer kolejek dla instalacji.

RETRYINT (*Interwał*|60)

Odstęp czasu (w sekundach), w którym procedura obsługi DLQ powinna ponownie przetwarzać komunikaty w DLQ, które nie mogły zostać przetworzone przy pierwszej próbie, i dla których zażądano powtórzonych prób. Domyślnie odstęp czasu między ponownymi próbami wynosi 60 sekund.

WAIT (*TAK|NIE*|*nnn*)

Określa, czy procedura obsługi DLQ powinna czekać na dotarcie kolejnych komunikatów do kolejki DLQ, gdy wykryje, że nie ma dalszych komunikatów, które mogą być przetwarzane.

YES

Procedura obsługi DLQ czeka bezterminowo.

NO

Procedura obsługi DLQ kończy się, gdy wykryje, że kolejka DLQ jest pusta lub nie zawiera żadnych komunikatów, które może przetworzyć.

nnn

Procedura obsługi DLQ oczekuje na *nnn* sekund, aby nowa praca dotarła przed zakończeniem, po wykryciu, że kolejka jest pusta lub nie zawiera żadnych komunikatów, które może przetworzyć.

Podaj wartość WAIT (YES) dla zajętych DLQ i WAIT (NO) lub WAIT (*nnn*) dla DLQs, które mają niski poziom aktywności. Jeśli procedura obsługi DLQ może zakończyć działanie, wywołaj ją ponownie, wywołując wyzwolenie. Więcej informacji na temat wyzwalań zawiera sekcja [Uruchamianie aplikacji produktu IBM MQ za pomocą wyzwalaczy](#).

Alternatywą dla dołączania danych sterujących w tabeli reguł jest podanie nazw DLQ i jego menedżera kolejek jako parametrów wejściowych dla komendy `runmqdlq`. Jeśli zostanie podana wartość zarówno w tabeli reguł, jak i jako dane wejściowe dla komendy `runmqdlq`, pierwszeństwo ma wartość podana w komendzie `runmqdlq`.

Jeśli w tabeli reguł zostanie dołączona pozycja danych kontrolnych, musi to być pozycja **pierwsza** w tabeli.

Reguły DLQ (wzorce i działania)

Opis zgodnych ze wzorcem słów kluczowych (tych, dla których dopasowywane są komunikaty w kolejce niedostarczonych komunikatów) oraz słów kluczowych działań (tych, które określają, w jaki sposób procedura obsługi DLQ ma przetwarzać pasujący komunikat). Udostępniana jest również przykładowa reguła.

Słowa kluczowe dopasowywania wzorca

Słowa kluczowe dopasowywania wzorca, które są używane do określania wartości, dla których dopasowywane są komunikaty w DLQ, są następujące. (Wszystkie słowa kluczowe dopasowywania wzorca są opcjonalne):

APPLIDAT (*ApplIdentityData*|*)

Wartość *ApplIdentityData* podana w deskrytorze komunikatu, MQMD, komunikatu w DLQ.

APPLNAME (*PutApplNazwa*|*)

Nazwa aplikacji, która wywołała wywołanie MQPUT lub MQPUT1 , zgodnie z wartością podaną w polu *PutApplName* deskryptora komunikatu MQMD komunikatu w DLQ.

APPLTYPE (*PutApplTyp*|*)

Wartość *PutApplType* określona w deskrytorze komunikatu MQMD komunikatu w DLQ.

DESTQ (*QueueName*|*)

Nazwa kolejki komunikatów, dla której jest przeznaczony komunikat.

DESTQM (*QueueManagerNazwa*|*)

Nazwa menedżera kolejek dla kolejki komunikatów, dla której jest przeznaczony komunikat.

FEEDBACK (*Opinia*|*)

Jeśli wartością parametru *MsgType* jest MQFB_REPORT, to *Opinia* opisuje charakter raportu.

Można używać nazw symbolicznych. Na przykład można użyć nazwy symbolicznej MQFB_COA, aby zidentyfikować te komunikaty w kolejce DLQ, które wymagają potwierdzenia ich przybycia do kolejek docelowych.

FORMAT (*Format*|*)

Nazwa, która jest używana przez nadawcę wiadomości do opisu formatu danych komunikatu.

MSGTYPE (*MsgType*|*)

Typ komunikatu w DLQ.

Można używać nazw symbolicznych. Na przykład można użyć nazwy symbolicznej MQMT_REQUEST, aby zidentyfikować te komunikaty w kolejce DLQ, które wymagają odpowiedzi.

PERSIST (*Trwałość*|*)

Wartość trwałości komunikatu. (Trwałość komunikatu określa, czy jest on zachowywanym restartami menedżera kolejek).

Można używać nazw symbolicznych. Na przykład można użyć nazwy symbolicznej MQPER_PERSISTENT w celu zidentyfikowania komunikatów w DLQ, które są trwałe.

PRZYCZYNA (*ReasonCode*|*)

Kod przyczyny opisujący, dlaczego komunikat został umieszczony w kolejce DLQ.

Można używać nazw symbolicznych. Na przykład można użyć nazwy symbolicznej MQRC_Q_FULL, aby zidentyfikować te komunikaty umieszczone w kolejce DLQ, ponieważ ich kolejki docelowe były pełne.

REPLYQ (*QueueName*|*)

Nazwa kolejki odpowiedzi określonej w deskrytorze komunikatu, MQMD, komunikatu w DLQ.

REPLYQM (*QueueManagerNazwa*|*)

Nazwa menedżera kolejek dla kolejki odpowiedzi, zgodnie z określonym w deskrytorze komunikatu, MQMD komunikatu w DLQ.

USERID (*UserIdentifier*|*)

Identyfikator użytkownika, który zainicjował komunikat w DLQ, określony w deskrytorze komunikatu, MQMD, komunikatu w kolejce DLQ.

Słowa kluczowe akcji

Słowa kluczowe działania, używane do opisywania sposobu przetwarzania komunikatu zgodnego z pasującym komunikatem, są następujące:

ACTION (DISCARD | IGNORE | RETRY | FWD)

Działanie, które ma zostać podjęte dla dowolnego komunikatu w DLQ, który jest zgodny ze wzorcem zdefiniowanym w tej regule.

ODRZUĆ

Usuń komunikat z kolejki DLQ.

IGNORE

Pozostaw komunikat w DLQ.

Ponów

Jeśli pierwsza próba umieszczenia komunikatu w kolejce docelowej nie powiedzie się, spróbuj ponownie. Słowo kluczowe RETRY ustawia liczbę prób podejmowanych w celu zaimplementowania działania. Słowo kluczowe RETRYINT danych sterujących określa odstęp czasu między kolejnymi próbami.

FWD

Przekaz komunikat do kolejki o nazwie określonej w słowie kluczowym FWDQ.

Należy określić słowo kluczowe ACTION.

FWDQ (*QueueName* | & DESTQ | & REPLYQ)

Nazwa kolejki komunikatów, do której ma zostać wyświetlony komunikat z żądaniem działania FWD (ACTION).

QueueName

Nazwa kolejki komunikatów. Wartość FWDQ (") jest niepoprawna.

& DESTQ

Wprowadź nazwę kolejki z pola *DestQName* w strukturze MQDLH.

& REPLYQ

Wprowadź nazwę kolejki z pola *ReplyToQ* w deskrytorze komunikatu, MQMD.

Aby uniknąć komunikatów o błędach, gdy reguła określająca wartość FWDQ (& REPLYQ) jest zgodna z komunikatem o pustym polu *ReplyToQ*, należy podać wartość REPLYQ (? *) w wzorzec komunikatu.

FWDQM (*QueueManagerNazwa* | & DESTQM | & REPLYQM | ' ')

Menedżer kolejek w kolejce, do której ma zostać wysłany komunikat.

QueueManagerName

Nazwa menedżera kolejek kolejki, do której ma zostać wysłany komunikat, gdy zażądano działania FWD (ACTION).

& DESTQM

Należy podać nazwę menedżera kolejek z pola *DestQMGrNazwa* w strukturze MQDLH.

& REPLYQM

Wprowadź nazwę menedżera kolejek z pola *ReplyToQMGr* w deskrytorze komunikatu, MQMD.

..

FWDQM ("), która jest wartością domyślną, identyfikuje lokalny menedżer kolejek.

HEADER (TAK|NIE)

Określa, czy kod MQDLH powinien pozostać w komunikacie, dla którego żądana jest ACTION (FWD). Domyślnie MQDLH pozostaje w komunikacie. Słowo kluczowe HEADER nie jest poprawne dla działań innych niż FWD.

PUTAUT (DEF| CTX)

Uprawnienie, z którym komunikaty powinny być umieszczane przez procedurę obsługi DLQ:

DEF

Służy do umieszczania komunikatów z uprawnieniami samej procedury obsługi DLQ.

CTX

Umieść komunikaty przy użyciu uprawnień identyfikatora użytkownika w kontekście komunikatu. Jeśli zostanie określona wartość PUTAUT (CTX), użytkownik musi mieć uprawnienia do przyjmowania tożsamości innych użytkowników.

PONÓW (*RetryCount*|1)

Liczba prób wykonania działania z zakresu od 1 do 999.999,999 (w odstępie czasu określonym w słowie kluczowym RETRYINT danych sterujących). Liczba prób podejmowanych przez procedurę obsługi DLQ w celu zaimplementowania konkretnej reguły jest specyficzna dla bieżącej instancji

procedury obsługi DLQ. Liczba ta nie jest utrwalana po restarcie. Jeśli procedura obsługi DLQ zostanie zrestartowana, liczba prób zastosowania reguły jest resetowana do zera.

Przykładowa reguła

Poniżej znajduje się przykładowa reguła z tabeli reguł procedury obsługi DLQ:

```
PERSIST(MQPER_PERSISTENT) REASON (MQRC_PUT_INHIBITED) +  
ACTION (RETRY) RETRY (3)
```

Ta reguła nakazuje procedurze obsługi DLQ dokonanie trzech prób dostarczenia do kolejki docelowej wszystkich komunikatów trwałych, które zostały umieszczone w kolejce DLQ, ponieważ komenda MQPUT i MQPUT1 zostały zablokowane.

Wszystkie słowa kluczowe, których można użyć w regule, są opisane w dalszej części tej sekcji. Na co zwrócić uwagę:

- Wartość domyślna dla słowa kluczowego, jeśli istnieje, jest podkreślona. W przypadku większości słów kluczowych wartością domyślną jest * (gwiazdka), która jest zgodna z dowolną wartością.
- Linia pionowa (|) oddziela alternatywy, tylko jedna z nich może być określona.
- Wszystkie słowa kluczowe oprócz ACTION są opcjonalne.

Konwencje tabeli reguł DLQ

Składnia, struktura i treść tabeli reguł procedur obsługi kolejki niedostarczonych komunikatów muszą być zgodne z tymi konwencjami.

Tabela reguł musi być zgodna z następującymi konwencjami:

- Tabela reguł musi zawierać co najmniej jedną regułę.
- Słowa kluczowe mogą wystąpić w dowolnej kolejności.
- Słowo kluczowe może zostać włączone tylko raz w dowolnej regule.
- W słowach kluczowych nie jest rozróżniana wielkość liter.
- Słowo kluczowe i jego wartość parametru muszą być oddzielone od innych słów kluczowych za pomocą co najmniej jednego znaku odstępu lub przecinka.
- Na początku lub na końcu reguły może znajdować się dowolna liczba odstępów oraz między słowami kluczowymi, interpunkcją i wartościami.
- Każda reguła musi zaczynać się od nowej linii.
- W systemach Windows ostatnia reguła w tabeli musi kończyć się znakiem powrotu karetki/znaku nowego wiersza. Aby to osiągnąć, należy nacisnąć klawisz Enter na końcu reguły, tak aby ostatni wiersz tabeli był pusty.
- Ze względu na przenośność, długość linii nie może być większa niż 72 znaki.
- Znak plus (+) jest używany jako ostatni niepusty znak w wierszu, aby wskazać, że reguła jest kontynuowana od pierwszego niepustego znaku w następnym wierszu. Znak minus (-) jako ostatni niepusty znak w wierszu wskazuje, że reguła jest kontynuowana od początku następnego wiersza. Znaki kontynuacji mogą występować w słowach kluczowych i parametrach.

Na przykład:

```
APPLNAME('ABC+  
D')
```

wyniki w "ABCD", oraz

```
APPLNAME('ABC-  
D')
```

wyniki w " ABC D'.

- Wiersze komentarza rozpoczynające się gwiazdką (*) mogą występować w dowolnym miejscu w tabeli reguł.
- Puste wiersze są ignorowane.
- Każda pozycja w tabeli reguł procedury obsługi DLQ składa się z jednego lub większej liczby słów kluczowych i powiązanych z nimi parametrów. Parametry muszą być zgodne z następującymi regułami składni:
 - Każda wartość parametru musi zawierać co najmniej jeden znaczący znak. Pojedynczy cudzysłów ograniczający w wartościach ujętych w znaki cudzysłowu nie jest uważany za znaczący. Na przykład te parametry są poprawne:

FORMAT (' ABC ')	3 znaczące znaki
FORMAT (ABC)	3 znaczące znaki
FORMAT (' A ')	1 znak znaczący
FORMAT (A)	1 znak znaczący
FORMAT (' ')	1 znak znaczący

Te parametry są niepoprawne, ponieważ nie zawierają znaczących znaków:

```
FORMAT ( ' ' )  
FORMAT ( )  
FORMAT ( )  
FORMAT
```

- Obsługiwane są znaki wieloznaczne. Znaku zapytania (?) można użyć zamiast dowolnego pojedynczego znaku, z wyjątkiem spacji kończących; można użyć gwiazdki (*) zamiast zera lub większej liczby sąsiadujących znaków. Gwiazdka (*) i znak zapytania (?) są **zawsze** interpretowani jako znaki wieloznaczne w wartościach parametrów.
- W parametrach tych słów kluczowych nie można dołączać znaków wieloznacznych: ACTION, HEADER, RETRY, FWDQ, FWDQM i PUTAUT.
- Odstępy końcowe w wartościach parametrów i odpowiadające im pola w komunikacie w DLQ nie są znaczące podczas wykonywania znaków wieloznacznych. Jednak początkowe i osadzone odstępy w łańcuchach, które są ujęte w apostrofach, są znaczące w przypadku zgodności ze znakami wieloznacznymi.
- Parametry liczbowe nie mogą zawierać znaku wieloznacznego znaku zapytania (?). Można użyć gwiazdki (*) zamiast całego parametru liczbowego, ale nie jako części parametru liczbowego. Na przykład są to poprawne parametry liczbowe:

MSGTYPE (2)	Zakwalifikowane są tylko komunikaty odpowiedzi
MSGTYPE (*)	Dowolny typ komunikatu jest zakwalifikowany
MSGTYPE (' * ')	Dowolny typ komunikatu jest zakwalifikowany

Jednak wartość MSGTYPE (' 2* ') nie jest poprawna, ponieważ zawiera gwiazdkę (*) jako część parametru liczbowego.

- Parametry liczbowe muszą mieścić się w zakresie od 0 do 999 999 999. Jeśli wartość parametru znajduje się w tym zakresie, jest ona akceptowana, nawet jeśli nie jest ona obecnie poprawna w polu, do którego odnosi się słowo kluczowe. Dla parametrów liczbowych można użyć nazw symbolicznych.
- Jeśli wartość łańcuchowa jest krótsza niż wartość pola w MQDLH lub MQMD, do którego odnosi się słowo kluczowe, to wartość jest dopełniona spacjami do długości pola. Jeśli wartość, z wyjątkiem gwiazdek, jest dłuższa niż pole, zostanie zdiagnozowany błąd. Na przykład są to wszystkie poprawne wartości łańcuchowe dla 8 pól znakowych:

'ABCDEFGH'	8 znaków
'A*C*E*G*I'	5 znaków z wyłączeniem gwiazdek
'*A*C*E*G*I*K*M*O* *'	8 znaków z wyłączeniem gwiazdek

- Łańcuchy zawierające spacje, małe litery lub znaki specjalne inne niż kropka (.), ukośnik (?), znak podkreślenia () i znak procentu (%) w pojedynczych cudzysłowach. Małe litery, które nie są ujęte w znaki apostrofowe, są składane wielkimi literami. Jeśli łańcuch zawiera cytat, należy użyć dwóch cudzysłowów, aby określić zarówno początek, jak i koniec cytowania. Jeśli długość łańcucha jest obliczana, każde wystąpienie podwójnych cudzysłowów jest traktowane jako pojedynczy znak.

Jak przetwarzana jest tabela reguł DLQ

Procedura obsługi kolejki niedostarczonych komunikatów przeszukuje tabelę reguł dla reguły, w której wzorec jest zgodny z komunikatem w kolejce DLQ.

Wyszukiwanie rozpoczyna się od pierwszej reguły w tabeli i jest kontynuowane sekwencyjnie przez tabelę. Gdy procedura obsługi DLQ znajdzie regułę z dopasowaniem wzorca, podejmuje ona działanie z tej reguły. Procedura obsługi DLQ zwiększa liczbę ponownych prób dla reguły o 1, gdy ta reguła stosuje tę regułę. Jeśli pierwsza próba zakończy się niepowodzeniem, procedura obsługi DLQ spróbuje ponownie, dopóki liczba prób nie będzie zgodna z liczbą określoną w słowie kluczowym RETRY. Jeśli wszystkie próby nie powiedzą się, procedura obsługi DLQ wyszuka następną zgodną regułę w tabeli.

Ten proces jest powtarzany dla kolejnych dopasowanych reguł, dopóki działanie nie powiedzie się. Jeśli dla każdej reguły dopasowania została podjęta próba, ile razy określono w słowie kluczowym RETRY, a wszystkie próby nie powiodły się, przyjmowana jest wartość ACTION (IGNORE). Przyjęto również działanie ACTION (IGNORE), jeśli nie zostanie znaleziona żadna zgodna reguła.

Uwaga:

1. Wzorce reguł dopasowywania są wyszukiwane tylko dla komunikatów w DLQ, które rozpoczynają się od MQDLH. Komunikaty, które nie rozpoczynają się od wywołania MQDLH, są okresowo zgłaszane jako błędne i pozostają w kolejce DLQ w nieskończoność.
2. Wszystkie słowa kluczowe wzorca mogą być domyślnie dozwolone, w taki sposób, że reguła może składać się tylko z działania. Należy jednak pamiętać o tym, że reguły tylko do działania są stosowane do wszystkich komunikatów w kolejce, które mają MQDLHs i które nie zostały jeszcze przetworzone zgodnie z innymi regułami w tabeli.
3. Poprawność tabeli reguł jest sprawdzana podczas uruchamiania procedury obsługi DLQ, a w tym czasie błędy są oznaczone flagą. Zmiany w tabeli reguł można wprowadzać w dowolnym momencie, ale zmiany te nie wchodzą w życie dopiero po restarcie procedury obsługi DLQ.
4. Procedura obsługi DLQ nie zmienia treści komunikatów, MQDLH ani deskryptora komunikatu. Procedura obsługi DLQ zawsze umieszcza komunikaty w innych kolejkach za pomocą opcji komunikatu MQPMO_PASS_ALL_CONTEXT.
5. Kolejne błędy składniowe w tabeli reguł mogą nie zostać rozpoznane, ponieważ tabela reguł ma na celu wyeliminowanie generowania powtarzających się błędów podczas sprawdzania poprawności.
6. Procedura obsługi DLQ otwiera DLQ z opcją MQOO_INPUT_AS_Q_DEF.
7. Wiele instancji procedury obsługi DLQ może działać współbieżnie z tą samą kolejką, korzystając z tej samej tabeli reguł. Jednak bardziej zwykle jest to relacja jeden do jednego między DLQ a procedurą obsługi DLQ.

Informacje pokrewne

Kolejki niedostarczane

[Rozwiązywanie problemów z niedostarczonym](#)

Sprawdzanie, czy wszystkie komunikaty DLQ są przetwarzane

Procedura obsługi kolejki niedostarczonych komunikatów przechowuje zapis wszystkich komunikatów w kolejce DLQ, które zostały wyświetlone, ale nie zostały usunięte.

Jeśli procedura obsługi DLQ jest używana jako filtr w celu wyodrębnienia niewielkiego podzbioru komunikatów z kolejki DLQ, procedura obsługi DLQ nadal musi zapisywać te komunikaty w DLQ, które nie zostały przetworzone. Ponadto procedura obsługi DLQ nie może zagwarantować, że nowe komunikaty dojeżdżające do kolejki DLQ są widoczne, nawet jeśli DLQ jest zdefiniowane jako FIFO (first-in-first-out). Jeśli kolejka nie jest pusta, kolejka DLQ jest okresowo ponownie skanowana w celu sprawdzenia wszystkich komunikatów.

Z tych powodów spróbuj upewnić się, że DLQ zawiera jak najmniejszej liczby komunikatów. Jeśli komunikaty, których nie można usunąć lub przesłać do innych kolejek (z jakiegokolwiek powodu), mogą gromadzić się w kolejce, obciążenie procedury obsługi DLQ zwiększa się, a sam DLQ może się wypełnić.

Istnieje możliwość podjęcia określonych działań, aby umożliwić obsługę opróżnienia DLQ przez procedurę obsługi DLQ. Na przykład, spróbuj nie używać ACTION (IGNORE), który pozostawia komunikaty w DLQ. (Pamiętaj, że dla komunikatów, które nie są jawnie adresowane przez inne reguły w tabeli, przyjmowana jest wartość ACTION (IGNORE).) Zamiast tego w przypadku komunikatów, które w przeciwnym razie należy zignorować, należy użyć działania, które przenosi komunikaty do innej kolejki. Na przykład:

```
ACTION (FWD) FWDQ (IGNORED.DEAD.QUEUE) HEADER (YES)
```

Analogicznie należy wprowadzić regułę końcową w tabeli a catchall w celu przetwarzania komunikatów, które nie zostały uwzględnione we wcześniejszych regułach w tabeli. Na przykład ostateczna reguła w tabeli może być taka, jak ta:

```
ACTION (FWD) FWDQ (REALLY.DEAD.QUEUE) HEADER (YES)
```

Powoduje to przekaz komunikatów, które zapadają do ostatecznej reguły w tabeli do kolejki REALLY.DEAD.QUEUE, gdzie mogą być przetwarzane ręcznie. Jeśli użytkownik nie ma takiej reguły, komunikaty prawdopodobnie pozostaną w kolejce DLQ w nieskończoność.

Przykładowa tabela reguł procedury obsługi DLQ

Przykładowa tabela reguł w kolejce niedostarczonych komunikatów dla komendy `runmqdlq` zawierająca jedną pozycję kontroli-dane i kilka reguł.

```
*****
*   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)
```

Informacje pokrewne

[Kolejki niedostarczane](#)

[Rozwiązywanie problemów z niedostarczonym](#)

[runmqdlq \(uruchomienie procedury obsługi kolejki niedostarczonych komunikatów\)](#)

Praca z kolejkami modelami

Menedżer kolejek tworzy *kolejkę dynamiczną*, jeśli odbierze wywołanie MQI z aplikacji, określając nazwę kolejki, która została zdefiniowana jako kolejka modelowa. Nazwa nowej kolejki dynamicznej jest generowana przez menedżer kolejek po utworzeniu kolejki. *Kolejka modelowa* jest to szablon, który określa atrybuty wszystkich kolejek dynamicznych utworzonych na podstawie tego szablonu. Kolejki modelowe stanowią wygodną metodę tworzenia kolejek zgodnie z wymaganiami.

Definiowanie kolejki modelowej

Komenda **DEFINE QMODEL** służy do definiowania kolejki modelowej z zestawem atrybutów w taki sam sposób, w jaki definiuje się kolejkę lokalną. Kolejki modelowe i kolejki lokalne mają ten sam zestaw atrybutów, z tym wyjątkiem, że w kolejkach modelowych można określić, czy utworzone kolejki dynamiczne mają charakter tymczasowy, czy trwałe. (Kolejki stałe są obsługiwane przez restarty menedżera kolejek, tymczasowe nie są dostępne). Na przykład:

```
DEFINE QMODEL (GREEN.MODEL.QUEUE) +
```

```
DESCR('Queue for messages from application X') +  
PUT (DISABLED) +  
GET (ENABLED) +  
NOTRIGGER +  
MSGDLVSQ (FIFO) +  
MAXDEPTH (1000) +  
MAXMSGL (2000) +  
USAGE (NORMAL) +  
DEFTYPE (PERMDYN)
```

Ta komenda tworzy definicję kolejki modelowej. Z atrybutu **DEFTYPE** można zauważyć, że rzeczywiste kolejki utworzone na podstawie tego szablonu są trwałymi kolejkami dynamicznymi. Wszystkie nieokreślone atrybuty są automatycznie kopiowane ze strony `SYSYTEM.DEFAULT.MODEL.QUEUE`, kolejka domyślna.

Atrybutów **LIKE** i **REPLACE** można używać podczas definiowania kolejek modelowych, w ten sam sposób, w jaki są używane z kolejkami lokalnymi.

Korzystanie z innych komend z kolejkami modelami

Za pomocą odpowiednich komend MQSC można wyświetlać lub zmieniać atrybuty kolejki modelowej, a także usuwać obiekty kolejki modelowej. Na przykład:

Użyj komendy **DISPLAY QUEUE**, aby wyświetlić atrybuty kolejki modelowej:

```
DISPLAY QUEUE (GREEN.MODEL.QUEUE)
```

Użyj komendy **ALTER QMODEL**, aby zmienić model, aby włączyć operacje umieszczania w dowolnej kolejce dynamicznej utworzonej z tego modelu:

```
ALTER QMODEL (BLUE.MODEL.QUEUE) PUT(ENABLED)
```

Aby usunąć tę kolejkę modelową, użyj komendy **DELETE QMODEL**:

```
DELETE QMODEL (RED.MODEL.QUEUE)
```

Informacje pokrewne

[ALTER QMODEL](#)

[DEFINE QMODEL](#)

[USUŃ QMODEL](#)

[WYŚWIETL KOLEJKĘ](#)

Praca z tematami administracyjnymi

Aby zarządzać tematami administracyjnymi, należy użyć komend MQSC.

Szczegółowe informacje na temat tych komend zawiera sekcja [Komendy MQSC](#).

Pojęcia pokrewne

[“Definiowanie tematu administracyjnego” na stronie 177](#)

Aby utworzyć temat administracyjny, należy użyć komendy MQSC **DEFINE TOPIC**. Definiując temat administracyjny, można opcjonalnie ustawić każdy atrybut tematu.

[“Wyświetlanie atrybutów obiektu tematu administracyjnego” na stronie 177](#)

Aby wyświetlić obiekt tematu administracyjnego, należy użyć komendy MQSC **DISPLAY TOPIC**.

[“Zmiana atrybutów tematu administracyjnego” na stronie 178](#)

Atrybuty tematu można zmieniać na dwa sposoby, korzystając z komendy **ALTER TOPIC** lub **DEFINE TOPIC** z atrybutem **REPLACE**.

[“Kopiowanie definicji tematu administracyjnego” na stronie 178](#)

Definicję tematu można skopiować za pomocą atrybutu **LIKE** w komendzie **DEFINE**.

“Usuwanie definicji tematu administracyjnego” na stronie 179

Aby usunąć temat administracyjny, można użyć komendy MQSC **DELETE TOPIC** .

Informacje pokrewne

Administracyjne obiekty tematów

Definiowanie tematu administracyjnego

Aby utworzyć temat administracyjny, należy użyć komendy MQSC **DEFINE TOPIC** . Definiując temat administracyjny, można opcjonalnie ustawić każdy atrybut tematu.

Dowolny atrybut tematu, który nie jest jawnie ustawiony, jest dziedziczony z domyślnego tematu administracyjnego, SYSTEM.DEFAULT.TOPIC(temat), który został utworzony podczas instalacji systemu.

Na przykład komenda **DEFINE TOPIC** , która jest następująca, definiuje temat o nazwie ORANGE . TOPIC o następujących parametrach:

- Rozwiązuje się w łańcuchu tematu ORANGE. Więcej informacji na temat sposobu użycia łańcuchów tematów zawiera sekcja Łączenie łańcuchów tematów.
- Każdy atrybut, który jest ustawiony na ASPARENT, używa atrybutu zdefiniowanego przez nadrzędny temat w tym temacie. To działanie jest powtarzane w drzewie tematów, jeśli jest to główny temat, SYSTEM.BASE.TOPIC . Więcej informacji na ten temat zawiera sekcja Drzewa tematów.

```
DEFINE TOPIC (ORANGE.TOPIC) +  
TOPICSTR (ORANGE) +  
DEFPRTY(ASPARENT) +  
NPMSGDLV(ASPARENT)
```

Uwaga:

- Wszystkie wartości atrybutów są wartościami domyślnymi, z wyjątkiem wartości w łańcuchu tematu. Są one pokazywane tutaj tylko jako ilustracja. Można je pominąć, jeśli użytkownik ma pewność, że wartości domyślne są takie, które mają zostać zmienione lub które nie zostały zmienione. Patrz także “Wyświetlanie atrybutów obiektu tematu administracyjnego” na stronie 177.
- Jeśli w tym samym menedżerze kolejek znajduje się już temat administracyjny o nazwie ORANGE.TOPIC-ta komenda nie powiedzie się. Użyj atrybutu REPLACE, jeśli chcesz nadpisać istniejącą definicję tematu, ale zobacz także “Zmiana atrybutów tematu administracyjnego” na stronie 178.

Informacje pokrewne

ZDEFINIUJ TEMAT

Wyświetlanie atrybutów obiektu tematu administracyjnego

Aby wyświetlić obiekt tematu administracyjnego, należy użyć komendy MQSC **DISPLAY TOPIC** .

Aby wyświetlić wszystkie tematy, należy użyć:

```
DISPLAY TOPIC(ORANGE.TOPIC)
```

Można selektywnie wyświetlać atrybuty, określając je pojedynczo za pomocą komendy **DISPLAY TOPIC** . Na przykład:

```
DISPLAY TOPIC(ORANGE.TOPIC) +  
TOPICSTR +  
DEFPRTY +  
NPMSGDLV
```

Ta komenda wyświetla trzy określone atrybuty w następujący sposób:

```
AMQ8633: Display topic details.  
TOPIC(ORANGE.TOPIC) TYPE(LOCAL)
```

```
TOPICSTR(ORANGE)
NPMSGDLV(ASPARENT)
```

```
DEFPRTY(ASPARENT)
```

Aby wyświetlić tematy ASPARENT, które są używane w czasie wykonywania, należy użyć komendy **DISPLAY TPSTATUS** . Na przykład:

```
DISPLAY TPSTATUS(ORANGE) DEFPRTY NPMSGDLV
```

Komenda wyświetla następujące szczegóły:

```
AMQ8754: Display topic status details.
TOPICSTR(ORANGE)           DEFPRTY(0)
NPMSGDLV(ALLAVAIL)
```

Po zdefiniowaniu tematu administracyjnego wszystkie atrybuty, które nie zostały jawnie określone w domyślnym temacie administracyjnym, są określane jako SYSTEM.DEFAULT.TOPIC. Aby sprawdzić, jakie są te atrybuty domyślne, należy użyć następującej komendy:

```
DISPLAY TOPIC (SYSTEM.DEFAULT.TOPIC)
```

Informacje pokrewne

[WYŚWIETL TEMAT](#)

[WYŚWIETL STATUS TPSTATUS](#)

Zmiana atrybutów tematu administracyjnego

Atrybuty tematu można zmieniać na dwa sposoby, korzystając z komendy **ALTER TOPIC** lub **DEFINE TOPIC** z atrybutem **REPLACE** .

Jeśli na przykład ma zostać zmieniony domyślny priorytet komunikatów dostarczanych do tematu o nazwie ORANGE.TOPIC, aby mieć wartość 5, należy użyć jednej z następujących komend.

- Za pomocą komendy **ALTER** :

```
ALTER TOPIC(ORANGE.TOPIC) DEFPRTY(5)
```

Ta komenda zmienia pojedynczy atrybut, który ma domyślny priorytet komunikatu dostarczanego do tego tematu na wartość 5; wszystkie pozostałe atrybuty pozostają takie same.

- Za pomocą komendy **DEFINE** :

```
DEFINE TOPIC(ORANGE.TOPIC) DEFPRTY(5) REPLACE
```

Ta komenda zmienia domyślny priorytet komunikatów dostarczonych do tego tematu. Wszystkie pozostałe atrybuty mają podane wartości domyślne.

Jeśli użytkownik zmieni priorytet komunikatów wysyłanych do tego tematu, nie ma to wpływu na istniejące komunikaty. Jednak każdy nowy komunikat używa określonego priorytetu, jeśli nie jest on udostępniany przez aplikację publikacją.

Informacje pokrewne

[ALTER TOPIC](#)

[WYŚWIETL TEMAT](#)

Kopiowanie definicji tematu administracyjnego

Definicję tematu można skopiować za pomocą atrybutu LIKE w komendzie **DEFINE** .

Na przykład:

```
DEFINE TOPIC (MAGENTA.TOPIC) +  
LIKE (ORANGE.TOPIC)
```

Ta komenda powoduje utworzenie tematu MAGENTA.TOPIC, z tymi samymi atrybutami, co w oryginalnym temacie, ORANGE.TOPIC, a nie te z domyślnego tematu administracyjnego systemu. Wprowadź nazwę tematu, który ma zostać skopiowany dokładnie tak, jak został wprowadzony podczas tworzenia tematu. Jeśli nazwa zawiera małe litery, należy ująć ją w znaki pojedynczego cudzysłowu.

Można również użyć tego formularza komendy **DEFINE**, aby skopiować definicję tematu, ale dokonać zmian w atrybutach oryginału. Na przykład:

```
DEFINE TOPIC (BLUE.TOPIC) +  
TOPICSTR (BLUE) +  
LIKE (ORANGE.TOPIC)
```

Istnieje również możliwość skopiowania atrybutów tematu BLUE.TOPIC (TEMAT) do tematu GREEN.TOPIC (temat) i określa, że gdy publikacje nie mogą być dostarczane do odpowiedniej kolejki subskrybenta, nie są umieszczane w kolejce niedostarczonych komunikatów. Na przykład:

```
DEFINE TOPIC (GREEN.TOPIC) +  
TOPICSTR (GREEN) +  
LIKE (BLUE.TOPIC) +  
USEDLQ (NO)
```

Informacje pokrewne

[ZDEFINIUIJ TEMAT](#)

Usuwanie definicji tematu administracyjnego

Aby usunąć temat administracyjny, można użyć komendy MQSC **DELETE TOPIC**.

Na przykład:

```
DELETE TOPIC (ORANGE.TOPIC)
```

Aplikacje nie będą mogły otworzyć tematu w celu opublikowania lub udostępnienia nowych subskrypcji przy użyciu nazwy obiektu ORANGE.TOPIC. Publikowanie aplikacji, które mają otwarty temat, są w stanie kontynuować publikowanie przetłumaczonego łańcucha tematu. Wszystkie subskrypcje już wprowadzone w tym temacie będą nadal otrzymywać publikacje po usunięciu tematu.

Aplikacje, które nie odwołują się do tego obiektu tematu, ale używają rozstrzygniętego łańcucha tematu, który ten obiekt tematu reprezentował, 'ORANGE' w tym przykładzie, kontynuują pracę. W tym przypadku dziedziczą właściwości z obiektu tematu wyżej w drzewie tematów. Więcej informacji na ten temat zawiera sekcja [Drzewa tematów](#).

Informacje pokrewne

[Usuń temat](#)

Praca z subskrypcjami

Do zarządzania subskrypcjami można używać komend MQSC.

Subskrypcje mogą być jednym z trzech typów zdefiniowanych w atrybucie **SUBTYPE** :

ADMINISTRATOR

Administracyjnie definiowane przez użytkownika.

PROXY

Subskrypcja utworzona wewnętrznie w celu kierowania publikacji między menedżerami kolejek.

Interfejs API

Tworzone programowo, na przykład przy użyciu wywołania MQSUB MQI.

Szczegółowe informacje na temat tych komend zawiera sekcja [Komendy MQSC](#) .

Pojęcia pokrewne

[“Definiowanie subskrypcji administracyjnej” na stronie 180](#)

Aby utworzyć subskrypcję administracyjną, należy użyć komendy MQSC **DEFINE SUB** . Można również użyć wartości domyślnej zdefiniowanej w domyślnej definicji subskrypcji lokalnej. Można także zmodyfikować parametry subskrypcji z poziomu domyślnej subskrypcji lokalnej SYSTEM.DEFAULT.SUB , który został utworzony podczas instalowania systemu.

[“Wyświetlanie atrybutów subskrypcji” na stronie 181](#)

Za pomocą komendy **DISPLAY SUB** można wyświetlić skonfigurowane atrybuty dowolnej subskrypcji znanej menedżerowi kolejek.

[“Zmiana atrybutów subskrypcji lokalnej” na stronie 182](#)

Atrybuty subskrypcji można zmienić na dwa sposoby, korzystając z komendy **ALTER SUB** lub **DEFINE SUB** z atrybutem **REPLACE** .

[“Kopiowanie definicji subskrypcji lokalnej” na stronie 182](#)

Definicję subskrypcji można skopiować za pomocą atrybutu **LIKE** w komendzie **DEFINE** .

[“Usuwanie subskrypcji lokalnej” na stronie 182](#)

Aby usunąć subskrypcję lokalną, można użyć komendy MQSC **DELETE SUB** .

Definiowanie subskrypcji administracyjnej

Aby utworzyć subskrypcję administracyjną, należy użyć komendy MQSC **DEFINE SUB** . Można również użyć wartości domyślnej zdefiniowanej w domyślnej definicji subskrypcji lokalnej. Można także zmodyfikować parametry subskrypcji z poziomu domyślnej subskrypcji lokalnej SYSTEM.DEFAULT.SUB , który został utworzony podczas instalowania systemu.

Na przykład następująca komenda **DEFINE SUB** definiuje subskrypcję o nazwie ORANGE o następujących parametrach:

- Trwała subskrypcja, co oznacza, że utrzymuje się ona po restarcie menedżera kolejek z nieograniczonym terminem utraty ważności.
- Otrzymuj publikacje wykonane w łańcuchu tematu ORANGE z priorytetami komunikatów ustawionym przez aplikacje publikowania.
- Publikacje dostarczone dla tej subskrypcji są wysyłane do kolejki lokalnej SUBQ, a ta kolejka musi być zdefiniowana przed zdefiniowaniem subskrypcji.

```
DEFINE SUB (ORANGE) +  
TOPICSTR (ORANGE) +  
DESTCLAS (PROVIDED) +  
DEST (SUBQ) +  
EXPIRY (UNLIMITED) +  
PUBPRTY (AS PUB)
```

Uwaga:

- Subskrypcja i nazwa łańcucha tematu nie muszą być zgodne.
- Wszystkie wartości atrybutów są wartościami domyślnymi, z wyjątkiem wartości w łańcuchu docelowym i łańcuchowym tematu. Są one pokazywane tutaj tylko jako ilustracja. Można je pominąć, jeśli użytkownik ma pewność, że wartości domyślne są takie, które mają zostać zmienione lub które nie zostały zmienione. Patrz także [“Wyświetlanie atrybutów subskrypcji” na stronie 181](#).
- Jeśli w tym samym menedżerze kolejek istnieje już subskrypcja lokalna o nazwie ORANGE, wykonanie tej komendy nie powiedzie się. Użyj atrybutu **REPLACE** , jeśli chcesz nadpisać istniejącą definicję kolejki, ale zobacz także [“Zmiana atrybutów subskrypcji lokalnej” na stronie 182](#).
- Jeśli kolejka SUBQ nie istnieje, wykonanie tej komendy nie powiedzie się.

Informacje pokrewne

[DEFINE SUB](#)

Wyświetlanie atrybutów subskrypcji

Za pomocą komendy **DISPLAY SUB** można wyświetlić skonfigurowane atrybuty dowolnej subskrypcji znanej menedżerowi kolejek.

Na przykład:

```
DISPLAY SUB(ORANGE)
```

Można selektywnie wyświetlać atrybuty, określając je pojedynczo. Na przykład:

```
DISPLAY SUB(ORANGE) +  
SUBID +  
TOPICSTR +  
DURABLE
```

Ta komenda wyświetla trzy określone atrybuty w następujący sposób:

```
AMQ8096: IBM MQ subscription inquired.  
SUBID(414D51204141412020202020202020EE921E4E20002A03)  
SUB(ORANGE) TOPICSTR(ORANGE)  
DURABLE(YES)
```

TOPICSTR jest rozstrzygniętym łańcuchem tematu, na którym działa ten subskrybent. Gdy subskrypcja jest zdefiniowana w celu użycia obiektu tematu, łańcuch tematu z tego obiektu jest używany jako przedrostek do łańcucha tematu udostępnianego podczas wykonywania subskrypcji. Identyfikator SUBID jest unikalnym identyfikatorem przypisanym przez menedżer kolejek po utworzeniu subskrypcji. Jest to przydatny atrybut do wyświetlenia, ponieważ niektóre nazwy subskrypcji mogą być długie lub w różnych zestawach znaków, dla których może stać się niepraktyczne.

Alternatywną metodą wyświetlania subskrypcji jest użycie identyfikatora SUBID:

```
DISPLAY SUB +  
SUBID(414D51204141412020202020202020EE921E4E20002A03) +  
TOPICSTR +  
DURABLE
```

Ta komenda daje te same dane wyjściowe, co wcześniej:

```
AMQ8096: IBM MQ subscription inquired.  
SUBID(414D51204141412020202020202020EE921E4E20002A03)  
SUB(ORANGE) TOPICSTR(ORANGE)  
DURABLE(YES)
```

Subskrypcje proxy w menedżerze kolejek nie są domyślnie wyświetlane. Aby wyświetlić je, należy określić **SUBTYPE** serwera PROXY lub ALL.

Aby wyświetlić atrybuty środowiska wykonawczego, można użyć komendy `DISPLAY SBSTATUS`. Na przykład użyj komendy:

```
DISPLAY SBSTATUS(ORANGE) NUMMSGs
```

Zostaną wyświetlone następujące dane wyjściowe:

```
AMQ8099: IBM MQ subscription status inquired.  
SUB(ORANGE)  
SUBID(414D51204141412020202020202020EE921E4E20002A03)  
NUMMSGs(0)
```

Po zdefiniowaniu subskrypcji administracyjnej wszystkie atrybuty, które nie zostały jawnie określone z subskrypcji domyślnej, są określane jako SYSTEM.DEFAULT.SUB. Aby sprawdzić, jakie są te atrybuty domyślne, należy użyć następującej komendy:

```
DISPLAY SUB (SYSTEM.DEFAULT.SUB)
```

Informacje pokrewne

[WYŚWIETL SUB](#)

Zmiana atrybutów subskrypcji lokalnej

Atrybuty subskrypcji można zmienić na dwa sposoby, korzystając z komendy **ALTER SUB** lub **DEFINE SUB** z atrybutem **REPLACE**.

Jeśli na przykład chcesz zmienić priorytet komunikatów dostarczanych do subskrypcji o nazwie ORANGE na 5, użyj jednej z następujących komend:

- Za pomocą komendy **ALTER** :

```
ALTER SUB(ORANGE) PUBPRTY(5)
```

Ta komenda zmienia pojedynczy atrybut, który ma priorytet komunikatów dostarczanych do tej subskrypcji na 5; wszystkie pozostałe atrybuty pozostają takie same.

- Za pomocą komendy **DEFINE** :

```
DEFINE SUB(ORANGE) PUBPRTY(5) REPLACE
```

Ta komenda zmienia nie tylko priorytet komunikatów dostarczanych do tej subskrypcji, ale wszystkie inne atrybuty, które mają podane wartości domyślne.

Jeśli użytkownik zmieni priorytet komunikatów wysyłanych do tej subskrypcji, nie ma to wpływu na istniejące komunikaty. Wszystkie nowe komunikaty mają jednak określony priorytet.

Informacje pokrewne

[ALTER SUB](#)

[DEFINE SUB](#)

Kopiowanie definicji subskrypcji lokalnej

Definicję subskrypcji można skopiować za pomocą atrybutu **LIKE** w komendzie **DEFINE**.

Na przykład:

```
DEFINE SUB(BLUE) +  
  LIKE(ORANGE)
```

Można również skopiować atrybuty elementu podrzędnego REAL do podkomendy THIRDSUBi określić, że identyfikatorem correlID dostarczoną publikacji jest THIRDSUB, a nie publikatorów correlID. Na przykład:

```
DEFINE SUB(THIRDSUB) +  
  LIKE(BLUE) +  
  DESTCORL(ORANGE)
```

Informacje pokrewne

[DEFINE SUB](#)

Usuwanie subskrypcji lokalnej

Aby usunąć subskrypcję lokalną, można użyć komendy MQSC **DELETE SUB**.

```
DELETE SUB(ORANGE)
```

Subskrypcję można również usunąć za pomocą identyfikatora SUBID:

```
DELETE SUB SUBID(414D51204141412020202020202020EE921E4E20002A03)
```

Informacje pokrewne

[USUŃ SUB](#)

Sprawdzanie komunikatów w subskrypcji

Gdy zdefiniowana jest subskrypcja, jest ona powiązana z kolejką. Opublikowane komunikaty zgodne z tą subskrypcją są umieszczane w tej kolejce.

O tym zadaniu

Należy pamiętać, że następujące komendy **runmqsc** pokazują tylko te subskrypcje, które otrzymały komunikaty.

Aby sprawdzić komunikaty znajdujące się obecnie w kolejce w celu wykonania subskrypcji, wykonaj następujące kroki:

Procedura

1. Informacje na temat sprawdzania komunikatów w kolejce dla typu subskrypcji DISPLAY SBSTATUS(*sub_name*) NUMMSG zawiera sekcja [“Wyświetlanie atrybutów subskrypcji” na stronie 181](#).
2. Jeśli wartość **NUMMSG** jest większa od zera, należy zidentyfikować kolejkę powiązaną z subskrypcją, wpisując DISPLAY SUB(*sub_name*) DEST.
3. Za pomocą nazwy kolejki zwróconej można wyświetlić komunikaty zgodnie z techniką opisaną w sekcji [“Przeglądanie kolejek” na stronie 152](#).

Informacje pokrewne

[WYŚWIETL STATUS SBSTATUS](#)

Praca z usługami

Obiekty usług są środkiem, za pomocą którego można zarządzać dodatkowymi procesami jako część menedżera kolejek. Za pomocą usług można definiować programy, które są uruchamiane i zatrzymane po uruchomieniu i zakończeniu menedżera kolejek. Usługi produktu IBM MQ są zawsze uruchamiane przy użyciu identyfikatora użytkownika, który uruchomił menedżer kolejek.

Aby zdefiniować nową definicję usługi IBM MQ, należy użyć komendy MQSC DEFINE SERVICE.

Obiekty usług mogą być jednym z następujących typów:

Serwer

Serwer jest obiektem usługi, który ma parametr **SERVTYPE** określony jako SERWER. Obiekt usługi serwera jest to definicja programu, który jest wykonywany podczas uruchamiania określonego menedżera kolejek. Obiekty usług serwera definiują programy, które zwykle są uruchamiane przez długi czas. Na przykład obiekt usługi serwera może być używany do wykonywania procesu monitora wyzwalacza, takiego jak **runmqtrm**.

Jednocześnie może być uruchamiana tylko jedna instancja obiektu usługi serwera. Status działających obiektów usług serwera może być monitorowany za pomocą komendy MQSC **DISPLAY SVSTATUS**.

Komenda

Komenda jest obiektem usługi, dla którego parametr **SERVTYPE** jest określony jako COMMAND. Obiekty usług komend są podobne do obiektów usług serwera, jednak wiele instancji obiektu usługi komendy

może być uruchamiane współbieżnie, a ich status nie może być monitorowany za pomocą komendy **MQSC DISPLAY SVSTATUS**.

Jeśli komenda **MQSC STOP SERVICE** jest wykonywana, nie jest wykonywane sprawdzanie, czy program uruchomiony przez komendę **MQSC START SERVICE** jest nadal aktywny przed zatrzymaniem programu.

Informacje pokrewne

WYŚWIETL STATUS SVSTATUS

Uruchom usługę

Zatrzymaj usługę

Definiowanie obiektu usługi

Obiekt usługi definiuje się za pomocą komendy **MQSC DEFINE SERVICE**.

Atrybuty, które należy zdefiniować, są następujące:

SERVTYPE

Definiuje typ obiektu usługi. Lista poprawnych wartości:

SERVER

Obiekt usługi serwera.

W danym momencie może być wykonywana tylko jedna instancja obiektu usługi serwera.

Status obiektów usług serwera może być monitorowany za pomocą komendy **MQSC DISPLAY SVSTATUS**.

KOMENDA

Obiekt usługi komendy.

Wiele instancji obiektu usługi komendy może być wykonywanych współbieżnie. Status obiektów usługi komendy nie może być monitorowany.

STARTCMD

Program, który jest wykonywany w celu uruchomienia usługi. Należy podać pełną ścieżkę do programu.

STARTARG

Argumenty przekazane do programu startowego.

STDERR

Określa ścieżkę do pliku, do którego ma zostać przekierowany błąd standardowy (stderr) programu usługowego.

STDOUT

Określa ścieżkę do pliku, do którego powinny zostać przekierowane standardowe wyjście (wyjście standardowe) programu usługowego.

STOPCMD

Program, który jest wykonywany w celu zatrzymania usługi. Należy podać pełną ścieżkę do programu.

STOPARG

Argumenty przekazane do programu zatrzymania.

CONTROL

Określa sposób, w jaki usługa ma być uruchomiona i zatrzymana:

RĘCZNE

Usługa nie jest automatycznie uruchamiana lub zatrzymana automatycznie. Jest on sterowany za pomocą komend **START SERVICE** i **STOP SERVICE**. Jest to wartość domyślna.

QMGR

Zdefiniowana usługa jest uruchamiana i zatrzymana w tym samym czasie, co menedżer kolejek, który jest uruchamiany i zatrzymany.

TYLKO startonly

Usługa ma zostać uruchomiona w tym samym czasie co menedżer kolejek, ale nie jest wymagana do zatrzymania, gdy menedżer kolejek jest zatrzymany.

Pojęcia pokrewne

[“Zarządzanie usługami” na stronie 185](#)

Za pomocą parametru **CONTROL** instancja obiektu usługi może być uruchamiana i zatrzymana automatycznie przez menedżer kolejek lub została uruchomiona i zatrzymana za pomocą komend MQSC **START SERVICE** i **STOP SERVICE**.

Informacje pokrewne

[Definiuj usługę](#)

[WYŚWIETL STATUS SVSTATUS](#)

[Uruchom usługę](#)

[Zatrzymaj usługę](#)

Zarządzanie usługami

Za pomocą parametru **CONTROL** instancja obiektu usługi może być uruchamiana i zatrzymana automatycznie przez menedżer kolejek lub została uruchomiona i zatrzymana za pomocą komend MQSC **START SERVICE** i **STOP SERVICE**.

Gdy instancja obiektu usługi jest uruchamiana, w dzienniku błędów menedżera kolejek zapisywany jest komunikat zawierający nazwę obiektu usługi i identyfikator procesu uruchomionego procesu. Poniżej przedstawiono przykładową pozycję dziennika dla obiektu usługi serwera:

```
02/15/2005 11:54:24 AM - Process(10363.1) User(mqm) Program(amqzmgr0)
Host(HOST_1) Installation(Installation1)
VRMF(7.1.0.0) QMgr(A.B.C)
AMQ5028: The Server 'S1' has started. ProcessId(13031).
```

```
EXPLANATION:
The Server process has started.
ACTION:
None.
```

Przykładowy wpis dziennika dla obiektu usługi komendy jest następujący:

```
02/15/2005 11:53:55 AM - Process(10363.1) User(mqm) Program(amqzmgr0)
Host(HOST_1) Installation(Installation1)
VRMF(7.1.0.0) QMgr(A.B.C)
AMQ5030: The Command 'C1' has started. ProcessId(13030).
```

```
EXPLANATION:
The Command has started.
ACTION:
None.
```

Gdy usługa serwera instancji zostanie zatrzymana, do dzienników błędów menedżera kolejek zostanie zapisany komunikat zawierający nazwę usługi i identyfikator procesu zakończenia. Poniżej znajduje się przykładowa pozycja dziennika dla obiektu usługi serwera:

```
02/15/2005 11:54:54 AM - Process(10363.1) User(mqm) Program(amqzmgr0)
Host(HOST_1) Installation(Installation1)
VRMF(7.1.0.0) QMgr(A.B.C)
AMQ5029: The Server 'S1' has ended. ProcessId(13031).
```

```
EXPLANATION:
The Server process has ended.
ACTION:
None.
```

Odsyłacze pokrewne

[“Dodatkowe zmienne środowiskowe” na stronie 186](#)

Gdy usługa jest uruchamiana, środowisko, w którym uruchamiany jest proces usługi, jest dziedziczone ze środowiska menedżera kolejek. Istnieje możliwość zdefiniowania dodatkowych zmiennych środowiskowych, które mają być ustawione w środowisku procesu usługi przez dodanie zmiennych, które mają zostać zdefiniowane w jednym z plików przestania środowiska produktu `service.env`.

Informacje pokrewne

[Zatrzymaj usługę](#)

[Uruchom usługę](#)

Dodatkowe zmienne środowiskowe



Gdy usługa jest uruchamiana, środowisko, w którym uruchamiany jest proces usługi, jest dziedziczone ze środowiska menedżera kolejek. Istnieje możliwość zdefiniowania dodatkowych zmiennych środowiskowych, które mają być ustawione w środowisku procesu usługi przez dodanie zmiennych, które mają zostać zdefiniowane w jednym z plików przestania środowiska produktu `service.env`.

Pliki, do których można dodawać zmienne środowiskowe

Istnieją dwa możliwe pliki, do których można dodać zmienne środowiskowe:



Plik `service.env` zasięgu komputera

Ten plik znajduje się w:

-  `/var/mqm` w systemach UNIX and Linux .
-  Katalog danych wybrany podczas instalacji w systemach Windows .

Plik `service.env` zasięgu menedżera kolejek

Ten plik znajduje się w katalogu danych menedżera kolejek. Na przykład położenie pliku przestania środowiska dla menedżera kolejek o nazwie `QMNAME` jest następujące:

- –  W systemach UNIX and Linux `/var/mqm/qmgrs/QMNAME/service.env`
- –  W systemach Windows
`C:\ProgramData\IBM\MQ\qmgrs\QMNAME\service.env`

Oba pliki są przetwarzane, o ile są dostępne, z definicjami w pliku zasięgu menedżera kolejek, które mają pierwszeństwo przed tymi definicjami w pliku zasięgu maszyny.

Zmienne środowiskowe, które można określić w programie `service.env`.

Dowolną zmienną środowiskową można określić w produkcie `service.env`. Na przykład, jeśli usługa IBM MQ uruchamia wiele komend, może okazać się przydatne ustawienie zmiennej środowiskowej `PATH` w pliku `service.env`. Wartości ustawione przez użytkownika nie mogą być zmiennymi środowiskowymi, na przykład `CLASSPATH= %CLASSPATH%` jest niepoprawne. Podobnie w systemie Linux `PATH= $PATH : /opt/mqm/bin` dałoby nieoczekiwane rezultaty.

Zmienna `CLASSPATH` musi być skapitalizowana, a instrukcja ścieżki klasy może zawierać tylko literały. Niektóre usługi (na przykład telemetria) ustawiają własną ścieżkę klas. Zostanie do niego dodana ścieżka `CLASSPATH` zdefiniowana w produkcie `service.env`.

Format zmiennych zdefiniowanych w pliku, `service.env` to lista par nazw i zmiennych wartości. Każda zmienna musi być zdefiniowana w nowym wierszu, a każda zmienna jest podejmowana w sposób jawny, w tym biały znak.

Przykład `service.env`

```
##*****#
##*                                           *#
##* <N_OCO_COPYRIGHT>                       *#
```

```

## Licensed Materials - Property of IBM                *#
##                                                    *#
## 63H9336                                           *#
## (C) Copyright IBM Corporation 2005, 2023.         *#
##                                                    *#
## <NOC_COPYRIGHT>                                   *#
##                                                    *#
#####
## Module Name: service.env                          *#
## Type       : IBM MQ service environment file      *#
## Function    : Define additional environment variables to be set *#
##            : for SERVICE programs.              *#
## Usage      : <VARIABLE>=<VALUE>                 *#
##                                                    *#
#####
MYLOC=/opt/myloc/bin
MYTMP=/tmp
TRACEDIR=/tmp/trace
MYINITQ=ACCOUNTS.INITIATION.QUEUE

```

Odsyłacze pokrewne

“Wymienne wstawki w definicjach usług” na stronie 187

W definicji obiektu usługi możliwe jest podstawienia leksemów. Znaczniki, które są zastępowane, są automatycznie zastępowane rozszerzonym tekstem, gdy wykonywany jest program usługowy. Znaczniki zastępcze mogą być pobierane z następującej listy wspólnych znaczników lub ze wszystkich zmiennych zdefiniowanych w pliku service.env.

Wymienne wstawki w definicjach usług

W definicji obiektu usługi możliwe jest podstawienia leksemów. Znaczniki, które są zastępowane, są automatycznie zastępowane rozszerzonym tekstem, gdy wykonywany jest program usługowy. Znaczniki zastępcze mogą być pobierane z następującej listy wspólnych znaczników lub ze wszystkich zmiennych zdefiniowanych w pliku service.env.




Poniżej znajdują się wspólne znaczniki, których można użyć do podstawienia znaczników w definicji obiektu usługi:

ŚCIEŻKA_INSTALACJI_MQ

Położenie, w którym zainstalowano produkt IBM MQ .

ŚCIEŻKA_DATOWANA_MQ

Położenie katalogu danych produktu IBM MQ :

-   W systemach UNIX and Linux położeniem katalogu danych produktu IBM MQ jest /var/mqm/
-  W systemach Windows położeniem katalogu danych produktu IBM MQ jest katalog danych wybrany podczas instalacji produktu IBM MQ .

QMNAME

Nazwa bieżącego menedżera kolejek.

MQ_SERVICE_NAME

Nazwa danej usługi.

PID MQ_SERVER_PID

Ten znacznik może być używany tylko przez argumenty **STOPARG** i **STOPCMD** .

W przypadku obiektów usług serwera ten znacznik jest zastępowany identyfikatorem procesu uruchomionym przez argumenty **STARTCMD** i **STARTARG** . W przeciwnym razie znacznik ten jest zastępowany wartością 0.

MQ_Q_MGR_DATA_PATH

Położenie katalogu danych menedżera kolejek.

MQ_Q_MGR_DATA_NAME

Przekształcona nazwa menedżera kolejek. Więcej informacji na temat transformacji nazw zawiera sekcja [Informacje o nazwach plików IBM MQ](#).

Aby użyć wymiennych wstawień, należy wstawić znacznik w obrębie znaków + do dowolnych łańcuchów **STARTCMD**, **STARTARG**, **STOPCMD**, **STOPARG**, **STDOUT** lub **STDERR**. Przykłady tego programu znajdują się w sekcji [“Przykłady korzystania z obiektów usług”](#) na stronie 188.

Przykłady korzystania z obiektów usług

Usługi w tej sekcji są zapisywane z separatorami ścieżek w stylu UNIX, z wyjątkiem sytuacji, w których określono inaczej.

Korzystanie z obiektu usługi serwera

W tym przykładzie przedstawiono sposób definiowania, używania i zmieniania obiektu usługi serwera w celu uruchomienia monitora wyzwalacza.

1. Obiekt usługi serwera jest zdefiniowany za pomocą komendy MQSC **DEFINE SERVICE** :

```
DEFINE SERVICE(S1) +  
CONTROL(QMGR) +  
SERVTYPE(SERVER) +  
STARTCMD('+MQ_INSTALL_PATH+bin/runmqtrm') +  
STARTARG('-m +QMNAME+ -q ACCOUNTS.INITIATION.QUEUE') +  
STOPCMD('+MQ_INSTALL_PATH+bin/amqsstop') +  
STOPARG('-m +QMNAME+ -p +MQ_SERVER_PID+')
```

gdzie:

+MQ_INSTALL_PATH+ jest tokenem reprezentującym katalog instalacyjny.

+QMNAME+ jest znacznikiem reprezentującym nazwę menedżera kolejek.

ACCOUNTS.INITIATION.QUEUE jest kolejką inicjującą.

Program amqsstop jest przykładowym programem dostarczonym z produktem IBM MQ, który żąda od menedżera kolejek przerwania wszystkich połączeń dla danego identyfikatora procesu. Program amqsstop generuje komendy PCF, dlatego serwer komend musi być uruchomiony.

+MQ_SERVER_PID+ jest znacznikiem reprezentującym identyfikator procesu przekazany do programu zatrzymania.

Listę wspólnych znaczników można znaleźć w sekcji [“Wymienne wstawki w definicjach usług”](#) na stronie 187.

2. Instancja obiektu usługi serwera zostanie wykonana po następnym uruchomieniu menedżera kolejek. Jednak natychmiast rozpoczniemy instancję obiektu usługi serwera za pomocą komendy MQSC **START SERVICE** :

```
START SERVICE(S1)
```

3. Status procesu usługi serwera jest wyświetlany za pomocą komendy MQSC **DISPLAY SVSTATUS** :

```
DISPLAY SVSTATUS(S1)
```

4. W tym przykładzie przedstawiono sposób zmiany obiektu usługi serwera i pobrania aktualizacji przez ręczne restartowanie procesu obsługi serwera. Obiekt usługi serwera jest zmieniany w taki sposób, że kolejka inicjująca jest określona jako JUPITER.INITIATION.QUEUE. Używana jest komenda **ALTER SERVICE MQSC**:

```
ALTER SERVICE(S1) +  
STARTARG('-m +QMNAME+ -q JUPITER.INITIATION.QUEUE')
```

Uwaga: Działająca usługa nie będzie odbierać żadnych aktualizacji definicji usługi, dopóki nie zostanie zrestartowana.

5. Proces obsługi serwera zostanie zrestartowany, aby zmiana została pobrana za pomocą komend MQSC **STOP SERVICE** i **START SERVICE** :

```
STOP SERVICE(S1)
```

Następnie:

```
START SERVICE(S1)
```

Proces usługi serwera jest restartowany i pobiera zmiany wprowadzone w produkcie [“4” na stronie 188](#).

Uwaga: Komenda MQSC **STOP SERVICE** może być używana tylko wtedy, gdy w definicji usługi jest określony argument **STOPCMD**.

Informacje pokrewne

[ALTER SERVICE](#)

[Definiuj usługę](#)

[WYŚWIETL STATUS SVSTATUS](#)

[Uruchom usługę](#)

[Zatrzymaj usługę](#)

Korzystanie z obiektu usługi komendy

W tym przykładzie przedstawiono sposób definiowania obiektu usługi komendy w celu uruchomienia programu, który zapisuje wpisy w dzienniku systemowym systemu operacyjnego, gdy menedżer kolejek jest uruchomiony lub zatrzymany.

1. Obiekt usługi komendy jest zdefiniowany za pomocą komendy MQSC **DEFINE SERVICE** :

```
DEFINE SERVICE(S2) +  
CONTROL(QMGR) +  
SERVTYPE(COMMAND) +  
STARTCMD('/usr/bin/logger') +  
STARTARG('Queue manager +QMNAME+ starting') +  
STOPCMD('/usr/bin/logger') +  
STOPARG('Queue manager +QMNAME+ stopping')
```

gdzie:

logger to komenda dostarczona przez system UNIX and Linux, która służy do zapisu w dzienniku systemowym.

+QMNAME+ jest znacznikiem reprezentującym nazwę menedżera kolejek.

Informacje pokrewne

[Definiuj usługę](#)

Korzystanie z obiektu usługi komendy, gdy menedżer kolejek kończy się tylko

W tym przykładzie przedstawiono sposób definiowania obiektu usługi komendy w celu uruchomienia programu, który zapisuje wpisy w dzienniku systemowym systemu operacyjnego, gdy menedżer kolejek jest zatrzymany.

1. Obiekt usługi komendy jest zdefiniowany za pomocą komendy MQSC **DEFINE SERVICE** :

```
DEFINE SERVICE(S3) +  
CONTROL(QMGR) +  
SERVTYPE(COMMAND) +  
STOPCMD('/usr/bin/logger') +  
STOPARG('Queue manager +QMNAME+ stopping')
```

gdzie:

Program `logger` jest przykładowym programem dostarczonym z produktem IBM MQ , który może zapisywać wpisy w dzienniku systemowym systemu operacyjnego.
+QMNAME+ jest znacznikiem reprezentującym nazwę menedżera kolejek.

Informacje pokrewne

[Definiuj usługę](#)

Więcej na temat przekazywania argumentów

W tym przykładzie przedstawiono sposób definiowania obiektu usługi serwera w celu uruchomienia programu o nazwie `runserv` , gdy uruchamiany jest menedżer kolejek.

Ten przykład jest zapisywany z separatorami ścieżek w stylu Windows .

Jednym z argumentów, które mają być przekazane do programu uruchamianego, jest łańcuch zawierający spację. Ten argument musi być przekazywany jako pojedynczy łańcuch. Aby to osiągnąć, należy użyć podwójnych cudzysłowów, jak pokazano w poniższej komendzie, w celu zdefiniowania obiektu usługi komendy:

1. Obiekt usługi serwera jest zdefiniowany za pomocą komendy MQSC **DEFINE SERVICE** :

```
DEFINE SERVICE(S1) SERVTYPE(SERVER) CONTROL(QMGR) +
STARTCMD('C:\Program Files\Tools\runserv.exe') +
STARTARG('-m +QMNAME+ -d "C:\Program Files\Tools\'') +
STDOUT('C:\Program Files\Tools\+MQ_SERVICE_NAME+.out')

DEFINE SERVICE(S4) +
CONTROL(QMGR) +
SERVTYPE(SERVER) +
STARTCMD('C:\Program Files\Tools\runserv.exe') +
STARTARG('-m +QMNAME+ -d "C:\Program Files\Tools\'') +
STDOUT('C:\Program Files\Tools\+MQ_SERVICE_NAME+.out')
```

gdzie:

+QMNAME+ jest znacznikiem reprezentującym nazwę menedżera kolejek.

"C:\Program Files\Tools\'" to łańcuch zawierający spację, która zostanie przekazana jako pojedynczy łańcuch.

Informacje pokrewne

[Definiuj usługę](#)

Automatyczne uruchamianie usługi

W tym przykładzie przedstawiono sposób definiowania obiektu usługi serwera, który może być używany do automatycznego uruchamiania monitora wyzwalacza podczas uruchamiania menedżera kolejek.

1. Obiekt usługi serwera jest zdefiniowany za pomocą komendy MQSC **DEFINE SERVICE** :

```
DEFINE SERVICE(TRIG_MON_START) +
CONTROL(QMGR) +
SERVTYPE(SERVER) +
STARTCMD('runmqtrm') +
STARTARG('-m +QMNAME+ -q +IQNAME+')
```

gdzie:

+QMNAME+ jest znacznikiem reprezentującym nazwę menedżera kolejek.

+IQNAME+ to zmienna środowiskowa zdefiniowana przez użytkownika w jednym z plików `service.env` , które reprezentują nazwę kolejki inicjuj.

Informacje pokrewne

[Definiuj usługę](#)

Zarządzanie obiektami w celu wyzwolenia

Produkt IBM MQ umożliwia automatyczne uruchamianie aplikacji, gdy spełnione są określone warunki w kolejce. Użytkownik może na przykład chcieć uruchomić aplikację, gdy liczba komunikatów w kolejce osiągnie określoną liczbę. Ta funkcja jest nazywana *wyzwalaniem*. Konieczne jest zdefiniowanie obiektów, które obsługują wyzwalenie.

Wyzwalanie opisane szczegółowo w sekcji [Uruchamianie aplikacji IBM MQ za pomocą wyzwalaczy](#).

Definiowanie kolejki aplikacji na potrzeby wyzwolenia

Kolejka aplikacji jest kolejką lokalną, która jest używana przez aplikacje do przesyłania komunikatów, przez interfejs MQI. Wyzwalanie wymaga zdefiniowania liczby atrybutów kolejki w kolejce aplikacji.

Samo wyzwolenie jest włączone przez atrybut **Trigger** (TRIGGER w komendach MQSC).

W tym przykładzie zdarzenie wyzwalające ma zostać wygenerowane, gdy w kolejce lokalnej MOTOR.INSURANCE.QUEUE, w następujący sposób:

```
DEFINE QLOCAL (MOTOR.INSURANCE.QUEUE) +  
PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS) +  
MAXMSGL (2000) +  
DEFPSIST (YES) +  
INITQ (MOTOR.INS.INIT.QUEUE) +  
TRIGGER +  
TRIGTYPE (DEPTH) +  
TRIGDPTH (100)+  
TRIGMPRI (5)
```

gdzie:

QLOCAL (MOTOR.INSURANCE.QUEUE)

Jest nazwą definiowanej kolejki aplikacji.

PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS)

Jest to nazwa definicji procesu definiująca aplikację, która ma zostać uruchomiona przez program monitora wyzwalacza.

MAXMSGL (2000)

Maksymalna długość komunikatów w kolejce.

DEFPSIST (YES)

Określa, że komunikaty w tej kolejce są domyślnie trwałe.

INITQ (MOTOR.INS.INIT.QUEUE)

Jest to nazwa kolejki inicjuj, w której menedżer kolejek ma umieścić komunikat wyzwalacza.

TRIGGER

Jest to wartość atrybutu wyzwalacza.

TRIGTYPE (DEPTH)

Określa, że zdarzenie wyzwalające jest generowane, gdy liczba komunikatów o wymaganym priorytecie (TRIGMPRI) osiągnie liczbę określoną w TRIGDPTH.

TRIGDPTH (100)

Liczba komunikatów wymaganych do wygenerowania zdarzenia wyzwalającego.

TRIGMPRI (5)

Jest to priorytet komunikatów, które mają być zliczane przez menedżera kolejek przy podejmowaniu decyzji o wygenerowaniu zdarzenia wyzwalającego. Zliczane są tylko komunikaty o priorytecie 5 lub wyższym.

Definiowanie kolejki inicjuj

Gdy wystąpi zdarzenie wyzwalające, menedżer kolejek umieszcza komunikat wyzwalacza w kolejce inicjuj, określonej w definicji kolejki aplikacji. Kolejki inicjuj nie mają żadnych ustawień specjalnych, ale można użyć następującej definicji kolejki lokalnej MOTOR.INS.INIT.QUEUE, aby uzyskać informacje na temat:

```
DEFINE QLOCAL (MOTOR.INS.INIT.QUEUE) +
GET (ENABLED) +
NOSHARE +
NOTRIGGER +
MAXMSGL (2000) +
MAXDEPTH (1000)
```

Definiowanie procesu

Aby utworzyć definicję procesu, należy użyć komendy DEFINE PROCESS. Definicja procesu definiuje aplikację, która ma być używana do przetwarzania komunikatów z kolejki aplikacji. Definicja kolejki aplikacji określa nazwę procesu, który ma zostać użyty, a tym samym wiąże kolejkę aplikacji z aplikacją, która ma być używana do przetwarzania jej komunikatów. Jest to realizowane za pomocą atrybutu PROCESS w kolejce aplikacji MOTOR.INSURANCE.QUEUE. Poniższa komenda MQSC definiuje wymagany proces, MOTOR.INSURANCE.QUOTE.PROCESS, zidentyfikowany w tym przykładzie:

```
DEFINE PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS) +
DESCR ('Insurance request message processing') +
APPLTYPE (UNIX) +
APPLICID ('/u/admin/test/IRMP01') +
USERDATA ('open, close, 235')
```

gdzie:

MOTOR.INSURANCE.QUOTE.PROCESS

Jest nazwą definicji procesu.

DESCR ('Insurance request message processing')

Opisuje program użytkowy, którego dotyczy ta definicja. Ten tekst jest wyświetlany, gdy używana jest komenda DISPLAY PROCESS. Może to pomóc w określeniu, co proces ma. Jeśli w łańcuchu używane są spacje, łańcuch należy ująć w pojedynczy cudzysłów.

APPLTYPE (UNIX)

Typ aplikacji, która ma zostać uruchomiona.

APPLICID ('/u/admin/test/IRMP01')

Jest nazwą pliku wykonywalnego aplikacji, który jest określony jako pełna nazwa pliku. W systemach Windows typową wartością APPLICID będzie c:\appl\test\irmp01.exe.

USERDATA ('open, close, 235')

Jest to dane zdefiniowane przez użytkownika, które mogą być używane przez aplikację.

Wyświetlanie atrybutów definicji procesu

Użyj komendy DISPLAY PROCESS, aby sprawdzić wyniki definicji. Na przykład:

```
DISPLAY PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS)

24 : DISPLAY PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS) ALL
AMQ8407: Display Process details.
DESCR ('Insurance request message processing')
APPLICID ('/u/admin/test/IRMP01')
USERDATA (open, close, 235)
PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS)
APPLTYPE (UNIX)
```

Można również użyć komendy MQSC ALTER PROCESS w celu zmiany istniejącej definicji procesu oraz komendy DELETE PROCESS w celu usunięcia definicji procesu.

Korzystanie z programu narzędziowego dmpmqmsg między dwoma systemami

Program narzędziowy **dmpmqmsg** (dawniej **qload**) jest włączany do produktu z produktu IBM MQ 8.0. Poprzednio program narzędziowy **qload** był dostępny jako SupportPac MO03.

Przegląd

Program narzędziowy **dmpmqmsg** umożliwia kopiowanie lub przenoszenie zawartości kolejki lub jej komunikatów do pliku. Ten plik może zostać zapisany w sposób wymagany i używany w późniejszym czasie w celu ponownego załadowania komunikatów z powrotem do kolejki.

Ważne: Plik ma określony format zrozumiały dla programu narzędziowego. Plik jest jednak czytelny dla użytkownika, dzięki czemu można go zaktualizować w edytorze przed przeladowaniem go. W przypadku edycji pliku nie wolno zmieniać jego formatu.

Możliwe zastosowania to:

- Zapisywanie komunikatów, które znajdują się w kolejce, do pliku. Być może w celu archiwizacji, a później ponownie przeladuj do kolejki.
- Przeladuj kolejkę za pomocą komunikatów, które zostały wcześniej zapisane w pliku.
- Usuwanie starych komunikatów z kolejki.
- 'Repowanie' komunikatów testowych z przechowywanego miejsca, nawet zachowując poprawny czas między komunikatami, jeśli jest to wymagane.



Ostrzeżenie: Pakiet SupportPac MO03 użył parametru **-l** w celu określenia powiązania lokalnego lub powiązania klienta. Parametr **-l** został zastąpiony przez parametr **-c**.

Produkt **-P** jest teraz używany na potrzeby informacji o stronie kodowej, a nie w produkcie **-c**.

Więcej informacji na temat komendy i dostępnych parametrów można znaleźć w sekcji [dmpmqmsg](#).

Przykład użycia programu narzędziowego dmpmqmsg w systemie Linux przy użyciu komputera z produktem Windows

Istnieje menedżer kolejek na komputerze z systemem Linux, który zawiera komunikaty w kolejce (*Q1*), które mają zostać przeniesione do innej kolejki (*Q2*) w tym samym menedżerze kolejek. Użytkownik chce zainicjować program narzędziowy **dmpmqmsg** na komputerze z produktem Windows.

Kolejka (*Q1*) zawiera cztery komunikaty, które zostały dodane przy użyciu przykładowego programu **amqspu**t (lokalnego menedżera kolejek) lub aplikacji **amqspu**tc (menedżer kolejek zdalnych).

Na komputerze z produktem Linux wyświetlane są następujące informacje:

```
display q1(Q1) CURDEPTH
      2 : display q1(Q1) CURDEPTH
AMQ8409: Display Queue details.
      QUEUE(Q1)
      TYPE(QLOCAL)
      CURDEPTH(4)
```

Ustaw zmienną środowiskową MQSERVER w taki sposób, aby wskazywała na menedżer kolejek w produkcie Linux. Na przykład:

```
set MQSERVER=SYSTEM.DEF.SVRCONN/TCP/veracruz.x.com(1414)
```

gdzie *veracruz* to nazwa komputera.

Uruchom program narzędziowy **dmpmqmsg**, aby odczytać z kolejki, *Q1*, a następnie zapisz dane wyjściowe w programie `c:\temp\mqload.txt`.

Połącz się jako klient zdalny z menedżerem kolejek *QM_VER*, działającym na hoście Linux i porcie określonym przez produkt *MQSERVER*. Połączenie jest osiągnięte jako klient zdalny przy użyciu atrybutu: `-c`.

```
dmpmqmsg -m QM_VER -i Q1 -f c:\temp\mqqload.txt -c
Read      - Files:    0  Messages:    4  Bytes:    22
Written - Files:    1  Messages:    4  Bytes:    22
```

Plik wyjściowy `c:\temp\mqqload.txt` zawiera tekst, używając formatu zrozumiałego dla programu narzędziowego **dmpmqmsg**.

Na komputerze z systemem Windows wydaj komendę **dmpmqmsg** (za pomocą opcji `-o` zamiast opcji `-i`) w celu załadowania kolejki (*Q2*) na komputerze Linux z pliku na komputerze z systemem Windows :

```
dmpmqmsg -m QM_VER -o Q2 -f c:\temp\mqqload.txt -c
Read      - Files:    1  Messages:    4  Bytes:    22
Written - Files:    0  Messages:    4  Bytes:    22
```

Na komputerze z produktem Linux należy pamiętać, że w kolejce znajdują się cztery komunikaty, które zostały odtworzone z pliku.

```
display ql(Q2) CURDEPTH
      6 : display ql(Q2) CURDEPTH
AMQ8409: Display Queue details.
      QUEUE(Q2)
      TYPE(QLOCAL)
      CURDEPTH(4)
```

Na komputerze z produktem Linux :

Usuń komunikaty z oryginalnej kolejki.

```
clear qllocal(Q1)
      4 : clear qllocal(Q1)
AMQ8022: IBM MQ queue cleared.
```

Upewnij się, że w oryginalnej kolejce nie ma więcej komunikatów:

```
display ql(Q1) CURDEPTH
      5 : display ql(Q1) CURDEPTH
AMQ8409: Display Queue details.
      QUEUE(Q1)
      TYPE(QLOCAL)
      CURDEPTH(0)
```

Opis komendy i jej parametrów można znaleźć w opisie komendy [dmpmqmsg](#).

Pojęcia pokrewne

“Przykłady użycia programu narzędziowego **dmpmqmsg**” na stronie 194

Proste sposoby korzystania z programu narzędziowego **dmpmqmsg** (dawniej **qload**). Ten program narzędziowy jest włączany do produktu z produktu IBM MQ 8.0.

Przykłady użycia programu narzędziowego **dmpmqmsg**

Proste sposoby korzystania z programu narzędziowego **dmpmqmsg** (dawniej **qload**). Ten program narzędziowy jest włączany do produktu z produktu IBM MQ 8.0.

Poprzednio program narzędziowy **qload** był dostępny jako SupportPac MO03.

Rozładowuje kolejkę do pliku

Użyj następujących opcji w wierszu komend, aby zapisać komunikaty, które znajdują się w kolejce, do pliku:

```
dmpmqmsg -m QM1 -i Q1 -f c:\myfile
```

Ta komenda pobiera kopię komunikatów z kolejki i zapisuje je w podanym pliku.

Rozładowuje kolejkę do serii plików

Kolejkę można rozładować do serii plików, używając znaku `insert` w nazwie pliku. W tym trybie każdy komunikat jest zapisywany do nowego pliku:

```
dmpmqmsg -m QM1 -i Q1 -f c:\myfile%n
```

Ta komenda powoduje wyładowanie kolejki do plików, `myfile1`, `myfile2`, `myfile3itd`.

Ładowanie kolejki z pliku

Aby przeladować kolejkę za pomocą komunikatów zapisanych w programie [“Rozładowuje kolejkę do pliku”](#) na stronie 194, w wierszu komend należy użyć następujących opcji:

```
dmpmqmsg -m QM1 -o Q1 -f c:\myfile%n
```

Ta komenda powoduje wyładowanie kolejki do plików, `myfile1`, `myfile2`, `myfile3itd`.

Ładowanie kolejki z serii plików

Kolejkę z serii plików można załadować przy użyciu znaku `insert` w nazwie pliku. W tym trybie każdy komunikat jest zapisywany do nowego pliku:

```
dmpmqmsg -m QM1 -o Q1 -f c:\myfile%n
```

Ta komenda ładuje kolejkę do plików, `myfile1`, `myfile2`, `myfile3itd`.

Kopiowanie komunikatów z jednej kolejki do innej

Zastąp parametr `file` w programie [“Rozładowuje kolejkę do pliku”](#) na stronie 194 inną nazwą kolejki i użyj następujących opcji:

```
dmpmqmsg -m QM1 -i Q1 -o Q2
```

Ta komenda umożliwia skopiowanie komunikatów z jednej kolejki do innej kolejki.

Kopiuj pierwsze 100 komunikatów z jednej kolejki do innej

Użyj komendy w poprzednim przykładzie i dodaj opcję `-r#100` :

```
dmpmqmsg -m QM1 -i Q1 -o Q2 -r#100
```

Przenieś komunikaty z jednej kolejki do innej

Odmiana w systemie [“Ładowanie kolejki z pliku”](#) na stronie 195. Należy zwrócić uwagę na rozróżnienie między używaniem `-i` (małymi literami), które tylko przegładną kolejkę, a `-I` (wielkimi literami), które niszczy proces pobierania z kolejki:

```
dmpmqmsg -m QM1 -I Q1 -o Q2
```

Przeniesienie wiadomości starsze niż jeden dzień z jednej kolejki do innej

W tym przykładzie przedstawiono sposób użycia wyboru wieku. Można wybrać komunikaty starsze niż, młodsze niż lub w przedziale wiekowych.

```
dmpmqmsg -m QM1 -I Q1 -o Q2 -T1440
```

Wyświetlanie wieku komunikatów znajdujących się obecnie w kolejce

W wierszu komend należy użyć następujących opcji:

```
dmpmqmsg -m QM1 -i Q1 -f stdout -dT
```

Praca z plikiem komunikatów

Po rozładowaniu komunikatu z kolejki, tak jak w programie [“Rozładowuje kolejkę do pliku”](#) na stronie 194, może być konieczne zmodyfikowanie pliku.

Można również zmienić format pliku, aby użyć jednej z opcji wyświetlania, które nie zostały podane w czasie, gdy wyładowano kolejkę.

Za pomocą programu narzędziowego **dmpmqmsg** można ponownie przetworzyć plik w wymaganym formacie, nawet po tym, jak odciążenie kolejki zostało wykonane. W wierszu komend należy użyć następujących opcji.

```
dmpmqmsg -f c:\oldfile -f c:\newfile -dA
```

Opis komendy i jej parametrów można znaleźć w opisie komendy [dmpmqmsg](#).

Administrowanie zdalnymi obiektami IBM MQ

W tej sekcji opisano sposób administrowania obiektami produktu IBM MQ w zdalnym menedżerze kolejek za pomocą komend MQSC oraz sposobu korzystania z obiektów kolejki zdalnej w celu sterowania miejscem docelowym komunikatów i komunikatów odpowiedzi.

Procedura

- Informacje na temat administrowania zdalnymi obiektami produktu IBM MQ można znaleźć w następujących podtematach:
 - [“Kanały i kolejkowanie zdalne”](#) na stronie 196
 - [“Zdalne administrowanie z lokalnego menedżera kolejek”](#) na stronie 198
 - [“Tworzenie lokalnej definicji kolejki zdalnej”](#) na stronie 204
 - [“Sprawdzanie, czy komendy asynchroniczne dla sieci rozproszonych zostały zakończone”](#) na stronie 206
 - [“Używanie definicji kolejek zdalnych jako aliasów”](#) na stronie 209
 - [“Konwersja danych”](#) na stronie 209

Kanały i kolejkowanie zdalne

Kanał można skonfigurować w taki sposób, aby udostępnił jednokierunkowe łącze komunikacyjne między lokalnym menedżerem kolejek i zdalnym menedżerem kolejek. Kanał może przenosić komunikaty przeznaczone dla dowolnej liczby kolejek w zdalnym menedżerze kolejek. Jeśli zdalny menedżer kolejek ma odpowiadać na komunikaty wysłane przez lokalny menedżer kolejek, można skonfigurować drugi kanał w celu wysyłania odpowiedzi ze zdalnego menedżera kolejek do lokalnego menedżera kolejek.

Menedżer kolejek komunikuje się z innym menedżerem kolejek, wysyłając komunikat i, jeśli jest to wymagane, odbierając odpowiedź. Odbiorcą menedżera kolejek może być:

- Na tym samym komputerze
- Na innym komputerze w tej samej lokalizacji (lub nawet po drugiej stronie świata)
- Uruchamianie na tej samej platformie co lokalny menedżer kolejek
- Uruchamianie na innej platformie obsługiwanej przez produkt IBM MQ

Te komunikaty mogą pochodzić z:

- Programy użytkowe napisane przez użytkownika, które przesyłają dane z jednego węzła do innego
- Aplikacje administracyjne napisane przez użytkownika, które korzystają z komend PCF lub MQAI
- IBM MQ Explorer.
- Wysyłanie menedżerów kolejek:
 - Instrumentacja komunikatów zdarzeń do innego menedżera kolejek
 - Komendy MQSC wydane z komendy **runmqsc** w trybie pośrednim (gdzie komendy są uruchamiane w innym menedżerze kolejek)

Zanim komunikat może zostać wysłany do zdalnego menedżera kolejek, lokalny menedżer kolejek wymaga mechanizmu wykrywania pojawiania się komunikatów i transportu ich składającego się z następujących elementów:

- Co najmniej jeden kanał
- Kolejka transmisji
- Inicjator kanału

W przypadku menedżera kolejek zdalnych w celu odebrania komunikatu wymagany jest obiekt nastuchiwania.

Kanał jest jednokierunkową łączem komunikacyjnym między dwoma menedżerami kolejek i może przenosić komunikaty przeznaczone dla dowolnej liczby kolejek w zdalnym menedżerze kolejek.

Każdy koniec kanału ma osobną definicję. Na przykład, jeśli jeden koniec jest nadawcą lub serwerem, drugi koniec musi być odbiornikiem lub requesterem. Prosty kanał składa się z *definicji kanału nadawczego* na końcu lokalnego menedżera kolejek i *definicji kanału odbiorczego* w zdalnym menedżerze kolejek. Te dwie definicje muszą mieć taką samą nazwę i razem tworzą jeden kanał komunikatów.

Jeśli zdalny menedżer kolejek ma odpowiadać na komunikaty wysłane przez lokalny menedżer kolejek, należy skonfigurować drugi kanał w celu wysyłania odpowiedzi z powrotem do lokalnego menedżera kolejek.

Aby zdefiniować kanały, należy użyć komendy MQSC **DEFINE CHANNEL**. W tej sekcji przykłady odnoszące się do kanałów używają domyślnych atrybutów kanału, o ile nie określono inaczej.

Na każdym końcu kanału znajduje się agent kanału komunikatów (MCA), kontrolujący wysyłanie i odbieranie komunikatów. Agent MCA pobiera komunikaty z kolejki transmisji i umieszcza je na łączu komunikacyjnym między menedżerami kolejek.

Kolejka transmisji jest wyspecjalizowaną kolejką lokalną, która tymczasowo przechowuje komunikaty, zanim agent MCA pobiera je i wysyła do zdalnego menedżera kolejek. Nazwę kolejki transmisji należy określić w *definicji kolejki zdalnej*.

Można zezwolić agentowi MCA na przesyłanie komunikatów przy użyciu wielu wątków. Ten proces jest znany jako *pipelining*. Pipelining umożliwia agentowi MCA bardziej efektywne przesyłanie komunikatów, poprawiając wydajność kanałów. Sekcja Atrybuty kanałów zawiera szczegółowe informacje na temat sposobu konfigurowania kanału w celu korzystania z pipekresów.

W sekcji “Przygotowywanie kanałów i kolejek transmisji do zdalnego administrowania” na stronie 199 opisano sposób korzystania z tych definicji w celu skonfigurowania zdalnego administrowania.

Więcej informacji na temat konfigurowania rozproszonego kolejkowania w ogólności znajduje się w sekcji Rozproszone komponenty kolejkowania.

Informacje pokrewne

[runmqsc \(uruchamianie komend MQSC\)](#)

Zdalne administrowanie przy użyciu klastrów

W sieci IBM MQ korzystającej z rozproszonego kolejkowania każdy menedżer kolejek jest niezależny. Jeśli jeden menedżer kolejek musi wysłać komunikaty do innego menedżera kolejek, musi zdefiniować kolejkę transmisji, kanał do zdalnego menedżera kolejek oraz definicję kolejki zdalnej dla każdej kolejki, do której ma być wysyłane komunikaty.

Klaster jest grupą menedżerów kolejek ustawionymi w taki sposób, że menedżery kolejek mogą komunikować się bezpośrednio z jedną siecią za pośrednictwem jednej sieci bez złożonej kolejki transmisji, kanału i definicji kolejek. Klastry mogą być łatwo skonfigurowane i zwykle zawierają menedżery kolejek, które są logicznie powiązane w pewien sposób i muszą współużytkować dane lub aplikacje. Nawet najmniejszy klaster redukuje koszty administrowania systemem.

Utworzenie sieci menedżerów kolejek w klastrze obejmuje mniejszą liczbę definicji niż utworzenie tradycyjnego rozproszonego środowiska kolejkowania. Dzięki mniejszej liczbie definicji można szybciej i łatwo skonfigurować lub zmienić sieć, a także zmniejszyć ryzyko popełnienia błędów w definicjach.






Aby skonfigurować klaster, potrzebny jest jeden nadawca klastra (CLUSDR) i jeden odbiornik klastra (CLUSRCVR) dla każdego menedżera kolejek. Nie są potrzebne żadne definicje kolejek transmisji ani definicje kolejek zdalnych. Zasady zdalnego administrowania są takie same, gdy używane są w klastrze, ale same definicje są znacznie uproszczone.

Zdalne administrowanie z lokalnego menedżera kolejek

Za pomocą komend MQSC i PCF można administrować zdalnym menedżerem kolejek z lokalnego menedżera kolejek.

Przygotowanie kolejek i kanałów jest zasadniczo takie samo w przypadku komend MQSC i PCF. W tej sekcji przykłady ilustrują komendy MQSC, ponieważ są one łatwiejsze do zrozumienia. Więcej informacji na temat pisania programów administracyjnych za pomocą komend PCF zawiera sekcja [“Korzystanie z formatów komend programowalnych IBM MQ”](#) na stronie 23.

Komendy MQSC wysyłane są do zdalnego menedżera kolejek interaktywnie lub z pliku tekstowego zawierającego komendy. Zdalny menedżer kolejek może znajdować się na tym samym komputerze lub na innym komputerze, zazwyczaj na innym komputerze. Menedżery kolejek można administrować zdalnie w innych środowiskach IBM MQ, w tym:

-  UNIX
-  Linux
-  Windows
-  IBM i
-  z/OS

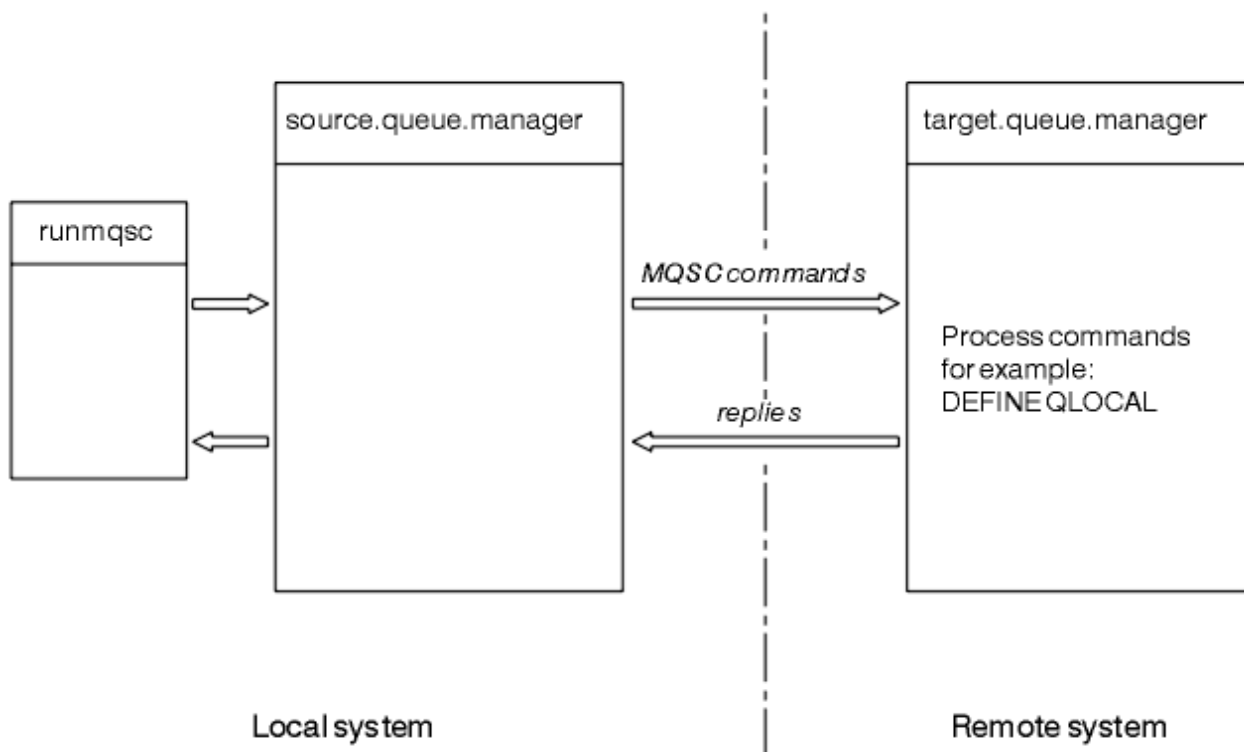
Aby zaimplementować zdalne administrowanie, należy utworzyć konkretne obiekty. Jeśli nie są wymagane specjalne wymagania, wartości domyślne, na przykład, dla maksymalnej długości komunikatu, są wystarczające.

Przygotowywanie menedżerów kolejek do zdalnego administrowania

Informacje na temat używania komend MQSC do przygotowywania menedżerów kolejek do zdalnego administrowania.

W programie [Rysunek 18 na stronie 199](#) jest wyświetlana konfiguracja menedżerów kolejek i kanałów, które są potrzebne do zdalnego administrowania za pomocą komendy **runmqsc**. Obiekt `source.queue.manager` jest źródłowym menedżerem kolejek, z którego można wydawać

komendy MQSC i do których zwracane są wyniki tych komend (komunikaty operatora). Obiekt `target.queue.manager` to nazwa docelowego menedżera kolejek, który przetwarza komendy i generuje dowolne komunikaty operatora.



Rysunek 18. Zdalne administrowanie za pomocą komend MQSC

W obu systemach, jeśli nie zostało to jeszcze wykonane:

- Utwórz menedżer kolejek i obiekty domyślne przy użyciu komendy `crtmqm`. Więcej informacji na ten temat zawiera sekcja [crtmqm](#).
- Uruchom menedżera kolejek za pomocą komendy `strmqm`. Więcej informacji na ten temat zawiera sekcja [strmqm](#).

W docelowym menedżerze kolejek:

- Kolejka komend `SYSTEM.ADMIN.COMMAND.QUEUE`, musi być obecna. Ta kolejka jest tworzona domyślnie, gdy tworzony jest menedżer kolejek.

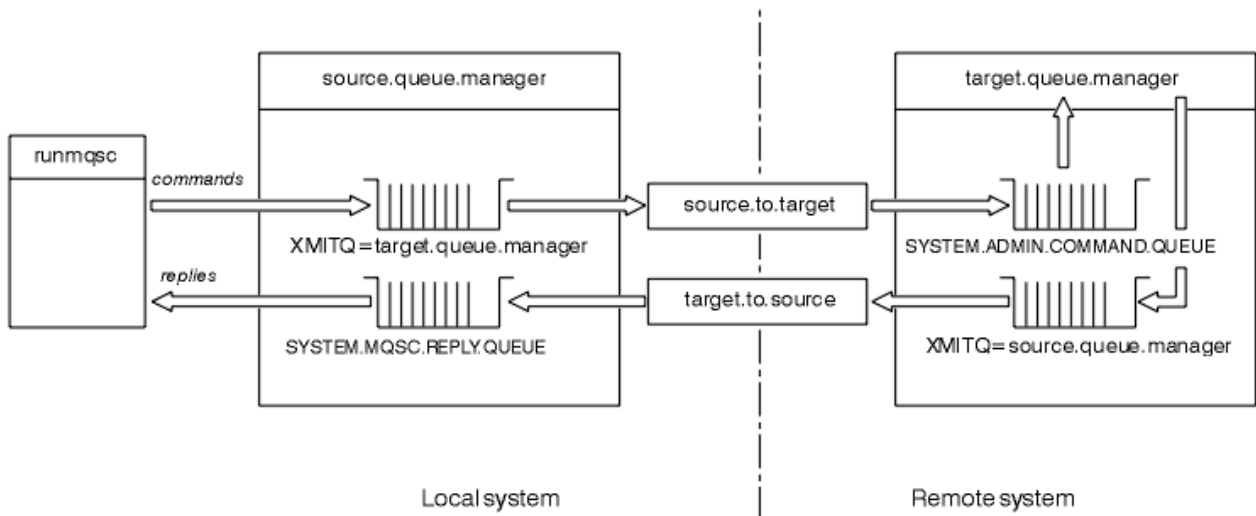
Komendy te należy uruchamiać lokalnie lub za pośrednictwem narzędzia sieciowego, takiego jak Telnet.

Przygotowywanie kanałów i kolejek transmisji do zdalnego administrowania

Jak używać komend MQSC do przygotowywania kanałów i kolejek transmisji dla zdalnego administrowania.

Aby zdalnie uruchamiać komendy MQSC, należy skonfigurować dwa kanały, po jednym dla każdego kierunku i powiązane z nimi kolejki transmisji. W tym przykładzie założono, że jako typ transportu używany jest protokół TCP/IP oraz że jest on używany jako adres TCP/IP.

Kanał `source.to.target` jest przeznaczony do wysyłania komend MQSC z źródłowego menedżera kolejek do docelowego menedżera kolejek. Jego nadawca znajduje się w `source.queue.manager`, a jego odbiornik znajduje się w `target.queue.manager`. Kanał `target.to.source` służy do zwracania danych wyjściowych komend i wszystkich komunikatów operatora, które są generowane dla źródłowego menedżera kolejek. Należy również zdefiniować kolejkę transmisji dla każdego kanału. Ta kolejka jest kolejką lokalną, której nadano nazwę odbierającego menedżera kolejek. Nazwa kolejki XMITQ musi być zgodna z nazwą zdalnego menedżera kolejek w celu zdalnego administrowania do pracy, chyba że używany jest alias menedżera kolejek. [Rysunek 19 na stronie 200](#) podsumowuje tę konfigurację.



Rysunek 19. Konfigurowanie kanałów i kolejek do zdalnego administrowania

Więcej informacji na temat konfigurowania kanałów zawiera sekcja [Konfigurowanie kolejki rozproszonego](#).

Definiowanie kanałów, programów nastuchujących i kolejek transmisji

W źródłowym menedżerze kolejek (`source.queue.manager`) wykonaj następujące komendy MQSC, aby zdefiniować kanały, nastuchiwanie i kolejkę transmisji:

1. Zdefiniuj kanał nadawczy w źródłowym menedżerze kolejek:

```
DEFINE CHANNEL ('source.to.target') +
CHLTYPE(SDR) +
CONNAME (RHX5498) +
XMITQ ('target.queue.manager') +
TRPTYPE(TCP)
```

2. Zdefiniuj kanał odbiorczy w źródłowym menedżerze kolejek:

```
DEFINE CHANNEL ('target.to.source') +
CHLTYPE(RCVR) +
TRPTYPE(TCP)
```

3. Zdefiniuj program nastuchujący w źródłowym menedżerze kolejek:

```
DEFINE LISTENER ('source.queue.manager') +
TRPTYPE (TCP)
```

4. Zdefiniuj kolejkę transmisji w źródłowym menedżerze kolejek:

```
DEFINE QLOCAL ('target.queue.manager') +
USAGE (XMITQ)
```

Wykonaj następujące komendy w docelowym menedżerze kolejek (`target.queue.manager`), aby utworzyć kanały, nastuchiwanie i kolejkę transmisji:

1. Zdefiniuj kanał nadawczy w docelowym menedżerze kolejek:

```
DEFINE CHANNEL ('target.to.source') +
CHLTYPE(SDR) +
CONNAME (RHX7721) +
XMITQ ('source.queue.manager') +
TRPTYPE(TCP)
```


2. Zdefiniuj kanał odbiorczy w docelowym menedżerze kolejek:

```
DEFINE CHANNEL ('source.to.target') +  
CHLTYPE(RCVR) +  
TRPTYPE(TCP)
```

3. Zdefiniuj program nasłuchujący w docelowym menedżerze kolejek:

```
DEFINE LISTENER ('target.queue.manager') +  
TRPTYPE (TCP)
```

4. Zdefiniuj kolejkę transmisji w docelowym menedżerze kolejek:

```
DEFINE QLOCAL ('source.queue.manager') +  
USAGE (XMITQ)
```

Uwaga: Nazwy połączeń TCP/IP określone dla atrybutu CONNAME w definicjach kanału nadawczego służą tylko do ilustracji. Jest to nazwa sieciowa komputera na *innym* końcu połączenia. Należy użyć wartości odpowiednich dla danej sieci.

Uruchamianie programów nasłuchujących i kanałów

W jaki sposób można używać komend MQSC do uruchamiania programów nasłuchujących i kanałów.

Uruchom oba programy nasłuchujące, korzystając z następujących komend MQSC:

1. Uruchom program nasłuchujący w źródłowym menedżerze kolejek, `source.queue.manager`, wydając następującą komendę MQSC:

```
START LISTENER ('source.queue.manager')
```

2. Uruchom program nasłuchujący w docelowym menedżerze kolejek, `target.queue.manager`, wydając następującą komendę MQSC:

```
START LISTENER ('target.queue.manager')
```

Uruchom oba kanały wysyłające za pomocą następujących komend MQSC:

1. Uruchom kanał nadawczy w źródłowym menedżerze kolejek, `source.queue.manager`, wydając następującą komendę MQSC:

```
START CHANNEL ('source.to.target')
```

2. Uruchom kanał nadawczy w docelowym menedżerze kolejek, `target.queue.manager`, wydając następującą komendę MQSC:

```
START CHANNEL ('target.to.source')
```

Automatyczna definicja kanałów

Można włączyć automatyczną definicję definicji połączenia odbiorcy i serwera, aktualizując obiekt menedżera kolejek za pomocą komendy MQSC ALTER QMGR (lub komendy PCF Zmiana menedżera kolejek).

Jeśli program IBM MQ odbierze żądanie przyłączenia przychodzącego i nie może znaleźć odpowiedniego kanału odbiorczego lub kanału połączenia z serwerem, automatycznie utworzy kanał. Definicje automatyczne są oparte na dwóch definicjach domyślnych dostarczanych z produktem IBM MQ: SYSTEM.AUTO.RECEIVER i SYSTEM.AUTO.SVRCONN.

Więcej informacji na temat automatycznego tworzenia definicji kanałów znajduje się w sekcji [Przygotowywanie kanałów](#). Więcej informacji na temat automatycznego definiowania kanałów dla klastrów zawiera sekcja [Praca z automatycznie zdefiniowanymi kanałami](#).

Zarządzanie serwerem komend w celu zdalnego administrowania

Informacje na temat uruchamiania, zatrzymywania i wyświetlania statusu serwera komend. Serwer komend jest obowiązkowy dla wszystkich funkcji administracyjnych związanych z komendami PCF, MQAI, a także dla zdalnego administrowania.

Z każdym menedżerem kolejek może być powiązany serwer komend. Serwer komend przetwarza wszystkie komendy przychodzące ze zdalnych menedżerów kolejek lub komendy PCF z aplikacji. Przedstawia on komendy do menedżera kolejek w celu przetworzenia i zwraca kod zakończenia lub komunikat operatora w zależności od pochodzenia komendy.

Uwaga: W przypadku zdalnego administrowania upewnij się, że docelowy menedżer kolejek jest uruchomiony. W przeciwnym razie komunikaty zawierające komendy nie mogą opuścić menedżera kolejek, z którego są one wydawane. Zamiast tego komunikaty te są kolejgowane w lokalnej kolejce transmisji, która służy do obsługi menedżera kolejek zdalnych. Należy unikać tej sytuacji.

Istnieją oddzielne komendy sterujące służące do uruchamiania i zatrzymywania serwera komend. Jeśli serwer komend jest uruchomiony, użytkownicy produktu IBM MQ for Windows lub IBM MQ for Linux (platformy x86 i x86-64) mogą wykonywać operacje opisane w poniższych sekcjach za pomocą Eksploratora IBM MQ. Więcej informacji na ten temat zawiera sekcja [“Administrowanie przy użyciu IBM MQ Explorer”](#) na stronie 134.

Uruchamianie serwera komend

W zależności od wartości atrybutu menedżera kolejek *SCMDSERV*, serwer komend jest uruchamiany automatycznie podczas uruchamiania menedżera kolejek lub musi być uruchamiany ręcznie. Wartość atrybutu menedżera kolejek można zmienić za pomocą komendy *MQSC ALTER QMGR*, określając parametr *SCMDSERV*. Domyślnie serwer komend jest uruchamiany automatycznie.

Jeśli parametr *SCMDSERV* jest ustawiony na wartość *MANUAL*, uruchom serwer komend przy użyciu komendy:

```
stimqcsv saturn.queue.manager
```

gdzie *saturn.queue.manager* jest menedżerem kolejek, dla którego uruchamiany jest serwer komend.

Wyświetlanie statusu serwera komend

W przypadku zdalnego administrowania należy upewnić się, że serwer komend w docelowym menedżerze kolejek jest uruchomiony. Jeśli nie jest uruchomiony, komendy zdalne nie mogą być przetwarzane. Wszystkie komunikaty zawierające komendy są kolejgowane w kolejce komend menedżera kolejek docelowych.

Aby wyświetlić status serwera komend dla menedżera kolejek, wprowadź następującą komendę *MQSC*:

```
DISPLAY QMSTATUS CMDSERV
```

Zatrzymywanie serwera komend

Aby zakończyć działanie serwera komend uruchomionego w poprzednim przykładzie, należy użyć następującej komendy:

```
endmqcsv saturn.queue.manager
```

Serwer komend można zatrzymać na dwa sposoby:

- W przypadku zatrzymania kontrolowanego należy użyć komendy `endmqcsv` z opcją `-c`, która jest wartością domyślną.
- Aby zatrzymać natychmiastowe zatrzymanie, użyj komendy `endmqcsv` z opcją `-i`.

Uwaga: Zatrzymanie menedżera kolejek powoduje również zakończenie powiązanego z nim serwera komend.

Wydawanie komend MQSC w zdalnym menedżerze kolejek

W celu uruchomienia komend MQSC w zdalnym menedżerze kolejek można użyć konkretnego formularza komendy `runmqsc`.

Serwer komend musi być uruchomiony w docelowym menedżerze kolejek, jeśli ma być zdalnie przetwarzający komendy MQSC. (Nie jest to konieczne w menedżerze kolejek źródłowych). Więcej informacji na temat uruchamiania serwera komend w menedżerze kolejek zawiera sekcja [“Zarządzanie serwerem komend w celu zdalnego administrowania”](#) na stronie 202.

W źródłowym menedżerze kolejek można następnie interaktywnie uruchamiać komendy MQSC w trybie pośrednim, wpisując:

```
runmqsc -w 30 -m source.queue.manager target.queue.manager
```

Ten formularz komendy `runmqsc` z flagą `-w` powoduje uruchomienie komend MQSC w trybie pośrednim, w których komendy są umieszczane (w zmodyfikowanym formularzu) w kolejce wejściowej serwera komend i wykonywane w kolejności.

W przypadku wpisania komendy MQSC jest ona przekierowana do zdalnego menedżera kolejek, w tym przypadku, `target.queue.manager`. Limit czasu jest ustawiony na 30 sekund. Jeśli odpowiedź nie zostanie odebrana w ciągu 30 sekund, na lokalnym (źródłowym) menedżerze kolejek zostanie wygenerowany następujący komunikat:

```
AMQ8416: MQSC timed out waiting for a response from the command server.
```

Po zatrzymaniu wydawania komend MQSC lokalny menedżer kolejek wyświetla wszystkie odpowiedzi z przekroczonym limitem czasu, które przybyły i odrzuciły wszelkie dalsze odpowiedzi.

Źródłowy menedżer kolejek jest domyślnie ustawiony na domyślny menedżer kolejek lokalnych. Jeśli w komendzie `runmqsc` zostanie podana opcja `-m LocalQmgrName`, można kierować komendy, które mają zostać wydane za pomocą dowolnego lokalnego menedżera kolejek.

W trybie pośrednim można również uruchomić plik komend MQSC w zdalnym menedżerze kolejek. Na przykład:

```
runmqsc -w 60 target.queue.manager < mycomds.in > report.out
```

gdzie `mycomds.in` jest plikiem zawierającym komendy MQSC, a `report.out` jest plikiem raportu.

Sugerowana metoda zdalnego wydawania komend

W przypadku wydawania komend w zdalnym menedżerze kolejek należy rozważyć zastosowanie następującego podejścia:

1. Umieść komendy MQSC, które mają być uruchamiane w systemie zdalnym w pliku komend.
2. Sprawdź, czy komendy MQSC są lokalnie, podając flagę `-v` w komendzie `runmqsc`.
Komendy `runmqsc` nie można używać do weryfikowania komend MQSC w innym menedżerze kolejek.
3. Sprawdź, czy plik komend działa lokalnie bez błędów.
4. Uruchom plik komend w systemie zdalnym.

W przypadku problemów z używaniem komend MQSC

Jeśli masz problemy z uruchamianiem zdalnie komendy MQSC, upewnij się, że masz:

- Uruchomiono serwer komend w docelowym menedżerze kolejek.
- Zdefiniowano poprawną kolejkę transmisji.
- Zdefiniowano dwa końce kanałów komunikatów dla obu tych kanałów:
 - Kanał, wzdłuż którego wysyłane są komendy.
 - Kanał, w którym mają zostać zwrócone odpowiedzi.
- W definicji kanału podano poprawną nazwę połączenia (CONNNAME).
- Uruchomiono programy nasłuchujące przed rozpoczęciem kanałów komunikatów.
- Sprawdź, czy interwał odłączania nie utracił ważności, na przykład, jeśli kanał został uruchomiony, ale po pewnym czasie został zamknięty. Jest to szczególnie ważne w przypadku ręcznego uruchomienia kanałów.
- Wysłane żądania z menedżera kolejek źródłowych, które nie mają sensu w docelowym menedżerze kolejek (na przykład żądania zawierające parametry, które nie są obsługiwane w zdalnym menedżerze kolejek).

Patrz także [“Rozwiązywanie problemów za pomocą komend MQSC” na stronie 20](#).

Praca z menedżerami kolejek w systemie z/OS

Komendy MQSC można wydawać do menedżera kolejek produktu z/OS z poziomu menedżera kolejek na platformach opisanych w tym podręczniku. Jednak aby to zrobić, należy zmodyfikować komendę `runmqsc` i definicje kanałów u nadawcy.

W szczególności należy dodać opcję `-x` do komendy `runmqsc` w węzle źródłowym, aby określić, że docelowy menedżer kolejek jest uruchomiony w systemie z/OS:

```
runmqsc -w 30 -x target.queue.manager
```

Tworzenie lokalnej definicji kolejki zdalnej

Lokalna definicja kolejki zdalnej jest definicją w lokalnym menedżerze kolejek, która odwołuje się do kolejki w zdalnym menedżerze kolejek.

Nie trzeba definiować kolejki zdalnej z pozycji lokalnej, ale jej zaletą jest to, że aplikacje mogą odwoływać się do zdalnej kolejki za pomocą jej lokalnie zdefiniowanej nazwy, zamiast określać nazwę kwalifikowaną za pomocą identyfikatora menedżera kolejek, w którym znajduje się zdalna kolejka.

Informacje o tym, jak działają lokalne definicje kolejek zdalnych

Aplikacja łączy się z lokalnym menedżerem kolejek, a następnie wywołuje wywołanie `MQOPEN`.

W wywołaniu otwartym podana nazwa kolejki jest nazwą zdalnej definicji kolejki w lokalnym menedżerze kolejek. Definicja kolejki zdalnej dostarcza nazwy kolejki docelowej, docelowego menedżera kolejek i opcjonalnie kolejki transmisji. Aby umieścić komunikat w kolejce zdalnej, aplikacja wysyła wywołanie `MQPUT`, określając uchwyt zwrócony z wywołania `MQOPEN`. Menedżer kolejek używa nazwy kolejki zdalnej i nazwy zdalnego menedżera kolejek w nagłówku transmisji na początku komunikatu. Informacje te są używane do kierowania komunikatu do jego poprawnego miejsca docelowego w sieci.

Administrator może sterować miejscem docelowym komunikatu, modyfikując definicję kolejki zdalnej.

W poniższym przykładzie pokazano, w jaki sposób aplikacja umieszcza komunikat w kolejce, której właścicielem jest zdalny menedżer kolejek. Aplikacja łączy się z menedżerem kolejek, na przykład `sa.tu.rn.queue.manager`. Właścicielem kolejki docelowej jest inny menedżer kolejek.

W wywołaniu funkcji `MQOPEN` aplikacja określa następujące pola:

Wartość pola	Opis
<i>ObjectName</i> CYAN.REMOTE.QUEUE	Określa lokalną nazwę obiektu kolejki zdalnej. Definiuje kolejkę docelową i docelowy menedżer kolejek.
<i>ObjectType</i> (Kolejka)	Identyfikuje ten obiekt jako kolejkę.
<i>ObjectQmgrName</i> Puste lub saturn.queue.manager	To pole jest opcjonalne. Jeśli ta wartość jest pusta, przyjmowana jest nazwa lokalnego menedżera kolejek. (Jest to menedżer kolejek, w którym istnieje definicja kolejki zdalnej).

Po tym czasie aplikacja wysyła wywołanie MQPUT w celu umieszczenia komunikatu w tej kolejce.

W lokalnym menedżerze kolejek można utworzyć lokalną definicję kolejki zdalnej przy użyciu następujących komend MQSC:

```
DEFINE QREMOTE (CYAN.REMOTE.QUEUE) +
DESCR ('Queue for auto insurance requests from the branches') +
RNAME (AUTOMOBILE.INSURANCE.QUOTE.QUEUE) +
RQMNAME (jupiter.queue.manager) +
XMITQ (INQUOTE.XMIT.QUEUE)
```

gdzie:

QREMOTE (CYAN.REMOTE.QUEUE)

Określa lokalną nazwę obiektu kolejki zdalnej. Jest to nazwa, którą aplikacje połączone z tym menedżerem kolejek muszą określić w wywołaniu komendy MQOPEN , aby otworzyć kolejkę AUTOMOBILE.INSURANCE.QUOTE.QUEUE w zdalnym menedżerze kolejek jupiter.queue.manager.

DESCR ('Queue for auto insurance requests from the branches')

Udostępnia dodatkowy tekst opisujący korzystanie z kolejki.

RNAME (AUTOMOBILE.INSURANCE.QUOTE.QUEUE)

Określa nazwę kolejki docelowej w zdalnym menedżerze kolejek. Jest to rzeczywista kolejka docelowa dla komunikatów wysyłanych przez aplikacje, które określają nazwę kolejki CYAN.REMOTE.QUEUE. Kolejka AUTOMOBILE.INSURANCE.QUOTE.QUEUE musi być zdefiniowana jako kolejka lokalna w zdalnym menedżerze kolejek.

RQMNAME (jupiter.queue.manager)

Określa nazwę zdalnego menedżera kolejek, do którego należy kolejka docelowa AUTOMOBILE.INSURANCE.QUOTE.QUEUE.

XMITQ (INQUOTE.XMIT.QUEUE)

Określa nazwę kolejki transmisji. Jest to opcjonalne. Jeśli nazwa kolejki transmisji nie jest określona, używana jest kolejka o takiej samej nazwie, jak nazwa zdalnego menedżera kolejek.

W obu przypadkach odpowiednia kolejka transmisji musi być zdefiniowana jako kolejka lokalna z atrybutem **Usage** , który określa, że jest to kolejka transmisji (USAGE (XMITQ) w komendach MQSC).

Alternatywny sposób umieszczania komunikatów w kolejce zdalnej

Użycie lokalnej definicji kolejki zdalnej nie jest jedynym sposobem umieszczania komunikatów w kolejce zdalnej. Aplikacje mogą określać pełną nazwę kolejki, w tym nazwę zdalnego menedżera kolejek, jako część wywołania MQOPEN . W takim przypadku lokalna definicja kolejki zdalnej nie jest wymagana. Oznacza to jednak, że aplikacje muszą znać nazwę zdalnego menedżera kolejek w czasie wykonywania albo muszą mieć dostęp do niej.

Korzystanie z innych komend z kolejkami zdalnymi

Za pomocą komend MQSC można wyświetlać lub zmieniać atrybuty obiektu kolejki zdalnej lub można usunąć obiekt kolejki zdalnej. Na przykład:

- Aby wyświetlić atrybuty kolejki zdalnej:

```
DISPLAY QUEUE (CYAN.REMOTE.QUEUE)
```

- Aby zmienić kolejkę zdalną w celu włączenia operacji put. Nie ma to wpływu na kolejkę docelową. Tylko aplikacje, które określają tę kolejkę zdalną:

```
ALTER QREMOTE (CYAN.REMOTE.QUEUE) PUT(ENABLED)
```

- Aby usunąć tę kolejkę zdalną. Nie ma to wpływu na kolejkę docelową, a tylko jej definicję lokalną:

```
DELETE QREMOTE (CYAN.REMOTE.QUEUE)
```

Uwaga: Usunięcie kolejki zdalnej powoduje usunięcie tylko lokalnej reprezentacji kolejki zdalnej. Nie należy usuwać samej kolejki zdalnej ani żadnych komunikatów na jej temat.

Definiowanie kolejki transmisji

Kolejka transmisji jest kolejką lokalną, która jest używana, gdy menedżer kolejek przekazuje komunikaty do zdalnego menedżera kolejek za pośrednictwem kanału komunikatów.

Kanał udostępnia jednokierunkowy odsyłacz do zdalnego menedżera kolejek. Komunikaty są kolejkowane w kolejce transmisji do momentu, gdy kanał będzie mógł je zaakceptować. Podczas definiowania kanału należy określić nazwę kolejki transmisji na końcu wysyłającego kanału komunikatów.

Atrybut USAGE komendy MQSC definiuje, czy kolejka jest kolejką transmisji, czy normalną kolejką.

Domyślne kolejki transmisji

Gdy menedżer kolejek wysyła komunikaty do zdalnego menedżera kolejek, identyfikuje kolejkę transmisji przy użyciu następującej sekwencji:

1. Kolejka transmisji określona w atrybucie XMITQ lokalnej definicji kolejki zdalnej.
2. Kolejka transmisji o tej samej nazwie, co docelowy menedżer kolejek. (Ta wartość jest wartością domyślną dla kolejki XMITQ lokalnej definicji kolejki zdalnej).
3. Kolejka transmisji określona w atrybucie DEFXMITQ lokalnego menedżera kolejek.

Na przykład następująca komenda MQSC tworzy domyślną kolejkę transmisji w systemie `source.queue.manager` dla komunikatów wybieranych do produktu `target.queue.manager`:

```
DEFINE QLOCAL ('target.queue.manager') +  
DESCR ('Default transmission queue for target qm') +  
USAGE (XMITQ)
```

Aplikacje mogą umieszczać komunikaty bezpośrednio w kolejce transmisji lub pośrednio przez definicję kolejki zdalnej. Patrz także [“Tworzenie lokalnej definicji kolejki zdalnej”](#) na stronie 204.

Sprawdzanie, czy komendy asynchroniczne dla sieci rozproszonych zostały zakończone

Wiele komend jest asynchronicznych w przypadku użycia w sieci rozproszonej. W zależności od komendy i stanu sieci po jego wydaniu, może zająć dużo czasu. Menedżer kolejek nie wydaje komunikatu po zakończeniu, dlatego konieczne jest inne sposoby sprawdzania, czy komenda została zakończona.

O tym zadaniu

Niemal każda zmiana konfiguracji, którą należy wprowadzić do klastra, może zakończyć się asynchronicznie. Jest to spowodowane wewnętrznym administrowaniem i aktualizowaniem cykli, które działają w klastrach. W przypadku hierarchii publikowania/subskrypcji każda zmiana konfiguracji, która ma wpływ na subskrypcje, może zakończyć się asynchronicznie. Nie zawsze jest to oczywiste z nazwy komendy.

Następujące komendy MQSC mogą być wykonywane asynchronicznie. Każda z tych komend ma odpowiednik PCF, a większość z nich jest również dostępna w produkcie IBM MQ Explorer . W przypadku uruchamiania w małej sieci bez obciążenia, komendy te są zwykle wykonywane w ciągu kilku sekund. Nie jest to jednak w przypadku większych i bardziej budzonych sieci. Ponadto komenda **REFRESH CLUSTER** może zająć dużo więcej czasu, szczególnie wtedy, gdy jest ona emitowana w wielu menedżerach kolejek jednocześnie.

Aby upewnić się, że te komendy zostały zakończone, należy sprawdzić, czy oczekiwane obiekty istnieją w menedżerach kolejek zdalnych.

Procedura

- ALTER QMGR

W przypadku komendy ALTER QMGR PARENT użyj `DISPLAY PUBSUB TYPE(PARENT) ALL` , aby śledzić status żądanej relacji nadrzędnej.

W przypadku komend ALTER QMGR REPOS i ALTER QMGR REPOSNL należy użyć komendy `DISPLAY CLUSQMGR QMTYPE` w celu potwierdzenia zakończenia.

- DEFINE CHANNEL, ALTER CHANNELi DELETE CHANNEL

W przypadku wszystkich parametrów wymienionych w tabeli ALTER CHANNEL parameters(Parametry ALTER CHANNEL) należy użyć komendy `DISPLAY CLUSQMGR` do monitorowania, gdy zmiany zostały rozpropagowane do klastra.

- DEFINE NAMELIST, ALTER NAMELISTi DELETE NAMELIST.

Jeśli w atrybucie **CLUSNL** obiektu **QMGR** używana jest wartość **NAMELIST** , to kolejka lub kanał klastra mogą mieć wpływ na ten obiekt. Monitoruj odpowiednio dla danego obiektu.

Zmiany wprowadzone w produkcie `SYSTEM.QPUBSUB.QQUEUE.NAMELIST` mogą mieć wpływ na tworzenie lub anulowanie subskrypcji proxy w hierarchii publikowania/subskrypcji. Użyj komendy `DISPLAY SUB SUBTYPE(PROXY)` , aby monitorować to działanie.

- Kolejki DEFINE, ALTER kolejki DELETE kolejki.

W przypadku wszystkich parametrów wymienionych w tabeli Parametry, które mogą zostać zwrócone przez komendę `DISPLAY QQUEUE`, należy użyć komendy `DISPLAY QCLUSTER` do monitorowania, gdy zmiany zostały rozpropagowane do klastra.

- DEFINE SUBi DELETE SUB

Po zdefiniowaniu pierwszej subskrypcji w łańcuchu tematu można utworzyć subskrypcje proxy w hierarchii publikowania/subskrypcji lub klastrze publikowania/subskrypcji. Podobnie, podczas usuwania ostatniej subskrypcji w łańcuchu tematu można anulować subskrypcje proxy w hierarchii publikowania/subskrypcji lub klastrze publikowania/subskrypcji.

Aby sprawdzić, czy komenda definiująca lub usuwała subskrypcję została zakończona, należy sprawdzić, czy oczekiwana subskrypcja proxy istnieje w innych menedżerach kolejek w sieci rozproszonej. Jeśli w klastrze używana jest *kierowanie bezpośrednio* , należy sprawdzić, czy oczekiwana subskrypcja proxy istnieje w innych częściowych repozytoriach w klastrze. Jeśli w klastrze jest używany *routing hosta tematów* , należy sprawdzić, czy oczekiwana subskrypcja proxy istnieje na pasujących hostach tematów. Użyj następującej komendy MQSC:

```
DISPLAY SUB(*) SUBTYPE(PROXY)
```

Tej samej kontroli należy użyć dla następujących równoważnych wywołań MQI subskrybuj i unsubscribe, gdy są one wydawane w klastrze lub w hierarchii:

- Subskrybuj za pomocą komendy `MQSUB`.
- Anuluj subskrypcję za pomocą komendy `MQCLOSE` z opcją `MQCO_REMOVE_SUB`.

- ZDEFINIUJ TEMAT, ALTER TOPIC i DELETE TOPIC

Aby sprawdzić, czy komenda definiująca, modyfikująca lub usuwająca temat w klastrze została zakończona, wyświetl temat w innych częściowych repozytoriach w klastrze (jeśli używany jest *kierowanie bezpośrednie*), lub na innych hostach tematów (jeśli używany jest *routing hosta tematów*).

Dla wszystkich parametrów wymienionych w tabeli Parametry, które mogą zostać zwrócone przez komendę `DISPLAY TOPIC`, należy użyć komendy `DISPLAY TCLUSTER`, aby monitorować, kiedy zmiany zostały rozpropagowane do klastra.

Uwaga:

- Parametr **CLUSTER** może mieć wpływ na tworzenie lub anulowanie subskrypcji proxy w klastrze publikowania/subskrypcji.
- Parametry **PROXYSUB** i **SUBSCOPE** mogą mieć wpływ na tworzenie lub anulowanie subskrypcji proxy w hierarchii publikowania/subskrypcji lub w klastrze publikowania/subskrybowania.
- Użyj komendy `DISPLAY SUB SUBTYPE (PROXYSUB)`, aby monitorować to działanie.

- ODŚWIEŻANIE KLASTRA

Jeśli uruchamiasz komendę **REFRESH CLUSTER**, odpytaj głębokość kolejki komend klastra. Przed wyszukaniem obiektów należy poczekać, aż osiągnie wartość zero i pozostanie na zero.

1. Użyj następującej komendy `MQSC`, aby sprawdzić, czy głębokość kolejki komend klastra wynosi zero.

```
DISPLAY QL(SYSTEM.CLUSTER.COMMAND.QUEUE) CURDEPTH
```

2. Należy powtórzyć operację sprawdzania, dopóki głębokość kolejki nie osiągnie wartości zero, a następnie w kolejnej operacji sprawdzania pozostanie na poziomie zero.

Komenda **REFRESH CLUSTER** usuwa i ponownie tworzy obiekty, a w dużych konfiguracjach może zająć dużo czasu. Więcej informacji na ten temat zawiera sekcja Uwagi dotyczące używania komendy REFRESH CLUSTER w przypadku klastrów publikowania/subskrybowania.

- TYP REFRESH QMGR (PROXYSUB)

Aby sprawdzić, czy komenda **REFRESH QMGR TYPE (PROXYSUB)** została zakończona, sprawdź, czy subskrypcje proxy zostały poprawione w innych menedżerach kolejek w sieci rozproszonej. Jeśli w klastrze używana jest opcja *kierowania bezpośredniego*, należy sprawdzić, czy subskrypcje proxy zostały poprawione w innych częściowych repozytoriach w klastrze. Jeśli w klastrze jest używany *routing hosta tematów*, należy sprawdzić, czy oczekiwane subskrypcje proxy zostały poprawione na zgodnych hostach tematów. Użyj następującej komendy `MQSC`:

```
DISPLAY SUB(*) SUBTYPE(PROXYSUB)
```

- Resetowanie klastra

Aby sprawdzić, czy komenda **RESET CLUSTER** została zakończona, należy użyć komendy `DISPLAY CLUSQMGR`.

- RESET QMGR TYPE (PUBSUB)

Aby sprawdzić, czy komenda **RESET QMGR** została zakończona, należy użyć komendy `DISPLAY PUBSUB TYPE (PARENT | CHILD)`.

Uwaga: Komenda **RESET QMGR** może spowodować anulowanie subskrypcji proxy w hierarchii publikowania/subskrypcji lub w klastrze publikowania/subskrypcji. Użyj komendy `DISPLAY SUB SUBTYPE (PROXYSUB)`, aby monitorować to działanie.

- Możliwe jest również monitorowanie innych kolejek systemowych, które, jak i kiedy komendy są zakończone, mają tendencję do osiągnięcia głębokości kolejki równej zero.

Na przykład można monitorować kolejkę SYSTEM . INTER . QMGR . CONTROL i kolejkę SYSTEM . INTER . QMGR . FANREQ . Patrz sekcja [Monitorowanie ruchu subskrypcji proxy w klastrach oraz Balancing producenci i konsumenci w sieciach publikowania/subskrypcji](#).

Co dalej

Jeśli te sprawdzenia nie potwierdzą, że komenda asynchroniczna została zakończona, może wystąpić błąd. Aby zbadać, najpierw należy sprawdzić dziennik dla menedżera kolejek, w którym została wydana komenda, a następnie (w przypadku klastra) sprawdzić dzienniki pełnego repozytorium klastra.

Informacje pokrewne

 [Asynchroniczne działanie komend CLUSTER w systemie z/OS](#)

Używanie definicji kolejek zdalnych jako aliasów

Oprócz lokalizacji kolejki w innym menedżerze kolejek można również użyć lokalnej definicji kolejki zdalnej dla aliasów menedżera kolejek i aliasów kolejek zwrotnych. Oba typy aliasów są tłumaczone za pomocą lokalnej definicji kolejki zdalnej. Aby komunikat dotarł do miejsca docelowego, należy skonfigurować odpowiednie kanały.

Alias menedżera kolejek

Alias jest procesem, za pomocą którego nazwa docelowego menedżera kolejek, określona w komunikacie, jest modyfikowana przez menedżer kolejek na trasie komunikatu. Alias menedżera kolejek są ważne, ponieważ można ich używać do sterowania miejscem docelowym komunikatów w sieci menedżerów kolejek.

W tym celu należy zmienić definicję kolejki zdalnej w menedżerze kolejek w punkcie kontroli. Aplikacja wysyłający nie ma informacji o tym, że określona nazwa menedżera kolejek jest aliasem.

Więcej informacji na temat aliasów menedżera kolejek zawiera sekcja [Czym są aliasy?](#).

Alias kolejek zwrotnych

Opcjonalnie aplikacja może określić nazwę kolejki zwrotnej, gdy w kolejce umieszcza *komunikat żądania* .

Jeśli aplikacja, która przetwarza komunikat, wyodrębnia nazwę kolejki odpowiedzi, wie, gdzie wystać *komunikat odpowiedzi* (jeśli jest to wymagane).

Alias kolejki odpowiedzi to proces, za pomocą którego kolejka odpowiedzi określona w komunikacie żądania jest zmieniana przez menedżer kolejek na trasie komunikatu. Aplikacja wysyłający nie jest świadoma, że podana nazwa kolejki odpowiedzi jest aliasem.

Alias kolejki odpowiedzi pozwala zmienić nazwę kolejki odpowiedzi i opcjonalnie jej menedżera kolejek. To z kolei umożliwia kontrolowanie, która trasa jest używana dla komunikatów odpowiedzi.

Więcej informacji na temat komunikatów żądań, komunikatów odpowiedzi i kolejek odpowiedzi można znaleźć w sekcji [Typy komunikatów](#) i [Kolejka odpowiedzi i menedżer kolejek](#).

Więcej informacji na temat aliasów kolejek odpowiedzi można znaleźć w temacie [Alias i aliasy kolejki odpowiedzi](#).

Konwersja danych

Dane komunikatu w zdefiniowanych formatach IBM MQ (znanych również jako formaty wbudowane) mogą być przekształcane przez menedżer kolejek z jednego kodowanego zestawu znaków na inny, pod warunkiem że oba zestawy znaków odnoszą się do jednego języka lub grupy podobnych języków.

Na przykład, konwersja między kodowanymi zestawami znaków z identyfikatorami (CCSID) 850 i 500 jest obsługiwana, ponieważ oba mają zastosowanie do języków zachodnioeuropejskich.

W przypadku konwersji znaków w kodzie EBCDIC nowej linii (NL) na format ASCII, należy zapoznać się z wszystkimi menedżerami kolejek.

Obsługiwane konwersje są zdefiniowane w sekcji Przetwarzanie konwersji danych.

Gdy menedżer kolejek nie może przekształcić komunikatów w wbudowanych formatach

Menedżer kolejek nie może automatycznie konwertować komunikatów we wbudowanych formatach, jeśli ich identyfikatory CCSID reprezentują różne grupy języków narodowych. Na przykład konwersja między identyfikatorem CCSID 850 i identyfikatorem CCSID 1025 (który jest zestawem znaków kodowanych EBCDIC dla języków przy użyciu skryptu cyrylicą) nie jest obsługiwana, ponieważ wiele znaków w jednym kodowanym zestawie znaków nie może być reprezentowane w drugim. Jeśli istnieje sieć menedżerów kolejek pracujących w różnych językach narodowych, a konwersja danych między niektórymi kodowanymi zestawami znaków nie jest obsługiwana, można włączyć konwersję domyślną.

V 9.0.0 W przypadku platform, do których ma zastosowanie `ccsid_part2.tbl`, w celu uzyskania dalszych informacji należy zapoznać się z informacjami w `ccsid_part2.tbl` sekcji "Określanie domyślnej konwersji danych" na stronie 213, Domyślna konwersja danych na platformach innych niż te, do których ma zastosowanie plik `ccsid_part2.tbl`, jest opisana w sekcji "Domyślna konwersja danych" na stronie 211.

Rozszerzona obsługa konwersji danych Unicode w produkcie IBM MQ 9.0

V 9.0.0

W wersjach wcześniejszych niż IBM MQ 9.0 poprzednie wersje produktu nie obsługiwały konwersji danych zawierających punkty kodowe Unicode poza podstawową wielojęzyczną płytą (punkty kodowe powyżej U+FFFF). Obsługa konwersji danych w standardzie Unicode była ograniczona do punktów kodowych zdefiniowanych w standardzie Unicode 3.0, zakodowanych w UTF-8 lub UCS-2, 2-bajtowym podzbiore o stałej szerokości UTF-16.

W produkcie IBM MQ 9.0 produkt IBM MQ obsługuje wszystkie znaki Unicode zdefiniowane w standardzie Unicode 8.0 w konwersji danych. Obejmuje to pełne wsparcie dla UTF-16, w tym pary odpowiedników (para dwubajtowych znaków UTF-16 z zakresu od X' D800 ' do X' DFFF ' , które reprezentują kod Unicode powyżej U+FFFF).

Łączenie sekwencji znaków jest również obsługiwane w przypadkach, gdy wstępnie złożony znak w jednym identyfikatorze CCSID jest odwzorowywany na sekwencję znaków łączących w innym CCSID.

Konwersja danych do i z Unicode i CCSID 1388, 1390, 1399, 4933, 5488 i 16884 została rozszerzona, na niektórych platformach, w celu obsługi wszystkich punktów kodowych zdefiniowanych obecnie dla tych identyfikatorów CCSID, w tym tych, które odwzorowują punkty kodowe na dodatkowe płaszczyzny Unicode.

W przypadku identyfikatorów CCSID 1390, 1399 i 16884, obejmuje to znaki zdefiniowane w standardzie JIS X 0213 (JIS2004).

Dodano także obsługę konwersji do i z kodu Unicode i sześciu nowych identyfikatorów CCSID (od 1374 do 1379).

Plik `ccsid_part2.tbl`

V 9.0.0

Z programu IBM MQ 9.0 udostępniany jest dodatkowy plik `ccsid_part2.tbl`.

Plik `ccsid_part2.tbl` ma pierwszeństwo przed plikiem `ccsid.tbl`, a ponadto:

- Umożliwia dodawanie i modyfikowanie pozycji CCSID
- Określa domyślną konwersję danych
- Określa dane dla różnych poziomów komend

Plik `ccsid_part2.tbl` stosuje się wyłącznie w przypadku następujących platform:

- ▶ **Linux** Wszystkie wersje systemu Linux
- ▶ **Solaris** Solaris
- ▶ **Windows** Windows

▶ **Windows** W produkcie IBM MQ 9.0 w systemie IBM MQ for Windows `ccsid_part2.tbl` domyślnie znajduje się w katalogu `MQDataRoot\conv\table`. Ponadto w systemie IBM MQ for Windows rejestruje wszystkie obsługiwane zestawy kodowe.

▶ **Solaris** ▶ **Linux** Z poziomu produktu IBM MQ 9.0 na platformach IBM MQ for Linux i Solaris `ccsid_part2.tbl` znajduje się w katalogu `MQDataRoot/conv/table`. W przypadku wszystkich platform Linux i Solaris obsługiwane zestawy kodów są przechowywane w tabelach konwersji udostępnianych przez produkt IBM MQ.

Mimo że plik `ccsid_part2.tbl` zastępuje istniejący plik `ccsid.tbl` używany w poprzednich wersjach produktu IBM MQ do dostarczania dodatkowych informacji o identyfikatorze CCSID, plik `ccsid.tbl` nadal jest analizowany przez program IBM MQ i dlatego nie może zostać usunięty.

Więcej informacji na ten temat zawiera sekcja [“Plik `ccsid_part2.tbl`”](#) na stronie 212.

ccsid.tbl plik

▶ **V 9.0.0** Na platformach innych niż te, do których ma zastosowanie produkt `ccsid_part2.tbl`, plik `ccsid.tbl` jest używany do następujących celów:

- ▶ **HP-UX** ▶ **AIX** Na platformach AIX i HP-UX obsługiwane zestawy kodów są przechowywane wewnętrznie przez system operacyjny.
- Określa on wszystkie dodatkowe zestawy kodowe. Aby określić dodatkowe zestawy kodowe, należy dokonać edycji `ccsid.tbl` (wskazówki dotyczące sposobu wykonania tej czynności znajdują się w pliku).
- Określa ona domyślną konwersję danych.

Informacje zapisane w programie `ccsid.tbl` można zaktualizować. Można to zrobić, jeśli na przykład w przyszłej wersji systemu operacyjnego obsługiwane są dodatkowe kodowane zestawy znaków.

Domyślna konwersja danych

▶ **V 9.0.0** Metoda domyślnej konwersji danych została zmieniona z IBM MQ 9.0 na następujące platformy:

- Wszystkie wersje systemu Linux
- Solaris
- Windows

Więcej informacji na ten temat zawiera `ccsid_part2.tbl` sekcja [“Określanie domyślnej konwersji danych”](#) na stronie 213.

Jeśli kanały są skonfigurowane między dwoma komputerami, w których konwersja danych nie jest normalnie obsługiwana, należy włączyć domyślną konwersję danych dla kanałów do pracy.

▶ **V 9.0.0** Na platformach innych niż te, do których ma zastosowanie produkt `ccsid_part2.tbl`, aby włączyć domyślną konwersję danych, należy zmodyfikować plik `ccsid.tbl` w celu określenia domyślnego identyfikatora CCSID EBCDIC i domyślnego identyfikatora CCSID ASCII. Instrukcje dotyczące tego, jak to zrobić, są zawarte w pliku. Należy to zrobić na wszystkich komputerach, które będą połączone za pomocą kanałów. Zrestartuj menedżer kolejek, aby zmiany zostały uwzględnione.

Domyślny proces konwersji danych jest następujący:

- Jeśli konwersja między źródłowym i docelowym identyfikatorem CCSID nie jest obsługiwana, ale identyfikatory CCSID w środowisku źródłowym i docelowym są zarówno EBCDIC, jak i ASCII, dane znakowe są przekazywane do aplikacji docelowej bez konwersji.
- Jeśli jeden identyfikator CCSID reprezentuje kodowany zestaw znaków ASCII, a drugi reprezentuje kodowany zestaw znaków EBCDIC, program IBM MQ przekształca dane przy użyciu domyślnych identyfikatorów CCSID konwersji danych zdefiniowanych w pliku `ccsid.tbl`.

Uwaga: Spróbuj ograniczyć przekształcanie znaków do tych, które mają te same wartości kodowe w kodowanym zestawie znaków określonym dla komunikatu i w domyślnym kodowanym zestawie znaków. Jeśli używany jest tylko zestaw znaków, który jest poprawny dla nazw obiektów produktu IBM MQ (zgodnie z definicją w sekcji [Nazewnictwo obiektów IBM MQ](#)) W ogólności, spełniasz ten wymóg. Wyjątki występują w przypadku identyfikatorów CCSID EBCDIC 290, 930, 1279 i 5026 używanych w Japonii, gdzie małe litery mają różne kody od tych, które są używane w innych identyfikatorach CCSID EBCDIC.

Przekształcanie komunikatów w formatach zdefiniowanych przez użytkownika

Menedżer kolejek nie może konwertować komunikatów w formatach zdefiniowanych przez użytkownika z jednego kodowanego zestawu znaków na inny. Jeśli zachodzi potrzeba konwersji danych w formacie zdefiniowanym przez użytkownika, należy podać wyjście konwersji danych dla każdego takiego formatu. Nie należy używać domyślnych identyfikatorów CCSID do przekształcania danych znakowych w formatach zdefiniowanych przez użytkownika. Więcej informacji na temat przekształcania danych w formatach zdefiniowanych przez użytkownika oraz zapisywania wyjść konwersji danych znajduje się w sekcji [Zapisywanie wyjść konwersji danych](#).

Zmiana identyfikatora CCSID menedżera kolejek

Po użyciu atrybutu CCSID komendy ALTER QMGR w celu zmiany identyfikatora CCSID menedżera kolejek należy zatrzymać i zrestartować menedżer kolejek, aby upewnić się, że wszystkie uruchomione aplikacje, w tym serwer komend i programy kanałów, zostaną zatrzymane i zrestartowane.

Jest to konieczne, ponieważ wszystkie aplikacje działające w momencie zmiany identyfikatora CCSID menedżera kolejek nadal korzystają z istniejącego identyfikatora CCSID.

Plik `ccsid_part2.tbl`

W programie IBM MQ 9.0plik `ccsid_part2.tbl` zastępuje istniejący plik `ccsid.tbl` używany w poprzednich wersjach produktu w celu dostarczenia dodatkowych informacji o identyfikatorze CCSID. Należy pamiętać, że plik `ccsid.tbl` nadal jest analizowany przez program IBM MQ i nie powinien być usuwany, a wpisy w programie `ccsid_part2.tbl` mają pierwszeństwo przed innymi wpisami w produkcie `ccsid.tbl`.

`ccsid_part2.tbl` znajduje się w katalogu `MQDataRoot/conv/table` :

- Wszystkie wersje systemu Linux
- Solaris

oraz katalog `MQDataRoot\conv\table` w systemie Windows

Należy pamiętać, że plik `ccsid_part2.tbl` nie jest dostępny na innych platformach. Plik ten jest używany do wykonywania następujących działań:

- Dodawanie lub modyfikowanie istniejących pozycji CCSID w celu ich użycia w konwersji danych IBM MQ
- Określanie domyślnej konwersji danych

Jako tego pliku należy użyć produktu `ccsid_part2.tbl`:

- Zawiera obsługę nowych wartości kodowania Unicode.
- Umożliwia określenie wersji pozycji CCSID, tak aby pozycje te miały zastosowanie tylko do wybranych poziomów komend.

Dodawanie lub modyfikowanie pozycji CCSID

Wpis w pliku `ccsid_part2.tbl` ma następujący format:

```
<CCSID number> <Base CCSID> <DBCS CodePage> <SBCS CodePage>  
<Type> <Encoding> <ACRI> <Name>
```

Przykładową pozycją dla identyfikatora CCSID 1200 (UTF-16) jest:

```
1200 1200 1200 1200 3 8 0 UTF-16
```

Uwaga: Więcej informacji na temat wartości ACRI zawiera komentarz w pliku `ccsid_part2.tbl`.

W formacie `ccsid_part2.tbl`:

Typ może być równy:

1=SBCS

2=DBCS

3=MBCS

Kodowanie może być równe:

1=EBCDIC

2 = ASCII

3 = ISO

4 = UCS-2

5 = UTF-8

6 = PWE

7 = GB18030

8 = UTF-16

9 = UTF-32

Podczas edytowania pliku:

- Komentarz można określić, korzystając z symbolu `#` na początku wiersza. Zapobiega to próbie przeanalizowania wiersza przez program IBM MQ.
- Nie można dostarczyć komentarzy wstawianych do wiersza.
- Należy upewnić się, że nie są tworzone puste wiersze.
- Nie wolno dodawać nowych pozycji na końcu pliku.

Nowe pozycje CCSID powinny zostać dodane przed informacjami w tabeli ACRI.

Określanie domyślnej konwersji danych

Można zdefiniować domyślne identyfikatory CCSID, które są używane do konwersji między ASCII lub podobnymi i EBCDIC CCSID, jeśli nie są obsługiwane żadne konwersje między dwoma identyfikatorami CCSID.

Jeśli ta funkcja zostanie włączona, domyślna konwersja jest używana dla nagłówków transmisji i komunikatów, a także może być używana w konwersji danych użytkownika.

Konwersje domyślne są włączone, tworząc dwa wiersze podobne do następujących:

```
default      0      500      1      1      0  
default      0      850      1      2      0
```

W pierwszym wierszu jest ustawiana wartość domyślna dla identyfikatorów CCSID EBCDIC na 500, a w drugim wierszu jest ustawiana wartość domyślna dla ASCII i podobnych identyfikatorów CCSID 850.

Określanie danych dla różnych poziomów komend

Aby określić pozycje CCSID dla różnych poziomów komend systemu IBM MQ, należy użyć symbolu dwukropka, a następnie poziomu komendy (lub poziomu komend) programu IBM MQ, do którego ma być stosowana kolejna sekcja.

Liczba ta reprezentuje minimalny poziom komendy, w którym musi być uruchomiony menedżer kolejek lub klient. Na przykład, jeśli bieżący menedżer kolejek jest poziomem komendy 900 i napotka flagę na poziomie 800 lub 900 komend, zostaną odczytane identyfikatory CCSID.

Jednak menedżer kolejek na poziomie 800 ignoruje wszystkie identyfikatory CCSID w sekcji 900.

Podany poziom komendy ma zastosowanie do wszystkich pozycji CCSID napotkanych po opcji poziomu komendy, dopóki nie zostanie znaleziona nowa opcja poziomu komendy.

Jeśli wymagane jest ustawienie poziomu komendy na wszystkich poziomach komend, należy określić wartość zero.

Podczas pierwszego analizowania `ccsid_part2.tbl` program IBM MQ traktuje wszystkie napotkane identyfikatory CCSID jako poprawne dla wszystkich poziomów komend produktu IBM MQ.

Kontrola wersji jest uruchamiana tylko wtedy, gdy program IBM MQ napotka pierwszą flagę na poziomie komendy.

Poniższy fragment kodu przedstawia przykład użycia kontroli wersji:

```
# Comment Block
# End of Comment Block
# Because no command level flag is specified and we're at the start of the file
# the following CCSIDs will be read on all versions
  819  819    0    819    1    3    0    IS08859-1
  923  923    0    923    1    3    0    IS08859-15
 1051 1051    0   1051    1    3    0    IBM-1051
# The colon :900 below shows that the CCSIDs after will only be for MQ cmd level 900 and above
:900
  8629 437    0    437    1    2    0    IBM-437
 12725 437    0    437    1    2    0    IBM-437
 16821 437    0    437    1    2    0    IBM-437
 20917 437    0    437    1    2    0    IBM-437
# The colon :0 below shows that the CCSIDs after will be for all version of MQ
:0
  4946 850    0    850    1    2    0    IBM-850
 33618 850    0    850    1    2    0    IBM-850
 61697 850    0    850    1    2    0    IBM-850
 61698 850    0    850    1    2    0    IBM-850
```

Windows

Linux

AIX

administrowanieMQ Telemetry

Produkt MQ Telemetry jest administrowany przy użyciu produktu IBM MQ Explorer lub w wierszu komend. Za pomocą eksploratora można skonfigurować kanały telemetryczne, sterować usługą telemetryczną i monitorować klienty MQTT połączone z produktem IBM MQ. Skonfiguruj zabezpieczenia produktu MQ Telemetry przy użyciu usług JAAS, TLS i menedżera uprawnień do obiektów produktu IBM MQ.

Administrowanie przy użyciu produktu IBM MQ Explorer

Za pomocą eksploratora można skonfigurować kanały telemetryczne, sterować usługą telemetryczną i monitorować klienty MQTT połączone z produktem IBM MQ. Skonfiguruj zabezpieczenia produktu MQ Telemetry przy użyciu usług JAAS, TLS i menedżera uprawnień do obiektów produktu IBM MQ.

Administrowanie za pomocą wiersza komend

Produkt MQ Telemetry można w pełni administrować w wierszu komend za pomocą komend IBM MQ [MQSC](#).

Dokumentacja produktu MQ Telemetry zawiera również przykładowe skrypty, które demonstrują podstawowe użycie aplikacji klienckiej IBM MQ Telemetry Transport v3.

Przed rozpoczęciem korzystania z nich należy zapoznać się z przykładami w przykładowych programach [IBM MQ Telemetry Transport](#).

Informacje pokrewne

[MQ Telemetry](#)

[Właściwości MQXR](#)

Linux

AIX

Konfigurowanie menedżera kolejek dla telemetrii w systemach Linux i AIX

Wykonaj poniższe czynności, aby skonfigurować menedżer kolejek do uruchamiania produktu MQ Telemetry. Można uruchomić zautomatyzowaną procedurę w celu skonfigurowania prostszej konfiguracji przy użyciu obsługi produktu MQ Telemetry dla produktu IBM MQ Explorer.

Zanim rozpocznie

1. Informacje na temat instalowania produktu IBM MQ oraz opcji MQ Telemetry zawiera sekcja [Instalowanie produktu MQ Telemetry](#).
2. Utwórz i uruchom menedżer kolejek. W tym zadaniu menedżer kolejek jest określany jako *qMgr*.
3. W ramach tego zadania można skonfigurować usługę telemetryczną (MQXR). Ustawienia właściwości MQXR są przechowywane w pliku właściwości specyficznym dla platformy: `mqxr_unix.properties`. Zwykle nie ma potrzeby bezpośredniego edytowania pliku właściwości MQXR, ponieważ prawie wszystkie ustawienia można konfigurować za pomocą komend administracyjnych MQSC lub IBM MQ Explorer. Jeśli zdecydujesz się bezpośrednio na edycję pliku, zatrzymaj menedżer kolejek przed dokonaniem zmian. Patrz: [właściwości MQXR](#).

O tym zadaniu

Obsługa MQ Telemetry dla produktu IBM MQ Explorer zawiera kreatora i przykładową procedurę komendy `sampleMQM`. Konfigurują one początkową konfigurację przy użyciu identyfikatora użytkownika gościa. Patrz sekcja [Sprawdzanie poprawności instalacji produktu MQ Telemetry przy użyciu programu IBM MQ Explorer i programów przykładowych IBM MQ Telemetry Transport](#).

Aby ręcznie skonfigurować produkt MQ Telemetry przy użyciu różnych schematów autoryzacji, należy wykonać kroki opisane w tej czynności.

Procedura

1. Otwórz okno komend w katalogu przykładów telemetrycznych.
Katalog przykładów telemetrycznych to `/opt/mqm/mqxr/samples`.
2. Utwórz kolejkę transmisji telemetrycznej.

```
echo "DEFINE QLOCAL('SYSTEM.MQTT.TRANSMIT.QUEUE') USAGE(XMITQ) MAXDEPTH(100000)" | runmqsc qMgr
```

Gdy usługa telemetryczna (MQXR) jest uruchamiana po raz pierwszy, tworzy ona `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Jest on tworzony ręcznie w tym zadaniu, ponieważ produkt `SYSTEM.MQTT.TRANSMIT.QUEUE` musi istnieć przed uruchomieniem usługi telemetrycznej (MQXR), aby autoryzować dostęp do niej.

3. Ustaw domyślną kolejkę transmisji

Gdy usługa telemetryczna (MQXR) jest uruchamiana po raz pierwszy, nie zmienia menedżera kolejek, aby `SYSTEM.MQTT.TRANSMIT.QUEUE` była domyślną kolejką transmisji.

Aby uczynić SYSTEM.MQTT.TRANSMIT.QUEUE domyślną kolejką transmisji, zmień domyślną właściwość kolejki transmisji. Zmień właściwość za pomocą konsoli IBM MQ Explorer lub za pomocą komendy w następującym przykładzie:

```
echo "ALTER QMGR DEFXMITQ('SYSTEM.MQTT.TRANSMIT.QUEUE')" | runmqsc qMgr
```

Zmiana domyślnej kolejki transmisji może kolidować z istniejącą konfiguracją. Przyczyną zmiany domyślnej kolejki transmisji na SYSTEM.MQTT.TRANSMIT.QUEUE jest ułatwienie wysyłania komunikatów bezpośrednio do klientów MQTT. Bez zmiany domyślnej kolejki transmisji należy dodać definicję kolejki zdalnej dla każdego klienta, który odbiera komunikaty produktu IBM MQ Explorer; patrz [“Bezpośrednie wysyłanie wiadomości do klienta”](#) na stronie 220.

4. Aby utworzyć jeden lub więcej identyfikatorów użytkowników, należy wykonać procedurę w produkcie [“Autoryzowanie klientów MQTT do uzyskiwania dostępu do obiektów IBM MQ”](#) na stronie 222. Identyfikatory użytkowników mają uprawnienia do publikowania, subskrybowania i wysyłania publikacji do klientów MQTT.
5. Zainstaluj usługę telemetryczną (MQXR)

```
cat /opt/<install_dir>/mqxr/samples/installMQXRService_unix.mqsc | runmqsc qMgr
```

Patrz także przykładowy kod w produkcie [Rysunek 20](#) na stronie 216.

6. Uruchom usługę

```
echo "START SERVICE(SYSTEM.MQXR.SERVICE)" | runmqsc qMgr
```

Usługa telemetryczna (MQXR) jest uruchamiana automatycznie po uruchomieniu menedżera kolejek.

Jest on uruchamiany ręcznie w tym zadaniu, ponieważ menedżer kolejek jest już uruchomiony.

7. Korzystając z produktu IBM MQ Explorer, skonfiguruj kanały telemetryczne do akceptowania połączeń od klientów produktu MQTT.

Kanały telemetryczne muszą być skonfigurowane w taki sposób, aby ich tożsamości były jednym z identyfikatorów zdefiniowanych w kroku 4.

Patrz także [DEFINE CHANNEL \(MQTT\)](#).

8. Sprawdź konfigurację, uruchamiając przykładową klienta.

Aby przykładowy klient współpracował z kanałem telemetryczny, kanał musi autoryzować klienta do publikowania, subskrybowania i odbierania publikacji. Przykładowy klient łączy się domyślnie z kanałem telemetrycznej na porcie 1883. Patrz także [IBM MQ Telemetry Transport przykładowe programy](#).

Przykład

[Rysunek 20](#) na stronie 216 wyświetla komendę `runmqsc`, aby ręcznie utworzyć SYSTEM.MQXR.SERVICE na serwerze Linux.

```
DEF      SERVICE(SYSTEM.MQXR.SERVICE) +
CONTROL(QMGR) +
DESCR('Manages clients using MQXR protocols such as MQTT') +
SERVTYPE(SERVER) +
STARTCMD('+MQ_INSTALL_PATH+/mqxr/bin/runMQXRService.sh') +
STARTARG('-m +QMNAME+ -d "+MQ_Q_MGR_DATA_PATH+" -g "+MQ_DATA_PATH+"') +
STOPCMD('+MQ_INSTALL_PATH+/mqxr/bin/endMQXRService.sh') +
STOPARG('-m +QMNAME+') +
STDOUT('+MQ_Q_MGR_DATA_PATH+/mqxr.stdout') +
STDERR('+MQ_Q_MGR_DATA_PATH+/mqxr.stderr')
```

Rysunek 20. installMQXRService_unix.mqsc

Konfigurowanie menedżera kolejek dla telemetrii w systemie Windows

Wykonaj poniższe czynności, aby skonfigurować menedżer kolejek do uruchamiania produktu MQ Telemetry. Można uruchomić zautomatyzowaną procedurę w celu skonfigurowania prostszej konfiguracji przy użyciu obsługi produktu MQ Telemetry dla produktu IBM MQ Explorer.

Zanim rozpoczniesz

1. Informacje na temat instalowania produktu IBM MQ oraz opcji MQ Telemetry zawiera sekcja [Instalowanie produktu MQ Telemetry](#).
2. Utwórz i uruchom menedżer kolejek. W tym zadaniu menedżer kolejek jest określany jako *qMgr*.
3. W ramach tego zadania można skonfigurować usługę telemetryczną (MQXR). Ustawienia właściwości MQXR są przechowywane w pliku właściwości specyficznym dla platformy: `mqxr_win.properties`. Zwykle nie ma potrzeby bezpośredniego edytowania pliku właściwości MQXR, ponieważ prawie wszystkie ustawienia można konfigurować za pomocą komend administracyjnych MQSC lub IBM MQ Explorer. Jeśli zdecydujesz się bezpośrednio na edycję pliku, zatrzymaj menedżer kolejek przed dokonaniem zmian. Patrz: [właściwości MQXR](#).

O tym zadaniu

Obsługa MQ Telemetry dla produktu IBM MQ Explorer zawiera kreatora i przykładową procedurę komendy `sampleMQM`. Konfigurują one początkową konfigurację przy użyciu identyfikatora użytkownika gościa. Patrz sekcja [Sprawdzanie poprawności instalacji produktu MQ Telemetry przy użyciu programu IBM MQ Explorer i programów przykładowych IBM MQ Telemetry Transport](#).

Aby ręcznie skonfigurować produkt MQ Telemetry przy użyciu różnych schematów autoryzacji, należy wykonać kroki opisane w tej czynności.

Procedura

1. Otwórz okno komend w katalogu przykładów telemetrycznych.
Katalog przykładów telemetrycznych to `WMQ program installation directory\mqxr\samples`.
2. Utwórz kolejkę transmisji telemetrycznej.

```
echo DEFINE QLOCAL('SYSTEM.MQTT.TRANSMIT.QUEUE') USAGE(XMITQ) MAXDEPTH(100000) | runmqsc qMgr
```

Gdy usługa telemetryczna (MQXR) jest uruchamiana po raz pierwszy, tworzy ona `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Jest on tworzony ręcznie w tym zadaniu, ponieważ produkt `SYSTEM.MQTT.TRANSMIT.QUEUE` musi istnieć przed uruchomieniem usługi telemetrycznej (MQXR), aby autoryzować dostęp do niej.

3. Ustaw domyślną kolejkę transmisji dla *qMgr*

```
echo ALTER QMGR DEFQMITQ('SYSTEM.MQTT.TRANSMIT.QUEUE') | runmqsc qMgr
```

Rysunek 21. Ustaw domyślną kolejkę transmisji

Gdy usługa telemetryczna (MQXR) jest uruchamiana po raz pierwszy, nie zmienia menedżera kolejek, aby `SYSTEM.MQTT.TRANSMIT.QUEUE` była domyślną kolejką transmisji.

Aby uczynić `SYSTEM.MQTT.TRANSMIT.QUEUE` domyślną kolejką transmisji, zmień domyślną właściwość kolejki transmisji. Zmień właściwość za pomocą IBM MQ Explorer lub za pomocą komendy w programie [Rysunek 21 na stronie 217](#).

Zmiana domyślnej kolejki transmisji może kolidować z istniejącą konfiguracją. Przyczyną zmiany domyślnej kolejki transmisji na `SYSTEM.MQTT.TRANSMIT.QUEUE` jest ułatwienie wysyłania komunikatów bezpośrednio do klientów MQTT. Bez zmiany domyślnej kolejki transmisji należy dodać definicję kolejki zdalnej dla każdego klienta, który odbiera komunikaty produktu IBM MQ; patrz [“Bezpośrednie wysyłanie wiadomości do klienta”](#) na stronie 220.

4. Aby utworzyć jeden lub więcej identyfikatorów użytkowników, należy wykonać procedurę w produkcie [“Autoryzowanie klientów MQTT do uzyskiwania dostępu do obiektów IBM MQ”](#) na stronie 222. Identyfikatory użytkowników mają uprawnienia do publikowania, subskrybowania i wysyłania publikacji do klientów MQTT.
5. Zainstaluj usługę telemetryczną (MQXR)

```
type
installMQXRService_win.mqsc | runmqsc qMgr
```

6. Uruchom usługę

```
echo START SERVICE(SYSTEM.MQXR.SERVICE) | runmqsc qMgr
```

Usługa telemetryczna (MQXR) jest uruchamiana automatycznie po uruchomieniu menedżera kolejek.

Jest on uruchamiany ręcznie w tym zadaniu, ponieważ menedżer kolejek jest już uruchomiony.

7. Korzystając z produktu IBM MQ Explorer, skonfiguruj kanały telemetryczne do akceptowania połączeń od klientów produktu MQTT.

Kanały telemetryczne muszą być skonfigurowane w taki sposób, aby ich tożsamości były jednym z identyfikatorów zdefiniowanych w kroku 4.

Patrz także [DEFINE CHANNEL \(MQTT\)](#).

8. Sprawdź konfigurację, uruchamiając przykładową klienta.

Aby przykładowy klient współpracował z kanałem telemetryczny, kanał musi autoryzować klienta do publikowania, subskrybowania i odbierania publikacji. Przykładowy klient łączy się domyślnie z kanałem telemetrycznej na porcie 1883. Patrz także [IBM MQ Telemetry Transport przykładowe programy](#).

Ręczne tworzenie `SYSTEM.MQXR.SERVICE`

Rysunek 22 na stronie 218 wyświetla komendę `runmqsc`, aby ręcznie utworzyć `SYSTEM.MQXR.SERVICE` na serwerze Windows.

```
DEF SERVICE(SYSTEM.MQXR.SERVICE) +
CONTROL(QMGR) +
DESCR('Manages clients using MQXR protocols such as MQTT') +
SERVTYPE(SERVER) +
STARTCMD('+MQ_INSTALL_PATH+mqxr\bin\runMQXRService.bat') +
STARTARG('-m +QMNAME+ -d "+MQ_Q_MGR_DATA_PATH+\" -g "+MQ_DATA_PATH+\"') +
STOPCMD('+MQ_INSTALL_PATH+mqxr\bin\endMQXRService.bat') +
STOPARG('-m +QMNAME+') +
STDOUT('+MQ_Q_MGR_DATA_PATH+mqxr.stdout') +
STDERR('+MQ_Q_MGR_DATA_PATH+mqxr.stderr')
```

Rysunek 22. `installMQXRService_win.mqsc`

Windows Linux AIX Konfigurowanie rozproszonego kolejkowania w celu wysyłania komunikatów do klientów MQTT

Aplikacje produktu IBM MQ mogą wysyłać komunikaty klientów produktu MQTT v3 przez publikowanie do subskrypcji utworzonej przez klienta lub bezpośrednio przez wystanie komunikatu. Niezależnie od tego, która metoda jest używana, komunikat jest umieszczany na serwerze `SYSTEM.MQTT.TRANSMIT.QUEUE` i wysyłany do klienta przez usługę telemetryczną (MQXR). Istnieje wiele sposobów umieszczania komunikatu w produkcie `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Publikowanie komunikatu w odpowiedzi na subskrypcję klienta MQTT

Usługa telemetryczna (MQXR) tworzy subskrypcję w imieniu klienta MQTT. Klient jest miejscem docelowym dla wszystkich publikacji, które są zgodne z subskrypcją wysłanej przez klienta. Usługi telemetryczne przekazuje zgodne publikacje z powrotem do klienta.

Klient MQTT jest połączony z serwerem IBM MQ jako menedżer kolejek, a jego nazwa menedżera kolejek jest ustawiona na wartość *ClientIdentifier*. Miejscem docelowym publikacji, które mają zostać wysłane do klienta, jest kolejka transmisji *SYSTEM.MQTT.TRANSMIT.QUEUE*. Usługa telemetryczna przekazuje komunikaty w systemie *SYSTEM.MQTT.TRANSMIT.QUEUE* do klientów MQTT, używając nazwy docelowego menedżera kolejek jako klucza dla konkretnego klienta.

Usługa telemetryczna (MQXR) otwiera kolejkę transmisji przy użyciu wartości *ClientIdentifier* jako nazwy menedżera kolejek. Usługa telemetryczna (MQXR) przekazuje uchwyt obiektu kolejki do wywołania *MQSUB*, aby przekazać publikacje, które są zgodne z subskrypcją klienta. W rozwiązaniu nazwy obiektu *ClientIdentifier* jest tworzona jako nazwa zdalnego menedżera kolejek, a kolejka transmisji musi być przetłumaczona na *SYSTEM.MQTT.TRANSMIT.QUEUE*. Korzystając ze standardowej rozdzielczości nazwy obiektu IBM MQ, produkt *ClientIdentifier* jest rozstrzygany w następujący sposób; patrz [Tabela 14 na stronie 219](#).

1. *ClientIdentifier* nie pasuje do niczego.

ClientIdentifier jest nazwą zdalnego menedżera kolejek. Nazwa ta nie jest zgodna z nazwą menedżera kolejek lokalnych, aliasem menedżera kolejek lub nazwą kolejki transmisji.

Nazwa kolejki nie jest zdefiniowana. Obecnie usługa telemetryczna (MQXR) ustawia *SYSTEM.MQTT.PUBLICATION.QUEUE* jako nazwę kolejki. Klient MQTT v3 nie obsługuje kolejek, dlatego rozstrzygnięta nazwa kolejki jest ignorowana przez klienta.

Właściwość lokalnego menedżera kolejek, Domyślna kolejka transmisji, musi być ustawiona na wartość *SYSTEM.MQTT.TRANSMIT.QUEUE*, co powoduje, że publikacja jest umieszczana na serwerze *SYSTEM.MQTT.TRANSMIT.QUEUE* w celu wystania do klienta.

2. *ClientIdentifier* jest zgodny z aliasem menedżera kolejek o nazwie *ClientIdentifier*.

ClientIdentifier jest nazwą zdalnego menedżera kolejek. Jest ona zgodna z nazwą aliasu menedżera kolejek.

Alias menedżera kolejek musi być zdefiniowany z *ClientIdentifier* jako nazwa zdalnego menedżera kolejek.

Ustawienie nazwy kolejki transmisji w definicji aliasu menedżera kolejek nie jest konieczne, aby domyślna transmisja została ustawiona na wartość *SYSTEM.MQTT.TRANSMIT.QUEUE*.

Tabela 14. Tłumaczenie nazw aliasu menedżera kolejek produktu MQTT					
	Wejście		Wyjście		
<i>ClientIdentifier</i>	Nazwa menedżera kolejek	Nazwa kolejki	Nazwa menedżera kolejek	Nazwa kolejki	Kolejka transmisji
Nic nie odpowiada	<i>ClientIdentifier</i>	<i>niezdefiniowane</i>	<i>ClientIdentifier</i>	<i>niezdefiniowane</i>	Domyślna kolejka transmisji. <i>SYSTEM.MQTT.TRANSMIT.QUEUE</i>
Jest zgodny z aliasem menedżera kolejek o nazwie <i>ClientIdentifier</i>	<i>ClientIdentifier</i>	<i>niezdefiniowane</i>	<i>ClientIdentifier</i>	<i>niezdefiniowane</i>	<i>SYSTEM.MQTT.TRANSMIT.QUEUE</i>

Więcej informacji na temat tłumaczenia nazw zawiera sekcja [Rozdzielczość nazw](#).

Każdy program IBM MQ może publikować w tym samym temacie. Publikacja jest wysyłana do subskrybentów, w tym do klientów MQTT v3, którzy mają subskrypcję tematu.

Jeśli temat administracyjny został utworzony w klastrze, z atrybutem CLUSTER(*clusterName*), dowolna aplikacja w klastrze może publikować do klienta, na przykład:

```
echo DEFINE TOPIC('MQTTExamples') TOPICSTR('MQTT Examples') CLUSTER(MQTT) REPLACE | runmqsc qMgr
```

Rysunek 23. Definiowanie tematu klastra w systemie Windows

Uwaga: Nie należy nadawać SYSTEM.MQTT.TRANSMIT.QUEUE atrybutu klastra.

Subskrybenci i publikatory klientów MQTT mogą łączyć się z różnymi menedżerami kolejek. Subskrybenci i publikatorzy mogą być częścią tego samego klastra lub mogą być połączone w hierarchii publikowania/subskrypcji. Publikacja jest dostarczana z publikatora do subskrybenta przy użyciu produktu IBM MQ.

Bezpośrednie wysyłanie wiadomości do klienta

Alternatywa dla klienta tworzącego subskrypcję i otrzymującego publikację zgodną z tematem subskrypcji, bezpośrednio wysyłają komunikat do klienta MQTT v3. Aplikacje klienckie produktu MQTT V3 nie mogą wysyłać komunikatów bezpośrednio, ale inne aplikacje, takie jak aplikacje produktu IBM MQ, mogą być wysyłane.

Aplikacja IBM MQ musi znać identyfikator `ClientIdentifier` klienta MQTT v3. Ponieważ klienci MQTT v3 nie mają kolejek, nazwa kolejki docelowej jest przekazywana do metody `messageArrived` klienta aplikacji MQTT v3 jako nazwa tematu. Na przykład w programie MQI utwórz deskryptor obiektu z klientem jako `ObjectQmgrName(Nazwa menedżera obiektów)`:

```
MQOD.ObjectQmgrName = ClientIdentifier ;  
MQOD.ObjectName = name ;
```

Rysunek 24. Deskryptor obiektu MQI w celu wysłania komunikatu do miejsca docelowego klienta produktu MQTT v3

Jeśli aplikacja jest napisana przy użyciu programu JMS, utwórz miejsce docelowe typu punkt z punktem, na przykład:

```
javax.jms.Destination jmsDestination =  
(javax.jms.Destination)jmsFactory.createQueue  
("queue://ClientIdentifier/name");
```

Rysunek 25. Miejsce docelowe JMS do wysłania komunikatu do klienta MQTT v3

Aby wysłać niezamówiony komunikat do klienta MQTT, należy użyć definicji kolejki zdalnej. Nazwa zdalnego menedżera kolejek musi zostać przetłumaczona na wartość `ClientIdentifier` klienta. Kolejka transmisji musi być przetłumaczona na `SYSTEM.MQTT.TRANSMIT.QUEUE`; patrz [Tabela 15](#) na stronie 221. Nazwa kolejki zdalnej może być dowolna. Klient otrzymuje go jako łańcuch tematu.

Tabela 15. Rozstrzygnięcie nazwy zdalnej definicji kolejki klienta MQTT

Wejście		Wyjście		
Nazwa kolejki	Nazwa menedżera kolejek	Nazwa kolejki	Nazwa menedżera kolejek	Kolejka transmisji
Nazwa definicji kolejki zdalnej	Pusta lub lokalna nazwa menedżera kolejek	Nazwa kolejki zdalnej używana jako łańcuch tematu	<i>ClientIdentifier</i>	SYSTEM.MQTT.TRANSMIT.QUEUE

Jeśli klient jest połączony, komunikat jest wysyłany bezpośrednio do klienta MQTT, który wywołuje metodę `messageArrived`. Patrz sekcja [Metoda messageArrived](#).

Jeśli klient rozłączył się z trwałą sesją, komunikat jest zapisywany w produkcie `SYSTEM.MQTT.TRANSMIT.QUEUE`. patrz [MQTT stateless and stateful sessions](#). Jest on przekazywany do klienta po ponownym nawiązaniu połączenia przez klienta z sesją.

Jeśli wysyłany jest komunikat nietrwały, jest on wysyłany do klienta z co najwyżej raz jakością usługi `QoS=0`. Jeśli komunikat trwały jest wysyłany bezpośrednio do klienta, domyślnie jest on wysyłany z dokładnie co najmniej raz jakością usługi `QoS=2`. Ponieważ klient może nie mieć mechanizmu trwałości, klient może zmniejszyć jakość usługi, którą akceptuje w przypadku komunikatów wysyłanych bezpośrednio. Aby zmniejszyć jakość usługi w przypadku komunikatów wysyłanych bezpośrednio do klienta, należy wykonać subskrypcję tematu `DEFAULT`. `QoS`. Określ maksymalną jakość usługi, którą może obsługiwać klient.

Windows

Linux

AIX

MQTT - identyfikacja, autoryzacja

i uwierzytelnianie klienta

Usługa telemetryczna (MQXR) publikuje lub subskrybuje tematy produktu IBM MQ w imieniu klientów MQTT za pomocą kanałów produktu MQTT. Administrator produktu IBM MQ konfiguruje tożsamość kanału produktu MQTT, która jest używana na potrzeby autoryzacji produktu IBM MQ. Administrator może zdefiniować wspólną tożsamość dla kanału lub użyć wartości `Username` lub `ClientIdentifier` klienta połączonego z kanałem.

Usługa telemetryczna (MQXR) może uwierzytelnić klienta przy użyciu nazwy użytkownika dostarczonej przez klienta lub za pomocą certyfikatu klienta. Nazwa użytkownika jest uwierzytelniana przy użyciu hasła udostępnionego przez klienta.

Podsumowując: Identyfikacja klienta to wybór tożsamości klienta. W zależności od kontekstu klient jest identyfikowany przez `ClientIdentifier` (identyfikator klienta), `Username` (Nazwa użytkownika), wspólną tożsamość klienta utworzoną przez administratora lub certyfikat klienta. Identyfikator klienta używany do sprawdzania autentyczności nie musi być tym samym identyfikatorem, który jest używany do autoryzacji.

Programy klienckie MQTT ustawiają wartości `Nazwa użytkownika` i `Hasło`, które są wysyłane do serwera przy użyciu kanału MQTT. Mogą także ustawiać właściwości TLS, które są wymagane do szyfrowania i uwierzytelniania połączenia. Administrator decyduje o tym, czy uwierzytelnić kanał MQTT, a także jak uwierzytelnić kanał.

Aby autoryzować klienta MQTT do uzyskiwania dostępu do obiektów IBM MQ, autoryzuj `ClientIdentifier` lub `Username` klienta albo autoryzuj wspólną tożsamość klienta. Aby umożliwić klientowi nawiązanie połączenia z serwerem IBM MQ, należy uwierzytelnić nazwę użytkownika `Nazwa użytkownika` lub użyć certyfikatu klienta. Skonfiguruj usługę JAAS w celu uwierzytelnienia nazwy użytkownika i skonfiguruj protokół TLS w celu uwierzytelnienia certyfikatu klienta.

Jeśli w kliencie zostanie ustawiona wartość `Hasło`, należy zaszyfrować połączenie przy użyciu sieci VPN lub skonfigurować kanał produktu MQTT tak, aby używał protokołu TLS, aby zachować hasło prywatne.

Zarządzanie certyfikatami klientów jest trudne. Z tego powodu, jeśli ryzyko związane z uwierzytelnianiem hasłem jest akceptowalne, uwierzytelnianie za pomocą hasła jest często używane do uwierzytelniania klientów.

Jeśli istnieje bezpieczny sposób zarządzania i przechowywania certyfikatu klienta, możliwe jest oparcie się na uwierzytelnianiu certyfikatów. Rzadko jednak zdarza się, że certyfikaty mogą być bezpiecznie zarządzane w różnych środowiskach, w których używana jest telemetria. Zamiast tego uwierzytelnianie urządzeń za pomocą certyfikatów klienta jest uzupełniane przez uwierzytelnianie haseł klientów na serwerze. Ze względu na dodatkową złożoność korzystanie z certyfikatów klientów jest ograniczone do wysoce wrażliwych aplikacji. Użycie dwóch form uwierzytelniania jest nazywane uwierzytelnianiem dwuskładniowym. Użytkownik musi znać jeden z czynników, na przykład hasło, i mieć inne, takie jak certyfikat.

W wysoce czułych aplikacjach, takich jak urządzenie chipowe i przypinane, urządzenie jest zamykane w trakcie produkcji, aby zapobiec ingerowaniu w wewnętrzny sprzęt i oprogramowanie. Zaufany, ograniczony czas, certyfikat klienta jest kopiowany do urządzenia. Urządzenie jest wdrażane w miejscu, w którym ma być używane. Dalsze uwierzytelnianie jest wykonywane za każdym razem, gdy urządzenie jest używane, albo przy użyciu hasła, albo na innym certyfikacie z karty Smart Card.

Windows

Linux

AIX

Tożsamość i autoryzacja klienta MQTT

Aby uzyskać dostęp do obiektów produktu IBM MQ, należy użyć identyfikatora klienta, nazwy użytkownika lub wspólnej tożsamości klienta.

Administrator produktu IBM MQ ma trzy opcje wyboru tożsamości kanału produktu MQTT. Administrator dokonuje wyboru podczas definiowania lub modyfikowania kanału MQTT używanego przez klienta. Tożsamość jest używana do autoryzowania dostępu do tematów produktu IBM MQ. Wybór dokonany jest w następującej kolejności:

1. Identyfikator klienta (patrz [USECLNTID](#)).
2. Tożsamość, którą administrator udostępnia dla kanału (MCAUSER kanału. Patrz [MCAUSER](#)).
3. Jeśli nie ma zastosowania żadne z poprzednich wyborów, Nazwa użytkownika przekazana z klienta MQTT (Nazwa użytkownika jest atrybutem klasy `MqttConnectOptions`). Musi on zostać ustawiony przed nawiązaniem połączenia z usługą. Jego wartością domyślną jest NULL).

Unikaj problemów: Tożsamość wybrana przez ten proces jest następnie przywołana, na przykład przez komendę `DISPLAY CHSTATUS (MQTT)`, jako użytkownik MCAUSER klienta. Należy pamiętać, że nie jest to koniecznie ta sama tożsamość, co MCAUSER kanału, o którym mowa w wyborze (2).

Komenda IBM MQ `setmqaut` służy do wybierania obiektów, które są autoryzowane do użycia przez tożsamość powiązaną z kanałem MQTT, i które czynności mają być używane. Na przykład następujący kod autoryzuje tożsamość kanału `MQTTClient` udostępnionego przez administratora menedżera kolejek QM1:

```
setmqaut -m QM1 -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p MQTTClient -all +put
setmqaut -m QM1 -t topic -n SYSTEM.BASE.TOPIC -p MQTTClient -all +pub +sub
```

Windows

Linux

AIX

Autoryzowanie klientów MQTT do uzyskiwania

dostępu do obiektów IBM MQ

Aby autoryzować klienty produktu MQTT do publikowania i subskrybowania obiektów produktu IBM MQ, należy wykonać poniższe kroki. Kroki są zgodne z czterema alternatywnymi wzorcami kontroli dostępu.

Zanim rozpoczniesz

Klienty MQTT są autoryzowane do uzyskiwania dostępu do obiektów w produkcie IBM MQ przez przypisanie tożsamości podczas łączenia się z kanałem pomiarowym. Administrator produktu IBM MQ konfiguruje kanał pomiarowy przy użyciu programu IBM MQ Explorer w celu nadania klientowi jednego z trzech typów tożsamości:

1. `ClientIdentifier`
2. Nazwa użytkownika
3. Nazwa przypisana przez administratora do kanału.

Niezależnie od tego, który typ jest używany, tożsamość musi być zdefiniowana jako nazwa użytkownika IBM MQ jako główna przez zainstalowaną usługę autoryzacji. Domyślna usługa autoryzacji w systemie Windows lub Linux jest nazywana menedżerem uprawnień do obiektów (Object Authority Manager-OAM). Jeśli korzystasz z OAM, tożsamość musi być zdefiniowana jako identyfikator użytkownika.

Użyj tożsamości, aby nadać klientowi lub kolekcji klientów uprawnienia do publikowania lub subskrybowania tematów zdefiniowanych w produkcie IBM MQ. Jeśli klient MQTT zasubskrybował dany temat, należy użyć tożsamości, aby nadać mu uprawnienia do otrzymywania wyników publikacji.

Trudno jest zarządzać systemem z dziesiątkami tysięcy klientów MQTT, z których każdy wymaga indywidualnych uprawnień dostępu. Jednym z rozwiązań jest zdefiniowanie wspólnych tożsamości i powiązanie poszczególnych klientów MQTT z jedną ze wspólnych tożsamości. Zdefiniuj tyle wspólnych tożsamości, ile potrzeba, aby zdefiniować różne kombinacje uprawnień. Innym rozwiązaniem jest napisanie własnej usługi autoryzacji, która może być łatwiejsza w obsłudze z tysiącami użytkowników niż system operacyjny.

Klientów produktu MQTT można łączyć ze wspólnymi tożsamościami na dwa sposoby, korzystając z OAM:

1. Zdefiniuj wiele kanałów pomiarowych, z których każdy ma inny identyfikator użytkownika, który administrator przydziela za pomocą programu IBM MQ Explorer. Klienci łączące się z różnymi numerami portów TCP/IP są powiązane z różnymi kanałami telemetrycznymi i przypisują różne tożsamości.
2. Zdefiniuj pojedynczy kanał pomiarowy, ale dla każdego klienta wybierz Nazwę użytkownika z niewielkiego zestawu identyfikatorów użytkowników. Administrator konfiguruje kanał pomiarowy w celu wybrania klienta Nazwa użytkownika jako jego tożsamości.

W tym zadaniu tożsamość kanału telemetrycznego jest nazywana *mqttUser*, niezależnie od tego, jak została ustawiona. Jeśli kolekcje klientów korzystają z różnych tożsamości, należy użyć wielu *mqttUsers*, po jednej dla każdej kolekcji klientów. Ponieważ zadanie korzysta z usługi OAM, każdy *mqttUser* musi być identyfikatorem użytkownika.

O tym zadaniu

W ramach tego zadania użytkownik ma do wyboru cztery wzorce kontroli dostępu, które można dostosować do konkretnych wymagań. Wzorce różnią się granulacją kontroli dostępu.

- [“Brak kontroli dostępu” na stronie 223](#)
- [“Kontrola dostępu gruboziarnistego” na stronie 223](#)
- [“Średnioziarnista kontrola dostępu” na stronie 224](#)
- [“Precyzyjna kontrola dostępu” na stronie 224](#)

Wynikiem tych modeli jest przypisanie do produktu *mqttUsers* zestawów uprawnień do publikowania i subskrybowania produktu IBM MQ, a także otrzymywania publikacji z produktu IBM MQ.

Brak kontroli dostępu

Klienci MQTT mają nadane uprawnienia administracyjne produktu IBM MQ i mogą wykonywać dowolne działania na dowolnym obiekcie.

Procedura

1. Utwórz identyfikator użytkownika *mqttUser*, który będzie działał jako tożsamość wszystkich klientów MQTT.
2. Dodaj program *mqttUser* do grupy *mqm*. Patrz sekcja [Dodawanie użytkownika do grupy w systemie Windows](#) lub [Dodawanie użytkownika do grupy w systemie Linux](#).

Kontrola dostępu gruboziarnistego

Klienci MQTT mają uprawnienia do publikowania i subskrybowania oraz wysyłania komunikatów do klientów MQTT. Nie mają uprawnień do wykonywania innych działań lub uzyskiwania dostępu do innych obiektów.

Procedura

1. Utwórz identyfikator użytkownika *mqttUser* , który będzie działał jako tożsamość wszystkich klientów MQTT .
2. Autoryzowanie produktu *mqttUser* do publikowania i subskrybowania wszystkich tematów oraz wysyłanie publikacji do klientów MQTT .

```
setmqaut -m qMgr -t topic -n SYSTEM.BASE.TOPIC -p mqttUser -all +pub +sub
setmqaut -m qMgr -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p mqttUser -all +put
```

Średnioziarnista kontrola dostępu

Klienci MQTT są podzielone na różne grupy w celu publikowania i subskrybowania różnych zestawów tematów oraz wysyłania komunikatów do klientów MQTT .

Procedura

1. W drzewie tematów publikowania/subskrypcji utwórz wiele identyfikatorów użytkowników, *mqttUsers* i wiele tematów administracyjnych.
2. Autoryzuj różne *mqttUsers* do różnych tematów.

```
setmqaut -m qMgr -t topic -n topic1 -p mqttUserA -all +pub +sub
setmqaut -m qMgr -t topic -n topic2 -p mqttUserB -all +pub +sub
```

3. Utwórz grupę *mqtt* dodaj wszystkie *mqttUsers* do grupy.
4. Autoryzuj *mqtt* do wysyłania tematów do klientów MQTT .

```
setmqaut -m qMgr -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p mqtt -all +put
```

Precyzyjna kontrola dostępu

Klienci MQTT są wbudowane w istniejący system kontroli dostępu, który autoryzuje grupy do wykonywania działań na obiektach.

O tym zadaniu

Identyfikator użytkownika jest przypisany do jednej lub większej liczby grup systemów operacyjnych, w zależności od wymaganych przez niego autoryzacji. Jeśli aplikacje produktu IBM MQ publikują i subskrybują ten sam obszar tematu co klienci MQTT , należy użyć tego modelu. Grupy są określane jako Publish X, Subscribe Y i mqtt .

Publish X

Członkowie grup produktu Publish X mogą publikować w produkcie *topicX*.

Subscribe Y

Członkowie grup produktu Subscribe Y mogą subskrybować produkt *topicY*.

mqtt

Członkowie grupy *mqtt* mogą wysyłać publikacje do klientów MQTT .

Procedura

1. Utwórz wiele grup, Publish X i Subscribe Y , które są przydzielane do wielu tematów administracyjnych w drzewie tematów publikowania/subskrypcji.
2. Utwórz grupę mqtt.
3. Utwórz wiele identyfikatorów użytkowników, *mqttUsers* i dodaj użytkowników do dowolnej z grup, w zależności od tego, do czego są uprawnieni.
4. Autoryzuj różne grupy produktów Publish X i Subscribe X do różnych tematów, a następnie autoryzuj grupę *mqtt* do wysyłania komunikatów do klientów MQTT .


```
setmqaut -m qMgr -t topic -n topic1 -p Publish X -all +pub
setmqaut -m qMgr -t topic -n topic1 -p Subscribe X -all +pub +sub
setmqaut -m qMgr -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p mqtt -all +put
```

Windows

Linux

AIX

Uwierzytelnianie klienta MQTT przy użyciu

hasła

Uwierzytelniaj nazwę użytkownika przy użyciu hasła klienta. Istnieje możliwość uwierzytelnienia klienta przy użyciu innej tożsamości do tożsamości używanej do autoryzowania klienta do publikowania i subskrybowania tematów.

Usługa telemetryczna (MQXR) korzysta z usługi JAAS w celu uwierzytelnienia klienta Nazwa użytkownika. Usługa JAAS korzysta z hasła dostarczonego przez klienta MQTT .

Administrator produktu IBM MQ podejmuje decyzję, czy uwierzytelnić nazwę użytkownika, czy nie w ogóle uwierzytelnić, konfigurując kanał MQTT , z którym łączy się klient. Klienci mogą być przypisani do różnych kanałów, a każdy kanał może być skonfigurowany do uwierzytelniania swoich klientów na różne sposoby. Za pomocą usługi JAAS można skonfigurować, które metody muszą uwierzytelniać klienta, i które opcjonalnie mogą uwierzytelnić klienta.

Wybór tożsamości do uwierzytelniania nie ma wpływu na wybór tożsamości do autoryzacji. Użytkownik może chcieć skonfigurować wspólną tożsamość w celu autoryzacji dla wygody administratora, ale uwierzytelnia każdego użytkownika w celu użycia tej tożsamości. Poniższa procedura przedstawia kroki w celu uwierzytelnienia poszczególnych użytkowników w celu użycia wspólnej tożsamości:

1. Administrator produktu IBM MQ ustawia tożsamość kanału produktu MQTT na dowolną nazwę, taką jak MQTTClientUser, za pomocą programu IBM MQ Explorer.
2. Administrator produktu IBM MQ autoryzuje produkt MQTTClient do publikowania i subskrybowania dowolnego tematu:

```
setmqaut -m QM1 -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p MQTTClient -all +put
setmqaut -m QM1 -t topic -n SYSTEM.BASE.TOPIC -p MQTTClient -all +pub +sub
```

3. Programista aplikacji klienckiej produktu MQTT tworzy obiekt MqttConnectOptions i ustawia wartości Username i Password przed nawiązaniem połączenia z serwerem.
4. Programista zabezpieczeń tworzy usługę JAAS LoginModule w celu uwierzytelnienia nazwy użytkownika przy użyciu hasła i dołącza ją do pliku konfiguracyjnego JAAS .
5. Administrator produktu IBM MQ konfiguruje kanał MQTT do uwierzytelniania Username klienta przy użyciu usługi JAAS.

Windows

Linux

AIX

Uwierzytelnianie klienta MQTT przy użyciu

protokołu TLS

Połączenia między klientem MQTT a menedżerem kolejek są zawsze inicjowane przez klient MQTT. Klient MQTT jest zawsze klientem SSL. Uwierzytelnianie klienta serwera i uwierzytelnianie serwera klienta MQTT są opcjonalne.

Dzięki udostępnieniu klienta z prywatnym podpisanym certyfikatem cyfrowym można uwierzytelnić klienta MQTT w produkcie WebSphere MQ. Administrator produktu WebSphere MQ może wymusić uwierzytelnianie klientów MQTT w menedżerze kolejek przy użyciu protokołu TLS. Tylko w ramach uwierzytelniania wzajemnego można zażądać uwierzytelnienia klienta.

Alternatywę dla protokołu SSL stanowią niektóre rodzaje wirtualnej sieci prywatnej (VPN), takie jak IPsec, uwierzytelniające punkty końcowe połączenia TCP/IP. Sieć VPN szyfruje każdy pakiet IP przepływający przez sieć. Po ustanowieniu połączenia VPN uzyskana sieć jest zaufana. Klienci MQTT można połączyć z kanałami pomiarowymi przy użyciu połączeń TCP/IP za pośrednictwem sieci VPN.

Uwierzytelnianie klienta przy użyciu protokołu TLS opiera się na założeniu, że klient ma klucz tajny. Klucz tajny to klucz prywatny klienta stosowany na potrzeby certyfikatu samopodpisanego lub klucz

udostępniony przez ośrodek certyfikacji. Ten klucz służy do podpisywania certyfikatu cyfrowego klienta. Odpowiedni klucz publiczny umożliwia weryfikację certyfikatu cyfrowego. Certyfikaty mogą być zaufanymi certyfikatami. Jeśli są połączone w łańcuch, można prześledzić łańcuch certyfikatów aż do zaufanego certyfikatu głównego. Podczas weryfikacji klienta wszystkie certyfikaty tworzące łańcuch certyfikatów udostępnione przez klienta są wysyłane na serwer. Serwer sprawdza łańcuch certyfikatów, aż znajdzie zaufany certyfikat. Zaufany certyfikat to certyfikat publiczny wygenerowany na podstawie certyfikatu samopodpisanego lub certyfikat główny wydany (zwykle) przez ośrodek certyfikacji. W ramach opcjonalnego ostatniego kroku zaufany certyfikat można porównać z aktualną listą odwołań certyfikatów.

Zaufanym certyfikatem może być certyfikat wystawiony przez ośrodek certyfikacji, który został już uwzględniony w bazie certyfikatów środowiska JRE. Może to być certyfikat samopodpisany lub dowolny certyfikat, który został dodany do magazynu kluczy kanału pomiarowego jako zaufany certyfikat.

Uwaga: Kanał pomiarowy ma połączony magazyn kluczy / magazyn zaufanych certyfikatów, w którym przechowywane są klucze prywatne do jednego lub większej liczby kanałów pomiarowych oraz wszystkie certyfikaty publiczne wymagane na potrzeby uwierzytelniania klientów. Baza certyfikatów środowiska JRE nigdy nie jest przywoływana, ponieważ kanał SSL musi mieć magazyn kluczy i jest to ten sam plik, który jest używany na potrzeby magazynu zaufanych certyfikatów kanału. Oznacza to, że jeśli uwierzytelnianie klienta wymaga certyfikatu głównego ośrodka CA, należy umieścić certyfikat główny w magazynie kluczy tego kanału nawet wtedy, gdy główny certyfikat ośrodka CA znajduje się już w bazie certyfikatów środowiska JRE. Baza certyfikatów środowiska JRE nigdy nie jest przywoływana.

Należy pomyśleć o zagrożeniach, jakim ma przeciwdziałać uwierzytelnianie klienta, oraz o rolach klienta i serwera w procesie przeciwdziałania tym zagrożeniom. Uwierzytelnienie samego certyfikatu klienta nie blokuje w pełni możliwości dostępu do systemu bez uprawnień. Jeśli inna osoba wejdzie w posiadanie urządzenia z klientem, urządzenie z klientem nie będzie działało z upoważnienia właściciela certyfikatu. Nigdy nie należy polegać na jednej formie obrony przed niepożądanymi atakami. Należy stosować przynajmniej uwierzytelnianie dwuskładnikowe i oprócz posiadania certyfikatu wymagać znajomości prywatnych informacji. Na przykład można zastosować usługę JAAS i uwierzytelniać klient przy użyciu hasła wystawionego przez serwer.

Dostanie się w niepowołane ręce stanowi główne zagrożenie dla certyfikatu klienta. Certyfikat jest przechowywany w kliencie w magazynie kluczy zabezpieczonym hasłem. W jaki sposób jest on umieszczany w magazynie kluczy? W jaki sposób klient MQTT uzyskuje hasło do magazynu kluczy? Jak wysoki poziom bezpieczeństwa zapewnia zabezpieczenie hasłem? Często urządzenie pomiarowe można łatwo odłączyć. W takiej sytuacji haker może włamać się do urządzenia w miejscu prywatnym. Czy sprzęt urządzenia musi być zabezpieczony przed manipulacją? Uznaje się, że rozpowszechnianie i ochrona certyfikatów po stronie klienta to trudne zadanie – jest to tzw. problem z zarządzaniem kluczami.

Dodatkowe zagrożenie stanowi nieprawidłowe używanie urządzenia w celu uzyskiwania dostępu do serwerów w niezamierzony sposób. Na przykład zmanipulowanie aplikacji MQTT umożliwia wykorzystanie luki w konfiguracji serwera przy użyciu uwierzytelnionej tożsamości klienta.

Aby móc uwierzytelniać klienta MQTT przy użyciu protokołu SSL, należy skonfigurować kanał pomiarowy i klient.

Pojęcia pokrewne

[“Konfiguracja kanału telemetrycznego dla uwierzytelniania klienta MQTT przy użyciu protokołu TLS” na stronie 226](#)

Administrator produktu IBM MQ konfiguruje kanały telemetryczne na serwerze. Każdy kanał jest konfigurowany tak, aby akceptował połączenie TCP/IP na innym porcie. Kanały TLS są konfigurowane tak, aby dostęp do plików kluczy był zabezpieczony hasłem. Jeśli kanał TLS zostanie zdefiniowany bez hasła lub pliku kluczy, kanał nie będzie akceptował połączeń TLS.

Informacje pokrewne

[Konfiguracja klienta MQTT na potrzeby uwierzytelniania klienta przy użyciu protokołu TLS](#)

Konfiguracja kanału telemetrycznego dla uwierzytelniania klienta MQTT przy użyciu protokołu TLS

Administrator produktu IBM MQ konfiguruje kanały telemetryczne na serwerze. Każdy kanał jest konfigurowany tak, aby akceptował połączenie TCP/IP na innym porcie. Kanały TLS są konfigurowane

tak, aby dostęp do plików kluczy był zabezpieczony hasłem. Jeśli kanał TLS zostanie zdefiniowany bez hasła lub pliku kluczy, kanał nie będzie akceptował połączeń TLS.

Ustaw właściwość `com.ibm.mq.MQTT.ClientAuth` kanału pomiarowego TLS na `REQUIRED`, aby wymusić na wszystkich klientach łączących się z tym kanałem w celu dostarczenia dowodów na to, że zweryfikowano certyfikaty cyfrowe. Certyfikaty klienta są uwierzytelniane przy użyciu certyfikatów z ośrodków certyfikacji, co prowadzi do zaufanego certyfikatu głównego. Jeśli certyfikat klienta jest samopodpisany lub podpisany przez certyfikat, który pochodzi z ośrodka certyfikacji, publicznie podpisane certyfikaty klienta lub ośrodka certyfikacji muszą być przechowywane w sposób bezpieczny na serwerze.

Umieść publicznie podpisany certyfikat klienta lub certyfikat z ośrodka certyfikacji w magazynie kluczy kanału telemetrycznego. Na serwerze podpisane publicznie certyfikaty są przechowywane w tym samym pliku kluczy, co certyfikaty podpisane prywatnie, a nie w osobnym magazynie zaufanych certyfikatów.

Serwer weryfikuje podpisy wszystkich certyfikatów klientów, które są wysyłane przy użyciu wszystkich certyfikatów publicznych i zestawów algorytmów szyfrowania, które ma. Serwer weryfikuje łańcuch kluczy. Menedżer kolejek można skonfigurować w taki sposób, aby testował certyfikat na liście odwołań certyfikatów. Właściwość listy nazw odwołań do menedżera kolejek to `SSLCRLNL`.

Jeśli którekolwiek z certyfikatów wysyła klient jest weryfikowane przez certyfikat w magazynie kluczy serwera, klient zostanie uwierzytelniony.

Administrator produktu IBM MQ może skonfigurować ten sam kanał pomiarowy, aby mógł używać usługi JAAS w celu sprawdzenia wartości `UserName` lub `ClientIdentifier` klienta przy użyciu klienta `Password`.

Tego samego magazynu kluczy można użyć dla wielu kanałów pomiarowych.

Weryfikacja co najmniej jednego certyfikatu cyfrowego w magazynie kluczy klienta zabezpieczonego hasłem na urządzeniu uwierzytelnia klienta na serwerze. Certyfikat cyfrowy jest używany tylko do uwierzytelniania przez produkt IBM MQ. Nie jest on używany do weryfikowania adresu TCP/IP klienta ani do ustawiania tożsamości klienta w celu autoryzacji lub rozliczania. Tożsamość klienta adoptowane przez serwer to Nazwa użytkownika lub `ClientIdentifier` klienta lub tożsamość utworzona przez administratora produktu IBM MQ.

Do uwierzytelniania klientów można również używać zestawów algorytmów szyfrowania TLS. Jeśli planowane jest użycie zestawów algorytmów szyfrowania SHA-2, należy zapoznać się z [“Wymagania systemowe dotyczące korzystania z zestawów algorytmów szyfrowania SHA-2 z kanałami produktu MQTT”](#) na stronie 229.

Pojęcia pokrewne

[“Konfiguracja kanału telemetrycznego do uwierzytelniania kanału przy użyciu protokołu TLS”](#) na stronie 228

Administrator produktu IBM MQ konfiguruje kanały telemetryczne na serwerze. Każdy kanał jest konfigurowany tak, aby akceptował połączenie TCP/IP na innym porcie. Kanały TLS są konfigurowane tak, aby dostęp do plików kluczy był zabezpieczony hasłem. Jeśli kanał TLS zostanie zdefiniowany bez hasła lub pliku kluczy, kanał nie będzie akceptował połączeń TLS.

Informacje pokrewne

[ZDEFINIUJ KANAŁ \(MQTT\)](#)

[ZMIENŃ KANAŁ \(MQTT\)](#)

[CipherSpecs i CipherSuites](#)

Uwierzytelnianie kanału telemetrycznego przy użyciu protokołu TLS

Połączenia między klientem MQTT a menedżerem kolejek są zawsze inicjowane przez klient MQTT. Klient MQTT jest zawsze klientem SSL. Uwierzytelnianie klienta serwera i uwierzytelnianie serwera klienta MQTT są opcjonalne.

Klient zawsze będzie próbował uwierzytelnić serwer, chyba że klient został skonfigurowany pod kątem używania specyfikacji szyfrowania obsługującej połączenia anonimowe. Jeśli uwierzytelnianie nie powiedzie się, połączenie nie zostanie nawiązane.

Alternatywę dla protokołu SSL stanowią niektóre rodzaje wirtualnej sieci prywatnej (VPN), takie jak IPsec, uwierzytelniające punkty końcowe połączenia TCP/IP. Sieć VPN szyfruje każdy pakiet IP przepływający przez sieć. Po ustanowieniu połączenia VPN uzyskana sieć jest zaufana. Klienci MQTT można połączyć z kanałami pomiarowymi przy użyciu połączeń TCP/IP za pośrednictwem sieci VPN.

Uwierzytelnianie serwera za pomocą protokołu SSL uwierzytelnia serwer, do którego użytkownik chce wysłać informacje poufne. Klient wykonuje sprawdzenia zgodne z certyfikatami wysłanym z serwera, z certyfikatami umieszczonym w magazynie zaufanych certyfikatów lub w jego składnicy JRE cacerts .

Składnica certyfikatów środowiska JRE to plik JKS cacerts. Znajduje się on w katalogu JRE InstallPath\lib\security\. Jest on instalowany z domyślnym hasłem changeit. Zaufane certyfikaty mogą być przechowywane w bazie certyfikatów środowiska JRE lub w magazynie zaufanych certyfikatów klienta. Nie można używać obu składnic. Należy skorzystać z magazynu zaufanych certyfikatów, jeśli certyfikaty publiczne i zaufane certyfikaty klienta mają być przechowywane w innym miejscu niż certyfikaty używane przez inne aplikacje Java. Należy skorzystać z bazy certyfikatów środowiska JRE, jeśli wszystkie aplikacje Java działające na kliencie mają używać wspólnej bazy certyfikatów. W przypadku wybrania bazy certyfikatów środowiska JRE należy przejrzeć certyfikaty w niej zawarte, aby upewnić się, że są to zaufane certyfikaty.

Konfigurację JSSE można zmodyfikować, podając innego dostawcę zaufania. Istnieje możliwość dostosowania dostawcy zaufania tak, aby wykonywał inne operacje sprawdzania certyfikatu. W niektórych środowiskach OSGi, które korzystały z klienta MQTT, środowisko udostępnia innego dostawcę zaufania.

Aby móc uwierzytelniać kanał pomiarowy przy użyciu protokołu TLS, należy skonfigurować serwer i klient.

Windows Linux AIX Konfiguracja kanału telemetrycznego do uwierzytelniania kanału przy użyciu protokołu TLS

Administrator produktu IBM MQ konfiguruje kanały telemetryczne na serwerze. Każdy kanał jest konfigurowany tak, aby akceptował połączenie TCP/IP na innym porcie. Kanały TLS są konfigurowane tak, aby dostęp do plików kluczy był zabezpieczony hasłem. Jeśli kanał TLS zostanie zdefiniowany bez hasła lub pliku kluczy, kanał nie będzie akceptował połączeń TLS.

Zapisz certyfikat cyfrowy serwera, podpisany za pomocą jego klucza prywatnego, w magazynie kluczy, który ma być używany przez kanał pomiarowy na serwerze. Jeśli chcesz przestać łączyć kluczy do klienta, zapisz wszystkie certyfikaty w jego łańcuchu kluczy w magazynie kluczy. Skonfiguruj kanał pomiarowy przy użyciu programu IBM MQ Explorer, aby używać protokołu TLS. Podaj ścieżkę do magazynu kluczy i frazę hasła, aby uzyskać dostęp do magazynu kluczy. Jeśli nie zostanie ustawiony numer portu TCP/IP kanału, numer portu kanału pomiarowego TLS zostanie domyślnie ustawiony na 8883.

Do uwierzytelniania kanału można również używać zestawów algorytmów szyfrowania TLS. Jeśli planowane jest użycie zestawów algorytmów szyfrowania SHA-2 , należy zapoznać się z [“Wymagania systemowe dotyczące korzystania z zestawów algorytmów szyfrowania SHA-2 z kanałami produktu MQTT”](#) na stronie 229.

Pojęcia pokrewne

[“Konfiguracja kanału telemetrycznego dla uwierzytelniania klienta MQTT przy użyciu protokołu TLS”](#) na stronie 226

Administrator produktu IBM MQ konfiguruje kanały telemetryczne na serwerze. Każdy kanał jest konfigurowany tak, aby akceptował połączenie TCP/IP na innym porcie. Kanały TLS są konfigurowane tak, aby dostęp do plików kluczy był zabezpieczony hasłem. Jeśli kanał TLS zostanie zdefiniowany bez hasła lub pliku kluczy, kanał nie będzie akceptował połączeń TLS.

Informacje pokrewne

[ZDEFINIUJ KANAŁ \(MQTT\)](#)

[ZMIENŃ KANAŁ \(MQTT\)](#)

Wymagania systemowe dotyczące korzystania z zestawów algorytmów szyfrowania SHA-2 z kanałami produktu MQTT

Jeśli używana jest wersja produktu Java, która obsługuje zestawy algorytmów szyfrowania SHA-2, można użyć tych pakietów w celu zabezpieczenia kanałów i aplikacji klienckich produktu MQTT (telemetria).

W przypadku produktu IBM MQ 8.0, który zawiera usługę telemetryczną (MQXR), minimalna wersja produktu Java to Java 7 z produktu IBM, SR6. Zestawy algorytmów szyfrowania SHA-2 są obsługiwane domyślnie w produkcie Java 7 z wersji IBM, SR4. W związku z tym można użyć zestawów algorytmów szyfrowania SHA-2 z usługą telemetryczną (MQXR) w celu zabezpieczenia kanałów produktu MQTT (telemetrycznych).

Jeśli używany jest klient MQTT z innym środowiskiem JRE, należy upewnić się, że obsługuje on również zestawy algorytmów szyfrowania SHA-2.

Pojęcia pokrewne

[“Konfiguracja kanału telemetrycznego do uwierzytelniania kanału przy użyciu protokołu TLS” na stronie 228](#)

Administrator produktu IBM MQ konfiguruje kanały telemetryczne na serwerze. Każdy kanał jest konfigurowany tak, aby akceptował połączenie TCP/IP na innym porcie. Kanały TLS są konfigurowane tak, aby dostęp do plików kluczy był zabezpieczony hasłem. Jeśli kanał TLS zostanie zdefiniowany bez hasła lub pliku kluczy, kanał nie będzie akceptował połączeń TLS.

Informacje pokrewne

[Usługa telemetryczna \(MQXR\)](#)

[ZDEFINIUJ KANAŁ \(MQTT\)](#)

[ZMIEN KANAŁ \(MQTT\)](#)

Prywatność publikacji na kanałach telemetrycznych

Prywatność publikacji MQTT wysyłanych w obu kierunkach w kanałach telemetrycznych jest zabezpieczona za pomocą protokołu TLS w celu szyfrowania transmisji przez połączenie.

Klienci MQTT, które łączą się z kanałami telemetrycznymi używają protokołu TLS do zabezpieczania prywatności publikacji przesyłanych w kanale za pomocą kryptografii klucza symetrycznego. Ponieważ punkty końcowe nie są uwierzytelniane, szyfrowanie kanału nie jest możliwe tylko w przypadku szyfrowania. Połączenie zabezpieczania prywatności z serwerem lub uwierzytelniania wzajemnego.

Alternatywę dla protokołu SSL stanowią niektóre rodzaje wirtualnej sieci prywatnej (VPN), takie jak IPsec, uwierzytelniające punkty końcowe połączenia TCP/IP. Sieć VPN szyfruje każdy pakiet IP przepływający przez sieć. Po ustanowieniu połączenia VPN uzyskana sieć jest zaufana. Klienci MQTT można połączyć z kanałami pomiarowymi przy użyciu połączeń TCP/IP za pośrednictwem sieci VPN.

W przypadku typowej konfiguracji, która szyfruje kanał i uwierzytelnia serwer, należy zapoznać się z [“Uwierzytelnianie kanału telemetrycznego przy użyciu protokołu TLS” na stronie 227](#).

Szyfrowanie połączeń TLS bez uwierzytelniania serwera naraża połączenie z atakami typu man-in-the-middle. Mimo, że informacje, które wymieniasz, są chronione przed podsłuchami, nie wiesz, z kim się wymieniasz. Jeśli sieć nie jest sterowana, użytkownik jest narażony na przechwytywanie transmisji danych IP i maskowanie jako punkt końcowy.

Szyfrowane połączenie TLS można utworzyć bez uwierzytelniania serwera, korzystając z wymiany kluczy Diffie-Hellman CipherSpec, która obsługuje anonimowy protokół TLS. Główny klucz tajny, współużytkowany między klientem i serwerem i używany do szyfrowania transmisji TLS, jest ustanawiany bez wymiany certyfikatu serwera prywatnego.

Ponieważ anonimowe połączenia są niezabezpieczone, większość implementacji protokołu TLS nie jest domyślnie używana do używania anonimowych specyfikacji CipherSpecs. Jeśli żądanie klienta dotyczące połączenia TLS zostanie zaakceptowane przez kanał pomiarowy, kanał musi mieć magazyn kluczy chroniony hasłem. Domyślnie, ponieważ implementacje protokołu TLS nie używają anonimowych specyfikacji CipherSpecs, magazyn kluczy musi zawierać podpisany przez klienta certyfikat, który może być uwierzytelniony przez klienta.

Jeśli używana jest anonimowa opcja CipherSpecs, magazyn kluczy serwera musi istnieć, ale nie musi zawierać żadnych prywatnych certyfikatów podpisanych.

Innym sposobem nawiązania szyfrowanego połączenia jest zastąpienie dostawcy zaufanego na kliencie we własnej implementacji. Dostawca zaufania nie uwierzytelniłby certyfikatu serwera, ale połączenie będzie szyfrowane.



Ostrzeżenie: W przypadku korzystania z protokołu TLS z produktem MQTT można używać dużych komunikatów, jednak podczas wykonywania tej czynności może wystąpić potencjalny wpływ na wydajność. Produkt MQTT jest zoptymalizowany pod kątem przetwarzania małych komunikatów (zwykle od 1KB do 1MB).

Windows

Linux

AIX

Konfiguracja TLS dla klientów MQTT Java

i kanałów telemetrycznych

Skonfiguruj protokół TLS do uwierzytelniania kanału pomiarowego i klienta MQTT Java, a następnie zaszyfruj przesyłanie komunikatów między nimi. MQTT Klienci Java używają produktu Java Secure Socket Extension (JSSE) do łączenia kanałów telemetrycznych za pomocą protokołu TLS. Alternatywę dla protokołu SSL stanowią niektóre rodzaje wirtualnej sieci prywatnej (VPN), takie jak IPsec, uwierzytelniające punkty końcowe połączenia TCP/IP. Sieć VPN szyfruje każdy pakiet IP przepływający przez sieć. Po ustanowieniu połączenia VPN uzyskana sieć jest zaufana. Klienci MQTT można połączyć z kanałami pomiarowymi przy użyciu połączeń TCP/IP za pośrednictwem sieci VPN.

Istnieje możliwość skonfigurowania połączenia między klientem Java MQTT a kanałem pomiarowym do korzystania z protokołu TLS przez TCP/IP. To, co jest zabezpieczone, zależy od tego, jak skonfigurować protokół TLS do korzystania z JSSE. Począwszy od konfiguracji najbardziej zabezpieczonej, można skonfigurować trzy różne poziomy zabezpieczeń:

1. Zezwól tylko zaufanym klientom MQTT na połączenie. Połącz klienta MQTT tylko z zaufanym kanałem telemetrycznym. Szyfruj komunikaty między klientem i menedżerem kolejek; patrz [“Uwierzytelnianie klienta MQTT przy użyciu protokołu TLS”](#) na stronie 225
2. Połącz klienta MQTT tylko z zaufanym kanałem telemetrycznym. Szyfruj komunikaty między klientem i menedżerem kolejek; patrz [“Uwierzytelnianie kanału telemetrycznego przy użyciu protokołu TLS”](#) na stronie 227.
3. Szyfruj komunikaty między klientem i menedżerem kolejek; patrz [“Prywatność publikacji na kanałach telemetrycznych”](#) na stronie 229.

Parametry konfiguracyjne JSSE

Zmodyfikuj parametry JSSE w taki sposób, aby zmieniły sposób konfigurowania połączenia TLS. Parametry konfiguracyjne JSSE są rozmieszczone w trzech zestawach:

1. [MQ Telemetry kanał](#)
2. [klientMQTT Java](#)
3. [JRE \(środowisko JRE\)](#)

Skonfiguruj parametry kanału telemetrycznego przy użyciu programu IBM MQ Explorer. Ustaw parametry klienta MQTT Java w atrybucie `MqttConnectionOptions.SSLProperties`. Zmodyfikuj parametry zabezpieczeń środowiska JRE, edytując pliki w katalogu zabezpieczeń środowiska JRE zarówno na kliencie, jak i na serwerze.

MQ Telemetry kanał

Ustaw wszystkie parametry kanału telemetrycznego TLS przy użyciu programu IBM MQ Explorer.

ChannelName

Parametr ChannelName jest wymaganym parametrem we wszystkich kanałach.

Nazwa kanału identyfikuje kanał powiązany z określonym numerem portu. Kanały nazw, które ułatwiają administrowanie zestawami klientów MQTT.

PortNumber

Parametr PortNumber jest opcjonalnym parametrem we wszystkich kanałach. Wartością domyślną jest 1883 dla kanałów TCP, a 8883 dla kanałów TLS.

Numer portu TCP/IP powiązany z tym kanałem. Klienci MQTT są połączone z kanałem przez określenie portu zdefiniowanego dla kanału. Jeśli kanał ma właściwości TLS, klient musi nawiązać połączenie przy użyciu protokołu TLS, na przykład:

```
MQTTClient mqttClient = new MqttClient( "ssl://www.example.org:8884", "clientId");
mqttClient.connect();
```

KeyFileNazwa

Parametr KeyFileName (Nazwa pliku kluczy) jest wymaganym parametrem dla kanałów TLS. Należy go pominąć w przypadku kanałów TCP.

KeyFileNazwa jest ścieżką do magazynu kluczy Java zawierającego certyfikaty cyfrowe, które są dostępne. Użyj JKS, JCEKS lub PKCS12 jako typu magazynu kluczy na serwerze.

Zidentyfikuj typ magazynu kluczy, używając jednego z następujących rozszerzeń:

- .jks
- .jceks
- .p12
- .pkcs12

Zakłada się, że magazyn kluczy z innym rozszerzeniem nazwy pliku jest plikiem kluczy JKS.

Istnieje możliwość połączenia jednego typu magazynu kluczy na serwerze z innymi typami magazynów kluczy na kliencie.

Umieść certyfikat prywatny serwera w magazynie kluczy. Certyfikat jest znany jako certyfikat serwera. Certyfikat może być samopodpisany lub częścią łańcucha certyfikatów, który jest podpisany przez ośrodek podpisujący.

Jeśli używany jest łańcuch certyfikatów, należy umieścić powiązane certyfikaty w magazynie kluczy serwera.

Certyfikat serwera i wszystkie certyfikaty w łańcuchu certyfikatów są wysyłane do klientów w celu uwierzytelnienia tożsamości serwera.

Jeśli parametr ClientAuth został ustawiony na wartość Required, magazyn kluczy musi zawierać wszystkie certyfikaty niezbędne do uwierzytelnienia klienta. Klient wysyła samopodpisany certyfikat lub łańcuch certyfikatów, a klient jest uwierzytelniany przez pierwszą weryfikację tego materiału względem certyfikatu w magazynie kluczy. Korzystając z łańcucha certyfikatów, jeden certyfikat może weryfikować wiele klientów, nawet jeśli są one wystawiane z różnymi certyfikatami klienta.

PassPhrase

Parametr PassPhrase jest wymaganym parametrem dla kanałów TLS. Należy go pominąć w przypadku kanałów TCP.

Fraza hasła używana jest do ochrony magazynu kluczy.

ClientAuth

Parametr `ClientAuth` jest opcjonalnym parametrem TLS. Domyślnie nie jest to uwierzytelnianie klienta. Należy go pominąć w przypadku kanałów TCP.

Ustaw wartość `ClientAuth`, jeśli usługa telemetryczna (MQXR) ma być uwierzytelniana przez klienta, zanim klient będzie mógł nawiązać połączenie z kanałem telemetryczny.

Jeśli zostanie ustawiona wartość `ClientAuth`, klient musi połączyć się z serwerem za pomocą protokołu TLS i uwierzytelnić serwer. W odpowiedzi na ustawienie parametru `ClientAuth` klient wysyła swój certyfikat cyfrowy do serwera i wszystkie inne certyfikaty w magazynie kluczy. Jego certyfikat cyfrowy jest znany jako certyfikat klienta. Certyfikaty te są uwierzytelniane na certyfikatach znajdujących się w magazynie kluczy kanału i w sklepie JRE `cacerts`.

CipherSuite

Parametr `CipherSuite` jest opcjonalnym parametrem TLS. Wartością domyślną jest próba wszystkich włączonych specyfikacji `CipherSpecs`. Należy go pominąć w przypadku kanałów TCP.

Aby użyć konkretnej specyfikacji szyfrowania `CipherSpec`, należy ustawić właściwość `CipherSuite` na nazwę `CipherSpec`, która musi być używana do nawiązywania połączenia TLS.

Usługa telemetryczna i klient MQTT negocjują wspólną specyfikację `CipherSpec` ze wszystkich obiektów `CipherSpecs`, które są włączone na każdym końcu. Jeśli określony parametr `CipherSpec` jest określony w obu lub obu końcach połączenia, musi być zgodny z atrybutem `CipherSpec` na drugim końcu.

Zainstaluj dodatkowe szyfry, dodając dodatkowych dostawców do rozszerzenia JSSE.

Federal Information Processing Standards (FIPS)

Tryb FIPS jest ustawieniem opcjonalnym. Domyślnie nie jest ona ustawiona.

Na panelu właściwości menedżera kolejek lub za pomocą opcji `runmqsc` ustaw wartość `SSLFIPS`. Opcja `SSLFIPS` określa, czy mają być używane tylko algorytmy z certyfikatem FIPS.

Lista nazw odwołań

Lista nazw odwołań jest ustawieniem opcjonalnym. Domyślnie nie jest ona ustawiona.

Na panelu właściwości menedżera kolejek lub za pomocą komendy `runmqsc` ustaw wartość `SSLCRLNL`. `SSLCRLNL` określa listę nazw obiektów informacji uwierzytelniających, które są używane do udostępniania położeń odwołań certyfikatów.

Nie są używane żadne inne parametry menedżera kolejek, które ustawiają właściwości TLS.

Klient portalu MQTTJava

Ustaw właściwości TLS dla klienta Java w katalogu `MqttConnectionOptions.SSLProperties`; na przykład:

```
java.util.Properties sslClientProperties = new Properties();
sslClientProperties.setProperty("com.ibm.ssl.keyStoreType", "JKS");
com.ibm.micro.client.mqttv3.MqttConnectOptions conOptions = new MqttConnectOptions();
conOptions.setSSLProperties(sslClientProperties);
```

Nazwy i wartości konkretnych właściwości są opisane w klasie `MqttConnectOptions`. W celu uzyskania odsyłaczy do dokumentacji interfejsu API klienta dla bibliotek klienta MQTT należy zapoznać się z informacjami w sekcji [Skorowidz programistyczny klienta MQTT](#).

Protokół

Protokół jest opcjonalny.

Protokół jest wybierany w trakcie negocjacji z serwerem telemetryczny. Jeśli wymagany jest konkretny protokół, można wybrać jeden z nich. Jeśli serwer telemetryczny nie obsługuje protokołu, połączenie nie powiedzie się.

ContextProvider

Element ContextProvider jest opcjonalny.

KeyStore

Opcja KeyStore jest opcjonalna. Skonfiguruj ją, jeśli na serwerze ustawiona jest wartość ClientAuth , aby wymusić uwierzytelnianie klienta.

Umieść certyfikat cyfrowy klienta, podpisanego przy użyciu jego klucza prywatnego, do magazynu kluczy. Podaj ścieżkę do magazynu kluczy i hasło. Typ i dostawca są opcjonalne. Domyślnym typem dostawcy jest JKS, a dostawca IBMJCE jest dostawcą domyślnym.

Określ inny dostawca magazynu kluczy, aby odwołał się do klasy, która dodaje nowego dostawcę magazynu kluczy. Przekaz nazwę algorytmu używanego przez dostawcę magazynu kluczy do utworzenia instancji KeyManagerFactory przez ustawienie nazwy menedżera kluczy.

TrustStore

Opcja TrustStore jest opcjonalna. Użytkownik może umieścić wszystkie certyfikaty zaufane w sklepie JRE cacerts .

Skonfiguruj magazyn zaufanych certyfikatów, jeśli chcesz mieć inny magazyn zaufanych certyfikatów dla klienta. Magazyn zaufanych certyfikatów może nie zostać skonfigurowany, jeśli serwer używa certyfikatu wystawionego przez znany ośrodek CA, który ma już certyfikat główny zapisany w produkcie cacerts.

Dodaj do magazynu zaufanych certyfikatów publicznie podpisany certyfikat serwera lub certyfikat główny, a następnie podaj ścieżkę i hasło magazynu zaufanych certyfikatów. Domyślnym typem dostawcy jest JKS, a dostawca IBMJCE jest dostawcą domyślnym.

Określ inny dostawca magazynu zaufanych certyfikatów, aby odwołał się do klasy, która dodaje nowego dostawcę magazynu zaufanych certyfikatów. Przekaz nazwę algorytmu używanego przez dostawcę magazynu zaufanych certyfikatów do utworzenia instancji fabryki TrustManagerFactory przez ustawienie nazwy menedżera zaufania.

JRE (środowisko JRE)

Inne aspekty zabezpieczeń produktu Java , które mają wpływ na działanie protokołu TLS zarówno na kliencie, jak i na serwerze, są konfigurowane w środowisku JRE. Pliki konfiguracyjne w systemie Windows znajdują się w katalogu *Java Installation Directory\jre\lib\security*. Jeśli używane jest środowisko JRE dostarczane z produktem IBM MQ , ścieżka jest wyświetlana w następującej tabeli:

Platforma	Ścieżka do pliku
Windows	<i>WMQ Installation Directory\java\jre\lib\security</i>
UNIX and Linux platformy	<i>WMQ Installation Directory/java/jre64/jre/lib/security</i>

Powszechnie znane uprawnienia do certyfikatów

Plik cacerts zawiera certyfikaty główne powszechnie znanych ośrodków certyfikacji. cacerts jest używany domyślnie, chyba że zostanie określony magazyn zaufanych certyfikatów. Jeśli używany jest sklep cacerts lub magazyn zaufanych certyfikatów nie jest udostępniany, należy przejrzeć i edytować listę osób podpisujących w produkcie cacerts , aby spełnić wymagania bezpieczeństwa.

Produkt cacerts można otworzyć za pomocą komendy IBM MQ `stmqikm`, która uruchamia program narzędziowy do zarządzania kluczami produktu IBM . Otwórz plik cacerts jako plik JKS, korzystając z hasła `changeit`. Zmodyfikuj hasło, aby zabezpieczyć plik.

Konfigurowanie klas zabezpieczeń

Użyj pliku `java.security`, aby zarejestrować dodatkowych dostawców zabezpieczeń i inne domyślne właściwości zabezpieczeń.

Uprawnienia

Użyj pliku `java.policy`, aby zmodyfikować uprawnienia nadane zasobom. `javaws.policy` nadaje uprawnienia do produktu `javaws.jar`

Siła szyfrowania

Niektóre statki JREs z obniżonym szyfrowaniem wytrzymałości. Jeśli nie można zaimportować kluczy do magazynów kluczy, przyczyną może być zredukowane szyfrowanie mocy. Należy spróbować uruchomić produkt **ikerman** przy użyciu komendy **strmqikm** lub pobrać silne, ale ograniczone pliki jurysdykcji z [IBM zestawów deweloperskich](#), informacji o zabezpieczeniach.

Ważne: Twój kraj pochodzenia może mieć ograniczenia w imporcie, posiadaniu, użyciu lub powrotnym wywozie do innego kraju, oprogramowania szyfrującego. Przed pobraniem lub korzystaniem z nieograniczonej strategii należy sprawdzić prawa swojego kraju. Sprawdź jego regulacje, a także jego strategię dotyczące importu, posiadania, używania i ponownego eksportu oprogramowania szyfrującego, w celu ustalenia, czy jest to dozwolone.

Zmodyfikuj zaufany dostawca, aby umożliwić klientowi nawiązanie połączenia z dowolnym serwerem.

W tym przykładzie przedstawiono sposób dodawania dostawcy zaufania i odniesienia go z kodu klienta MQTT. W tym przykładzie nie jest przeprowadzane uwierzytelnianie klienta lub serwera. Wynikowe połączenie TLS jest szyfrowane bez uwierzytelnienia.

Fragment kodu w produkcie Rysunek 26 na stronie 234 ustawia dostawcę zaufanych certyfikatów `AcceptAllProviders` i menedżera zaufania dla klienta MQTT.

```
java.security.Security.addProvider(new AcceptAllProvider());
java.util.Properties sslClientProperties = new Properties();
sslClientProperties.setProperty("com.ibm.ssl.trustManager", "TrustAllCertificates");
sslClientProperties.setProperty("com.ibm.ssl.trustStoreProvider", "AcceptAllProvider");
conOptions.setSSLProperties(sslClientProperties);
```

Rysunek 26. Fragment kodu klienta produktu MQTT

```
package com.ibm.mq.id;
public class AcceptAllProvider extends java.security.Provider {
private static final long serialVersionUID = 1L;
public AcceptAllProvider() {
super("AcceptAllProvider", 1.0, "Trust all X509 certificates");
put("TrustManagerFactory.TrustAllCertificates",
AcceptAllTrustManagerFactory.class.getName());
}
}
```

Rysunek 27. `AcceptAllProvider.java`

```
protected static class AcceptAllTrustManagerFactory extends
javax.net.ssl.TrustManagerFactorySpi {
public AcceptAllTrustManagerFactory() {}
protected void engineInit(java.security.KeyStore keystore) {}
protected void engineInit(
javax.net.ssl.ManagerFactoryParameters parameters) {}
protected javax.net.ssl.TrustManager[] engineGetTrustManagers() {
return new javax.net.ssl.TrustManager[] { new AcceptAllX509TrustManager() };
}
}
```

Rysunek 28. `AcceptAllTrustManagerFactory.java`

```

protected static class AcceptAllX509TrustManager implements
javax.net.ssl.X509TrustManager {
public void checkClientTrusted(
java.security.cert.X509Certificate[] certificateChain,
String authType) throws java.security.cert.CertificateException {
report("Client authtype=" + authType);
for (java.security.cert.X509Certificate certificate : certificateChain) {
report("Accepting:" + certificate);
}
}
public void checkServerTrusted(
java.security.cert.X509Certificate[] certificateChain,
String authType) throws java.security.cert.CertificateException {
report("Server authtype=" + authType);
for (java.security.cert.X509Certificate certificate : certificateChain) {
report("Accepting:" + certificate);
}
}
public java.security.cert.X509Certificate[] getAcceptedIssuers() {
return new java.security.cert.X509Certificate[0];
}
private static void report(String string) {
System.out.println(string);
}
}

```

Rysunek 29. *AcceptAllX509TrustManager.java*

Windows

Linux

AIX

Konfiguracja kanału telemetrycznego JAAS

Skonfiguruj usługę JAAS w taki sposób, aby uwierzytlniała Nazwę użytkownika wysłanego przez klienta.

Administrator produktu IBM MQ konfiguruje, które kanały produktu MQTT wymagają uwierzytelniania klienta przy użyciu usługi JAAS. Podaj nazwę konfiguracji JAAS dla każdego kanału, który ma wykonać uwierzytelnianie JAAS. Kanały mogą korzystać z tej samej konfiguracji JAAS lub mogą używać różnych konfiguracji JAAS. Konfiguracje są definiowane w produkcie *WMQData directory\qmgrs\qMgrName\mqxr\jaas.config*.

Plik *jaas.config* jest organizowany przez nazwę konfiguracji JAAS. Pod każdą nazwą konfiguracji znajduje się lista konfiguracji logowania; patrz [Rysunek 30](#) na stronie 236.

JAAS udostępnia cztery standardowe moduły logowania. Standardowe moduły logowania NT i UNIX mają ograniczoną wartość.

Moduł JndiLogin

Uwierzytlnia się przed usługą katalogową skonfigurowaną w interfejsie JNDI (Java Naming and Directory Interface).

Krb5LoginModule

Uwierzytlnia się przy użyciu protokołów Kerberos.

NTLoginModule

Uwierzytlnia się przy użyciu informacji o zabezpieczeniach systemu NT dla bieżącego użytkownika.

Moduł UnixLogin

Służy do uwierzytelniania przy użyciu informacji o zabezpieczeniach produktu UNIX dla bieżącego użytkownika.

Problem z używaniem produktu NTLoginModule lub UnixLoginModule polega na tym, że usługa telemetryczna (MQXR) działa z tożsamością *mqm*, a nie z tożsamością kanału MQTT. *mqm* jest tożsamością przekazywaną do produktu NTLoginModule lub UnixLoginModule w celu uwierzytelnienia, a nie tożsamości klienta.

Aby rozwiązać ten problem, napisz własny moduł logowania lub użyj innych standardowych modułów logowania. Przykładowy *JAASLoginModule.java* jest dostarczany razem z produktem MQ Telemetry. Jest to implementacja interfejsu *javax.security.auth.spi.LoginModule*. Użyj go do opracowania własnej metody uwierzytelniania.

Wszystkie nowe klasy LoginModule , które zostaną określone, muszą znajdować się w ścieżce klasy usługi telemetrii (MQXR). Nie należy umieszczać klas w katalogach IBM MQ , które znajdują się w ścieżce klasy. Utwórz własne katalogi i zdefiniuj całą ścieżkę klasy dla usługi telemetrycznej (MQXR).

Ścieżkę klas używaną przez usługę telemetryczną (MQXR) można rozszerzyć, ustawiając ścieżkę klasy w pliku `service.env` . CLASSPATH musi być skapitalizowana, a instrukcja ścieżki klasy może zawierać tylko literały. W zmiennej CLASSPATH nie można używać zmiennych; na przykład wartość `CLASSPATH=%CLASSPATH%` jest niepoprawna. Usługa telemetryczna (MQXR) ustawia własną ścieżkę klasy. Zostanie do niego dodana ścieżka CLASSPATH zdefiniowana w produkcie `service.env` .

Usługa telemetryczna (MQXR) udostępnia dwa wywołania zwrotne, które zwracają wartości Nazwa użytkownika i Hasło dla klienta połączony z kanałem MQTT . W obiekcie `MqttConnectOptions` ustawiane są wartości `Nazwa użytkownika` i `Hasło` . Zapoznaj się z przykładem dostępu do opcji `Nazwa użytkownika` i `Hasł`ów sekcji [Rysunek 31 na stronie 237](#) .

Przykłady

Przykład pliku konfiguracyjnego JAAS z jedną nazwaną konfiguracją, `MQXRConfig`.

```
MQXRConfig {
samples.JAASLoginModule required debug=true;
//com.ibm.security.auth.module.NTLoginModule required;
//com.ibm.security.auth.module.Krb5LoginModule required
//      principal=principal@your_realm
//      useDefaultCcache=TRUE
//      renewTGT=true;
//com.sun.security.auth.module.NTLoginModule required;
//com.sun.security.auth.module.UnixLoginModule required;
//com.sun.security.auth.module.Krb5LoginModule required
//      useTicketCache="true"
//      ticketCache="${user.home}/${}tickets";
};
```

Rysunek 30. Przykładowy plik `jaas.config`

Przykład modułu logowania JAAS zakodowanego w taki sposób, aby otrzymywać wartości `Username` i `Password` udostępniane przez klienta MQTT .

```

public boolean login()
throws javax.security.auth.login.LoginException {
    javax.security.auth.callback.Callback[] callbacks =
    new javax.security.auth.callback.Callback[2];
    callbacks[0] = new javax.security.auth.callback.NameCallback("NameCallback");
    callbacks[1] = new javax.security.auth.callback.PasswordCallback(
    "PasswordCallback", false);
    try {
        callbackHandler.handle(callbacks);
        String username = ((javax.security.auth.callback.NameCallback) callbacks[0])
        .getName();
        char[] password = ((javax.security.auth.callback.PasswordCallback) callbacks[1])
        .getPassword();
        // Accept everything.
        if (true) {
            loggedIn = true;
        } else
        throw new javax.security.auth.login.FailedLoginException("Login failed");

        principal= new JAASPrincipal(username);

    } catch (java.io.IOException exception) {
        throw new javax.security.auth.login.LoginException(exception.toString());
    } catch (javax.security.auth.callback.UnsupportedCallbackException exception) {
        throw new javax.security.auth.login.LoginException(exception.toString());
    }

    return loggedIn;
}

```

Rysunek 31. Przykładowa metoda `JAASLoginModule.Login()`

Informacje pokrewne

[Klasa MQXR AuthCallback](#)

Rozwiązywanie problemu: moduł logowania JAAS nie jest wywoływany przez usługę telemetryczną

V 9.0.0 administrowanie IBM MQ Light

Produkt MQ Light można administrować przy użyciu produktu IBM MQ Explorer lub w wierszu komend. Za pomocą Eksploratora można skonfigurować kanały i monitorować klienty MQ Light połączone z produktem IBM MQ. Skonfiguruj zabezpieczenia produktu MQ Light przy użyciu protokołu TLS i JAAS.

Zanim rozpocziesz

Więcej informacji na temat instalowania programu AMQP na platformie zawiera sekcja [Wybieranie opcji instalacji](#). Zainstaluj komponent usługi AMQP za pomocą odświeżenia produktu IBM MQ 8.0.0.4 , a nie pakietu poprawek 8.0.0.4 . Nie można zainstalować komponentu AMQP w wersji menedżera kolejek w wersji wcześniejszej niż 8.0.0.4.

Administrowanie przy użyciu produktu IBM MQ Explorer

Użyj eksploratora, aby skonfigurować kanały AMQP i monitorować klienty MQ Light połączone z produktem IBM MQ. Zabezpieczenia produktu MQ Light można skonfigurować przy użyciu protokołów TLS i JAAS.

Administrowanie za pomocą wiersza komend

MQ Light można administrować w wierszu komend za pomocą komend IBM MQ [MQSC](#) .

V 9.0.0 Wyświetlanie obiektów IBM MQ w użyciu przez klienty MQ Light

Istnieje możliwość wyświetlenia różnych zasobów produktu IBM MQ , które są używane przez klienty MQ Light , na przykład połączenia i subskrypcje.

Połączenia

Gdy usługa AMQP jest uruchomiona, nowe Hconns są tworzone i połączone z menedżerem kolejek. Ta pula urządzeń Hconns jest używana, gdy klienci MQ Light publikują komunikaty. Serwer Hconns można wyświetlić za pomocą komendy **DISPLAY CONN** . Na przykład:

```
DISPLAY CONN(*) TYPE(CONN) WHERE (APPLDESC LK 'WebSphere MQ Advanced Message Queuing Protocol*')
```

Ta komenda wyświetla również połączenia Hconns specyficzne dla klienta. Atrybuty Hconns, które mają pusty atrybut identyfikatora klienta, to Hconns używane w puli

Gdy klient MQ Light łączy się z kanałem AMQP, nowy Hconn jest połączony z menedżerem kolejek. Ten Hconn jest używany do asynchronicznego konsumowania komunikatów dla subskrypcji utworzonych przez klienta MQ Light . Używając komendy **DISPLAY CONN** , można wyświetlić Hconn używany przez konkretnego klienta MQ Light . Na przykład:

```
DISPLAY CONN(*) TYPE(CONN) WHERE (CLIENTID EQ 'recv_abcd1234')
```

Subskrypcje utworzone przez klienty

Gdy klient MQ Light subskrybuje temat, tworzona jest nowa subskrypcja programu IBM MQ . Nazwa subskrypcji zawiera następujące informacje:

- Nazwa klienta. Jeśli klient dołączył do subskrypcji współużytkowanej, używana jest nazwa zasobu współużytkowanego.
- Wzorzec tematu, który został subskrybowany przez klienta
- Przedrostek. Przedrostek to `private` , jeśli klient utworzył subskrypcję niewspółużytkowaną, lub `share` , jeśli klient dołączył do subskrypcji współużytkowanej.

Aby wyświetlić subskrypcje w użyciu przez określonego klienta MQ Light , uruchom komendę **DISPLAY SUB** i przefiltrować przedrostek `private` :

```
DISPLAY SUB(':private:*')
```

Aby wyświetlić współużytkowane subskrypcje, które są używane przez wiele klientów, należy uruchomić komendę **DISPLAY SUB** i filtrować prefiks `share` :

```
DISPLAY SUB(':share:*')
```

Ze względu na to, że subskrypcje współużytkowane mogą być używane przez wiele klientów MQ Light , użytkownik może chcieć wyświetlić klienty, które obecnie konsumują komunikaty z subskrypcji współużytkowanej. Można to zrobić, wyświetlając listę Hconns, które mają obecnie uchwyt otwarty w kolejce subskrypcji. Aby wyświetlić klienty aktualnie korzystające z zasobu współużytkowanego, wykonaj następujące kroki:

1. Znajdź nazwę kolejki używaną przez subskrypcję współużytkowaną jako miejsce docelowe. Na przykład:

```
DISPLAY SUB(':private:recv_e298452:public') DEST
5 : DISPLAY SUB(':private:recv_e298452:public') DEST
AMQ8096: WebSphere MQ subscription inquired.
SUBID(414D5120514D31202020202020202020707E0A565C2D0020)
SUB(:private:recv_e298452:public)
DEST(SYSTEM.MANAGED.DURABLE.560A7E7020002D5B)
```

2. Uruchom komendę **DISPLAY CONN** , aby znaleźć uchwyty otwarte dla tej kolejki:

```
DISPLAY CONN(*) TYPE(HANDLE) WHERE (OBJNAME
EQ SYSTEM.MANAGED.DURABLE.560A7E7020002D5B)
```

```
21 : DISPLAY CONN(*) TYPE(HANDLE) WHERE(OBJNAME EQ
SYSTEM.MANAGED.DURABLE.560A7E7020002D5B)
```

```
AMQ8276: Display Connection details.
CONN(707E0A56642B0020)
EXTCONN(414D5143514D312020202020202020)
TYPE(HANDLE)
```

```
OBJNAME(SYSTEM.BASE.TOPIC)      OBJTYPE(TOPIC)
```

```
OBJNAME(SYSTEM.MANAGED.DURABLE.560A7E7020002961)
OBJTYPE(Queue)
```

3. Dla każdego z uchwytów wyświetl identyfikator klienta MQ Light , który ma otwarty uchwyt:

```
DISPLAY CONN(707E0A56642B0020) CLIENTID
23 : DISPLAY CONN(707E0A56642B0020) CLIENTID
```

```
AMQ8276: Display Connection details.
CONN(707E0A56642B0020)
EXTCONN(414D5143514D312020202020202020)
TYPE(CONN)
```

```
CLIENTID(recv_8f02c9d)
DISPLAY CONN(707E0A565F290020) CLIENTID
24 : DISPLAY CONN(707E0A565F290020) CLIENTID
```

```
AMQ8276: Display Connection details.
CONN(707E0A565F290020)
EXTCONN(414D5143514D312020202020202020)
TYPE(CONN)
CLIENTID(recv_86d8888)
```

V 9.0.0

MQ Light - identyfikacja, autoryzacja i uwierzytelnianie klienta

Podobnie jak w przypadku innych aplikacji klienckich produktu IBM MQ , połączenia AMQP można zabezpieczyć na wiele sposobów.

Do zabezpieczania połączeń AMQP z produktem IBM MQ można używać następujących opcji zabezpieczeń:

- [Rekordy uwierzytelniania kanału](#)
- [Uwierzytelnianie połączenia](#)
- Konfiguracja użytkownika MCA kanału
- Definicje uprawnień produktu IBM MQ
- [Połączenia TLS](#)

Z perspektywy zabezpieczeń nawiąże połączenie składa się z następujących dwóch kroków:

- Rozstrzygnięcie, czy połączenie powinno być kontynuowane
- Podejmowanie decyzji o tym, która tożsamość IBM MQ oznacza, że aplikacja zakłada późniejsze sprawdzanie uprawnień.

Poniższe informacje przedstawia różne konfiguracje produktu IBM MQ oraz kroki, które są wykonywane, gdy klient AMQP podejmuje próbę nawiązania połączenia. Nie wszystkie konfiguracje produktu IBM MQ używają wszystkich opisanych kroków. Na przykład niektóre konfiguracje nie używają protokołu TLS do połączeń wewnątrz firewalla przedsiębiorstwa, a niektóre konfiguracje używają protokołu TLS, ale nie używają certyfikatów klienta do uwierzytelniania. W wielu środowiskach niestandardowe lub niestandardowe moduły JAAS nie są używane.

Nawiązanie połączenia

W poniższych krokach opisano, co się dzieje, gdy połączenie jest nawiązane przez klienta AMQP. Kroki określają, czy połączenie jest kontynuowane, a tożsamość IBM MQ aplikacji zakłada, że uprawnienia są sprawdzane:

1. Jeśli klient otwiera połączenie TLS z produktem IBM MQ i udostępnia certyfikat, menedżer kolejek próbuje sprawdzić poprawność certyfikatu klienta.

2. Jeśli klient udostępnia informacje autoryzacyjne nazwy użytkownika i hasła, menedżer kolejek otrzymuje ramkę SASL AMQP, a konfiguracja MQ CONNAUTH jest sprawdzana.
3. Sprawdzane są reguły uwierzytelniania kanału MQ (na przykład, czy adres IP i nazwa wyróżniająca certyfikatu TLS są poprawne)
4. Kanał MCAUSER jest sprawdzany, chyba że reguły uwierzytelniania kanału określają inaczej.
5. Jeśli moduł JAAS został skonfigurowany, jest wywoływany
6. Sprawdzenie uprawnień CONNECT (MQ) zastosowane do wynikowego identyfikatora użytkownika MQ .
7. Połączenie nawiązane z założonym tożsamością IBM MQ .

Publikowanie komunikatu

W poniższych krokach opisano, co się dzieje, gdy komunikat jest publikowany przez klienta AMQP. Kroki określają, czy połączenie jest kontynuowane, a tożsamość IBM MQ aplikacji zakłada, że uprawnienia są sprawdzane:

1. Ramka dołączania odsyłacza AMQP dociera do menedżera kolejek. Uprawnienie do publikowania IBM MQ dla określonego łańcucha tematu jest sprawdzane pod kątem tożsamości użytkownika MQ nawiązanej podczas nawiązywania połączenia.
2. Komunikat jest publikowany w określonym łańcuchu tematu.

Subskrybowanie wzorca tematu

W poniższych krokach opisano, co się dzieje, gdy klient AMQP subskrybuje wzorec tematu. Kroki określają, czy połączenie jest kontynuowane, a tożsamość IBM MQ aplikacji zakłada, że uprawnienia są sprawdzane:

1. Ramka dołączania odsyłacza AMQP dociera do menedżera kolejek. Uprawnienie do subskrypcji produktu IBM MQ dla określonego wzorca tematu jest sprawdzane pod kątem tożsamości użytkownika MQ nawiązanej podczas nawiązywania połączenia.
2. Subskrypcja została utworzona.

V 9.0.0 Tożsamość i autoryzacja klienta MQ Light

Aby uzyskać dostęp do obiektów produktu IBM MQ , należy użyć identyfikatora klienta MQ Light , nazwy użytkownika produktu MQ Light lub wspólnej tożsamości klienta zdefiniowanej w kanale lub w regule uwierzytelniania kanału.

Administrator dokonuje wyboru przy definiowaniu lub modyfikowaniu kanału AMQP, konfigurując ustawienie CONNAUTH menedżera kolejek lub definiując reguły uwierzytelniania kanału. Tożsamość jest używana do autoryzowania dostępu do tematów produktu IBM MQ . Wybór dokonany jest w oparciu o następujące elementy:

1. Atrybut kanału USECLNTID.
2. Atrybut ADOPTCTX reguły CONNAUTH menedżera kolejek.
3. Atrybut MCAUSER zdefiniowany w kanale.
4. Atrybut USERSRC zgodny z regułą uwierzytelniania kanału.

Unikaj problemów: Tożsamość wybrana przez ten proces jest następnie przywołana, na przykład przez komendę DISPLAY CHSTATUS (AMQP), jako użytkownik MCAUSER klienta. Należy pamiętać, że nie jest to koniecznie ta sama tożsamość, co MCAUSER kanału, o którym mowa w wyborze (2).

Komenda IBM MQ **setmqaut** służy do wybierania obiektów, które są autoryzowane do użycia przez tożsamość powiązaną z kanałem AMQP, i które czynności mają być używane. Na przykład następujące

komendy zezwalają na tożsamość kanału AMQPClient udostępnionego przez administratora menedżera kolejek QM1:

```
setmqaut -m QM1 -t topic -n SYSTEM.BASE.TOPIC -p AMQPClient -all +pub +sub
```

i

```
setmqaut -m QM1 -t qmgr -p AMQPClient -all +connect
```

V 9.0.0 Uwierzytelnianie klienta MQ Light przy użyciu hasła

Uwierzytelnij nazwę użytkownika MQ Light przy użyciu hasła klienta. Istnieje możliwość uwierzytelnienia klienta przy użyciu innej tożsamości z tożsamości używanej do autoryzowania klienta do publikowania i subskrybowania tematów.

Usługa AMQP może używać produktu MQ CONNAUTH lub JAAS do uwierzytelniania nazwy użytkownika klienta. Jeśli jedna z tych opcji jest skonfigurowana, hasło udostępniane przez klienta jest weryfikowane przez konfigurację MQ CONNAUTH lub moduł JAAS .

Poniższa procedura przedstawia przykładowe kroki w celu uwierzytelnienia poszczególnych użytkowników w stosunku do użytkowników i haseł lokalnego systemu operacyjnego oraz, jeśli powiedzie się, adoptować wspólną tożsamość AMQPUser:

1. Administrator produktu IBM MQ ustawia tożsamość MCAUSER kanału AMQP na dowolną nazwę, taką jak AMQPUser, za pomocą programu IBM MQ Explorer.
2. Administrator produktu IBM MQ autoryzuje produkt AMQPUser do publikowania i subskrybowania dowolnego tematu:

```
setmqaut -m QM1 -t topic -n SYSTEM.BASE.TOPIC -p AMQPUser -all +pub +sub +connect
```

3. Administrator produktu IBM MQ konfiguruje regułę IDPWOS CONNAUTH w celu sprawdzenia nazwy użytkownika i hasła prezentowanego przez klienta. W regule CONNAUTH należy ustawić CHCKCLNT (REQUIRED) i ADOPTCTX (NO).

Uwaga: Zalecane jest użycie reguł uwierzytelniania kanału i ustawienie atrybutu kanału MCAUSER na użytkownika, który nie ma uprawnień, w celu umożliwienia większej kontroli nad połączeniami z menedżerem kolejek.

V 9.0.0 Prywatność publikacji w kanałach

Prywatność publikacji AMQP wysłanych w dowolnym kierunku w kanałach AMQP jest zabezpieczona za pomocą protokołu TLS do szyfrowania transmisji przez połączenie.

Klienci AMQP łączące się z kanałami AMQP korzystają z protokołu TLS w celu zabezpieczenia prywatności publikacji przesyłanych w kanale za pomocą kryptografii klucza symetrycznego. Ponieważ punkty końcowe nie są uwierzytelniane, szyfrowanie kanału nie jest możliwe tylko w przypadku szyfrowania. Połączenie zabezpieczania prywatności z serwerem lub uwierzytelniania wzajemnego.

Zamiast używać protokołu TLS, niektóre rodzaje sieci VPN (Virtual Private Network-VPN), takie jak IPsec, uwierzytelniają punkty końcowe połączenia TCP/IP. Sieć VPN szyfruje każdy pakiet IP przepływający przez sieć. Po ustanowieniu połączenia VPN uzyskana sieć jest zaufana. Klienci AMQP można łączyć z kanałami AMQP za pomocą protokołu TCP/IP przez sieć VPN.

Szyfrowanie połączeń TLS bez uwierzytelniania serwera naraża połączenie z atakami typu man-in-the-middle. Mimo, że informacje, które wymieniasz, są chronione przed podsłuchami, nie wiesz, z kim się wymieniasz. Jeśli sieć nie jest sterowana, użytkownik jest narażony na przechwytywanie transmisji danych IP i maskowanie jako punkt końcowy.

Szyfrowane połączenie TLS można utworzyć bez uwierzytelniania serwera, korzystając z wymiany kluczy Diffie-Hellman CipherSpec , która obsługuje anonimowy protokół TLS. Główny klucz tajny,

współużytkowany między klientem i serwerem i używany do szyfrowania transmisji TLS, jest ustanawiany bez wymiany certyfikatu serwera prywatnego.

Ponieważ anonimowe połączenia są niezabezpieczone, większość implementacji protokołu TLS nie jest domyślnie używana do używania anonimowych specyfikacji CipherSpecs. Jeśli żądanie klienta dotyczące połączenia TLS jest akceptowane przez kanał AMQP, kanał kluczy musi być chroniony przez hasło. Domyślnie, ponieważ implementacje protokołu TLS nie używają anonimowych specyfikacji CipherSpecs, magazyn kluczy musi zawierać podpisany przez klienta certyfikat, który może być uwierzytelniony przez klienta.

Jeśli używana jest anonimowa opcja CipherSpecs, magazyn kluczy serwera musi istnieć, ale nie musi zawierać żadnych prywatnych certyfikatów podpisanych.

Innym sposobem nawiązania szyfrowanego połączenia jest zastąpienie dostawcy zaufanego na kliencie we własnej implementacji. Dostawca zaufania nie uwierzytelniłby certyfikatu serwera, ale połączenie będzie szyfrowane.

V 9.0.0 Konfigurowanie klientów MQ Light za pomocą protokołu TLS

Klienci produktu MQ Light można skonfigurować w taki sposób, aby używały protokołu TLS w celu ochrony danych przepływających przez sieć oraz do uwierzytelniania tożsamości menedżera kolejek, z którym łączy się klient.

Aby użyć protokołu TLS dla połączenia z klientem MQ Light do kanału AMQP, należy upewnić się, że menedżer kolejek został skonfigurowany do obsługi protokołu TLS. Sekcja [Konfigurowanie protokołu TLS w menedżerach kolejek](#) opisuje sposób konfigurowania magazynu kluczy, z którego menedżer kolejek odczytuje certyfikaty TLS.

Gdy menedżer kolejek został skonfigurowany przy użyciu magazynu kluczy, należy skonfigurować atrybuty TLS na kanale AMQP, z którym będą się łączyć klienci. Kanały AMQP mają cztery atrybuty związane z konfiguracją TLS w następujący sposób:

SSLCAUTH

Atrybut SSLCAUTH służy do określania, czy menedżer kolejek powinien wymagać klienta MQ Light, aby mógł przedstawić certyfikat klienta w celu zweryfikowania jego tożsamości.

SSLCIPH

Atrybut SSLCIPH określa szyfr, który powinien być używany przez kanał do kodowania danych w przepływie TLS.

SSLPEER

Atrybut SSLPEER jest używany do określania nazwy wyróżniającej (DN) certyfikatu klienta, który musi być zgodny, jeśli połączenie ma być dozwolone.

CERTLABL

CERTLABL określa certyfikat, który menedżer kolejek powinien przedstawić klientowi. Magazyn kluczy menedżera kolejek może zawierać wiele certyfikatów. Ten atrybut umożliwia określenie certyfikatu, który ma być używany dla połączeń z tym kanałem. Jeśli nie zostanie określony żaden parametr CERTLABL, zostanie użyty certyfikat w repozytorium kluczy menedżera kolejek z etykietą odpowiadającą atrybutowi CERTLABL menedżera kolejek.

Jeśli kanał AMQP został skonfigurowany z atrybutami TLS, należy zrestartować usługę AMQP za pomocą następującej komendy:

```
STOP SERVICE(SYSTEM.AMQP.SERVICE) START SERVICE(SYSTEM.AMQP.SERVICE)
```

Gdy klient MQ Light łączy się z kanałem AMQP zabezpieczonym przez protokół TLS, klient weryfikuje tożsamość certyfikatu prezentowanego przez menedżera kolejek. W tym celu należy skonfigurować klienta MQ Light za pomocą magazynu zaufanych certyfikatów zawierającego certyfikat menedżera kolejek. Kroki, które należy wykonać, zależą od używanego klienta MQ Light.

- For the MQ Light client for Node JS API documentation, see <https://www.npmjs.com/package/mqlight>

- Informacje na temat klienta MQ Light dla dokumentacji interfejsu API języka Java znajdują się w sekcji <https://mqlight.github.io/java-mqlight/>.
- Dokumentację produktu MQ Light Client for Ruby można znaleźć w sekcji <https://www.rubydoc.info/github/mqlight/ruby-mqlight/>.
- Dokumentację klienta MQ Light dla języka Python można znaleźć w sekcji <https://python-mqlight.readthedocs.org/en/latest/>.

V 9.0.0 Rozłączanie klientów MQ Light z menedżera kolejek

Jeśli chcesz odłączyć produkt MQ Light od menedżera kolejek, uruchom komendę PURGE CHANNEL lub zatrzymaj połączenie z klientem MQ Light.

- Uruchom komendę **PURGE CHANNEL**. Na przykład:

```
PURGE CHANNEL(MYAMQP) CLIENTID('recv_28dbb7e')
```

- Alternatywnie można zatrzymać połączenie, które klient MQ Light używa do rozłączenia klienta, wykonując następujące kroki:

1. Znajdź połączenie, z którego korzysta klient, uruchamiając komendę **DISPLAY CONN**. Na przykład:

```
DISPLAY CONN(*) TYPE(CONN) WHERE (CLIENTID EQ 'recv_28dbb7e')
```

Dane wyjściowe komendy są następujące:

```
DISPLAY CONN(*) TYPE(CONN) WHERE(CLIENTID EQ 'recv_28dbb7e')
40 : DISPLAY CONN(*) TYPE(CONN) WHERE(CLIENTID EQ 'recv_28dbb7e')
AMQ8276: Display Connection details.
CONN(707E0A565F2D0020)
EXTCONN(414D5143514D31202020202020202020)
TYPE(CONN)
CLIENTID(recv_28dbb7e)
```

2. Zatrzymaj połączenie. Na przykład:

```
STOP CONN(707E0A565F2D0020)
```

Administrowanie rozgłaszanie

Informacje zawarte w tej sekcji umożliwiają zapoznanie się z zadaniami administracyjnymi produktu IBM MQ Multicast, takimi jak zmniejszenie wielkości komunikatów rozsyłania grupowego i włączenie konwersji danych.

Pierwsze kroki w trybie rozsyłania grupowego

Te informacje umożliwiają rozpoczęcie pracy z tematami rozsyłania grupowego produktu IBM MQ i obiektami informacji o komunikacji.

O tym zadaniu

IBM MQ Rozsyłanie komunikatów rozsyłania grupowego korzysta z sieci w celu dostarczania komunikatów przez odwzorowanie tematów na adresy grupowe. Poniższe zadania umożliwiają szybkie przetestowanie, czy wymagany adres IP i port są poprawnie skonfigurowane do rozsyłania grupowego przesyłania komunikatów.

Tworzenie obiektu **COMMINFO** dla rozsyłania grupowego

Obiekt informacji o komunikacji (COMMINFO) zawiera atrybuty powiązane z transmisją rozsyłania grupowego. Więcej informacji na temat parametrów obiektu COMMINFO zawiera sekcja [DEFINE COMMINFO](#).

Użyj następującego przykładu wiersza komend, aby zdefiniować obiekt COMMINFO dla rozsyłania grupowego:

```
DEFINE COMMINFO(MC1) GRPADDR(group address) PORT(port number)
```

gdzie *MC1* jest nazwą obiektu COMMINFO, *adres grupy* jest grupowym adresem IP grupowym lub nazwą DNS, a *numer_portu* jest portem do przesłania (wartością domyślną jest 1414).

Zostanie utworzony nowy obiekt COMMINFO o nazwie *MC1*. Ta nazwa jest nazwą, którą należy określić podczas definiowania obiektu TOPIC w następnym przykładzie.

Tworzenie obiektu TOPIC na potrzeby rozsyłania grupowego

Temat jest tematem informacji, które są publikowane w komunikacji publikowania/subskrypcji, a temat jest definiowany przez utworzenie obiektu TOPIC. Obiekty TOPIC mają dwa parametry, które definiują, czy mogą być używane z rozsyłaniem grupowym, czy nie. Są to następujące parametry: **COMMINFO** i **MCAST**.

- **COMMINFO** Ten parametr określa nazwę obiektu informacji o komunikacji rozsyłania grupowego. Więcej informacji na temat parametrów obiektu COMMINFO zawiera sekcja [DEFINE COMMINFO](#).
- **MCAST** Ten parametr określa, czy rozsyłanie grupowe jest dozwolone na tej pozycji w drzewie tematów.

Użyj następującego przykładu wiersza komend, aby zdefiniować obiekt TOPIC na potrzeby rozsyłania grupowego:

```
DEFINE TOPIC(ALLSPORTS) TOPICSTR('Sports') COMMINFO(MC1) MCAST(ENABLED)
```

Zostanie utworzony nowy obiekt TOPIC o nazwie *ALLSPORTS*. Ma on łańcuch tematu *Sport*, powiązany z nim obiekt informacji o komunikacji nosi nazwę *MC1* (jest to nazwa określona podczas definiowania obiektu COMMINFO w poprzednim przykładzie), a rozsyłanie grupowe jest włączone.

Testowanie publikowania/subskrypcji rozsyłania grupowego

Po utworzeniu obiektów TOPIC i COMMINFO mogą one być testowane przy użyciu przykładu `amqspubc` i przykładu `amqssubc`. Więcej informacji na temat tych przykładów można znaleźć w sekcji [Programy przykładowe publikowania/subskrypcji](#).

1. Otwórz dwa okna wiersza komend; pierwszy wiersz komend jest przeznaczony dla przykładu publikowania produktu `amqspubc`, a drugi wiersz komend jest przeznaczony dla przykładu subskrypcji produktu `amqssubc`.
2. W wierszu komend 1 wpisz następującą komendę:

```
amqspubc Sports QM1
```

gdzie *Sports* to łańcuch tematu obiektu TOPIC zdefiniowany w poprzednim przykładzie, a *QM1* to nazwa menedżera kolejek.

3. W wierszu komend 2 wpisz następującą komendę:

```
amqssubc Sports QM1
```

gdzie *Sports* i *QM1* są takie same, jak używane w kroku "2" na stronie 244.

4. Wpisz `Hello world` w wierszu komend 1. Jeśli port i adres IP określone w obiekcie COMMINFO są poprawnie skonfigurowane; przykład `amqssubc`, który nasłuchuje na porcie dla publikacji z podanego adresu, dane wyjściowe `Hello world` w wierszu komend 2.

IBM MQ Topologia tematu rozsyłania grupowego

W tym przykładzie przedstawiono informacje na temat topologii tematów IBM MQ Multicast.

IBM MQ Obsługa rozsyłania grupowego wymaga, aby każde poddrzewo miało własną grupę rozgłaszania i strumień danych w ramach ogólnej hierarchii.

Schemat adresowania IP *sieci klasowej* ma określoną przestrzeń adresową dla adresów rozsyłania. Pełny zakres rozsyłania adresu IP to adresy od 224.0.0.0 do 239.255.255.255, ale niektóre z tych adresów są zarezerwowane. Aby uzyskać listę zarezerwowanych adresów, należy skontaktować się z administratorem systemu albo zapoznać się z tematem <https://www.iana.org/assignments/multicast-addresses> w celu uzyskania dodatkowych informacji. Zaleca się użycie adresu rozsyłania grupowego o zasięgu lokalnym z zakresu od 239.0.0.0 do 239.255.255.255.

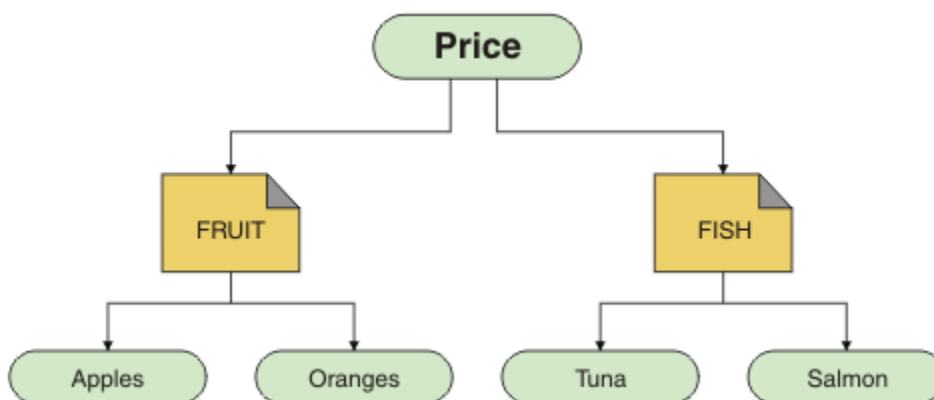
Na poniższym diagramie istnieją dwa możliwe strumienie danych rozsyłania grupowego:

```
DEF COMMINFO(MC1) GRPADDR(239.XXX.XXX.XXX
)
DEF COMMINFO(MC2) GRPADDR(239.YYY.YYY.YYY)
```

gdzie 239.XXX.XXX.XXX i 239.YYY.YYY.YYY są poprawnymi adresami rozsyłania grupowego.

Te definicje tematów są używane do tworzenia drzewa tematów w sposób przedstawiony na poniższym diagramie:

```
DEFINE TOPIC(FRUIT) TOPICSTRING('Price/FRUIT') MCAST(ENABLED) COMMINFO(MC1)
DEFINE TOPIC(FISH) TOPICSTRING('Price/FISH') MCAST(ENABLED) COMMINFO(MC2)
```



Każdy obiekt informacji o komunikacji rozsyłania (multicast communication information-COMMINFO) reprezentuje inny strumień danych, ponieważ ich adresy grupowe są różne. W tym przykładzie temat FRUIT jest zdefiniowany do użycia obiektu COMMINFO MC1, temat FISH jest zdefiniowany do użycia obiektu COMMINFO MC2, a węzeł Price nie ma definicji rozsyłania grupowego.

IBM MQ W przypadku łańcuchów tematów rozsyłanie grupowe jest ograniczone do 255 znaków. To ograniczenie oznacza, że należy zachować ostrożność przy użyciu nazw węzłów i węzłów-liści w obrębie drzewa. Jeśli nazwy węzłów i liści są zbyt długie, łańcuch tematu może zawierać więcej niż 255 znaków i zwrócić kod przyczyny 2425 (0979) (RC2425): MQRC_TOPIC_STRING_ERROR. Zaleca się, aby łańcuchy tematów były tak krótkie, jak to tylko możliwe, ponieważ dłuższe łańcuchy tematów mogą mieć szkodliwy wpływ na wydajność.

Kontrolowanie wielkości komunikatów rozsyłania grupowego

Informacje zawarte w tej sekcji umożliwiają zapoznanie się z formatem komunikatów produktu IBM MQ oraz zmniejszenie wielkości komunikatów produktu IBM MQ.

Komunikaty produktu IBM MQ zawierają pewną liczbę powiązanych z nimi atrybutów, które są zawarte w deskrypcji komunikatu. W przypadku małych komunikatów atrybuty te mogą reprezentować większość ruchu danych i mogą mieć istotny szkodliwy wpływ na szybkość transmisji. IBM MQ

Rozsyłanie grupowe umożliwia użytkownikowi konfigurowanie, które z tych atrybutów są przesyłane wraz z komunikatem.

Obecność atrybutów komunikatu, innych niż łańcuch tematu, zależy od tego, czy obiekt COMMINFO wskazuje, że należy je wysłać, czy nie. Jeśli atrybut nie zostanie przesłany, aplikacja odbierający zastosuje wartość domyślną. Domyślne wartości MQMD nie muszą być takie same, jak wartość MQMD_DEFAULT, i są opisane w dalszej części w Tabeli 17 na stronie 246.

Obiekt COMMINFO zawiera atrybut MCPROP, który steruje sposobem przepływu wielu pól MQMD i właściwości użytkownika z komunikatem. Ustawiając wartość tego atrybutu na odpowiedni poziom, można kontrolować wielkość komunikatów IBM MQ Multicast:

MCPROP

Właściwości rozsyłania sterują liczbą właściwości MQMD i właściwości użytkownika przepływających wraz z komunikatem.

ALL

Przesyłane są wszystkie właściwości użytkownika i wszystkie pola deskryptora MQMD.

Odpowiedź

Przesyłane są tylko właściwości użytkownika oraz pola MQMD dotyczące odpowiadania na komunikaty. Są to następujące właściwości:

- MsgType
- MessageId
- CorrelId
- ReplyToQ
- Menedżer_kolejek_zwrotnych

UŻYTKOWNIK

Przesyłane są tylko właściwości użytkownika.

BRAK

Nie są przesyłane właściwości użytkownika ani pola MQMD.

COMPAT

Ta wartość powoduje, że transmisja komunikatu jest wykonywana w trybie zgodnym z RMM, co pozwala na pewne współdziałanie z bieżącymi aplikacjami XMS oraz aplikacjami IBM Integration Bus RMM.

Atrybuty komunikatów rozsyłania grupowego

Atrybuty komunikatów mogą pochodzić z różnych miejsc, takich jak MQMD, pola w tabeli MQRFH2i właściwości komunikatu.

Poniższa tabela zawiera informacje o tym, co się dzieje, gdy komunikaty są wysyłane z uwzględnieniem wartości MCPROP, a także wartości domyślnej używanej, gdy atrybut nie jest wysyłany.

Tabela 17. Atrybuty przesyłania komunikatów i sposób ich powiązania z rozsyłaniem grupowym

Atrybut	Działanie podczas korzystania z rozsyłania grupowego	Domyślna, jeśli nie przesłano
TopicString	Zawsze włączone	Nie dotyczy
MQMQ StrucId	Nieprzesłane	Nie dotyczy
Wersja MQMD	Nieprzesłane	Nie dotyczy
Raport	Dołączony, jeśli nie jest domyślny	0
MsgType	Dołączony, jeśli nie jest domyślny	MQMT_DATAGRAM
Utrata ważności	Dołączony, jeśli nie jest domyślny	0

Tabela 17. Atrybuty przesyłania komunikatów i sposób ich powiązania z rozsyłaniem grupowym (kontynuacja)

Atrybut	Działanie podczas korzystania z rozsyłania grupowego	Domyślna, jeśli nie przestano
Opinie	Dołączony, jeśli nie jest domyślny	0
Kodowanie	Dołączony, jeśli nie jest domyślny	MQENC_NORMAL (equiv)
CodedCharSetId	Dołączony, jeśli nie jest domyślny	1208
Format	Dołączony, jeśli nie jest domyślny	MQRFH2
Priorytet	Dołączony, jeśli nie jest domyślny	4
Trwałość	Dołączony, jeśli nie jest domyślny	MQPER_NOT_PERSISTENT
MsgId	Dołączony, jeśli nie jest domyślny	NULL
CorrelId	Dołączony, jeśli nie jest domyślny	NULL
BackoutCount	Dołączony, jeśli nie jest domyślny	0
ReplyToQ	Dołączony, jeśli nie jest domyślny	Wartość pusta
ReplyToQMgr	Dołączony, jeśli nie jest domyślny	Wartość pusta
UserIdentifier	Dołączony, jeśli nie jest domyślny	Wartość pusta
AccountingToken	Dołączony, jeśli nie jest domyślny	NULL
Typ IType PutApp	Dołączony, jeśli nie jest domyślny	MQAT_JAVA
Nazwa IName PutApp	Dołączony, jeśli nie jest domyślny	Wartość pusta
PutDate	Dołączony, jeśli nie jest domyślny	Wartość pusta
PutTime	Dołączony, jeśli nie jest domyślny	Wartość pusta
Dane_pochodzenia_aplikacji	Dołączony, jeśli nie jest domyślny	Wartość pusta
GroupID	Wykluczone	Nie dotyczy
Numer_kolejny_komunikatu	Wykluczone	Nie dotyczy
Depozycja	Wykluczone	Nie dotyczy
MsgFlags	Wykluczone	Nie dotyczy
OriginalLength	Wykluczone	Nie dotyczy
UserProperties	Uwzględnione	Nie dotyczy

Odsyłacze pokrewne

[ALTER COMMINFO](#)

Informacje pokrewne

[KOMENDA DEFINE](#)

Włączanie konwersji danych na potrzeby przesyłania komunikatów w trybie Multicast

Informacje zawarte w tej sekcji umożliwiają zrozumienie sposobu działania konwersji danych w programie IBM MQ Multicast Messaging.

IBM MQ Multicast to współużytkowany, bezpołączeniowy protokół, a więc nie jest możliwe, aby każdy klient składał konkretne żądania konwersji danych. Każdy klient zasubskrybowany do tego samego

strumienia rozsyłania grupowego odbiera te same dane binarne. Dlatego też, jeśli wymagana jest konwersja danych IBM MQ , konwersja jest wykonywana lokalnie na każdym kliencie.

W przypadku instalacji z różnymi platformami większość klientów wymaga danych w formacie, który nie jest rodzimą formatem aplikacji przekazującej. W takiej sytuacji wartości **CCSID** i **ENCODING** obiektu rozsyłania grupowego COMMINFO mogą zostać użyte do zdefiniowania kodowania transmisji komunikatu pod kątem wydajności.

IBM MQ Multicast obsługuje konwersję danych z ładunku komunikatu dla następujących wbudowanych formatów:

- MQADMIN
- MQEVENT,
- MQPCF
- MQRFH,
- MQRFH2
- MQSTR

Oprócz tych formatów można również zdefiniować własne formaty i użyć wyjścia konwersji danych [MQDXP-Data-conversion exit parameter](#) .

Informacje na temat konwersji danych na temat programowania zawiera sekcja [Konwersja danych w interfejsie MQI na potrzeby przesyłania komunikatów rozsyłania grupowego](#).

Więcej informacji na temat konwersji danych zawiera sekcja [Konwersja danych](#).

Więcej informacji na temat wyjść konwersji danych i `ClientExitPath` zawiera sekcja [ClientExitPath](#) w pliku konfiguracyjnym klienta.

Monitorowanie aplikacji rozsyłania grupowego

Informacje zawarte w tej sekcji umożliwiają zapoznanie się z administrowaniem i monitorowaniem programu IBM MQ Multicast.

Status bieżących publikatorów i subskrybentów dla ruchu rozsyłania grupowego (na przykład liczba wystanych i odebranych komunikatów lub liczba utraconych komunikatów) jest okresowo przesyłana do serwera od klienta. Po odebraniu statusu atrybut `COMMEV` obiektu COMMINFO określa, czy menedżer kolejek umieszcza komunikat zdarzenia w systemie SYSTEM.ADMIN.PUBSUB.EVENT. Komunikat o zdarzeniu zawiera odebrane informacje o statusie. Informacje te stanowią bezcenną pomoc diagnostyczną w znalezieniu źródła problemu.

Komenda MQSC **DISPLAY CONN** służy do wyświetlania informacji o połączeniu z aplikacjami połączonymi z menedżerem kolejek. Więcej informacji na temat komendy **DISPLAY CONN** zawiera sekcja [DISPLAY CONN](#).

Aby wyświetlić status publikatorów i subskrybentów, należy użyć komendy MQSC **DISPLAY TPSTATUS** . Więcej informacji na temat komendy **DISPLAY TPSTATUS** zawiera sekcja [DISPLAY TPSTATUS](#).

COMMEV i indyktor niezawodności komunikatów rozsyłania grupowego

Wskaźnik niezawodności, używany w połączeniu z atrybutem **COMMEV** obiektu COMMINFO, jest kluczowym elementem monitorowania publikatorów i subskrybentów produktu IBM MQ Multicast. Indykator niezawodności (pole **MSGREL** zwracane w komendach statusu publikowania lub subskrypcji) jest indykatorem IBM MQ , który ilustruje procent transmisji, które nie zawierają błędów. Czasami komunikaty muszą zostać ponownie przesłane z powodu błędu transmisji, co znajduje odzwierciedlenie w wartości **MSGREL**. Potencjalne przyczyny błędów transmisji to powolne subskrybenty, sieci zajęte i wyłączenia sieci. Produkt **COMMEV** określa, czy komunikaty o zdarzeniach są generowane dla uchwytów rozsyłania grupowego, które są tworzone przy użyciu obiektu COMMINFO, i które są ustawione na jedną z trzech możliwych wartości:

WYŁĄCZONE

Komunikaty o zdarzeniach nie są zapisywane.

WŁĄCZONY

Komunikaty zdarzeń są zawsze zapisywane, z częstotliwością zdefiniowaną w parametrze **COMMINFO MONINT**.

WYJĄTEK

Komunikaty zdarzeń są zapisywane, jeśli niezawodność komunikatu jest poniżej progu niezawodności. Poziom niezawodności komunikatów wynoszący 90% lub mniej wskazuje, że może wystąpić problem z konfiguracją sieci lub że co najmniej jeden z aplikacji publikowania/subskrypcji działa zbyt wolno:

- Wartość **MSGREL (100, 100)** wskazuje, że nie wystąpiły żadne problemy w krótkim czasie ani w długoterminowym przedziale czasu.
- Wartość **MSGREL (80, 60)** wskazuje, że 20% komunikatów ma obecnie problemy, ale jest to także poprawa w zakresie wartości długoterminowej 60.

Klienci mogą kontynuować przesyłanie i odbieranie ruchu rozsyłania grupowego nawet wtedy, gdy połączenie pojedyncze z menedżerem kolejek zostanie zerwane, dlatego dane mogą być nieaktualne.

Niezawodność komunikatów rozsyłania grupowego

Informacje zawarte w tej sekcji umożliwiają poznanie sposobu ustawiania historii subskrypcji i komunikatów programu IBM MQ Multicast.

Kluczowym elementem pokonywania niepowodzenia transmisji przy użyciu rozsyłania grupowego jest buforowanie przestanych danych (historia komunikatów, które mają być przechowywane w wysyłającym końcu łącza) przez program IBM MQ. Ten proces oznacza, że w procesie wprowadzania aplikacji nie jest wymagane buforowanie komunikatów, ponieważ produkt IBM MQ zapewnia niezawodność. Wielkość tej historii jest konfigurowanych za pomocą obiektu informacji o komunikacji (COMMINFO), zgodnie z opisem w poniższych informacjach. Większy bufor transmisji oznacza, że w razie potrzeby retransmitowane jest więcej historii transmisji, ale ze względu na charakter rozsyłania grupowego, nie można zapewnić 100% gwarantowanego dostarczenia.

Historia komunikatu IBM MQ Multicast jest sterowana w obiekcie informacji o komunikacji (COMMINFO) przez atrybut **MSGHIST**:

MSGHIST

Ta wartość określa wielkość historii komunikatów w kilobajtach, która jest przechowywana przez system w celu obsługi retransmisji w przypadku NACKs (ujemnych potwierdzeń).

Wartość 0 zapewnia najniższy poziom niezawodności. Wartością domyślną jest 100 kB.

Nowa historia subskrypcji produktu IBM MQ Multicast jest kontrolowana w obiekcie informacji o komunikacji (COMMINFO) za pomocą atrybutu **NSUBHIST**:

NSUBHIST

Historia nowego subskrybenta decyduje o tym, czy subskrybent łączący się ze strumieniem publikacji otrzymuje tyle danych, ile jest aktualnie dostępnych, czy też tylko te publikacje, które pojawiły się od czasu subskrypcji.

BRAK

Wartość NONE powoduje, że nadajnik przekazuje tylko publikację, która została wykonana od momentu subskrypcji. NONE jest wartością domyślną.

ALL

Wartość ALL powoduje, że nadajnik retransmituje tyle historii tematu, co jest znane. W niektórych przypadkach sytuacja taka może dać podobne zachowanie do zachowanych publikacji.

Uwaga: Użycie wartości ALL może mieć szkodliwy wpływ na wydajność, jeśli istnieje duża historia tematów, ponieważ cała historia tematów jest retransmitowanych.

Informacje pokrewne

[KOMENDA DEFINE](#)

[ALTER COMMINFO](#)

Zaawansowane zadania rozsyłania grupowego

Te informacje umożliwiają zapoznanie się z zaawansowanymi zadaniami administrowania IBM MQ Multicast, takimi jak konfigurowanie plików `.ini` i współdziałanie z programem IBM MQ LLM.

Uwagi dotyczące zabezpieczeń w instalacji typu Multicast można znaleźć w sekcji [Bezpieczeństwo w wielu rzutowaniu](#).

Mostkowanie między domenami publikowania/subskrybowania rozsyłania grupowego i niegrupowego

Te informacje umożliwiają zrozumienie, co się dzieje, gdy publikator inny niż rozsyłany grupowy publikuje w temacie IBM MQ Multicast włączony.

Jeśli publikator inny niż rozsyłanie grupowe publikuje w temacie, który jest zdefiniowany jako **MCAST** włączony, a opcja **BRIDGE** jest włączona, menedżer kolejek przekazuje komunikat przy użyciu rozsyłania grupowego bezpośrednio do wszystkich subskrybentów, które mogą nasłuchiwać. Publikator rozsyłania grupowego nie może publikować w tematach, które nie są włączone w trybie rozsyłania grupowego.

Istniejące tematy mogą być rozsyłane grupowo, ustawiając parametry **MCAST** i **COMMINFO** dla obiektu tematu. Więcej informacji na temat tych parametrów zawiera sekcja [Początkowe pojęcia dotyczące rozsyłania grupowego](#).

Atrybut obiektu **COMMINFO BRIDGE** steruje publikacjami z aplikacji, które nie korzystają z rozsyłania grupowego. Jeśli parametr **BRIDGE** ma wartość **ENABLED**, a parametr **MCAST** tematu jest również ustawiony na wartość **ENABLED**, publikacje z aplikacji, które nie korzystają z rozsyłania grupowego, są połączone mostem z aplikacjami, które są używane. Więcej informacji na temat parametru **BRIDGE** zawiera sekcja [DEFINE COMMINFO](#).

Konfigurowanie plików `.ini` dla rozsyłania grupowego

Informacje zawarte w tej sekcji umożliwiają zrozumienie pól IBM MQ Multicast w plikach `.ini`.

Dodatkową konfigurację IBM MQ Multicast można dokonać w pliku `ini`. Konkretny plik `ini`, który musi być używany, zależy od typu aplikacji:

- Klient: Skonfiguruj plik `MQ_DATA_PATH/mqclient.ini`.
- Menedżer kolejek: Skonfiguruj plik `MQ_DATA_PATH/qmgrs/QMNAME/qm.ini`.

gdzie `MQ_DATA_PATH` to położenie katalogu danych IBM MQ (`/var/mqm/mqclient.ini`), a `QMNAME` to nazwa menedżera kolejek, którego dotyczy plik `.ini`.

Plik `.ini` zawiera pola używane do precyzyjnego dostrajania zachowania IBM MQ Multicast:

```
Multicast:
Protocol      = IP | UDP
IPVersion     = IPv4 | IPv6 | ANY | BOTH
LimitTransRate = DISABLED | STATIC | DYNAMIC
TransRateLimit = 100000
SocketTTL     = 1
Batch         = NO
Loop          = 1
Interface     = <IPAddress>
FeedbackMode  = ACK | NACK | WAIT1
HeartbeatTimeout = 20000
HeartbeatInterval = 2000
```

Protokół

UDP

W tym trybie pakiety są wysyłane za pomocą protokołu UDP. Elementy sieci nie mogą zapewnić asysty w dystrybucji rozsyłania grupowego, ponieważ są one jednak w trybie IP. Format pakietu pozostaje kompatybilny z PGM. Jest to wartość domyślna.

Adres IP

W tym trybie nadajnik wysyła surowe pakiety IP. Elementy sieciowe obsługiwane przez program PGM pomagają w niezawodnej dystrybucji pakietów rozsyłania grupowego. Ten tryb jest w pełni zgodny ze standardem PGM.

IPVERSION**IPv4**

Komunikują się wyłącznie za pomocą protokołu IPv4 . Jest to wartość domyślna.

IPv6

Komunikują się wyłącznie za pomocą protokołu IPv6 .

ANY

W zależności od tego, który protokół jest dostępny, komunikują się za pomocą IPv4, IPv6 lub obu tych protokołów.

BOTH

Obsługuje komunikację przy użyciu produktów IPv4 i IPv6.

Szybkość LimitTrans**WYŁĄCZONE**

Nie ma kontroli szybkości transmisji. Jest to wartość domyślna.

STATYCZNA

Implementuje sterowanie statyczną szybkością transmisji. Nadajnik nie prześle w tempie przekraczającym szybkość określoną przez parametr TransRateLimit.

DYNAMICZNA

Nadajnik dostosowuje swoją szybkość transmisji zgodnie z informacją zwrotną, którą otrzymuje od odbiorców. W tym przypadku limit szybkości transmisji nie może być większy niż wartość określona za pomocą parametru Limit TransRate. Nadajnik stara się osiągnąć optymalną szybkość transmisji.

Limit TransRate

Limit szybkości transmisji danych w kb/s.

SocketTTL

Wartość parametru SocketTTL określa, czy ruch rozsyłania grupowego może przechodzić przez router, czy też liczba routerów, które może przejść przez router.

Wsadowe

Określa, czy komunikaty są wsadowe, czy wysyłane natychmiast. Istnieją dwie możliwe wartości:

- *NIE* Komunikaty nie są wsadowe, są wysyłane natychmiast.
- *TAK* Komunikaty są wsadowe.

Pętla

Aby włączyć pętlę rozsyłania grupowego, należy ustawić wartość 1 . Pętla rozsyłania grupowego definiuje, czy wysłane dane są zapętlone z powrotem do hosta, czy nie.

Interfejs

Adres IP interfejsu, na którym przepływa rozsyłanie grupowe. Więcej informacji i rozwiązywanie problemów zawiera sekcja Testowanie aplikacji rozsyłania grupowego w sieci bez rozsyłania grupowego i Ustawianie odpowiedniej sieci dla ruchu rozsyłania grupowego .

FeedbackMode**NACK**

Informacje zwrotne za pomocą ujemnych potwierdzeń. Jest to wartość domyślna.

ACK

Informacje zwrotne za pomocą pozytywnych potwierdzeń.

WAIT1

Sprzężenie zwrotne za pomocą pozytywnych potwierdzeń, w których nadajnik oczekuje tylko na 1 ACK od dowolnego odbiorców.

HeartbeatTimeout

Limit czasu pulsu w milisekundach. Wartość 0 wskazuje, że zdarzenia przekroczenia limitu czasu pulsu nie są zgłaszane przez odbiornik lub odbiorniki tematu. Wartością domyślną jest 20000.

HeartbeatInterval

Odstęp czasu pulsu w milisekundach. Wartość 0 oznacza, że pulsy nie są wysyłane. Przedział czasu pulsu musi być znacznie mniejszy niż wartość **HeartbeatTimeout**, aby uniknąć fałszywych zdarzeń limitu czasu pulsu. Wartość domyślna to 2000.

Współdziałanie rozsyłania grupowego z usługą IBM MQ komunikatów w trybie niskim

Te informacje umożliwiają zrozumienie współdziałania między programem IBM MQ Multicast i IBM MQ Low Latency Messaging (LLM).

Podstawowe przesyłanie ładunku jest możliwe dla aplikacji za pomocą LLM, z inną aplikacją używaniem rozsyłania grupowego w celu wymiany komunikatów w obu kierunkach. Chociaż rozsyłanie grupowe korzysta z technologii LLM, sam produkt LLM nie jest osadzony. W związku z tym możliwe jest zainstalowanie zarówno programu LLM, jak i IBM MQ Multicast, a także obsługi i serwisowania obu produktów oddzielnie.

Aplikacje LLM, które komunikują się z rozsyłaniem grupowym, mogą wymagać wysyłania i odbierania właściwości komunikatów. Właściwości komunikatu produktu IBM MQ i pola MQMD są przesyłane jako właściwości komunikatu LLM z określonymi kodami właściwości komunikatu LLM, jak pokazano w poniższej tabeli:

Właściwość IBM MQ	Typ właściwości LLM produktu IBM MQ	Rodzaj nieruchomości LLM	Kod właściwości LLM
MQMD.Report	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1001
MQMD.MsgType	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1002
MQMD.Expiry	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1003
MQMD.Feedback	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1004
MQMD.Encoding	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1005
MQMD.CodedCharSetId	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1006
MQMD.Format	RMM_MSG_PROP_BYTES	łańcuch właściwości LLM_PROP_KIND_String	-1007
MQMD.Priority	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1008
MQMD.Persistence	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1009
MQMD.MsgId	RMM_MSG_PROP_BYTES	LLM_PROP_KIND_ByteArray	-1010
MQMD.BackoutCount	RMM_MSG_PROP_INT32	LLM_PROP_KIND_Int32	-1012
MQMD.ReplyToQ	RMM_MSG_PROP_BYTES	łańcuch właściwości LLM_PROP_KIND_String	-1013
MQMD.ReplyToQMger	RMM_MSG_PROP_BYTES	łańcuch właściwości LLM_PROP_KIND_String	-1014
MQMD.PutDate	RMM_MSG_PROP_BYTES	łańcuch właściwości LLM_PROP_KIND_String	-1020
MQMD.PutTime	RMM_MSG_PROP_BYTES	łańcuch właściwości LLM_PROP_KIND_String	-1021

Tabela 18. Właściwości komunikatu produktu IBM MQ do odwzorowań właściwości LLM produktu IBM MQ (kontynuacja)

Właściwość IBM MQ	Typ właściwości LLM produktu IBM MQ	Rodzaj nieruchomości LLM	Kod właściwości LLM
MQMD.ApplOriginData	RMM_MSG_PROP_BYTES	łańcuch właściwości LLM_PROP_KIND_String	-1022
MQPubOptions	RMM_MSG_PROP_INT32	LLM_PROP_KIND_int32	-1053

Więcej informacji na temat LLM można znaleźć w dokumentacji produktu LLM: [IBM MQ Low Latency Messaging](#)(Przesyłanie komunikatów o niskim opóźnieniu).

IBM i administrowanie IBM MQ for IBM i

Wprowadza dostępne metody administrowania produktem IBM MQ w systemie IBM i.

Do zadań administracyjnych należą: tworzenie, uruchamianie, modyfikowanie, wyświetlanie, zatrzymywanie i usuwanie klastrów, procesów i obiektów produktu IBM MQ (menedżery kolejek, kolejki, listy nazw, definicje procesów, kanały, kanały połączeń klientów, obiekty nastuchiwania, usługi i obiekty informacji uwierzytelniających).

Szczegółowe informacje na temat administrowania produktem IBM MQ for IBM i znajdują się w następujących odsyłaczach:

- [“Zarządzanie produktem IBM MQ for IBM i za pomocą komend CL” na stronie 253](#)
- [“Alternatywne sposoby administrowania produktem IBM MQ for IBM i” na stronie 267](#)
- [“Zarządzanie pracą dla produktu IBM i” na stronie 272](#)

Pojęcia pokrewne

[“Dostępność, tworzenie kopii zapasowych, odtwarzanie i restartowanie w systemie IBM i” na stronie 280](#)
Informacje zawarte w tej sekcji pozwalają zrozumieć, w jaki sposób produkt IBM MQ for IBM i korzysta z funkcji obsługi kronikowania IBM i w celu wspomagania jej tworzenia i odtwarzania strategii.

Odsyłacze pokrewne

[“WYCISZANIE IBM MQ for IBM i” na stronie 325](#)

W tej sekcji opisano sposób wyciszania (na koniec z wdziękiem) IBM MQ for IBM i.

Informacje pokrewne

[Zmiana informacji konfiguracyjnych w systemie IBM i](#)

[Informacje o nazwach bibliotek menedżera kolejek produktu IBM MQ for IBM i](#)

[Konfigurowanie zabezpieczeń w systemie IBM i](#)

[Procedura obsługi kolejki niedostarczanych komunikatów w systemie IBM i](#)

[Określanie problemów z aplikacjami produktu IBM MQ for IBM i](#)

[Instalowalne usługi i komponenty w systemie IBM i](#)

[Obiekty systemowe i domyślne w systemie IBM i](#)

IBM i Zarządzanie produktem IBM MQ for IBM i za pomocą komend CL

Informacje zawarte w tej sekcji umożliwiają zrozumienie komend IBM MQ IBM i.

Do większości grup komend produktu IBM MQ, w tym komend powiązanych z menedżerami kolejek, kolejkami, tematami, kanałami, listami nazw, definicjami procesów i obiektami informacji uwierzytelniających, można uzyskać dostęp za pomocą odpowiedniej komendy **WRK***.

Główną komendą w zestawie jest **WRKMQM**. Ta komenda umożliwia, na przykład, wyświetlenie listy wszystkich menedżerów kolejek w systemie wraz z informacjami o statusie. Alternatywnie można przetworzyć wszystkie komendy specyficzne dla menedżera kolejek, korzystając z różnych opcji dla każdej pozycji.

Za pomocą komendy **WRKMQM** można wybrać konkretne obszary każdego menedżera kolejek, na przykład pracę z kanałami, tematami lub kolejkami, a także z poziomu wybranych obiektów wybranych.

Rejestrowanie definicji aplikacji produktu IBM MQ

Podczas tworzenia lub dostosowywania aplikacji produktu IBM MQ przydatne jest zachowanie rekordu ze wszystkich utworzonych definicji produktu IBM MQ . Ten rekord może być używany dla:

- Cele odtwarzania
- Konserwacja
- Wycofywanie aplikacji IBM MQ

Definicje aplikacji produktu IBM MQ można rejestrować w 1 z 2 sposobów:

1. Tworzenie programów w języku CL w celu wygenerowania definicji IBM MQ dla serwera.
2. Tworzenie plików tekstowych MQSC jako elementów SRC w celu wygenerowania definicji produktu IBM MQ przy użyciu języka komend wieloplatformowych IBM MQ .

Szczegółowe informacje na temat definiowania obiektów kolejki można znaleźć w sekcji [“Komendy skryptowe \(MQSC\)”](#) na stronie 11 i [“Korzystanie z formatów komend programowalnych IBM MQ”](#) na stronie 23.

Informacje pokrewne

Skorowidz komend CL produktu IBM MQ for IBM i

Przed rozpoczęciem korzystania z programu IBM MQ for IBM i za pomocą komend CL

Te informacje umożliwiają uruchomienie podsystemu IBM MQ i utworzenie lokalnego menedżera kolejek.

Zanim rozpocznie

Upewnij się, że podsystem IBM MQ jest uruchomiony (za pomocą komendy STRSBS QMQM/QMQM) i że kolejka zadań skojarzona z tym podsystemem nie jest wstrzymana. Domyślnie podsystem IBM MQ i kolejka zadań mają nazwę QMQM w bibliotece QMQM.

O tym zadaniu

Uruchamianie menedżera kolejek za pomocą wiersza komend produktu IBM i

Procedura

1. Utwórz lokalny menedżer kolejek, wydając komendę CRTMQM z wiersza komend systemu IBM i .
Podczas tworzenia menedżera kolejek dostępna jest opcja tworzenia menedżera kolejek jako domyślnego menedżera kolejek. Domyślnym menedżerem kolejek (którego może być tylko jeden) jest menedżer kolejek, do którego ma zastosowanie komenda CL, jeśli pominięto parametr nazwy menedżera kolejek (MQMNAME).
2. Uruchom lokalny menedżer kolejek, wydając komendę STRMQM z wiersza komend systemu IBM i .
Jeśli uruchomienie menedżera kolejek trwa dłużej niż kilka sekund, program IBM MQ będzie wyświetlał komunikaty o statusie w sposób sporny, co spowoduje, że postęp uruchamiania zostanie szczegółowo wyświetlony. Więcej informacji na temat tych komunikatów zawiera sekcja [Komunikaty i kody przyczyny](#).

Co dalej

Menedżer kolejek można zatrzymać, wydając komendę ENDMQM z wiersza komend systemu IBM i , a następnie sterując menedżerem kolejek, wydając inne komendy IBM MQ z wiersza komend IBM i .

Zdalne menedżery kolejek nie mogą być uruchamiane zdalnie, ale muszą być tworzone i uruchamiane w swoich systemach przez lokalnych operatorów. Wyjątkiem jest miejsce, w którym istnieją zdalne obiekty operacyjne (poza programem IBM MQ for IBM i), które umożliwiają wykonywanie takich operacji.

Administrator kolejki lokalnej nie może zatrzymać zdalnego menedżera kolejek.

Uwaga: W ramach wygaszania systemu IBM MQ należy wyciszyć aktywne menedżery kolejek. Jest to opisane w sekcji [“WYCISZANIE IBM MQ for IBM i”](#) na stronie 325.

IBM i Tworzenie obiektów produktu IBM MQ for IBM i

Te informacje umożliwiają zapoznanie się z metodami tworzenia obiektów IBM MQ dla produktu IBM i.

Zanim rozpoczniesz

Poniższe zadania sugerują różne sposoby korzystania z programu IBM MQ for IBM i z poziomu wiersza komend.

O tym zadaniu

Istnieją dwie metody online do tworzenia obiektów IBM MQ, które są następujące:

Procedura

1. Za pomocą komendy Create (Tworzenie), na przykład: **Create MQM Queue**, komenda: **CRTMQMQ**
2. Za pomocą komendy Praca z obiektem MQM, po której następuje komenda F6, na przykład: komenda **Work with MQM Queues : WRKMQMQ**

Co dalej

Listę wszystkich komend można znaleźć w sekcji [Komendy CL programu IBM MQ for IBM i](#).

Uwaga: Wszystkie komendy MQM można składać z menu Komendy menedżera kolejek komunikatów. Aby wyświetlić to menu, wpisz GO CMDMQM w wierszu komend i naciśnij klawisz Enter.

Po wybraniu komendy z tego menu system automatycznie wyświetli panel podpowiedzi. Aby wyświetlić panel podpowiedzi dla komendy wpisanej bezpośrednio w wierszu komend, należy nacisnąć klawisz F4 przed naciśnięciem klawisza Enter.

Tworzenie kolejki lokalnej przy użyciu komendy CRTMQMQ

Procedura

1. Wpisz CHGMQM w wierszu komend i naciśnij klawisz F4.
2. Na panelu **Tworzenie kolejki MQM** (Create MQM Queue panel) wpisz nazwę kolejki, która ma zostać utworzona w polu Queue name (Utwórz kolejkę). Aby określić mieszaną nazwę elementu pracy, należy ująć nazwę w apostrofach.
3. Wpisz *LCL w polu Queue type.
4. Podaj nazwę menedżera kolejek, o ile nie jest używany domyślny menedżer kolejek, i naciśnij klawisz Enter. Można wpisać dowolną z wartości, używając nowej wartości. Przewiń do przodu, aby wyświetlić kolejne pola. Opcje używane dla klastrów znajdują się na końcu listy opcji.
5. Po zmianie jakichkolwiek wartości należy nacisnąć klawisz Enter, aby utworzyć kolejkę.

Tworzenie kolejki lokalnej za pomocą komendy WRKMQMQ

Procedura

1. W wierszu komend wpisz WRKMQMQ.
2. Wprowadź nazwę menedżera kolejek.

3. Jeśli chcesz wyświetlić panel podpowiedzi, naciśnij klawisz F4. Panel podpowiedzi jest przydatny w celu zmniejszenia liczby wyświetlanych kolejek poprzez określenie ogólnej nazwy kolejki lub typu kolejki.
4. Należy nacisnąć klawisz Enter , a następnie zostanie wyświetlony panel **Praca z kolejkami MQM** . Istnieje możliwość nadpisania dowolnej wartości z nową wartością. Przewiń do przodu, aby wyświetlić kolejne pola. Opcje używane dla klastrów znajdują się na końcu listy opcji.
5. Aby utworzyć nową kolejkę, należy nacisnąć klawisz F6 . W tym celu należy wybrać panel **CRTMQMQ** . Informacje na temat tworzenia kolejki zawiera sekcja [“Tworzenie kolejki lokalnej przy użyciu komendy CRTMQMQ”](#) na stronie 255 . Po utworzeniu kolejki ponownie zostanie wyświetlony panel **Praca z kolejkami MQM** . Nowa kolejka zostanie dodana do listy po naciśnięciu przycisku F5=Refresh.

Zmiana atrybutów menedżera kolejek

O tym zadaniu

Aby zmienić atrybuty menedżera kolejek określonego w komendzie **CHGMQM** , należy określić atrybuty i wartości, które mają zostać zmienione. Na przykład, aby zmienić atrybuty produktu `jupiter.queue.manager`, należy użyć następujących opcji:

Procedura

Wpisz **CHGMQM** w wierszu komend i naciśnij klawisz F4 .

Wyniki

Komenda zmienia używaną kolejkę niedostarczonych komunikatów i umożliwia zahamowanie zdarzeń.

IBM i

Praca z kolejkami lokalnymi w systemie IBM i

Ta sekcja zawiera przykłady niektórych komend, których można użyć do zarządzania kolejkami lokalnymi. Wszystkie wyświetlane komendy są również dostępne za pomocą opcji z poziomu panelu komend **WRKMQMQ**.

Definiowanie kolejki lokalnej

W przypadku aplikacji lokalny menedżer kolejek jest menedżerem kolejek, z którym połączona jest aplikacja. Kolejkami zarządzanymi przez lokalny menedżer kolejek są dane lokalne względem tego menedżera kolejek.

Użyj komendy **CRTMQMQ QTYPE *LCL** , aby utworzyć definicję kolejki lokalnej, a także utworzyć strukturę danych, która jest wywoływana jako kolejka. Parametry kolejki można także modyfikować z poziomu domyślnej kolejki lokalnej.

W tym przykładzie definiowana przez nas kolejka, `orange.local.queue`, ma następujące cechy:

- Jest on włączony dla operacji pobierania, jest wyłączony dla operacji put i działa na zasadzie FIFO (first-in-first-first-out).
- Jest to kolejka *zwykła* , to znaczy nie jest kolejką inicjującą ani kolejką transmisji i nie generuje komunikatów wyzwacza.
- Maksymalna głębokość kolejki wynosi 1000 komunikatów. Maksymalna długość komunikatu to 2000 bajtów.

Następująca komenda wykonuje tę komendę w domyślnym menedżerze kolejek:

```
CRTMQMQ QNAME('orange.local.queue') QTYPE(*LCL)
TEXT('Queue for messages from other systems')
PUTENBL(*NO)
GETENBL(*YES)
TRGENBL(*NO)
MSGDLYSEQ(*FIFO)
MAXDEPTH(1000)
```


Uwaga:

1. USAGE *NORMAL wskazuje, że ta kolejka nie jest kolejką transmisji.
2. Jeśli w tym samym menedżerze kolejek znajduje się już kolejka lokalna o nazwie `orange.local.queue`, wykonanie tej komendy nie powiedzie się. Użyj atrybutu REPLACE *YES, jeśli chcesz nadpisać istniejącą definicję kolejki, ale zobacz także [“Zmiana atrybutów kolejki lokalnej”](#) na stronie 258.

Definiowanie kolejki niedostarczonych komunikatów

Każdy menedżer kolejek musi mieć kolejkę lokalną, która ma być używana jako kolejka niedostarczonych komunikatów, dzięki czemu komunikaty, których nie można dostarczyć do ich poprawnego miejsca docelowego, mogą zostać zapisane w celu późniejszego pobrania. Należy jawnie przekazać menedżerowi kolejek informacje o kolejce niedostarczonych komunikatów. W tym celu należy podać kolejkę niedostarczonych komunikatów w komendzie **CRTMQM** lub użyć komendy **CHGMQM** w celu określenia jednego z nich później. Należy również zdefiniować kolejkę niedostarczonych komunikatów, zanim będzie można jej użyć.

Wraz z produktem dostarczana jest przykładowa kolejka niedostarczonych komunikatów o nazwie `SYSTEM.DEAD.LETTER.QUEUE`. Ta kolejka jest tworzona automatycznie podczas tworzenia menedżera kolejek. Jeśli jest to wymagane, można zmodyfikować tę definicję. Nie ma potrzeby zmieniania jej nazwy, chociaż można je określić.

Kolejka niedostarczonych komunikatów nie ma specjalnych wymagań, z wyjątkiem sytuacji, gdy:

- Musi to być kolejka lokalna.
- Atrybut MAXMSGL (maksymalna długość komunikatu) musi włączyć kolejkę w celu uwzględnienia największych komunikatów, które menedżer kolejek musi obsłużyć **plus** wielkość nagłówka niedostarczonych komunikatów (MQDLH).

Produkt IBM MQ udostępnia procedurę obsługi kolejki niedostarczonych komunikatów, która umożliwia określenie sposobu, w jaki komunikaty znalezione w kolejce niedostarczonych komunikatów mają być przetwarzane lub usuwane. Więcej informacji na ten temat zawiera sekcja [Procedura obsługi kolejki niedostarczonych komunikatów w produkcie IBM MQ for IBM i](#).

Wyświetlanie domyślnych atrybutów obiektów

Podczas definiowania obiektu IBM MQ przyjmuje on wszystkie atrybuty, które nie zostały określone na podstawie obiektu domyślnego. Na przykład podczas definiowania kolejki lokalnej kolejka dziedziczy wszystkie atrybuty, które są pomijane w definicji z domyślnej kolejki lokalnej, która jest nazywana `SYSTEM.DEFAULT.LOCAL.QUEUE`. Aby dokładnie sprawdzić, jakie są te atrybuty, należy użyć następującej komendy:

```
DSPMQMQ QNAME(SYSTEM.DEFAULT.LOCAL.QUEUE) MQMNAME(MYQUEUEMANAGER)
```

Kopiowanie definicji kolejki lokalnej

Definicję kolejki można skopiować za pomocą komendy `CPYMQMQ`. Na przykład:

```
CPYMQMQ FROMQ('orange.local.queue') TOQ('magenta.queue') MQMNAME(MYQUEUEMANAGER)
```

Ta komenda tworzy kolejkę o takich samych atrybutach, co oryginalna kolejka `orange.local.queue`, a nie domyślna kolejka lokalna systemu.

Można również użyć komendy **CPYMQMQ**, aby skopiować definicję kolejki, ale zastępując jedną lub więcej zmian w atrybutach oryginału. Na przykład:

```
CPYMQMQ FROMQ('orange.local.queue') TOQ('third.queue') MQMNAME(MYQUEUEMANAGER)
MAXMSGLEN(1024)
```

Ta komenda kopiuje atrybuty kolejki `orange.local.queue` do kolejki `third.queue`, ale określa, że maksymalna długość komunikatu w nowej kolejce ma wynosić 1024 bajty, a nie 2000.

Uwaga: Gdy używana jest komenda **CPYMQMQ**, kopiowane są tylko atrybuty kolejki, a nie komunikaty w kolejce.

Zmiana atrybutów kolejki lokalnej

Atrybuty kolejki można zmieniać na dwa sposoby, używając komendy **CHGMQMQ** lub komendy **CPYMQMQ** z atrybutem `REPLACE *YES`. W programie [“Definiowanie kolejki lokalnej”](#) na stronie 256zdefiniowano kolejkę `orange.local.queue`. Jeśli, na przykład, należy zwiększyć maksymalną długość komunikatu w tej kolejce do 10 000 bajtów.

- Za pomocą komendy **CHGMQMQ**:

```
CHGMQMQ QNAME('orange.local.queue') MQMNAME(MYQUEUEMANAGER) MAXMSGLEN(10000)
```

Ta komenda zmienia jeden atrybut o maksymalną długość komunikatu. Wszystkie pozostałe atrybuty pozostają takie same.

- Za pomocą komendy **CRTMQMQ** z opcją `REPLACE *YES`, na przykład:

```
CRTMQMQ QNAME('orange.local.queue') QTYPE(*LCL) MQMNAME(MYQUEUEMANAGER)
MAXMSGLEN(10000) REPLACE(*YES)
```

Ta komenda zmienia nie tylko maksymalną długość komunikatu, ale wszystkie inne atrybuty, które mają podane wartości domyślne. Kolejka jest teraz włączona, podczas gdy poprzednio została ona zablokowana. Włączona opcja `put` jest wartością domyślną, określoną przez kolejkę `SYSTEM.DEFAULT.LOCAL.QUEUE`, chyba że została ona zmieniona.

Jeśli **zmniejsz** maksymalną długość komunikatu w istniejącej kolejce, to istniejące komunikaty nie będą miały wpływu na tę wartość. Wszystkie nowe komunikaty muszą jednak spełniać nowe kryteria.

Czyszczenie kolejki lokalnej

Aby usunąć wszystkie komunikaty z kolejki lokalnej o nazwie `magenta.queue`, należy użyć następującej komendy:

```
CLRMQMQ QNAME('magenta.queue') MQMNAME(MYQUEUEMANAGER)
```

Nie można wyczyścić kolejki, jeśli:

- Istnieją niezatwierdzone komunikaty, które zostały umieszczone w kolejce w punkcie synchronizacji.
- Aplikacja aktualnie ma otwartą kolejkę.

Usuwanie kolejki lokalnej

Aby usunąć kolejkę lokalną, należy użyć komendy **DLTMQMQ**.

Kolejka nie może zostać usunięta, jeśli ma na niej niezatwierdzone komunikaty lub jeśli jest używana.

Włączanie dużych kolejek

IBM MQ obsługuje kolejki większe niż 2 GB. Informacje na temat włączania obsługi dużych plików w programie IBM i można znaleźć w dokumentacji systemu operacyjnego.

Dokumentację produktu IBM i można znaleźć w podręczniku [IBM Documentation](#).

Niektóre programy narzędziowe mogą nie być w stanie obsłużyć plików większych niż 2 GB. Przed włączeniem obsługi dużych plików należy sprawdzić w dokumentacji systemu operacyjnego informacje o ograniczeniach dotyczących takiego wsparcia.

IBM i Praca z kolejkami aliasami w systemie IBM i

Ta sekcja zawiera przykłady niektórych komend, których można użyć do zarządzania kolejkami aliasowymi. Wszystkie wyświetlane komendy są również dostępne za pomocą opcji z poziomu panelu komend **WRKMQMQ**.

Kolejka aliasowa (znana niekiedy jako alias kolejki) udostępnia metodę przekierowywania wywołań MQI. Kolejka aliasowa nie jest kolejką rzeczywistą, ale definicją, która jest tłumaczona na rzeczywistą kolejkę. Definicja kolejki aliasowej zawiera nazwę kolejki docelowej, która jest określona przez atrybut TGTQNAME .

Gdy aplikacja określa kolejkę aliasową w wywołaniu MQI, menedżer kolejek rozstrzyga rzeczywistą nazwę kolejki w czasie wykonywania.

Na przykład aplikacja została opracowana w celu umieszczenia komunikatów w kolejce o nazwie `my.alias.queue`. Określa ona nazwę tej kolejki, gdy tworzy żądanie **MQOPEN** , a pośrednio, jeśli umieszcza komunikat w tej kolejce. Aplikacja nie ma informacji o tym, że kolejka jest kolejką aliasową. Dla każdego wywołania MQI za pomocą tego aliasu menedżer kolejek rozstrzyga rzeczywistą nazwę kolejki, która może być kolejką lokalną lub zdalną zdefiniowaną w tym menedżerze kolejek.

Zmieniając wartość atrybutu TGTQNAME , można przekierować wywołania MQI do innej kolejki, prawdopodobnie w innym menedżerze kolejek. Jest to przydatne w przypadku konserwacji, migracji i równoważenia obciążenia.

Definiowanie kolejki aliasowej

Następująca komenda tworzy kolejkę aliasową:

```
CRTMQMQ QNAME('my.alias.queue') QTYPE(*ALS) TGTQNAME('yellow.queue')
MQMNAME(MYQUEUEMANAGER)
```

Ta komenda przekierowuje wywołania MQI, które określają `my.alias.queue` do kolejki `yellow.queue`. Komenda nie tworzy kolejki docelowej; wywołania MQI nie powiodą się, jeśli kolejka `yellow.queue` nie istnieje w czasie wykonywania.

Jeśli zmienisz definicję aliasu, możesz przekierować wywołania MQI do innej kolejki. Na przykład:

```
CHGMQM QNAME('my.alias.queue') TGTQNAME('magenta.queue') MQMNAME(MYQUEUEMANAGER)
```

Ta komenda przekierowuje wywołania MQI do innej kolejki, `magenta.queue`.

Można również użyć kolejek aliasowych, aby utworzyć pojedynczą kolejkę (kolejka docelowa), która ma różne atrybuty dla różnych aplikacji. W tym celu należy zdefiniować dwa aliasy, po jednym dla każdej aplikacji. Załóżmy, że istnieją dwie aplikacje:

- Aplikacja ALPHA może umieszczać komunikaty w systemie `yellow.queue`, ale nie ma uprawnień do pobierania z niego komunikatów.
- Aplikacja BETA może pobrać komunikaty z produktu `yellow.queue`, ale nie może umieszczać na nim komunikatów.

Można to zrobić za pomocą następujących komend:

```
/* This alias is put enabled and get disabled for application ALPHA */
CRTMQMQ QNAME('alphas.alias.queue') QTYPE(*ALS) TGTQNAME('yellow.queue')
PUTENBL(*YES) GETENBL(*NO) MQMNAME(MYQUEUEMANAGER)

/* This alias is put disabled and get enabled for application BETA */
CRTMQMQ QNAME('betas.alias.queue') QTYPE(*ALS) TGTQNAME('yellow.queue')
PUTENBL(*NO) GETENBL(*YES) MQMNAME(MYQUEUEMANAGER)
```

Opcja ALPHA używa nazwy kolejki `alphas.alias.queue` w jej wywołaniach MQI; BETA używa nazwy kolejki `betas.alias.queue`. Oba mają dostęp do tej samej kolejki, ale na różne sposoby.

Atrybut `REPLACE *YES` można używać podczas definiowania kolejek aliasowych, w taki sam sposób, w jaki używane są te atrybuty z kolejkami lokalnymi.

Korzystanie z innych komend z kolejkami aliasami

Do wyświetlenia lub zmiany atrybutów kolejki aliasowej można użyć odpowiednich komend. Na przykład:

```
* Display the alias queue's attributes */
DSPMQMQ QNAME('alphas.alias.queue') MQMNAME(MYQUEUEMANAGER)

/* ALTER the base queue name, to which the alias resolves. */
/* FORCE = Force the change even if the queue is open. */
CHQMCMQ QNAME('alphas.alias.queue') TGTQNAME('orange.local.queue') FORCE(*YES)
MQMNAME(MYQUEUEMANAGER)
```



Praca z kolejkami modelami w systemie IBM i

Ta sekcja zawiera przykłady niektórych komend, których można użyć do zarządzania kolejkami modelowymi. Wszystkie wyświetlane komendy są również dostępne za pomocą opcji z poziomu panelu komend **WRKMQMQ**.

Menedżer kolejek tworzy kolejkę dynamiczną, jeśli odbierze wywołanie MQI z aplikacji określając nazwę kolejki, która została zdefiniowana jako kolejka modelowa. Nazwa nowej kolejki dynamicznej jest generowana przez menedżer kolejek po utworzeniu kolejki. Kolejka modelowa jest szablonem, który określa atrybuty wszystkich kolejek dynamicznych utworzonych na podstawie tej kolejki.

Kolejki modelowe stanowią wygodną metodę tworzenia kolejek, ponieważ są one wymagane.

Definiowanie kolejki modelowej

Kolejkę modelową można zdefiniować za pomocą zestawu atrybutów w taki sam sposób, w jaki definiuje się kolejkę lokalną. Kolejki modelowe i kolejki lokalne mają ten sam zestaw atrybutów, z tym wyjątkiem, że w kolejkach modelowych można określić, czy utworzone kolejki dynamiczne mają charakter tymczasowy, czy trwałe. (Trwałe kolejki są obsługiwane przez restarty menedżera kolejek, tymczasowe nie są). Na przykład:

```
CRTMQMQ QNAME('green.model.queue') QTYPE(*MDL) DFNTYPE(*PERMDYN)
```

Ta komenda tworzy definicję kolejki modelowej. Z atrybutu `DFNTYPE` rzeczywiste kolejki utworzone na podstawie tego szablonu są trwałymi kolejkami dynamicznymi. Atrybuty, które nie zostały określone, są automatycznie kopiowane z domyślnej kolejki produktu `SYSYSTEM.DEFAULT.MODEL.QUEUE`.

Atrybutu `REPLACE *YES` można używać podczas definiowania kolejek modelowych w taki sam sposób, w jaki są używane z kolejkami lokalnymi.

Korzystanie z innych komend z kolejkami modelami

Za pomocą odpowiednich komend można wyświetlać lub zmieniać atrybuty kolejki modelowej. Na przykład:

```
/* Display the model queue's attributes */
DSPMQMQ MQMNAME(MYQUEUEMANAGER) QNAME('green.model.queue')

/* ALTER the model queue to enable puts on any */
/* dynamic queue created from this model. */
CHGMQM MQMNAME(MYQUEUEMANAGER) QNAME('blue.model.queue') PUTENBL(*YES)
```

IBM i

Praca z wyzwalaniem w systemie IBM i

Ta sekcja zawiera informacje na temat wyzwalania i przetwarzania definicji procesów.

Produkt IBM MQ udostępnia narzędzie do automatycznego uruchamiania aplikacji, gdy spełnione są określone warunki w kolejce. Jednym z przykładów warunków jest to, że liczba komunikatów w kolejce osiągnie określoną liczbę. Ten obiekt jest nazywany *wyzwalaniem* i jest szczegółowo opisany w sekcji [Kanały wyzwalane](#).

Co to jest wyzwalanie?

Menedżer kolejek definiuje pewne warunki, które tworzą zdarzenia wyzwalające. Jeśli wyzwalanie jest włączone dla kolejki i wystąpi zdarzenie wyzwalające, menedżer kolejek wysyła komunikat wyzwalacza do kolejki o nazwie kolejka inicjujący. Obecność komunikatu wyzwalacza w kolejce inicjuj. wskazuje, że wystąpiło zdarzenie wyzwalające.

Komunikaty wyzwalacza wygenerowane przez menedżer kolejek nie są trwałe. Dzięki temu możliwe jest zmniejszenie rejestrowania (w ten sposób poprawianie wydajności) i minimalizacja duplikatów podczas restartu, a więc poprawa czasu restartu.

Co to jest monitor wyzwalacza?

Program, który przetwarza kolejkę inicjujący, jest nazywany aplikacją monitora wyzwalacza, a jego funkcją jest odczytanie komunikatu wyzwalacza i podjęcie odpowiednich działań w oparciu o informacje zawarte w komunikacie wyzwalacza. Zazwyczaj to działanie polegałoby na uruchomieniu innej aplikacji w celu przetworzenia kolejki, która spowodowała wygenerowanie komunikatu wyzwalacza. Z punktu widzenia menedżera kolejek nie ma nic specjalnego w przypadku aplikacji wyzwalacza-monitor- jest to kolejna aplikacja, która odczytuje komunikaty z kolejki (kolejka inicjuj).

Zmiana atrybutów wprowadzania zadania w monitorze wyzwalacza

Monitor wyzwalacza podany jako komenda **STRMQMTRM** wprowadza zadanie dla każdego komunikatu wyzwalacza, korzystając z domyślnego opisu zadania systemowego QDFTJOB. Ma to ograniczenia w tym, że wprowadzone zadania są zawsze nazywane QDFTJOB i mają atrybuty domyślnego opisu zadania wraz z listą bibliotek *SYSVAL. Produkt IBM MQ udostępnia metodę przesłaniania tych atrybutów. Na przykład można dostosować wprowadzone zadania w taki sposób, aby miały bardziej znaczące nazwy zadań:

1. W opisie zadania podaj opis, który ma być używany, na przykład wartości protokołowania.
2. Określ dane środowiska definicji procesu używanej w procesie wyzwalającym:

```
CHGMQMPRC PRCNAME(MY_PROCESS) MQMNAME(MHA3) ENVDATA ('JOB(MYLIB/TRIGJOB)')
```

Monitor wyzwalacza wykonuje SBMJOB , korzystając z podanego opisu.

Istnieje możliwość nadpisania innych atrybutów SBMJOB poprzez określenie odpowiedniego słowa kluczowego i wartości w danych środowiskowych definicji procesu. Jedynym wyjątkiem tego parametru jest słowo kluczowe CMD , ponieważ ten atrybut jest wypełniany przez monitor wyzwalacza. Przykład

komendy do określenia danych środowiskowych definicji procesu, w której zarówno nazwa zadania, jak i opis mają być zmienione:

```
CHGMQMPRC PRCNAME(MY_PROCESS) MQMNAME(MHA3) ENVDATA ('JOB(MYLIB/TRIGJOB)
JOB(TRIGGER)')
```

Definiowanie kolejki aplikacji na potrzeby wyzwalania

Kolejka aplikacji jest kolejką lokalną, która jest używana przez aplikacje do przesyłania komunikatów, przez interfejs MQI. Wyzwalanie wymaga zdefiniowania liczby atrybutów kolejki w kolejce aplikacji. Samo wyzwołanie jest włączone przez atrybut TRGENBL.

W tym przykładzie zdarzenie wyzwalające ma być generowane, gdy w kolejce lokalnej `motor.insurance.queue` znajduje się 100 komunikatów o priorytecie 5 lub wyższym, w następujący sposób:

```
CRTMQMQ MQMNAME(MYQUEUEMANAGER) QNAME('motor.insurance.queue') QTYPE(*LCL)
PRCNAME('motor.insurance.quote.process') MAXMSGLEN(2000)
DFTMSGPST(*YES) INITQNAME('motor.ins.init.queue')
TRGENBL(*YES) TRGTYPE(*DEPTH) TRGDEPTH(100) TRGMSGPTY(5)
```

gdzie parametry są następujące:

MQMNAME(MYQUEUEMANAGER)

Nazwa menedżera kolejek.

QNAME('motor.insurance.queue')

Nazwa definiowanej kolejki aplikacji.

PRCNAME('motor.insurance.quote.process')

Nazwa aplikacji, która ma zostać uruchomiona przez program monitora wyzwalacza.

MAXMSGLEN(2000)

Maksymalna długość komunikatów w kolejce.

DFTMSGPST(*YES)

Komunikaty w tej kolejce są domyślnie trwałe.

INITQNAME('motor.ins.init.queue')

Nazwa kolejki inicjuj., w której menedżer kolejek ma umieścić komunikat wyzwalacza.

TRGENBL(*YES)

Wartość atrybutu wyzwalacza.

TRGTYPE(*DEPTH)

Zdarzenie wyzwalające jest generowane, gdy liczba komunikatów o wymaganym priorytecie (**TRGMSGPTY**) osiągnie liczbę określoną w **TRGDEPTH**.

TRGDEPTH(100)

Liczba komunikatów wymaganych do wygenerowania zdarzenia wyzwalającego.

TRGMSGPTY(5)

Priorytet komunikatów, które mają być zliczane przez menedżera kolejek przy podejmowaniu decyzji o tym, czy ma zostać wygenerowane zdarzenie wyzwalające. Zliczane są tylko komunikaty o priorytecie 5 lub wyższym.

Definiowanie kolejki inicjuj

Gdy wystąpi zdarzenie wyzwalające, menedżer kolejek umieszcza komunikat wyzwalacza w kolejce inicjuj. określonej w definicji kolejki aplikacji. Kolejki inicjuj nie mają specjalnych ustawień, ale można użyć następującej definicji kolejki lokalnej `motor.ins.init.queue` dla wskazówek:

```
CRTMQMQ MQMNAME(MYQUEUEMANAGER) QNAME('motor.ins.init.queue') QTYPE(*LCL)
GETENBL(*YES) SHARE(*NO) TRGTYPE(*NONE)
```

Tworzenie definicji procesu

Aby utworzyć definicję procesu, należy użyć komendy **CRTMQMPC**. Definicja procesu wiąże kolejkę aplikacji z aplikacją, która ma przetwarzać komunikaty z kolejki. Jest to realizowane za pomocą atrybutu PRCNAME w kolejce aplikacji motor.insurance.queue. Następująca komenda tworzy wymagany proces, motor.insurance.quote.process, zidentyfikowany w tym przykładzie:

```
CRTMQMPC MQMNAME(MYQUEUEMANAGER) PRCNAME('motor.insurance.quote.process')  
TEXT('Insurance request message processing')  
APPTYPE(*OS400) APPID(MQTEST/TESTPROG)  
USRDATA('open, close, 235')
```

gdzie parametry są następujące:

MQMNAME(MYQUEUEMANAGER)

Nazwa menedżera kolejek.

PRCNAME('motor.insurance.quote.process')

Nazwa definicji procesu.

TEXT('Insurance request message processing')

Opis programu aplikacji, do którego odnosi się ta definicja. Ten tekst jest wyświetlany, gdy używana jest komenda **DSPMQMPC**. Może to pomóc w określeniu, co proces ma. Jeśli w łańcuchu używane są spacje, łańcuch należy ująć w pojedynczy cudzysłów.

APPTYPE(*OS400)

Typ aplikacji, która ma zostać uruchomiona.

APPID(MQTEST/TESTPROG)

Nazwa pliku wykonywalnego aplikacji, podana jako pełna nazwa pliku.

USRDATA('open, close, 235')

Dane zdefiniowane przez użytkownika, które mogą być używane przez aplikację.

Wyświetlanie definicji procesu

Aby sprawdzić wyniki definicji, należy użyć komendy **DSPMQMPC**. Na przykład:

```
MQMNAME(MYQUEUEMANAGER) DSPMQMPC('motor.insurance.quote.process')
```

Można również użyć komendy **CHGMQMPC**, aby zmienić istniejącą definicję procesu, a następnie komendę **DLTMQMPC**, aby usunąć definicję procesu.



Komunikacja między dwoma systemami IBM MQw systemie IBM i

Ten przykład kodowania ilustruje sposób konfigurowania dwóch systemów IBM MQ for IBM i za pomocą komend CL, dzięki czemu mogą komunikować się ze sobą.

Systemy te nazywane są SYSTEMA i SYSTEMB, a używany protokół komunikacyjny to TCP/IP.

Wykonaj następującą procedurę:

1. Utwórz menedżera kolejek w systemie SYSTEMA, nazywając go QMGRA1.

```
CRTMQM MQMNAME(QMGRA1) TEXT('System A - Queue +  
Manager 1') UDLMMSGQ(SYSTEM.DEAD.LETTER.QUEUE)
```

2. Uruchom ten menedżer kolejek.

```
STRMQM MQMNAME(QMGRA1)
```

3. Zdefiniuj obiekty IBM MQ w systemie SYSTEMA , które mają być wysyłane do menedżera kolejek w systemie SYSTEMB.

```
/* Transmission queue */
CRTMQMQ QNAME(XMITQ.TO.QMGRB1) QTYPE(*LCL) +
MQMNAME(QMGRA1) TEXT('Transmission Queue +
to QMGRB1') MAXDEPTH(5000) USAGE(*TMQ)

/* Remote queue that points to a queue called TARGETB */
/* TARGETB belongs to queue manager QMGRB1 on SYSTEMB */
CRTMQMQ QNAME(TARGETB.ON.QMGRB1) QTYPE(*RMT) +
MQMNAME(QMGRA1) TEXT('Remote Q pointing +
at Q TARGETB on QMGRB1 on Remote System +
SYSTEMB') RMTQNAME(TARGETB) +
RMTMQMNAME(QMGRB1) TMQNAME(XMITQ.TO.QMGRB1)

/* TCP/IP sender channel to send messages to the queue manager on SYSTEMB*/
CRTMQMCHL CHLNAME(QMGRA1.TO.QMGRB1) CHLTYPE(*SDR) +
MQMNAME(QMGRA1) TRPTYPE(*TCP) +
TEXT('Sender Channel From QMGRA1 on +
SYSTEMA to QMGRB1 on SYSTEMB') +
CONNNAME(SYSTEMB) TMQNAME(XMITQ.TO.QMGRB1)
```

4. Utwórz menedżera kolejek w systemie SYSTEMB, nazywając go QMGRB1.

```
CRTMQM MQMNAME(QMGRB1) TEXT('System B - Queue +
Manager 1') UDLMSGQ(SYSTEM.DEAD.LETTER.QUEUE)
```

5. Uruchom menedżera kolejek w systemie SYSTEMB.

```
STRMQM MQMNAME(QMGRB1)
```

6. Zdefiniuj obiekty produktu IBM MQ , które mają być odbierane przez menedżera kolejek w systemie SYSTEMA.

```
/* Local queue to receive messages on */
CRTMQMQ QNAME(TARGETB) QTYPE(*LCL) MQMNAME(QMGRB1) +
TEXT('Sample Local Queue for QMGRB1')

/* Receiver channel of the same name as the sender channel on SYSTEMA */
CRTMQMCHL CHLNAME(QMGRA1.TO.QMGRB1) CHLTYPE(*RCVR) +
MQMNAME(QMGRB1) TRPTYPE(*TCP) +
TEXT('Receiver Channel from QMGRA1 to +
QMGRB1')
```

7. Na koniec uruchom program nasłuchujący TCP/IP w systemie SYSTEMB , aby kanał mógł zostać uruchomiony. W tym przykładzie używany jest port domyślny o numerze 1414.

```
STRMQMLSR MQMNAME(QMGRB1)
```

Użytkownik jest teraz gotowy do wysyłania komunikatów testowych między produktem SYSTEMA i produktem SYSTEMB. Korzystając z jednej z dostarczonych przykładów, umieść serię komunikatów w zdalnej kolejce w systemie SYSTEMA.

Uruchom kanał w systemie SYSTEMA, korzystając z komendy **STRMQMCHL** lub używając komendy **WRKMQMCHL** i wprowadzając żądanie uruchomienia (opcja 14) dla kanału nadawczego.

Kanał powinien przejść do statusu RUNNING, a komunikaty są wysyłane do kolejki TARGETB w systemie SYSTEMB.

Sprawdź komunikaty, wydając komendę:

```
WRKMQMMSG QNAME(TARGETB) MQMNAME(QMGRB1).
```


Ten przykład zawiera przykładowy program CL programu IBM i AMQSAMP4 .

```

/*****/
/*
/* Program name: AMQSAMP4
/*
/* Description: Sample CL program defining MQM queues
/* to use with the sample programs
/* Can be run, with changes as needed, after
/* starting the MQM
/*
/* <N_OCO_COPYRIGHT> */
/* Licensed Materials - Property of IBM
/*
/* 63H9336
/* (c) Copyright IBM Corp. 1993, 2023. All Rights Reserved.
/*
/* US Government Users Restricted Rights - Use, duplication or
/* disclosure restricted by GSA ADP Schedule Contract with
/* IBM Corp.
/* <NOC_COPYRIGHT> */
/*
/*****/
/*
/* Function:
/*
/* AMQSAMP4 is a sample CL program to create or reset the
/* MQI resources to use with the sample programs.
/*
/* This program, or a similar one, can be run when the MQM
/* is started - it creates the objects if missing, or resets
/* their attributes to the prescribed values.
/*
/*
/*
/* Exceptions signaled: none
/* Exceptions monitored: none
/*
/* AMQSAMP4 takes a single parameter, the Queue Manager name
/*
/*****/
QSYS/PGM PARM(&QMGRNAME)

/*****/
/* Queue Manager Name Parameter
/*****/
QSYS/DCL VAR(&QMGRNAME) TYPE(*CHAR)

/*****/
/* EXAMPLES OF DIFFERENT QUEUE TYPES
/*
/* Create local, alias and remote queues
/*
/* Uses system defaults for most attributes
/*
/*****/
/* Create a local queue */
CRTMQMQ QNAME('SYSTEM.SAMPLE.LOCAL') +
MQMNAME(&QMGRNAME) +
QTYPE(*LCL) REPLACE(*YES) +
+
TEXT('Sample local queue') /* description */+
SHARE(*YES) /* Shareable */+
DFTMSGPST(*YES) /* Persistent messages OK */

/* Create an alias queue */
CRTMQMQ QNAME('SYSTEM.SAMPLE.ALIAS') +
MQMNAME(&QMGRNAME) +
QTYPE(*ALS) REPLACE(*YES) +
+
TEXT('Sample alias queue') +
DFTMSGPST(*YES) /* Persistent messages OK */+
TGTONAME('SYSTEM.SAMPLE.LOCAL')

/* Create a remote queue - in this case, an indirect reference */

```

```

/* is made to the sample local queue on OTHER queue manager */
CRTMQMQ QNAME('SYSTEM.SAMPLE.REMOTE') +
MQMNAME(&QMGRNAME) +
QTYPE(*RMT) REPLACE(*YES) +
+
TEXT('Sample remote queue')/* description */+
DFTMSGPST(*YES) /* Persistent messages OK */+
RMTQNAME('SYSTEM.SAMPLE.LOCAL') +
RMTMQMNAME(OTHER) /* Queue is on OTHER */

/* Create a transmission queue for messages to queues at OTHER */
/* By default, use remote node name */
CRTMQMQ QNAME('OTHER') /* transmission queue name */+
MQMNAME(&QMGRNAME) +
QTYPE(*LCL) REPLACE(*YES) +
TEXT('Transmission queue to OTHER') +
USAGE(*TMQ) /* transmission queue */

/*****
/* SPECIFIC QUEUES AND PROCESS USED BY SAMPLE PROGRAMS */
/* */
/* Create local queues used by sample programs */
/* Create MQI process associated with sample initiation queue */
/* */
*****/
/* General reply queue */
CRTMQMQ QNAME('SYSTEM.SAMPLE.REPLY') +
MQMNAME(&QMGRNAME) +
QTYPE(*LCL) REPLACE(*YES) +
+
TEXT('General reply queue') +
DFTMSGPST(*NO) /* Not Persistent */

/* Queue used by AMQSINQ4 */
CRTMQMQ QNAME('SYSTEM.SAMPLE.INQ') +
MQMNAME(&QMGRNAME) +
QTYPE(*LCL) REPLACE(*YES) +
+
TEXT('Queue for AMQSINQ4') +
SHARE(*YES) /* Shareable */+
DFTMSGPST(*NO) /* Not Persistent */+
+
TRGENBL(*YES) /* Trigger control on */+
TRGTYPE(*FIRST)/* Trigger on first message*/+
PRCNAME('SYSTEM.SAMPLE.INQPROCESS') +
INITQNAME('SYSTEM.SAMPLE.TRIGGER')

/* Queue used by AMQSSET4 */
CRTMQMQ QNAME('SYSTEM.SAMPLE.SET') +
MQMNAME(&QMGRNAME) +
QTYPE(*LCL) REPLACE(*YES) +
+
TEXT('Queue for AMQSSET4') +
SHARE(*YES) /* Shareable */+
DFTMSGPST(*NO)/* Not Persistent */+
+
TRGENBL(*YES) /* Trigger control on */+
TRGTYPE(*FIRST)/* Trigger on first message*/+
PRCNAME('SYSTEM.SAMPLE.SETPROCESS') +
INITQNAME('SYSTEM.SAMPLE.TRIGGER')

/* Queue used by AMQSECH4 */
CRTMQMQ QNAME('SYSTEM.SAMPLE.ECHO') +
MQMNAME(&QMGRNAME) +
QTYPE(*LCL) REPLACE(*YES) +
+
TEXT('Queue for AMQSECH4') +
SHARE(*YES) /* Shareable */+
DFTMSGPST(*NO)/* Not Persistent */+
+
TRGENBL(*YES) /* Trigger control on */+
TRGTYPE(*FIRST)/* Trigger on first message*/+
PRCNAME('SYSTEM.SAMPLE.ECHOPROCESS') +
INITQNAME('SYSTEM.SAMPLE.TRIGGER')

/* Initiation Queue used by AMQSTRG4, sample trigger process */
CRTMQMQ QNAME('SYSTEM.SAMPLE.TRIGGER') +
MQMNAME(&QMGRNAME) +
QTYPE(*LCL) REPLACE(*YES) +
TEXT('Trigger queue for sample programs')

/* MQI Processes associated with triggered sample programs */

```

```

/*
/****** Note - there are versions of the triggered samples *****/
/****** in different languages - set APPID for these *****/
/****** process to the variation you want to trigger *****/
/*
CRTMQMPRC PRCNAME('SYSTEM.SAMPLE.INQPROCESS') +
MQMNAME(&QMGRNAME) +
REPLACE(*YES) +
+
TEXT('Trigger process for AMQSINQ4') +
ENVDATA('JOBPTY(3)') /* Submit parameter */ +
/** Select the triggered program here **/ +
APPID('QMOM/AMQSINQ4') /* C */ +
/* APPID('QMOM/AMQOINQ4') /* COBOL */ +
/* APPID('QMOM/AMQ3INQ4') /* RPG - ILE */

CRTMQMPRC PRCNAME('SYSTEM.SAMPLE.SETPROCESS') +
MQMNAME(&QMGRNAME) +
REPLACE(*YES) +
+
TEXT('Trigger process for AMQSSET4') +
ENVDATA('JOBPTY(3)') /* Submit parameter */ +
/** Select the triggered program here **/ +
APPID('QMOM/AMQSSET4') /* C */ +
/* APPID('QMOM/AMQOSET4') /* COBOL */ +
/* APPID('QMOM/AMQ3SET4') /* RPG - ILE */

CRTMQMPRC PRCNAME('SYSTEM.SAMPLE.ECHOPROCESS') +
MQMNAME(&QMGRNAME) +
REPLACE(*YES) +
+
TEXT('Trigger process for AMQSECH4') +
ENVDATA('JOBPTY(3)') /* Submit parameter */ +
/** Select the triggered program here **/ +
APPID('QMOM/AMQSECH4') /* C */ +
/* APPID('QMOM/AMQOECH4') /* COBOL */ +
/* APPID('QMOM/AMQ3ECH4') /* RPG - ILE */

/*****
/*
/* Normal return.
/*
/*****
SNDPGMMMSG MSG('AMQSAMP4 Completed creating sample +
objects for ' *CAT &QMGRNAME)
RETURN
ENDPGM

/*****
/*
/* END OF AMQSAMP4
/*
/*****

```

IBM i Alternatywne sposoby administrowania produktem IBM MQ for IBM i

Użycie komend CL jest preferowaną metodą administrowania produktem IBM MQ for IBM i. Można jednak używać różnych innych metod administracyjnych, w tym komend MQSC, komend PCF i zdalnego administrowania.

Do administrowania produktem IBM MQ for IBM i zwykle są używane komendy języka CL programu IBM i. Przegląd tych komend znajduje się w sekcji [“Zarządzanie produktem IBM MQ for IBM i za pomocą komend CL”](#) na stronie 253.

Zdarzeń instrumentacji produktu IBM MQ można użyć do monitorowania działania menedżerów kolejek. Informacje na temat zdarzeń instrumentacji produktu IBM MQ oraz sposobu ich użycia zawiera sekcja [Zdarzenia instrumentacji](#).

Można użyć dowolnej z metod administracyjnych opisanych w następujących podtematach, jako alternatywy dla komend CL programu IBM i :

Obiekty produktu IBM MQ for IBM i są administrować lokalnie lub zdalnie.

Administrowanie lokalne oznacza wykonywanie zadań administracyjnych na wszystkich menedżerach kolejek, które zostały zdefiniowane w systemie lokalnym. W produkcie IBM MQ można uznać to za administrowanie lokalne, ponieważ nie są zaangażowane żadne kanały produktu IBM MQ, to znaczy komunikacja jest zarządzana przez system operacyjny. Aby wykonać ten typ zadania, należy zalogować się do systemu zdalnego i wprowadzić z niego komendy albo utworzyć proces, który będzie mógł wydać komendy.

Produkt IBM MQ obsługuje administrowanie z jednego punktu za pomocą tego, co jest określane jako *zdalne administrowanie*. Zdalne administrowanie polega na wysyłaniu komunikatów sterujących programowalnych formatów komend (PCF) do partycji SYSTEM.ADMIN.COMMAND.QUEUE w docelowym menedżerze kolejek.

Istnieje wiele sposobów generowania komunikatów PCF. Są to:

1. Pisanie programu za pomocą komunikatów PCF. Więcej informacji zawiera sekcja [“Administrowanie przy użyciu komend PCF w systemie IBM i”](#) na stronie 269.
2. Pisanie programu za pomocą interfejsu MQAI, który wysyła komunikaty PCF. Więcej informacji zawiera sekcja [“Korzystanie z interfejsu MQAI w celu uproszczenia korzystania z systemów PCF”](#) na stronie 34.
3. Za pomocą programu IBM MQ Explorer można korzystać z programu IBM MQ for Windows, który umożliwia korzystanie z graficznego interfejsu użytkownika (GUI) i generuje poprawne komunikaty PCF. Więcej informacji zawiera sekcja [“Korzystanie z IBM MQ Explorer z IBM MQ for IBM i”](#) na stronie 270.
4. Komenda **STRMQMQSC** służy do wysyłania komend pośrednio do zdalnego menedżera kolejek. Więcej informacji zawiera sekcja [“Administrowanie za pomocą komend MQSC w systemie IBM i”](#) na stronie 268.

Na przykład można wydać komendę zdalną w celu zmiany definicji kolejki w zdalnym menedżerze kolejek.

Niektóre komendy nie mogą być wydawane w ten sposób, w szczególności w przypadku tworzenia lub uruchamiania menedżerów kolejek i uruchamiania serwerów komend. Aby wykonać ten typ zadania, należy zalogować się do systemu zdalnego i wydać z niego komendy lub utworzyć proces, który będzie mógł wydawać komendy dla użytkownika.

Te informacje umożliwiają zapoznanie się z informacjami na temat komend MQSC oraz sposób ich użycia w celu administrowania produktem IBM MQ for IBM i.

Komendy skryptowe IBM MQ (MQSC) są zapisywane w postaci czytelnej dla użytkownika, czyli w tekście EBCDIC. Za pomocą komend MQSC używane są zarządzanie obiektami menedżera kolejek, w tym samym menedżerem kolejek, kolejkami, definicjami procesów, listami nazw, kanałami, kanałami połączeń klientów, obiektami nasłuchiwanymi, usługami, tematami i obiektami informacji uwierzytelniających.

Komendy MQSC są emitowane do menedżera kolejek za pomocą komendy CL **STRMQMQSC** IBM MQ. Ta metoda jest tylko metodą wsadową, biorąc pod uwagę jej dane wejściowe ze źródłowego zbioru fizycznego w systemie bibliotecznym serwera. Domyślną nazwą tego źródłowego zbioru fizycznego jest QMQSC.



Ostrzeżenie: Nie należy używać biblioteki QTEMP jako biblioteki źródłowej do komendy STRMQMQSC, ponieważ użycie biblioteki QTEMP jest ograniczone. Do komendy należy użyć innej biblioteki jako pliku wejściowego.

Produkt IBM MQ for IBM i nie dostarcza pliku źródłowego o nazwie QMQSC. Aby przetworzyć komendy MQSC, należy utworzyć plik źródłowy QMQSC w wybranym przez użytkownika bibliotece, wydając następującą komendę:

```
CRTRCPFB FILE(MYLIB/QMQSC) RCDLEN(240) TEXT('IBM MQ - MQSC Source')
```

Źródło MQSC jest przechowywane w elementach znajdujących się w tym pliku źródłowym. Aby pracować z członkami, należy wprowadzić następującą komendę:

```
WRKMBRPDM MYLIB/QMQSC
```

Teraz można dodawać nowych członków i utrzymywać istniejące.

Komendy MQSC można również wprowadzić interaktywnie, wydając komendę RUNMQSC lub:

1. Wpisz nazwę menedżera kolejek i naciśnij klawisz Enter , aby uzyskać dostęp do panelu wyników programu **WRKMQM** .
2. Wybranie opcji F23=More options na tym panelu.
3. Wybór opcji 26 dla aktywnego menedżera kolejek na panelu pokazanego w programie [Rysunek 32 na stronie 269](#).

Aby zakończyć taką sesję MQSC, wpisz end.

[Rysunek 32 na stronie 269](#) to wyciąg z pliku komend MQSC, który przedstawia komendę MQSC (DEFINE QLOCAL) z jej atrybutami.

```
.  
.   
DEFINE QLOCAL(ORANGE.LOCAL.QUEUE) REPLACE +  
DESCR(' ') +  
PUT(ENABLED) +  
DEFPRTY(0) +  
DEFPSIST(NO) +  
GET(ENABLED) +  
MAXDEPTH(5000) +  
MAXMSGL(1024) +  
DEFSOPT(SHARED) +  
NOHARDENBO +  
USAGE(NORMAL) +  
NOTRIGGER;  
.   
.
```

Rysunek 32. Wyodrębnij z pliku komend MQSC myprog.in

W celu zapewnienia przenośności w środowiskach IBM MQ należy ograniczyć długość wiersza w plikach komend MQSC do 72 znaków. Znak plus wskazuje, że komenda jest kontynuowana w następnym wierszu.

Atrybuty obiektów określone w MQSC są wyświetlane w tej sekcji wielkimi literami (na przykład RQMNAME), chociaż nie są rozróżniane wielkości liter.

Uwaga:

1. Format pliku MQSC nie zależy od jego położenia w systemie plików.
2. Nazwy atrybutów MQSC są ograniczone do ośmiu znaków.
3. Komendy MQSC są dostępne na innych platformach, w tym w produkcji z/OS.

Opis każdej komendy MQSC i jej składni można znaleźć w sekcji [“Komendy skryptowe \(MQSC\)”](#) na stronie [11](#).

IBM i

Administrowanie przy użyciu komend PCF w systemie IBM i

Celem komend programu IBM MQ programmable command format (PCF) jest umożliwienie programowania zadań administracyjnych w programie administracyjnym. W ten sposób można tworzyć kolejki i definicje procesów, a także zmieniać menedżery kolejek, z programu.

Komendy PCF obejmują ten sam zakres funkcji udostępnianych przez komendy MQSC. Jednak w przeciwieństwie do komend MQSC, komendy PCF i ich odpowiedzi nie są w formacie tekstowym, który można odczytać.

Można napisać program, aby wydać komendy PCF do dowolnego menedżera kolejek w sieci z jednego węzła. W ten sposób można zarówno scentralizować, jak i automatyzować zadania administracyjne.

Każda komenda PCF jest strukturą danych osadzoną w części danych aplikacji komunikatu produktu IBM MQ . Każda komenda jest wysyłana do docelowego menedżera kolejek przy użyciu funkcji MQI MQPUT w taki sam sposób, jak w przypadku innych komunikatów. Serwer komend w menedżerze kolejek odbierającym komunikat interpretuje go jako komunikat komendy i uruchamia komendę. Aby uzyskać odpowiedzi, aplikacja wysyła wywołanie MQGET, a dane odpowiedzi są zwracane w innej strukturze danych. Aplikacja może następnie przetworzyć odpowiedź i podjąć odpowiednie działania.

W skrócie, są to niektóre z rzeczy, które programista aplikacji musi określić, aby utworzyć komunikat komendy PCF:

deskryptor komunikatu

Jest to standardowy deskryptor komunikatu produktu IBM MQ , w którym:

- Typ komunikatu (*MsgType*) to MQMT_REQUEST.
- Format komunikatu (*Format*) to MQFMT_ADMIN.

Dane aplikacji

Zawiera komunikat PCF zawierający nagłówek PCF, w którym:

- Typ komunikatu PCF (*Type*) określa komendę MQCFT_COMMAND.
- Identyfikator komendy określa komendę, na przykład *Change Queue* (MQCMD_CHANGE_Q).

Zmiany PCF to komendy PCF, które zawierają komendy MQSC w tekście komunikatu. Komendy PCF można używać do wysyłania komend do zdalnego menedżera kolejek. Więcej informacji na ten temat zawiera sekcja [“Korzystanie z interfejsu MQAI w celu uproszczenia korzystania z systemów PCF”](#) na stronie 34.

Pełny opis struktur danych PCF i sposób ich implementowania zawiera sekcja [Struktury komend i odpowiedzi](#).

Korzystanie z IBM MQ Explorer z IBM MQ for IBM i

Te informacje umożliwiają administrowanie produktem IBM MQ for IBM i przy użyciu programu IBM MQ Explorer.

IBM MQ for Windows (platformyx86) i IBM MQ for Linux (platformyx86 i x86-64) udostępniają interfejs administracyjny, nazywany Eksploratorem IBM MQ , w celu wykonywania zadań administracyjnych jako alternatywa dla komend CL, sterowania lub MQSC.

IBM MQ Explorer umożliwia lokalne lub zdalne administrowanie siecią z poziomu komputera z uruchomionym systemem Windows (platformax86) lub Linux (platformyx86 i x86-64), wskazując IBM MQ Explorer w menedżerach kolejek i klastrach, które są zainteresowane.

Za pomocą IBM MQ Explorer można:

- Uruchamianie i zatrzymywanie menedżera kolejek (tylko na komputerze lokalnym).
- Zdefiniuj, wyświetl i zmień definicje obiektów produktu IBM MQ , takich jak kolejki, tematy i kanały.
- Przeglądaj komunikaty w kolejce.
- Uruchamianie i zatrzymywanie kanału.
- Wyświetlanie informacji o statusie kanału.
- Wyświetlanie menedżerów kolejek w klastrze.
- Sprawdź, które aplikacje, użytkownicy lub kanały mają otwartą określoną kolejkę.
- Utwórz nowy klaster menedżera kolejek przy użyciu kreatora **Utwórz nowy klaster** .
- Dodaj menedżera kolejek do klastra za pomocą kreatora **Dodawanie menedżera kolejek do klastra** .
- Zarządzanie obiektem informacji uwierzytelniającej, używanym z zabezpieczeniami kanału Transport Layer Security (TLS).

Korzystając z poradnictwa internetowego, można:

- Definiowanie i sterowanie różnymi zasobami, w tym menedżerami kolejek, kolejkami, kanałami, definicjami procesów, kanałami połączeń klientów, obiektami nasłuchiwaniami, tematami, usługami, listami nazw i klastrami.
- Uruchamianie lub zatrzymywanie menedżera kolejek i powiązanych z nim procesów.
- Wyświetlanie menedżerów kolejek i powiązanych z nimi obiektów na stacji roboczej lub na innych stacjach roboczych.
- Sprawdź status menedżerów kolejek, klastrów i kanałów.

Przed podjęciem próby użycia produktu IBM MQ Explorer do zarządzania serwerem IBM MQ na komputerze serwera należy upewnić się, że spełnione są następujące wymagania. Sprawdź, czy:

1. Serwer komend jest uruchomiony dla dowolnego menedżera kolejek, który jest administrowany, uruchamiany na serwerze przez komendę CL **STRMQCSVR**.
2. Odpowiedni program nasłuchujący TCP/IP istnieje dla każdego menedżera kolejek zdalnych. Jest to program nasłuchujący IBM MQ uruchomiony za pomocą komendy **STRMQLSR**.
3. Kanał połączenia z serwerem o nazwie SYSTEM.ADMIN.SVRCONN istnieje w każdym zdalnym menedżerze kolejek. Ten kanał należy utworzyć samodzielnie. Jest on obowiązkowy dla każdego administrowanego zdalnego menedżera kolejek. Bez niego zdalne administrowanie nie jest możliwe.
4. Sprawdź, czy kolejka SYSTEM.MQEXPLORER.REPLY.MODEL istnieje.

IBM i Zarządzanie serwerem komend do zdalnego administrowania w systemie IBM i

Te informacje umożliwiają poznanie zdalnego administrowania serwerem komend produktu IBM MQ for IBM i.

Z każdym menedżerem kolejek może być powiązany serwer komend. Serwer komend przetwarza wszystkie komendy przychodzące ze zdalnych menedżerów kolejek lub komendy PCF z aplikacji. Przedstawia on komendy do menedżera kolejek w celu przetworzenia i zwraca kod zakończenia lub komunikat operatora w zależności od pochodzenia komendy.

Serwer komend jest obowiązkowy dla wszystkich administrowania z użyciem systemów PCF, MQAI, a także dla zdalnego administrowania.

Uwaga: W przypadku zdalnego administrowania należy upewnić się, że uruchomiony jest docelowy menedżer kolejek. W przeciwnym razie komunikaty zawierające komendy nie mogą opuścić menedżera kolejek, z którego są one wydawane. Zamiast tego komunikaty te są kolejkowane w lokalnej kolejce transmisji, która służy do obsługi menedżera kolejek zdalnych. Unikaj tej sytuacji, jeśli w ogóle możliwe.

Istnieją oddzielne komendy sterujące służące do uruchamiania i zatrzymywania serwera komend. Za pomocą programu IBM MQ Explorer można wykonywać operacje opisane w poniższych sekcjach.

Uruchamianie i zatrzymywanie serwera komend

Aby uruchomić serwer komend, należy użyć następującej komendy CL:

```
STRMQCSVR MQMNAME('saturn.queue.manager')
```

gdzie `saturn.queue.manager` jest menedżerem kolejek, dla którego uruchamiany jest serwer komend.

Aby zatrzymać serwer komend, należy użyć jednej z następujących komend CL:

1.

```
ENDMQCSVR MQMNAME('saturn.queue.manager') OPTION(*CNTRLD)
```

w celu wykonania kontrolowanego zatrzymania, gdzie `saturn.queue.manager` jest menedżerem kolejek, dla którego zatrzymano serwer komend. Jest to opcja domyślna, co oznacza, że można pominąć `OPTION(*CNTRLD)`.

```
2. ENDMQMCSVR MQMNAME('saturn.queue.manager') OPTION(*IMMED)
```

aby wykonać natychmiastowe zatrzymanie, gdzie `saturn.queue.manager` jest menedżerem kolejek, dla którego zatrzymano serwer komend.

Wyświetlanie statusu serwera komend

W przypadku zdalnego administrowania należy upewnić się, że serwer komend w docelowym menedżerze kolejek jest uruchomiony. Jeśli nie jest uruchomiony, komendy zdalne nie mogą być przetwarzane. Wszystkie komunikaty zawierające komendy są kolejgowane w kolejce komend menedżera kolejek docelowych `SYSTEM.ADMIN.COMMAND.QUEUE`.

Aby wyświetlić status serwera komend dla menedżera kolejek, wywołanego tutaj `saturn.queue.manager`, komenda CL jest następująca:

```
DSPMQMCSVR MQMNAME('saturn.queue.manager')
```

Wydadź tę komendę na komputerze docelowym. Jeśli serwer komend jest uruchomiony, zostanie wyświetlony panel wyświetlany w programie [Rysunek 33 na stronie 272](#) :

```
Display MQM Command Server (DSPMQMCSVR)
```

```
Queue manager name . . . . . > saturn.queue.manager
```

```
MQM Command Server Status. . . . > RUNNING
```

```
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display  
F24=More keys
```

Rysunek 33. Wyświetl panel serwera komend MQM

IBM i Zarządzanie pracą dla produktu IBM i

W tej sekcji opisano sposób, w jaki produkt IBM MQ obsługuje żądania pracy, a także szczegółowe informacje na temat opcji dostępnych do priorytetyzowania i kontrolowania zadań powiązanych z produktem IBM MQ.

Ostrzeżenie

Nie należy zmieniać obiektów zarządzania pracą produktu IBM MQ, chyba że w pełni rozumie się pojęcia związane z zarządzaniem pracą IBM i i IBM MQ.

Dodatkowe informacje dotyczące podsystemów i opisów zadań można znaleźć w sekcji [Zarządzanie pracą](#) w dokumentacji produktu IBM i. Należy zwrócić szczególną uwagę na sekcje w sekcjach [Uruchamianie zadań](#) i [Zadania wsadowe](#).

Produkt IBM MQ for IBM i zawiera środowisko IBM i UNIX oraz wątki produktu IBM i. **Nie** dokonuj żadnych zmian w obiektach w Integrated File System (IFS).

Podczas normalnych operacji menedżer kolejek produktu IBM MQ uruchamia wiele zadań wsadowych w celu wykonania różnych zadań. Domyślnie te zadania wsadowe są uruchamiane w podsystemie QMQM, który jest tworzony podczas instalowania programu IBM MQ.

Zarządzanie pracą odnosi się do procesu dostosowywania zadań IBM MQ w celu uzyskania optymalnej wydajności z systemu lub prostszego administrowania.

Można na przykład:

- Zmień priorytet uruchamiania zadań, aby jeden menedżer kolejek był bardziej aktywny niż inny.
- Przekieruj dane wyjściowe pewnej liczby zadań do określonej kolejki wyjściowej.
- Wszystkie zadania określonego typu są uruchamiane w określonym podsystemie.
- Wyodrębnia błędy do podsystemu.

Zarządzanie pracą jest wykonywane przez utworzenie lub zmianę opisów zadań powiązanych z zadaniami IBM MQ . Zarządzanie pracą można skonfigurować dla:

- Cała instalacja produktu IBM MQ .
- Poszczególne menedżery kolejek.
- Pojedyncze zadania dla poszczególnych menedżerów kolejek.

IBM i Zadania IBM MQ dla produktu IBM i

Jest to tabela zadań produktu IBM MQ for IBM i oraz krótki opis każdego z nich.

Po uruchomieniu menedżera kolejek wyświetlane są niektóre lub wszystkie z następujących zadań wsadowych uruchomionych w ramach profilu użytkownika QMQM w podsystemie IBM MQ . Zadania są opisane krótko w programie [Tabela 19 na stronie 273](#).

Za pomocą opcji 22 na panelu **Praca z menedżerem kolejek** (WRKMQM) można wyświetlić wszystkie zadania połączone z menedżerem kolejek. Obiekty nastuchiwania można wyświetlać przy użyciu komendy WRKMQMLSR.

<i>Tabela 19. IBM MQ Zadania.</i>	
Nazwa zadania	Funkcja
AMQALMPX	Procesor punktów kontrolnych, który okresowo pobiera punkty kontrolne kroniki.
AMQZMUC0	Menedżer programu narzędziowego. To zadanie wykonuje newralgiczne programy narzędziowe menedżera kolejek, na przykład menedżer łańcucha kroniki.
AMQZXMA0	Kontroler wykonywania, który jest pierwszym zadaniem uruchomionym przez menedżer kolejek. Obsługuje żądania MQCONN i uruchamia procesy agenta w celu przetwarzania wywołań interfejsu API produktu IBM MQ .
AMQZFUMA	Menedżer uprawnień do obiektów (OAM).
AMQZLAA0	Agenty menedżera kolejek, które wykonują większość prac dla aplikacji, które łączą się z menedżerem kolejek przy użyciu funkcji MQCNO_STANDARD_BINDING.
AMQZLSA0	Agent menedżera kolejek.
AMQZMUFO	Menedżer narzędzi
AMQZMGRO	Kontroler procesu. To zadanie jest używane do uruchamiania programów nastuchujących i usług oraz zarządzania nimi.
AMQZMURO	Menedżer programu narzędziowego. To zadanie wykonuje newralgiczne programy narzędziowe menedżera kolejek, na przykład menedżer łańcucha kroniki.
AMQFQPUB	Umieszczony w kolejce demon publikowania/subskrypcji.
AMQFCXBA	Zadanie robocze brokera.
RUNMQBRK	Zadanie sterujące brokerem.
AMQRMPPA	Zadanie zestawiania procesów kanału.

Tabela 19. IBM MQ Zadania. (kontynuacja)	
Nazwa zadania	Funkcja
AMQCRSTA	Responder kanału wywołany przez protokół TCP/IP.
AMQCRS6B	Kanał odbiorczy LU62 i połączenie klienta (patrz uwaga).
AMQRRMFA	Menedżer repozytorium dla klastrów.
AMQCLMAA	Niewątkowy program nasłuchujący TCP/IP.
AMQPCSEA	Procesor komend PCF, który obsługuje żądania PCF i zdalne żądania administracyjne.
RUNMQTRM	Monitor wyzwalacza.
RUNMQDLQ	Procedura obsługi kolejki niewystanych wiadomości.
RUNMQCHI	Inicjator kanału.
RUNMQCHL	Zadanie kanału nadawczego, które jest uruchamiane dla każdego kanału nadawczego.
RUNMQLSR	Wątki nasłuchiwanie TCP/IP.
AMQRCMLA	Kanał komend MQSC dla kanału i komend PCF.

Uwaga: Zadanie odbiornika LU62 jest uruchamiane w podsystemie komunikacyjnym i przyjmuje jego właściwości środowiska wykonawczego z pozycji routingu i komunikacji, które są używane do uruchamiania zadania. Więcej informacji na ten temat zawiera sekcja [Initiated end \(Receiver\)](#).

IBM i Obiekty zarządzania pracą w systemie IBM i

Po zainstalowaniu produktu IBM MQ w bibliotece QMQM dostarczane są różne obiekty, które mają pomóc w zarządzaniu pracą. Te obiekty to te, które są niezbędne do uruchomienia zadań produktu IBM MQ we własnym podsystemie.

Przykładowe opisy zadań są udostępnione dla dwóch zadań wsadowych IBM MQ. Jeśli dla zadania IBM MQ nie zostanie podany żaden konkretny opis zadania, zostanie on uruchomiony z domyślnym opisem zadania QMQMJOB.

Obiekty zarządzania pracą, które są dostarczane podczas instalowania produktu IBM MQ, są wyświetlane w programie [Tabela 20 na stronie 274](#), a obiekty utworzone dla menedżera kolejek są wymienione w sekcji [Tabela 21 na stronie 275](#).

Uwaga: Obiekty zarządzania pracą można znaleźć w bibliotece QMQM, a obiekty menedżera kolejek można znaleźć w bibliotece menedżera kolejek.

Tabela 20. Obiekty zarządzania pracą		
Nazwa	Typ	Opis
AMQALMPX	*JOB	Opis zadania używany przez proces punktu kontrolnego
AMQZLAA0	*JOB	Opis zadania, który jest używany przez procesy agenta IBM MQ
AMQZLSA0	*JOB	Agent menedżera kolejek izolowanych powiązań
AMQZXMA0	*JOB	Opis zadania używany przez kontrolery wykonawcze serwera IBM MQ
QMQM	*SBSD	Podsystem, w którym uruchamiane są wszystkie zadania IBM MQ
QMQM	*JOBQ	Kolejka zadań przyłączona do podanego podsystemu
QMQMJOB	*JOB	Domyślny opis zadania IBM MQ, używany w przypadku, gdy dla zadania nie ma określonego opisu zadania

<i>Tabela 20. Obiekty zarządzania pracą (kontynuacja)</i>		
Nazwa	Typ	Opis
QMOMMSG	*MSGQ	Domyślna kolejka komunikatów dla zadań programu IBM MQ .
QMOMRUN20	*CLS	Opis klasy dla zadań o wysokim priorytecie IBM MQ
QMOMRUN35	*CLS	Opis klasy dla zadań o średnim priorytecie IBM MQ
QMOMRUN50	*CLS	Opis klasy dla zadań o niskim priorytecie IBM MQ

<i>Tabela 21. Obiekty zarządzania pracą utworzone dla menedżera kolejek</i>		
Nazwa	Typ	Opis
AMQA000000	*JRNRCV	Dziennik lokalny dziennika
Komenda AMQAJRN	*JRN	Kronika lokalna
AMQJRNINF	*USRSPC	Obszar użytkownika, który jest aktualizowany przy użyciu najnowszych dzienników wymaganych przy uruchamianiu i odtwarzaniu nośników menedżera kolejek. Ten obszar użytkownika może być odpytywany przez aplikację w celu określenia, które dzienniki wymagają archiwizacji i które mogą zostać bezpiecznie usunięte.
AMQAJRNMSG	*MSGQ	Kolejka komunikatów kroniki lokalnej
AMQCRC6B	*PGM	Program uruchamiający połączenie LU6.2
AMQRFOLD	*PLIK	Plik definicji migrowanego kanału menedżera kolejek
QMOMMSG	*MSGQ	Kolejka komunikatów menedżera kolejek

W jaki sposób produkt IBM MQ używa obiektów zarządzania pracą w systemie IBM i

W tej sekcji opisano sposób, w jaki produkt IBM MQ korzysta z obiektów zarządzania pracą, a także udostępnia przykłady konfiguracji.



Ostrzeżenie: Nie zmieniaj ustawień pozycji kolejki zadań w podsystemie QMOM, aby ograniczyć liczbę zadań dozwolonych w podsystemie według priorytetu. Próba wykonania tej czynności może spowodować zatrzymanie podstawowych zadań produktu IBM MQ po ich przestaniu i spowodowanie, że uruchomienie menedżera kolejek nie powiedzie się.

Aby dowiedzieć się, jak skonfigurować zarządzanie pracą, należy najpierw zrozumieć, w jaki sposób produkt IBM MQ korzysta z opisów zadań.

Opis zadania używany do uruchomienia zadania steruje wieloma atrybutami zadania. Na przykład:

- Kolejka zadań, w której zadanie jest umieszczone w kolejce i na którym działa podsystem, w którym uruchamiane jest zadanie.
- Dane routingu używane do uruchamiania zadania i klasy, które są używane przez zadanie na potrzeby jego parametrów wykonawczych.
- Kolejka wyjściowa używana przez zadanie do drukowania plików.

Proces uruchamiania zadania IBM MQ można wziąć pod uwagę w trzech krokach:

1. IBM MQ wybiera opis zadania.

Produkt IBM MQ używa następującej techniki w celu określenia opisu zadania, który ma zostać użyty dla zadania wsadowego:

- a. Poszukaj w bibliotece menedżera kolejek opisu zadania o takiej samej nazwie, jak nazwa zadania. Więcej szczegółowych informacji na temat biblioteki menedżera kolejek zawiera sekcja [Informacje o nazwach bibliotek menedżera kolejek produktu IBM MQ for IBM i](#).
 - b. Poszukaj domyślnego opisu zadania QMQMJOB in bibliotece menedżera kolejek.
 - c. Poszukaj w bibliotece QMQM opisu zadania o nazwie takiej samej, jak nazwa zadania.
 - d. Użyj domyślnego opisu zadania, QMQMJOB in bibliotece QMQM.
2. Zadanie jest wprowadzane do kolejki zadań.

Opisy zadań dostarczone wraz z produktem IBM MQ zostały domyślnie ustawione w celu umieszczenia zadań w kolejce zadań QMQM w bibliotece QMQM. Kolejka zadań QMQM jest przyłączona do podanego podsystemu QMQM, więc domyślnie zadania są uruchamiane w podsystemie QMQM.

3. Zadanie jest wprowadzane do podsystemu i przechodzi przez kroki routingu.

Gdy zadanie zostanie wprowadzone do podsystemu, dane routingu określone w opisie zadania są używane do znalezienia pozycji routingu dla zadania.

Dane routingu muszą być zgodne z jednym z wpisów routingu zdefiniowanych w podsystemie QMQM, a to definiuje, które z dostarczonych klas (QMQRUN20, QMQRUN35 lub QMQRUN50) są używane przez zadanie.

Uwaga: Jeśli zadania IBM MQ nie wydają się być uruchamiane, upewnij się, że podsystem jest uruchomiony, a kolejka zadań nie jest wstrzymana,

Jeśli zmodyfikowano obiekty zarządzania pracą produktu IBM MQ, upewnij się, że wszystko jest powiązane poprawnie. Na przykład, jeśli w opisie zadania określona zostanie kolejka zadań inna niż QMQM/QMQM, należy upewnić się, że dla podsystemu zostanie wykonana komenda ADDJOBQE, czyli QMQM.

Dla każdego zadania udokumentowanego w programie Tabela 19 na stronie 273 można utworzyć opis zadania, korzystając z następującego arkusza jako przykładu:

```

What is the queue manager library name? _____
Does job description AMQZXMA0 exist in the queue manager library? Yes No
Does job description QMQMJOB in exist in the queue manager library? Yes No
Does job description AMQZXMA0 exist in the QMQM library? Yes No
Does job description QMQMJOB in exist in the QMQM library? Yes No

```

W przypadku odpowiedzi Nie na wszystkie te pytania należy utworzyć globalny opis zadania QMQMJOB in bibliotece QMQM.

Kolejka komunikatów produktu IBM MQ

Kolejka komunikatów produktu IBM MQ, QMQMMSG, jest tworzona w każdej bibliotece menedżera kolejek. Komunikaty systemu operacyjnego są wysyłane do tej kolejki po zakończeniu zadań menedżera kolejek, a program IBM MQ wysyła komunikaty do kolejki. Na przykład, aby raportować, które dzienniki są potrzebne podczas uruchamiania. Aby ułatwić monitorowanie, należy zachować liczbę komunikatów w tej kolejce komunikatów, których wielkość jest możliwa do zarządzania.

IBM i Domyślne przykłady systemów dla IBM i

W tych przykładach przedstawiono sposób, w jaki niezmodyfikowana instalacja produktu IBM MQ działa w przypadku, gdy niektóre standardowe zadania są wprowadzane w czasie uruchamiania menedżera kolejek.

Najpierw uruchamiane jest zadanie kontrolera wykonywania AMQZXMA0.

1. Wprowadź komendę **STRMQM** dla menedżera kolejek TESTQM.
2. Produkt IBM MQ wyszukuje w bibliotece menedżera kolejek QMTESTQM, po pierwsze, opis zadania AMQZXMA0, a następnie opis zadania QMQMJOB in.

Żaden z tych opisów zadań nie istnieje, dlatego program IBM MQ wyszukuje opis zadania AMQZXMA0 w bibliotece produktu QMQM. Ten opis zadania istnieje, dlatego jest używany do wprowadzania zadania.

3. Opis zadania korzysta z domyślnej kolejki zadań serwera IBM MQ , dlatego zadanie jest wprowadzane do kolejki zadań QMQM/QMQM.
4. Dane routingu w opisie zadania AMQZXMA0 to QMQMRUN20, dlatego system przeszukuje pozycje routingu podsystemu dla tych danych, które są zgodne z danymi.

Domyślnie pozycja routingu o numerze kolejnym 9900 zawiera dane porównania, które są zgodne z wartością QMQMRUN20, dlatego zadanie jest uruchamiane z klasą zdefiniowaną w tej pozycji routingu, która jest również nazywana QMQMRUN20.
5. Klasa QMQM/QMQMRUN20 ma priorytet uruchomienia ustawiony na 20, dlatego zadanie AMQZXMA0 jest uruchamiane w podsystemie QMQM z tym samym priorytetem, co większość zadań interaktywnych w systemie.

Następnie zostanie uruchomione zadanie procesu punktu kontrolnego AMQALMPX.

1. Produkt IBM MQ wyszukuje w bibliotece menedżera kolejek QMTESTQM, po pierwsze, opis zadania AMQALPMX, a następnie opis zadania QMQMJOB.

Żaden z tych opisów zadań nie istnieje, dlatego program IBM MQ wyszukuje opisy zadań AMQALPMX i QMQMJOB w bibliotece produktu QMQM.

Opis zadania AMQALPMX nie istnieje, ale komenda QMQMJOB jest używana, dlatego zadanie QMQMJOB jest używane do wprowadzania zadania.

Uwaga: Opis zadania QMQMJOB jest zawsze używany dla zadań IBM MQ , które nie mają własnego opisu zadania.

2. Opis zadania korzysta z domyślnej kolejki zadań serwera IBM MQ , dlatego zadanie jest wprowadzane do kolejki zadań QMQM/QMQM.
3. Dane routingu dla opisu zadania QMQMJOB to QMQMRUN35, dlatego system przeszukuje pozycje routingu podsystemu dla tych, które są zgodne z danymi.

Domyślnie pozycja routingu o numerze kolejnym 9910 zawiera dane porównawcze zgodne z wartością QMQMRUN35, dlatego zadanie jest uruchamiane z klasą zdefiniowaną na tej pozycji routingu, która jest również nazywana QMQMRUN35.
4. Klasa QMQM/QMQMRUN35 ma priorytet uruchomienia ustawiony na 35, więc zadanie AMQALPMX działa w podsystemie QMQM z niższym priorytetem niż większość zadań interaktywnych w systemie, ale wyższym priorytetem niż większość zadań wsadowych.

IBM i

Konfigurowanie przykładów zarządzania pracą w systemie IBM i

Informacje zawarte w tej sekcji umożliwiają poznanie sposobów zmiany i tworzenia opisów zadań programu IBM MQ w celu zmiany atrybutów środowiska wykonawczego zadań produktu IBM MQ .

Klucz do elastyczności zarządzania pracą produktu IBM MQ polega na dwupoziomowym sposobie wyszukiwania opisów zadań przez produkt IBM MQ :

- W przypadku tworzenia lub zmiany opisów zadań w bibliotece menedżera kolejek zmiany te zastępują globalne opisy zadań w QMQM, ale zmiany te są lokalne i mają wpływ tylko na ten konkretny menedżer kolejek.
- Jeśli w bibliotece QMQM zostaną utworzone lub zmienione opisy zadań globalnych, te opisy zadań będą miały wpływ na wszystkie menedżery kolejek w systemie, o ile nie zostaną nadpisane lokalnie dla poszczególnych menedżerów kolejek.

1. W poniższym przykładzie zwiększa się priorytet zadań sterowania kanałem dla pojedynczego menedżera kolejek.

Aby menedżer repozytorium i zadania inicjatora kanału, AMQRRMFA i RUNMQCHI były uruchamiane tak szybko, jak to możliwe, dla menedżera kolejek TESTQM, wykonaj następujące kroki:

- a. Utwórz duplikaty lokalne opisu zadania QMQM/QMQMJOBDD z nazwami procesów produktu IBM MQ , które mają być sterowane w bibliotece menedżera kolejek. Na przykład:

```
CRTDUPOBJ OBJ(QMQMJOBDD) FROMLIB(QMQM) OBJTYPE(*JOBDD) TOLIB(QMTESTQM)
NEWOBJ (RUNMQCHI)
CRTDUPOBJ OBJ(QMQMJOBDD) FROMLIB(QMQM) OBJTYPE(*JOBDD) TOLIB(QMTESTQM)
NEWOBJ (AMQRRMFA)
```

- b. Zmień parametr danych routingu w opisie zadania, aby upewnić się, że zadania korzystają z klasy QMQMRUN20 .

```
CHGJOBDD JOBD(QMTESTQM/RUNMQCHI) RTGDTA('QMQMRUN20')
CHGJOBDD JOBD(QMTESTQM/AMQRRMFA) RTGDTA('QMQMRUN20')
```

Zadania AMQRRMFA i RUNMQCHI dla menedżera kolejek TESTQM są teraz następujące:

- Użyj nowych opisów zadań lokalnych w bibliotece menedżera kolejek
 - Uruchom z priorytetem 20, ponieważ klasa QMQMRUN20 jest używana, gdy zadania są wprowadzane do podsystemu.
2. W poniższym przykładzie definiowana jest nowa klasa priorytetu uruchomienia dla podsystemu QMQM.

- a. Utwórz duplikat klasy w bibliotece QMQM, aby zezwolić innym menedżerom kolejek na dostęp do klasy, wydając następującą komendę:

```
CRTDUPOBJ OBJ(QMQMRUN20) FROMLIB(QMQM) OBJTYPE(*CLS) TOLIB(QMQM)
NEWOBJ (QMQMRUN10)
```

- b. Zmień klasę w taki sposób, aby miała nowy priorytet uruchamiania, wydając następującą komendę:

```
CHGCLS CLS(QMQM/QMQMRUN10) RUNPTY(10)
```

- c. Dodaj nową definicję klasy do podsystemu, wydając następującą komendę:

```
ADDRTGE SBSDD(QMQM/QMQM) SEQNBR(8999) CMPVAL('QMQMRUN10') PGM(QSYS/QCMD)
CLS(QMQM/QMQMRUN10)
```

Uwaga: Dla numeru kolejnego routingu można określić dowolną wartość liczbową, ale wartości te muszą być uporządkowane sekwencyjnie. Ten numer porządkowy informuje podsystem o kolejności, w jakiej pozycje routingu mają być przeszukiwane w celu dopasowania danych routingu.

- d. Zmień lokalny lub globalny opis zadania tak, aby używało nowej klasy priorytetu, wydając następującą komendę:

```
CHGJOBDD JOBD(QMQMlibname/QMQMJOBDD) RTGDTA('QMQMRUN10')
```

Teraz wszystkie zadania menedżera kolejek powiązane z biblioteką QMlibrarnyname używają priorytetu uruchomienia 10.

3. W poniższym przykładzie uruchamiany jest menedżer kolejek we własnym podsystemie.

Aby wykonać wszystkie zadania dla menedżera kolejek TESTQM w podsystemie QBATCH, wykonaj następujące kroki:

- a. Utwórz lokalny duplikat opisu zadania QMQM/QMQMJOBDD w bibliotece menedżera kolejek za pomocą komendy

```
CRTDUPOBJ OBJ(QMQMJOBDD) FROMLIB(QMQM) OBJTYPE(*JOBDD) TOLIB(QMTESTQM)
```

- b. Zmień parametr kolejki zadań w opisie zadania, aby upewnić się, że zadania korzystają z kolejki zadań QBATCH.

```
CHGJOBDB JOBDB(QMTESTQM/QMQMJOBDB) JOBQ(*LIBL/QBATCH)
```

Uwaga: Kolejka zadań jest powiązana z opisem podsystemu. Jeśli zostanie określone, że zadania są w kolejce zadań, należy sprawdzić, czy definicja kolejki zadań jest zdefiniowana w SBSDB. Użyj komendy DSPSBSDB dla podsystemu i wybierz opcję 6, pozycje kolejki zadań.

Wszystkie zadania dla menedżera kolejek TESTQM teraz:

- Użyj nowego domyślnego opisu lokalnego zadania w bibliotece menedżera kolejek
- Są wprowadzane do kolejki zadań QBATCH.

Aby upewnić się, że zadania są kierowane i prawidłowo szeregowane:

- Utwórz pozycje routingu dla zadań IBM MQ w podsystemie QBATCH, lub
- Polegaj na pozycji routingu, która wywołuje QCMD, niezależnie od tego, jakie dane routingu są używane.

Ta opcja działa tylko wtedy, gdy opcja Maksymalna liczba aktywnych zadań dla kolejki zadań QBATCH jest ustawiona na *NOMAX. Wartością domyślną systemu jest 1.

4. W poniższym przykładzie tworzony jest inny podsystem IBM MQ .

a. Utwórz duplikat podsystemu w bibliotece QMQM, wydając następującą komendę:

```
CRTDUPOBJ OBJ(QMQM) FROMLIB(QMQM) OBJTYPE(*SBSD) TOLIB(QMQM) NEWOBJ(QMQM2)
```

b. Usuń kolejkę zadań QMQM, wydając następującą komendę:

```
RMVJOBQE SBSDB(QMQM/QMQM2) JOBQ(QMQM/QMQM)
```

c. Utwórz nową kolejkę zadań dla podsystemu, wydając następującą komendę:

```
CRTJOBQ JOBQ(QMQM/QMQM2) TEXT('Job queue for IBM MQ Queue Manager')
```

d. Dodaj pozycję kolejki zadań do podsystemu, wydając następującą komendę:

```
ADDJOBQE SBSDB(QMQM/QMQM2) JOBQ(QMQM/QMQM2) MAXACT(*NOMAX)
```

e. Utwórz duplikat QMQMJOBDB w bibliotece menedżera kolejek, wydając następującą komendę:

```
CRTDUPOBJ OBJ(QMQMJOBDB) FROMLIB(QMQM) OBJTYPE(*JOBDB) TOLIB(QMlibraryname)
```

f. Zmień opis zadania, aby użyć nowej kolejki zadań, wydając następującą komendę:

```
CHGJOBDB JOBDB(QMlibraryname/QMQMJOBDB) JOBQ(QMQM/QMQM2)
```

g. Uruchom podsystem, wydając następującą komendę:

```
STRSBS SBSDB(QMQM/QMQM2)
```

Uwaga:

- a. Podsystem można określić w dowolnej bibliotece. Jeśli z jakiegoś powodu produkt zostanie ponownie zainstalowany lub biblioteka QMQM zostanie zastąpiona, wszystkie wprowadzone zmiany zostaną usunięte.
 - b. Wszystkie zadania menedżera kolejek powiązane z nazwą biblioteki QMlibraryname są teraz uruchamiane w podsystemie QMQM2.
5. W poniższym przykładzie gromadzone są wszystkie dane wyjściowe dla danego typu zadania.

Aby zebrać wszystkie procesy punktu kontrolnego, AMQALMPX, protokoły zadań dla wielu menedżerów kolejek w jednej kolejce wyjściowej, wykonaj następujące kroki:

- a. Tworzenie kolejki wyjściowej, na przykład

```
CRTOUTQ OUTQ(MYLIB/CHKPTLOGS)
```

- b. Utwórz globalny duplikat opisu zadania QMQM/QMQMJOB, korzystając z nazwy procesu IBM MQ, który ma być sterowany, na przykład

```
CRTDUPOBJ OBJ(QMQMJOB) FROMLIB(QMQM) OBJTYPE(*JOB) NEWOBJ(AMQALMPX)
```

- c. Zmień parametr kolejki wyjściowej w opisie zadania tak, aby wskazywał na nową kolejkę wyjściową, a następnie zmień poziom rejestrowania zadania tak, aby wszystkie komunikaty były zapisywane w protokole zadania.

```
CHGJOB JOB(QMQM/AMQALMPX) OUTQ(MYLIB/CHKPTLOGS) LOG(4 00 *SECLVL)
```

Wszystkie zadania produktu IBM MQ AMQALMPX, dla wszystkich menedżerów kolejek, używają nowego opisu zadania globalnego AMQALMPX, pod warunkiem, że w lokalnej bibliotece menedżera kolejek nie ma lokalnych przestaniających opisów zadań.

Wszystkie zbiory buforowy protokołu zadania dla tych zadań są teraz zapisywane w kolejce wyjściowej CHKPTLOGS w bibliotece MYLIB.

Uwaga:

- a. Powyższy przykład działa tylko wtedy, gdy QPJOBLOG, lub dowolny zbiór drukarski, ma wartość *JOB dla parametru kolejki wyjściowej. W poprzednim przykładzie zbiór QSYS/QPDJOBLOG wymaga zbioru OUTQ ustawionego na *JOB.
- b. Aby zmienić zbiór wydruku systemowego, należy użyć komendy CHGPRTF. Na przykład:

```
CHGPRTF PRTF(QJOBLOG) OUTQ(*JOB)
```

Opcja *JOB wskazuje, że opisy zadań muszą być używane.

- c. Do określonej kolejki wyjściowej można wysyłać wszystkie pliki buforowy powiązane z zadaniami IBM MQ. Należy jednak sprawdzić, czy używany zbiór drukarski ma odpowiednią wartość dla parametru OUTQ.

IBM i Dostępność, tworzenie kopii zapasowych, odtwarzanie i restartowanie w systemie IBM i

Informacje zawarte w tej sekcji pozwalają zrozumieć, w jaki sposób produkt IBM MQ for IBM i korzysta z funkcji obsługi kronikowania IBM i w celu wspomaganie jej tworzenia i odtwarzania strategii.

Przed zapoznaniem się z tą sekcją należy zapoznać się ze standardowymi metodami tworzenia i odtwarzania kopii zapasowych produktu IBM i oraz z użyciem dzienników i powiązanych z nimi dzienników w systemie IBM i. Więcej informacji na temat tych tematów zawiera sekcja [Składowanie i odtwarzanie](#).

Aby zrozumieć strategię tworzenia i odtwarzania kopii zapasowych, należy najpierw zrozumieć, w jaki sposób produkt IBM MQ for IBM i organizuje swoje dane w systemie plików IBM i i zintegrowanym systemie plików (IFS).

Program IBM MQ for IBM i przechowuje swoje dane w pojedynczej bibliotece dla każdej instancji menedżera kolejek oraz w plikach strumieniowych w systemie plików IFS.

Biblioteki specyficzne dla menedżera kolejek zawierają kroniki, dzienniki i obiekty wymagane do sterowania zarządzaniem pracą menedżera kolejek. Katalogi i pliki systemu plików IFS zawierają pliki konfiguracyjne produktu IBM MQ, opisy obiektów produktu IBM MQ oraz dane, które zawierają.

Każda zmiana tych obiektów, która jest odtwarzalna w przypadku awarii systemu, jest rejestrowana w kronice *przed*, która jest stosowana do odpowiedniego obiektu. Powoduje to, że takie zmiany mogą zostać odzyskane przez odtworzenie informacji zapisanych w kronice.

Produkt IBM MQ for IBM i można skonfigurować do korzystania z wielu instancji menedżera kolejek na różnych serwerach w celu zapewnienia zwiększonej dostępności menedżera kolejek i przyspieszenia odtwarzania w przypadku awarii serwera lub menedżera kolejek.

IBM i Dzienniki menedżera kolejek w systemie IBM i

Te informacje umożliwiają zrozumienie, w jaki sposób produkt IBM MQ for IBM i używa kronik w swojej operacji do sterowania aktualizacjami obiektów lokalnych.

Każda biblioteka menedżera kolejek zawiera kronikę dla tego menedżera kolejek, a kronika ma nazwę QM *GRLIB*/AMQ *A* JRN, gdzie QM *GRLIB* jest nazwą biblioteki menedżera kolejek, a *A* jest literą, *A* w przypadku menedżera kolejek z pojedynczą instancją, który jest unikalny dla instancji menedżera kolejek.

QM *GRLIB* przyjmuje nazwę QM, po której następuje nazwa menedżera kolejek w unikalnym formacie. Na przykład menedżer kolejek o nazwie TEST ma bibliotekę menedżera kolejek o nazwie QMTEST. Biblioteka menedżera kolejek może zostać określona podczas tworzenia menedżera kolejek za pomocą komendy **CRTMQM**.

Kroniki skojarzone są z dziennikami, które zawierają kronikowane informacje. Odbiorniki są obiektami, do których informacje mogą być tylko dopisane i w końcu wypełnią je.

Dzienniki używają cennej ilości miejsca na dysku z informacjami o nieaktualnych danych. Można jednak umieścić te informacje w pamięci trwałej w celu zminimalizowania tego problemu. Jeden dziennik jest przyłączony do kroniki w dowolnym określonym czasie. Jeśli dziennik osiągnie wstępnie określony rozmiar progu, zostanie on odłączony i zastąpiony nowym dziennikiem. Istnieje możliwość określenia progu dzienników podczas tworzenia menedżera kolejek przy użyciu parametru **CRTMQM** i parametru **THRESHOLD**.

Dzienniki powiązane z lokalną kroniką IBM MQ for IBM i istnieją w każdej bibliotece menedżera kolejek i adoptować konwencję nazewnictwa w następujący sposób:

```
AMQ Arnnnnn
```

where

A

jest literą A-Z. Jest to A dla menedżerów kolejek pojedynczych instancji. Różni się on w zależności od instancji menedżera kolejek z wieloma instancjami.

nnnnn

jest dziesiątą 00000 to 99999, która jest zwiększana o 1 dla następnego dziennika w sekwencji.

o

jest dziesiątą 0 to 9, która jest zwiększana o 1 za każdym razem, gdy odtwarzany jest odbiornik.

Kolejność kronik jest oparta na dacie. Jednak nazewnictwo następnej kroniki jest oparte na następujących regułach:

1. AMQA1nnnnn przechodzi do AMQA1(nnnnn+1), a nnnnn zawija, gdy osiągnie 99999. Na przykład AMQA099999 przechodzi do AMQA000000, a AMQA999999 przechodzi do AMQA900000.
2. Jeśli kronika o nazwie wygenerowanej przez regułę 1 już istnieje, komunikat CPI70E3 jest wysyłany do kolejki komunikatów QSYSOPR, a automatyczne przełączanie odbiornika jest zatrzymywane.

Aktualnie podłączony odbiornik jest nadal używany do czasu zbadania problemu i ręcznego podłączenia nowego dziennika.

3. Jeśli w sekwencji nie jest dostępna żadna nowa nazwa (to znaczy, że w systemie znajdują się wszystkie możliwe nazwy dzienników), należy wykonać obie poniższe czynności:

- a. Dzienniki usuwania nie są już potrzebne (patrz [“Zarządzanie kronikami w systemie IBM i”](#) na stronie 286).
- b. Rejestruj zmiany w kronice w najnowszym dzienniku za pomocą (**RCDMQMIMG**) a następnie powtórz poprzedni krok. Pozwala to na ponowne wykorzystanie starych nazw dzienników.

W kronice AMQAJRN używana jest opcja MNGRCV (*SYSTEM) , która umożliwia systemowi operacyjnemu automatyczne zmienianie dzienników po osiągnięciu wartości progowej. Więcej informacji na temat zarządzania odbiornikami przez system można znaleźć w sekcji *IBM i Składowanie i odtwarzanie*.

Domyślna wartość progowa dziennika wynosi 100 000 kB. Tę wartość można ustawić na większą wartość podczas tworzenia menedżera kolejek. Początkowa wartość atrybutu LogReceiverSize jest zapisywana w sekcji LogDefaults w pliku mqsr.ini .

Gdy dziennik wykracza poza określony próg, odbiornik jest odłączony i tworzony jest nowy dziennik, dziedziczący atrybuty z poprzedniego dziennika. Zmiany atrybutów LogReceiverSize lub LogASP po utworzeniu menedżera kolejek są ignorowane, gdy system automatycznie podłącza nowy dziennik

Więcej informacji na temat konfigurowania systemu zawiera sekcja [Zmiana informacji konfiguracyjnych w systemie IBM i](#) .

Jeśli konieczna jest zmiana wielkości dzienników po utworzeniu menedżera kolejek, należy utworzyć nowy dziennik i ustawić jego właściciela na QMQM przy użyciu następujących komend:

```
CRTJRNRCV JRNRCV(QM GRLIB/AMQ Arnnnnn) THRESHOLD(xxxxxx) +
TEXT('MQM LOCAL JOURNAL RECEIVER')
CHGOBJOWN OBJ(QM GRLIB/AMQ Arnnnnn) OBJTYPE(*JRNRCV) NEWOWN(QMQM)
```

where

QMGRLIB

Jest nazwą biblioteki menedżera kolejek.

A

Jest to identyfikator instancji (zwykle A).

rnnnnn

Czy następny dziennik w kolejności nazw jest opisany wcześniej

xxxxxx

Jest nowym progiem odbiornika (w kB)

Uwaga: Maksymalna wielkość odbiornika jest zarządzana przez system operacyjny. Aby sprawdzić tę wartość, należy sprawdzić słowo kluczowe THRESHOLD w komendzie **CRTJRNRCV** .

Teraz przyłączaj nowy dziennik do kroniki AMQAJRN za pomocą komendy:

```
CHGJRN JRN(QMGRLIB/AMQ A JRN) JRNRCV(QMGRLIB/AMQ Annnnnn)
```

Szczegółowe informacje na temat zarządzania tymi dziennikami zawiera sekcja [“Zarządzanie kronikami w systemie IBM i”](#) na stronie 286 .

IBM i

Wykorzystanie kroniki menedżera kolejek w systemie IBM i

Te informacje umożliwiają zrozumienie, w jaki sposób produkt IBM MQ for IBM i używa kronik w swojej operacji do sterowania aktualizacjami obiektów lokalnych.

Trwałe aktualizacje kolejek komunikatów mają miejsce w dwóch etapach. Rekordy reprezentujące aktualizację zostaną najpierw zapisane w kronice, a następnie plik kolejki zostanie zaktualizowany.

W związku z tym dzienniki mogą stać się bardziej aktualne niż pliki kolejki. Aby upewnić się, że przetwarzanie restartu rozpoczyna się od spójnego punktu, produkt IBM MQ używa punktów kontrolnych.

Punkt kontrolny jest punktem w czasie, gdy rekord opisany w kronice jest taki sam, jak rekord w kolejce. Sam punkt kontrolny składa się z serii rekordów dziennika wymaganych do zrestartowania menedżera

kolejek. Na przykład stan wszystkich transakcji (tj. jednostek pracy) aktywnych w momencie punktu kontrolnego.

Punkty kontrolne są generowane automatycznie przez program IBM MQ. Są one podejmowane, gdy menedżer kolejek jest uruchamiany i zamykany, a po zarejestrowaniu określonej liczby operacji.

Można wymusić wykonanie punktu kontrolnego przez menedżer kolejek, wydając komendę RCDMQMIMG w odniesieniu do wszystkich obiektów w menedżerze kolejek i wyświetlając wyniki w następujący sposób:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(Q_MGR_NAME) DSPJRNTA(*YES)
```

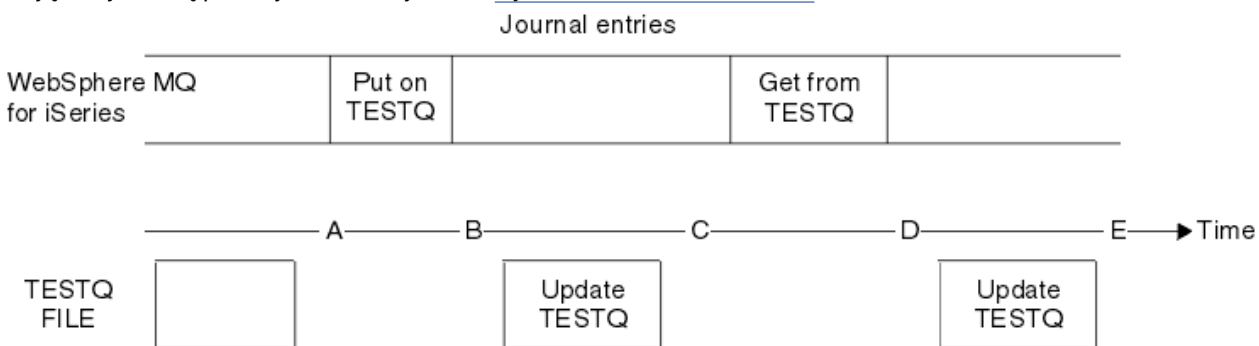
W miarę jak kolejki obsługują kolejne komunikaty, rekord punktu kontrolnego staje się niespójny z bieżącym stanem kolejek.

Po zrestartowaniu programu IBM MQ w dzienniku znajduje się najnowszy rekord punktu kontrolnego. Informacje te są przechowywane w pliku punktu kontrolnego, który jest aktualizowany na końcu każdego punktu kontrolnego. Rekord punktu kontrolnego reprezentuje najbardziej aktualny punkt spójności między dziennikiem a danymi. Dane z tego punktu kontrolnego są używane do odbudowywania kolejek w miarę ich istnienia w czasie punktu kontrolnego. Po ponownym utworzeniu kolejek następuje odtwarzanie dziennika w celu przywrócenia kolejek do stanu, w którym znajdowały się one przed awarią systemu lub z zamkniętą.

Aby zrozumieć, w jaki sposób produkt IBM MQ korzysta z kroniki, należy rozważyć przypadek kolejki lokalnej o nazwie TESTQ w menedżerze kolejek TEST. Jest to reprezentowane przez plik IFS:

```
/QIBM/UserData/mqm/qmgrs/TEST/queues
```

Jeśli określony komunikat jest umieszczany w tej kolejce, a następnie pobrany z kolejki, działania, które mają miejsce, są pokazywane na rysunku [Rysunek 34 na stronie 283](#).



Rysunek 34. Sekwencja zdarzeń podczas aktualizowania obiektów MQM

Pięć punktów od A do E, przedstawionych na diagramie, reprezentuje punkty w czasie, które definiują następujące stany:

- A** Reprezentacja kolejki IFS kolejki jest spójna z informacjami zawartymi w kronice.
- B** Pozycja kroniki jest zapisywana w kronice definiującej operację Put w kolejce.
- C** Odpowiednia aktualizacja zostanie wykonana w kolejce.
- D** Pozycja kroniki jest zapisywana w kronice definiującej operację Get z kolejki.
- E** Odpowiednia aktualizacja zostanie wykonana w kolejce.

Kluczem do możliwości odtwarzania produktu IBM MQ for IBM i jest to, że użytkownik może zapisać reprezentację pliku IFS TESTQ jak w czasie A, a następnie odzyskać reprezentację pliku IFS TESTQ, jak w czasie E, odtwarzając zeskładowany obiekt i odtwarzając pozycje w kronice od czasu A.

Ta strategia jest używana przez produkt IBM MQ for IBM i do odtwarzania trwałych komunikatów po awarii systemu. Program IBM MQ odtwarza określoną pozycję w dziennikach i zapewnia, że po uruchomieniu odtwarza pozycje w kronikach od tego momentu. Ten wpis uruchamiania jest okresowo przeliczany w taki sposób, że program IBM MQ musi wykonać co najmniej niezbędne powtórzenie przy następnym uruchomieniu.

Produkt IBM MQ umożliwia indywidualne odtwarzanie obiektów. Wszystkie informacje trwałe odnoszące się do obiektu są zapisywane w lokalnych kronikach IBM MQ for IBM i. Każdy obiekt IBM MQ, który zostanie uszkodzony lub uszkodzony, może zostać całkowicie odbudowany na podstawie informacji znajdujących się w kronice.

Więcej informacji na temat zarządzania odbiornikami przez system zawiera sekcja [“Dostępność, tworzenie kopii zapasowych, odtwarzanie i restartowanie w systemie IBM i”](#) na stronie 280.

IBM i **Obrazy nośników w systemie IBM i**

W systemie IBM i obraz nośnika jest kompletną kopią obiektu IBM MQ, który jest rejestrowany w kronice. Niektóre uszkodzone lub uszkodzone obiekty mogą być automatycznie odtwarzane z obrazu nośnika.

Obiekt IBM MQ o długim czasie trwania może reprezentować dużą liczbę pozycji kroniki, wracając do punktu, w którym został utworzony. Aby tego uniknąć, program IBM MQ for IBM i ma pojęcie nośnika obrazu obiektu.

Ten obraz nośnika jest kompletną kopią obiektu IBM MQ zapisaną w kronice. Jeśli obraz obiektu jest zabrany, obiekt może zostać odbudowany, odtwarzając pozycje kroniki z tego obrazu. Pozycja w kronice, która reprezentuje punkt powtarzania dla każdego obiektu IBM MQ, jest określana jako pozycja odtwarzania nośnika. Program IBM MQ śledzi następujące elementy:

- Pozycja odtwarzania nośnika dla każdego obiektu menedżera kolejek.
- Najstarsza pozycja z tego zestawu (szczegółowe informacje zawiera komunikat o błędzie AMQ7462 w podręczniku [“Zarządzanie kronikami w systemie IBM i”](#) na stronie 286).

Obrazy obiektu *CTLG i obiektu *MQM są wykonywane regularnie, ponieważ te obiekty mają kluczowe znaczenie dla restartu menedżera kolejek.

Obrazy innych obiektów są podejmowane, gdy jest to wygodne. Domyślnie obrazy wszystkich obiektów są brane pod uwagę, gdy menedżer kolejek jest wyłączone za pomocą komendy **ENDMQM** z parametrem **ENDCCTJOB (*YES)**. Ta operacja może zająć znaczną ilość czasu dla bardzo dużych menedżerów kolejek. Jeśli zachodzi potrzeba szybkiego zamknięcia, należy podać parametr **RCDMQMIMG (*NO)** z parametrem **ENDCCTJOB (*YES)**. W takich przypadkach zaleca się zarejestrowaniem kompletnego obrazu nośnika w kronikach po zrestartowaniu menedżera kolejek za pomocą następującej komendy:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(Q_MGR_NAME)
```

Program IBM MQ automatycznie rejestruje obraz obiektu, jeśli znajdzie wygodny punkt, w którym obiekt może być w sposób kompaktowy opisany przez mały wpis w kronice. Może to jednak nigdy nie wystąpić w przypadku niektórych obiektów, na przykład kolejek, które stale zawierają dużą liczbę komunikatów.

Zamiast zezwalać na kontynuowanie przez niepotrzebnie długi czas najstarszej pozycji odtwarzania nośników, należy użyć komendy IBM MQ **RCDMQMIMG**, która umożliwia ręczne wykonanie obrazu wybranych obiektów.

Odtwarzanie z obrazów nośników

Program IBM MQ automatycznie odtwarza niektóre obiekty ze swojego obrazu nośnika, jeśli zostanie znalezione uszkodzone lub uszkodzone. W szczególności dotyczy to specjalnych obiektów *MQM i *CTLG w ramach normalnego uruchamiania menedżera kolejek. Jeśli jakkolwiek transakcja punktu

synchronizacji była niekompletna w momencie ostatniego zamknięcia menedżera kolejek, każda kolejka, której to dotyczy, jest również odzyskiwana automatycznie, w celu zakończenia operacji uruchamiania.

Inne obiekty należy odzyskać ręcznie za pomocą komendy IBM MQ RCRMQM0BJ. Ta komenda odtwarza pozycje w kronice w celu ponownego utworzenia obiektu IBM MQ . Jeśli obiekt IBM MQ stanie się uszkodzony, jedynymi poprawnymi działaniami są usunięcie go lub ponowne utworzenie przez tę metodę. Należy jednak pamiętać, że nietrwale komunikaty nie mogą być odtwarzane w ten sposób.

IBM i Punkty kontrolne w systemie IBM MQ for IBM i

Punkty kontrolne są podejmowane w różnych momentach w celu udostępnienia znanego spójnego punktu początkowego do odtwarzania.

Proces punktu kontrolnego AMQALMPX jest odpowiedzialny za wykonanie punktu kontrolnego w następujących punktach:

- Uruchamianie menedżera kolejek (STRMQM).
- Zamknięcie menedżera kolejek (ENDMQM).
- Po upływie określonego czasu od ostatniego punktu kontrolnego (domyślny okres wynosi 30 minut), a od poprzedniego punktu kontrolnego zapisano minimalną liczbę rekordów dziennika (wartość domyślna to 100).
- Po zapisaniu pewnej liczby rekordów dziennika. Wartość domyślna to 10 000.
- Po przekroczeniu wartości progowej kroniki i automatycznie tworzony jest nowy dziennik.
- Gdy pełny obraz nośnika jest brany pod uwagę:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(Q_MGR_NAME) DSPJRNDTA(*YES)
```

IBM i Kopie zapasowe danych produktu IBM MQ for IBM i

Ten temat zawiera informacje na temat dwóch typów kopii zapasowych IBM MQ dla każdego menedżera kolejek.

Dla każdego menedżera kolejek istnieją dwa typy kopii zapasowych IBM MQ , które należy rozważyć:

- Tworzenie kopii zapasowych danych i dzienników.
Aby upewnić się, że oba zestawy danych są spójne, należy wykonać tę czynności tylko po zamknięciu menedżera kolejek.
- Tworzenie kopii zapasowej kroniki.
Można to zrobić, gdy menedżer kolejek jest aktywny.

W przypadku obu metod należy znaleźć nazwy katalogu IFS menedżera kolejek i biblioteki menedżera kolejek. Informacje te można znaleźć w pliku konfiguracyjnym produktu IBM MQ (mqc.ini). Więcej informacji na ten temat zawiera sekcja [Sekcja QueueManager](#).

Aby wykonać oba typy kopii zapasowych, wykonaj następujące procedury:

Tworzenie kopii zapasowej danych i kronik konkretnego menedżera kolejek

Uwaga: Jeśli menedżer kolejek jest uruchomiony, nie należy używać żądania składowania podczas użycia. Takie żądanie nie może zostać zakończone, chyba że wszystkie definicje kontroli transakcji z oczekującymi zmianami zostaną zatwierdzone lub wycofane. Jeśli ta komenda jest używana, gdy menedżer kolejek jest aktywny, połączenia kanału mogą nie kończyć się normalnie. Zawsze należy używać następującej procedury.

1. Utwórz pusty dziennik, używając komendy:

```
CHGJRN JRN(QMTEST/AMQAJRN) JRNRCV(*GEN)
```

2. Użyj komendy **RCDMQMIMG** , aby zarejestrować obraz MQM dla wszystkich obiektów IBM MQ , a następnie wymuszaj punkt kontrolny przy użyciu komendy:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) DSPJRNDTA(*YES) MQMNAME(TEST)
```

3. Zakończ kanały i upewnij się, że menedżer kolejek nie jest uruchomiony. Jeśli menedżer kolejek jest uruchomiony, zatrzymaj go za pomocą komendy **ENDMQM** .
4. Utwórz kopię zapasową biblioteki menedżera kolejek, wydając następującą komendę:

```
SAVLIB LIB(QMTEST)
```

5. Utwórz kopię zapasową katalogów IFS menedżera kolejek, wydając następującą komendę:

```
SAV DEV(...) OBJ((' /QIBM/UserData/mqm/qmgrs/test '))
```

Tworzenie kopii zapasowej dziennika określonego menedżera kolejek

Ponieważ wszystkie istotne informacje są przechowywane w kronikach, o ile w danym momencie wykonywane jest pełne składowanie, można utworzyć częściowe kopie zapasowe, zapisując dzienniki. Rejestrują wszystkie zmiany od czasu utworzenia pełnej kopii zapasowej i są wykonywane przez wydanie następujących komend:

1. Utwórz pusty dziennik, używając komendy:

```
CHGJRN JRN(QMTEST/AMQAJRN) JRNRCV(*GEN)
```

2. Użyj komendy **RCDMQMIMG** , aby zarejestrować obraz MQM dla wszystkich obiektów IBM MQ , a następnie wymuszaj punkt kontrolny przy użyciu komendy:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) DSPJRNDTA(*YES) MQMNAME(TEST)
```

3. Zeskładuj dzienniki za pomocą komendy:

```
SAVOBJ OBJ(AMQ*) LIB(QMTEST) OBJTYPE(*JRNRCV) .....
```

Prosta strategia tworzenia kopii zapasowych polega na wykonywaniu pełnych kopii zapasowych bibliotek produktu IBM MQ co tydzień i tworzeniu kopii zapasowych dziennika dziennego. To, oczywiście, zależy od tego, jak skonfigurować strategię składowania dla przedsiębiorstwa.

Zarządzanie kronikami w systemie IBM i

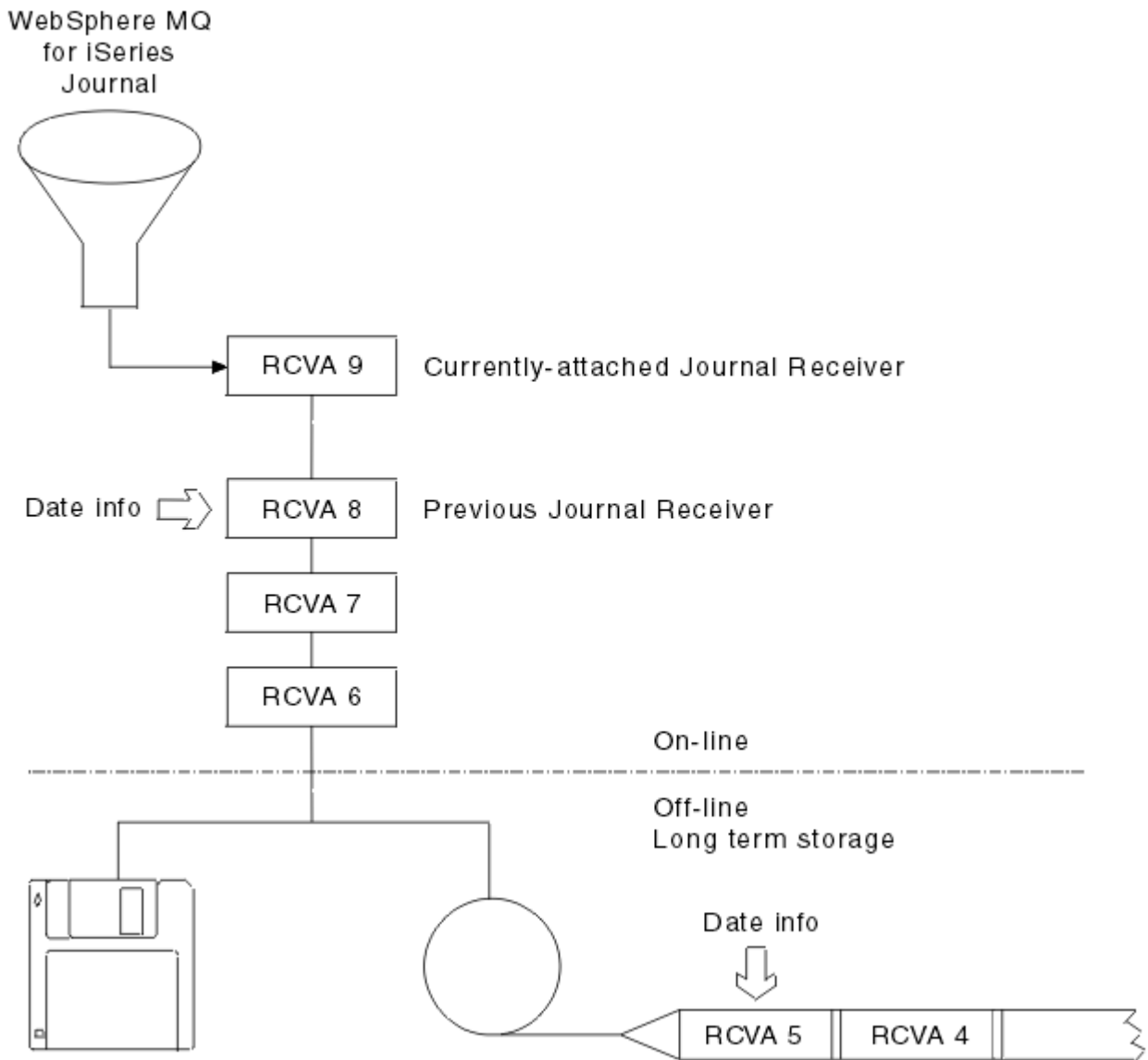
W ramach strategii składowania, należy zadbać o swoje dzienniki. Z różnych powodów użyteczne jest usuwanie dzienników z bibliotek produktu IBM MQ :

- Zwalnianie miejsca; dotyczy to wszystkich dzienników
- Aby zwiększyć wydajność podczas uruchamiania (STRMQM)
- W celu zwiększenia wydajności obiektów rekreacyjnych (RCRMQMOBJ)

Przed usunięciem dziennika należy zwrócić uwagę na to, że istnieje kopia zapasowa i że nie jest już potrzebny dziennik.

Dzienniki mogą być usuwane z biblioteki menedżera kolejek *po* odłączeniu ich od kronik i zeskładowanych, pod warunkiem, że są one dostępne do odtworzenia, jeśli jest to konieczne dla operacji odtwarzania.

Pojęcie zarządzania kronikami jest przedstawione w sekcji [Rysunek 35 na stronie 287](#).



Rysunek 35. Kronikowanie w systemie IBM i

Ważne jest, aby wiedzieć, jak daleko w kronikach IBM MQ prawdopodobnie trzeba będzie przejść, aby określić, kiedy dziennik, który został zeszkodowany, może zostać usunięty z biblioteki menedżera kolejek, a gdy sama kopia zapasowa może zostać usunięta.

Program IBM MQ wysyła dwa komunikaty do kolejki komunikatów menedżera kolejek (QMQMMSG w bibliotece menedżera kolejek), aby określić ten czas. Komunikaty te są wysyłane po uruchomieniu, gdy zmienia się lokalny dziennik, a użytkownik RCDMQIMG używa do wymuszenia punktu kontrolnego. Są to dwa następujące komunikaty:

AMQ7460

Uruchamianie punktu odtwarzania. Ten komunikat definiuje datę i godzinę uruchomienia pozycji startowej, z której program IBM MQ odtwarza dziennik w przypadku uruchomienia procedury odtwarzania po uruchomieniu. Jeśli dziennik, który zawiera ten rekord, jest dostępny w bibliotekach produktu IBM MQ, ten komunikat zawiera również nazwę dziennika zawierającego rekord.

AMQ7462

Najstarsza pozycja odtwarzania nośnika. Ten komunikat definiuje datę i godzinę najstarszej pozycji, która ma zostać użyta do ponownego utworzenia obiektu na podstawie jego obrazu nośnika.

Zidentyfikowany dziennik jest najstarszy, który jest wymagany. Żadne inne dzienniki programu IBM MQ ze starszymi terminami tworzenia nie są już potrzebne. Jeśli wyświetlane są tylko gwiazdki, należy odtworzyć kopie zapasowe z podanej daty, aby określić, która jest najstarszym dziennikiem.

Gdy te komunikaty są protokołowane, program IBM MQ zapisuje także obiekt przestrzeni użytkownika w bibliotece menedżera kolejek, który zawiera tylko jedną pozycję: nazwę najstarszego dziennika, który musi być przechowywany w systemie. Ten obszar użytkownika nosi nazwę AMQJRNINF, a dane są zapisywane w formacie:

```
JJJJJJJJJLLLLLLLLLLLLYYYYMMDDHHMMSSmmm
```

gdzie:

JJJJJJJJJJ

Jest najstarszą nazwą dziennika, której nadal potrzebuje produkt IBM MQ .

LLLLLLLLLL

Jest to nazwa biblioteki dziennika.

YYYY

Jest rokiem najstarszej pozycji kroniki, która jest wymagana przez produkt IBM MQ .

MM

Jest to miesiąc najstarszej pozycji kroniki, która jest wymagana przez produkt IBM MQ .

DD

Jest to dzień najstarszej pozycji kroniki, która jest wymagana przez produkt IBM MQ .

HH

Jest godzina najstarszej pozycji kroniki, której potrzebuje IBM MQ .

SS

Jest to liczba sekund najstarszej pozycji kroniki, która jest wymagana przez produkt IBM MQ .

mmm

Jest to milisekundy najstarszej pozycji kroniki, która jest wymagana przez produkt IBM MQ .

Gdy najstarszy dziennik został usunięty z systemu, w tym obszarze użytkownika znajdują się gwiazdki (*) dla nazwy dziennika.

Uwaga: Okresowe wykonywanie programu RCDMQMIMG OBJ (*ALL) OBJTYPE (*ALL) DSPJRNDTA (*YES) może zaoszczędzić czas uruchamiania systemu IBM MQ i zmniejszyć liczbę dzienników lokalnych, które należy zeszkładować i odtworzyć w celu odtworzenia.

Produkt IBM MQ for IBM i nie odnosi się do dzienników, chyba że wykonuje się przejście do odtwarzania dla uruchamiania lub do ponownego utworzenia obiektu. Jeśli stwierdzi, że nie istnieje kronika, która nie jest obecna, wysła komunikat AMQ7432 do kolejki komunikatów menedżera kolejek (QMOMMSG), raportując datę i godzinę pozycji kroniki, która musi zakończyć proces odzyskiwania.

Jeśli tak się stanie, odtwórz wszystkie dzienniki, które po tej dacie zostały odłączone od kopii zapasowej, aby umożliwić pomyślny przebieg odtwarzania.

Należy zachować dziennik, który zawiera pozycję uruchamiania, oraz wszystkie kolejne dzienniki dostępne w bibliotece menedżera kolejek.

Przechowuj dziennik zawierający najstarsze Media Recovery Entryi wszystkie kolejne dzienniki, które są dostępne przez cały czas i które znajdują się w bibliotece menedżera kolejek lub w kopii zapasowej.

Wymuszenie punktu kontrolnego:

- Jeśli dziennik nazwany w AMQ7460 nie jest zaawansowany, oznacza to, że istnieje niekompletna jednostka pracy, która musi zostać zatwierdzona lub wycofana.
- Jeśli dziennik nazwany w AMQ7462 nie jest zaawansowany, oznacza to, że istnieje jeden lub więcej uszkodzonych obiektów.

w systemie IBM i

Ten temat zawiera informacje dotyczące odtwarzania jednego lub większej liczby menedżerów kolejek z kopii zapasowej lub z komputera zdalnego.

Jeśli konieczne jest odzyskanie jednego lub większej liczby menedżerów kolejek produktu IBM MQ z kopii zapasowej, wykonaj następujące kroki.

1. Wygaś menedżery kolejek produktu IBM MQ .
2. Znajdź najnowszy zestaw kopii zapasowych, składający się z najnowszej pełnej kopii zapasowej, a następnie kopii zapasowej dzienników.
3. Wykonaj operację RSTLIB, używając pełnej kopii zapasowej, aby odtworzyć biblioteki danych produktu IBM MQ do stanu w momencie tworzenia pełnej kopii zapasowej, wprowadzając następujące komendy:

```
RSTLIB LIB(QMQLIB1) .....
RSTLIB LIB(QMQLIB2) .....
```

Jeśli dziennik został częściowo zeskładowany w jednej kopii zapasowej dziennika, a następnie został w pełni zapisany w kolejnej kopii zapasowej, odtwórz tylko w pełni zeskładowany. Odtwarzanie kronik pojedynczo, w porządku chronologicznym.

4. Wykonaj operację RST, aby odtworzyć katalogi IFS systemu IBM MQ do systemu plików IFS, używając następującej komendy:

```
RST DEV(...) OBJ(('QIBM/UserData/mqm/qmgrs/testqm')) ...
```

5. Uruchom menedżer kolejek komunikatów. Spowoduje to odtworzenie wszystkich rekordów dziennika zapisanych od czasu utworzenia pełnej kopii zapasowej i odtworzenie wszystkich obiektów IBM MQ do stanu spójnego w momencie tworzenia kopii zapasowej kroniki.

Aby odtworzyć pełny menedżer kolejek na innym komputerze, należy wykonać poniższą procedurę w celu odtworzenia wszystkich elementów z biblioteki menedżera kolejek. (TEST jako przykładową nazwę menedżera kolejek).

1. CRTMQM TEST
2. DLTLIB LIB(QMTEST)
3. RSTLIB SAVLIB(QMTEST) DEV(*SAVF) SAVF(QMGRLIBSAV)
4. Usuń następujące pliki IFS:

```
/QIBM/UserData/mqm/qmgrs/TEST/QMQMCHKPT
/QIBM/UserData/mqm/qmgrs/TEST/qmanager/QMQMOBJCAT
/QIBM/UserData/mqm/qmgrs/TEST/qmanager/QMANAGER
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.AUTH.DATA.QUEUE/q
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.CHANNEL.INITQ/q
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.CLUSTER.COMMAND.QUEUE/q
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.CLUSTER.REPOSITORY.QUEUE/q
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.CLUSTER.TRANSMIT.QUEUE/q
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.PENDING.DATA.QUEUE/q
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.ADMIN.COMMAND.QUEUE/q
```

5. STRMQM TEST
6. RCRMQMOBJ OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(TEST)

IBM i

Te informacje umożliwiają zrozumienie różnych sposobów odtwarzania dzienników.

Najczęstszym działaniem jest odtworzenie kopii zapasowej dziennika do biblioteki menedżera kolejek, jeśli ponownie potrzebny jest odbiornik, który został usunięty z powodu późniejszej funkcji odtwarzania.

Jest to proste zadanie i wymagane jest, aby dzienniki zostały odtworzone za pomocą standardowej komendy IBM i RSTOBJ:

```
RSTOBJ OBJ(QMQMDATA/AMQA000005) OBJTYPE(*JRRCV) .....
```

Może być konieczne odtworzenie szeregu dzienników, a nie pojedynczego odbiornika. Na przykład, AMQA000007 jest najstarszym odbiornikiem w bibliotekach IBM MQ, a zarówno AMQA000005, jak i AMQA000006 muszą zostać odtworzone.

W takim przypadku odtwarza odbiorniki indywidualnie w odwrotnym porządku chronologicznym. Nie zawsze jest to konieczne, ale jest dobrą praktyką. W ciężkich sytuacjach może być konieczne użycie komendy IBM i WRKJRNA w celu powiązania odtworzonych dzienników z kroniką.

Podczas odtwarzania dzienników system automatycznie tworzy przyłączony dziennik z nową nazwą w sekwencji dziennika. Jednak nowa generowana nazwa może być taka sama, jak dziennik, który należy odtworzyć. Aby rozwiązać ten problem, wymagana jest ręczna interwencja; aby utworzyć nowy dziennik nazw w kolejności, a nowy dziennik przed odtworzeniem dziennika.

Na przykład można rozważyć problem z zapisaną kroniką AMQAJRN i następującymi dziennikami:

- AMQA000000
- AMQA100000
- AMQA200000
- AMQA300000
- AMQA400000
- AMQA500000
- AMQA600000
- AMQA700000
- AMQA800000
- AMQA900000

Podczas odtwarzania kroniki AMQAJRN do biblioteki menedżera kolejek system automatycznie tworzy dziennik AMQA000000. Ten automatycznie generowany dziennik jest w konflikcie z jednym z istniejących dzienników (AMQA000000), które mają zostać odtworzone, których nie można odtworzyć.

Rozwiązaniem jest:

1. Utwórz ręcznie następny dziennik (patrz [“Dzienniki menedżera kolejek w systemie IBM i” na stronie 281](#)):

```
CRTJRRCV JRRCV(QMQLIB/AMQA900001) THRESHOLD(XXXXX)
```

2. Ręcznie utwórz kronikę z dziennikiem:

```
CRTJRN JRN(QMGLIB/AMQAJRN) MNGRCV(*SYSTEM) +  
JRRCV(QMGLIB/AMQA900001) MSGQ(QMGLIB/AMQAJRNMSG)
```

3. Odtwórz dzienniki lokalne AMQA000000 do AMQA900000.

IBM i

Menedżery kolejek z wieloma instancjami w systemie IBM i

Menedżery kolejek z wieloma instancjami poprawiają dostępność, automatycznie przełączając się na serwer rezerwowy, jeśli ten aktywny serwer ulegnie awarii. Serwery aktywne i rezerwowe są wieloma instancjami tego samego menedżera kolejek; współużytkują one te same dane menedżera kolejek. Jeśli

aktywna instancja nie powiedzie się, należy przestać jej kronikę do rezerwowej bazy danych, która przejmuje tę operację, tak aby menedżer kolejek mógł odbudować kolejki.

Skonfiguruj systemy IBM i , na których są uruchomione menedżery kolejek z wieloma instancjami, tak aby w przypadku awarii aktywnej instancji menedżera kolejek dziennik, który jest używany, był dostępny dla instancji rezerwowej, która przejmuje tę instancję. Użytkownik może zaprojektować własne zadania konfiguracyjne i administracyjne, aby dziennik z aktywnej instancji był dostępny dla instancji, która przejmuje tę kronikę. Jeśli komunikaty nie mają być tracone, należy upewnić się, że kronika rezerwowa jest spójna z dziennikiem aktywnym w punkcie awarii. Projekt można dostosować z jednej z dwóch konfiguracji opisanych za pomocą przykładów w kolejnych tematach, które zachowują spójność.

1. Należy utworzyć kopię lustrzaną kroniki z systemu, w którym działa aktywna instancja menedżera kolejek, do systemów, w których działają instancje rezerwowe.
2. Umieść kronikę w niezależnej puli pamięci dyskowej (IASP), która jest przenoszalna z systemu, w którym działa aktywna instancja, do instancji rezerwowej.

Pierwsze rozwiązanie nie wymaga żadnego dodatkowego sprzętu ani oprogramowania, ponieważ używa podstawowych pul ASP. Drugie rozwiązanie wymaga przełączalnych pul IASP, które wymagają obsługi technologii klastrowej IBM i , która jest dostępna jako osobno wyceniony produkt licencji IBM i 5761-SS1 opcja 41.

IBM i **Niezawodność i dostępność w systemie IBM i**

Menedżery kolejek z wieloma instancjami mają na celu zwiększenie dostępności aplikacji. Ograniczenia technologiczne i fizyczne oznaczają, że potrzebne są różne rozwiązania, aby sprostać wymaganiom odzyskiwania po awarii, tworzenia kopii zapasowych menedżerów kolejek i ciągłych operacji.

Konfigurując niezawodność i dostępność, można wymienić dużą liczbę czynników, co skutkuje czterema odrębnymi punktami projektowym:

Odtwarzanie po awarii

Zoptymalizowany do odzyskiwania po wystąpieniu poważnej awarii, która niszczy wszystkie lokalne zasoby aplikacyjne.

Odtwarzanie po awarii w systemie IBM i jest często oparte na geograficznym zapisem lustrzanym puli IASP.

Składowanie

Zoptymalizowany pod kątem odtwarzania po nieudanej zlokalizowanej awarii, często występuje błąd ludzki lub nieprzewidziany problem techniczny.

Produkt IBM MQ udostępnia menedżery kolejek kopii zapasowych, które okresowo będą tworzyć kopie zapasowe menedżerów kolejek. Można również użyć replikacji asynchronicznej dzienników menedżera kolejek w celu poprawienia waluty kopii zapasowej.

Dostępność

Zoptymalizowane do odtwarzania operacji, które szybko dają wygląd prawie nieprzerwanej usługi po przewidywalnych awariach technicznych, takich jak awaria serwera lub dysku.

Odzyskiwanie jest zazwyczaj mierzone w minutach, przy czym wykrywanie jest czasami dłuższe niż proces odtwarzania. Menedżer kolejek z wieloma instancjami wspomaga konfigurowanie *dostępności*.

Praca ciągła

Zoptymalizowane pod kątem udostępniania nieprzerwanej usługi.

Rozwiązania w zakresie pracy ciągłej muszą rozwiązywać problemy związane z wykrywaniem i prawie zawsze angażować się w te same prace przy użyciu więcej niż jednego systemu i albo przy użyciu pierwszego rezultatu, albo też, jeśli poprawność jest ważniejsza, porównując co najmniej dwa wyniki.

Menedżer kolejek z wieloma instancjami wspomaga konfigurowanie *dostępności*. Jedna instancja menedżera kolejek jest aktywna w danym momencie. Przełączenie na instancję rezerwową trwa od nieco więcej niż 10 sekund do 15 minut lub więcej, w zależności od tego, jak system jest skonfigurowany, załadowany i dostrojony.

Menedżer kolejek z wieloma instancjami może dać wygląd prawie nieprzerwanej usługi, jeśli jest używana z możliwością ponownego połączenia IBM MQ MQI clients, które są w stanie kontynuować przetwarzanie bez konieczności świadomego wyłączenia menedżera kolejek przez program użytkowy. Patrz temat [Automated client reconnection](#)(Automated client reconnection).

IBM i

Komponenty rozwiązania wysokiej dostępności w systemie IBM i

Utwórz rozwiązanie wysokiej dostępności, używając menedżerów kolejek z wieloma instancjami, udostępniając stabilną pamięć masową w sieci dla danych menedżera kolejek, replikację kroniki lub stabilną pamięć masową IASP dla dzienników menedżera kolejek i przy użyciu klientów z możliwością ponownego połączenia, aplikacji skonfigurowanych jako restartowalne usługi menedżera kolejek.

Menedżer kolejek o wielu instancjach reaguje na wykrywanie awarii menedżera kolejek, wznawiając uruchamianie innej instancji menedżera kolejek na innym serwerze. Aby zakończyć proces uruchamiania, instancja musi mieć dostęp do współużytkowanych danych menedżera kolejek w sieciowej pamięci masowej i do jej kopii dziennika lokalnego menedżera kolejek.

Aby utworzyć rozwiązanie wysokiej dostępności, należy zarządzać dostępnością danych menedżera kolejek, waluty dziennika lokalnego menedżera kolejek i zbudować aplikacje klienckie z możliwością ponownego połączenia albo wdrożyć aplikacje jako usługi menedżera kolejek w celu automatycznego restartu po wznowieniu menedżera kolejek. Automatyczne ponowne nawiązywanie połączenia przez klient nie jest obsługiwane przez produkt IBM MQ classes for Java.

Dane menedżera kolejek

Umieść dane menedżera kolejek w sieciowej pamięci masowej, która jest współużytkowana, wysoce dostępna i niezawodna, prawdopodobnie za pomocą dysków o poziomie RAID 1 lub większej. System plików musi spełniać wymagania dotyczące współużytkowanego systemu plików dla menedżerów kolejek z wieloma instancjami. Więcej informacji na temat wymagań dla współużytkowanych systemów plików zawiera sekcja [Wymagania dotyczące współużytkowanych systemów plików](#). Network File System Version 4 (NFS4) jest protokołem spełniający te wymagania.

Kroniki menedżera kolejek

Należy również skonfigurować dzienniki produktu IBM i używane przez instancje menedżera kolejek, tak aby instancja rezerwowa mogła odtworzyć dane jej menedżera kolejek w stanie spójnym. W przypadku nieprzerwanej usługi oznacza to, że konieczne jest odtworzenie dzienników do ich stanu, gdy aktywna instancja nie powiodła się. W przeciwieństwie do rozwiązań do tworzenia kopii zapasowych lub usuwania skutków awarii, odtwarzanie dzienników do wcześniejszego punktu kontrolnego nie jest wystarczające.

Nie można fizycznie współużytkować dzienników między wieloma systemami IBM i w pamięci masowej sieciowej. Aby odtworzyć dzienniki menedżera kolejek do stanu spójnego w punkcie awarii, należy przesłać kronikę fizyczną, która była lokalna, do aktywnej instancji menedżera kolejek w momencie awarii nowej instancji, która została aktywowana, lub w celu zachowania kopii lustrzanych kroniki w działających rezerwowych instancjach. Kronika lustrzana jest zdalną repliką kroniki, która została dokładnie zsynchronizowana z kroniką lokalną należącą do instancji, której wykonanie nie powiodło się.

Trzy konfiguracje są punktami uruchamiania dla projektowania sposobu zarządzania kronikami dla menedżera kolejek z wieloma instancjami,

1. Użycie zsynchronizowanej replikacji kroniki (zapisu lustrzanego kroniki) z aktywnej puli ASP instancji do rezerwowych ASP instancji.
2. Przesyłanie puli IASP, która została skonfigurowana do przechowywania kroniki menedżera kolejek z aktywnej instancji do instancji rezerwowej, która trwa przejmując jako aktywną instancję.
3. Korzystanie z zsynchronizowanych dodatkowych kopii lustrzanych IASP.

Więcej informacji na temat umieszczania danych menedżera kolejek na iASPs komendzie CRTMQM systemu IBM MQ IBM i zawiera sekcja [ASP](#).

Więcej informacji na ten temat zawiera sekcja [Wysoka dostępność](#) w podręczniku IBM Documentation.

Aplikacje

Aby zbudować klient w celu automatycznego ponownego nawiązania połączenia z menedżerem kolejek, gdy rezerwowany menedżer kolejek wznowi działanie, należy połączyć aplikację z menedżerem kolejek przy użyciu MQCONNX i określić MQCNO_RECONNECT_Q_MGR w polu **MQCNO** Opcje . Więcej informacji na temat projektowania aplikacji klienckich do odtwarzania można znaleźć w sekcji [Programy przykładowe o wysokiej dostępności dla trzech przykładowych programów z użyciem klientów z możliwością ponownego połączenia oraz Odtwarzanie aplikacji](#) .

IBM i Tworzenie współużytkowanego zasobu sieciowego dla danych menedżera kolejek przy użyciu serwera NetServer w systemie IBM i

Utwórz współużytkowany zasób sieciowy na serwerze IBM i w celu przechowywania danych menedżera kolejek. Skonfiguruj połączenia z dwóch serwerów, które mają być hostem instancji menedżera kolejek, aby uzyskać dostęp do współużytkowanego zasobu sieciowego.

Zanim rozpoczniesz

- Do wykonania tej czynności wymagane są trzy serwery IBM i . Współużytkowany zasób sieciowy jest zdefiniowany na jednym z serwerów, GAMMA. Pozostałe dwa serwery, ALPHA i BETA, mają łączyć się z GAMMA.
- Zainstaluj produkt IBM MQ na wszystkich trzech serwerach.
- Zainstaluj program System i Navigator; patrz [System i Navigator](#).

O tym zadaniu

- Utwórz katalog menedżera kolejek na GAMMA i ustaw poprawne prawa własności i uprawnienia dla profili użytkowników QMQM i QMQMADM. Katalog i uprawnienie można łatwo utworzyć, instalując produkt IBM MQ w systemie GAMMA.
- Użyj opcji System i Navigator , aby utworzyć zasób współużytkowany do katalogu danych menedżera kolejek w GAMMA.
- Utwórz katalogi na ALPHA i BETA, które wskazują na zasób współużytkowany.

Procedura

1. W przypadku GAMMA utwórz katalog, w którym mają być hostem dane menedżera kolejek z profilem użytkownika produktu QMQM jako właścicielem, a QMQMADM jako grupą podstawową.

Wskazówka:

Szybkim i niezawodnym sposobem na utworzenie katalogu z poprawnymi uprawnieniami jest zainstalowanie produktu IBM MQ w systemie GAMMA.

Jeśli później nie ma potrzeby uruchamiania programu IBM MQ w systemie GAMMA, należy zdeinstalować produkt IBM MQ. Po deinstalacji katalog /QIBM/UserData/mqm/qmgrs pozostaje w GAMMA z profilem użytkownika właściciela QMQM , a QMQMADM grupą podstawową.

Zadanie korzysta z katalogu /QIBM/UserData/mqm/qmgrs w GAMMA dla zasobu współużytkowanego.

2. Uruchom kreator System i Navigator **Dodaj połączenie** i nawiąże połączenie z systemem GAMMA.
 - a) Kliknij dwukrotnie ikonę **System i Navigator** na pulpicie produktu Windows .
 - b) Kliknij przycisk **Tak** , aby utworzyć połączenie.
 - c) Postępuj zgodnie z instrukcjami w kreatorze **Dodaj połączenie** i utwórz połączenie z systemu IBM i do GAMMA.

Połączenie z GAMMA jest dodawane do obszaru **Moje połączenia**.

3. Dodaj nowy współużytkowany zasób plisowy dla GAMMA.
 - a) In the **System i Navigator** window, click the File Shares folder in My Connections/GAMMA/File Systems.

- b) W oknie **Moje zadania** kliknij opcję **Zarządzaj współużytkownikami IBM i NetServer**.
- Na pulpicie zostanie otwarte nowe okno **IBM i NetServer -GAMMA** i wyświetlane są obiekty współużytkowane.
- c) Kliknij prawym przyciskiem myszy folder **Shared Objects > Plik > Nowy > Plik**.
- Zostanie otwarte nowe okno **IBM i NetServer File Share - GAMMA**.
- d) Nadaj nazwę, na przykład **WMQ**, nazwę zasobu współużytkowanego.
- e) Ustaw kontrolę dostępu na **Read/Write**.
- f) Wybierz opcję **Nazwa ścieżki**, przeglądając katalog **/QIBM/UserData/mqm/qmgrs**, który został wcześniej utworzony, i kliknij przycisk **OK**.
- Okno **IBM i NetServer File Share-GAMMA** zostanie zamknięte, a program **WMQ** zostanie wyświetlony w oknie obiektów współużytkowanych.
4. Kliknij prawym przyciskiem myszy pozycję **WMQ** w oknie obiektów współużytkowanych. Kliknij opcję **Plik > Uprawnienia**.

Zostanie otwarte okno **Qmgrs Permissions-GAMMA** (Uprawnienia Qmgrs-GAMMA) dla obiektu **/QIBM/UserData/mqm/qmgrs**.

- a) Sprawdź następujące uprawnienia dla produktu **QMQM**, jeśli nie zostały jeszcze ustawione:
- Read
 - Write
 - Execute
 - Management
 - Existence
 - Alter
 - Reference
- b) Sprawdź następujące uprawnienia dla produktu **QMQMADM**, jeśli nie zostały jeszcze ustawione:
- Read
 - Write
 - Execute
 - Reference
- c) Dodaj inne profile użytkowników, które mają nadać uprawnienia do produktu **/QIBM/UserData/mqm/qmgrs**.
- Na przykład można nadać domyślny profil użytkownika (publiczny) **Read** i uprawnienia **Execute** do produktu **/QIBM/UserData/mqm/qmgrs**.
5. Sprawdź, czy wszystkie profile użytkowników, którym nadano dostęp do produktu **/QIBM/UserData/mqm/qmgrs** w systemie **GAMMA**, mają takie samo hasło, jakie są dostępne na serwerach, które uzyskują dostęp do **GAMMA**.

W szczególności należy upewnić się, że profile użytkowników produktu **QMQM** na innych serwerach, które mają dostęp do zasobu współużytkowanego, mają takie samo hasło, jak profil użytkownika produktu **QMQM** w systemie **GAMMA**.

Wskazówka: Kliknij folder **My Connections/GAMMA/Users and Groups** w oknie **System i Navigator**, aby ustawić hasła. Alternatywnie można użyć komend **CHFUSRPRF** i **CHGPWD**.

Wyniki

Sprawdź, czy można uzyskać dostęp do **GAMMA** z innych serwerów, korzystając z zasobu współużytkowanego. Jeśli wykonujesz inne zadania, sprawdź, czy można uzyskać dostęp do **GAMMA** z poziomu **ALPHA** i **BETA**, używając ścieżki **/QNTC/GAMMA/WMQ**. Jeśli katalog **/QNTC/GAMMA** nie istnieje

w systemie ALPHA lub BETA, należy utworzyć ten katalog. W zależności od domeny NetServer przed utworzeniem tego katalogu może być konieczne wykonanie IPL ALPHA lub BETA.

```
CRTDIR DIR('/QNTC/GAMMA')
```

Po sprawdzeniu, czy masz dostęp do /QNTC/GAMMA/WMQ z ALPHA lub BETA, wydając komendę, CRTMQM MQMNAME('QM1') MQMDIRP('/QNTC/GAMMA/WMQ') tworzy /QIBM/UserData/mqm/qmgrs/QM1 na GAMMA.

Co dalej

Menedżer kolejek o wielu instancjach należy utworzyć, wykonując kroki opisane w jednym z zadań: [“Tworzenie menedżera kolejek z wieloma instancjami przy użyciu zapisu lustrzanego dziennika i serwera NetServer w systemie IBM i” na stronie 306](#) lub [“Przekształcanie menedżera kolejek w pojedynczej instancji w menedżer kolejek z wieloma instancjami przy użyciu serwera NetServer i zapisu lustrzanego kroniki w systemie IBM i” na stronie 310](#).

IBM i **Wydajność przełączania awaryjnego w systemie IBM i**

Czas potrzebny do wykrycia instancji menedżera kolejek nie powiódł się, a następnie wznowienie przetwarzania w trybie gotowości może się wahać od kilkudziesięciu sekund do piętnastu minut lub więcej w zależności od konfiguracji. Wydajność musi być poważną kwestią przy projektowaniu i testowaniu rozwiązania wysokiej dostępności.

W celu określenia, czy menedżer kolejek z wieloma instancjami ma być używany do replikacji kroniki, czy do korzystania z puli IASP, należy rozważyć zalety i wady. Zapis lustrzany wymaga, aby menedżer kolejek zapisał synchronicznie do kroniki zdalnej. Z punktu widzenia sprzętu nie ma to wpływu na wydajność, ale z perspektywy oprogramowania występuje większa długość ścieżki zapisanej w kronice zdalnej, niż tylko do kroniki lokalnej, co może być w pewnym stopniu zmniejszone o wydajność działającego menedżera kolejek. Jednak gdy rezerwowy menedżer kolejek przejmie, opóźnienie synchronizacji lokalnej kroniki ze zdalnej kroniki utrzymanej przez aktywną instancję przed jej niepowodzeniem, jest zwykle niewielkie w porównaniu z czasem, w którym program IBM i wykryje i prześle pulę IASP do serwera, na którym działa instancja rezerwowa menedżera kolejek. Czasy transferu IASP mogą być nawet od dziesięciu do piętnastu minut, a nie być zakończone w sekundach. Czas przesyłania IASP zależy od liczby obiektów, które muszą być *udostępnione*, gdy pula IASP jest przesyłana do systemu rezerwowego, oraz wielkość ścieżek dostępu lub indeksów, które muszą zostać scalone.

Gdy rezerwowy menedżer kolejek przejmie, opóźnienie synchronizacji lokalnej kroniki ze zdalnej kroniki utrzymanej przez aktywną instancję przed jej niepowodzeniem, jest zwykle niewielkie w porównaniu z czasem, w którym program IBM i wykryje i prześle niezależną ASP do serwera, na którym działa instancja rezerwowa menedżera kolejek. Czasy transferu niezależnej puli ASP mogą być nawet od dziesięciu do piętnastu minut, a nie mogą być wykonywane w sekundach. Czas transferu niezależnej ASP zależy od liczby obiektów, które muszą być *różnicowane*, gdy niezależna ASP jest przesyłana do systemu rezerwowego, oraz wielkość ścieżek dostępu lub indeksów, które muszą zostać scalone.

Jednak przesyłanie kroniki nie jest jedynym czynnikiem wpływającym na czas potrzebny do pełnego wznowienia działania instancji rezerwowej. Należy również rozważyć czas potrzebny na zwolnienie blokady przez system plików z kolejki na dane menedżera kolejek, które sygnalizuje instancję rezerwową, aby spróbować kontynuować jego uruchomienie, a także czas potrzebny do odtworzenia kolejek z kroniki, aby instancja mogła ponownie rozpocząć przetwarzanie komunikatów. Te inne źródła opóźnień są dodawane do czasu, w którym ma zostać uruchomiona instancja rezerwowa. Łączny czas przełączania składa się z następujących komponentów,

Czas wykrywania awarii

Czas potrzebny na zwolnienie blokady przez system NFS na dane menedżera kolejek oraz instancję rezerwową, aby kontynuować proces uruchamiania.

Czas przesyłania

W przypadku klastra wysokiej dostępności (HA), czas potrzebny IBM i do przesłania puli IASP z systemu udostępniającego aktywną instancję do instancji rezerwowej, a w przypadku replikacji

kroniki-czas potrzebny do zaktualizowania dziennika lokalnego w stanie gotowości z danymi ze zdalnej repliki.

Czas restartowania

Czas potrzebny dla nowo aktywnej instancji menedżera kolejek na odbudowanie kolejek od ostatniego punktu kontrolnego w odtworzonym kronice i wznowienie przetwarzania komunikatów.

Uwaga:

Jeśli instancja rezerwowa, która została przejęta, jest skonfigurowana do synchronicznego replikowania do poprzednio aktywnej instancji, to uruchomienie może być opóźnione. Nowa aktywowana instancja może nie być w stanie zreplikować do jej kroniki zdalnej, jeśli kronika zdalna znajduje się na serwerze, który udostępnił poprzednio aktywną instancję, a serwer nie powiódł się.

Wartość domyślna czasu oczekiwania na odpowiedź synchroniczną wynosi jedną minutę. Istnieje możliwość skonfigurowania maksymalnego opóźnienia przed wyświetleniem limitu czasu replikacji. Alternatywnie można skonfigurować instancje rezerwowe w taki sposób, aby były uruchamiane przy użyciu replikacji asynchronicznej do instancji aktywnej, która się nie powiodła. Następnie przełącza się na replikację synchroniczną, gdy instancja, która się nie powiodła, ponownie działa w trybie gotowości. To samo rozważanie dotyczy korzystania z synchronicznych niezależnych kopii lustrzanych ASP.

Dla tych komponentów można wykonać osobne pomiary według planu bazowego, aby pomóc w ocenie ogólnego czasu przełączenia awaryjnego, a także aby uwzględnić decyzję, którą podejście konfiguracyjne ma zostać użyte. Przy podejmowaniu decyzji o najlepszej konfiguracji należy także rozważyć, w jaki sposób inne aplikacje na tym samym serwerze będą przełączać awaryjnie oraz czy istnieją procesy tworzenia kopii zapasowych lub odtwarzania po awarii, które już korzystają z puli IASP.

Czasy transferu IASP mogą zostać skrócone, dostrajanie konfiguracji klastra:

1. Profile użytkowników w systemach w klastrze powinny mieć ten sam identyfikator GUID i UID, co eliminuje konieczność zmiany identyfikatorów UID i GUID w procesie udostępniania.
2. Zminimalizuj liczbę obiektów bazy danych w systemie i podstawowych pulach dyskowych użytkownika, ponieważ należy je scalić, aby utworzyć tabelę odniesień dla grupy pul dyskowych.
3. Dodatkowe wskazówki dotyczące wydajności można znaleźć w dokumentacji technicznej (Redbook) IBM, *Implementing PowerHA for IBM i, SG24-7405*.

Konfiguracja przy użyciu podstawowych pul ASP, zapisu lustrzanego dziennika i małej konfiguracji powinna być przełączana w kolejności kilkudziesięciu sekund.

IBM i Przegląd łączących możliwości technologii klastrowej produktu IBM i z klastrami IBM MQ

Uruchamianie produktu IBM MQ w systemie IBM i i korzystanie z możliwości łączenia w klastry w programie IBM i może zapewnić bardziej wszechstronne rozwiązanie wysokiej dostępności, niż w przypadku korzystania z technologii klastrowej IBM MQ.

Aby ta możliwość była dostępna, należy skonfigurować:

1. Klastry na komputerze z produktem IBM i; patrz [“IBM iKlastry”](#) na stronie 297
2. Niezależna pula pamięci dyskowej (IASP), do której przenoszony jest menedżer kolejek; patrz [“Niezależne pule pamięci dyskowej \(IASPs\)”](#) na stronie 297
3. Grupa zasobów klastra (cluster resource group-CRG); patrz [“Grupy zasobów klastra urządzeń”](#) na stronie 297, w której definiuje się:
 - Domena odzyskiwania zasobów
 - IASP
 - Program obsługi wyjścia; patrz [“Program obsługi wyjścia CRG urządzenia”](#) na stronie 298

IBM iKlastry

Klaster produktu IBM i jest kolekcją instancji, czyli IBM i komputerów lub partycji, które są logicznie połączone.

Celem tej grupy jest umożliwienie utworzenia kopii zapasowej każdej instancji, co eliminuje pojedynczy punkt awarii i zwiększa elastyczność aplikacji i danych. W przypadku utworzenia klastra różne typy grup zasobów klastra (cluster resource group-CRG) mogą być skonfigurowane do zarządzania aplikacjami, danymi i urządzeniami w klastrze.

Więcej informacji na ten temat można znaleźć w sekcji [Tworzenie klastra](#) i [Tworzenie klastra \(Create Cluster-CRTCLU\)](#).

Niezależne pule pamięci dyskowej (IASPs)

Pula IASP to typ ASP użytkowników, który służy jako rozszerzenie pamięci masowej na poziomie jednopoziomowym. Jest to element pamięci masowej, który ze względu na swoją niezależność od pamięci systemowej może być łatwo manipulowany bez konieczności IPL systemu.

Pula IASP może być łatwo przełączana do innej instancji systemu operacyjnego lub replikowana do docelowej puli IASP w innej instancji systemu operacyjnego. W celu przełączenia puli IASP między instancjami można użyć dwóch metod:

- Pierwsza metoda wymaga wszystkich komputerów w klastrze i przełączalnej wieży dyskowej zawierającej pulę IASP, która ma być podłączona za pomocą pętli łącza o dużej szybkości (HSL).
- Druga metoda wymaga, aby instancje systemu operacyjnego były partycjami na tym samym komputerze IBM i, na którym procesory wejścia/wyjścia (IOP) mogą być przełączane między partycjami. Do zreplikowania puli IASP nie jest potrzebny żaden specjalny sprzęt. Replikacja jest wykonywana za pomocą protokołu TCP/IP w sieci.

Więcej informacji można znaleźć w opisie komendy [Konfigurowanie puli ASP urządzenia \(Configure Device ASP-CFGDEVASP\)](#).

Grupy zasobów klastra urządzeń

Istnieje kilka typów grup zasobów klastra (CRG). Więcej informacji na temat dostępnych typów grup zasobów klastra znajduje się w sekcji [Grupa zasobów klastra](#).

Ten temat koncentruje się na grupie zasobów klastra urządzeń. Grupa zasobów klastra urządzeń:

- Opisuje i zarządza zasobami urządzeń, takimi jak niezależne pule pamięci dyskowej (IASPs).
- Definiuje domenę odzyskiwania zasobów węzłów klastra
- Przypisuje urządzenie,
- Przypisuje program obsługi wyjścia, który obsługuje zdarzenia klastrowe.

Domena odzyskiwania zasobów oznacza, który węzeł klastra będzie traktowany jako węzeł podstawowy. Pozostałe węzły są uważane za kopie zapasowe. Węzły zapasowe są również uporządkowane w domenie odzyskiwania zasobów, określając, który węzeł jest pierwszą kopią zapasową, drugą kopią zapasową i tak dalej, w zależności od liczby węzłów w domenie odzyskiwania zasobów.

W przypadku awarii węzła podstawowego program obsługi wyjścia jest uruchamiany na wszystkich węzłach w domenie odzyskiwania zasobów. Program obsługi wyjścia działający przy pierwszej kopii zapasowej może następnie dokonać niezbędnych inicjacji w celu utworzenia węzła w nowym węźle podstawowym.

Więcej informacji na ten temat zawiera sekcja [Tworzenie grup zasobów klastra urządzeń](#) i komenda [Tworzenie grupy zasobów klastra \(Create Cluster Resource Group-CRTCRG\)](#).

Program obsługi wyjścia CRG urządzenia

Usługa zasobów klastra systemu operacyjnego wywołuje program obsługi wyjścia grupy zasobów klastra urządzeń, gdy zdarzenie występuje w jednym z węzłów, które definiuje domena odzyskiwania; na przykład zdarzenie przełączenia awaryjnego lub przełączenia awaryjnego.

Zdarzenie przełączenia awaryjnego występuje, gdy węzeł podstawowy klastra ulegnie awarii, a grupy zasobów klastra są przełączane ze wszystkimi zasobami, którymi zarządzają, a zdarzenie przełączenia jest wykonywane, gdy konkretna grupa zasobów klastra zostanie ręcznie przełączona z węzła podstawowego do węzła zapasowego.

W każdym przypadku program obsługi wyjścia jest w trakcie inicjowania i uruchamiania wszystkich programów, które były uruchomione w poprzednim węźle podstawowym, co przekształca pierwszy węzeł zapasowy w nowy węzeł podstawowy.

Na przykład za pomocą programu IBM MQ należy uruchomić program obsługi wyjścia, aby uruchomić podsystem IBM MQ (QMQM) oraz menedżery kolejek. Menedżery kolejek powinny być skonfigurowane do automatycznego uruchamiania programów następujących i usług, takich jak monitory wyzwalacza.

Konfiguracja przełączalnej puli IASP

Produkt IBM MQ można skonfigurować w taki sposób, aby korzystać z możliwości łączenia w klastry w produkcie IBM i. W tym celu:

1. Tworzenie klastra IBM i między systemami centrum przetwarzania danych
2. Przenieś menedżer kolejek do puli IASP.

“Przenoszenie lub usuwanie menedżera kolejek z niezależną pulą pamięci dyskowej lub z niej” na stronie 299 zawiera przykładowy kod pomocny podczas wykonywania tej operacji.

3. Należy utworzyć grupę CRG definiującą domenę odzyskiwania zasobów, pulę IASP i program obsługi wyjścia.

“Konfigurowanie grupy zasobów klastra urządzeń” na stronie 298 zawiera przykładowy kod pomocny podczas wykonywania tej operacji.

Pojęcia pokrewne

“Niezależne ASP i wysoka dostępność” na stronie 319

Niezależne ASP umożliwiają przenoszenie aplikacji i danych między serwerami. Elastyczność niezależnych pul ASP oznacza, że są one podstawą dla niektórych rozwiązań wysokiej dostępności IBM i. Rozważając użycie puli ASP lub niezależnej ASP dla kroniki menedżera kolejek, należy rozważyć inną konfigurację wysokiej dostępności w oparciu o niezależne ASP.

Konfigurowanie grupy zasobów klastra urządzeń

Przykładowy program do konfigurowania grupy zasobów klastra urządzeń (CRG).

O tym zadaniu

W poniższym przykładzie należy zauważyć, że:

- [PRIMARY SITE NAME] i [BACKUP SITE NAME] mogą być dowolnymi dwoma odrębnymi łańcuchami o długości ośmiu znaków lub mniejszej.
- [PRIMARY IP] i [BACKUP IP] to adresy IP, które mają być używane do zapisu lustrzanego.

Procedura

1. Zidentyfikuj nazwę klastra.
2. Zidentyfikuj nazwę i bibliotekę programu obsługi wyjścia grupy zasobów klastra.
3. Określ nazwę węzła podstawowego i węzłów zapasowych, które mają być zdefiniowane przez tę grupę CRG.

4. Zidentyfikuj pulę IASP, która ma być zarządzana przez tę grupę zasobów klastra, i upewnij się, że została ona utworzona w węźle podstawowym.
5. Utwórz opis urządzenia w węzłach tworzenia kopii zapasowych za pomocą komendy:

```
CRTDEVASP DEVD([IASP NAME]) RSRNAME([IASP NAME])
```

6. Dodaj adres IP przejścia do wszystkich węzłów za pomocą komendy:

```
ADDTCPIFC INTNETADR(' [TAKEOVER IP]') LIND([LINE DESC])
SUBNETMASK(' [SUBNET MASK]') AUTOSTART(*NO)
```

7. Uruchom adres IP przejścia tylko w węźle podstawowym, korzystając z komendy:

```
STRTCPIFC INTNETADR(' [TAKEOVER IP]')
```

8. Opcjonalne: Jeśli pula IASP jest przetwarzalna, wywołaj tę komendę:

```
CRTCRG CLUSTER([CLUSTER NAME]) CRG( [CRG NAME]) CRGTYPE(*DEV) EXITPGM([EXIT LIB]/[EXIT
NAME])
USRPRF([EXIT PROFILE]) RCYDMN(( [PRIMARY NODE] *PRIMARY) ([BACKUP NAME] *BACKUP))
EXITPGMFMT(EXTP0200) CFGOBJ(( [IASP NAME] *DEV *ONLINE ' [TAKEOVER IP]')
```

9. Opcjonalne: Jeśli ma być tworzona kopia lustrzana puli IASP, wywołaj tę komendę:

```
CRTCRG CLUSTER([CLUSTER NAME]) CRG([CRG NAME]) CRGTYPE(*DEV) EXITPGM([EXIT LIB]/[EXIT NAME])
USRPRF([EXIT PROFILE]) RCYDMN(( [PRIMARY NODE] *PRIMARY *LAST [PRIMARY SITE NAME] (' [PRIMARY
IP]'))
[BACKUP NAME] *BACKUP *LAST [BACKUP SITE NAME] (' [BACKUP IP]')) ) EXITPGMFMT(EXTP0200)
CFGOBJ(( [IASP NAME] *DEV *ONLINE ' [TAKEOVER IP]'))
```

IBM i *Przenoszenie lub usuwanie menedżera kolejek z niezależną pulą pamięci dyskowej lub z niej*
 Przykładowy program do przenoszenia menedżera kolejek do niezależnej puli pamięci dyskowej (IASP) i komend do usuwania menedżera kolejek z puli IASP.

O tym zadaniu

W poniższym przykładzie należy zauważyć, że:

- [NAZWA MANAGER] to nazwa menedżera kolejek.
- [IASP NAME] to nazwa puli IASP.
- [MANAGER LIBRARY] to nazwa biblioteki menedżera kolejek.
- [KATALOG MENEDŻERA] to nazwa katalogu menedżera kolejek.

Procedura

1. Zidentyfikuj węzeł podstawowy i węzły zapasowe.
2. Wykonaj następującą procedurę w węźle podstawowym:
 - a) Upewnij się, że menedżer kolejek został zakończony.
 - b) Upewnij się, że pula IASP jest w stanie wyczyszczenia przy użyciu komendy

```
VRYCFG CFGOBJ([IASP NAME]) CFGTYPE(*DEV) STATUS(*ON)
```

- c) Utwórz katalog menedżerów kolejek w puli IASP.

W katalogu głównym znajduje się katalog o nazwie IASP, który jest następujący:

```
QSH CMD('mkdir -p /[IASP_NAME]/QIBM/UserData/mqm/qmgrs/')
```

- d) Przenieś obiekty IFS menedżera do katalogu menedżerów kolejek, który został właśnie utworzony w ramach puli IASP, używając następującej komendy:

```
QSH CMD('mv /QIBM/UserData/mqm/qmgrs/[MANAGER NAME]
/[IASP NAME]/QIBM/UserData/mqm/qmgrs')
```

- e) Utwórz tymczasowy zbiór składowania o nazwie MGRLIB , korzystając z komendy:

```
CRTSAVF QGPL/MGRLIB
```

- f) Zapisz bibliotekę menedżera kolejek w pliku składowania MGRLIB za pomocą następującej komendy:

```
SAVLIB LIB([MANGER LIBRARY]) DEV(*SAVF) SAVF(QGPL/MGRLIB)
```

- g) Usuń bibliotekę menedżera kolejek za pomocą następującej komendy, a następnie zignoruj wszystkie komunikaty zapytaniem:

```
DLTLIB [MANAGER LIBRARY]
```

- h) Odtwórz bibliotekę menedżera kolejek do puli IASP za pomocą następującej komendy:

```
RSTLIB SAVLIB([MANAGER LIBRARY]) DEV(*SAVF) SAVF(QGPL/MGRLIB)
RSTASPDEV([IASP NAME])
```

- i) Usuń tymczasowy zbiór składowania za pomocą następującej komendy:

```
DLTF FILE(QGPL/MGRLIB)
```

- j) Utwórz dowiązanie symboliczne do obiektów IFS menedżera kolejek w puli IASP, korzystając z następującej komendy:

```
ADDLNK OBJ('/[IASP NAME]/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')
NEWLNK('/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')
```

- k) Dołącz do puli IASP za pomocą następującej komendy:

```
SETASPGRP [IASP NAME]
```

- l) Uruchom menedżera kolejek za pomocą komendy:

```
STRMQM [MANAGER NAME]
```

3. Wykonaj następującą procedurę w węźle zapasowym lub węzłach:

- a) Utwórz tymczasowy katalog menedżera kolejek za pomocą następującej komendy:

```
QSH CMD('mkdir -p /[IASP NAME]/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')
```

- b) Utwórz dowiązanie symboliczne do katalogu tymczasowego menedżera kolejek za pomocą następującej komendy:

```
ADDLNK OBJ('/[IASP NAME]/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')
NEWLNK('/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')
```

- c) Usuń katalog tymczasowy za pomocą następującej komendy:

```
QSH CMD('rm -r /[IASP NAME]')
```

- d) Na końcu pliku /QIBM/UserData/mqm/mqs.inidodaj następujące elementy:

```
QueueManager:
Name=[MANAGER NAME]
Prefix=/QIBM/UserData/mqm
Library=[MANAGER LIBRARY]
Directory=[MANAGER DIRECTORY]
```

4. Aby usunąć menedżera kolejek z puli IASP, należy wprowadzić następujące komendy:

- a) VRYCFG CFGOBJ ([IASP NAME]) CFGTYPE (*DEV) STATUS (*ON)
- b) SETASPGRP [NAZWA PULI IASP]
- c) ENDMQM [NAZWA MENEDŻERA]
- d) DLTMQM [NAZWA MENEDŻERA]

IBM i Konfiguracja kroniki kopii lustrzanej dla ASP w systemie IBM i

Skonfiguruj stabilny menedżer kolejek z wieloma instancjami przy użyciu synchronicznej replikacji między kronikami lustrzanymi.

W konfiguracji menedżera kolejek z kopią lustrzaną używane są kroniki utworzone w podstawowych lub niezależnych pulach pamięci dyskowej (ASP).

W systemie IBM dane menedżera kolejek są zapisywane w kronikach i w systemie plików. Dzienniki zawierają główną kopię danych menedżera kolejek. Kroniki są współużytkowane między systemami przy użyciu synchronicznej lub asynchronicznej replikacji kroniki. Aby zrestartować instancję menedżera kolejek, wymagane jest połączenie dzienników lokalnych i zdalnych. Restartowanie menedżera kolejek odczytuje rekordy kroniki z miksu dzienników lokalnych i zdalnych na serwerze oraz dane menedżera kolejek w systemie plików współużytkowanych sieci. Dane w systemie plików przyspiesza restartowanie menedżera kolejek. Punkty kontrolne są zapisywane w systemie plików, zaznaczając punkty synchronizacji między systemem plików a kronikami. Rekordy kroniki przechowywane przed punktem kontrolnym nie są wymagane w przypadku typowego restartu menedżera kolejek. Jednak dane w systemie plików mogą nie być aktualne, a rekordy kroniki po punkcie kontrolnym są używane do zakończenia restartu menedżera kolejek. Dane w dziennikach przyłączonych do instancji są aktualizowane, dzięki czemu restartowanie może zakończyć się pomyślnie.

Ale nawet rekordy kroniki mogą nie być aktualne, jeśli kronika zdalna na serwerze rezerwowym była asynchronicznie replikowana, a awaria wystąpiła przed zsynchronizowaną. W przypadku podjęcia decyzji o zrestartowaniu menedżera kolejek za pomocą kroniki zdalnej, która nie jest zsynchronizowana, instancja rezerwowego menedżera kolejek może ponownie przetworzyć komunikaty usunięte przed niepowodzeniem działania aktywnej instancji lub nie przetwarzać komunikatów, które zostały odebrane przed nieudanym niepowodzeniem.

Inną, rzadką możliwością jest to, że system plików zawiera najnowszy rekord punktu kontrolnego, a niezsynchronizowana kronika zdalna w systemie rezerwowym nie jest dostępna. W takim przypadku menedżer kolejek nie jest restartowany automatycznie. Istnieje możliwość oczekiwania na zsynchronizowanie kroniki zdalnej lub na zimne uruchomienie rezerwowego menedżera kolejek z systemu plików. Mimo że w tym przypadku system plików zawiera bardziej aktualny punkt kontrolny danych menedżera kolejek niż kronika zdalna, może nie zawierać wszystkich komunikatów, które zostały przetworzone przed nieudaną instancją. Niektóre komunikaty mogą być ponownie przetwarzane, a niektóre nie przetworzone, po zimnym restarcie, które nie jest zsynchronizowane z kronikami.

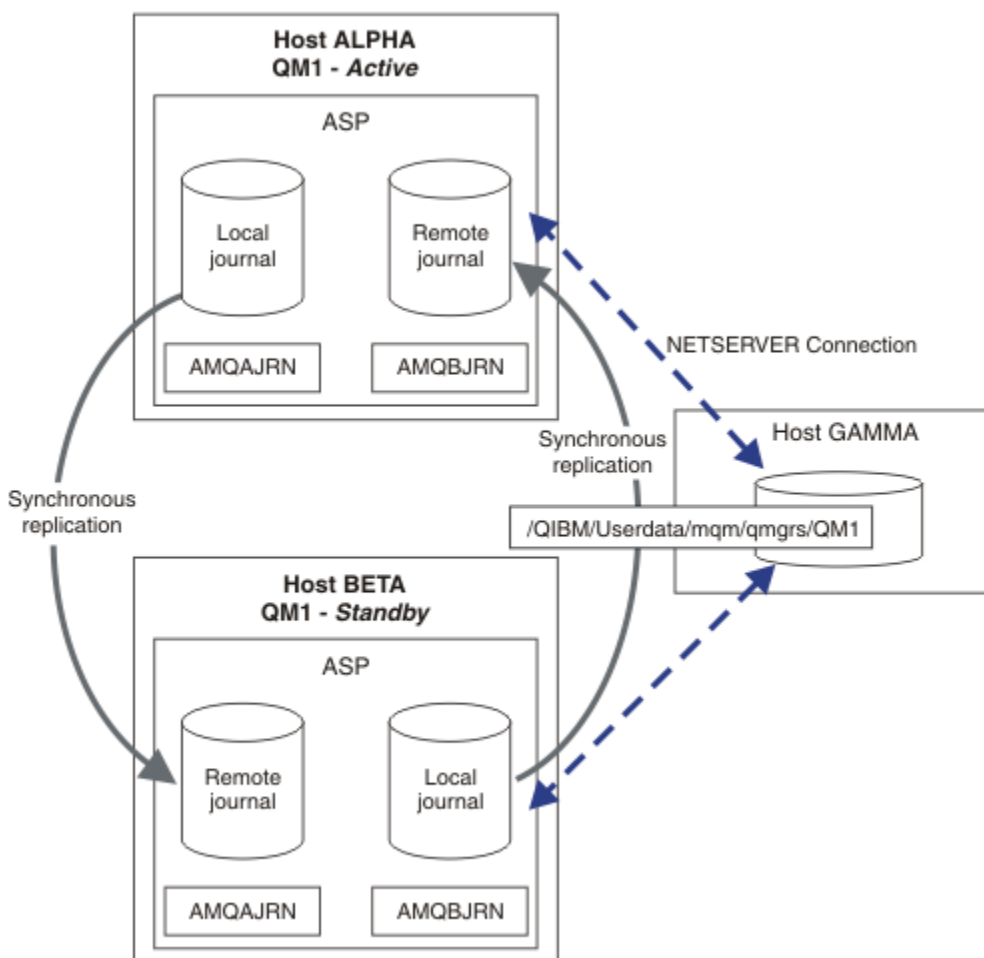
W przypadku menedżera kolejek z wieloma instancjami system plików jest również używany do kontrolowania, która instancja menedżera kolejek jest aktywna, a która jest w stanie gotowości. Aktywna instancja uzyskuje blokadę danych menedżera kolejek. Rezerwowa baza danych oczekuje na uzyskanie blokady, a po jej udostępnieniu staje się instancją aktywną. Blokada jest zwalniana przez aktywną instancję, jeśli kończy się normalnie. Blokada jest zwalniana przez system plików, jeśli system plików wykryje, że aktywna instancja nie powiodła się lub nie może uzyskać dostępu do systemu plików. System plików musi spełniać wymagania dotyczące wykrywania awarii. Patrz sekcja [Wymagania dotyczące współużytkowanych systemów plików](#).

Architektura menedżerów kolejek z wieloma instancjami w systemie IBM i zapewnia automatyczne restartowanie po awarii serwera lub menedżera kolejek. Obsługuje również odtwarzanie danych menedżera kolejek po awarii systemu plików, w którym przechowywane są dane menedżera kolejek.

W produkcie [Rysunek 36 na stronie 302](#), jeśli działanie ALPHA nie powiedzie się, można ręcznie zrestartować produkt QM1 w wersji beta, korzystając z kroniki kopii lustrzanej. Dodanie funkcji menedżera kolejek z wieloma instancjami do produktu QM1 powoduje, że instancja rezerwowa produktu QM1

automatycznie wznawia działanie BETA, jeśli aktywna instancja w systemie ALPHA nie powiedzie się. Produkt QM1 może również zostać automatycznie wznowiony, jeśli jest to serwer ALPHA, który nie udaje się, a nie tylko aktywną instancję serwera QM1. Gdy BETA stanie się hostem aktywnej instancji menedżera kolejek, instancja rezerwowa może zostać uruchomiona w systemie ALPHA.

Rysunek 36 na stronie 302 przedstawia konfigurację, która odzwierciedla dzienniki między dwiema instancjami menedżera kolejek przy użyciu serwera NetServer do przechowywania danych menedżera kolejek. Wzorzec może zostać rozwinięty w taki sposób, aby uwzględnił więcej kronik, a tym samym więcej instancji. Należy postępować zgodnie z regułami nazewnictwa dziennika, które zostały opisane w temacie "Dzienniki menedżera kolejek w systemie IBM i" na stronie 281. Obecnie liczba działających instancji menedżera kolejek jest ograniczona do dwóch, jeden jest aktywny, a jeden jest w stanie gotowości.



Rysunek 36. Tworzenie kopii lustrzanej kroniki menedżera kolejek

Kronika lokalna dla systemu QM1 na hoście ALPHA nosi nazwę AMQAJRN (lub pełniejsza, QMQM1/AMQAJRN), a w kronice BETA kronika QMQM1/AMQBJRN. Każdy dziennik lokalny replikuje do zdalnych kronik we wszystkich pozostałych instancjach menedżera kolejek. Jeśli menedżer kolejek jest skonfigurowany z dwiema instancjami, kronika lokalna jest replikowana do jednej kroniki zdalnej.

Replikacja kroniki zdalnej *SYNC lub *ASYN

Kopie lustrzane dzienników IBM i są tworzone z wykorzystaniem synchronicznej (*SYNC) lub asynchronicznej (*ASYN) kronikowanie; patrz Zarządzanie kronikami zdalnymi.

Tryb replikacji w programie Rysunek 36 na stronie 302 ma wartość *SYNC, a nie *ASYN. *ASYN jest szybszy, ale jeśli wystąpi awaria, gdy stan kroniki zdalnej ma wartość *ASYNPEND, kronika lokalna i zdalna nie są spójne. Kronika zdalna musi się nadrobić zalegając z kroniką lokalną. Jeśli

zostanie wybrana opcja *SYNC, system lokalny oczekuje na kronikę zdalną przed powrotem z wywołania, które wymaga wykonania zakończonego zapisu. Dzienniki lokalne i zdalne zwykle pozostają spójne ze jednym. Tylko wtedy, gdy operacja *SYNC trwa dłużej niż wyznaczona godzina¹ zdalne kronikowanie jest dezaktywowane, czy dzienniki są wyłączone z synchronizacji. Błąd jest rejestrowany w kolejce komunikatów kroniki i do kolejki QSYSOPR. Menedżer kolejek wykrywa ten komunikat, zapisuje błąd w dzienniku błędów menedżera kolejek i dezaktywuje zdalną replikację kroniki menedżera kolejek. Instancja aktywnego menedżera kolejek wznowia działanie bez zdalnego kronikowania w tym kronice. Gdy serwer zdalny jest ponownie dostępny, należy ręcznie reaktywować synchroniczną replikację kroniki zdalnej. Dzienniki są następnie resynchronizowane.

Problem z konfiguracją *SYNC / *ASYNC zilustrowany w programie [Rysunek 36](#) na stronie 302 jest sposobem, w jaki instancja rezerwowa menedżera kolejek na BETA przejmuje kontrolę. Gdy tylko instancja menedżera kolejek w BETA zapisuje swój pierwszy trwały komunikat, próbuje on zaktualizować kronikę zdalną w systemie ALPHA. Jeśli przyczyną kontroli przechodzących od ALPHA do BETA była awaria ALPHA, a ALPHA jest nadal wyłączona, zdalne kronikowanie do funkcji ALPHA nie powiodło się. BETA oczekuje na odpowiedź ALPHA, a następnie dezaktywuje zdalne kronikowanie i wznowia przetwarzanie komunikatów tylko przy użyciu kronikowania lokalnego. BETA musi czekać chwilę, aby wykryć, że ALPHA jest wyłączony, powodując okres nieaktywności.

Wybór między ustawieniem kronikowania zdalnego na wartość *SYNC lub *ASYNC jest wyłączony. Tabela 22 na stronie 303 podsumowuje kompromisy między użyciem kronikowania *SYNC i *ASYNC między parą menedżerów kolejek:

<i>Tabela 22. Opcje zdalnego kronikowania</i>			
Aktywny	W trybie gotowości	*SYNC	*ASYNC
*SYNC		<ol style="list-style-type: none"> 1. Spójne przetłoczenie i przetłoczenie awaryjne 2. Instancja rezerwowa nie zostanie wznowiona natychmiast po przetłoczeniu awaryjnym. 3. Zdalne kronikowanie musi być dostępne przez cały czas 4. Wydajność menedżera kolejek zależy od zdalnego kronikowania 	<ol style="list-style-type: none"> 1. Spójne przetłoczenie i przetłoczenie awaryjne 2. W przypadku, gdy serwer rezerwowy jest dostępny, zdalne kronikowanie musi być przetłoczone na *SYNC . 3. Kronikowanie zdalne musi pozostać dostępne po zrestartowaniu 4. Wydajność menedżera kolejek zależy od zdalnego kronikowania
*ASYNC		<ol style="list-style-type: none"> 1. To nie jest sensowna kombinacja 	<ol style="list-style-type: none"> 1. Niektóre komunikaty mogą zostać utracone lub zduplikowane po przetłoczeniu awaryjnym lub przetłoczeniu awaryjnym. 2. Instancja rezerwowa nie musi być dostępna przez cały czas, aby aktywna instancja była kontynuowana bez opóźnienia. 3. Wydajność nie zależy od zdalnego kronikowania

***SYNC / *ASYNC**

Aktywna instancja menedżera kolejek używa kronikowania *SYNC i gdy instancja rezerwowa zostanie uruchomiona, natychmiast próbuje użyć kronikowania *ASYNC .

¹ Wyznaczony czas wynosi 60 sekund na IBM i 5 i w zakresie od 1 do 3600 sekund, począwszy od wersji IBM i 6.1 .

1. Kronika zdalna jest transakcyjnie spójna z kroniką lokalną aktywnego menedżera kolejek. Jeśli menedżer kolejek zostanie przełączony na instancję rezerwową, może zostać wznowiony natychmiast. Instancja rezerwowa jest zwykle wznowiana bez utraty lub duplikowania komunikatów. Komunikaty są tracone lub duplikowane tylko wtedy, gdy kronikowanie zdalne nie powiodło się od ostatniego punktu kontrolnego, a poprzednio aktywny menedżer kolejek nie może zostać zrestartowany.
2. Jeśli menedżer kolejek kończy się niepowodzeniem w instancji rezerwowej, może nie być w stanie natychmiast uruchomić tego menedżera. Instancja rezerwowego menedżera kolejek jest aktywowana za pomocą kronikowania *SYNC . Przyczyna przełączenia awaryjnego może uniemożliwić zdalne kronikowanie na serwerze, na którym znajduje się instancja rezerwowa. Menedżer kolejek czeka, aż problem zostanie wykryty przed przetworami wszelkich trwałych komunikatów. Błąd jest rejestrowany w kolejce komunikatów kroniki i do kolejki QSYSOPR. Menedżer kolejek wykrywa ten komunikat, zapisuje błąd w dzienniku błędów menedżera kolejek i dezaktywuje zdalną replikację kroniki menedżera kolejek. Instancja aktywnego menedżera kolejek wznowia działanie bez zdalnego kronikowania w tym kronice. Gdy serwer zdalny jest ponownie dostępny, należy ręcznie reaktywować synchroniczną replikację kroniki zdalnej. Dzienniki są następnie resynchronizowane.
3. Serwer, do którego replikowana jest kronika zdalna, musi być zawsze dostępny w celu obsługi kroniki zdalnej. Kronika zdalna jest zwykle replikowana na tym samym serwerze, na którym znajduje się rezerwowy menedżer kolejek. Serwer może stać się niedostępny. Błąd jest rejestrowany w kolejce komunikatów kroniki i do kolejki QSYSOPR. Menedżer kolejek wykrywa ten komunikat, zapisuje błąd w dzienniku błędów menedżera kolejek i dezaktywuje zdalną replikację kroniki menedżera kolejek. Instancja aktywnego menedżera kolejek wznowia działanie bez zdalnego kronikowania w tym kronice. Gdy serwer zdalny jest ponownie dostępny, należy ręcznie reaktywować synchroniczną replikację kroniki zdalnej. Dzienniki są następnie resynchronizowane.
4. Zdalne kronikowanie jest wolniejsze od lokalnego kronikowania i jest znacznie wolniejsze, jeśli serwery są rozdzielane dużą odległością. Menedżer kolejek musi czekać na zdalne kronikowanie, co zmniejsza wydajność menedżera kolejek.

Konfiguracja *SYNC / *SYNC między parą serwerów ma niekorzyść opóźnienia w wznowianiu instancji rezerwowej po przełączeniu awaryjnym. Konfiguracja *SYNC / *ASYNCR nie ma tego problemu.

Wartość *SYNC / *SYNC nie gwarantuje utraty komunikatu po przełączeniu awaryjnym lub przełączeniu awaryjnym, o ile jest dostępna kronika zdalna. Jeśli chcesz zmniejszyć ryzyko utraty wiadomości po przełączeniu awaryjnym lub przełączeniu, masz dwie możliwości wyboru. Zatrzymaj aktywną instancję, jeśli kronika zdalna staje się nieaktywna, lub utwórz zdalne kroniki na więcej niż jednym serwerze.

***SYNC / *ASYNCR**

Aktywna instancja menedżera kolejek używa kronikowania *SYNC i gdy instancja rezerwowego menedżera kolejek zostanie uruchomiona, używa kronikowania *ASYNCR . Wkrótce po udostępnieniu serwera udostępniającego nową instancję rezerwową operator systemu musi przełączyć kronikę zdalną w aktywnej instancji na *SYNC. Gdy operator przełącza zdalne kronikowanie z *ASYNCR na *SYNC , aktywna instancja jest wstrzymana, jeśli kronika zdalna ma status *ASYNCRPEND. Aktywna instancja menedżera kolejek czeka, aż pozostałe pozycje kroniki zostaną przesłane do kroniki zdalnej. Gdy kronika zdalna zsynchronizowana jest z kroniką lokalną, nowa rezerwowa baza danych jest ponownie spójna ponownie z nową aktywną instancją. Z perspektywy zarządzania wieloinstancyjowymi menedżerami kolejek, w konfiguracji *SYNC / *ASYNCR operator systemu IBM i ma dodatkowe zadanie. Oprócz restartowania zakończonej niepowodzeniem instancji menedżera kolejek operator musi zmienić zdalne kronikowanie na wartość *SYNC .

1. Kronika zdalna jest transakcyjnie spójna z kroniką lokalną aktywnego menedżera kolejek. Jeśli aktywna instancja menedżera kolejek zostanie przełączona lub nie zostanie przełączona na instancję rezerwową, instancja rezerwowa może zostać natychmiast wznowiona. Instancja rezerwowa jest zwykle wznowiana bez utraty lub duplikowania komunikatów. Komunikaty są tracone lub duplikowane tylko wtedy, gdy kronikowanie zdalne nie powiodło się od ostatniego punktu kontrolnego, a poprzednio aktywny menedżer kolejek nie może zostać zrestartowany.

2. Operator systemu musi przełączyć kronikę zdalną z *ASYNC na *SYNC zaraz po tym, jak system, na którym znajduje się aktywna instancja, stanie się ponownie dostępny. Operator może czekać na wychwytywanie kroniki zdalnej przed przełączeniem kroniki zdalnej na *SYNC. Alternatywnie operator może natychmiast przełączyć instancję zdalną na wartość *SYNC i wymusić oczekiwanie przez aktywną instancję do momentu, aż dziennik instancji rezerwowej zostanie przechwycony. Gdy kronikowanie zdalne jest ustawione na wartość *SYNC, instancja rezerwowa jest generalnie spójna z aktywną instancją. Komunikaty są tracone lub duplikowane tylko wtedy, gdy kronikowanie zdalne nie powiodło się od ostatniego punktu kontrolnego, a poprzednio aktywny menedżer kolejek nie może zostać zrestartowany.
3. Gdy konfiguracja została odtworzona z przełączenia lub przełączenia awaryjnego, serwer, na którym znajduje się kronika zdalna, musi być dostępny cały czas.

Wybierz opcję *SYNC / *ASYNC , aby menedżer kolejek w stanie gotowości wznowił się szybko po przełączeniu awaryjnym. Należy ręcznie odtworzyć ustawienie kroniki zdalnej na wartość *SYNC dla nowej aktywnej instancji. Konfiguracja *SYNC / *ASYNC jest zgodna z normalnym wzorcem administrowania parą menedżerów kolejek z wieloma instancjami. Po awarii jednej instancji istnieje czas, po którym instancja rezerwowa zostanie zrestartowana, w trakcie której nie można przełączać się na instancję aktywną.

***ASYNC / *ASYNC**

Zarówno serwery udostępniające aktywne, jak i rezerwowe menedżery kolejek są skonfigurowane do korzystania ze zdalnego kronikowania *ASYNC .

1. Po przełączeniu lub przełączeniu awaryjnym menedżer kolejek jest kontynuowany z kroniką na nowym serwerze. Dziennik może nie być zsynchronizowany po przełączeniu lub przełączeniu awaryjnym. W rezultacie komunikaty mogą zostać utracone lub zduplikowane.
2. Aktywna instancja działa, nawet jeśli serwer udostępniający rezerwowy menedżer kolejek nie jest dostępny. Kronika lokalna jest replikowana asynchronicznie z serwerem rezerwowym, gdy jest on dostępny.
3. Zdalne kronikowanie nie ma wpływu na wydajność lokalnego menedżera kolejek.

Wybierz *ASYNC / *ASYNC , jeśli wydajność jest podstawowym wymaganiem, a po przełączeniu awaryjnym lub przełączeniu jest przygotowany do poluzowania lub zduplikowania niektórych komunikatów.

***ASYNC / *SYNC**

Nie ma powodu, aby używać tej kombinacji opcji.

Aktywacja menedżera kolejek ze zdalnej kroniki

Kroniki są replikowane synchronicznie lub asynchronicznie. Kronika zdalna może nie być aktywna lub może być wychwytywana przez kronikę lokalną. Kronika zdalna może być wychwytywana, nawet jeśli jest synchronicznie replikowana, ponieważ mogła zostać niedawno aktywowana. Reguły, które menedżer kolejek ma zastosowanie do stanu kroniki zdalnej, której używa podczas uruchamiania, są następujące.

1. Uruchomienie rezerwowe nie powiedzie się, jeśli musi się odtworzyć z kroniki zdalnej w stanie gotowości, a status kroniki to *FAILED lub *INACTPEND.
2. Gdy rozpoczyna się aktywacja rezerwowej bazy danych, zdalny status kroniki w systemie rezerwowym musi mieć wartość *ACTIVE lub *INACTIVE. Jeśli stan ma wartość *INACTIVE, aktywacja może zakończyć się niepowodzeniem, jeśli nie wszystkie dane kroniki zostały zreplikowane.

Błąd ten występuje, jeśli dane menedżera kolejek w systemie plików sieciowych mają bardziej aktualny rekord punktu kontrolnego niż ten, który znajduje się w kronice zdalnej. Niepowodzenie jest mało prawdopodobne, dopóki kronika zdalna aktywuje się dobrze w ciągu ostatnich 30 minut maksymalnego odstępu czasu między punktami kontrolnymi. Jeśli rezerwowy menedżer kolejek odczyta ostatni rekord punktu kontrolnego z systemu plików, nie uruchamia się go.

Dostępne są następujące opcje: Zaczekaj, aż kronika lokalna na serwerze aktywnym może zostać odtworzona, lub zimny uruchom rezerwowy menedżer kolejek. Jeśli zostanie wybrana opcja zimnego

startu, menedżer kolejek zostanie uruchomiony bez danych kroniki i będzie się opierał na spójności i kompletności danych menedżera kolejek w systemie plików.

Uwaga: W przypadku zimnego uruchamiania menedżera kolejek należy uruchomić ryzyko utraty lub duplikowania komunikatów po ostatnim punkcie kontrolnym. Transakcje komunikatów zostały zapisane w kronice, ale niektóre transakcje mogły nie zostać zapisane w danych menedżera kolejek w systemie plików. Po zimnym uruchomieniu menedżera kolejek zostaje uruchomiony nowy dziennik, a transakcje, które nie są zapisywane w danych menedżera kolejek w systemie plików, są tracone.

3. Aktywacja menedżera kolejek w trybie gotowości oczekuje na zmianę statusu kroniki zdalnej w stanie gotowości na zmianę z *ASYNCPEND lub *SYNCPEND na *ASYNCPEND lub *SYNCPEND. Komunikaty są okresowo zapisywane do protokołu zadania kontrolera wykonania.

Uwaga: W tym przypadku aktywacja jest oczekująca na zdalną kronikę lokalną w rezerwowym menedżerze kolejek, który jest aktywowany. Menedżer kolejek czeka również na pewien czas przed kontynuowaniem pracy bez kroniki zdalnej. Oczekuje, kiedy próbuje ona synchronicznie zapisywać do swojego dziennika zdalnego (lub dzienników), a kronika nie jest dostępna.

4. Aktywacja zostanie zatrzymana, jeśli status kroniki zmieni się na *FAILED lub *INACTPEND.

Nazwy i stany dzienników lokalnych i zdalnych, które mają być używane w aktywowaniu, są zapisywane w dzienniku błędów menedżera kolejek.

IBM i Tworzenie menedżera kolejek z wieloma instancjami przy użyciu zapisu lustrzanego dziennika i serwera NetServer w systemie IBM i

Utwórz menedżer kolejek z wieloma instancjami, który ma być uruchamiany na dwóch serwerach IBM i . Dane menedżera kolejek są przechowywane na trzecim serwerze IBM i przy użyciu serwera NetServer. W kronice menedżera kolejek tworzona jest kopia lustrzana między dwoma serwerami przy użyciu zdalnego kronikowania. Komenda **ADDQMJRN** jest używana do uproszczenia tworzenia kronik zdalnych.

Zanim rozpoczniesz

1. Zadanie wymaga trzech serwerów IBM i . Zainstaluj produkt IBM MQ na dwóch z nich, ALPHA i BETA w tym przykładzie. IBM MQ musi być co najmniej w wersji 7.0.1.1.
2. Trzeci serwer to serwer IBM i , połączony przez serwer NetServer z systemem ALPHA i BETA. Jest on używany do współużytkowania danych menedżera kolejek. Nie musi on mieć instalacji produktu IBM MQ . Warto zainstalować produkt IBM MQ na serwerze jako krok tymczasowy, aby skonfigurować katalogi i uprawnienia menedżera kolejek.
3. Upewnij się, że profil użytkownika produktu QMQM ma takie samo hasło na wszystkich trzech serwerach.
4. Zainstaluj serwer IBM i NetServer; patrz [i5/OS NetServer](#).

O tym zadaniu

Wykonaj następujące kroki, aby utworzyć konfigurację pokazanej w programie [Rysunek 37](#) na stronie 309. Dane menedżera kolejek są połączone za pomocą serwera IBM i NetServer.

- Utwórz połączenia z ALPHA i BETA do katalogu współużytkowanego w GAMMA, który ma przechowywać dane menedżera kolejek. Zadanie konfiguruje także niezbędne uprawnienia, profile użytkowników i hasła.
- Dodaj pozycje relacyjnej bazy danych (RDBE) do systemów IBM i , które mają uruchamiać instancje menedżera kolejek. Pozycje RDBE są używane do łączenia się z systemami IBM i używalnymi do zdalnego kronikowania.
- Utwórz menedżera kolejek QM1 na serwerze IBM i , ALPHA.
- Dodaj informacje sterujące menedżera kolejek dla produktu QM1 na innym serwerze IBM i , BETA.
- Utwórz kroniki zdalne na obu serwerach IBM i dla obu instancji menedżera kolejek. Każdy menedżer kolejek zapisuje w dzienniku lokalnym. Kronika lokalna jest replikowana do kroniki zdalnej. Komenda **ADDQMJRN** upraszcza dodawanie kronik i połączeń.

- Uruchom menedżer kolejek, zezwalając na instancję rezerwową.

Procedura

1. Wykonaj zadanie “Tworzenie współużytkowanego zasobu sieciowego dla danych menedżera kolejek przy użyciu serwera NetServer w systemie IBM i” na stronie 293.

W rezultacie, ALPHA i BETA mają współużytkowany zasób /QNTC/GAMMA/WMQ, który wskazuje /QIBM/UserData/mqm/qmgrs na GAMMA. Profile użytkowników QMQM i QMQMADM mają niezbędne uprawnienia, a program QMQM dopasowuje hasła we wszystkich trzech systemach.

2. Dodaj pozycje relacyjnej bazy danych (RDBE) do systemów IBM i, które mają być hostem instancji menedżera kolejek.
 - a) W systemie ALPHA utwórz połączenie z BETA.

```
ADDRDBDIRE RDB(BETA) RMTLOCNAME(BETA *IP) RMTAUTMTH(*USRIDPWD)
```

- b) W systemie BETA utwórz połączenia z ALPHA.

```
ADDRDBDIRE RDB(ALPHA) RMTLOCNAME(ALPHA *IP) RMTAUTMTH(*USRIDPWD)
```

3. Utwórz menedżera kolejek QM1 w systemie ALPHA, zapisując dane menedżera kolejek w GAMMA.

```
CRTMQM MQMNAME(QM1) UDLMSGQ(SYSTEM.DEAD.LETTER.QUEUE)
MQMDIRP(' /QNTC/GAMMA/WMQ')
```

Ścieżka, która używa serwera NetServer do tworzenia danych menedżera kolejek.

4. Uruchom w systemie ALPHA. Komenda dodaje zdalną kronikę w systemie BETA.

```
ADDQMJRN MQMNAME(QM1) RMTJRN RDB(BETA)
```

tworzy pozycje kroniki w kronice lokalnej w systemie ALPHA, gdy aktywna instancja jest w systemie ALPHA. Kronika lokalna w systemie ALPHA jest replikowana do kroniki zdalnej w systemie BETA.

5. Użyj tej komendy, aby sprawdzić dane konfiguracyjne produktu IBM MQ utworzone na potrzeby komendy ALPHA.

Informacje te są potrzebne w następnym kroku.

W tym przykładzie w systemie ALPHA tworzona jest następująca konfiguracja:

```
Name=QM1
Prefix=/QIBM/UserData/mqm
Library=QMOM1
Directory=QM1
DataPath= /QNTC/GAMMA/WMQ /QM1
```

6. Utwórz instancję menedżera kolejek QM1 w systemie BETA przy użyciu komendy. Uruchom następującą komendę na BETA, aby zmodyfikować informacje sterujące menedżera kolejek w BETA.

```
ADDQMINF MQMNAME(QM1)
PREFIX('/QIBM/UserData/mqm')
MQMDIR(QM1)
MQMLIB(QMOM1)
DATAPATH('/QNTC/GAMMA/WMQ /QM1')
```

Wskazówka: Skopiuj i wklej informacje konfiguracyjne. Sekcja menedżera kolejek jest taka sama w przypadku wartości ALPHA i BETA.

7. Uruchom na BETA. Komenda dodaje kronikę lokalną w systemie BETA i kronikę zdalną w systemie ALPHA.

```
ADDQMQRN MQMNAME(QM1) RMTJNRDB(ALPHA)
```

tworzy pozycje kroniki w kronice lokalnej na BETA, gdy aktywna instancja jest na BETA. Kronika lokalna na BETA jest replikowana do kroniki zdalnej w systemie ALPHA.

Uwaga: Alternatywnym rozwiązaniem może być skonfigurowanie zdalnego kronikowania z BETA na ALPHA z wykorzystaniem kronikowania asynchronicznego.

Ta komenda służy do konfigurowania asynchronicznego kronikowania z BETA do ALPHA, zamiast komendy w kroku [“7”](#) na stronie 307.

```
ADDQMQRN MQMNAME (QM1) RMTJNRDB (ALPHA) RMTJRNDLV (*ASYNCR)
```

Jeśli serwer lub kronikowanie w systemie ALPHA jest źródłem niepowodzenia, BETA zaczyna się bez oczekiwania na replikowanie nowych pozycji kroniki do ALPHA.

Przełącz tryb replikacji na *SYNCR, używając komendy, gdy ALPHA jest ponownie w trybie z połączeniem.

Za pomocą informacji w programie [“Konfiguracja kroniki kopii lustrzanej dla ASP w systemie IBM i”](#) na stronie 301 można zdecydować, czy dzienniki mają być synchronicznie, asynchronicznie, czy też z mieszaniną obu tych dzienników. Wartością domyślną jest replika synchroniczna z 60-sekundniowym okresem oczekiwania na odpowiedź ze zdalnej kroniki.

8. Sprawdź, czy kroniki typu ALPHA i BETA są włączone, a także czy jest to status replikacji kroniki zdalnej.
 - a) W PRZYPADKU ALFA:

```
WRKMQRN MQMNAME(QM1)
```

- b) W dniu BETA:

```
WRKMQRN MQMNAME(QM1)
```

9. Uruchom instancje menedżera kolejek na serwerze ALPHA i BETA.
 - a) Uruchom pierwszą instancję w systemie ALPHA, co powoduje, że jest ona aktywna. Włączanie przełączania się do instancji rezerwowej.

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

- b) Uruchom drugą instancję BETA, udostępniając ją jako instancję rezerwową.

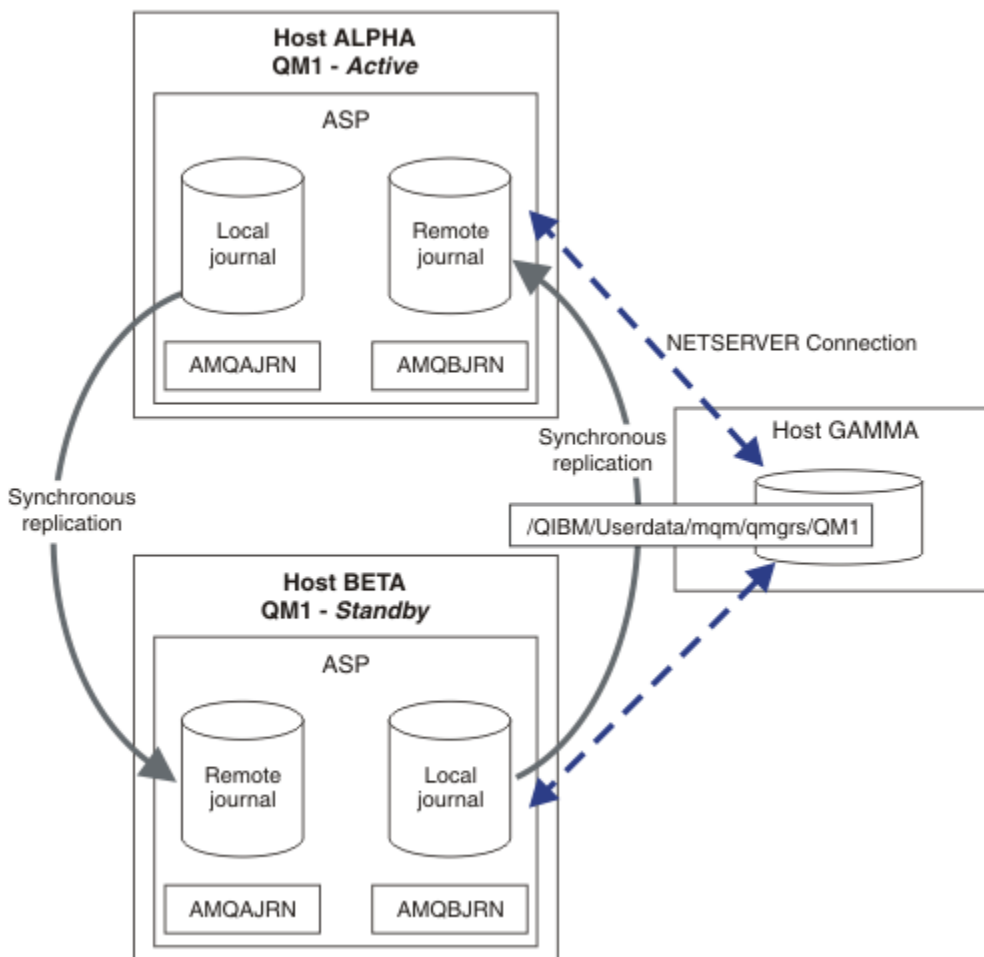
```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

Wyniki

Użyj do sprawdzenia statusu menedżera kolejek:

1. Status instancji menedżera kolejek w systemie ALPHA powinien być taki sam jak status.
2. Powinien być wyświetlany status instancji menedżera kolejek w BETA.

Przykład



Rysunek 37. Konfiguracja kroniki kopii lustrzanej

Co dalej

- Sprawdź, czy instancje aktywne i rezerwowe przełączają się automatycznie. Przykładowe programy o wysokiej dostępności można uruchomić w celu przetestowania przełącznika. Patrz sekcja [Programy przykładowe o wysokiej dostępności](#). Programy przykładowe są klientami C'. Można je uruchamiać z poziomu platformy Windows lub Unix.

1. Uruchom programy przykładowe wysokiej dostępności.
2. W przypadku ALPHA zakończ działanie menedżera kolejek żądającego przełączenia:

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) ALSWITCH(*YES)
```

3. Sprawdź, czy instancja elementu BETA jest aktywna.
4. Restartuj w systemie ALPHA

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

- Poszukaj alternatywnych konfiguracji wysokiej dostępności:

1. Użyj opcji NetServer, aby umieścić dane menedżera kolejek na serwerze Windows.

2. Zamiast zdalnego kronikowania, aby utworzyć kopię lustrzaną kroniki menedżera kolejek, należy zapisać kronikę w niezależnej puli ASP. Aby przenieść niezależną ASP z systemu ALPHA do BETA, należy użyć technologii klastrowej produktu IBM i .

IBM i Przekształcanie menedżera kolejek w pojedynczej instancji w menedżer kolejek z wieloma instancjami przy użyciu serwera NetServer i zapisu lustrzanego kroniki w systemie IBM i

Przekształć menedżer kolejek w pojedynczej instancji w menedżer kolejek z wieloma instancjami. Przenieś dane menedżera kolejek do współużytkowanego zasobu sieciowego połączonego przez serwer NetServer. Utwórz kopię lustrzaną kroniki menedżera kolejek na drugim serwerze IBM i , używając zdalnego kronikowania.

Zanim rozpoczniesz

1. Zadanie wymaga trzech serwerów IBM i . Istniejąca instalacja produktu IBM MQ na serwerze ALPHA w tym przykładzie musi być co najmniej na poziomie IBM WebSphere MQ 7.0.1 Fix Pack 1. Komenda ALPHA umożliwia uruchomienie menedżera kolejek o nazwie QM1 w tym przykładzie.
2. Zainstaluj produkt IBM MQ na drugim serwerze IBM i , w tym przykładzie BETA.
3. Trzeci serwer to serwer IBM i , połączony przez serwer NetServer z systemem ALPHA i BETA. Jest on używany do współużytkowania danych menedżera kolejek. Nie musi on mieć instalacji produktu IBM MQ . Warto zainstalować produkt IBM MQ na serwerze jako krok tymczasowy, aby skonfigurować katalogi i uprawnienia menedżera kolejek.
4. Upewnij się, że profil użytkownika produktu QMQM ma takie samo hasło na wszystkich trzech serwerach.
5. Zainstaluj serwer IBM i NetServer; patrz [i5/OS NetServer](#).

O tym zadaniu

Wykonaj następujące kroki, aby przekształcić pojedynczy menedżer kolejek instancji w menedżer kolejek z wieloma instancjami, który jest wyświetlany w produkcie [Rysunek 38 na stronie 314](#). Pojedynczy menedżer kolejek instancji jest usuwany w zadaniu, a następnie ponownie tworzony, przechowując dane menedżera kolejek w zasobie współużytkowanym w sieci połączonym przez serwer NetServer. Ta procedura jest bardziej niezawodna niż przenoszenie katalogów i plików menedżera kolejek do współużytkowanego zasobu sieciowego za pomocą komendy **CPY** .

- Utwórz połączenia z ALPHA i BETA do katalogu współużytkowanego w GAMMA, który ma przechowywać dane menedżera kolejek. Zadanie konfiguruje także niezbędne uprawnienia, profile użytkowników i hasła.
- Dodaj pozycje relacyjnej bazy danych (RDBE) do systemów IBM i , które mają uruchamiać instancje menedżera kolejek. Pozycje RDBE są używane do łączenia się z systemami IBM i używalnymi do zdalnego kronikowania.
- Zapisz dzienniki i definicje menedżera kolejek, zatrzymaj menedżer kolejek i usuń go.
- Utwórz ponownie menedżer kolejek, przechowując dane menedżera kolejek w zasobie współużytkowanym w GAMMA.
- Dodaj drugą instancję menedżera kolejek do drugiego serwera.
- Utwórz kroniki zdalne na obu serwerach IBM i dla obu instancji menedżera kolejek. Każdy menedżer kolejek zapisuje w dzienniku lokalnym. Kronika lokalna jest replikowana do kroniki zdalnej. Komenda **ADDQMJRN** upraszcza dodawanie kronik i połączeń.
- Uruchom menedżer kolejek, zezwalając na instancję rezerwową.

Uwaga:

W kroku "4" na [stronie 311](#) czynności można usunąć menedżer kolejek z pojedynczą instancją, QM1. Usunięcie menedżera kolejek powoduje usunięcie wszystkich trwałych komunikatów w kolejkach. Z tego powodu przed przekształceniem menedżera kolejek należy zakończyć przetwarzanie wszystkich komunikatów przechowywanych przez menedżera kolejek. Jeśli przetwarzanie wszystkich komunikatów

nie jest możliwe, należy utworzyć kopię zapasową biblioteki menedżera kolejek przed krokiem “4” na stronie 311. Odtwórz bibliotekę menedżera kolejek po kroku “5” na stronie 311.

Uwaga:

W kroku “5” na stronie 311 czynności, należy ponownie utworzyć QM1. Mimo że menedżer kolejek ma taką samą nazwę, ma on inny identyfikator menedżera kolejek. W klastrze menedżerów kolejek używany jest identyfikator menedżera kolejek. Aby usunąć i ponownie utworzyć menedżer kolejek w klastrze, należy najpierw usunąć menedżer kolejek z klastra. Patrz sekcja Usuwanie menedżera kolejek z klastra: alternatywna metoda lub Usuwanie menedżera kolejek z klastra. Po ponownym utworzeniu menedżera kolejek dodaj go do klastra. Mimo że ma ona taką samą nazwę jak poprzednio, to wydaje się, że jest to nowy menedżer kolejek dla innych menedżerów kolejek w klastrze.

Procedura

1. Wykonaj zadanie “Tworzenie współużytkowanego zasobu sieciowego dla danych menedżera kolejek przy użyciu serwera NetServer w systemie IBM i” na stronie 293.

W rezultacie, ALPHA i BETA mają współużytkowany zasób /QNTC/GAMMA/WMQ, który wskazuje /QIBM/UserData/mqm/qmgrs na GAMMA. Profile użytkowników QMQM i QMQMADM mają niezbędne uprawnienia, a program QMQM dopasowuje hasła we wszystkich trzech systemach.

2. Dodaj pozycje relacyjnej bazy danych (RDBE) do systemów IBM i , które mają być hostem instancji menedżera kolejek.

- a) W systemie ALPHA utwórz połączenie z BETA.

```
ADDRDBDIRE RDB(BETA) RMTLOCNAME(BETA *IP) RMTAUTMTH(*USRIDPWD)
```

- b) W systemie BETA utwórz połączenia z ALPHA.

```
ADDRDBDIRE RDB(ALPHA) RMTLOCNAME(ALPHA *IP) RMTAUTMTH(*USRIDPWD)
```

3. Utwórz skrypty, które ponownie tworzą obiekty menedżera kolejek.

```
QSAVEQMR LCLQMRNAM(QM1) FILENAME(' *CURLIB/QMQSC(QM1) ')  
OUTPUT(*REPLACE) MAKEAUTH(*YES) AUTHFN(' *CURLIB/QMAUT(QM1) ')
```

4. Zatrzymaj menedżer kolejek i usuń go.

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) ENDCCTJOB(*YES) RCDMQMIMG(*YES) TIMEOUT(15)  
DLTMQM MQMNAME(QM1)
```

5. Utwórz menedżera kolejek QM1 w systemie ALPHA, zapisując dane menedżera kolejek w GAMMA.

```
CRTMQM MQMNAME(QM1) UDLMSGQ(SYSTEM.DEAD.LETTER.QUEUE)  
MQMDIRP(' /QNTC/GAMMA/WMQ ')
```

Ścieżka, która używa serwera NetServer do tworzenia danych menedżera kolejek.

6. Ponownie utwórz obiekty menedżera kolejek dla QM1 z zapisanych definicji.

```
STRMQMQSC SRCMBR(QM1) SRCFILE(*CURLIB/QMQSC) MQMNAME(QM1)
```

7. Zastosuj autoryzacje z zapisanych informacji.

- a) Skompiluj zapisany program autoryzacji.

```
CRTCLPGM PGM(*CURLIB/QM1) SRCFILE(*CURLIB/QMAUT)  
SRCMBR(QM1) REPLACE(*YES)
```

b) Uruchom program, aby zastosować autoryzację.

```
CALL PGM(*CURLIB/QM1)
```

c) Odśwież informacje o zabezpieczeniach dla produktu QM1.

```
RFRMQMAUT MQMNAME(QM1)
```

8. Uruchom w systemie ALPHA. Komenda dodaje zdalną kronikę w systemie BETA.

```
ADDMQMJRN MQMNAME(QM1) RMTJRNRDB(BETA)
```

tworzy pozycje kroniki w kronice lokalnej w systemie ALPHA, gdy aktywna instancja jest w systemie ALPHA. Kronika lokalna w systemie ALPHA jest replikowana do kroniki zdalnej w systemie BETA.

9. Użyj tej komendy, aby sprawdzić dane konfiguracyjne produktu IBM MQ utworzone na potrzeby komendy ALPHA.

Informacje te są potrzebne w następnym kroku.

W tym przykładzie w systemie ALPHA tworzona jest następująca konfiguracja:

```
Name=QM1
Prefix=/QIBM/UserData/mqm
Library=QMQM1
Directory=QM1
DataPath= /QNTC/GAMMA/WMQ /QM1
```

10. Utwórz instancję menedżera kolejek QM1 w systemie BETA przy użyciu komendy. Uruchom następującą komendę na BETA, aby zmodyfikować informacje sterujące menedżera kolejek w BETA.

```
ADDMQMINF MQMNAME(QM1)
PREFIX('/QIBM/UserData/mqm')
MQMDIR(QM1)
MQMLIB(QMQM1)
DATAPATH('/QNTC/GAMMA/WMQ /QM1')
```

Wskazówka: Skopiuj i wklej informacje konfiguracyjne. Sekcja menedżera kolejek jest taka sama w przypadku wartości ALPHA i BETA.

11. Uruchom na BETA. Komenda dodaje kronikę lokalną w systemie BETA i kronikę zdalną w systemie ALPHA.

```
ADDMQMJRN MQMNAME(QM1) RMTJRNRDB(ALPHA)
```

tworzy pozycje kroniki w kronice lokalnej na BETA, gdy aktywna instancja jest na BETA. Kronika lokalna na BETA jest replikowana do kroniki zdalnej w systemie ALPHA.

Uwaga: Alternatywnym rozwiązaniem może być skonfigurowanie zdalnego kronikowania z BETA na ALPHA z wykorzystaniem kronikowania asynchronicznego.

Ta komenda służy do konfigurowania asynchronicznego kronikowania z BETA do ALPHA, zamiast komendy w kroku [“7”](#) na stronie 307.

```
ADDMQMJRN MQMNAME(QM1) RMTJRNRDB(ALPHA) RMTJRNDLV(*ASYNC)
```

Jeśli serwer lub kronikowanie w systemie ALPHA jest źródłem niepowodzenia, BETA zaczyna się bez oczekiwania na replikowanie nowych pozycji kroniki do ALPHA.

Przełącz tryb replikacji na *SYNC, używając komendy, gdy ALPHA jest ponownie w trybie z połączeniem.

Za pomocą informacji w programie “Konfiguracja kroniki kopii lustrzanej dla ASP w systemie IBM i” na stronie 301 można zdecydować, czy dzienniki mają być synchronicznie, asynchronicznie, czy też z mieszaniną obu tych dzienników. Wartością domyślną jest replika synchroniczna z 60-sekundyniowym okresem oczekiwania na odpowiedź ze zdalnej kroniki.

12. Sprawdź, czy kroniki typu ALPHA i BETA są włączone, a także czy jest to status replikacji kroniki zdalnej.

a) W PRZYPADKU ALFA:

```
WRKMQMJRN MQMNAME(QM1)
```

b) W dniu BETA:

```
WRKMQMJRN MQMNAME(QM1)
```

13. Uruchom instancje menedżera kolejek na serwerze ALPHA i BETA.

a) Uruchom pierwszą instancję w systemie ALPHA, co powoduje, że jest ona aktywna. Włączanie przełączania się do instancji rezerwowej.

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

b) Uruchom drugą instancję BETA, udostępniając ją jako instancję rezerwową.

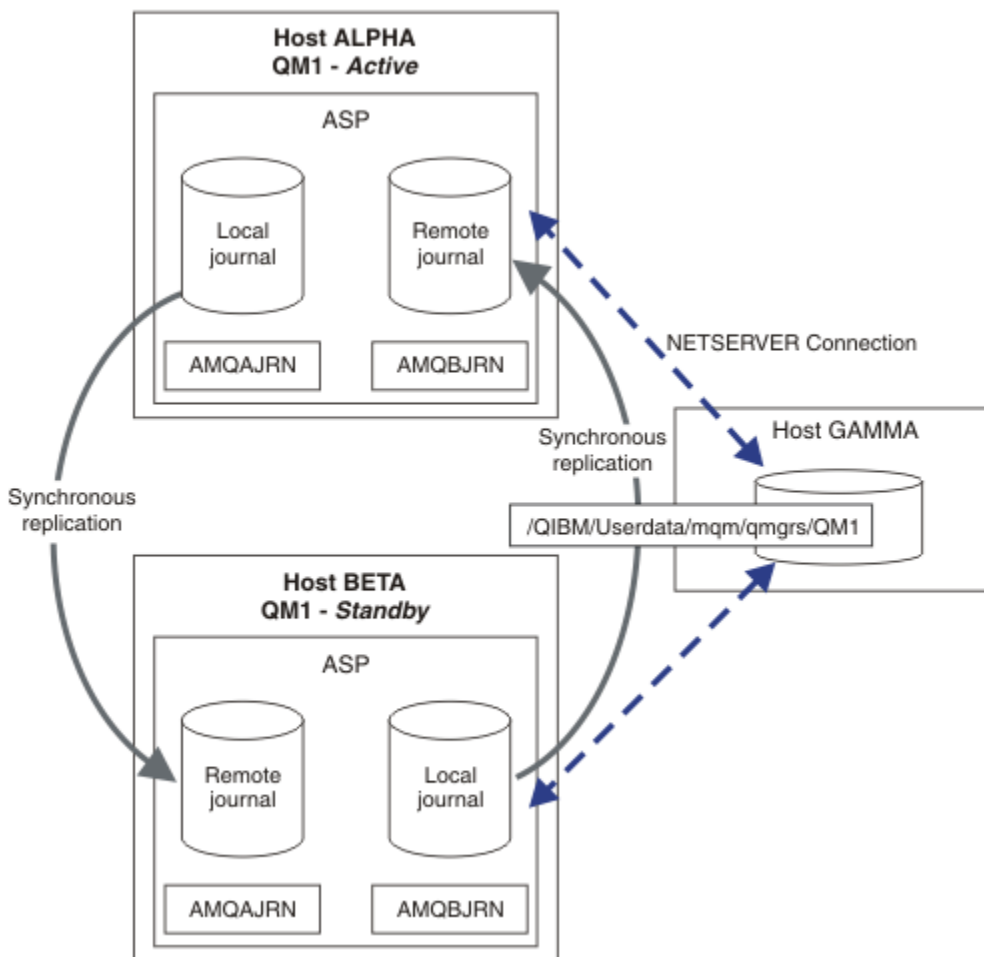
```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

Wyniki

Użyj do sprawdzenia statusu menedżera kolejek:

1. Status instancji menedżera kolejek w systemie ALPHA powinien być taki sam jak status.
2. Powinien być wyświetlany status instancji menedżera kolejek w BETA.

Przykład



Rysunek 38. Konfiguracja kroniki kopii lustrzanej

Co dalej

- Sprawdź, czy instancje aktywne i rezerwowe przełączają się automatycznie. Przykładowe programy o wysokiej dostępności można uruchomić w celu przetestowania przełącznika. Patrz sekcja [Programy przykładowe o wysokiej dostępności](#). Programy przykładowe są klientami C'. Można je uruchamiać z poziomu platformy Windows lub Unix.

1. Uruchom programy przykładowe wysokiej dostępności.
2. W przypadku ALPHA zakończ działanie menedżera kolejek żądającego przełączenia:

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) ALSWITCH(*YES)
```

3. Sprawdź, czy instancja elementu BETA jest aktywna.
4. Restartuj w systemie ALPHA

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

- Poszukaj alternatywnych konfiguracji wysokiej dostępności:

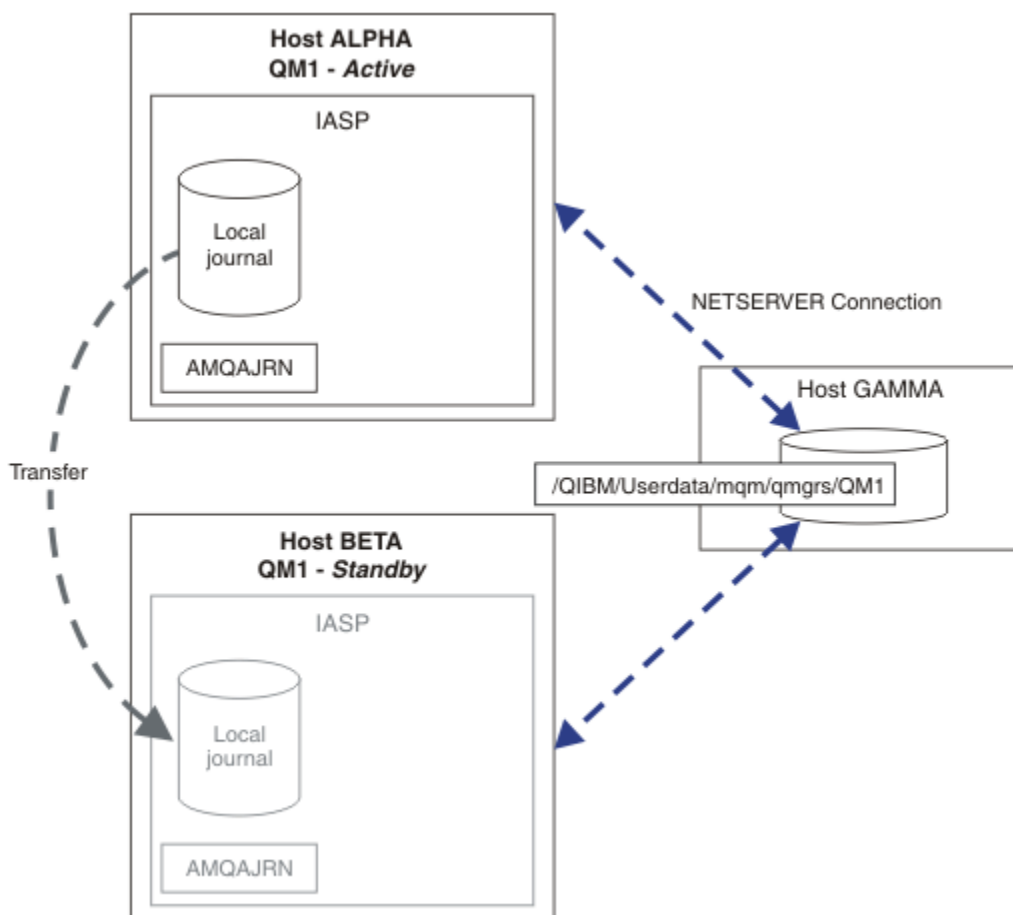
1. Użyj opcji NetServer, aby umieścić dane menedżera kolejek na serwerze Windows.

- Zamiast zdalnego kronikowania, aby utworzyć kopię lustrzaną kroniki menedżera kolejek, należy zapisać kronikę w niezależnej puli ASP. Aby przenieść niezależną ASP z systemu ALPHA do BETA, należy użyć technologii klastrowej produktu IBM i .

IBM i Konfiguracja kroniki komutowanej niezależnej ASP w systemie IBM i

Nie ma potrzeby replikowania kroniki niezależnej ASP w celu utworzenia konfiguracji menedżera kolejek z wieloma instancjami. Konieczne jest zautomatyzowanie sposobu przesyłania niezależnej ASP z aktywnego menedżera kolejek do rezerwowego menedżera kolejek. Istnieją alternatywne rozwiązania wysokiej dostępności, które można zastosować przy użyciu niezależnej ASP, z których nie wszystkie wymagają użycia menedżera kolejek z wieloma instancjami.

W przypadku korzystania z niezależnej ASP nie ma potrzeby tworzenia kopii lustrzanej kroniki menedżera kolejek. Jeśli zainstalowano zarządzanie klastrami, a serwery udostępniające instancje menedżera kolejek znajdują się w tej samej grupie zasobów klastra, to kronika menedżera kolejek może zostać automatycznie przeniesiona na inny serwer w niewielkiej odległości od aktywnego serwera, jeśli host, na którym działa ta aktywna instancja, nie powiedzie się. Kronikę można również przesłać ręcznie, jako część planowanego przetąacza, lub można napisać procedurę komendy, aby programowo przesłać niezależną ASP.



Rysunek 39. Przesyłanie kroniki menedżera kolejek za pomocą niezależnej ASP

W przypadku operacji menedżera kolejek z wieloma instancjami dane menedżera kolejek muszą być przechowywane w współużytkowanym systemie plików. System plików może być udostępniany na wielu różnych platformach. Nie można przechowywać danych menedżera kolejek z wieloma instancjami w puli ASP lub niezależnej puli ASP.

Współużytkowany system plików wykonuje dwie role w konfiguracji: te same dane menedżera kolejek są współużytkowane przez wszystkie instancje menedżera kolejek. System plików musi mieć

stabilny protokół blokujący, który zapewnia, że tylko jedna instancja menedżera kolejek ma dostęp do danych menedżera kolejek po jego uruchomieniu. Jeśli menedżer kolejek nie powiedzie się lub komunikacja z serwerem plików jest przerwa, system plików musi zwolnić blokadę danych menedżera kolejek przechowywanej przez aktywną instancję, która nie komunikuje się już z systemem plików. Instancja rezerwowego menedżera kolejek może wtedy uzyskać dostęp do odczytu/zapisu danych menedżera kolejek. Protokół systemu plików musi być zgodny z zestawem reguł, aby poprawnie działać z menedżerami kolejek z wieloma instancjami; patrz [“Komponenty rozwiązania wysokiej dostępności w systemie IBM i”](#) na stronie 292.

Mechanizm blokujący przekształca do postaci szeregowej komendę uruchomienia menedżera kolejek i określa, która instancja menedżera kolejek jest aktywna. Gdy menedżer kolejek stanie się aktywny, odbudowuje on swoje kolejki z kroniki lokalnej, którą użytkownik lub klaster wysokiej dostępności został przeniesiony na serwer rezerwowy. Klienci z możliwością ponownego połączenia, które oczekują na ponowne nawiązanie połączenia z tym samym menedżerem kolejek, zostaną ponownie połączone, a wszystkie transakcje inflight są wycofane. Aplikacje, które są skonfigurowane do uruchamiania jako usługi menedżera kolejek, są uruchamiane.

Należy upewnić się, że kronika lokalna z instancji menedżera kolejek aktywnych w niezależnej puli ASP została przeniesiona na serwer udostępniający nowo aktywowaną rezerwową instancję menedżera kolejek, konfigurując menedżer zasobów klastra lub ręcznie przesyłając niezależną pulę ASP. Korzystanie z niezależnych pul ASP nie wyklucza konfigurowania zdalnych kronik i zapisu lustrzanego, jeśli użytkownik zdecyduje się na użycie niezależnej ASP do tworzenia kopii zapasowych i usuwania skutków awarii oraz do korzystania ze zdalnego zapisu lustrzanego dla konfiguracji menedżera kolejek z wieloma instancjami.

Jeśli wybrano opcję korzystania z niezależnej ASP, można rozważyć zastosowanie alternatywnych konfiguracji o wysokiej dostępności. Tło dla tych rozwiązań zostało opisane w sekcji [“Niezależne ASP i wysoka dostępność”](#) na stronie 319.

1. Zamiast używać menedżerów kolejek z wieloma instancjami, należy zainstalować i skonfigurować menedżer kolejek z pojedynczą instancją w całości na niezależnej ASP, a następnie za pomocą usług wysokiej dostępności produktu IBM i zakończyć działanie menedżera kolejek. Prawdopodobnie konieczne jest rozszerzenie rozwiązania za pomocą monitora menedżera kolejek w celu wykrycia, czy menedżer kolejek nie uległ niezależnie od serwera. Jest to podstawa rozwiązania podanego w sekcji *Supportpac MC41: Configuring IBM MQ for iSeries for High Availability*.
2. Należy użyć niezależnych ASP i międzyośrodkowe zapisu lustrzanego (XSM), aby utworzyć kopię lustrzaną niezależnej ASP zamiast przełączać niezależną ASP na magistrali lokalnej. Rozszerza zakres geograficzny rozwiązania niezależnej ASP do zakresu czasu, jaki pozwala na zapisywanie rekordów dziennika w dużej odległości.

IBM i *Tworzenie menedżera kolejek z wieloma instancjami przy użyciu niezależnej ASP i serwera NetServer w systemie IBM i*

Utwórz menedżer kolejek z wieloma instancjami, który ma być uruchamiany na dwóch serwerach IBM i . Dane menedżera kolejek są przechowywane na serwerze IBM i przy użyciu serwera NetServer. Kronika menedżera kolejek jest przechowywana w niezależnej puli ASP. Aby przestać niezależną ASP zawierającą kronikę menedżera kolejek do drugiego serwera IBM i , należy użyć technologii klastrowej produktu IBM i lub procedury ręcznej.

Zanim rozpoczniesz

1. Zadanie wymaga trzech serwerów IBM i . Zainstaluj produkt IBM MQ na dwóch z nich, ALPHA i BETA w tym przykładzie. IBM MQ musi być co najmniej w wersji 7.0.1.1.
2. Trzeci serwer to serwer IBM i , połączony przez serwer NetServer z systemem ALPHA i BETA. Jest on używany do współużytkowania danych menedżera kolejek. Nie musi on mieć instalacji produktu IBM MQ . Warto zainstalować produkt IBM MQ na serwerze jako krok tymczasowy, aby skonfigurować katalogi i uprawnienia menedżera kolejek.
3. Upewnij się, że profil użytkownika produktu QMQM ma takie samo hasło na wszystkich trzech serwerach.
4. Zainstaluj serwer IBM i NetServer; patrz [i5/OS NetServer](#).

5. Utwórz procedury przesyłania niezależnej ASP z menedżera kolejek, który uległ awarii, do rezerwowej, która się przejęła. Niektóre z technik w programie *SupportPac MC41: Configuring IBM MQ for iSeries for High Availability* mogą być pomocne przy projektowaniu procedur przesyłania niezależnych ASP.

O tym zadaniu

Wykonaj następujące kroki, aby utworzyć konfigurację pokazanej w programie [Rysunek 40 na stronie 318](#). Dane menedżera kolejek są połączone za pomocą serwera IBM i NetServer.

- Utwórz połączenia z ALPHA i BETA do katalogu współużytkowanego w GAMMA, który ma przechowywać dane menedżera kolejek. Zadanie konfiguruje także niezbędne uprawnienia, profile użytkowników i hasła.
- Utwórz menedżera kolejek QM1 na serwerze IBM i , ALPHA.
- Dodaj informacje sterujące menedżera kolejek dla produktu QM1 na innym serwerze IBM i , BETA.
- Uruchom menedżer kolejek, zezwalając na instancję rezerwową.

Procedura

1. Wykonaj zadanie [“Tworzenie współużytkowanego zasobu sieciowego dla danych menedżera kolejek przy użyciu serwera NetServer w systemie IBM i”](#) na stronie 293.

W rezultacie, ALPHA i BETA mają współużytkowany zasób /QNTC/GAMMA/WMQ, który wskazuje /QIBM/UserData/mqm/qmgrs na GAMMA. Profile użytkowników QMQM i QMQMADM mają niezbędne uprawnienia, a program QMQM dopasowuje hasła we wszystkich trzech systemach.

2. Utwórz menedżera kolejek QM1 w systemie ALPHA, zapisując dane menedżera kolejek w GAMMA.

```
CRTMQM MQMNAME(QM1) UDLMSGQ(SYSTEM.DEAD.LETTER.QUEUE)
MQMDIRP(' /QNTC/GAMMA/WMQ ')
```

Ścieżka, która używa serwera NetServer do tworzenia danych menedżera kolejek.

3. Użyj tej komendy, aby sprawdzić dane konfiguracyjne produktu IBM MQ utworzone na potrzeby komendy ALPHA.

Informacje te są potrzebne w następnym kroku.

W tym przykładzie w systemie ALPHA tworzona jest następująca konfiguracja:

```
Name=QM1
Prefix=/QIBM/UserData/mqm
Library=QMOM1
Directory=QM1
DataPath= /QNTC/GAMMA/WMQ /QM1
```

4. Utwórz instancję menedżera kolejek QM1 w systemie BETA przy użyciu komendy. Uruchom następującą komendę na BETA, aby zmodyfikować informacje sterujące menedżera kolejek w BETA.

```
ADDQMINF MQMNAME(QM1)
PREFIX('/QIBM/UserData/mqm')
MQMDIR(QM1)
MQMLIB(QMOM1)
DATAPATH(' /QNTC/GAMMA/WMQ /QM1 ')
```

Wskazówka: Skopiuj i wklej informacje konfiguracyjne. Sekcja menedżera kolejek jest taka sama w przypadku wartości ALPHA i BETA.

5. Uruchom instancję menedżera kolejek na serwerze ALPHA i BETA.
 - a) Uruchom pierwszą instancję w systemie ALPHA, co powoduje, że jest ona aktywna. Włączanie przełączania się do instancji rezerwowej.

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

b) Uruchom drugą instancję BETA, udostępniając ją jako instancję rezerwową.

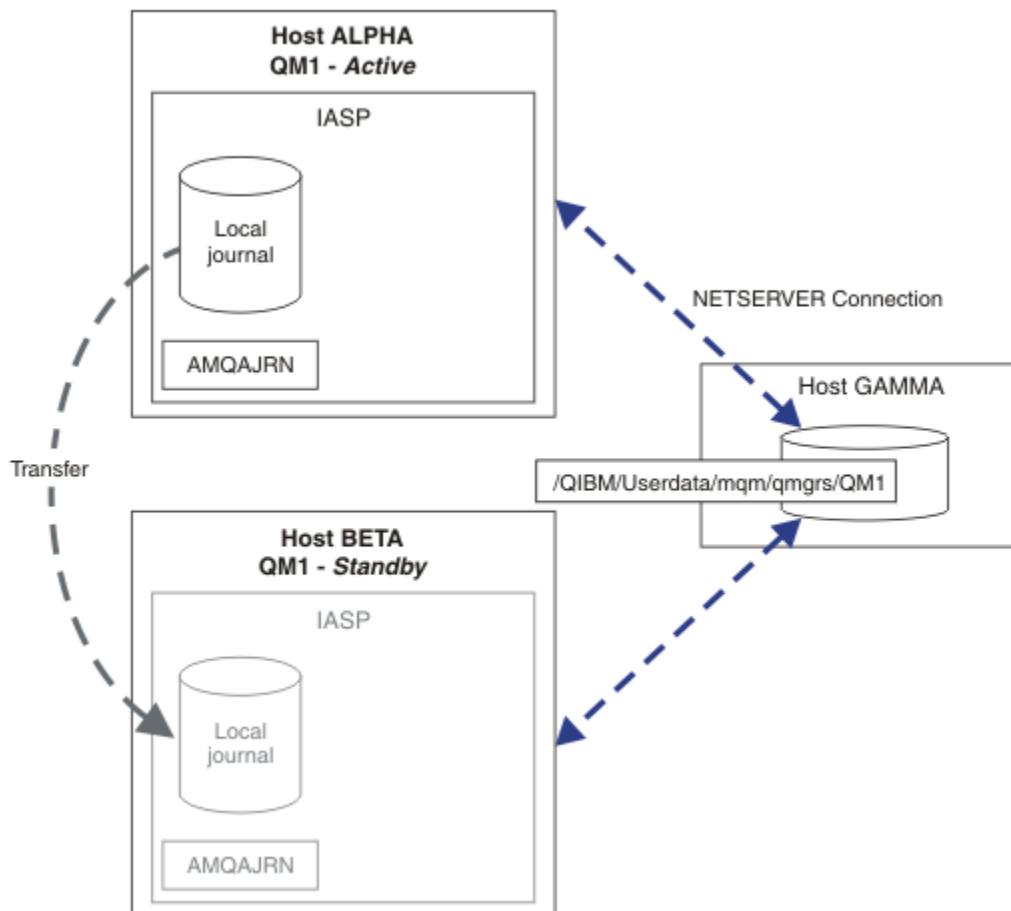
```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

Wyniki

Użyj do sprawdzenia statusu menedżera kolejek:

1. Status instancji menedżera kolejek w systemie ALPHA powinien być taki sam jak status.
2. Powinien być wyświetlany status instancji menedżera kolejek w BETA.

Przykład



Rysunek 40. Przesyłanie kroniki menedżera kolejek za pomocą niezależnej ASP

Co dalej

- Sprawdź, czy instancje aktywne i rezerwowe przełączają się automatycznie. Przykładowe programy o wysokiej dostępności można uruchomić w celu przetestowania przełącznika. Patrz sekcja [Programy przykładowe o wysokiej dostępności](#). Programy przykładowe są klientami C'. Można je uruchamiać z poziomu platformy Windows lub Unix.
1. Uruchom programy przykładowe wysokiej dostępności.
 2. W przypadku ALPHA zakończ działanie menedżera kolejek żądającego przełączenia:

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) ALSWITCH(*YES)
```

3. Sprawdź, czy instancja elementu BETA jest aktywna.
4. Restartuj w systemie ALPHA

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

- Poszukaj alternatywnych konfiguracji wysokiej dostępności:
 1. Użyj opcji NetServer , aby umieścić dane menedżera kolejek na serwerze IBM i .
 2. Zamiast używać niezależnej ASP do przesyłania kroniki menedżera kolejek do serwera rezerwowego, należy użyć zdalnego kronikowania, aby utworzyć kopię lustrzaną kroniki na serwerze rezerwowym.

IBM i *Niezależne ASP i wysoka dostępność*

Niezależne ASP umożliwiają przenoszenie aplikacji i danych między serwerami. Elastyczność niezależnych pul ASP oznacza, że są one podstawą dla niektórych rozwiązań wysokiej dostępności IBM i . Rozważając użycie puli ASP lub niezależnej ASP dla kroniki menedżera kolejek, należy rozważyć inną konfigurację wysokiej dostępności w oparciu o niezależne ASP.

Pule pamięci dyskowej (ASP) są blokiem budynku architektury IBM i . Jednostki dyskowe są zgrupowane w jedną pulę ASP. Umieszczając obiekty w różnych pulach ASP, można chronić dane w jednej puli ASP przed awariami dysków w innej puli ASP.

Każdy serwer IBM i ma co najmniej jedną *podstawową* pulę ASP, znaną jako systemowa ASP. Jest ona oznaczona jako ASP1, a czasami nazywana *SYSBAS. Istnieje możliwość skonfigurowania do 31 dodatkowych podstawowych pul ASP użytkowników *użytkownik* , które nie mogą być rozróżniane z systemowej puli ASP z punktu widzenia aplikacji, ponieważ współużytkują tę samą przestrzeń nazw. Dzięki zastosowaniu wielu podstawowych pul ASP do dystrybucji aplikacji na wielu dyskach można zwiększyć wydajność i skrócić czas odtwarzania. Korzystanie z wielu podstawowych pul ASP może również zapewnić pewien stopień odseparowania od awarii dysku, ale nie poprawia niezawodności w całości.

Niezależne pule ASP są specjalnym typem ASP. Są one często nazywane niezależnymi pulami dyskowymi. Niezależne pule dyskowe są kluczowym komponentem wysokiej dostępności produktu IBM i . Można przechowywać dane i aplikacje, które uważają się za niezależne od bieżącego systemu, do którego są podłączone w niezależnych jednostkach pamięci dyskowej. Można skonfigurować przełączalne lub nieprzełączalne niezależne ASP. Z perspektywy dostępności użytkownik jest zwykle zainteresowany przełączalnymi niezależnymi ASP, które mogą być automatycznie przesyłane z serwera na serwer. W rezultacie można przenieść aplikacje i dane na niezależną ASP z serwera na serwer.

W przeciwieństwie do podstawowych ASP użytkowników, niezależne pule ASP nie współużytkują tej samej przestrzeni nazw, co systemowa pula ASP. Aplikacje, które pracują z ASP użytkowników, wymagają zmian w pracy z niezależną ASP. Należy sprawdzić, czy używane oprogramowanie oraz oprogramowanie innych firm działa w niezależnym środowisku ASP.

Gdy niezależna ASP jest przyłączona do innego serwera, przestrzeń nazw niezależnej ASP musi być połączona z przestrzenią nazw systemowej ASP. Ten proces jest nazywany *udostępnionym* niezależną ASP. Istnieje możliwość udostępnienia niezależnej ASP bez IPLing serwera. Obsługa technologii klastrowej jest wymagana do automatycznego przesyłania niezależnych pul ASP z jednego serwera do drugiego.

Budowanie niezawodnych rozwiązań z niezależnymi ASP

Kronikowanie niezależnej ASP, a nie kronikowanie w puli ASP i korzystanie z replikacji kroniki, stanowi alternatywne sposoby udostępniania rezerwowego menedżera kolejek z kopią kroniki lokalnej z instancji menedżera kolejek, w której wystąpił błąd. Aby automatycznie przestać niezależną ASP do innego serwera, należy zainstalować i skonfigurować obsługę technologii klastrowej. Istnieje wiele rozwiązań wysokiej dostępności dla niezależnych pul ASP opartych na obsłudze klastrów oraz niskopoziomowego zapisu lustrzanego dysków, które można łączyć z menedżerami kolejek z wieloma instancjami lub zastępować je za pomocą menedżerów kolejek z wieloma instancjami.

Poniższa lista zawiera opis komponentów, które są niezbędne do zbudowania niezawodnego rozwiązania opartego na niezależnych ASP.

Tworzenie kroniki

Menedżery kolejek i inne aplikacje używają kronik lokalnych do bezpiecznego zapisywania danych trwałych na dysku w celu ochrony przed utratą danych w pamięci z powodu awarii serwera. Czasami jest to konsystencja z punktu widzenia punktu w czasie. Nie gwarantuje to spójności wielu aktualizacji, które mają miejsce w danym okresie.

Kontrola transakcji

Korzystając z transakcji globalnych, można koordynować aktualizacje komunikatów i baz danych, tak aby dane zapisywane w kronice były spójne. Zapewnia spójność przez pewien okres czasu, używając protokołu zatwierdzania dwufazowego.

Dysk przełączany

Dyski przełączane są zarządzane przez grupę zasobów klastra urządzeń (CRG) w klastrze wysokiej dostępności. Grupa CRG przełącza niezależne ASP automatycznie na nowy serwer w przypadku nieplanowanego wyłączenia. Grupy zasobów klastra są geograficznie ograniczone do zakresu lokalnej magistrali we/wy.

Konfigurując kronikę lokalną w przełączalnej niezależnej puli ASP, można przestać kronikę do innego serwera i wznowić przetwarzanie komunikatów. Nie są tracone żadne zmiany w trwałych komunikatach wykonanych bez sterowania punktem synchronizacji ani zatwierdzone z kontrolą punktu synchronizacji, chyba że niezależna ASP nie powiedzie się.

Jeśli używane jest zarówno kronikowanie, jak i kontrola transakcji dla przełączalnych niezależnych pul ASP, można przestać kroniki bazy danych i dzienniki menedżera kolejek do innego serwera, a następnie wznowić przetwarzanie transakcji bez utraty spójności lub zatwierdzonych transakcji.

Międzyośrodkowe zapis lustrzany (XSM)

XSM odzwierciedla podstawową niezależną ASP do geograficznie zdalnej, dodatkowej niezależnej ASP w sieci TCP/IP i automatycznie przekazuje sterowanie w przypadku awarii. Użytkownik ma możliwość skonfigurowania synchronicznego lub asynchronicznego zapisu lustrzanego. Synchroniczny zapis lustrzany zmniejsza wydajność menedżera kolejek, ponieważ tworzona jest kopia lustrzana danych przed wykonaniem operacji zapisu w systemie produkcyjnym, ale gwarantuje, że dodatkowa niezależna ASP jest aktualna. Jeśli używany jest asynchroniczny zapis lustrzany, nie można zagwarantować, że dodatkowa niezależna ASP jest aktualna. Asynchroniczny zapis lustrzany zachowuje spójność dodatkowej niezależnej ASP.

Istnieją trzy technologie XSM.

Geograficzny zapis lustrzany

Geograficzny zapis lustrzany jest rozszerzeniem technologii klastrowej, co umożliwia przełączanie niezależnych pul ASP na całym obszarze. Ma on zarówno tryb synchroniczny, jak i asynchroniczny. Wysoką dostępność można zagwarantować tylko w trybie synchronicznym, ale separacja niezależnych pul ASP może mieć zbyt wiele wpływu na wydajność. Istnieje możliwość połączenia geograficznego zapisu lustrzanego z dyskiem przełączanym w celu zdalnego udostępniania wysokiej dostępności lokalnie i odtwarzania po awarii.

Zapis lustrzany MAN

Zapis lustrzany MAN to usługa na poziomie urządzenia, która zapewnia szybki lokalny synchroniczny zapis lustrzany na dłuższych odległościach niż magistrala lokalna. Można je połączyć z menedżerem kolejek z wieloma instancjami w celu zapewnienia wysokiej dostępności menedżera kolejek oraz poprzez posiadanie dwóch kopii niezależnej ASP, wysokiej dostępności kroniki menedżera kolejek.

Globalny zapis lustrzany

Globalny zapis lustrzany to usługa na poziomie urządzenia, która zapewnia asynchroniczny zapis lustrzany i nadaje się do tworzenia kopii zapasowych i odtwarzania po awarii na dłuższych odległościach, ale nie jest normalnym wyborem w przypadku wysokiej dostępności, ponieważ utrzymuje spójność tylko w czasie, a nie w walucie.

Kluczowe punkty decyzyjne, które należy wziąć pod uwagę, są

ASP lub niezależna ASP?

Nie ma potrzeby uruchamiania klastra IBM i HA w celu używania menedżerów kolejek z wieloma instancjami. Można wybrać niezależne ASP, jeśli korzystasz już z niezależnych ASP lub masz wymagania dotyczące dostępności dla innych aplikacji, które wymagają niezależnych ASP. Być może warto połączyć niezależne pule ASP z menedżerami kolejek z wieloma instancjami w celu zastąpienia monitorowania menedżera kolejek jako środka wykrywającego awarię menedżera kolejek.

Dostępność?

Jaki jest cel czasu odtwarzania (RTO)? Jeśli wymagany jest wygląd niemal niezakłóconego zachowania, to które rozwiązanie ma najszybszy czas odtwarzania?

Dostępność kroniki?

W jaki sposób eliminuje się kronikę jako pojedynczy punkt awarii. Użytkownik może przyjąć rozwiązanie sprzętowe, korzystając z urządzeń RAID 1 lub lepiej, albo połączyć lub użyć rozwiązania programowego przy użyciu dzienników repliki lub zapisu lustrzanego dysków.

Odległość?

Określa, jak daleko są aktywne i rezerwowe instancje menedżera kolejek. Czy użytkownicy mogą tolerować degradację wydajności replikowania synchronicznie na odległość większą niż 250 metrów?

Umiejętności?

Istnieje praca, która ma zostać wykonana w celu zautomatyzowania czynności administracyjnych związanych z utrzymaniem i regularnym wykonywaniem tego rozwiązania. Umiejętności wymagane do automatyzacji są różne dla rozwiązań opartych na ASP i niezależnych ASP.

IBM i *Usuwanie menedżera kolejek z wieloma instancjami w systemie IBM i*

Przed usunięciem menedżera kolejek z wieloma instancjami należy zatrzymać zdalne kronikowanie i usunąć instancje menedżera kolejek.

Zanim rozpocznie

1. W tym przykładzie dwie instancje menedżera kolejek produktu QM1 są zdefiniowane na serwerach ALPHA i BETA. ALPHA jest aktywną instancją, a BETA jest w stanie gotowości. Dane menedżera kolejek powiązane z menedżerem kolejek QM1 są przechowywane na serwerze IBM i GAMMA przy użyciu serwera NetServer. Więcej informacji zawiera sekcja "Tworzenie menedżera kolejek z wieloma instancjami przy użyciu zapisu lustrzanego dziennika i serwera NetServer w systemie IBM i" na stronie 306.
2. Należy połączyć wartości ALPHA i BETA, tak aby wszystkie zdefiniowane zdalne kroniki mogły zostać usunięte przez produkt IBM MQ.
3. Sprawdź, czy dostęp do współużytkowanego zasobu plikowego katalogu /QNTC i katalogu serwera można uzyskać za pomocą komend systemowych **EDTF** lub **WRKLNK**.

O tym zadaniu

Przed usunięciem menedżera kolejek z wieloma instancjami z serwera za pomocą komendy **DLTMQM** należy usunąć wszystkie instancje menedżera kolejek na innych serwerach za pomocą komendy **RMVMQMINF**.

Usunięcie instancji menedżera kolejek za pomocą komendy **RMVMQMINF** powoduje usunięcie lokalnych i zdalnych kronik z przedrostkiem AMQ, które są powiązane z instancją. Informacje konfiguracyjne dotyczące instancji menedżera kolejek, lokalne na serwerze, są również usuwane.

Nie należy uruchamiać komendy **RMVMQMINF** na serwerze, na którym znajduje się pozostała instancja menedżera kolejek. Dzięki temu produkt **DLTMQM** nie działa poprawnie.

Usuń menedżera kolejek za pomocą komendy **DLTMQM**. Dane menedżera kolejek są usuwane z współużytkowanego zasobu sieciowego. Lokalne i zdalne kroniki z przedrostkiem AMQ i powiązane z tą instancją są usuwane. Produkt **DLTMQM** usuwa również informacje konfiguracyjne dotyczące instancji menedżera kolejek, które są lokalne na serwerze.

W tym przykładzie istnieją tylko dwie instancje menedżera kolejek. Produkt IBM MQ obsługuje działającą konfigurację z wieloma instancjami, która zawiera jedną aktywną instancję menedżera kolejek i jedną instancję rezerwową. Jeśli utworzono dodatkowe instancje menedżera kolejek, które mają być używane w działających konfiguracjach, usuń je za pomocą komendy **RMVMQMINF** przed usunięciem pozostałej instancji.

Procedura

1. Uruchom komendę **CHGMQMJRN RMTJRNSTS** (*INACTIVE) na każdym serwerze, aby zdalne kronikowanie między instancjami menedżera kolejek było nieaktywne.

a) W PRZYPADKU ALFA:

```
CHGMQMJRN MQMNAME('QM1')
RMTJRNRDB('BETA') RMTJRNSTS(*INACTIVE)
```

b) W dniu BETA:

```
CHGMQMJRN MQMNAME('QM1')
RMTJRNRDB('ALPHA') RMTJRNSTS(*INACTIVE)
```

2. Aby zatrzymać obie instancje produktu QM1, należy uruchomić komendę **ENDMQM** w komendzie ALPHA, czyli aktywną instancję menedżera kolejek.

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) INSTANCE(*ALL) ENDCCTJOB(*YES)
```

3. Uruchom komendę **RMVMQMINF** w systemie ALPHA, aby usunąć zasoby menedżera kolejek dla instancji z wartości ALPHA i BETA.

```
RMVMQMINF MQMNAME(QM1)
```

Produkt **RMVMQMINF** usuwa informacje o konfiguracji menedżera kolejek dla produktu QM1 z poziomu ALPHA. Jeśli nazwa kroniki jest poprzedzona przedrostkiem AMQ, zostanie usunięta kronika lokalna powiązana z QM1 z ALPHA. Jeśli nazwa kroniki jest poprzedzona przedrostkiem AMQ, a utworzona została kronika zdalna, usuwa ona także kronikę zdalną z BETA.

4. Uruchom komendę **DLTMQM** na BETA, aby usunąć QM1.

```
DLTMQM MQMNAME(QM1)
```

DLTMQM usuwa dane menedżera kolejek z zasobu sieciowego GAMMA. Usuwa z systemu BETA informacje o konfiguracji menedżera kolejek dla produktu QM1. Jeśli nazwa kroniki jest poprzedzona przedrostkiem AMQ, to usuwa dziennik lokalny powiązany z programem QM1 z programu BETA. Jeśli nazwa kroniki jest poprzedzona przedrostkiem AMQ, a utworzona została kronika zdalna, usuwa ona również kronikę zdalną z ALPHA.

Wyniki

Produkty **DLTMQM** i **RMVMQMINF** usuwają kroniki lokalne i zdalne utworzone przez produkty **CRTMQM** i **ADDMQJRN**. Komendy te usuwają również dzienniki. Dzienniki i dzienniki muszą być zgodne z konwencją nazewnictwa o nazwach rozpoczynających się od AMQ. Produkty **DLTMQM** i **RMVMQMINF** usuwają obiekty menedżera kolejek, dane menedżera kolejek oraz informacje o konfiguracji menedżera kolejek z programu `mq.s.ini`.

Co dalej

Alternatywnym podejściem jest wydanie następujących komend po dezaktywowaniu kronikowania w kroku "1" na stronie 322 i przed zakończeniem instancji menedżera kolejek. Albo, jeśli nie przestrzegano konwencji nazewnictwa, należy usunąć dzienniki i dzienniki według nazwy.

1. W PRZYPADKU ALFA:

```
RMVMQMJRN MQMNAME('QM1') RMTJNRDB('BETA')
```

2. W dniu BETA:

```
RMVMQMJRN MQMNAME('QM1') RMTJNRDB('ALPHA')
```

Po usunięciu dzienników kontynuuj od pozostałych kroków.

IBM i Tworzenie kopii zapasowej menedżera kolejek z wieloma instancjami w systemie IBM i

W tej procedurze przedstawiono sposób tworzenia kopii zapasowych obiektów menedżera kolejek na serwerze lokalnym oraz danych menedżera kolejek na sieciowym serwerze plików. Dostosuj przykład, aby utworzyć kopię zapasową danych dla innych menedżerów kolejek.

Zanim rozpocznieasz

W tym przykładzie dane menedżera kolejek powiązane z menedżerem kolejek QM1 są przechowywane na serwerze IBM i o nazwie GAMMA, przy użyciu serwera NetServer. Więcej informacji zawiera sekcja [“Tworzenie menedżera kolejek z wieloma instancjami przy użyciu zapisu lustrzanego dziennika i serwera NetServer w systemie IBM i”](#) na stronie 306. Produkt IBM MQ jest zainstalowany na serwerach, ALPHA i BETA. Menedżer kolejek, QM1, jest skonfigurowany w systemie ALPHA i BETA.

O tym zadaniu

Produkt IBM i nie obsługuje zapisywania danych z katalogu zdalnego. Zeszkładuj dane menedżera kolejek w zdalnym systemie plików, używając procedur tworzenia kopii zapasowych lokalnych na serwerze systemu plików. W tym zadaniu system plików sieciowych znajduje się na serwerze IBM i, GAMMA. Kopia zapasowa danych menedżera kolejek jest tworzona w zbiorze składowania w GAMMA.

Jeśli sieciowy system plików był w systemie Windows lub Linux, dane menedżera kolejek można zapisać w pliku skompresowanym, a następnie zapisać dane. W przypadku systemu rezerwowego, takiego jak Tivoli Storage Manager, należy użyć go do utworzenia kopii zapasowej danych menedżera kolejek.

Procedura

1. Utwórz zbiór składowania w systemie ALPHA dla biblioteki menedżera kolejek powiązanej z produktem QM1.

Użyj nazwy biblioteki menedżera kolejek, aby określić nazwę zbioru składowania.

```
CRTSAVF FILE(QGPL/QMQM1)
```

2. Zeszkładuj bibliotekę menedżera kolejek w zbiorze składowania w systemie ALPHA.

```
SAVLIB LIB(QMQM1) DEV(*SAVF) SAVF(QGPL/QMQM1)
```

3. Utwórz zbiór składowania dla katalogu danych menedżera kolejek w GAMMA.

Użyj nazwy menedżera kolejek, aby nazwać zbiór składowania.

```
CRTSAVF FILE(QGPL/QMDQM1)
```

4. Zapisz kopię danych menedżera kolejek z katalogu lokalnego w GAMMA.

```
SAV DEV('/QSYS.LIB/QGPL.LIB/QMDQM1.FILE') OBJ('/QIBM/Userdata/mqm/qmgrs/QM1')
```

Komendy służące do konfigurowania menedżerów kolejek z wieloma instancjami

Produkt IBM MQ zawiera komendy ułatwiające konfigurowanie replikacji kroniki, dodawanie nowych instancji menedżera kolejek i konfigurowanie menedżerów kolejek w celu użycia niezależnej ASP.

Komendy kronik służące do tworzenia kronik lokalnych i zdalnych oraz zarządzania nimi są następujące:

ADDMQMJRN,

Za pomocą tej komendy można utworzyć nazwane kroniki lokalne i zdalne dla instancji menedżera kolejek oraz skonfigurować, czy replikacja jest synchroniczna, czy asynchroniczna, jaki jest limit czasu synchronicznego i czy kronika zdalna ma być natychmiast aktywowana.

CHGMQMJRN

Komenda modyfikuje limit czasu, status i parametry dostarczania wpływające na dzienniki repliki.

Komenda RMVMQMJRN

Usuwa nazwane *zdalne* kroniki z instancji menedżera kolejek.

WRKMQMJRN,

Wyświetla status dzienników lokalnych i zdalnych dla instancji menedżera kolejek lokalnych.

Dodaj kolejne instancje menedżera kolejek i zarządzaj nimi za pomocą następujących komend, które modyfikują plik `mq5.ini`.

ADDMQMINF

Komenda korzysta z informacji wyodrębnionych z pliku `mq5.ini` za pomocą komendy `DSPMQMINF` w celu dodania nowej instancji menedżera kolejek na innym serwerze IBM i.

RMVMQMINF

Usuń instancję menedżera kolejek. Użyj tej komendy, aby usunąć instancję istniejącego menedżera kolejek lub usunąć informacje konfiguracyjne dla menedżera kolejek, który został usunięty z innego serwera.

Komenda **CRTMQM** ma trzy parametry pomocne podczas konfigurowania menedżera kolejek z wieloma instancjami,

MQMDIRP (*DFT | *przedrostek-katalogu*)

Ten parametr służy do wybierania punktu podłączenia, który jest odwzorowany na dane menedżera kolejek w sieciowej pamięci masowej.

ASP (*SYSTEM|*ASPDEV| *numer-puli-pamięci-dyskowej*)

Podaj `*SYSTEM` lub *numer-puli-pamięci-dyskowej*, aby umieścić kronikę menedżera kolejek w systemie lub w podstawowej ASP użytkowników. Wybierz opcję `*ASPDEV`, a także ustaw nazwę urządzenia za pomocą parametru **ASPDEV**, aby umieścić kronikę menedżera kolejek w niezależnej puli ASP.

ASPDEV (*ASP|*nazwa-urządzenia*)

Podaj *nazwa-urządzenia* podstawowego lub dodatkowego niezależnego urządzenia ASP. Wybranie opcji `*ASP` ma taki sam wynik, jak podanie wartości **ASP** (`*SYSTEM`).

Uwagi dotyczące wydajności i przetaczania awaryjnego dysków w systemie IBM i

Aby zwiększyć wydajność i niezawodność, należy użyć różnych pul pamięci dyskowej.

Jeśli w aplikacjach używana jest duża liczba trwałych komunikatów lub dużych komunikatów, czas przeznaczony na zapisywanie tych komunikatów na dysku staje się istotnym czynnikiem w wydajności systemu.

Upewnij się, że masz wystarczającą aktywację dysku, aby poradzić sobie z tą możliwością, lub weź pod uwagę osobną pulę pamięci dyskowej (Auxiliary Storage Pool-ASP), w której mają być przechowywane dzienniki menedżera kolejek.

Przy tworzeniu menedżera kolejek przy użyciu parametru ASP produktu **CRTMQM** można określić, która pula ASP ma być przechowywana biblioteka i kroniki menedżera kolejek. Domyślnie biblioteka i kroniki menedżera kolejek oraz dane IFS są przechowywane w systemowej ASP.

Pule ASP umożliwiają izolowanie obiektów na jednej lub większej ilości konkretnych jednostek dyskowych. Może to również zmniejszyć utratę danych ze względu na awarię nośnika dyskowego. W większości przypadków utracono tylko dane przechowywane na jednostkach dyskowych w danej ASP.

Zalecane jest przechowywanie biblioteki menedżera kolejek i danych kroniki w oddzielnych pulach ASP użytkowników w systemie plików root systemu IFS w celu zapewnienia przełączenia awaryjnego i zmniejszenia rywalizacji o dysk.

Więcej informacji na ten temat zawiera sekcja [Składowanie i odtwarzanie](#).

IBM i Korzystanie z komendy SAVLIB w celu składowania bibliotek IBM MQ w systemie IBM i

Nie można używać produktu SAVLIB LIB (*ALLUSR) do zapisywania bibliotek produktu IBM MQ, ponieważ te biblioteki mają nazwy rozpoczynające się od litery Q.

Za pomocą programu SAVLIB LIB(QM*) można składać wszystkie biblioteki menedżera kolejek, ale tylko wtedy, gdy używane jest urządzenie składowania inne niż *SAVF. W przypadku systemu DEV (*SAVF) należy użyć komendy SAVLIB dla każdej biblioteki menedżera kolejek i każdej biblioteki menedżera kolejek w systemie.

IBM i WYCISZANIE IBM MQ for IBM i

W tej sekcji opisano sposób wyciszenia (na koniec z wdziękiem) IBM MQ for IBM i.

Aby wyciszyć IBM MQ for IBM i:

1. Zaloguj się do nowej interaktywnej sesji IBM MQ for IBM i, upewniając się, że nie uzyskujesz dostępu do żadnych obiektów.
2. Upewnij się, że:
 - Uprawnienie *ALLOBJ lub uprawnienie do zarządzania obiektami dla biblioteki QMQM
 - Wystarczające uprawnienia do korzystania z komendy ENDSBS
3. Poinformuj wszystkich użytkowników, którzy mają zamiar zatrzymać produkt IBM MQ for IBM i.
4. Sposób kontynuowania zależy od tego, czy chcesz zamknąć (wygasić) pojedynczy menedżer kolejek (gdzie inne mogą istnieć) (patrz [“Zamykanie pojedynczego menedżera kolejek dla produktu IBM MQ for IBM i”](#) na stronie 326) lub wszystkich menedżerów kolejek (patrz [“Zamykanie wszystkich menedżerów kolejek w systemie IBM MQ for IBM i”](#) na stronie 327).

ENDMQM, parametr ENDCCTJOB (*YES)

Parametr ENDMQM ENDCCTJOB (*YES) działa inaczej w programie IBM MQ for IBM i 6.0, a następnie w porównaniu z poprzednimi wersjami.

W poprzednich wersjach, jeśli podano wartość ENDCCTJOB (*YES), program MQ wymuszał zakończenie aplikacji dla użytkownika.

W systemie IBM MQ for IBM i 6.0 lub nowszym, jeśli zostanie podana wartość ENDCCTJOB (*YES), aplikacje nie zostaną zakończone, ale zostaną odłączone od menedżera kolejek.

Jeśli określono wartość ENDCCTJOB (*YES) i istnieją aplikacje, które nie zostały zapisane w celu wykrycia zakończenia menedżera kolejek, podczas następnego wywołania nowego wywołania MQI wywołanie zostanie zwrócone z błędem MQRC_CONNECTION_BROKEN (2009).

Jako alternatywę dla komendy ENDCCTJOB (*YES) należy użyć parametru ENDCCTJOB (*NO) i użyć opcji 22 WRKMQM (Praca z zadaniami), aby ręcznie zakończyć wszystkie zadania aplikacji, które zapobiegą restartowaniu menedżera kolejek.

IBM i Zamykanie pojedynczego menedżera kolejek dla produktu IBM MQ for IBM i

Ten temat zawiera informacje na temat trzech typów zamykania systemu.

W kolejnych procedurach używamy przykładowej nazwy menedżera kolejek QMgr1 i przykładowej nazwy podsystemu SUBX. Jeśli to konieczne, zastąp te nazwy własnymi wartościami.

Planowane zamknięcie systemu

Planowane zamknięcie menedżera kolejek w systemie IBM i

1. Przed zamknięciem wykonaj:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(QMgr1) DSPJRNTA(*YES)
```

2. Aby zamknąć menedżera kolejek, wykonaj następujące czynności:

```
ENDMQM MQMNAME(QMgr1) OPTION(*CNTRLD)
```

Jeśli QMgr1 nie kończy się, kanał lub aplikacje są prawdopodobnie zajęte.

3. Jeśli musisz natychmiast zamknąć QMgr1, wykonaj następujące czynności:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)  
ENDCCTJOB(*YES) TIMEOUT(15)
```

Nieplanowane zamknięcie systemu

1. Aby zamknąć menedżera kolejek, wykonaj następujące czynności:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)
```

Jeśli QMgr1 nie kończy się, kanał lub aplikacje są prawdopodobnie zajęte.

2. Jeśli konieczne jest natychmiastowe zamknięcie QMgr1, wykonaj następujące czynności:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)  
ENDCCTJOB(*YES) TIMEOUT(15)
```

Zamknij się w nienormalnych warunkach

1. Aby zamknąć menedżera kolejek, wykonaj następujące czynności:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)
```

Jeśli QMgr1 nie kończy się, przejdź do kroku 3, pod warunkiem, że:

- QMgr1 znajduje się we własnym podsystemie lub
 - Można zakończyć wszystkie menedżery kolejek, które współużytkują ten sam podsystem, co QMgr1. Użyj procedury nieplanowanego zamknięcia dla wszystkich takich menedżerów kolejek.
2. Po wykonaniu wszystkich kroków procedury dla wszystkich menedżerów kolejek współużytkującej podsystem (SUBX na naszych przykładach) wykonaj następujące czynności:

```
ENDSBS SUBX *IMMED
```

Jeśli wykonanie tej komendy nie powiedzie się, należy zamknąć wszystkie menedżery kolejek, używając procedury nieplanowanego zamknięcia systemu, a następnie wykonać IPL na komputerze.

Ostrzeżenie: Nie należy używać ENDJOBABN dla zadań IBM MQ , które kończą się niepowodzeniem w wyniku działania ENDJOB lub ENDSBS, chyba że użytkownik jest przygotowany do wykonania IPL na komputerze natychmiast po jego wykonaniu.

3. Uruchom podsystem, wykonując następujące czynności:

```
STRSBS SUBX
```

4. Zamknij menedżera kolejek natychmiast, wykonując następujące czynności:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)  
ENDCCTJOB(*YES) TIMEOUT(10)
```

5. Zrestartuj menedżer kolejek, wykonując następujące czynności:

```
STRMQM MQMNAME(QMgr1)
```

Jeśli ta operacja nie powiedzie się, a użytkownik:

- Restartowanie komputera, wykonując IPL, lub
- Mają tylko jeden menedżer kolejek

Dokręć pamięć współużytkowaną IBM MQ , wykonując następujące czynności:

```
ENDMQM MQMNAME(*ALL) OPTION(*IMMED)  
ENDCCTJOB(*YES) TIMEOUT(15)
```

przed powtórzonym krokiem 5.

Jeśli restart menedżera kolejek trwa dłużej niż kilka sekund, program IBM MQ automatycznie dodaje komunikaty o statusie do dziennika zadania, w którym szczegółowo określa się postęp uruchamiania.

Jeśli nadal występują problemy z restartowaniem menedżera kolejek, skontaktuj się z działem wsparcia produktu IBM . Wszelkie dalsze działania, które mogą zostać podjęte, mogą spowodować uszkodzenie menedżera kolejek, pozostawiając produkt IBM MQ , który nie może zostać odzyskany.

IBM i Zamykanie wszystkich menedżerów kolejek w systemie IBM MQ for IBM i

Ten temat zawiera informacje na temat trzech typów zamykania systemu.

Procedury są prawie takie same, jak w przypadku pojedynczego menedżera kolejek, ale w miarę możliwości przy użyciu *ALL menedżera kolejek zamiast nazwy menedżera kolejek, a w przeciwnym razie za pomocą komendy wielokrotnie za pomocą każdej nazwy menedżera kolejek, z kolei. Podczas wykonywania procedur używamy przykładowej nazwy menedżera kolejek QMgr1 i przykładowej nazwy podsystemu SUBX. Zastąp je własnymi.

Planowane zamknięcie systemu

1. Jedna godzina przed zamknięciem, wykonaj:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(QMgr1) DSPJRNTA(*YES)
```

Powtórz tę czynność dla każdego menedżera kolejek, który ma zostać zamknięty.

2. Aby zamknąć menedżera kolejek, wykonaj następujące czynności:

```
ENDMQM MQMNAME(QMgr1) OPTION(*CNTRLD)
```

Powtórz tę czynność dla każdego menedżera kolejek, który ma zostać zamknięty; oddzielne komendy mogą być uruchamiane równoległe.

Jeśli dowolny menedżer kolejek nie zakończy się w rozsądnym czasie (na przykład 10 minut), przejdź do kroku 3.

3. Aby natychmiast zamknąć wszystkie menedżery kolejek, wykonaj następujące czynności:

```
ENDMQM MQMNAME(*ALL) OPTION(*IMMED)
ENDCCTJOB(*YES) TIMEOUT(15)
```

Nieplanowane zamknięcie systemu

1. Aby zamknąć menedżera kolejek, wykonaj następujące czynności:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)
```

Powtórz tę czynność dla każdego menedżera kolejek, który ma zostać zamknięty; oddzielne komendy mogą być uruchamiane równoległe.

Jeśli menedżery kolejek nie zostaną zakończone, kanał lub aplikacje będą prawdopodobnie zajęte.

2. Jeśli konieczne jest natychmiastowe zamknięcie menedżerów kolejek, wykonaj następujące czynności:

```
ENDMQM MQMNAME(*ALL) OPTION(*IMMED)
ENDCCTJOB(*YES) TIMEOUT(15)
```

Zamknij się w nienormalnych warunkach

1. Aby zamknąć menedżery kolejek, wykonaj następujące czynności:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)
```

Powtórz tę czynność dla każdego menedżera kolejek, który ma zostać zamknięty; oddzielne komendy mogą być uruchamiane równoległe.

2. Zakończ działanie podsystemów (SUBX na naszych przykładach), wykonując następujące czynności:

```
ENDSBS SUBX *IMMED
```

Powtórz tę czynność dla każdego podsystemu, który ma zostać zamknięty; oddzielne komendy mogą być uruchamiane równoległe.

Jeśli wykonanie tej komendy nie powiedzie się, wykonaj IPL w systemie.

Ostrzeżenie: Nie należy używać ENDJOBABN dla zadań, które kończą się niepowodzeniem w wyniku działania ENDJOB lub ENDSBS, chyba że użytkownik jest przygotowany do wykonania IPL w systemie natychmiast po jego zakończeniu.

3. Uruchom podsystemy, wykonując następujące czynności:

```
STRSBS SUBX
```

Powtórz tę czynność dla każdego podsystemu, który ma zostać uruchomiony.

4. Zamknij menedżery kolejek w dół natychmiast, wykonując następujące czynności:

```
ENDMQM MQMNAME(*ALL) OPTION(*IMMED)
ENDCCTJOB(*YES) TIMEOUT(15)
```

5. Zrestartuj menedżery kolejek, wykonując następujące czynności:

STRMQM MQMNAME(QMgr1)

Powtórz tę czynność dla każdego menedżera kolejek, który ma zostać uruchomiony.

Jeśli jakkolwiek restart menedżera kolejek trwa dłużej niż kilka sekund, program IBM MQ będzie wyświetlał komunikaty o statusie w sposób sporny, co spowoduje, że postęp uruchamiania będzie wyświetlany w sposób sporny.

Jeśli nadal występują problemy z restartowaniem dowolnego menedżera kolejek, skontaktuj się z działem wsparcia produktu IBM . Wszelkie dalsze działania, które mogą zostać podjęte, mogą spowodować uszkodzenie menedżerów kolejek, pozostawiając produkt MQSeries lub IBM MQ , który nie może zostać odzyskany.

z/OS

administrowanieIBM MQ for z/OS

Administrowanie menedżerami kolejek i powiązаныmi zasobami obejmuje zadania, które często są wykonywane w celu aktywowania zasobów i zarządzania nimi. Wybierz metodę administrowania menedżerami kolejek i powiązаныmi zasobami.

Produkt IBM MQ for z/OS może być sterowany i zarządzany za pomocą zestawu programów narzędziowych i programów dostarczanych wraz z produktem. Do administrowania produktem IBM MQ for z/OSmożna użyć komend IBM MQ Script (MQSC) lub Programmable Command Formats (PCFs). Informacje na temat używania komend dla produktu IBM MQ for z/OSzawiera sekcja [“Wysyłanie komend do programu IBM MQ for z/OS”](#) na stronie 330.

Produkt IBM MQ for z/OS udostępnia także zestaw programów narzędziowych, które ułatwiają administrowanie systemem. Więcej informacji na temat różnych programów narzędziowych oraz sposobu ich używania zawiera sekcja [“Programy narzędziowe serwera IBM MQ for z/OS”](#) na stronie 338.

Szczegółowe informacje na temat administrowania programem IBM MQ for z/OS i różnych czynności administracyjnych, które mogą być wykonywane przez użytkownika, można znaleźć w następujących odsyłaczach:

Pojęcia pokrewne

[“administrowanieIBM MQ”](#) na stronie 5

Aby administrować menedżerami kolejek produktu IBM MQ i powiązаныmi zasobami, należy wybrać preferowaną metodę z zestawu zadań, które mogą być używane do aktywowania i zarządzania tymi zasobami.

Zadania pokrewne

[“Administrowanie lokalnymi obiektami IBM MQ”](#) na stronie 143

Istnieje możliwość administrowania lokalnymi obiektami produktu IBM MQ w celu obsługi aplikacji, które używają interfejsu kolejki komunikatów (Message Queue Interface-MQI).

[“Administrowanie zdalnymi obiektami IBM MQ”](#) na stronie 196

W tej sekcji opisano sposób administrowania obiektami produktu IBM MQ w zdalnym menedżerze kolejek za pomocą komend MQSC oraz sposobu korzystania z obiektów kolejki zdalnej w celu sterowania miejscem docelowym komunikatów i komunikatów odpowiedzi.

Informacje pokrewne

[IBM MQ for z/OS pojęcia](#)

[Planowanie](#)

[Planowanie środowiska IBM MQ w systemie z/OS](#)

[Konfigurowanie](#)

[Konfigurowanie produktu z/OS](#)

[Skorowidz formatów komend programowalnych](#)

[Skorowidz komend MQSC](#)

[Korzystanie z programów narzędziowych IBM MQ for z/OS](#)

Wysyłanie komend do programu IBM MQ for z/OS

Aby sterować menedżerem kolejek, można użyć komend skryptowych IBM MQ (MQSC) w trybie wsadowym lub interaktywnym.

Produkt IBM MQ for z/OS obsługuje komendy MQSC, które mogą być wydawane z następujących źródeł:

- Konsola z/OS lub jej odpowiednik (na przykład SDSF/TSO).
- Wejściowe zestawy danych inicjowania.
- Dostarczony program narzędziowy do przetwarzania wsadowego, CSQUTIL, przetwarzający listę komend w sekwencyjnym zestawie danych.
- Odpowiednio autoryzowana aplikacja, wysyłając komendę jako komunikat do kolejki wejściowej komend. Aplikacja może mieć jedną z następujących wartości:
 - Program regionu wsadowego
 - Aplikacja CICS
 - Aplikacja IMS
 - Aplikacja TSO
 - Program użytkowy lub program narzędziowy w innym systemie IBM MQ

Tabela 24 na stronie 333 zawiera podsumowanie komend MQSC oraz źródeł, z których mogą być wydawane.

Duża część funkcjonalności tych komend jest dostępna w wygodny sposób z poziomu IBM MQ for z/OS operacji i paneli sterujących.

Zmiany wprowadzone w definicjach zasobów menedżera kolejek przy użyciu komend (bezpośrednio lub pośrednio) są zachowywane po restarcie podsystemu IBM MQ .

Program IBM MQ for z/OS obsługuje również komendy PCF (Programmable Command Format). Te uproszczenia upraszczają tworzenie aplikacji dla administrowania produktem IBM MQ. Komendy MQSC są w postaci czytelnej dla użytkownika, podczas gdy program PCF umożliwia aplikacjom tworzenie żądań i odczytywanie odpowiedzi bez konieczności analizowania łańcuchów tekstowych. Podobnie jak w przypadku komend MQSC, aplikacje wydają komendy PCF, wysyłając je jako komunikaty do kolejki wejściowej komend. Więcej informacji na temat używania komend PCF i szczegółów dotyczących komend zawiera dokumentacja produktu [Skorowidz formatów komend programowalnych](#) .

Definicje prywatne i globalne w systemie IBM MQ for z/OS

Podczas definiowania obiektu w systemie IBM MQ for z/OS można zdecydować, czy ma być współużytkowana ta definicja z innymi menedżerami kolejek (definicja *globalna*), czy definicja obiektu ma być używana tylko przez jeden menedżer kolejek (definicja *prywatna*). Jest to nazywane obiektem *rozporządzania*.

Definicja globalna

Jeśli menedżer kolejek należy do grupy współużytkowania kolejek, można wybrać opcję współużytkowania wszystkich definicji obiektów, które zostaną określone przez innych członków grupy. Oznacza to, że użytkownik musi zdefiniować obiekt tylko raz, zmniejszając całkowitą liczbę definicji wymaganych dla całego systemu.

Definicje obiektów globalnych znajdują się w *repozytorium współużytkowanym* (współużytkowana baza danych Db2) i są dostępne dla wszystkich menedżerów kolejek w grupie współużytkowania kolejek. Te obiekty mają dyspozycję grupy.

Definicja prywatna

Aby utworzyć definicję obiektu, która jest wymagana tylko przez jeden menedżer kolejek lub jeśli menedżer kolejek nie jest elementem grupy współużytkowania kolejek, można utworzyć definicje obiektów, które nie są współużytkowane z innymi elementami grupy współużytkowania kolejek.

Definicje obiektów prywatnych znajdują się na stronie zerowej definiującego menedżera kolejek. Te obiekty mają dyspozycję QMGR.

Istnieje możliwość tworzenia prywatnych definicji dla wszystkich typów obiektów IBM MQ z wyjątkiem struktur CF (czyli kanałów, list nazw, definicji procesów, kolejek, menedżerów kolejek, definicji klas pamięci masowej i obiektów informacji uwierzytelniających) oraz globalnych definicji dla wszystkich typów obiektów z wyjątkiem menedżerów kolejek.

Produkt IBM MQ automatycznie kopiuje definicję obiektu grupy do zestawu stron zero dla każdego menedżera kolejek, który go używa. Kopię definicji można tymczasowo zmienić, jeśli użytkownik chce, a program IBM MQ umożliwia odświeżenie kopii zestawu stron z kopii repozytorium, jeśli jest to wymagane.

Program IBM MQ zawsze próbuje odświeżyć zestaw stron kopii z kopii repozytorium podczas uruchamiania (w przypadku komend kanału, jest to wykonywane po restarcie inicjatora kanału) lub jeśli obiekt grupy jest zmieniany.

Uwaga: Kopia definicji jest odświeżana z definicji grupy, tylko wtedy, gdy definicja grupy została zmieniona po utworzeniu kopii definicji.

Zapewnia to, że kopie zestawu stron odzwierciedlają wersję w repozytorium, w tym wszelkie zmiany, które zostały wprowadzone w przypadku, gdy menedżer kolejek był nieaktywny. Kopie są odświeżane przez generowanie komend DEFINE REPLACE, dlatego istnieją okoliczności, w których odświeżanie nie jest wykonywane, na przykład:

- Jeśli kopia kolejki jest otwarta, odświeżenie, które zmienia użycie kolejki, nie powiedzie się.
- Jeśli kopia kolejki ma w niej komunikaty, odświeżenie, które usuwa tę kolejkę, nie powiedzie się.
- Jeśli kopia kolejki wymagałaby zmiany w instrukcji ALTER z FORCE.

W tych okolicznościach odświeżanie nie jest wykonywane na tej kopii, ale jest wykonywane na kopiach we wszystkich innych menedżerach kolejek.

Jeśli menedżer kolejek jest wyłączony, a następnie zrestartowany autonomicznie, usuwane są wszystkie lokalne kopie obiektów, chyba że na przykład kolejka ma powiązane komunikaty.

Istnieje trzecia dyspozycja obiektu, która ma zastosowanie tylko do kolejek lokalnych. Umożliwia to tworzenie kolejek współużytkowanych. Definicja dla kolejki współużytkowanej jest wstrzymana w repozytorium współużytkowanym i jest dostępna dla wszystkich menedżerów kolejek w grupie współużytkowania kolejek. Ponadto komunikaty znajdujące się w kolejce współużytkowanej są również dostępne dla wszystkich menedżerów kolejek w grupie współużytkowania kolejek. Jest to opisane w sekcji [Kolejki współużytkowane i grupy współużytkowania kolejek](#). Współużytkowane kolejki mają dyspozycję obiektu SHARED.

W poniższej tabeli podsumowano wpływ opcji rozporządzania obiektów dla menedżerów kolejek uruchomionych samodzielnie oraz jako element grupy współużytkowania kolejek.

Dyspozycja	Autonomiczny menedżer kolejek	Członek grupy współużytkowania kolejki
QMGR	Definicja obiektu wstrzymana na stronie o wartości zero.	Definicja obiektu wstrzymana na stronie o wartości zero.
GRUPA	są niedozwolone.	Definicja obiektu przechowana w repozytorium współużytkowanym. Kopia lokalna wstrzymana na stronie zerowej dla każdego menedżera kolejek w grupie.
Współużytkowane	są niedozwolone.	Definicja kolejki utrzymana we współużytkowanym repozytorium. Komunikaty dostępne dla dowolnego menedżera kolejek w grupie.

Manipulowanie definicjami globalnymi

Aby zmienić definicję obiektu przechowanego w repozytorium współużytkowanym, należy określić, czy ma zostać zmieniona wersja w repozytorium, czy też lokalna kopia na stronie ma wartość zero. Aby to zrobić, należy użyć dyspozycji obiektu jako części komendy.

z/OS Kierowanie komend do różnych menedżerów kolejek w systemie z/OS

Za pomocą *zasięgu komendy* można określić, który menedżer kolejek jest uruchamiany przez komendę.

Można wybrać opcję wykonania komendy w menedżerze kolejek, w której jest ona wprowadzana, lub w innym menedżerze kolejek w grupie współużytkowania kolejek. Można również wybrać opcję wydawania konkretnej komendy równoległe we wszystkich menedżerach kolejek w grupie współużytkowania kolejek. Jest to możliwe zarówno w przypadku komend MQSC, jak i komend PCF.

Wartość ta jest określana przez *zasięg komendy*. Zasięg komendy jest używany wraz z dyspozycją obiektu w celu określenia wersji obiektu, z którym ma zostać wykonana praca.

Na przykład można zmienić niektóre atrybuty obiektu, którego definicja znajduje się w repozytorium współużytkowanym.

- Można zmienić wersję tylko w jednym menedżerze kolejek i nie wprowadzać żadnych zmian w wersji w repozytorium ani tych, które są używane przez inne menedżery kolejek.
- Może zaistnieć potrzeba zmiany wersji w repozytorium współużytkowanym dla przyszłych użytkowników, ale pozostawienie istniejących kopii nie zostanie zmienione.
- Zmiana wersji w repozytorium współużytkowanym może być konieczna, ale zmiany mają być natychmiast odzwierciedlone we wszystkich menedżerach kolejek w grupie współużytkowania kolejek, które przechowują kopię obiektu na ich stronie ustawionej zero.

Za pomocą zasięgu komendy można określić, czy komenda jest wykonywana w tym menedżerze kolejek, innym menedżerze kolejek, czy w każdym menedżerze kolejek. Rozporządzenie obiektu służy do określania, czy obiekt, który jest manipulowany, znajduje się w repozytorium współużytkowanym (obiekt grupy), czy też jest kopią lokalną o zerowym zestawie stron (obiekt menedżera kolejek).

Nie trzeba określać zasięgu komendy i rozporządzenia obiektu do pracy z kolejką współużytkowaną, ponieważ każdy menedżer kolejek w grupie współużytkowania kolejek obsługuje tę kolejkę współużytkowaną jako pojedynczą kolejkę.

z/OS Podsumowanie komend dla IBM MQ for z/OS

Ten temat zawiera informacje na temat głównych komend MQSC i PCF.

Tabela 23 na stronie 332 zawiera podsumowanie komend MQSC i PCF, które są dostępne w produkcji IBM MQ for z/OS w celu zmiany, zdefiniowania, usunięcia i wyświetlenia obiektów IBM MQ .

<i>Tabela 23. Podsumowanie głównych komend MQSC i PCF według typu obiektu</i>				
Komenda MQSC	Zmień	Definiowanie	WYŚWIETL	USUŃ
Komenda PCF	Zmień	Utwórz/Kopiuj	Zapytania	Usuń
AUTHINFO	X	X	X	X
STATUS CFSTATUS			X	
CFSTRUCT	X	X	X	X
CHANNEL	X	X	X	X
CHSTATUS			X	
NAMELIST	X	X	X	X
PROCES	X	X	X	X

Tabela 23. Podsumowanie głównych komend MQSC i PCF według typu obiektu (kontynuacja)

Komenda MQSC	Zmień	Definiowanie	WYŚWIETL	USUŃ
QALIAS	M	M	M	M
QCLUSTER			M	
QLOCAL	M	M	M	M
QMGR	X		X	
QMODEL	M	M	M	M
QREMOTE	M	M	M	M
QUEUE	P	P	X	P
QSTATUS			X	
STGCLASS	X	X	X	X

Klucz do symboli tabeli:

- M = tylko MQSC
- P = tylko PCF
- X = obie

Istnieje wiele innych komend MQSC i PCF, które umożliwiają zarządzanie innymi zasobami produktu IBM MQ, a także wykonywanie innych działań oprócz tych podsumowanych w programie [Tabela 23 na stronie 332](#).

Tabela 24 na stronie [333](#) wyświetla każdą komendę MQSC i w przypadku, gdy każda komenda może zostać wydana z:

- Wejściowy zestaw danych inicjowania CSQINP1
- Wejściowy zestaw danych inicjowania CSQINP2
- z/OS console (lub równoważny)
- SYSTEM.COMMAND.INPUT, kolejka i serwer komend (z aplikacji, CSQUTIL lub zestawu danych wejściowych inicjowania CSQINPX)

Tabela 24. Źródła, z których można uruchamiać komendy MQSC

Komenda	CSQINP1	CSQINP2	konsola z/OS	Kolejka wejściowa komend i serwer
ALTER AUTHINFO		X	X	X
ALTER BUFFPOOL		X	X	X
ALTER CFSTRUCT		X	X	X
ZMIEŃ KANAŁ		X	X	X
ALTER NAMELIST		X	X	X
ALTER PSID			X	X
ALTER PROCESS		X	X	X
ALTER QALIAS		X	X	X
ALTER QLOCAL		X	X	X

Tabela 24. Źródła, z których można uruchamiać komendy MQSC (kontynuacja)

Komenda	CSQINP1	CSQINP2	konsola z/OS	Kolejka wejściowa komend i serwer
ALTER QMGR		X	X	X
ALTER QMODEL		X	X	X
ALTER QREMOTE		X	X	X
ZMIENŹ ZABEZPIECZENIA	X	X	X	X
ALTER STGCLASS		X	X	X
ALTER SUB		X	X	X
ALTER TOPIC		X	X	X
ZMIANA ŚLEDZENIA	X	X	X	X
DZIENNIK ARCHIWUM	X	X	X	X
BACKUP CFSTRUCT			X	X
CLEAR QLOCAL		X	X	X
DEFINE AUTHINFO		X	X	X
DEFINIOWANIE BUFETU	X	X		
DEFINE CFSTRUCT		X	X	X
Zdefiniowanie kanału		X	X	X
ZDEFINIUJ DZIENNIK			X	X
DEFINIUJ LISTĘ NAZW		X	X	X
ZDEFINIUJ PROCES		X	X	X
DEFINE PSID	X		X	X
ZDEFINIUJ ALIAS QALIAS		X	X	X
DEFINE QLOCAL		X	X	X
DEFINE QMODEL		X	X	X
ZDEFINIUJ QREMOTE		X	X	X
DEFINE STGCLASS		X	X	X
DEFINE SUB			X	X
ZDEFINIUJ TEMAT		X	X	X
USUŃ INFORMACJE O AUTORYZACJI		X	X	X
USUŃ BUFFPOOL			X	X
USUŃ CFSTRUCT		X	X	X
Usuń kanał			X	X
USUŃ NAZWĘ LISTY		X	X	X
Usuń proces		X	X	X

Tabela 24. Źródła, z których można uruchamiać komendy MQSC (kontynuacja)

Komenda	CSQINP1	CSQINP2	konsola z/OS	Kolejka wejściowa komend i serwer
USUŃ IDENTYFIKATOR PSID			X	X
USUŃ QALIAS		X	X	X
USUŃ QLOCAL		X	X	X
USUŃ QMODEL		X	X	X
USUŃ QREMOTE		X	X	X
USUŃ KLASĘ STGCLASS		X	X	X
USUŃ SUB		X	X	X
Usuń temat		X	X	X
WYŚWIETL ARCHIWUM	X	X	X	X
WYŚWIETLENIE INFORMACJI UWIERZYTELNIAJĄCYCH		X	X	X
WYŚWIETL STATUS CFSTATUS			X	X
WYŚWIETL CFSTRUCT		X	X	X
WYŚWIETL KANAŁ		X	X	X
WYŚWIETL STATUS CHSTATUS			X	X
WYŚWIETL CLUSQMgr			X	X
WYŚWIETLAJ CMDSERV	X	X	X	X
WYŚWIETL KONTEKST		X	X	X
WYŚWIETL CHINIT		X	X	X
WYŚWIETL GRUPĘ		X	X	X
WYŚWIETL DZIENNIK	X	X	X	X
WYŚWIETLANIE LISTY NAZW		X	X	X
WYŚWIETL PROCES		X	X	X
WYŚWIETL ALIAS QALIAS		X	X	X
WYŚWIETLANIE KLASTRA QCLUSTER		X	X	X
WYŚWIETL QLOCAL		X	X	X
WYŚWIETL QMgr		X	X	X
WYŚWIETLANIE MODELU QMODEL		X	X	X
WYŚWIETL QREMOTE		X	X	X
WYŚWIETL STATUS QSTATUS		X	X	X
WYŚWIETL KOLEJKĘ		X	X	X

Tabela 24. Źródła, z których można uruchamiać komendy MQSC (kontynuacja)

Komenda	CSQINP1	CSQINP2	konsola z/OS	Kolejka wejściowa komend i serwer
WYŚWIETL ZABEZPIECZENIA			X	X
WYŚWIETL KLASĘ STGCLASS		X	X	X
WYŚWIETL SUB		X	X	X
WYŚWIETL TEMAT		X	X	X
WYŚWIETL SYSTEM	X	X	X	X
WYŚWIETL WĄTEK		X	X	X
WYŚWIETL ŚLEDZENIE	X	X	X	X
WYŚWIETL WYKORZYSTANIE		X	X	X
MOVE QLOCAL		X	X	X
KANAŁ PING			X	X
ODZYSKIWANIE BSDS	X	X	X	X
ODZYSKIWANIE CFSTRUCT			X	X
ODŚWIEŻ KLASTER		X	X	X
ODŚWIEŻ MENEDŻERA KOLEJEK		X	X	X
REFRESH SECURITY		X	X	X
Resetuj kanał			X	X
Resetowanie klastra		X	X	X
ZRESETUJ QSTATS		X	X	X
RESETUJ POTOK TPIPE			X	X
Rozstrzyganie kanału			X	X
ROZSTRZYGNIJ WĄTPLIWOŚĆ		X	X	X
WZNÓW MENEDŻERA KOLEJEK			X	X
ZABEZPIECZENIA RVERIFY		X	X	X
USTAW ARCHIWUM	X	X	X	X
USTAW DZIENNIK	X	X	X	X
USTAW SYSTEM	X	X	X	X
KANAŁ POCZĄTKOWY			X	X
URUCHOM CHINIT		X	X	X
START CMDSERV	X	X	X	
Uruchom proces nastuchujący			X	X
START QMGR			X	
URUCHOMIENIE ŚLEDZENIA	X	X	X	X

Tabela 24. Źródła, z których można uruchamiać komendy MQSC (kontynuacja)

Komenda	CSQINP1	CSQINP2	konsola z/OS	Kolejka wejściowa komend i serwer
Zamknij kanał			X	X
ZATRZYMAJ CHINIT			X	X
STOP CMDSERV	X	X	X	
Zatrzymaj proces nasłuchujący			X	X
STOP QMGR			X	X
ZATRZYMAJ ŚLEDZENIE	X	X	X	X
Menedżer kolejki zawieszony			X	X

W komendach MQSC każdy opis komendy identyfikuje źródła, z których można uruchomić komendę.

Komendy inicjowania dla IBM MQ for z/OS

Komendy inicjowania mogą być używane do sterowania uruchamianiem menedżera kolejek.

Komendy w wejściowych zestawach danych inicjowania są przetwarzane, gdy produkt IBM MQ jest inicjowany podczas uruchamiania menedżera kolejek. Z wejściowych zestawów danych inicjowania można wydać trzy typy komend:

- Komendy służące do definiowania obiektów produktu IBM MQ, które nie mogą być definiowane w innym miejscu, na przykład DEFINE BUFFPOOL.

Te komendy muszą znajdować się w zestawie danych identyfikowany przez nazwę DD CSQINP1. Są one przetwarzane przed fazą restartowania inicjowania. Nie mogą być one wystawiane za pomocą konsoli, paneli operacyjno-sterujących lub aplikacji. Odpowiedzi na te komendy są zapisywane do sekwencyjnego zestawu danych, do którego odwołuje się instrukcja CSQOUT1 procedury uruchomionego zadania.

- Komendy służące do definiowania obiektów IBM MQ, które są odtwarzalne po restarcie. Te definicje muszą być określone w zestawie danych identyfikowany przez nazwę DD CSQINP2. Są one zapisywane w zestawie stron zero. CSQINP2 jest przetwarzany po fazie restartu inicjowania. Odpowiedzi na te komendy są zapisywane do sekwencyjnego zestawu danych, do którego odwołuje się instrukcja CSQOUT2 procedury uruchomionego zadania.
- Komendy służące do manipulowania obiektami IBM MQ. Te komendy muszą być również określone w zestawie danych identyfikowany przez nazwę DD CSQINP2. Na przykład przykładowa komenda IBM MQ zawiera instrukcję ALTER QMGR w celu określenia kolejki niedostarczonych komunikatów dla podsystemu. Odpowiedź na te komendy jest zapisywana w wyjściowym zestawie danych CSQOUT2.

Uwaga: Jeśli obiekty IBM MQ są zdefiniowane w CSQINP2, program IBM MQ próbuje ponownie zdefiniować je za każdym razem, gdy uruchamiany jest menedżer kolejek. Jeśli obiekty już istnieją, próba ich zdefiniowania nie powiedzie się. Jeśli zachodzi potrzeba zdefiniowania obiektów w CSQINP2, można uniknąć tego problemu, używając parametru REPLACE w komendach DEFINE, jednak nadpisuje to wszystkie zmiany wprowadzone podczas poprzedniego uruchomienia menedżera kolejek.

Przykładowe elementy zestawu danych inicjowania są dostarczane wraz z produktem IBM MQ for z/OS. Są one opisane w sekcji [Przykładowe definicje dostarczane wraz z produktem IBM MQ](#).

Komendy inicjowania dla rozproszonego kolejkowania

Dla komendy START CHINIT można również użyć zestawu danych inicjowania CSQINP2 . Jeśli potrzebna jest seria innych komend do zdefiniowania rozproszonego środowiska kolejkwania (na przykład uruchamianie programów nasłuchujących), program IBM MQ udostępnia trzeci zestaw danych wejściowych inicjowania o nazwie CSQINPX, który jest przetwarzany jako część procedury uruchomionej zadania inicjatora kanału.

Komendy MQSC zawarte w zestawie danych są wykonywane na końcu inicjowania inicjatora kanału, a dane wyjściowe są zapisywane w zestawie danych określonym przez instrukcję CSQOUTX DD. Zestaw danych inicjowania CSQINPX może być używany do uruchamiania programów nasłuchujących na przykład.

Przykładowy element zestawu danych inicjowania inicjatora kanału jest dostarczany razem z produktem IBM MQ for z/OS. Jest on opisany w sekcji [Przykładowe definicje dostarczane wraz z produktem IBM MQ](#).

Komendy inicjowania dla publikowania/subskrypcji

Jeśli wymagana jest seria komend do zdefiniowania środowiska publikowania/subskrypcji (na przykład podczas definiowania subskrypcji), produkt IBM MQ udostępnia czwarty zestaw danych wejściowych inicjowania o nazwie CSQINPT.

Komendy MQSC zawarte w zestawie danych są wykonywane po zakończeniu inicjowania publikowania/subskrypcji, a dane wyjściowe są zapisywane w zestawie danych określonym przez instrukcję CSQOUTT DD. Zestaw danych inicjowania CSQINPT może być używany do definiowania subskrypcji.

Przykładowy element zestawu danych inicjowania publikowania/subskrypcji jest dostarczany z produktem IBM MQ for z/OS. Jest on opisany w sekcji [Przykładowe definicje dostarczane wraz z produktem IBM MQ](#).

z/OS

Programy narzędziowe serwera IBM MQ for z/OS

Produkt IBM MQ for z/OS udostępnia zestaw programów narzędziowych, które mogą być pomocne w administrowaniu systemem.

Produkt IBM MQ for z/OS udostępnia zestaw programów narzędziowych ułatwiających wykonywanie różnych zadań administracyjnych, w tym:

- Zarządzanie strategiami bezpieczeństwa komunikatów.
- Wykonaj zadania tworzenia kopii zapasowej, odtwarzania i reorganizacji.
- Wydawanie komend i definicji obiektów procesu.
- Generowanie wyjść konwersji danych.
- Zmodyfikuj zestaw danych programu startowego.
- Wyświetla informacje na temat dzienników.
- Wydrukuj dzienniki.
- Skonfiguruj tabele produktu Db2 i inne programy narzędziowe produktu Db2 .
- Przetwarzaj komunikaty w kolejce niedostarczonych komunikatów.

Program narzędziowy strategii bezpieczeństwa komunikatów

Program narzędziowy strategii bezpieczeństwa komunikatów (CSQOUTIL) jest uruchamiany jako autonomiczny program narzędziowy do zarządzania strategiami bezpieczeństwa komunikatów. Więcej informacji na ten temat zawiera sekcja [Program narzędziowy strategii bezpieczeństwa komunikatów \(CSQOUTIL\)](#) .

Program narzędziowy CSQUTIL

Jest to program narzędziowy służący do tworzenia i odtwarzania kopii zapasowych, a także do reorganizacji zadań. Więcej informacji na ten temat zawiera sekcja [Program narzędziowy CSQUTIL](#) .

Program narzędziowy do obsługi wyjścia konwersji danych

Program narzędziowy do obsługi wyjścia konwersji danych produktu IBM MQ for z/OS (**CSQUCVX**) jest uruchamiany jako autonomiczny program narzędziowy do tworzenia procedur wyjścia konwersji danych.

Program narzędziowy do spisu zasobów dziennika zmian

Program narzędziowy do spisywania zasobów dziennika zmian w programie IBM MQ for z/OS (**CSQJU003**) jest uruchamiany jako autonomiczny program narzędziowy do zmiany zestawu danych programu startowego (BSDS). Za pomocą tego programu narzędziowego można wykonywać następujące funkcje:

- Dodaj lub usuń aktywne lub archiwalne zestawy danych dziennika.
- Podaj hasła dla dzienników archiwalnych.

Program narzędziowy do odwzorowywania dziennika drukowania

Program narzędziowy do odwzorowywania dziennika drukowania IBM MQ for z/OS (**CSQJU004**) jest uruchamiany jako autonomiczny program narzędziowy w celu wyświetlenia następujących informacji:

- Nazwa zestawu danych dziennika i powiązanie RBA dziennika dla obu kopii wszystkich aktywnych i archiwalnych zestawów danych dziennika. Jeśli podwójne rejestrowanie nie jest aktywne, istnieje tylko jedna kopia zestawów danych.
- Aktywne zestawy danych dziennika dostępne dla nowych danych dziennika.
- Zawartość kolejki rekordów punktów kontrolnych w zestawie danych programu startowego (BSDS).
- Zawartość rekordu historii komendy dziennika archiwalnego.
- Znaczniki czasu systemu i programu narzędziowego.

Program narzędziowy do drukowania dziennika

Program narzędziowy do drukowania dziennika (**CSQ1LOGP**) jest uruchamiany jako autonomiczny program narzędziowy. Można uruchomić program narzędziowy, określając:

- Zestaw danych programu startowego (BSDS)
- Aktywne dzienniki (bez BSDS)
- Dzienniki archiwalne (bez BSDS)

Program narzędziowy grupy współużytkowania kolejki

Program narzędziowy grupy współużytkowania kolejki (**CSQ5PQSG**) jest uruchamiany jako autonomiczny program narzędziowy do konfigurowania tabel produktu Db2 i wykonywania innych zadań produktu Db2 wymaganych dla grup współużytkowania kolejek.

Program narzędziowy do preformatowania aktywnego dziennika

Program narzędziowy do preformatowania aktywnego dziennika (**CSQJUFMT**) formatuje aktywne zestawy danych dziennika, zanim zostaną one użyte przez menedżera kolejek. Jeśli zestawy danych aktywnego dziennika są wstępnie sformatowane przez program narzędziowy, wydajność zapisu w dzienniku zostanie poprawiona w pierwszym przejściu menedżera kolejek za pomocą aktywnych dzienników.

Program narzędziowy do obsługi kolejki niedostarczonych komunikatów

Program narzędziowy do obsługi niedostarczonych komunikatów w kolejce (**CSQUDLQH**) jest uruchamiany jako autonomiczny program narzędziowy. Sprawdza on komunikaty, które znajdują się w kolejce niedostarczonych komunikatów i przetwarza je zgodnie z zestawem reguł, które są dostępne w programie narzędziowym.

CSQUTIL, program narzędziowy dla IBM MQ for z/OS

Program narzędziowy CSQUTIL jest dostarczany razem z programem IBM MQ for z/OS , który ułatwia wykonywanie zadań związanych z tworzeniem i odtwarzaniem kopii zapasowych, odtwarzaniem i reorganizacją oraz do wydawania komend i definicji obiektów procesu.

Więcej informacji na temat programu narzędziowego CSQUTIL znajduje się w sekcji [Program narzędziowy IBM MQ \(CSQUTIL\)](#). Korzystając z tego programu narzędziowego, można wywołać następujące funkcje:

KOMENDA

Wydawanie komend MQSC, rejestrowanie definicji obiektów i tworzenie plików definicji kanału klienta.

COPY

Aby odczytać zawartość nazwanej kolejki komunikatów IBM MQ for z/OS lub zawartość wszystkich kolejek nazwanego zestawu stron, należy umieścić je w pliku sekwencyjnym i zachować oryginalną kolejkę.

STRONA KOPII

Kopiowanie całych zestawów stron do większych zbiorów stron.

PUSTE

Aby usunąć zawartość nazwanej kolejki komunikatów IBM MQ for z/OS lub zawartości wszystkich kolejek nazwanego zestawu stron, zachowując definicje kolejek.

Formatowanie

Aby sformatować zestawy stron IBM MQ for z/OS .

ŁADOWANIE

Odtworzenie zawartości nazwanej kolejki komunikatów IBM MQ for z/OS lub zawartości wszystkich kolejek nazwanej strony ustawionej na podstawie pliku sekwencyjnego utworzonego przez funkcję COPY.

PAGEINFO

Aby wyodrębnić informacje o zestawie stron z jednego lub większej liczby zestawów stron.

RESETPAGE

Aby skopiować wszystkie zestawy stron do innych zestawów danych zestawu stron, a następnie wyzeruj informacje dziennika w kopii.

KOPIUJ

Kopiowanie zawartości kolejki do zestawu danych w czasie, gdy menedżer kolejek jest w trybie bez połączenia.

SDEFS

Aby utworzyć zestaw komend definiujących obiekty w czasie, gdy menedżer kolejek jest w trybie bez połączenia.

ŁADUJ

Aby odtworzyć komunikaty z docelowego zestawu danych dla wcześniejszej operacji COPY lub SCOPY. SLOAD przetwarza pojedynczą kolejkę.

Włącznik/wyłącznik

Aby przetestować lub wystąpić zapytanie do kolejki transmisji powiązanej z kanałami nadawczym klastra.

XPARM

Aby przekształcić moduł ładujący parametr inicjatora kanału w atrybuty menedżera kolejek (dla celów migracji).

Operacyjny IBM MQ for z/OS

Te podstawowe procedury umożliwiają działanie produktu IBM MQ for z/OS.

You can also perform the operations described in this section using the IBM MQ Explorer, which is distributed with IBM MQ for Windows, IBM MQ for Linux (x86 and x86-64 platforms) and SupportPac MS0T. Aby uzyskać więcej informacji, patrz [“Administrowanie przy użyciu IBM MQ Explorer”](#) na stronie 134 i IBM Support & downloads.

Ta sekcja zawiera informacje na temat następujących tematów:

Wydawanie komend menedżera kolejek w systemie z/OS

Komendy sterujące produktu IBM MQ można wydawać z poziomu konsoli produktu z/OS lub z programu narzędziowego CSQUTIL. Komendy mogą używać przedrostka komendy (command prefix string-CPF) w celu wskazania, który podsystem IBM MQ przetwarza komendę.

Większość środowiska operacyjnego produktu IBM MQ można sterować za pomocą komend IBM MQ . Produkt IBM MQ for z/OS obsługuje zarówno typy komend MQSC, jak i PCF tych komend. W tym temacie opisano sposób określania atrybutów za pomocą komend MQSC, a więc odnosi się do tych komend i atrybutów przy użyciu ich nazw komend MQSC, a nie ich nazw PCF. Szczegółowe informacje na temat składni komend MQSC można znaleźć w sekcji [Komendy MQSC](#). Szczegółowe informacje na temat składni komend PCF można znaleźć w sekcji [“Korzystanie z formatów komend programowalnych IBM MQ”](#) na stronie 23. Użytkownik, który jest odpowiednio autoryzowanym użytkownikiem, może wydawać komendy IBM MQ z następujących:

- Wejściowe zestawy danych inicjowania (opisane w sekcji [“Komendy inicjowania dla IBM MQ for z/OS”](#) na stronie 337).
- Konsola z/OS lub odpowiednik, taki jak SDSF
- Procedura komendy get dla systemu głównego z/OS , MGCRC (SVC 34)
- Program narzędziowy IBM MQ CSQUTIL (opisany w sekcji [Program narzędziowy IBM MQ.](#))
- Aplikacja użytkownika, która może być:
 - Program CICS
 - Program TSO
 - Program wsadowy z/OS
 - Program IMS

Więcej informacji na ten temat zawiera sekcja [“Pisanie programów do administrowania programem IBM MQ for z/OS”](#) na stronie 362 .

Większość funkcji tych komend jest udostępniana w wygodny sposób przez operacje i panele kontrolne, dostępne z poziomu TSO i ISPF, a także opisane w sekcji [“Operations and control panels for IBM MQ for z/OS”](#) na stronie 347.

Więcej informacji na ten temat zawiera sekcja

- [“Wydawanie komend z konsoli programu z/OS lub jej odpowiedników”](#) na stronie 341
 - [Łańcuchy przedrostków komend](#)
 - [Korzystanie z konsoli z/OS do wydawania komend](#)
 - [Odpowiedzi komend](#)
- [Wydawanie komend z programu narzędziowego CSQUTIL](#)

Wydawanie komend z konsoli programu z/OS lub jej odpowiedników

Wszystkie komendy produktu IBM MQ można wprowadzić z poziomu konsoli produktu z/OS lub jej odpowiednika. Komendy IBM MQ można również wydawać z dowolnego miejsca, w którym można wydawać komendy produktu z/OS , takie jak SDSF lub przez program przy użyciu makra MGCRC.

Maksymalna ilość danych, które mogą być wyświetlane w wyniku komendy wpisanej w konsoli, wynosi 32 kB.

Uwaga:

1. Komendy IBM MQ nie można wydawać za pomocą formatu komend IMS/SSR z terminalu IMS . Ta funkcja nie jest obsługiwana przez adapter IMS .
2. Pole wejściowe udostępniane przez SDSF może nie być wystarczająco długie dla niektórych komend, w szczególności dla tych, które są dostępne dla kanałów.

Łańcuchy przedrostka komendy

Każda komenda IBM MQ musi być poprzedzona łańcuchem przedrostka komendy (CPF), jak to pokazano na [Rysunek 41](#) na stronie 342.

Ponieważ w systemie z/OS może działać więcej niż jeden podsystem IBM MQ , CPF jest używany do wskazania, który podsystem IBM MQ przetwarza komendę. Na przykład, aby uruchomić menedżer kolejek dla podsystemu o nazwie CSQ1, gdzie CPF to ' +CSQ1 ', należy wydać komendę +CSQ1 START QMGR z konsoli operatora. Ten CPF musi być zdefiniowany w tabeli nazw podsystemów (dla podsystemu CSQ1). Jest to opisane w sekcji [Definiowanie łańcuchów przedrostka komendy \(CPF\)](#). W przykładach łańcuch " +CSQ1 " jest używany jako przedrostek komendy.

Używanie konsoli z/OS do wydawania komend

Można wpisać proste komendy z konsoli z/OS , na przykład komendę DISPLAY w programie [Rysunek 41](#) na stronie 342. Jednak w przypadku złożonych komend lub w przypadku zestawów komend często używane są inne metody wydawania komend.

```
+CSQ1 DISPLAY QUEUE(TRANSMIT.QUEUE.PROD) TYPE(QLLOCAL)
```

Rysunek 41. Wydawanie komendy DISPLAY z poziomu konsoli produktu z/OS

Odpowiedzi komend

Bezpośrednie odpowiedzi na komendy są wysyłane do konsoli, która wydała komendę. Produkt IBM MQ obsługuje funkcję *Extended Console Support* (EMCS) dostępną w produkcie z/OS, a więc można użyć konsoli z 4-bajtowymi identyfikatorami. Ponadto wszystkie komendy z wyjątkiem START QMGR i STOP QMGR obsługują użycie komend i znaczników odpowiedzi (CARTs), gdy komenda jest uruchamiana przez program przy użyciu makra MGCRC.

Wydawanie komend z programu narzędziowego CSQUTIL

Komendy można wydawać z sekwencyjnego zestawu danych, korzystając z funkcji COMMAND programu narzędziowego CSQUTIL. Ten program narzędziowy przesyła komendy, jako komunikaty, do *systemowej kolejki komend* i czeka na odpowiedź, która jest drukowana wraz z oryginalnymi komendami w SYSPRINT. Szczegółowe informacje na ten temat zawiera sekcja [Program narzędziowy IBM MQ](#).

Uruchamianie i zatrzymywanie menedżera kolejek w systemie z/OS

Ten temat zawiera wprowadzenie do zatrzymywania i uruchamiania menedżera kolejek.

W tej sekcji opisano sposób uruchamiania i zatrzymywania menedżera kolejek. Zawiera on informacje na temat następujących tematów:

- [“Zanim rozpocznie IBM MQ” na stronie 343](#)

- [“Uruchamianie menedżera kolejek” na stronie 343](#)
- [“Zatrzymywanie menedżera kolejek” na stronie 345](#)

Uruchamianie i zatrzymywanie menedżera kolejek jest stosunkowo proste. Gdy menedżer kolejek zostanie zatrzymany w normalnych warunkach, jego ostatnią czynnością jest wykonanie punktu kontrolnego zakończenia. Ten punkt kontrolny i dzienniki dają menedżerowi kolejek informacje, które muszą zostać zrestartowane.

Ta sekcja zawiera informacje na temat komend START i STOP i zawiera krótki przegląd uruchamiania po wystąpieniu nieprawidłowego zakończenia.

Zanim rozpocznieš IBM MQ

Po zainstalowaniu produktu IBM MQ jest on zdefiniowany jako formalny podsystem z/OS . Ten komunikat jest wyświetlany podczas ładowania programu początkowego (IPL) programu z/OS:

```
CSQ3110I +CSQ1 CSQ3UR00 - SUBSYSTEM ssnm INITIALIZATION COMPLETE
```

gdzie *ssnm* jest nazwą podsystemu IBM MQ .

Od tej chwili można uruchomić menedżer kolejek dla tego podsystemu z dowolnej konsoli z/OS , która została autoryzowana do wydawania komend sterowania systemem ; Oznacza to, że jest to grupa komend SYS produktu z/OS . Komendę START należy wydać z poziomu autoryzowanej konsoli, nie można jej wydać za pomocą JES lub TSO.

Jeśli używane są grupy współużytkowania kolejek, przed uruchomieniem menedżera kolejek należy najpierw uruchomić usługi RRS, a następnie produkt Db2.

Uruchamianie menedżera kolejek

Menedżer kolejek uruchamia się, wydając komendę START QMGR. Nie można jednak pomyślnie użyć komendy START, chyba że użytkownik ma odpowiednie uprawnienia. Informacje o zabezpieczeniach produktu IBM MQ można znaleźć w sekcji Konfigurowanie zabezpieczeń w systemie z/OS . Rysunek 42 na stronie 343 przedstawia przykłady komendy START. (Należy pamiętać, że należy poprzedzić komendę IBM MQ z przedrostkiem komendy (CPF).)

```
+CSQ1 START QMGR
+CSQ1 START QMGR PARM(NEWLOG)
```

Rysunek 42. Uruchamianie menedżera kolejek z konsoli produktu z/OS

Więcej informacji na temat składni komendy START QMGR zawiera sekcja [START QMGR](#) .

Nie można uruchomić menedżera kolejek jako zadania wsadowego lub uruchomić go za pomocą komendy z/OS START. Te metody prawdopodobnie uruchamiają przestrzeń adresową dla IBM MQ , która kończy się nieprawidłowo. Nie można również uruchomić menedżera kolejek z programu narzędziowego CSQUTIL lub podobnej aplikacji użytkownika.

Można jednak uruchomić menedżer kolejek z autoryzowanego programu APF, przekazując komendę START QMGR do usługi z/OS MGCRC (SVC 34).

Jeśli używane są grupy współużytkowania kolejek, powiązane systemy Db2 i RRS muszą być aktywne podczas uruchamiania menedżera kolejek.

Opcje uruchamiania

Po uruchomieniu menedżera kolejek ładowany jest moduł parametrów systemowych. Nazwę modułu parametru systemowego można określić na jeden z dwóch sposobów:

- W przypadku parametru PARM komendy /cpf START QMGR, na przykład

```
/cpf START QMGR PARM(CSQ1ZPRM)
```

- Za pomocą parametru w procedurze uruchamiania, na przykład, kod instrukcji JCL EXEC jako

```
//MQM EXEC PGM=CSQYASCP,PARM='ZPARAM(CSQ1ZPRM)'
```

Moduł parametrów systemowych udostępnia informacje określone podczas dostosowywania menedżera kolejek.

V 9.0.3 W produkcie IBM MQ 9.0.3 można użyć opcji **QMGRPROD**, aby określić produkt, dla którego ma być rejestrowana składnia menedżera kolejek, oraz opcję **AMSPROD**, aby określić odpowiednik opcji AMS (jeśli jest to używane). Szczegółowe informacje na temat dozwolonych wartości można znaleźć w komendzie MQSC [START QMGR](#).

V 9.0.3 Poniżej przedstawiono przykładową instrukcję JCL EXEC:

```
//MQM EXEC PGM=CSQYASCP,PARM='QMGRPROD(MQ)'
```

Więcej informacji na temat rejestrowania użycia produktu zawiera publikacja [z/OS MVS Product Management](#).

Można również użyć opcji ENVPARM, aby zastąpić jeden lub większą liczbę parametrów w procedurze JCL dla menedżera kolejek.

Na przykład można zaktualizować procedurę uruchamiania menedżera kolejek, tak aby zmienna DDname CSQINP2 była zmienną. Oznacza to, że można zmienić nazwę DDname CSQINP2 bez zmiany procedury uruchamiania. Jest to przydatne podczas implementowania zmian, udostępniania wycofań dla operatorów oraz operacji menedżera kolejek.

Założmy, że procedura uruchamiania dla menedżera kolejek CSQ1 wyglądała jak [Rysunek 43](#) na stronie 344.

```
//CSQ1MSTR PROC INP2=NORM
//MQMESA EXEC PGM=CSQYASCP
//STEPLIB DD DISP=SHR,DSN=thlqual.SCSQANLE
// DD DISP=SHR,DSN=thlqual.SCSQAUTH
// DD DISP=SHR,DSN=db2qual.SDSNLOAD
//BSDS1 DD DISP=SHR,DSN=myqual.BSDS01
//BSDS2 DD DISP=SHR,DSN=myqual.BSDS02
//CSQP0000 DD DISP=SHR,DSN=myqual.PSID00
//CSQP0001 DD DISP=SHR,DSN=myqual.PSID01
//CSQP0002 DD DISP=SHR,DSN=myqual.PSID02
//CSQP0003 DD DISP=SHR,DSN=myqual.PSID03
//CSQINP1 DD DISP=SHR,DSN=myqual.CSQINP(CSQ1INP1)
//CSQINP2 DD DISP=SHR,DSN=myqual.CSQINP(CSQ1&INP2.)
//CSQOUT1 DD SYSOUT=*
//CSQOUT2 DD SYSOUT=*
```

Rysunek 43. Przykładowa procedura rozruchu

Jeśli następnie menedżer kolejek zostanie uruchomiony za pomocą komendy:

```
+CSQ1 START QMGR
```


CSQINP2 jest elementem o nazwie CSQ1NORM.

Załóżmy jednak, że wprowadzisz nowy pakiet programów do produkcji, tak aby następnym razem, gdy zostanie uruchomiony menedżer kolejek CSQ1, zostaną pobrane definicje CSQINP2 z elementu CSQ1NEW. Aby to zrobić, należy uruchomić menedżer kolejek za pomocą następującej komendy:

```
+CSQ1 START QMGR ENVPARM('INP2=NEW')
```

i zamiast CSQ1NORM zostanie użyta wartość CSQ1NEW. Uwaga: z/OS ogranicza specyfikacje KEYWORD=value dla parametrów symbolicznych (tak jak w INP2=NEW) do 255 znaków.

Rozpoczęcie po nieprawidłowym zakończeniu

Program IBM MQ automatycznie wykrywa, czy restart następuje po normalnym zamknięciu systemu, czy też nieprawidłowym zakończeniu.

Uruchomienie menedżera kolejek po nieprawidłowym zakończeniu jest inne niż uruchomienie komendy po wydaniu komendy STOP QMGR. Po zatrzymaniu QMGR system kończy pracę w uporządkowany sposób i przed zatrzymaniem zajmuje punkt kontrolny zakończenia. Po zrestartowaniu menedżera kolejek używa on informacji z systemowego punktu kontrolnego i dziennika odtwarzania w celu określenia statusu systemu podczas jego zamknięcia.

Jeśli jednak menedżer kolejek zakończy działanie w sposób nieprawidłowy, kończy działanie bez możliwości zakończenia pracy lub wykonania punktu kontrolnego zakończenia. Po zrestartowaniu menedżera kolejek następuje odświeżenie wiedzy o jej statusie podczas kończenia pracy przy użyciu informacji w dzienniku i powiadomienie użytkownika o statusie różnych zadań. Zwykle proces restartowania rozwiązuje wszystkie niespójne stany. Jednak w niektórych przypadkach konieczne jest podjęcie konkretnych kroków w celu usunięcia niespójności.

Komunikaty użytkownika podczas uruchamiania

Po pomyślnym uruchomieniu menedżera kolejek menedżer kolejek tworzy zestaw komunikatów uruchamiania.

Zatrzymywanie menedżera kolejek

Przed zatrzymaniem menedżera kolejek wszystkie komunikaty związane z zapisami (WTOR) związane z produktem IBM MQ muszą odbierać odpowiedzi, na przykład w celu uzyskania żądań dziennika. Każda komenda w produkcie [Rysunek 44 na stronie 345](#) kończy działanie działającego menedżera kolejek.

```
+CSQ1 STOP QMGR
+CSQ1 STOP QMGR MODE(QUIESCE)
+CSQ1 STOP QMGR MODE(FORCE)
+CSQ1 STOP QMGR MODE(RESTART)
```

Rysunek 44. Zatrzymywanie menedżera kolejek

Wartością domyślną komendy STOP QMGR jest STOP QMGR MODE (QUIESCE).

W trybie QUIESCE program IBM MQ nie zezwala na tworzenie żadnych nowych wątków połączeń, ale umożliwia kontynuację istniejących wątków. kończy się on tylko wtedy, gdy wszystkie wątki zostały zakończone. Aplikacje mogą żądać powiadomienia w przypadku wygaszania menedżera kolejek. Dlatego w miarę możliwości należy użyć trybu QUIESCE, tak aby aplikacje, które zażądały powiadomienia, miały możliwość rozłączenia. Szczegółowe informacje zawiera sekcja [Co się dzieje podczas kończenia pracy](#).

Jeśli menedżer kolejek nie zakończy działania w rozsądnym czasie w odpowiedzi na komendę STOP QMGR MODE (QUIESCE), użyj komendy DISPLAY CONN, aby określić, czy istnieją wątki połączenia, i wykonaj niezbędne kroki, aby zakończyć powiązane aplikacje. Jeśli nie ma wątków, wydaj komendę STOP QMGR MODE (FORCE).

Komendy STOP QMGR MODE (QUIESCE) i STOP QMGR MODE (FORCE) wytuskiwać IBM MQ z programu MVS Automatic Restart Manager (ARM), uniemożliwiając ARM automatyczne restartowanie menedżera kolejek. Komenda STOP QMGR MODE (RESTART-STOP QMGR MODE) działa w ten sam sposób, co komenda STOP QMGR MODE (FORCE), z tym wyjątkiem, że nie dereguluje IBM MQ z ARM. Oznacza to, że menedżer kolejek kwalifikuje się do natychmiastowego automatycznego restartu.

Jeśli podsystem IBM MQ nie jest zarejestrowany w menedżerze ARM, komenda STOP QMGR MODE (RESTART) jest odrzucana, a do konsoli z/OS jest wysyłany następujący komunikat:

```
CSQY205I ARM element arm-element is not registered
```

Jeśli ten komunikat nie zostanie wyświetlony, menedżer kolejek zostanie zrestartowany automatycznie. Więcej informacji na temat ARM zawiera sekcja [“Korzystanie z programu z/OS Automatic Restart Manager \(ARM\)”](#) na stronie 424.

Jeśli parametr STOP QMGR MODE (FORCE) nie zakończy działania menedżera kolejek, należy anulować tylko przestrzeń adresową menedżera kolejek.

Jeśli menedżer kolejek jest zatrzymany przez anulowanie przestrzeni adresowej lub przez użycie komendy STOP QMGR MODE (FORCE), spójność jest utrzymywana z podłączonymi systemami CICS lub IMS . Resynchronizacja zasobów jest uruchamiana, gdy menedżer kolejek jest restartowany i jest zakończony w momencie nawiązania połączenia z systemem CICS lub IMS .

Uwaga: Po zatrzymaniu menedżera kolejek może zostać wyświetlony komunikat IEF352I . Produkt z/OS wysyła ten komunikat, jeśli wykryje, że nie można oznaczyć przestrzeni adresowej jako bezużytecznej, co spowodowałoby narażenie na integralność. Ten komunikat można zignorować.

Komunikaty zatrzymania

Po wydaniu komendy STOP QMGR, można pobrać komunikaty CSQY009I i CSQY002I, na przykład:

```
CSQY009I +CSQ1 ' STOP QMGR' COMMAND ACCEPTED FROM  
USER(userid), STOP MODE(FORCE)  
CSQY002I +CSQ1 QUEUE MANAGER STOPPING
```

Gdzie `userid` jest identyfikatorem użytkownika, który wydał komendę STOP QMGR, a parametr MODE zależy od tego, który został podany w komendzie.

Po pomyślnym zakończeniu działania komendy STOP na konsoli produktu z/OS wyświetlane są następujące komunikaty:

```
CSQ9022I +CSQ1 CSQYASCP ' STOP QMGR' NORMAL COMPLETION  
CSQ3104I +CSQ1 CSQ3EC0X - TERMINATION COMPLETE
```

Jeśli korzystasz z ARM i nie określiłeś MODE (RESTART), wyświetlany jest również następujący komunikat:

```
CSQY204I +CSQ1 ARM DEREGISTER for element arm-element type  
arm-element-type successful
```

Nie można zrestartować menedżera kolejek, dopóki nie zostanie wyświetlony następujący komunikat:

```
CSQ3100I +CSQ1 CSQ3EC0X - SUBSYSTEM ssnm READY FOR START COMMAND
```

z/OS

Operations and control panels for IBM MQ for z/OS

Za pomocą operacji IBM MQ i paneli sterujących można wykonywać zadania administracyjne w obiektach IBM MQ. Ten temat zawiera wprowadzenie do komend i paneli sterujących.

Panele te są używane do definiowania, wyświetlania, modyfikowania i usuwania obiektów IBM MQ. Użyj paneli do codziennej administracji oraz do wprowadzania małych zmian w obiektach. Jeśli konfigurowasz lub zmieniasz wiele obiektów, użyj funkcji COMMAND programu narzędziowego CSQUTIL.

Panele operatorskie i kontrolne obsługują elementy sterujące inicjatora kanału (na przykład w celu uruchomienia kanału lub nasłuchiwanie TCP/IP), w celu łączenia w klastry i dla zabezpieczeń. Umożliwiają one również wyświetlanie informacji o wątkach i wykorzystaniu zestawu stron.

Panele działają wysyłając komendy MQSC typu IBM MQ do menedżera kolejek za pomocą kolejki wejściowej komend systemowych.

Uwaga:

1. Panele kontrolne i kontrolne programu z/OS IBM MQ (CSQOREXX) mogą nie obsługiwać wszystkich nowych funkcji i parametrów dodanych począwszy od wersji 7. Na przykład nie ma paneli do bezpośredniej manipulacji obiektami tematów lub subskrypcjami.

Użycie jednego z następujących obsługiwanych mechanizmów pozwala na administrowanie definicjami publikowania/subskrypcji i innymi kontrolami systemowymi, które nie są bezpośrednio dostępne z innych paneli:

- a. IBM MQ Eksplorator
- b. konsola z/OS
- c. Komunikaty PCF (Programmable Command Format)
- d. Funkcja COMMAND CSQUTIL

Należy zauważyć, że ogólne działanie produktu **Command** na panelach CSQOREXX pozwala na wydanie dowolnej poprawnej komendy MQSC, w tym komend związanych z SMDs. Można użyć wszystkich komend, które są używane przez funkcję COMMAND dla problemów CSQUTIL.

2. Komendy IBM MQ nie można wydawać bezpośrednio z poziomu wiersza komend w panelach.
3. Aby można było używać operacji i paneli sterujących, należy mieć poprawne uprawnienia zabezpieczeń. Jest to opisane w sekcji [Identyfikatory użytkowników dla zabezpieczeń komend i zabezpieczeń zasobów komend](#).
4. Nie można podać ID użytkownika i hasła przy użyciu CSQUTIL lub paneli CSQOREXX. Zamiast tego, jeśli ID użytkownika ma uprawnienie UPDATE do profilu BATCH w MQCONN, można pominąć ustawienie **CHKLOCL**(*REQUIRED*). Więcej informacji na ten temat zawiera sekcja [Korzystanie z produktu CHKLOCL w przypadku aplikacji powiązanych lokalnie](#).

z/OS

Wywołanie i reguły dla operacji i paneli sterujących

Za pomocą paneli ISPF można sterować IBM MQ i wydawać komendy sterujące.

Sposób uzyskiwania dostępu do paneli kontrolnych i paneli sterujących produktu IBM MQ

Jeśli menu podstawowych opcji ISPF/PDF zostało zaktualizowane dla produktu IBM MQ, można uzyskać dostęp do operacji IBM MQ i paneli sterujących z tego menu. Szczegółowe informacje na temat aktualizowania menu zawiera sekcja [Czynność 20: Konfigurowanie operacji i paneli kontrolnych](#).

Dostęp do operacji IBM MQ i paneli sterujących można uzyskać z panelu procesora komend TSO (zwykle jest to opcja 6 w menu podstawowych opcji ISPF/PDF). Nazwa komendy `exec`, która ma być uruchamiana, to `CSQOREXX`. Ma on dwa parametry; `th1qual` to kwalifikator wysokiego poziomu dla używanych bibliotek IBM MQ, a `lang1letter` to litera identyfikująca biblioteki języków narodowych, które mają być używane (na przykład E dla U.S. angielski). Parametry te można pominąć, jeśli biblioteki produktu IBM MQ są zainstalowane na stałe w konfiguracji ISPF. Alternatywnie można wydać komendę `CSQOREXX` z wiersza komend TSO.

Panele te są przeznaczone do stosowania przez operatorów i administratorów przy minimum formalnym szkoleniu. Zapoznaj się z tymi instrukcjami z uruchomionym panelem i wypróbuj inne sugerowane zadania.

Uwaga: Podczas korzystania z paneli tymczasowych kolejek dynamicznych z nazwami w postaci `SYSTEM.CSQOREXX.*` są tworzone.

Zasady dotyczące paneli operacyjno-sterujących

Więcej informacji na temat ogólnych reguł dotyczących łańcuchów znaków i nazw IBM MQ zawiera sekcja [Reguły nazewnictwa obiektów IBM MQ](#). Jednak istnieją pewne reguły, które dotyczą tylko operacji i paneli sterujących:

- Nie należy ujmować łańcuchów, na przykład opisów, w pojedynczych lub podwójnych cudzysłowach.
- Jeśli w polu tekstowym zostanie umieszczony apostrof lub znak cudzysłowu, nie trzeba go powtarzać ani dodać znaku zmiany znaczenia. Znaki są zapisywane dokładnie tak, jak je wpisujesz, na przykład:

```
This is Maria's queue
```

Procesor panelu podwaja je, aby przekazać je do programu IBM MQ. Jeśli jednak musi obciąć swoje dane, aby to zrobić, to robi to.

- W większości pól można używać wielkich lub małych liter, a po naciśnięciu klawisza `Enter` są one składowane wielkimi literami. Wyjątkami są:
 - Nazwy klas pamięci masowej i nazwy struktur narzędzia CF, które muszą zaczynać się od litery od A do Z, a następnie wielkie litery od A do Z lub znaki liczbowe.
 - Niektóre pola, które nie zostały przetłumaczone. Są to:
 - ID aplikacji
 - Opis
 - Dane środowiska
 - Nazwy obiektów (ale jeśli używana jest mała nazwa obiektu, można nie być w stanie wprowadzić jej w konsoli produktu z/OS)
 - Nazwa systemu zdalnego
 - Dane wyzwalacza
 - Dane użytkownika
- W nazwach, wiodące odstępy i wiodące znaki podkreślenia są ignorowane. Oznacza to, że nazwy obiektów nie mogą być rozpoczynające się od znaków odstępu ani znaków podkreślenia.
- Podkreślenia są używane w celu pokazania zakresu pustych pól. Po naciśnięciu klawisza `Enter` końcowe znaki podkreślenia są zastępowane odstępami.
- Wiele opisów i pól tekstowych jest prezentowanych w wielu częściach, a każda część jest obsługiwana przez produkt IBM MQ niezależnie. Oznacza to, że odstępy końcowe są zachowywane, a tekst nie jest ciągły.

Puste pola

Po określeniu działania **Definiuj** dla obiektu IBM MQ , każde pole na panelu definiowania zawiera wartość. Aby uzyskać informacje o tym, gdzie IBM MQ pobiera wartości, należy zapoznać się z pomocą ogólną (rozszerzona pomoc). Jeśli wpisujesz pole puste, a odstępy nie są dozwolone, program IBM MQ umieszcza domyślną wartość instalacji w polu lub wyświetli zachętę do wprowadzenia wymaganej wartości.

Po określeniu działania **Zmień** dla obiektu IBM MQ , każde pole na panelu zmiany zawiera bieżącą wartość dla tego pola. Jeśli wpisujesz pole zawierające spacje, a odstępy nie są dozwolone, wartość tego pola pozostaje niezmienną.

z/OS

Obiekty i działania w systemie z/OS

Panele operacyjno-kontrolne oferują wiele różnych typów obiektów oraz wiele działań, które można na nich wykonać.

Czynności te są wyświetlane na panelu początkowym i umożliwiają manipulowanie obiektami i wyświetlanie informacji o nich. Obiekty te obejmują wszystkie obiekty IBM MQ wraz z dodatkowymi obiektami. Obiekty zaliczają się do następujących kategorii.

- Kolejki, procesy, obiekty informacji uwierzytelniających, listy nazw, klasy pamięci masowej i struktury CF
- Kanały
- Obiekty klastra
- Menedżer kolejek i zabezpieczenia
- Połączenia
- SYSTEM

W sekcji Działania można znaleźć tabelę odniesień do działań, które można wykonać za pomocą obiektów IBM MQ .

Kolejki, procesy, obiekty informacji uwierzytelniających, listy nazw, klasy pamięci masowej i struktury CF

Są to podstawowe obiekty produktu IBM MQ . Każdy typ może mieć wiele. Można je wyświetlić, wymieniać z filtrem, definiować i usuwać, a także mieć atrybuty, które mogą być wyświetlane i modyfikowane, za pomocą listy lub ekranu, listy z filtrami, definiowania działań LIKE, MANAGE i ALTER. (Obiekty są usuwane za pomocą czynności MANAGE).

Ta kategoria składa się z następujących obiektów:

QLOCAL	Kolejka lokalna
QREMOTE	Kolejka zdalna
QALIAS	Kolejka aliasowa dla pośredniego odwołania do kolejki
QMODEL	Kolejka modelowa do dynamicznego definiowania kolejek
QUEUE	Dowolny typ kolejki
QSTATUS	Status kolejki lokalnej
PROCES	Informacje o aplikacji, która ma zostać uruchomiona, gdy wystąpi zdarzenie wyzwajające
AUTHINFO	Informacje o uwierzytelnianiu: definicje wymagane do wykonania sprawdzania listy odwołań certyfikatów (CRL) przy użyciu serwerów LDAP
NAMELIST	Lista nazw, takich jak kolejki lub klastry
STGCLASS	Klasa pamięci masowej
CFSTRUCT	struktura CF (Coupling Facility)

STATUS	Status struktury CF
CFSTATUS	

Kanały

Kanały są używane do rozproszonego kolejkowania. Może istnieć wiele spośród każdego typu i można je wyświetlić na liście z filtrem, zdefiniowanym, usuniętym, wyświetlanym i zmienionym. Mają również inne funkcje dostępne za pomocą działań START, STOP i PERFORM. Funkcja PERFORM umożliwia resetowanie, wykonywanie komendy ping i rozwiązywanie problemów z kanałami.

Ta kategoria składa się z następujących obiektów:

CHANNEL	Dowolny typ kanału
Nadawca	Kanał nadawcy
SERVER	Kanał serwera
Odbiorca	Kanał odbiorcy
Requester	Kanał requestera
CLUSRCVR	Kanał odbiorczy klastra
CLUSDR	Kanał wysyłający klastry
SVRCONN	Kanał połączenia serwera
CLNTCONN	Kanał połączenia klienckiego
CHSTATUS	Status połączenia kanału

Obiekty klastra

Obiekty klastra są tworzone automatycznie dla kolejek i kanałów, które należą do klastra. Definicje kolejek podstawowych i kanałów mogą znajdować się w innym menedżerze kolejek. Może istnieć wiele spośród każdego typu, a nazwy mogą być duplikowane. Mogą one być wyświetlane na liście, wyświetlane z filtrem i wyświetlane. Działania PERFORM, START i STOP są również dostępne w ramach akcji LIST.

Ta kategoria składa się z następujących obiektów:

CLUSQ	Kolejka klastra utworzona dla kolejki, która należy do klastra
CLUSCHL	Kanał klastra, utworzony dla kanału, który należy do klastra
CLUSQMgr	Menedżer kolejek klastra, taki sam, jak kanał klastra, ale identyfikowany przez jego nazwę menedżera kolejek

Kanały klastra i menedżery kolejek klastra mają działania PERFORM, START i STOP, ale tylko pośrednio za pomocą działania DISPLAY.

Menedżer kolejek i zabezpieczenia

Menedżer kolejek i obiekty zabezpieczeń mają jedną instancję. Mogą one być wyświetlane na liście i mają atrybuty, które mogą być wyświetlane i zmieniane (za pomocą czynności LIST lub DISPLAY oraz ALTER), a także mają inne funkcje dostępne przy użyciu działania PERFORM.

Ta kategoria składa się z następujących obiektów:

Menedżer	Menedżer kolejek: działanie PERFORM udostępnia funkcje klastra zawieszania i wznawiania.
Zabezpieczenia	Funkcje zabezpieczeń: działanie PERFORM udostępnia funkcje odświeżania i ponownego sprawdzania

Połączenie

Połączenia mogą być wyświetlane na liście, wyświetlane z filtrem i wyświetlane.

Ta kategoria składa się tylko z obiektu połączenia, CONNECT.

System

Zbiór innych funkcji. Ta kategoria składa się z następujących obiektów:

SYSTEM	Funkcje systemowe
CONTROL	Synonim systemu

Dostępne są następujące funkcje:

LIST lub DISPLAY	Wyświetl informacje o wykorzystaniu grupy współużytkowania kolejek, kolejkowania rozproszonego, zestawu stron lub zestawu danych.
WYKONAJ	Odświeżanie lub resetowanie grupowania
START	Uruchamianie inicjatora kanału lub programów nasłuchujących
STOP	Zatrzymaj inicjator kanału lub obiekty nasłuchiwanie

Działania

Działania, które można wykonać dla każdego typu obiektu, są przedstawione w poniższej tabeli:

Obiekt	Zmień	Zdefiniuj podobne	Zarządza nie (1)	Lista lub wyświetlanie	Lista z filtrem	Wykonaj	Uruchom	Zatrzymaj
AUTHINFO	X	X	X	X	X			
STATUS CFSTATUS				X				
CFSTRUCT	X	X	X	X	X			
CHANNEL	X	X	X	X	X	X	X	X
CHSTATUS				X	X			
CLNTCONN	X	X	X	X	X			
CLUSCHL				X	X	X (2)	X (2)	X (2)
CLUSQ				X	X			
CLUSQMGR				X	X	X (2)	X (2)	X (2)
CLUSRCVR	X	X	X	X	X	X	X	X
CLUSDR	X	X	X	X	X	X	X	X
CONNECT				X	X			
CONTROL				X		X	X	X
Menedżer	X			X		X		
NAMELIST	X	X	X	X	X			
PROCES	X	X	X	X	X			
QALIAS	X	X	X	X	X			

Tabela 25. Poprawne działania i działania panelu sterującego dla obiektów produktu IBM MQ (kontynuacja)

Obiekt	Zmień	Zdefiniuj podobne	Zarządza nie (1)	Lista lub wyświetlanie	Lista z filtrem	Wykonaj	Uruchom	Zatrzymaj
QLOCAL	X	X	X	X	X			
QMODEL	X	X	X	X	X			
QREMOTE	X	X	X	X	X			
QSTATUS				X	X			
QUEUE	X	X	X	X	X			
Odbiorca	X	X	X	X	X	X	X	X
Requester	X	X	X	X	X	X	X	X
Zabezpieczenia	X			X		X		
Nadawca	X	X	X	X	X	X	X	X
SERVER	X	X	X	X	X	X	X	X
SVRCONN	X	X	X	X	X		X	X
STGCLASS	X	X	X	X	X			
SYSTEM				X		X	X	X

Uwaga:

1. Udostępnia funkcje Delete i inne funkcje.
2. Korzystanie z czynności Wyświetl listę lub wyświetlanie

z/OS Dyspozycje obiektów w systemie z/OS

Użytkownik może określić *dyspozycję* obiektu, z którym będzie potrzebna praca. Dyspozycja oznacza miejsce, w którym zachowana jest **definicja** obiektu, oraz sposób zachowania obiektu.

Dyspozycja jest istotna tylko w przypadku, gdy użytkownik pracuje z dowolnym z następujących typów obiektów:

- kolejki
- kanały
- procesy
- Listy nazw
- Klasy pamięci masowej
- obiekty informacji uwierzytelniającej

Jeśli użytkownik pracuje z innymi typami obiektów, rozdysponowanie nie jest brane pod uwagę.

Dozwolone wartości to:

Q

QMGR. Definicje obiektów znajdują się w zestawie stron menedżera kolejek i są dostępne tylko przez menedżer kolejek.

C

KOPIUJ. Definicje obiektów znajdują się w zestawie stron menedżera kolejek i są dostępne tylko przez menedżer kolejek. Są to lokalne kopie obiektów zdefiniowanych jako mające dyspozycję GROUP.

P

PRYWATNE. Definicje obiektów znajdują się w zestawie stron menedżera kolejek i są dostępne tylko przez menedżer kolejek. Obiekty zostały zdefiniowane jako mające dyspozycję QMGR lub COPY.

G

GRUPA. Definicje obiektów znajdują się w repozytorium współużytkowanym i są dostępne dla wszystkich menedżerów kolejek w grupie współużytkowania kolejek.

S

SHARED. To rozporządzenie ma zastosowanie tylko do kolejek lokalnych. Definicje kolejek znajdują się we współużytkowanym repozytorium i są dostępne dla wszystkich menedżerów kolejek w grupie współużytkowania kolejek.

A

WSZYSTKIE. Jeśli menedżer kolejek działań jest docelowym menedżerem kolejek lub *, uwzględniane są obiekty **wszystkich**. W przeciwnym razie obiekty QMGR i COPY są uwzględniane tylko w tych obiektach. Jest to opcja domyślna.

z/OS Wybór menedżera kolejek, wartości domyślnych i poziomów przy użyciu panelu sterującego ISPF w systemie z/OS

Aby sterować menedżerami kolejek, można użyć komendy CSQOREXX exec w ISPF.

Podczas przeglądania panelu początkowego użytkownik nie jest połączony z żadnym menedżerem kolejek. Jednak po naciśnięciu klawisza Enter użytkownik jest połączony z menedżerem kolejek lub menedżerem kolejek w grupie współużytkowania kolejek o nazwie określonej w polu **Nazwa połączenia**. To pole można pozostawić puste; oznacza to, że użytkownik korzysta z domyślnego menedżera kolejek dla aplikacji wsadowych. Jest to zdefiniowane w CSQBDEFV (więcej informacji na ten temat zawiera [Task 19: Set up Batch, TSO i RRS adapters](#) (Zadanie 19: Konfigurowanie adapterów wsadowych, TSO i RRS)).

Pole **Docelowy menedżer kolejek** służy do określania menedżera kolejek, w którym mają być wykonywane działania. Jeśli to pole zostanie puste, zostanie ono domyślnie określone w menedżerze kolejek określonym w polu **Nazwa połączenia**. Istnieje możliwość określenia docelowego menedżera kolejek, z którym nie jest nawiązany połączenie. W takim przypadku zwykle należy podać nazwę obiektu menedżera kolejek zdalnych, który udostępnia definicję aliasu menedżera kolejek (nazwa ta jest używana jako nazwa *ObjectQMgrName* podczas otwierania kolejki wejściowej komend). Aby to zrobić, należy dysponować odpowiednimi kolejkami i kanałami ustawionym w celu uzyskania dostępu do zdalnego menedżera kolejek.

Pole **Menedżer kolejek działań** umożliwia określenie menedżera kolejek, który znajduje się w tej samej grupie współużytkowania kolejki, co menedżer kolejek określony w polu **Docelowy menedżer kolejek** jako menedżer kolejek, w którym mają zostać wykonane żądania. Jeśli w tym polu zostanie podana wartość *, działania, które zostaną wysłane, zostaną wykonane we wszystkich menedżerach kolejek w grupie współużytkowania kolejek. Jeśli to pole zostanie puste, wartością domyślną będzie wartość podana w polu **Docelowy menedżer kolejek**. Pole **Menedżer kolejek działań** odpowiada za pomocą modyfikatora komendy CMDSCOPE opisanego w sekcji [Komendy MQSC](#).

Wartości domyślne menedżera kolejek

Jeśli pozostajesz puste pola menedżera kolejek lub jeśli chcesz połączyć się z grupą współużytkowania kolejek, po naciśnięciu klawisza **Enter** zostanie otwarte okno dodatkowe. To okno służy do potwierdzania nazw menedżerów kolejek, które będą używane. Naciśnij klawisz **Enter**, aby kontynuować. Po powrocie do panelu początkowego po wykonaniu niektórych żądań znajdują się pola wypełnione rzeczywistymi nazwami.

Poziomy menedżera kolejek

Panele Operations and Control działają poprawnie tylko z menedżerami kolejek uruchomionym w systemie z/OS pod adresem IBM WebSphere MQ 710 lub nowszym.

Jeśli te warunki nie są spełnione, prawdopodobnie działania działają tylko częściowo, niepoprawnie lub w ogóle, oraz że odpowiedzi z menedżera kolejek nie są rozpoznawane.

Jeśli menedżer kolejek działań nie jest na poziomie lub wyższym niż IBM MQ 8.0.0, niektóre pola nie są wyświetlane, a niektóre wartości nie mogą zostać wprowadzone. Nieliczne obiekty i działania są niedozwolone. W takich przypadkach zostanie otwarte okno dodatkowe z prośbą o potwierdzenie, że chcesz kontynuować.

z/OS *Korzystanie z klawiszy funkcyjnych i wiersza komend przy użyciu paneli sterujących ISPF w systemie z/OS*

Aby korzystać z paneli, należy użyć klawiszy funkcyjnych lub wpisać odpowiednie komendy w obszarze komend panelu sterującego ISPF.

- klawisze funkcyjne
 - Przetwarzanie działań
 - “Wyświetlanie komunikatów użytkownika programu IBM MQ” na stronie 354
 - Anulowanie działań
 - Uzyskiwanie pomocy
- Za pomocą wiersza komend

Klawisze funkcyjne

Klawisze funkcyjne mają specjalne ustawienia dla IBM MQ. (Oznacza to, że nie można użyć wartości domyślnych ISPF dla klawiszy funkcyjnych; jeśli wcześniej użyto komendy KEYLIST OFF ISPF w dowolnym miejscu, należy wpisać KEYLIST ON w obszarze komend wszystkich operacji i panelu sterującego, a następnie nacisnąć klawisz Enter, aby włączyć ustawienia IBM MQ).

Te ustawienia klawiszy funkcyjnych mogą być wyświetlane na panelach, jak to pokazano na Rysunek 45 na stronie 355. Jeśli ustawienia nie są wyświetlane, wpisz PFSHOW w obszarze komend na dowolnym panelu operacji i panelu sterowania, a następnie naciśnij klawisz **Enter**. Aby usunąć wyświetlanie ustawień, należy użyć komendy PFSHOW OFF.

Ustawienia klawiszy funkcyjnych w panelach operacji i sterowania są zgodne z normami CUA. Chociaż można zmienić ustawienie klucza za pomocą zwykłych procedur ISPF (takich jak program narzędziowy **KEYLIST**), użytkownik nie jest zalecany do tego celu.

Uwaga: Użycie komend **PFSHOW** i **KEYLIST** ma wpływ na inne logiczne wyświetlane ekrany ISPF, a ich ustawienia pozostają w momencie opuszczania paneli operacyjno-sterujących.

Przetwarzanie działań

Naciśnij klawisz **Enter**, aby wykonać działanie żądane na panelu. Informacje z panelu są wysyłane do menedżera kolejek w celu przetworzenia.

Po każdym naciśnięciu klawisza **Enter** w panelach program IBM MQ generuje jeden lub większą liczbę komunikatów operatora. Jeśli operacja zakończyła się pomyślnie, zostanie wyświetlony komunikat z potwierdzeniem CSQ9022I, w przeciwnym razie zostaną wyświetlone komunikaty o błędach.

Wyświetlanie komunikatów użytkownika programu IBM MQ

Naciśnij klawisz funkcyjny F10 w dowolnym panelu, aby wyświetlić komunikaty użytkownika programu IBM MQ.

Anulowanie działań

Na panelu początkowym, zarówno F3, jak i F12, wyjdź z paneli operacji i sterowania, a następnie zwróć użytkownika do ISPF. Do menedżera kolejek nie są wysyłane żadne informacje.

W każdym innym panelu naciśnij klawisze funkcyjne F3 lub F12, aby opuścić bieżący panel **ignorowanie wpisanych danych od ostatniego naciśnięcia klawisza Enter**. Ponownie, żadne informacje nie są wysyłane do menedżera kolejek.

- Naciśnięcie klawisza F3 powoduje przejście z powrotem do panelu początkowego.
- F12 powoduje powrót do poprzedniego panelu.

Uzyskiwanie pomocy

Z każdym panelem są powiązane panele pomocy. Panele pomocy korzystają z protokołów ISPF:

- Naciśnij klawisz funkcyjny F1 na dowolnym panelu, aby wyświetlić pomoc ogólną (rozszerzoną pomoc) dotyczącą zadania.
- Naciśnij klawisz funkcyjny F1 , używając kursora w dowolnym polu, aby wyświetlić konkretną pomoc dotyczącą tego pola.
- Naciśnij klawisz funkcyjny F5 z dowolnego panelu pomocy w polu, aby uzyskać ogólną pomoc.
- Naciśnij klawisz funkcyjny F3 , aby powrócić do panelu podstawowego, czyli na panelu, z którego naciśnięto klawisz funkcyjny F1.
- Naciśnij klawisz funkcyjny F6 z dowolnego panelu pomocy, aby uzyskać pomoc na temat klawiszy funkcyjnych.

Jeśli informacje pomocnicze są zawarte na drugiej lub kolejnych stronach, w prawym górnym rogu panelu zostanie wyświetlony indykator **Więcej** . Użyj tych klawiszy funkcyjnych, aby poruszać się po stronach pomocy:

- F11 , aby przejść do następnej strony pomocy (jeśli jest dostępna).
- F10 , aby powrócić do poprzedniej strony pomocy (jeśli jest jedna).

Za pomocą wiersza komend

Nie ma potrzeby korzystania z wiersza komend do wydawania komend używanych przez operacje i panele kontrolne, ponieważ są one dostępne z poziomu klawiszy funkcyjnych. W wierszu komend można wprowadzić normalne komendy ISPF (np. **PFSHOW**).

W komendzie ISPF PANELID ON wyświetlana jest nazwa bieżącego panelu CSQOREXX.

Wiersz komend jest początkowo wyświetlany w pozycji domyślnej u dołu paneli, niezależnie od tego, jakie ustawienia ISPF mają. Aby zmienić pozycję wiersza komend, można użyć komendy SETTINGS ISPF z dowolnej operacji i paneli sterujących. Ustawienia są zapamiętywane dla kolejnych sesji z panelami operacyjnymi i kontrolnymi.

Korzystanie z paneli operacji i sterowania w systemie z/OS

Ten temat zawiera informacje na temat początkowego panelu sterującego wyświetlanego z CSQOREXX.

Rysunek 45 na stronie 355 przedstawia panel, który jest wyświetlany po uruchomieniu sesji panelu.

```
IBM MQ for z/OS - Main Menu
Complete fields. Then press Enter.
Action . . . . . 1      0. List with filter      4. Manage
                    1. List or Display      5. Perform
                    2. Define like        6. Start
                    3. Alter            7. Stop
                    8. Command
Object type . . . . . CHANNEL +
Name . . . . . *
Disposition . . . . . A  Q=Qmgr, C=Copy, P=Private, G=Group,
                        S=Shared, A=All
Connect name . . . . . MQ1C - local queue manager or group
Target queue manager . . . MQ1C
- connected or remote queue manager for command input
Action queue manager . . . MQ1C - command scope in group
Response wait time . . . . 30 5 - 999 seconds
(C) Copyright IBM Corporation 1993, 2023. All rights reserved.
Command ==>
F1=Help      F2=Split   F3=Exit     F4=Prompt   F9=SwapNext F10=Messages
F12=Cancel
```

Rysunek 45. Panel początkowy operacji i sterowania IBM MQ

Z poziomu tego panelu można wykonywać działania, takie jak:

- Wybierz lokalny menedżer kolejek oraz informacje o tym, czy komendy wydawane w tym menedżerze kolejek, w zdalnym menedżerze kolejek, czy w innym menedżerze kolejek mają być wykonywane w tej

samej grupie współużytkowania kolejki, co lokalny menedżer kolejek. W przypadku konieczności zmiany nazwy menedżera kolejek należy wpisać nazwę menedżera kolejek.

- Wybierz działanie, które ma zostać wykonane, wpisując odpowiednią liczbę w polu **Działanie** .
- Określ typ obiektu, z którym chcesz pracować. Naciśnij klawisz funkcyjny F1 , aby uzyskać pomoc dotyczącą typów obiektów, jeśli użytkownik nie jest pewny, czym są.
- Określ dyspozycję typu obiektu, z którym chcesz pracować.
- Wyświetl listę obiektów określonego typu. Wpisz gwiazdkę (*) w polu **Nazwa** i naciśnij klawisz **Enter** , aby wyświetlić listę obiektów (określonego typu), które zostały już zdefiniowane w menedżerze kolejek działań. Następnie można wybrać jeden lub więcej obiektów do pracy w sekwencji. Wszystkie działania są dostępne z listy.

Uwaga: Zalecane jest dokonanie wyboru, które spowoduje wyświetlenie listy obiektów, a następnie wykonanie prac z tej listy. Użyj działania **Wyświetl** , ponieważ jest to dozwolone dla wszystkich typów obiektów.

Korzystanie z narzędzia Command Facility w systemie z/OS

Użyj edytora, aby wprowadzić lub zmienić komendy MQSC, które mają być przekazywane do menedżera kolejek.

W panelu podstawowym CSQOPRIA wybierz opcję **8 Command**(Komenda 8), aby uruchomić narzędzie Command Facility.

Zostanie przedstawiona sesja edycji pliku sekwencyjnego *prefiks.CSQUTIL.COMMANDS*, używane jako dane wejściowe dla funkcji CSQUTIL COMMAND; patrz [Wydawanie komend do IBM MQ](#).

Przedrostek komendy (CPF) nie musi być poprzedzany przedrostkiem.

Można kontynuować wykonywanie komend MQSC w kolejnych wierszach, przeryjąc bieżący wiersz z następującymi znakami kontynuacji: + lub -. Alternatywnie można użyć trybu edycji linii w celu udostępnienia długich komend MQSC lub wartości długich wartości atrybutów w komendzie.

edycja wiersza

Aby użyć edycji linii, przesunij kursor na odpowiedni wiersz na panelu edycji i użyj klawisza **F4** , aby wyświetlić pojedynczy wiersz w panelu przewijalnym. Pojedyncza linia może zawierać do 32 760 bajtów danych.

Aby opuścić linię, edytuj:

- **Wyjście F3** powoduje zapisanie zmian wprowadzonych w wierszu i wyjściu.
- **F12 Anuluj** powoduje powrót do panelu edycji odrzucających zmiany wprowadzone w wierszu.

Aby odrzucić zmiany wprowadzone w sesji edycji, należy użyć komendy **F12 Anuluj** , aby zakończyć sesję edycji, pozostawiając zawartość pliku bez zmian. Komendy nie są wykonywane.

Wykonywanie komend

Po zakończeniu wprowadzania komend MQSC przerwij sesję edycji z programem **Wyjście F3** , aby zapisać zawartość pliku i wywołaj komendę CSQUTIL, aby przekazać te komendy do menedżera kolejek. Dane wyjściowe z przetwarzania komend są przechowywane w pliku *prefix.CSQUTIL.OUTPUT*. Sesja edycji zostanie automatycznie otwarta w tym pliku, aby można było wyświetlić odpowiedzi. Naciśnij klawisz **F3 exit** , aby wyjść z tej sesji i powrócić do menu głównego.

Praca z obiektami IBM MQ w systemie z/OS

Wiele z zadań opisanych w tej dokumentacji obejmuje manipulowanie obiektami IBM MQ . Typy obiektów to: menedżery kolejek, kolejki, definicje procesów, listy nazw, kanały, kanały połączeń klientów, obiekty nastuchiwania, usługi i obiekty informacji uwierzytelniających.

- [Definiowanie prostych obiektów kolejki](#)
- [Definiowanie innych typów obiektów](#)

- Praca z definicjami obiektów
- Praca z listami nazw

Definiowanie prostych obiektów kolejki

Aby zdefiniować nowy obiekt, należy użyć istniejącej definicji jako podstawy dla tego obiektu. Można to zrobić na jeden z trzech sposobów:

- Wybierając obiekt, który jest elementem listy wyświetlonej w wyniku opcji wybranych na panelu początkowym. Następnie należy wprowadzić typ działania 2 (**Definiuj podobne**) w polu działania obok wybranego obiektu. Nowy obiekt ma atrybuty wybranego obiektu, z wyjątkiem rozporządzenia. Następnie można zmienić dowolne atrybuty w nowym obiekcie w zależności od tego, który jest wymagany.
- Na panelu początkowym wybierz typ czynności **Definiuj podobne do** , wprowadź typ obiektu definiowanego w polu **Typ obiektu** i wprowadź nazwę konkretnego istniejącego obiektu w polu **Nazwa** . Nowy obiekt ma takie same atrybuty, jak nazwa obiektu, którego nazwa została określona w polu **Nazwa** , z wyjątkiem rozporządzenia. Następnie można zmienić dowolne atrybuty w nowej definicji obiektu zgodnie z tym, co jest wymagane.
- Po wybraniu typu działania **Definiuj jako** , określając typ obiektu, a następnie pozostawiając pole **Nazwa** puste. Następnie można zdefiniować nowy obiekt i ma on atrybuty domyślne zdefiniowane dla danej instalacji. Następnie można zmienić dowolne atrybuty w nowej definicji obiektu zgodnie z tym, co jest wymagane.

Uwaga: Nie wpisuje się nazwy obiektu definiowanego na panelu początkowym, ale na panelu **Definiuj** , z którym jest wyświetlany.

W poniższym przykładzie przedstawiono sposób definiowania kolejki lokalnej przy użyciu istniejącej kolejki jako szablonu.

Definiowanie kolejki lokalnej

Aby zdefiniować lokalny obiekt kolejki z poziomu paneli operacji i sterowania, należy użyć istniejącej definicji kolejki jako podstawy dla nowej definicji. Istnieje kilka paneli do wykonania. Po zakończeniu wszystkich paneli i upewnieniu się, że atrybuty są poprawne, naciśnij klawisz Enter, aby wysłać definicję do menedżera kolejek, który następnie tworzy rzeczywistą kolejkę.

Użyj działania **Definiuj podobne** albo na panelu początkowym, albo dla pozycji obiektu na liście wyświetlonej w wyniku opcji wybranych na panelu początkowym.

Na przykład, począwszy od panelu początkowego, wypełniaj następujące pola:

Działanie	2 (Definiuj podobne)
Typ obiektu	QLOCAL
Nazwa	QUEUE.YOU.LIKE. Jest to nazwa kolejki, która udostępnia atrybuty dla nowej kolejki.

Naciśnij klawisz Enter, aby wyświetlić panel **Zdefiniuj kolejkę lokalną** . Pole nazwy kolejki jest puste, dzięki czemu można podać nazwę nowej kolejki. Opis jest taki, że w kolejce, na której bazuje się ta nowa definicja. W tym polu należy wpisać własny opis dla nowej kolejki.

Wartości w pozostałych polach są wartościami kolejkowania, dla których jest używana ta nowa kolejka, z wyjątkiem rozporządzenia. Pola te można wpisać w zależności od tego, które pola są wymagane. Na przykład wpisz Y w polu **Umieść włączone** (jeśli nie jest jeszcze Y), jeśli odpowiednio autoryzowane aplikacje mogą umieszczać komunikaty w tej kolejce.

Pomoc do pola można uzyskać, przesuwając kursor w pole i naciskając klawisz funkcyjny F1. Pomoc dotycząca pól zawiera informacje na temat wartości, które mogą być używane dla każdego atrybutu.

Po zakończeniu pierwszego panelu naciśnij klawisz funkcyjny F8 , aby wyświetlić drugi panel.

Wskazówki:

1. Na tym etapie nie należy naciskać klawisza Enter. W przeciwnym razie kolejka zostanie utworzona, zanim będzie możliwe zakończenie pozostałych pól. (Jeśli naciśniesz klawisz Enter przedwcześnie, nie martw się; zawsze możesz zmienić swoją definicję później.)
2. Nie należy naciskać klawiszy funkcyjnych F3 ani F12, albo dane wpisane do systemu zostaną utracone.

Naciśnij klawisz funkcyjny F8 wielokrotnie, aby zobaczyć i zakończyć pozostałe panele, w tym definicję wyzwalacza, element sterujący zdarzenia i panele raportowania wycofania.

Po zakończeniu lokalnej definicji kolejki

Po zakończeniu definiowania należy nacisnąć klawisz Enter, aby wysłać informacje do menedżera kolejek w celu przetworzenia. Menedżer kolejek tworzy kolejkę zgodnie z podaną definicją. Jeśli nie chcesz, aby kolejka została utworzona, naciśnij klawisz funkcyjny F3, aby wyjść z definicji i anulować ją.

Definiowanie innych typów obiektów

Aby zdefiniować inne typy obiektów, należy użyć istniejącej definicji jako podstawy dla nowej definicji, tak jak wyjaśniono to w sekcji [Definiowanie kolejki lokalnej](#).

Użyj działania **Definiuj podobne** albo na panelu początkowym, albo dla pozycji obiektu na liście wyświetlonej w wyniku opcji wybranych na panelu początkowym.

Na przykład, począwszy od panelu początkowego, wypełniaj następujące pola:

Działanie	2 (Definiuj podobne)
Typ obiektu	QALIAS, NAMELIST, PROCESS, CHANNEL i inne obiekty zasobów.
Nazwa	Pozostaw to pole puste lub wprowadź nazwę istniejącego obiektu tego samego typu.

Naciśnij klawisz Enter, aby wyświetlić odpowiednie panele DEFINE. Wypełnij pola zgodnie z wymaganiami, a następnie ponownie naciśnij klawisz Enter, aby wysłać informacje do menedżera kolejek.

Podobnie jak w przypadku definiowania kolejki lokalnej, zdefiniowanie innego typu obiektu wymaga wykonania kilku paneli, które mają być wykonane. Zdefiniowanie listy nazw wymaga wykonania dodatkowych czynności, zgodnie z opisem w sekcji ["Praca z listami nazw" na stronie 359](#).

Praca z definicjami obiektów

Jeśli obiekt został zdefiniowany, można określić działanie w polu **Działanie**, aby zmienić, wyświetlić lub zarządzać nim.

W każdym przypadku można wykonać jedną z następujących czynności:

- Wybierz obiekt, z którym chcesz pracować, z listy wyświetlonej w wyniku opcji wybranych na panelu początkowym. Na przykład po wprowadzeniu wartości 1 w polu **Działanie** w celu wyświetlenia obiektów, Queue w polu **Typ obiektu** i * w polu **Nazwa**, zostanie wyświetlona lista wszystkich kolejek zdefiniowanych w systemie. Następnie można wybrać z tej listy kolejkę, z którą będzie potrzebna praca.
- Zaczynij od panelu początkowego, w którym określasz obiekt, z którym pracujesz, wypełniając pola **Typ obiektu** i **Nazwa**.

Modyfikowanie definicji obiektu

Aby zmienić definicję obiektu, określ działanie 3 i naciśnij klawisz Enter, aby wyświetlić panele ALTER. Panele te są bardzo podobne do paneli DEFINE. Użytkownik może zmienić żądane wartości. Po zakończeniu wprowadzania zmian naciśnij klawisz Enter, aby wysłać informacje do menedżera kolejek.

Wyświetlanie definicji obiektu

Aby wyświetlić szczegóły obiektu bez możliwości ich zmiany, należy określić działanie 1 i nacisnąć klawisz Enter, aby wyświetlić panele DISPLAY. Panele te są podobne do paneli DEFINE, z tym wyjątkiem, że nie można zmienić żadnego z tych pól. Zmień nazwę obiektu, aby wyświetlić szczegółowe informacje o innym obiekcie.

Usuwanie obiektu

Aby usunąć obiekt, należy określić działanie 4 (Zarządzanie), a działanie **Usuń** jest jednym z działań prezentowanych w menu wynikowym. Wybierz czynność **Usuń**.

Poproszono Cię o potwierdzenie żądania. Jeśli naciśniesz klawisz funkcyjny F3 lub F12, żądanie zostanie anulowane. Jeśli naciśniesz klawisz Enter, żądanie zostanie potwierdzone i przekazane do menedżera kolejek. Określony obiekt jest następnie usuwany.

Uwaga: Nie można usunąć większości typów obiektów kanału, chyba że inicjator kanału jest uruchomiony.

Praca z listami nazw

Podczas pracy z listami nazw należy postępować tak, jak w przypadku innych obiektów.

W przypadku czynności DEFINE LIKE lub ALTER należy nacisnąć klawisz funkcyjny F11, aby dodać nazwy do listy lub zmienić nazwy na liście. Wiąże się to z pracą z edytorem ISPF i wszystkimi dostępnymi komendami edycji ISPF. Każdą nazwę należy wprowadzić na liście nazw w osobnym wierszu.

Jeśli w ten sposób używany jest edytor ISPF, ustawienia klawiszy funkcyjnych są normalnym ustawieniami ISPF, a **nie** są to ustawienia używane przez inne operacje i panele kontrolne.

Jeśli na liście konieczne jest określenie małych liter, należy w wierszu komend panelu edytora podać wartość CAPS (OFF). W takim przypadku wszystkie listy nazw, które są edytowane w przyszłości, są pisane małymi literami, dopóki nie zostanie podana wartość CAPS (ON).

Po zakończeniu edycji listy nazw naciśnij klawisz funkcyjny F3, aby zakończyć sesję edycji ISPF. Następnie naciśnij klawisz Enter, aby wystać zmiany do menedżera kolejek.

Uwaga: Jeśli na tym etapie nie naciśniesz klawisza Enter, ale zamiast tego klawisza funkcyjnego naciśnij klawisz F3, zostaną utracone wszystkie aktualizacje, o których wpisano.

Implementowanie systemu przy użyciu wielu kolejek transmisji klastra

Nie ma znaczenia, czy kanał jest używany w pojedynczym klastrze, czy też w klastrze nakładającym się. Gdy kanał jest wybrany i uruchomiony, kanał wybiera kolejkę transmisji w zależności od definicji.

Procedura

- Jeśli używana jest opcja DEFCLXQ, patrz [“Korzystanie z automatycznej definicji kolejek i przetaczania” na stronie 359](#).
- Jeśli używane jest podejście etapowo, patrz [“Zmiana kanałów nadajnika klastra przy użyciu metody stopniowej” na stronie 360](#).

Korzystanie z automatycznej definicji kolejek i przetaczania

Użyj tej opcji, jeśli planujesz użyć opcji DEFCLXQ. Dla każdego kanału zostanie utworzona kolejka i każdy nowy kanał.

Procedura

1. Zapoznaj się z definicją SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE i w razie potrzeby zmień atrybuty. Ta kolejka jest zdefiniowana w podzbiorze SCSQPROC (csq4insx).

2. Utwórz SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE Kolejka modelowa.
3. Zastosuj strategię bezpieczeństwa dla tej kolejki modelowej i SYSTEM.CLUSTER.TRANSMIT. * * .
Dla z/OS identyfikator użytkownika uruchomionego zadania inicjatora kanału:

- Kontrola dostępu do klasy (MQADMIN) dla

```
ssid.CONTEXT.SYSTEM.CLUSTER.TRANSMIT.channelname
```

- Aktualizacja dostępu do klasy (MQQUEUE) dla

```
ssid.SYSTEM.CLUSTER.TRANSMIT.channelname
```

Zmiana kanałów nadajnika klastra przy użyciu metody stopniowej

Użyj tej opcji, jeśli planujesz użycie etapowego podejścia. Ten proces umożliwia przenoszenie się do nowych kanałów wysyłających klastry w różnych momentach, aby dostosować je do potrzeb przedsiębiorstwa.

Zanim rozpocznie

- Zidentyfikuj aplikacje biznesowe i kanały, które są używane.
- W przypadku kolejek, które są używane, wyświetlane są klastry, w których znajdują się one.
- Wyświetl kanały, aby wyświetlić nazwy połączeń, nazwy menedżerów kolejek zdalnych i klastry obsługiwane przez kanał.

O tym zadaniu

- Utwórz kolejkę transmisji. W systemie z/OS można rozważyć, który zestaw stron jest używany dla kolejki.
- Skonfiguruj strategię bezpieczeństwa dla kolejki.
- Zmień dowolne monitorowanie kolejek, tak aby uwzględniła tę nazwę kolejki.
- Zdecyduj, które kanały mają używać tej kolejki transmisji. Kanały powinny mieć podobną nazwę, więc znaki ogólne " * ' w polu CLCHNAME identyfikują kanał.
- Gdy użytkownik jest gotowy do korzystania z nowej funkcji, zmień kolejkę transmisji, aby określić nazwę kanałów, w których ma być używana ta kolejka transmisji. Na przykład CLUSTER1.TOPARIS lub CLUSTER1.* lub *.TOPARIS
- Uruchom kanały

Procedura

1. Użyj komendy DIS CLUSQMGR(xxxx) XMITQ , aby wyświetlić kanały nadajnika klastra zdefiniowane w klastrze, gdzie xxxx jest nazwą zdalnego menedżera kolejek.
2. Skonfiguruj profil zabezpieczeń dla kolejki transmisji i przekaz uprawnienia dostępu do kolejki inicjatorowi kanału.
3. Zdefiniuj kolejkę transmisji, która ma być używana, i określ wartość USAGE (XMITQ) INDXTYPE (CORRELID) SHARE i CLCHNAME (wartość)

Identyfikator użytkownika uruchomionego zadania inicjatora kanału musi mieć następujący dostęp:

```
alter class(MQADMIN) ssid.CONTEXT.SYSTEM.CLUSTER.TRANSMIT.channel
update class(MQQUEUE ssid.SYSTEM.CLUSTER.TRANSMIT.channel
```

a ID użytkownika za pomocą komendy SWITCH wymaga następującego dostępu:

```
alter cl(MQADMIN) ssid.QUEUE.queueName
```


4. Zatrzymaj i zrestartuj kanały.

Zmiana kanału ma miejsce wtedy, gdy kanał jest uruchamiany za pomocą komendy MQSC, lub gdy używany jest program CSQUTIL. Można określić kanały, które mają zostać zrestartowane za pomocą SWITCH CHANNEL (*) STATUS programu CSQUTIL.

Jeśli występują problemy podczas uruchamiania kanału, zatrzymaj kanał, rozwiąż problemy i zrestartuj kanał.

Należy pamiętać, że można zmienić atrybut CLCHNAME tak często, jak jest to konieczne.

Wartość parametru CLCHNAME jest używana podczas uruchamiania kanału, więc można zmienić definicję CLCHNAME, gdy kanał będzie nadal używać definicji od momentu uruchomienia kanału. Kanał używa nowej definicji po zrestartowaniu.

Cofanie zmiany w systemie z/OS

Jeśli wyniki nie są tak, jak się tego spodziewasz, konieczne jest posiadanie procesu wycofywania zmian.

Co może pójść nie tak?

Jeśli nowa kolejka transmisji nie jest tym, czego oczekujesz:

1. Sprawdź, czy wartość CLCHNAME jest tak, jak się spodziewa
2. Przejrzyj protokół zadania, aby sprawdzić, czy proces przełączania został zakończony. Jeśli nie, odczekaj i sprawdź nową kolejkę transmisji kanału później.

W przypadku korzystania z wielu kolejek transmisji klastra ważne jest, aby definicje kolejek transmisji były jawnie projektowane i unikać skomplikowanych nakładających się konfiguracji. W ten sposób można się upewnić, że w razie wystąpienia problemów można wrócić do oryginalnych kolejek i konfiguracji.

W przypadku napotkania problemów podczas przenoszenia do innej kolejki transmisji należy rozwiązać wszystkie problemy, zanim będzie można kontynuować zmianę.

Istniejące żądanie zmiany musi zostać zakończone, zanim możliwe będzie wykonanie nowego żądania zmiany. Na przykład:

1. Zdefiniuj nową kolejkę transmisji z maksymalną głębokością jednego i 10 komunikatów oczekujących na wysłanie.
2. Zmień kolejkę transmisji tak, aby określała nazwę kanału w parametrze CLCHNAME.
3. Zatrzymaj i zrestartuj kanał. Próba przeniesienia komunikatów nie powiodła się i zgłasza problemy.
4. Zmień wartość parametru CLCHNAME w kolejce transmisji, aby była pusta.
5. Zatrzymaj i zrestartuj kanał. Kanał nadal próbuje i uzupełnia oryginalne żądanie, dlatego kanał będzie nadal używać nowej kolejki transmisji.
6. Należy rozwiązać problemy i zrestartować kanał, tak aby przenoszenie komunikatów zostało zakończone pomyślnie.

Gdy kanał zostanie zrestartowany przy następnym restarcie kanału, zostaną one zmienione, więc jeśli parametr CLCHNAME został ustawiony na wartość pustą, kanał nie będzie używać określonej kolejki transmisji.

W tym przykładzie zmiana parametru CLCHNAME w kolejce transmisji na odstępy nie musi oznaczać, że kanał używa SYSTEM.CLUSTER.TRANSMIT , ponieważ mogą istnieć inne kolejki transmisji, których parametr CLCHNAME jest zgodny z nazwą kanału. Na przykład nazwa ogólna lub wartość atrybutu DEFCLXQ menedżera kolejek może być ustawiona na kanał, dlatego kanał używa kolejki dynamicznej zamiast SYSTEM.CLUSTER.TRANSMIT .

z/OS **Pisanie programów do administrowania programem IBM MQ for z/OS**

Aby administrować menedżerem kolejek, można napisać własne programy użytkowe. Ten temat zawiera informacje na temat wymagań dotyczących pisania własnych programów administracyjnych.

Początek ogólnych informacji o interfejsie programistycznym

Ten zestaw tematów zawiera wskazówki i wskazówki, które umożliwiają wydawanie komend produktu IBM MQ z poziomu programu aplikacji IBM MQ .

Uwaga: W tym temacie wywołania MQI są opisane za pomocą notacji języka C. Typowe wywołania wywołań w językach COBOL, PL/I i assembler można znaleźć w podręczniku [Function calls](#) (wywołania funkcji).

Zrozumienie, jak to wszystko działa

W schemacie, procedura wydawania komend z aplikacji jest następująca:

1. Zbuduj komendę IBM MQ w typie komunikatu IBM MQ o nazwie *komunikat żądania*. Komenda może być w formacie MQSC lub PCF.
2. Wyślij (użyj komendy MQPUT) Ten komunikat do kolejki specjalnej wywołał kolejkę wejściową komendy systemowej. Procesor komend IBM MQ uruchamia komendę.
3. Pobieranie (za pomocą komendy MQGET) Wyniki komendy jako *komunikaty odpowiedzi* w kolejce odpowiedzi. Komunikaty te zawierają komunikaty użytkownika, które należy określić, aby określić, czy komenda zakończyła się pomyślnie, a jeśli tak, to jakie były wyniki.

Następnie należy do programu użytkowego, aby przetworzyć wyniki.

Ten zestaw tematów zawiera:

z/OS **Przygotowywanie kolejek do programów administracyjnych**

Programy administracyjne wymagają pewnej liczby predefiniowanych kolejek na potrzeby wprowadzania komend systemowych i odbierania odpowiedzi.

Ta sekcja ma zastosowanie do komend w formacie MQSC. Odpowiednik w PCF znajduje się w sekcji [“Korzystanie z formatów komend programowalnych IBM MQ”](#) na stronie 23.

Zanim możliwe będzie wydanie wszystkich wywołań MQPUT lub MQGET , należy najpierw zdefiniować, a następnie otworzyć kolejki, które będą używane.

Definiowanie systemowej kolejki wejściowej komend

Kolejka wejściowa komend systemowych jest kolejką lokalną o nazwie SYSTEM.COMMAND.INPUT. Podany zestaw danych inicjowania CSQINP2 , thlqual.SCSQPROC(CSQ4INSG), zawiera definicję domyślną dla kolejki wejściowej komend systemowych. Aby zachować kompatybilność z produktem IBM MQ na innych platformach, należy użyć aliasu tej kolejki, nazywanego SYSTEM.ADMIN.COMMAND.QUEUE jest również dostarczana. Więcej informacji na ten temat zawiera sekcja [Przykładowe definicje dostarczane wraz z produktem IBM MQ](#) .

Definiowanie kolejki odpowiedzi

Aby odbierać komunikaty odpowiedzi z procesora komend IBM MQ , należy zdefiniować kolejkę odpowiedzi. Może to być dowolna kolejka z atrybutami, które umożliwiają umieszczanie na nim komunikatów odpowiedzi. Jednak w przypadku normalnego działania należy określić następujące atrybuty:

- UŻYCIE (NORMALNE)
- NOTRIGGER (o ile aplikacja nie korzysta z wyzwalania)

Należy unikać używania trwałych komunikatów dla komend, ale jeśli zostanie wybrana taka opcja, kolejka odpowiedzi nie może być tymczasową kolejką dynamiczną.

Podany zestaw danych inicjalizacji CSQINP2 , thlqual.SCSQPROC(CSQ4INSG), zawiera definicję kolejki modelowej o nazwie SYSTEM.COMMAND.REPLY.MODEL. Tego modelu można użyć do utworzenia dynamicznej kolejki odpowiedzi.

Uwaga: Odpowiedzi wygenerowane przez procesor komend mogą mieć długość do 15 000 bajtów.

Jeśli jako kolejka odpowiedzi używana jest stała kolejka dynamiczna, aplikacja powinna zezwalać na zakończenie wszystkich operacji PUT i GET przed podjęciem próby usunięcia kolejki. W przeciwnym razie można zwrócić wartość MQRC2055 (MQRC_Q_NOT_EMPTY). Jeśli wystąpi taka sytuacja, spróbuj usunąć kolejkę ponownie po kilku sekundach.

Otwieranie kolejki wejściowej komend systemowych

Zanim będzie można otworzyć kolejkę wejściową systemu, program użytkowy musi być połączony z menedżerem kolejek. Aby wykonać tę operację, należy użyć wywołania MQI MQCONN lub MQCONNX .

Następnie użyj wywołania MQI MQOPEN , aby otworzyć systemową kolejkę wejściową komendy. Aby użyć tego wywołania:

1. Ustaw parametr **Options** na wartość MQOO_OUTPUT.
2. Ustaw pola deskryptora obiektu MQOD w następujący sposób:

ObjectType

MQOT_Q (obiekt jest kolejką)

ObjectName

SYSTEM.COMMAND.INPUT

ObjectQMgrName

Jeśli chcesz wysłać komunikaty żądań do lokalnego menedżera kolejek, pozostaw to pole puste. Oznacza to, że komendy są przetwarzane lokalnie.

Jeśli chcesz, aby komendy IBM MQ były przetwarzane w zdalnym menedżerze kolejek, należy umieścić jego nazwę w tym miejscu. Konieczne jest również posiadanie poprawnych kolejek i połączeń, zgodnie z opisem w sekcji [Rozproszone kolejkowanie i klastry](#).

Otwieranie kolejki odpowiedzi

Aby pobrać odpowiedzi z komendy IBM MQ , należy otworzyć kolejkę odpowiedzi. Jednym ze sposobów na to jest określenie kolejki modelowej SYSTEM.COMMAND.REPLY.MODEL w wywołaniu MQOPEN , aby utworzyć stałą kolejkę dynamiczną jako kolejkę zwrotną. Aby użyć tego wywołania:

1. Ustaw parametr **Options** na wartość MQOO_INPUT_SHARED
2. Ustaw pola deskryptora obiektu MQOD w następujący sposób:

ObjectType

MQOT_Q (obiekt jest kolejką)

ObjectName

Nazwa kolejki odpowiedzi. Jeśli podana nazwa kolejki jest nazwą obiektu kolejki modelowej, menedżer kolejek tworzy kolejkę dynamiczną.

ObjectQMgrName

Aby otrzymywać odpowiedzi na temat lokalnego menedżera kolejek, pozostaw to pole puste.

DynamicQName

Należy określić nazwę kolejki dynamicznej, która ma zostać utworzona.

Serwer komend jest komponentem produktu IBM MQ, który współpracuje z komponentem procesora komend. Można wysłać sformatowane komunikaty do serwera komend, który interpretuje komunikaty, uruchamia żądania administracyjne i wysyła odpowiedzi z powrotem do aplikacji administracyjnej.

Serwer komend odczytuje komunikaty żądań z kolejki wejściowej komend systemowych, weryfikuje je i przekazuje te poprawne jako komendy do procesora komend. Procesor komend przetwarza komendy i umieszcza wszystkie odpowiedzi w odpowiedzi na komunikaty wysłane do kolejki odpowiedzi. Pierwszy komunikat odpowiedzi zawiera komunikat użytkownika CSQN205I. Więcej informacji na ten temat zawiera sekcja [“Interpretowanie komunikatów odpowiedzi z serwera komend”](#) na stronie 368. Serwer komend przetwarza także komendy inicjatora kanału i grupy współużytkowania kolejki, niezależnie od miejsca, z którego są one wydawane.

Identyfikowanie menedżera kolejek, który przetwarza komendy

Menedżer kolejek przetwarzający komendy, które są uruchamiane z programu administracyjnego, jest menedżerem kolejek, który jest właścicielem kolejki wejściowej komend systemowych, do której jest umieszczany komunikat.

Uruchamianie serwera komend

Zwykle serwer komend jest uruchamiany automatycznie podczas uruchamiania menedżera kolejek. Staje się on dostępny, gdy tylko komunikat CSQ9022I 'START QMGR' NORMAL COMPLETION zostanie zwrócony z komendy START QMGR. Serwer komend jest zatrzymany, gdy wszystkie połączone zadania zostały rozłączone podczas fazy zakończenia systemu.

Użytkownik może samodzielnie sterować serwerem komend za pomocą komend START CMDSERV i STOP CMDSERV. Aby zapobiec automatyczności uruchamiania serwera komend po restarcie systemu IBM MQ, można dodać komendę STOP CMDSERV do zestawów danych inicjowania CSQINP1 lub CSQINP2. Nie jest to jednak zalecane, ponieważ uniemożliwia przetwarzanie wszystkich komend inicjatora kanału lub grupy współużytkowania kolejek.

Komenda STOP CMDSERV zatrzymuje serwer komend natychmiast po zakończeniu przetwarzania bieżącego komunikatu lub natychmiast, jeśli żadne komunikaty nie są przetwarzane.

Jeśli serwer komend został zatrzymany przez komendę STOP CMDSERV w programie, żadne inne komendy z programu nie mogą być przetwarzane. Aby zrestartować serwer komend, należy wywołać komendę START CMDSERV z poziomu konsoli z/OS.

Jeśli podczas działania menedżera kolejek zostanie zatrzymany i zrestartowany serwer komend, wszystkie komunikaty, które znajdują się w kolejce wejściowej komend systemowych po zatrzymaniu serwera komend, zostaną przetworzone po zrestartowaniu serwera komend. Jeśli jednak menedżer kolejek zostanie zatrzymany i zrestartowany po zatrzymaniu serwera komend, po zrestartowaniu serwera komend przetwarzane są tylko komunikaty trwałe w kolejce wejściowej komend systemowych. Wszystkie nietrwałe komunikaty w kolejce wejściowej komend systemowych są traczone.

Wysyłanie komend do serwera komend

Dla każdej komendy należy utworzyć komunikat zawierający komendę, a następnie umieścić go w kolejce wejściowej komend systemowych.

Budowanie komunikatu zawierającego komendy produktu IBM MQ

Komendy produktu IBM MQ można włączyć w programie użytkowym, budując komunikaty żądań, które zawierają wymagane komendy. Dla każdej takiej komendy użytkownik:

1. Utwórz bufor zawierający łańcuch znaków reprezentujący komendę.
2. Wywołaj wywołanie MQPUT, podając nazwę buforu w parametrze **buffer** wywołania.

Najprostszym sposobem, aby to zrobić w języku C, jest zdefiniowanie buforu za pomocą 'char'. Na przykład:

```
char message_buffer[ ] = "ALTER QLOCAL(SALES) PUT(ENABLED)";
```

Podczas budowania komendy należy użyć łańcucha znakowego zakończony znakiem o kodzie zero. Nie określaj łańcucha przedrostka komendy (CPF) na początku komendy zdefiniowanej w ten sposób. Oznacza to, że nie trzeba zmieniać skryptów komend, jeśli mają być uruchamiane w innym menedżerze kolejek. Należy jednak wziąć pod uwagę, że CPF jest dołączany do wszystkich komunikatów odpowiedzi, które są umieszczane w kolejce odpowiedzi.

Serwer komend składowuje wszystkie małe litery na wielkie litery, chyba że znajdują się one w cudzysłowie.

Komendy mogą mieć dowolną długość maksymalnie do 32 762 znaków.

Umieszczanie komunikatów w kolejce wejściowej komend systemowych

Użyj wywołania MQPUT , aby umieścić komunikaty żądań zawierające komendy w kolejce wejściowej komend systemowych. W tym wywołaniu określa się nazwę kolejki odpowiedzi, która została już otwarta.

Aby użyć wywołania MQPUT :

1. Ustaw następujące parametry MQPUT :

Hconn

Uchwyt połączenia zwracany przez wywołanie MQCONN lub MQCONNX .

Hobj

Uchwyt obiektu zwrócony przez wywołanie komendy MQOPEN dla kolejki wejściowej komend systemowych.

BufferLength

Długość sformatowanej komendy.

Buffer

Nazwa buforu zawierającego komendę.

2. Ustaw następujące pola MQMD:

MsgType

MQMT_REQUEST

Format

MQFMT_STRING lub MQFMT_NONE

Jeśli ta sama strona kodowa nie jest używana jako menedżer kolejek, ustaw odpowiednio opcję *CodedCharSetId* i ustaw wartość MQFMT_STRING, aby serwer komend mógł przekształcić komunikat. Nie należy ustawiać parametru MQFMT_ADMIN, ponieważ powoduje to, że komenda ma być interpretowana jako PCF.

ReplyToQ

Nazwa kolejki odpowiedzi.

ReplyToQMgr

Jeśli chcesz, aby odpowiedzi były wysyłane do lokalnego menedżera kolejek, pozostaw to pole puste. Jeśli komendy produktu IBM MQ mają być wysyłane do zdalnego menedżera kolejek, należy umieścić jego nazwę w tym miejscu. Konieczne jest również skonfigurowanie poprawnych kolejek i połączeń, zgodnie z opisem w sekcji [Rozproszone kolejkowanie i klastry](#).

3. W razie potrzeby ustaw wszystkie inne pola MQMD. W przypadku komend zwykle należy używać nietrwałych komunikatów.
4. Ustaw wszystkie opcje *PutMsgOpts* zgodnie z wymaganiami.

Jeśli zostanie określona wartość MQPMO_SYNCPOINT (wartość domyślna), należy wykonać wywołanie MQPUT z wywołaniem punktu synchronizacji.

Korzystanie z komendy MQPUT1 i kolejki wejściowej komend systemowych

Jeśli chcesz umieścić tylko jeden komunikat w kolejce wejściowej komend systemowych, możesz użyć wywołania MQPUT1. Połączenie to łączy funkcje MQOPEN, a następnie MQPUT jednego komunikatu, po którym następuje MQCLOSE, a wszystko to w jednym wywołaniu. Jeśli skorzystasz z tego wywołania, zmodyfikuj odpowiednio parametry. Szczegółowe informacje na ten temat zawiera sekcja [Wycinanie jednego komunikatu w kolejce przy użyciu wywołania MQPUT1](#).

Pobieranie odpowiedzi do komend

Serwer komend wysyła odpowiedź do kolejki odpowiedzi dla każdego otrzymanego komunikatu żądania. Każda aplikacja administracyjna musi odbierać i obsługiwać komunikaty odpowiedzi.

Gdy procesor komend przetwarza komendy, wszystkie komunikaty odpowiedzi są umieszczane w kolejce odpowiedzi określonej w wywołaniu komendy MQPUT. Serwer komend wysyła komunikaty odpowiedzi z taką samą trwałością, jak otrzymana przez niego komunikat komendy.

Oczekiwanie na odpowiedź

Użyj wywołania MQGET, aby pobrać odpowiedź z komunikatu żądania. Jeden komunikat żądania może wygenerować kilka komunikatów odpowiedzi. Szczegółowe informacje na ten temat zawiera sekcja ["Interpretowanie komunikatów odpowiedzi z serwera komend"](#) na stronie 368.

Istnieje możliwość określenia przedziału czasu, przez który wywołanie MQGET będzie oczekiwać na wygenerowanie komunikatu odpowiedzi. Jeśli nie uzyskasz odpowiedzi, skorzystaj z listy kontrolnej zaczynając od tematu ["Jeśli nie otrzymujesz odpowiedzi"](#) na stronie 368.

Aby użyć wywołania MQGET :

1. Ustaw następujące parametry:

Hconn

Uchwyt połączenia zwracany przez wywołanie MQCONN lub MQCONNX.

Hobj

Uchwyt obiektu zwracany przez wywołanie MQOPEN dla kolejki odpowiedzi.

Buffer

Nazwa obszaru, do którego ma zostać wysłana odpowiedź.

BufferLength

Długość buforu, w którym ma zostać odebranie odpowiedzi. Musi to być co najmniej 80 bajtów.

2. Aby mieć pewność, że odpowiedzi uzyskasz tylko z wywołanej komendy, należy określić odpowiednie pola *MsgId* i *CorrelId*. Zależą one od opcji raportu, MQMD_REPORT, określonych w wywołaniu komendy MQPUT :

MQRO_NONE

Binarne zero, '00 ... 00' (24 znaki puste).

MQRO_NEW_MSG_ID

Binarne zero, '00 ... 00' (24 znaki puste).

Jest to wartość domyślna, jeśli żadna z tych opcji nie została określona.

MQRO_PASS_MSG_ID

MsgId z MQPUT.

MQRO_NONE

MsgId z wywołania MQPUT.

MQRO_COPY_MSG_ID_TO_CORREL_ID (Identyfikator CORREL_ID)

MsgId z wywołania MQPUT.

Jest to wartość domyślna, jeśli żadna z tych opcji nie została określona.

MQRO_PASS_CORREL_ID

CorrelId z wywołania MQPUT .

Więcej informacji na temat opcji raportu zawiera sekcja Opcje raportu i flagi komunikatów.

3. Ustaw następujące pola *GetMsgOpts* :

Options

MQGMO_WAIT

Jeśli ta sama strona kodowa nie jest używana jako menedżer kolejek, należy ustawić wartość MQGMO_CONVERT i ustawić wartość *CodedCharSetId* odpowiednio w deskryptywie MQMD.

WaitInterval

Aby uzyskać odpowiedzi z lokalnego menedżera kolejek, spróbuj wykonać 5 sekund. Zakodowana w milisekundach, staje się to 5 000. Aby uzyskać odpowiedzi ze zdalnego menedżera kolejek oraz komendy kontroli kanału i statusu, spróbuj wykonać 30 sekund. Zakodowane w milisekundach, to staje się to 30 000.

Usunięte komunikaty

Jeśli serwer komend stwierdzi, że komunikat żądania nie jest poprawny, usuwa ten komunikat i zapisuje komunikat CSQN205I w nazwanej kolejce odpowiedzi. Jeśli nie ma kolejki odpowiedzi, komunikat CSQN205I jest umieszczany w kolejce niedostarczonych komunikatów. Kod powrotu w tym komunikacie wskazuje, dlaczego oryginalny komunikat żądania nie był poprawny:

00D5020F Nie jest to typ MQMT_REQUEST.

00D50210 Ma ona zerową długość.

00D50212 Jest ona dłuższa niż 32 762 bajty.

00D50211 Zawiera on wszystkie odstępy.

00D5483E Wymagane jest przekształcenie, ale *Format* nie było typu MQFMT_STRING.

Inne Patrz sekcja Kody serwera komend .

Deskryptor komunikatu odpowiedzi serwera komend

Dla każdego komunikatu odpowiedzi ustawione są następujące pola deskryptora komunikatu MQMD:

MsgType MQMT_REPLY

Feedback MQFB_NONE

Encoding MQENC_NATIVE

Priority Tak jak w przypadku deskryptora MQMD w komunikacie, który został wygenerowany.

Persistence Tak jak w przypadku deskryptora MQMD w komunikacie, który został wygenerowany.

CorrelId Zależy od opcji raportu MQPUT .

ReplyToQ Brak.

Serwer komend ustawia pole *Options* struktury MQPMO na MQPMO_NO_SYNCPOINT. Oznacza to, że można pobierać odpowiedzi w miarę ich tworzenia, a nie jako grupę w następnym punkcie synchronizacji.

Interpretowanie komunikatów odpowiedzi z serwera komend

Każdy komunikat żądania poprawnie przetworzony przez produkt IBM MQ generuje co najmniej dwa komunikaty odpowiedzi. Każdy komunikat odpowiedzi zawiera pojedynczy komunikat użytkownika produktu IBM MQ .

Długość odpowiedzi zależy od komendy, która została wydana. Najdłuższa odpowiedź, jaką można uzyskać, pochodzi z tabeli NAMELIST, która może mieć długość do 15 000 bajtów.

Pierwszy komunikat użytkownika, CSQN205I, zawsze zawiera:

- Liczba odpowiedzi (w postaci dziesiętnej), które mogą być używane jako licznik w pętli, aby uzyskać resztę odpowiedzi. Liczba ta obejmuje ten pierwszy komunikat.
- Kod powrotu z preprocesora komend.
- Kod przyczyny, który jest kodem przyczyny z procesora komend.

Ten komunikat nie zawiera CPF.

Na przykład:

```
CSQN205I    COUNT=    4, RETURN=0000000C, REASON=00000008
```

Pole COUNT ma długość 8 bajtów i jest wyrównane do prawej strony. Zawsze zaczyna się od pozycji 18, czyli od razu po COUNT=. Pole RETURN ma długość 8 bajtów w postaci szesnastkowej i znajduje się bezpośrednio po RETURN= na pozycji 35. Pole PRZYCZYNA ma długość 8 bajtów w postaci szesnastkowej i jest bezpośrednio po REASON= na pozycji 52.

Jeśli wartością parametru RETURN= jest 00000000, a wartością parametru REASON= jest 00000004, to zestaw komunikatów odpowiedzi jest niekompletny. Po pobraniu odpowiedzi wskazanych przez komunikat CSQN205I wywołaj dalsze wywołanie MQGET , aby zaczekać na kolejny zestaw odpowiedzi. Pierwszym komunikatem w następnym zestawie odpowiedzi jest ponownie CSQN205I, co wskazuje na to, ile odpowiedzi ma miejsce i czy nadal istnieje więcej do nich.

Więcej informacji na temat poszczególnych komunikatów można znaleźć w dokumentacji produktu [IBM MQ for z/OS : komunikaty, zakończenie i kody przyczyny](#) .

Jeśli używana jest funkcja języka innego niż angielski, tekst i układ odpowiedzi są różne od tych, które zostały tutaj przedstawione. Jednak wielkość i pozycja kodów licznika i powrotu w komunikacie CSQN205I są takie same.

Jeśli nie otrzymujesz odpowiedzi

Istnieje szereg kroków, które można wykonać, jeśli odpowiedź na żądanie do serwera komend nie zostanie odezvana.

Jeśli odpowiedź na komunikat żądania nie zostanie wysłana, należy skorzystać z następującej listy kontrolnej:

- Czy serwer komend jest uruchomiony?
- Czy *WaitInterval* jest wystarczająco długi?
- Czy kolejki komend i odpowiedzi komend systemu są poprawnie zdefiniowane?
- Czy wywołania MQOPEN do tych kolejek powiodły się?
- Czy w kolejkach komend MQPUT i MQGET są włączone zarówno komendy systemowe-wejściowe, jak i odpowiedzi-do kolejek?
- Czy rozważono zwiększenie atrybutów MAXDEPTH i MAXMSGL dla kolejek użytkownika?
- Czy są używane poprawnie pola *CorrelId* i *MsgId* ?

- Czy menedżer kolejek jest nadal uruchomiony?
- Czy komenda została poprawnie zbudowana?
- Czy wszystkie połączenia zdalne są zdefiniowane i działają poprawnie?
- Czy wywołania MQPUT zostały poprawnie zdefiniowane?
- Czy kolejka odpowiedzi została zdefiniowana jako tymczasowa kolejka dynamiczna zamiast trwałej kolejki dynamicznej? (Jeśli komunikat żądania jest trwały, konieczne jest użycie trwałej kolejki dynamicznej dla odpowiedzi).

Gdy serwer komend generuje odpowiedzi, ale nie może zapisać ich w określonej kolejce odpowiedzi, zapisuje je w kolejce niedostarczonych komunikatów.

Przekazywanie komend za pomocą MGCRE

Za pomocą odpowiedniej autoryzacji program użytkowy może wykonywać żądania do wielu menedżerów kolejek przy użyciu procedury usługi z/OS .

Jeśli użytkownik ma poprawną autoryzację, może przekazać komendy IBM MQ z programu do wielu menedżerów kolejek za pomocą usługi MGCRE (SVC 34) z/OS . Wartość CPF identyfikuje konkretny menedżer kolejek, do którego kierowane jest polecenie. Więcej informacji na temat funkcji CPF zawiera sekcja [Identyfikatory użytkowników dla zabezpieczeń komend i zabezpieczeń zasobów komend](#) i ["Wydawanie komend menedżera kolejek w systemie z/OS" na stronie 341](#).

Jeśli używany jest MGCRE, można użyć znacznika Command and Response Token (CART), aby uzyskać bezpośrednie odpowiedzi na komendę.

Przykłady komend i ich odpowiedzi

Ten temat zawiera szereg przykładów komend do serwera komend oraz odpowiedzi z serwera komend.

Poniżej znajduje się kilka przykładów komend, które można zbudować w komunikatach IBM MQ , a także komunikaty użytkownika, które są odpowiedziami. O ile nie określono inaczej, każdy wiersz odpowiedzi jest osobnym komunikatem.

- [Komunikaty z komendy DEFINE](#)
- [Komunikaty z komendy DELETE](#)
- [Komunikaty z komend DISPLAY](#)
- [Komunikaty z komend z CMDSCOPE](#)
- [Komunikaty z komend, które generują komendy z CMDSCOPE](#)

Komunikaty z komendy DEFINE

the following command:

```
DEFINE QLOCAL(Q1)
```

Generuje następujące komunikaty:

```
CSQ9022I  COUNT=    2, RETURN=00000000, REASON=00000000
CSQ9022I +CSQ1 CSQMMSGP ' DEFINE QLOCAL' NORMAL COMPLETION
```

Te komunikaty odpowiedzi są generowane podczas normalnego zakończenia.

Komunikaty z komendy DELETE

the following command:

```
DELETE QLOCAL(Q2)
```

Generuje następujące komunikaty:

```
CSQN205I  COUNT=    4, RETURN=0000000C, REASON=00000008  
CSQM125I +CSQ1 CSQMUQLC QLOCAL(Q2) QSGDISP(QMGR) WAS NOT FOUND  
CSQM090E +CSQ1 CSQMUQLC FAILURE REASON CODE X'00D44002'  
CSQ9023E +CSQ1 CSQMUQLC 'DELETE QLOCAL' ABNORMAL COMPLETION
```

Te komunikaty wskazują, że kolejka lokalna o nazwie Q2 nie istnieje.

Komunikaty z komend DISPLAY

Poniższe przykłady przedstawiają odpowiedzi z niektórych komend DISPLAY.

Wyszukiwanie nazwy kolejki niedostarczonych komunikatów

Aby znaleźć nazwę kolejki niedostarczonych komunikatów dla menedżera kolejek, należy wywołać tę komendę z poziomu programu użytkowego:

```
DISPLAY QMGR DEADQ
```

Zwracane są następujące trzy komunikaty użytkownika, z których można wyodrębnić wymaganą nazwę:

```
CSQN205I  COUNT=    3, RETURN=00000000, REASON=00000000  
CSQM409I +CSQ1 QMNAME(CSQ1) DEADQ(SYSTEM.DEAD.QUEUE  
CSQ9022I +CSQ1 CSQMDRTS 'DISPLAY QMGR' NORMAL COMPLETION
```

Komunikaty z komendy DISPLAY QUEUE

W poniższych przykładach przedstawiono sposób, w jaki wyniki komendy zależą od atrybutów podanych w tej komendzie.

Przykład 1

Kolejkę lokalną definiuje się za pomocą komendy:

```
DEFINE QLOCAL(Q1) DESCR('A sample queue') GET(ENABLED) SHARE
```

Jeśli w programie aplikacji zostanie wydana następująca komenda:

```
DISPLAY QUEUE(Q1) SHARE GET DESCR
```

te trzy komunikaty użytkownika są zwracane:

```

CSQN205I  COUNT=    3, RETURN=00000000, REASON=00000000
CSQM401I +CSQ1 QUEUE(Q1                                ) TYPE(
QLOCAL ) QSGDISP(QMGR  )
DESCR(A sample queue
) SHARE GET(ENABLED )
CSQ9022I +CSQ1 CSQMDMSG ' DISPLAY QUEUE' NORMAL COMPLETION

```

Uwaga: W tym miejscu wyświetlany jest drugi komunikat CSQM401I zajmujący cztery wiersze.

Przykład 2

Dwie kolejki mają nazwy zaczynające się od litery A:

- A1 jest kolejką lokalną z jej atrybutem PUT ustawionym na DISABLED.
- A2 jest kolejką zdalną, której atrybut PUT jest ustawiony na ENABLED.

Jeśli w programie aplikacji zostanie wydana następująca komenda:

```

DISPLAY QUEUE(A*) PUT

```

te cztery komunikaty użytkownika są zwracane:

```

CSQN205I  COUNT=    4, RETURN=00000000, REASON=00000000
CSQM401I +CSQ1 QUEUE(A1                                ) TYPE(
QLOCAL ) QSGDISP(QMGR  )
PUT(DISABLED )
CSQM406I +CSQ1 QUEUE(A2                                ) TYPE(
QREMOTE ) PUT(ENABLED )
CSQ9022I +CSQ1 CSQMDMSG ' DISPLAY QUEUE' NORMAL COMPLETION

```

Uwaga: Drugi i trzeci komunikat, CSQM401I i CSQM406I, są wyświetlane tutaj, zajmując trzy i dwa wiersze.

Komunikaty z komendy DISPLAY NAMELIST

Za pomocą komendy definiuje się listę nazw:

```

DEFINE NAMELIST(N1) NAMES(Q1,SAMPLE_QUEUE)

```

Jeśli w programie aplikacji zostanie wydana następująca komenda:

```

DISPLAY NAMELIST(N1) NAMES NAMCOUNT

```

Zwracane są następujące trzy komunikaty użytkownika:

```

CSQN205I  COUNT=    3, RETURN=00000000, REASON=00000000
CSQM407I +CSQ1 NAMELIST(N1                             ) QS
GDISP(QMGR  ) NAMCOUNT(    2) NAMES(Q1
,SAMPLE_QUEUE
)
CSQ9022I +CSQ1 CSQMDMSG ' DISPLAY NAMELIST' NORMAL COMPLETION

```

Uwaga: W tym miejscu wyświetlany jest drugi komunikat CSQM407I, który zajmuje trzy wiersze.

Komunikaty z komend z CMDSCOPE

Poniższe przykłady przedstawiają odpowiedzi z komend, które zostały wprowadzone za pomocą atrybutu CMDSCOPE.

Komunikaty z komendy ALTER PROCESS

the following command:

```
ALT PRO(V4) CMDSCOPE(*)
```

tworzy następujące komunikaty:

```
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQN137I !MQ25 'ALT PRO' command accepted for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 5, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'ALT PRO' command responses from MQ26
CSQM125I !MQ26 CSQMMSGP PROCESS(V4) QSGDISP(QMGR) WAS NOT FOUND
CSQM090E !MQ26 CSQMMSGP FAILURE REASON CODE X'00D44002'
CSQ9023E !MQ26 CSQMMSGP 'ALT PRO' ABNORMAL COMPLETION
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'ALT PRO' command responses from MQ25
CSQ9022I !MQ25 CSQMMSGP 'ALT PRO' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=0000000C, REASON=00000008
CSQN123E !MQ25 'ALT PRO' command for CMDSCOPE(*) abnormal completion
```

Komunikaty te mówią, że komenda została wprowadzona w menedżerze kolejek MQ25 i została wysłana do dwóch menedżerów kolejek (MQ25 i MQ26). Komenda zakończyła się pomyślnie w produkcie MQ25, ale definicja procesu nie istnieje na serwerze MQ26, dlatego komenda ta nie powiodła się dla tego menedżera kolejek.

Komunikaty z komendy DISPLAY PROCESS

the following command:

```
DIS PRO(V*) CMDSCOPE(*)
```

tworzy następujące komunikaty:

```
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQN137I !MQ25 'DIS PRO' command accepted for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 5, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DIS PRO' command responses from MQ26
CSQM408I !MQ26 PROCESS(V2) QSGDISP(COPY)
CSQM408I !MQ26 PROCESS(V3) QSGDISP(QMGR)
CSQ9022I !MQ26 CSQMDRTS 'DIS PROCESS' NORMAL COMPLETION
CSQN205I COUNT= 7, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DIS PRO' command responses from MQ25
CSQM408I !MQ25 PROCESS(V2) QSGDISP(COPY)
CSQM408I !MQ25 PROCESS(V2) QSGDISP(GROUP)
CSQM408I !MQ25 PROCESS(V3) QSGDISP(QMGR)
CSQM408I !MQ25 PROCESS(V4) QSGDISP(QMGR)
CSQ9022I !MQ25 CSQMDRTS 'DIS PROCESS' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQN122I !MQ25 'DIS PRO' command for CMDSCOPE(*) normal completion
```

Komunikaty te mówią, że komenda została wprowadzona w menedżerze kolejek MQ25 i została wysłana do dwóch menedżerów kolejek (MQ25 i MQ26). Wyświetlane są informacje o wszystkich procesach w każdym menedżerze kolejek, których nazwy rozpoczynają się od litery V.

Komunikaty z komendy DISPLAY CHSTATUS

the following command:

```
DIS CHS(VT) CMDSCOPE(*)
```

tworzy następujące komunikaty:

```
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQN137I !MQ25 'DIS CHS' command accepted for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 4, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DIS CHS' command responses from MQ25
CSQM422I !MQ25 CHSTATUS(VT) CHLDISP(PRIVATE) CONNAME( ) CURRENT STATUS(STOPPED)
CSQ9022I !MQ25 CSQXDRTS ' DIS CHS' NORMAL COMPLETION
CSQN205I COUNT= 4, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DIS CHS' command responses from MQ26
CSQM422I !MQ26 CHSTATUS(VT) CHLDISP(PRIVATE) CONNAME( ) CURRENT STATUS(STOPPED)
CSQ9022I !MQ26 CSQXDRTS ' DIS CHS' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQN122I !MQ25 'DIS CHS' command for CMDSCOPE(*) normal completion
```

Komunikaty te mówią, że komenda została wprowadzona w menedżerze kolejek MQ25 i została wysłana do dwóch menedżerów kolejek (MQ25 i MQ26). Informacje są wyświetlane na temat statusu kanału w każdym menedżerze kolejek.

Komunikaty z komendy STOP CHANNEL

the following command:

```
STOP CHL(VT) CMDSCOPE(*)
```

Generuje następujące komunikaty:

```
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQN137I !MQ25 'STOP CHL' command accepted for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'STOP CHL' command responses from MQ25
CSQM134I !MQ25 CSQMTCHL STOP CHL(VT) COMMAND ACCEPTED
SQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'STOP CHL' command responses from MQ26
CSQM134I !MQ26 CSQMTCHL STOP CHL(VT) COMMAND ACCEPTED
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'STOP CHL' command responses from MQ26
CSQ9022I !MQ26 CSQXCRPS ' STOP CHL' NORMAL COMPLETION
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'STOP CHL' command responses from MQ25
CSQ9022I !MQ25 CSQXCRPS ' STOP CHL' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQN122I !MQ25 'STOP CHL' command for CMDSCOPE(*) normal completion
```

Komunikaty te mówią, że komenda została wprowadzona w menedżerze kolejek MQ25 i została wysłana do dwóch menedżerów kolejek (MQ25 i MQ26). Kanał VT kanału został zatrzymany w każdym menedżerze kolejek.

Komunikaty z komend, które generują komendy z CMDSCOPE

the following command:

```
DEF PRO(V2) QSGDISP(GROUP)
```

Generuje następujące komunikaty:

```
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQM122I !MQ25 CSQMMSGP ' DEF PRO' COMPLETED FOR QSGDISP(GROUP)
CSQN138I !MQ25 'DEFINE PRO' command generated for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DEFINE PRO' command responses from MQ25
CSQ9022I !MQ25 CSQMMSGP ' DEFINE PROCESS' NORMAL COMPLETION
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DEFINE PRO' command responses from MQ26
CSQ9022I !MQ26 CSQMMSGP ' DEFINE PROCESS' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQN122I !MQ25 'DEFINE PRO' command for CMDSCOPE(*) normal completion
```

Te komunikaty mówią, że komenda została wprowadzona w menedżerze kolejek MQ25. Gdy obiekt został utworzony w repozytorium współużytkowanym, wygenerowano inną komendę i wysłano ją do wszystkich aktywnych menedżerów kolejek w grupie współużytkowania kolejek (MQ25 i MQ26).

z/OS

Zarządzanie zasobami IBM MQ w systemie z/OS

Odsyłacze znajdujące się w tym temacie umożliwiają znalezienie sposobu zarządzania zasobami używanymi przez produkt IBM MQ for z/OS, na przykład zarządzanie plikami dzienników, zestawami danych, zestawami stron, pulami buforów i strukturami narzędzia CF.

Użyj poniższych odsyłaczy, aby uzyskać szczegółowe informacje na temat różnych zadań administracyjnych, które mogą być wykonywane podczas korzystania z produktu IBM MQ for z/OS:

- [“Zarządzanie dziennikami” na stronie 375](#)
- [“Zarządzanie zestawem danych programu startowego \(BSDS\)” na stronie 383](#)
- [“Zarządzanie zestawami stron” na stronie 392](#)
- [“Tworzenie kopii zapasowych i odtwarzanie zestawów stron” na stronie 399](#)
- [“Tworzenie kopii zapasowych i odtwarzanie kolejek za pomocą komendy CSQUTIL” na stronie 403](#)
- [“Zarządzanie pulami buforów” na stronie 403](#)
- [“Zarządzanie grupami współużytkowania kolejek i współużytkowanymi kolejkami w systemie z/OS” na stronie 404](#)

Pojęcia pokrewne

[“administrowanie IBM MQ for z/OS” na stronie 329](#)

Administrowanie menedżerami kolejek i powiązаныmi zasobami obejmuje zadania, które często są wykonywane w celu aktywowania zasobów i zarządzania nimi. Wybierz metodę administrowania menedżerami kolejek i powiązаныmi zasobami.

[“Wysyłanie komend do programu IBM MQ for z/OS” na stronie 330](#)

Aby sterować menedżerem kolejek, można użyć komend skryptowych IBM MQ (MQSC) w trybie wsadowym lub interaktywnym.

[“Odtwarzanie i restartowanie w systemie z/OS” na stronie 415](#)

W tym temacie opisano mechanizmy odtwarzania i restartowania używane przez produkt IBM MQ.

Odsyłacze pokrewne

[“Programy narzędziowe serwera IBM MQ for z/OS” na stronie 338](#)

Produkt IBM MQ for z/OS udostępnia zestaw programów narzędziowych, które mogą być pomocne w administrowaniu systemem.

Informacje pokrewne

[IBM MQ for z/OS pojęcia](#)

[Planowanie środowiska IBM MQ w systemie z/OS](#)

[Konfigurowanie menedżerów kolejek w systemie z/OS](#)

[Skorowidz formatów komend programowalnych](#)

[Skorowidz komend MQSC](#)

[Korzystanie z programów narzędziowych IBM MQ for z/OS](#)

Zarządzanie dziennikami

W tym temacie opisano sposób zarządzania plikami dziennika produktu IBM MQ , w tym procesem archiwizacji dziennika, przy użyciu kompresji rekordów dziennika, odtwarzania rekordów dziennika i drukowania rekordów dziennika.

W tym temacie opisano zadania związane z zarządzaniem dziennikami produktu IBM MQ . Zawiera on następujące sekcje:

Archiwizowanie dzienników za pomocą komendy ARCHIVE LOG

Autoryzowany operator może archiwizować bieżące zestawy danych aktywnego dziennika produktu IBM MQ za każdym razem, gdy jest to wymagane przy użyciu komendy ARCHIVE LOG.

Po wywołaniu komendy ARCHIVE LOG program IBM MQ obcina bieżące aktywne zestawy danych dziennika, a następnie uruchamia asynchroniczny proces przenoszenia danych i aktualizuje BSDS w rekordzie procesu przenoszenia.

Komenda ARCHIVE LOG ma opcję MODE (QUIESCE). W przypadku tej opcji zadania i użytkownicy produktu IBM MQ są wyciszani po punkcie zatwierdzania, a wynikowy punkt spójności jest przechwytywany w bieżącym aktywnym dzienniku, zanim zostanie on odciążony.

Rozważmy użycie opcji MODE (QUIESCE) podczas planowania strategii składowania na potrzeby odtwarzania po stronie. Tworzy on systemowy punkt spójności, który minimalizuje liczbę niespójności danych, gdy dziennik archiwum jest używany z najbardziej aktualną kopią zapasową zestawu stron podczas odtwarzania. Na przykład:

```
ARCHIVE LOG MODE(QUIESCE)
```

Jeśli komenda [ARCHIVE LOG](#) zostanie wydana bez podania parametru TIME, wówczas okres wyciszania będzie domyślnie ustawiony na wartość parametru QUIESCE makra CSQ6ARVP . Jeśli czas wymagany dla trybu ARCHIVE LOG MODE (QUIESCE) jest krótszy niż czas określony, komenda zakończy się pomyślnie. W przeciwnym razie wykonanie komendy kończy się niepowodzeniem, gdy upłynie okres. Przedział czasu można określić jawnie za pomocą opcji TIME, na przykład:

```
ARCHIVE LOG MODE(QUIESCE) TIME(60)
```

Ta komenda określa okres wyciszania do 60 sekund przed wystąpieniem w dzienniku ARCHIVE LOG.

Uwaga: Użycie opcji TIME w momencie, gdy czas jest krytyczny, może znacznie zakłócić dostępność produktu IBM MQ dla wszystkich zadań i użytkowników korzystających z zasobów IBM MQ .

Domyślnie komenda jest przetwarzana asynchronicznie od momentu wprowadzenia komendy. (Aby wykonać komendę synchronicznie z innymi komendami IBM MQ , należy użyć opcji WAIT (YES) z opcją QUIESCE, ale należy pamiętać, że konsola z/OS jest zablokowana z poziomu danych wejściowych komendy IBM MQ dla całego okresu QUIESCE.)

W okresie wyciszania:

- Zadania i użytkownicy w menedżerze kolejek mogą przechodzić przez przetwarzanie zatwierdzania, ale są zawieszeni, jeśli próbują zaktualizować dowolny zasób IBM MQ po zatwierdzeniu.
- Może to mieć wpływ na zadania i użytkowników, którzy tylko odczytują dane, ponieważ mogą oczekiwać na blokady wstrzymane przez zadania lub użytkowników, które zostały zawieszono.
- Nowe zadania mogą być uruchamiane, ale nie mogą aktualizować danych.

Dane wyjściowe komendy DISPLAY LOG używają komunikatu CSQV400I w celu wskazania, że wygaszanie jest aktywne. Na przykład:

```
CSQJ322I +CSQ1 DISPLAY LOG report ...
Parameter  Initial value      SET value
-----
INBUFF     60
OUTBUFF    400
MAXRTU     2
MAXARCH    2
TWOACTV    YES
TWOARCH    YES
TWOBSDS    YES
OFFLOAD    YES
MAXCNOFF   0
WRTHRSRSH 20
DEALLCT    0
COMPLLOG   NONE
ZHYWRITE   NO
End of LOG report
CSQJ370I +CSQ1 LOG status report ...
Copy %Full PPRC DSName
 1      68 NO  VICY.CSQ1.LOGCOPY1.DS01
 2      68 NO  VICY.CSQ1.LOGCOPY2.DS01
Restarted at 2014-04-15 09:49:30 using RBA=000000000891B000
Latest RBA=000000000891CCF8
Offload task is AVAILABLE
Full logs to offload - 0 of 4
CSQV400I +CSQ1 ARCHIVE LOG QUIESCE CURRENTLY ACTIVE
CSQ9022I +CSQ1 CSQJC001 ' DISPLAY LOG ' NORMAL COMPLETION
```



Ostrzeżenie: zHyperZapis nie jest włączony w IBM MQ 9.0 , więc jedyną dozwoloną wartością jest *NO* .

Po wyciszeniu wszystkich aktualizacji rekord historii wygaszania w BSDS zostanie zaktualizowany z datą i godziną obciążenia aktywnych zestawów danych dziennika, a następnie z ostatnim zapisanym RBA w bieżącym aktywnym zestawach danych dziennika. Produkt IBM MQ obcina bieżące aktywne zestawy danych dziennika, przełącza się do następnego dostępnego aktywnego zestawu danych dziennika i wysyła komunikat CSQJ311I informujący, że proces odciażający został uruchomiony.

Jeśli aktualizacje nie mogą być wyciszane przed upływem okresu wyciszenia, program IBM MQ wysyła komunikat CSQJ317I, a przetwarzanie dziennika ARCHIVE kończy się. Bieżące aktywne zestawy danych dziennika nie są obcinane ani przełączane na następne dostępne zestawy danych dziennika, a proces odciażania nie jest uruchomiony.

Niezależnie od tego, czy wyciszanie powiodło się, wszyscy zawieszono użytkownicy i zadania zostają wznowione, a program IBM MQ wysyła komunikat CSQJ312I informujący, że wyciszanie zostało zakończone, a działanie aktualizacji jest wznowiane.

Jeśli dziennik ARCHIVE jest generowany, gdy bieżący aktywny dziennik jest ostatnim dostępnym aktywnym zestawem danych dziennika, komenda nie jest przetwarzana, a program IBM MQ wysyła następujący komunikat:

```
CSQJ319I - csect-name CURRENT ACTIVE LOG DATA SET IS THE LAST
AVAILABLE ACTIVE LOG DATA SET. ARCHIVE LOG PROCESSING
WILL BE TERMINATED
```


Jeśli komenda ARCHIVE LOG jest wydawana, gdy inna komenda ARCHIVE LOG jest już w toku, nowa komenda nie jest przetwarzana, a program IBM MQ wysyła następujący komunikat:

```
CSQJ318I - ARCHIVE LOG COMMAND ALREADY IN PROGRESS
```

Więcej informacji na temat komunikatów wysyłanych podczas archiwizacji zawiera sekcja [Komunikaty dla produktu IBM MQ for z/OS](#).

Restartowanie procesu archiwizowania dziennika po awarii

Jeśli wystąpił problem podczas procesu archiwizacji dziennika (na przykład problem z przydzielaniem lub podłączeniem taśm), archiwizacja aktywnego dziennika może zostać zawieszona. Można anulować proces archiwizacji i zrestartować go za pomocą komendy ARCHIVE LOG CANCEL OFFLOAD. Ta komenda anuluje przetwarzanie odciążające aktualnie w toku i restartuje proces archiwizacji. Rozpoczyna się on od najstarszego zestawu danych dziennika, który nie został zarchiwizowany, a następnie przechodzi przez wszystkie aktywne zestawy danych dziennika, które wymagają odciążania. Wszystkie operacje archiwizowania dziennika, które zostały zawieszony, są restartowane.

Tej komendy należy używać tylko wtedy, gdy użytkownik ma pewność, że bieżące zadanie archiwizacji dziennika nie działa, lub jeśli ma zostać zrestartowany poprzednio, próba, która się nie powiodła. Jest to spowodowane tym, że wykonanie komendy może spowodować nieprawidłowe zakończenie zadania przenoszenia, co może spowodować wykonanie zrzutu.

Kontrolowanie archiwizacji i rejestrowania

Za pomocą makr CSQ6LOGP, CSQ6ARVPi CSQ6SYSP można sterować kompresją, drukowaniem, archiwizowaniem, odzyskiwaniem i rejestrowaniem. Należy zauważyć, że zmiany w obiektach prywatnych są rejestrowane tylko w dziennikach produktu IBM MQ. Protokołowane są również zmiany w obiektach GROUP (podobnie jak współużytkowane kanały przychodzące), ponieważ definicje są propagowane wokół grupy i przechowywane lokalnie.

Wiele aspektów archiwizacji i rejestrowania sterowanych jest za pomocą parametrów ustawionych za pomocą makr CSQ6LOGP, CSQ6ARVP i CSQ6SYSP modułu parametrów systemowych, gdy menedżer kolejek jest dostosowany. Szczegółowe informacje na temat tych makr można znaleźć w sekcji [Czynność 17: Krawiec modułu parametrów systemowych](#).

Niektóre z tych parametrów można zmienić, gdy menedżer kolejek jest uruchomiony za pomocą komend IBM MQ MQSC SET LOG, SET SYSTEM i SET ARCHIVE. Są one wyświetlane w programie [Tabela 26](#) na stronie 377:

komenda SET	Parametry
LOG	WRTHRSH, MAXARCH, DEALLCT, MAXRTU, COMPLOG
ARCHIWALNE	Wszystkie
SYSTEM	LOGLOAD

Za pomocą komend MQSC DISPLAY LOG, DISPLAY ARCHIVE i DISPLAY SYSTEM można wyświetlić ustawienia wszystkich parametrów. Te komendy pokazują również informacje o statusie archiwizacji i rejestrowania.

Sterowanie kompresją dziennika

Kompresję rekordów dziennika można włączać i wyłączać za pomocą jednej z tych opcji.

- Komendy SET i DISPLAY LOG w MQSC; patrz Komendy MQSC .
- Wywoływanie interfejsu PCF. Patrz: [“Wprowadzenie do formatów komend programowalnych IBM MQ” na stronie 22](#)
- Korzystanie z makra CSQ6LOGP w module parametrów systemowych; patrz [Korzystanie z komendy CSQ6LOGP](#)

Drukowanie rekordów dziennika

Za pomocą programu narzędziowego CSQ1LOGP można wyodrębnić i wydrukować rekordy dziennika. Odpowiednie instrukcje zawiera sekcja [Program narzędziowy do drukowania dziennika](#).

Odtwarzanie dzienników

Zwykle nie ma potrzeby tworzenia kopii zapasowej i odtwarzania dzienników produktu IBM MQ , zwłaszcza w przypadku korzystania z funkcji rejestrowania podwójnego. Jednak w rzadkich przypadkach, takich jak błąd we/wy w dzienniku, może być konieczne odtworzenie dzienników. Użyj usług metod dostępu, aby usunąć i ponownie zdefiniować zestaw danych, a następnie skopiować do niego odpowiedni podwójny dziennik.

Odrzucanie zestawów danych dziennika archiwalnego

Zestawy danych dziennika archiwalnego można odrzucić, a dzienniki są usuwane automatycznie lub ręcznie.

Należy zachować wystarczającą ilość danych dziennika, aby można było wykonać odzyskiwanie jednostki pracy, odtwarzanie nośników zestawu stron, jeśli zestaw stron został utracony, lub odtwarzanie nośników struktury systemu CF, jeśli utracono strukturę systemu CF. Nie należy usuwać zestawów danych dziennika archiwalnego, które mogą być wymagane do odtwarzania; jeśli te zestawy danych dziennika archiwalnego zostaną usunięte, może nie być możliwe wykonanie wymaganych operacji odtwarzania.

Jeśli potwierdzono, że zestawy danych dziennika archiwalnego mogą być odrzucane, można to zrobić w jeden z następujących sposobów:

- [Automatyczne usuwanie zestawu danych dziennika archiwalnego](#)
- [Ręczne usuwanie zestawów danych dziennika archiwalnego](#)

Automatyczne usuwanie zestawu danych dziennika archiwalnego

W celu automatycznego usuwania zestawów danych dziennika archiwalnego można użyć systemu DASD lub systemu zarządzania taśmami. Okres przechowywania dla zestawów danych dziennika archiwalnego produktu IBM MQ jest określany przez pole okresu przechowywania ARCRETN w makrze instalacji CSQ6ARVP (więcej informacji można znaleźć w sekcji [Używanie komendy CSQ6ARVP](#)).

Wartość domyślna dla okresu przechowywania określa, że dzienniki archiwalne mają być przechowywane przez 9999 dni (maksimum).

Ważne: Istnieje możliwość zmiany okresu przechowywania, ale należy upewnić się, że można dostosować liczbę planowanych cykli składowania.

Produkt IBM MQ używa wartości okresu przechowywania jako wartości parametru RETPD zadania JCL podczas tworzenia zestawów danych dziennika archiwalnego.

Okres przechowywania ustawiony przez podsystem zarządzania pamięcią masową MVS/DFP (SMS) może zostać przestonięty przez ten parametr IBM MQ . Zwykle okres przechowywania jest ustawiony na mniejszą wartość, która jest określona przez produkt IBM MQ lub SMS. Administrator pamięci

masowej i administrator produktu IBM MQ muszą uzgodnić wartość okresu przechowywania, która jest odpowiednia dla produktu IBM MQ.

Uwaga: Produkt IBM MQ nie ma zautomatyzowanej metody usuwania informacji o zestawach danych dziennika archiwalnego z BSDS, ponieważ niektóre systemy zarządzania taśmami udostępniają zewnętrzne ręczne nadpisanie okresów przechowywania. Z tego powodu informacje na temat zestawu danych dziennika archiwalnego mogą nadal znajdować się w BSDS długo po upływie czasu przechowywania zestawu danych, a zestaw danych został zarysowany przez system zarządzania taśmami. Z kolei maksymalna liczba zestawów danych dziennika archiwalnego mogła zostać przekroczona, a dane z serwera BSDS mogły zostać usunięte, zanim zestaw danych osiągnie datę ważności.

Jeśli zestawy danych dziennika archiwalnego są usuwane automatycznie, pamiętaj o tym, że operacja nie aktualizuje listy dzienników archiwalnych w BSDS. Program BSDS można zaktualizować za pomocą programu narzędziowego do spisywania zasobów dziennika zmian, zgodnie z opisem w sekcji [“Zmiana BSDS”](#) na stronie 385. Aktualizacja nie jest niezbędna. Rejestrowanie starych dzienników archiwalnych odpadów jest w BSDS, ale nie ma innej szkody.

Ręczne usuwanie zestawów danych dziennika archiwalnego

Wszystkie rekordy dziennika należy przechowywać tak daleko wstecz, jak najniższy poziom RBA określony w komunikatach CSQIO24I i CSQIO25I. Ten kod RBA jest uzyskiwany za pomocą komendy DISPLAY USAGE, która została wydana podczas tworzenia punktu odtwarzania przy użyciu metody [Metoda 1: pełna kopia zapasowa](#).

Przeczytaj Tworzenie punktu odtwarzania dla zasobów niewspółużytkowanych przed usunięciem dzienników.

Znajdź i odrzuć zestawy danych dziennika archiwalnego

Po ustaleniu minimalnego dziennika RBA wymaganego do odtwarzania można znaleźć zestawy danych dziennika archiwalnego, które zawierają tylko wcześniejsze rekordy dziennika, wykonując następującą procedurę:

1. W celu wydrukowania zawartości BSDS należy użyć programu narzędziowego do tworzenia map dzienników. Przykład danych wyjściowych znajduje się w sekcji [Program narzędziowy do odwzorowywania dziennika drukowania](#).
2. Znajdź sekcje danych wyjściowych zatytułowane ARCHIVE LOG COPY n DATA SETS. Jeśli używane jest podwójne rejestrowanie, istnieją dwie sekcje. Kolumny o etykietach STARTRBA i ENDRBA pokazują zakres RBAs zawartych w każdym tomie. Znajdź woluminy z zakresami, które zawierają minimalny RBA, który został znaleziony przy użyciu komunikatów CSQIO24I i CSQIO25I. Są to najwcześniejsze woluminy, które należy zachować. Jeśli używane jest podwójne rejestrowanie, istnieją dwa takie woluminy.

Jeśli żaden wolumin nie ma odpowiedniego zakresu, zastosowanie ma jeden z następujących przypadków:

- Minimalna wartość RBA nie została jeszcze zarchiwizowana i można usunąć wszystkie woluminy dziennika archiwalnego.
- Lista woluminów dziennika archiwalnego w BSDS zawiniętych wokół, gdy liczba woluminów przekroczyła liczbę dozwoloną przez parametr MAXARCH makra CSQ6LOGP. Jeśli BSDS nie zarejestrują woluminu dziennika archiwalnego, nie można go użyć do odtwarzania. W związku z tym należy rozważyć dodanie informacji o istniejących woluminach do BSDS. Odpowiednie instrukcje można znaleźć w sekcji [“Zmiany w dziennikach archiwalnych”](#) na stronie 388.

Należy również rozważyć zwiększenie wartości MAXARCH. Więcej informacji na ten temat zawiera sekcja [Korzystanie z komendy CSQ6LOGP](#).

3. Usuń dowolny zestaw danych dziennika archiwalnego lub wolumin z wartością ENDRBA, która jest mniejsza niż wartość STARTRBA najwcześniejszego woluminu, który ma być zachowany. Jeśli używane jest podwójne rejestrowanie, należy usunąć oba te kopie.

Ponieważ pozycje BSDS są zawijane, pierwsze kilka pozycji w sekcji dziennika archiwum BSDS może być nowsze niż pozycje na końcu. Spójrz na kombinację daty i czasu i porównaj ich wiek. Nie zakładaj, że można usunąć wszystkie pozycje przed pozycją dla dziennika archiwalnego, który zawiera minimalną wartość LOGRBA.

Usuń zestawy danych. Jeśli archiwa znajdują się na taśmie, kasuj taśmy. Jeśli znajdują się na urządzeniu DASD, należy uruchomić program narzędziowy z/OS, aby usunąć każdy zestaw danych. Następnie, jeśli BSDS ma wyświetlać tylko istniejące woluminy archiwalne, należy użyć narzędzia do zarządzania spisem zasobów (CSQJU003) w celu usunięcia pozycji dla odrzuconych woluminów. Przykład można znaleźć w sekcji [“Zmiany w dziennikach archiwalnych”](#) na stronie 388.

Efekt rozdrabiania dziennika

Długo działające transakcje mogą spowodować rekordy dziennika pracy, które obejmują zestawy danych dziennika. Produkt IBM MQ obsługuje ten scenariusz przy użyciu funkcji śledzenia dziennika. Technika przenosi rekordy dziennika w celu zoptymalizowania ilości zatrzymanych danych dziennika oraz czasu restartu menedżera kolejek.

Gdy jednostka pracy jest uznawana za długa, reprezentacja każdego rekordu dziennika jest zapisywana dalej w dół dziennika. Jest to nazywane *shunting log*. Jest on bardziej szczegółowo opisany w sekcji [Pliki dziennika](#).

Menedżer kolejek używa tych zaciemnionych rekordów dziennika zamiast oryginałów po awarii w celu zapewnienia integralności jednostki pracy. Są do tego dwie korzyści:

- zmniejszono ilość danych w dzienniku, które muszą być zachowane dla jednostki koordynacji pracy;
- mniej danych dziennika musi być przetwarzanych w czasie restartu menedżera kolejek, dlatego menedżer kolejek jest restartowany szybciej

Nieokreślone rekordy dziennika nie zawierają wystarczających informacji na temat operacji odtwarzania nośników.

Dane przechowywane w dzienniku są używane w dwóch różnych celach: odzyskiwanie nośników i jednostka koordynacji pracy. Jeśli wystąpi awaria nośnika, która ma wpływ na strukturę CF lub zestaw stron, menedżer kolejek może odtworzyć nośnik do punktu awarii, odtwarzając wcześniejszą kopię i aktualizując dane znajdujące się w dzienniku. Trwałe działanie wykonywane w jednostce pracy jest rejestrowane w dzienniku, tak aby w przypadku niepowodzenia można było odtworzyć lub usunąć blokady w przypadku zmienionych zasobów. Ilość danych dziennika, które należy zachować w celu włączenia odtwarzania menedżera kolejek, wpływa na te dwa elementy.

Aby możliwe było odtwarzanie nośników, należy zachować wystarczającą ilość danych w dzienniku, aby możliwe było odtwarzanie nośnika z co najmniej najnowszej kopii nośnika i aby można było wycofać się z pamięci. (Twoja strona może zastrzec możliwość odzyskiwania ze starszych kopii zapasowych). W przypadku jednostki integralności pracy należy zachować dane dziennika dla najstarszego w locie lub niepewnych jednostek pracy.

Aby pomóc w zarządzaniu systemem, menedżer kolejek wykrywa stare jednostki pracy w każdym archiwum dziennika i raportuje je w komunikatach CSQJ160 i CSQJ161. Zadanie wewnętrzne odczytuje informacje z dziennika pracy dla tych starych jednostek pracy i ponownie zapisuje je w bardziej związanym formularzu do bieżącej pozycji w dzienniku. Komunikat CSQR026 wskazuje, kiedy to się stało. Komenda MQSC DISPLAY USAGE TYPE TYPE (DATASET) może również pomóc w zarządzaniu zatrzymaniem danych dziennika. Komenda raportuje 3 informacje o odtwarzaniu:

1. Ilość dziennika musi być zachowana na potrzeby odtwarzania jednostki pracy
2. Ilość dziennika musi być zachowana na potrzeby odtwarzania nośników w zestawach stron.
3. dla menedżera kolejek w grupie współużytkowania kolejek, ile dziennika musi być zachowany dla odtwarzania nośników struktur CF

Dla każdego z tych elementów podejmowana jest próba odwzorowania najstarszych danych dziennika wymaganych do zestawu danych. Jako nowe jednostki pracy zaczynają się i zatrzymują się, oczekujemy (1) przejścia na bardziej aktualną pozycję w dzienniku. Jeśli nie jest on poruszany, długo działające

komunikaty UOW ostrzegają o tym, że wystąpił problem. (2) odnosi się do odtwarzania nośnika zestawu stron, jeśli menedżer kolejek ma zostać teraz wyłączony i zrestartowany. Nie wiadomo o tym, kiedy ostatnio utworzono kopię zapasową zestawów stron, lub które mogą być używane, jeśli wystąpiła awaria zestawu stron. Zwykle jest on przenoszony na bardziej aktualną pozycję w dzienniku podczas przetwarzania punktu kontrolnego, ponieważ zmiany przechowywane w pulach buforów są zapisywane w zestawach stron. W (3) menedżer kolejek nie wie o kopiach zapasowych struktury CF podejmowanych w tym menedżerze kolejek lub w innych menedżerach kolejek w grupie współużytkowania kolejek. Jednak odtwarzanie struktury systemu CF wymaga scalenia danych dziennika ze wszystkich menedżerów kolejek w grupie współużytkowania kolejek, które zostały powiązane ze strukturą systemu CF od czasu utworzenia ostatniej kopii zapasowej. Oznacza to, że dane dziennika są identyfikowane za pomocą numeru kolejnego rekordu dziennika (lub numeru LRSN), który jest oparty na datowach i może być stosowany w całej grupie współużytkowania kolejek, a nie na poziomie RBA, który może być różny w różnych menedżerach kolejek w grupie współużytkowania kolejek. Zwykle jest on przenoszony do bardziej aktualnego położenia w dzienniku, ponieważ komendy BACKUP CFSTRUCT są wykonywane na tym lub innych menedżerach kolejek w grupie współużytkowania kolejek.

Resetowanie dziennika menedżera kolejek

Ten temat zawiera informacje na temat resetowania dziennika menedżera kolejek.

Nie należy zezwalać na zawijanie dziennika RBA menedżera kolejek od końca zakresu RBA dziennika do wartości 0, ponieważ prowadzi to do wyłączenia menedżera kolejek, a wszystkie trwałe dane staną się nieodtwarzalne. Koniec dziennika RBA jest albo wartością FFFFFFFF (jeśli używane jest 6-bajtowe RBAs), albo FFFFFFFFFF (jeśli używane są 8-bajtowe RBAs).

Menedżer kolejek wysyła komunikaty [CSQI045I](#), [CSQI046E](#), [CSQI047E](#), [CSQJ031Di](#) [CSQJ032E](#) , aby wskazać, że używany zakres dziennika jest znaczący i że należy zaplanować podjęcie działań, aby uniknąć nieplanowanego wyłączenia.

Menedżer kolejek kończy działanie z kodem przyczyny [00D10257](#) , gdy wartość RBA osiągnie wartość FFF80000000 (jeśli używane są 6-bajtowe dzienniki RBAs) lub FFFFFFFC00000000 (jeśli używane są 8-bajtowe RBAs dziennika).

Jeśli używane są 6-bajtowe RBAs dziennika, należy rozważyć przekształcenie menedżera kolejek tak, aby używany był 8-bajtowy dziennik RBA, a nie resetowanie dziennika menedżera kolejek, postępując zgodnie z procesem opisanym w sekcji [Implementowanie większego względnego adresu bajtu dziennika](#). Przekształcenie menedżera kolejek w taki sposób, aby używało 8-bajtowych dzienników RBA, wymaga krótszego wyłączenia niż zresetowanie dziennika, a następnie wydłuża czas przed zresetowaniem dziennika.

Komunikat [CSQJ034I](#) wydany podczas inicjowania menedżera kolejek wskazuje koniec zakresu RBA dziennika dla menedżera kolejek jako skonfigurowany i może być używany do określenia, czy są używane 6-bajtowe lub 8-bajtowe RBAs dziennika.

Poniżej przedstawiono procedurę resetowania dziennika menedżera kolejek:

1. Rozstrzygnij wszystkie nierozstrzygnięte jednostki pracy. Liczba nierozstrzygniętych jednostek pracy jest wyświetlana w momencie uruchomienia menedżera kolejek w komunikacie CSQR005I jako liczbę WĄTPLIWYCH. W każdym punkcie kontrolnym i przy zamykaniu menedżera kolejek menedżer kolejek automatycznie wysyła komendę

DISPLAY CONN(*) TYPE(CONN) ALL WHERE(UOWSTATE EQ UNRESOLVED) , aby udostępnić informacje na temat nierozstrzygniętych jednostek pracy.

Informacje na temat rozwiązywania jednostek odzyskiwania można znaleźć w sekcji [Rozstrzygnięte jednostki odzyskiwania w wątpliwość](#) . Ostatecznym odwołaniem jest użycie komendy MQSC **RESOLVE INDOUBT** do ręcznego rozstrzygnięcia niepewnych jednostek odzyskiwania.

2. Zamknij menedżera kolejek w sposób czysty.

Można użyć zarówno opcji **STOP QMGR** , jak i **STOP QMGR MODE(FORCE)** , ponieważ obie te komendy opróżnią wszystkie zmienione strony z pul buforów na pagesety.

3. Jeśli menedżer kolejek jest częścią grupy współużytkowania kolejek, należy wykonać kopie zapasowe CFSTRUCT w innych menedżerach kolejek dla wszystkich struktur w grupie współużytkowania kolejek. Zapewnia to, że ostatnie kopie zapasowe nie znajdują się w dzienniku tego menedżera kolejek oraz że ten dziennik menedżera kolejek nie jest wymagany do odtwarzania CFSTRUCT.
4. Zdefiniuj nowe dzienniki i BSDS za pomocą komendy CSQJU003 (więcej informacji na temat korzystania z programu narzędziowego do spisywania zasobów dziennika zmian znajduje się w sekcji [Program narzędziowy do spisywania zasobów dziennika zmian](#)).
5. Uruchom komendę **CSQUTIL RESETPAGE** dla wszystkich zestawów stron dla tego menedżera kolejek (więcej informacji na temat korzystania z tej funkcji zawiera sekcja [Kopiowanie strony i resetowanie dziennika](#)). Należy pamiętać, że zestaw stron RBAs można resetować niezależnie, dlatego można przedstawić wiele współbieżnych zadań (na przykład jeden dla każdego zestawu stron) w celu skrócenie czasu, jaki upłynął dla tego kroku.
6. Restartowanie menedżera kolejek

Pojęcia pokrewne

“Implementowanie większego adresu bajtu względnego w dzienniku” na stronie 382

Przed IBM MQ 8.0produkt IBM MQ for z/OS użył 6-bajtowego dziennika RBA w celu zidentyfikowania położenia danych w dzienniku. W produkcie IBM MQ 8.0dziennik RBA dziennika może mieć długość 8 bajtów, co zwiększa czas sprzed zresetowania dziennika.


Implementowanie większego adresu bajtu względnego w dzienniku


Przed IBM MQ 8.0produkt IBM MQ for z/OS użył 6-bajtowego dziennika RBA w celu zidentyfikowania położenia danych w dzienniku. W produkcie IBM MQ 8.0dziennik RBA dziennika może mieć długość 8 bajtów, co zwiększa czas sprzed zresetowania dziennika.

Ta nowa funkcja musi być jawnie włączona. Informacje na temat włączania 8-bajtowego dziennika RBA zawiera sekcja [Planowanie zwiększenia maksymalnego zakresu dziennika adresowalnego](#) .

Aby włączyć 8-bajtowy dziennik RBA w jednym menedżerze kolejek produktu IBM MQ for z/OS , wykonaj poniższe instrukcje w wyświetlonej kolejności:

1. Włącz IBM MQ 8.0 nowe funkcje za pomocą [OPMODE](#).

 W przypadku menedżerów kolejek w grupie współużytkowania kolejek nie ma potrzeby tworzenia łącznej liczby wyłączeń grupy współużytkowania kolejek. Z kolei można zatrzymać każdy menedżer kolejek, włączyć go dla opcji `OPMODE = (NEWFUNC,800)` lub `OPMODE = (NEWFUNC,900)` i zrestartować go.

 Gdy wszystkie menedżery kolejek w grupie współużytkowania kolejek działają z opcją `OPMODE = (NEWFUNC,800)` lub `OPMODE = (NEWFUNC,900)`, wykonaj następujące kroki dla każdego menedżera kolejek w grupie współużytkowania kolejek do czasu, aż wszystkie menedżery kolejek nie będą uruchomione z nowym BSDS.

2. Przydziel nowe zestawy danych BSDS o podobnych atrybutach do bieżącego BSDS. Można dostosować przykładowy plik CSQ4BSDS i usunąć wszelkie nieistotne instrukcje lub użyć istniejącego zadania JCL, ale zmienić nazwę BSDS na podobny do `++HLQ++.NEW.BSDS01`.

Uwagi:

- a. Sprawdź atrybuty nowego BSDS. Jedynym atrybutem, który może ulec zmianie, jest wielkość BSDS.
- b. Nowe BSDS zawiera więcej danych, które są dostępne dla bieżącego BSDS, dlatego należy upewnić się, że nowe zestawy danych są przydzielane z wystarczającą ilością dostępnego miejsca. Informacje na temat zalecanych wartości podczas definiowania nowego BSDS zawiera sekcja [Planowanie środowiska rejestrowania](#) oraz tematy powiązane z zalecanymi wartościami.
3. Zamknij menedżera kolejek w sposób czysty.
4. Uruchom program narzędziowy [BSDS conversion utility \(CSQJUCNV\)](#) , aby przekształcić istniejące BSDS w nowe zestawy danych BSDS. Wykonanie tej operacji zwykle trwa kilka sekund.

Istniejące BSDS nie zostaną zmienione podczas tego procesu, a użytkownik może użyć go do zainicjowania menedżera kolejek w przypadku niepomyślnej konwersji.

5. Zmień nazwę bieżącego BSDS, aby stał się starym BSDS, a nowy BSDS stanie się bieżącym BSDS, dzięki czemu nowe zestawy danych będą używane przy następnym restarcie menedżera kolejek. Za pomocą komendy DFSMS Access Method Services ALTER można na przykład użyć komendy:

```
ALTER '++HLQ++.BSDS01' NEWNAME('++HLQ++.OLD.BSDS01')
ALTER '++HLQ++.NEW.BSDS01' NEWNAME('++HLQ++.BSDS01')
```

Upewnij się, że wydajesz komendy, aby zmienić nazwy zarówno części danych, jak i indeksu klastra VSAM.

6. Zrestartuj menedżer kolejek. Powinna ona rozpoczynać się w tym samym czasie, co w przypadku korzystania z 6-bajtowego dziennika RBA.

Jeśli menedżer kolejek nie zostanie zrestartowany z powodu niepowodzenia uzyskania dostępu do przekształconego BSDS, spróbuj zidentyfikować przyczynę niepowodzenia, rozwiąż problem i spróbuj ponownie wykonać operację. Jeśli jest to wymagane, skontaktuj się z działem wsparcia produktu IBM, aby uzyskać pomoc.

W razie potrzeby zmiana może zostać wycofana w tym miejscu przez:

- a. Zmiana nazwy bieżącego BSDS, aby stać się nowym BSDS.
- b. Zmiana nazwy starego BSDS, aby stał się bieżącym BSDS.
- c. Restartowanie menedżera kolejek.

Po pomyślnym zrestartowaniu menedżera kolejek przy użyciu przekształconego serwera BSDS nie należy uruchamiać menedżera kolejek przy użyciu starych BSDS.

7. Komunikat CSQJ034I jest generowany podczas inicjowania menedżera kolejek w celu wskazania końca dziennika RBA dla menedżera kolejek jako skonfigurowanego. Potwierdź, że koniec wyświetlanego zakresu dziennika RBA to FFFFFFFFFFFFFFFF. Oznacza to, że używany jest 8-bajtowy dziennik RBA.

Uwaga: Aby umożliwić 8-bajtowy dziennik RBA w nowym menedżerze kolejek produktu IBM MQ 9.0, zanim zostanie on uruchomiony po raz pierwszy, należy najpierw utworzyć pusty format BSDS w wersji 1 i użyć go jako danych wejściowych dla programu narzędziowego do konwersji BSDS w celu utworzenia formatu BSDS w wersji 2. Informacje na temat sposobu przeprowadzania tego procesu znajdują się w sekcji [Tworzenie zestawów danych startowych i zestawów danych dziennika](#).

Informacje pokrewne

[Planowanie zwiększenia maksymalnego zakresu dziennika adresowalnego](#)

[Względny adres w bajtach względnego dziennika](#)

[Program narzędziowy do konwersji BSDS \(CSQJUCNV\)](#)

Zarządzanie zestawem danych programu startowego (BSDS)

Zestaw danych programu startowego (BSDS) jest używany do odwołania do zestawów danych dziennika oraz do rekordów dziennika. W tym temacie opisano sposób sprawdzania, zmiany i odzyskiwania danych BSDS.

Więcej informacji na ten temat zawiera sekcja [Zestaw danych programu startowego](#).

W tym temacie opisano zadania związane z zarządzaniem zestawem danych programu startowego. Zawiera on następujące sekcje:

- [“Znajdowanie informacji o tym, co zawiera BSDS” na stronie 384](#)
- [“Zmiana BSDS” na stronie 385](#)
- [“Odtwarzanie BSDS” na stronie 389](#)

Znajdowanie informacji o tym, co zawiera BSDS

Aby sprawdzić zawartość BSDS, można użyć programu narzędziowego do odwzorowywania dzienników wydruków (CSQJU004).

Program narzędziowy do tworzenia map dzienników (CSQJU004) jest narzędziem wsadowym, które zawiera listę informacji zapisanych w BSDS. Instrukcje na temat jego uruchamiania zawiera sekcja [Program narzędziowy do odwzorowywania dziennika drukowania](#).

BSDS zawiera:

- [Znaczniki czasu](#)
- [Status aktywnego zestawu danych dziennika](#)

Datowniki w BSDS

Dane wyjściowe programu narzędziowego do odwzorowywania dziennika wydruku przedstawiają datowniki, które służą do rejestrowania daty i godziny różnych zdarzeń systemowych, które są przechowywane w BSDS.

W sekcji nagłówka raportu znajdują się następujące znaczniki czasu:

DATOWNIK SYSTEMOWY

Odzwierciedla datę i godzinę ostatniej aktualizacji BSDS. Znacznik czasu BSDS może być aktualizowany, gdy:

- Zostanie uruchomiony menedżer kolejek.
- Próg zapisu jest osiągany podczas wykonywania operacji zapisu w dzienniku. W zależności od określonej liczby buforów wyjściowych i szybkości działania systemu, BSDS może być aktualizowane kilka razy w ciągu sekundy lub może nie być aktualizowane przez kilka sekund, minut lub nawet godzin. Szczegółowe informacje na temat progu zapisu można znaleźć w parametrze WRTHRSZ makra CSQ6LOGP w sekcji [Korzystanie z komendy CSQ6LOGP](#).
- IBM MQ spada do jednego trybu BSDS z normalnego trybu z podwójnym trybem BSDS z powodu błędu. Taka operacja może wystąpić, gdy żądanie pobrania, wstawienia, punktu, aktualizacji lub usunięcia rekordu BSDS nie powiedzie się. Jeśli wystąpi ten błąd, program IBM MQ aktualizuje znacznik czasu w pozostałych BSDS w celu wymuszenia niezgodności znacznika czasu z wyłączonym BSDS.

DATOWNIK PROGRAMU NARZĘDZIOWEGO

Data i godzina zmiany zawartości BSDS przez program narzędziowy do zarządzania spisem dziennika zmian (CSQJU003).

Następujące znaczniki czasu są dołączane do części raportu aktywnego i archiwalnego zestawu danych dziennika:

Data aktywnego dziennika

Data utworzenia aktywnego wpisu dziennika w bazie danych BSDS, czyli po zakończeniu operacji NEWLOG CSQJU003 .

Czas aktywnego dziennika

Czas utworzenia aktywnej pozycji dziennika w BSDS, czyli po zakończeniu operacji NEWLOG CSQJU003 .

Data dziennika archiwalnego

Data utworzenia wpisu dziennika archiwalnego w BSDS, czyli w momencie, gdy wykonano komendę CSQJU003 NEWLOG lub gdy archiwum zostało wykonane.

Czas dziennika archiwalnego

Czas utworzenia wpisu dziennika archiwalnego w BSDS, czyli w momencie, gdy wykonano komendę CSQJU003 NEWLOG lub gdy archiwum zostało wykonane.

Status aktywnego zestawu danych dziennika

BSDS rejestruje status aktywnego zestawu danych dziennika jako jeden z następujących:

Nowy

Zestaw danych został zdefiniowany, ale nigdy nie był używany przez produkt IBM MQ, lub dziennik został obcięty do punktu przed pierwszym użyciu zestawu danych. W obu przypadkach zestaw danych rozpoczynający i kończący wartości RBA jest resetowany do zera.

REUSABLE

Zestaw danych został zdefiniowany, ale nigdy nie był używany przez produkt IBM MQ, albo zestaw danych został odciążony. W danych wyjściowych odwzorowania dziennika wydruku wartość początkowa RBA dla ostatniego zestawu danych REUSABLE jest równa wartości początkowej wartości RBA ostatniego zestawu danych dziennika archiwalnego.

NIE REUSABLE

Zestaw danych zawiera rekordy, które nie zostały odciążone.

ZATRZYMANE

Procesor odciążający napotkał błąd podczas odczytywania rekordu i nie można było uzyskać tego rekordu z drugiej kopii aktywnego dziennika.

obcięte

Albo:

- Wystąpił błąd we/wy, a program IBM MQ zatrzymał zapis do tego zestawu danych. Aktywny zestaw danych dziennika jest odciążony, począwszy od uruchomienia RBA, a następnie kontynuuje się do ostatniego poprawnego segmentu rekordu w obciętym zestawie danych aktywnego dziennika. RBA ostatniego poprawnego segmentu rekordu jest niższa niż końcowy RBA zestawu danych aktywnego dziennika. Protokołowanie jest przełączane do następnego dostępnego aktywnego zestawu danych dziennika i jest kontynuowane bez nieprzerwanego działania.

lub wersji

- Wywołano funkcję ARCHIVE LOG, która obcięła aktywny dziennik.

Status zostanie wyświetlony w danych wyjściowych narzędzia mapy dziennika wydruku.

Zmiana BSDS

Nie ma potrzeby wykonywania specjalnych kroków w celu zachowania aktualizacji BSDS z rekordami zdarzeń rejestrowania, ponieważ program IBM MQ automatycznie wykonuje te czynności.

Można jednak zmienić BSDS, jeśli jest to możliwe z następujących działań:

- Dodaj więcej aktywnych zestawów danych dziennika.
- Kopiowanie zestawów danych aktywnego dziennika do nowo przydzielonych zestawów danych, na przykład przy udostępniających większe aktywne przydziały dziennika.
- Przenoszenie zestawów danych dziennika do innych urządzeń.
- Odzyskaj uszkodzone BSDS.
- Odrzuć przestarzałe zestawy danych dziennika archiwalnego.

Serwer BSDS można zmienić, uruchamiając program narzędziowy do rejestrowania zmian w dzienniku (CSQJU003). Ten program narzędziowy należy uruchomić tylko wtedy, gdy menedżer kolejek jest nieaktywny lub można uzyskać niespójne wyniki. Działanie programu narzędziowego jest kontrolowane przez instrukcje w zestawie danych SYSIN. W tej sekcji przedstawiono kilka przykładów. Pełne instrukcje można znaleźć w sekcji [Program narzędziowy do rejestrowania dziennika zmian](#).

Zestaw danych aktywnego dziennika można skopiować tylko wtedy, gdy menedżer kolejek jest nieaktywny, ponieważ program IBM MQ przydziela aktywne zestawy danych dziennika jako wyłączne (DISP = OLD) podczas uruchamiania menedżera kolejek.

W tym temacie opisano sposób zmiany aktywnych dzienników za pomocą BSDS.

Za pomocą programu narzędziowego do rejestrowania zmian można dodawać, usuwać i zapisywać pozycje w BSDS dla aktywnych dzienników. Przykłady są wyświetlane tylko w tym miejscu. W tym miejscu należy zastąpić nazwy zestawów danych wyświetlane z tymi, które mają być używane. Więcej informacji na temat programu narzędziowego znajduje się w sekcji Program narzędziowy do rejestrowania dziennika zmian.

Więcej informacji na ten temat zawierają następujące sekcje:

- [Dodawanie pozycji rekordów do BSDS](#)
- [Usuwanie informacji o aktywnym zestawie danych dziennika z bazy danych BSDS](#)
- [Rejestrowanie informacji na temat zestawu danych dziennika w BSDS](#)
- [Zwiększanie wielkości aktywnego dziennika](#)
- [korzystanie z CSQJUFMT](#)

Dodawanie wpisów akt do BSDS

Jeśli aktywny dziennik został oznaczony jako "zatrzymany", nie jest ponownie wykorzystywany do rejestrowania, ale nadal jest używany do odczytu. Za pomocą usług metod dostępu można zdefiniować nowe aktywne zestawy danych dziennika, a następnie użyć programu narzędziowego do rejestrowania zasobów dziennika zmian w celu zarejestrowania nowych zestawów danych w BSDS. Na przykład:

```
NEWLOG DSNAME=MQM111.LOGCOPY1.DS10,COPY1
NEWLOG DSNAME=MQM111.LOGCOPY2.DS10,COPY2
```

Jeśli zawartość starego aktywnego zestawu danych dziennika jest kopiowana do nowego, można również podać zakres RBA oraz znaczniki czasu początkowego i końcowego w funkcji NEWLOG.

Usuwanie informacji o aktywnym zestawie danych dziennika z BSDS

Aby usunąć informacje o aktywnym zestawie danych dziennika z serwera BSDS, można użyć następujących elementów:

```
DELETE DSNAME=MQM111.LOGCOPY1.DS99
DELETE DSNAME=MQM111.LOGCOPY2.DS99
```

Rejestrowanie informacji o zestawie danych dziennika w BSDS

Aby zarejestrować informacje na temat istniejącego aktywnego zestawu danych dziennika w BSDS, należy użyć:

```
NEWLOG DSNAME=MQM111.LOGCOPY1.DS10,COPY2,STARTIME=19930212205198,
ENDTIME=19930412205200,STARTRBA=6400,ENDRBA=94FF
```

Może być konieczne wstawienie rekordu zawierającego ten typ informacji w BSDS, ponieważ:

- Pozycja dla zestawu danych została usunięta, ale jest potrzebna ponownie.
- Kopiujesz zawartość jednego aktywnego zestawu danych do innego zestawu danych.
- Odtwarzasz BSDS z kopii zapasowej.

Zwiększanie wielkości aktywnego dziennika

Istnieją dwie metody osiągnięcia tego procesu.

1. Gdy menedżer kolejek jest aktywny:

- a. Zdefiniuj nowe większe zestawy danych dziennika za pomocą JCL.
- b. Za pomocą komendy MQSC DEFINE LOG dodaj nowe zestawy danych dziennika do aktywnego menedżera kolejek.
- c. Użyj komendy MQSC ARCHIVE LOG, aby przenieść bieżący aktywny dziennik, aby był nowym, większym logem.
- d. Poczekaj na zakończenie archiwizacji mniejszego aktywnego zestawu danych dziennika.
- e. Zamknij menedżer kolejek za pomocą programu narzędziowego CSQJU003 w celu usunięcia starych, małych aktywnych dzienników.
- f. Zrestartuj menedżer kolejek.

2. Gdy menedżer kolejek jest nieaktywny:

- a. Zatrzymaj menedżer kolejek. Ten krok jest wymagany, ponieważ program IBM MQ przydziela wszystkie aktywne zestawy danych dziennika do wyłącznego użytku w przypadku, gdy jest on aktywny.
- b. Użyj opcji Access Method Services ALTER z opcją NEWNAME, aby zmienić nazwy aktywnych zestawów danych dziennika.
- c. Aby zdefiniować większe aktywne zestawy danych dziennika, należy użyć funkcji DEFINE usług metod dostępu.

Ponowne użycie starych nazw zestawów danych nie jest konieczne, aby możliwe było uruchamianie programu narzędziowego do spisywania zasobów dziennika zmian w celu ustanowienia nowych nazw w BSDSs. Stare nazwy zestawów danych i poprawne zakresy RBA znajdują się już w BSDSs.

- d. Użyj programu Access Method Services REPRO, aby skopiować stare (zmienione nazwy) zestawy danych do ich odpowiednich nowych zestawów danych.

Uwaga: Ten krok może zająć dużo czasu, dlatego przedsiębiorstwo może nie działać w tym okresie.

- e. Uruchom menedżer kolejek.

Jeśli wszystkie zestawy danych dziennika mają taką samą wielkość, system będzie działał bardziej spójnie i wydajnie. Jeśli zestawy danych dziennika nie mają takiej samej wielkości, trudniej jest śledzić dzienniki systemu, a więc można je zmarnować.

Korzystanie z CSQJUFMT

Podczas zwiększania rozmiaru aktywnego dziennika nie należy uruchamiać formatu CSQJUFMT.

W przypadku uruchomienia funkcji CSQJUFMT (w celu zapewnienia korzyści wydajności przy pierwszym zapisaniu menedżera kolejek w nowym aktywnym dzienniku) otrzymujesz komunikaty:

```
IEC070I 203-204,XS95GTLX,REPRO02,OUTPUT,B857,SPMG02, 358
IEC070I MG.W.MG4E.LOGCOPY1.DS02,MG.W.MG4E.LOGCOPY1.DS02.DATA,
IDC3302I ACTION ERROR ON MG.W.MG4E.LOGCOPY1.DS02
IDC3351I ** VSAM I/O RETURN CODE IS 28 - RPLFDBWD = X'2908001C'
IDC31467I MAXIMUM ERROR LIMIT REACHED.
```

```
IDC0005I NUMBER OF RECORDS PROCESSED WAS 0
```

Dodatkowo, jeśli korzystasz z usługi Access Method Services REPRO, upewnij się, że zdefiniujesz nowy pusty dziennik.

Jeśli do kopiowania starego (o zmienionej nazwie) zestawu danych do jego nowego zestawu danych jest używana opcja REPRO, wartością domyślną jest NOREPLACE.

Oznacza to, że REPRO nie zastępuje rekordu, który już znajduje się na wyznaczonym zbiorze danych. Gdy formatowanie jest wykonywane w zestawie danych, wartość RBA jest resetowana. Wynik netto to zestaw danych, który nie jest pusty po sformatowaniu.

W tym temacie opisano sposób zmiany dzienników archiwalnych.

Użytkownik może dodawać, usuwać i zmieniać hasło wpisów w BSDS dla dzienników archiwalnych. Przykłady są wyświetlane tylko w tym miejscu. W tym miejscu należy zastąpić nazwy zestawów danych wyświetlane z tymi, które mają być używane. Więcej informacji na temat programu narzędziowego znajduje się w sekcji [Program narzędziowy do rejestrowania dziennika zmian](#).

- [Dodawanie dziennika archiwalnego](#)
- [Usuwanie dziennika archiwalnego](#)
- [Zmiana hasła dziennika archiwalnego](#)

Dodawanie dziennika archiwalnego

Gdy odtwarzanie obiektu zależy od odczytu istniejącego zestawu danych dziennika archiwalnego, BSDS musi zawierać informacje na temat tego zestawu danych, aby program IBM MQ mógł go znaleźć. Aby zarejestrować informacje na temat istniejącego zestawu danych dziennika archiwalnego w BSDS, należy użyć następującej składni:

```
NEWLOG DSN=CSQARC1.ARCHLOG1.E00021.T2205197.A0000015,COPY1VOL=CSQV04,  
UNIT=TAPE,STARTRBA=3A190000,ENDRBA=3A1F0FFF,CATALOG=NO
```

Usuwanie dziennika archiwalnego

Aby usunąć cały zestaw danych dziennika archiwalnego dla jednego lub większej liczby woluminów, należy użyć następującej komendy:

```
DELETE DSN=CSQARC1.ARCHLOG1.E00021.T2205197.A0000015,COPY1VOL=CSQV04
```

Zmiana hasła do dziennika archiwalnego

Jeśli zostanie zmienione hasło istniejącego zestawu danych dziennika archiwalnego, należy również zmienić informacje w BSDS.

1. Wyświetl listę BSDS, korzystając z programu narzędziowego do tworzenia map dzienników.
2. Usuń wpis dla zestawu danych dziennika archiwalnego ze zmienionym hasłem, używając funkcji DELETE programu narzędziowego CSQJU003 (patrz temat [Program narzędziowy do rejestrowania dziennika zmian](#)).
3. Nadaj nazwę zestawu danych dla nowego zestawu danych dziennika archiwalnego. Użyj funkcji NEWLOG programu narzędziowego CSQJU003 (patrz temat [Program narzędziowy do spisywania zasobów dziennika zmian](#)), a następnie podaj nowe hasło, uruchamianie i kończenie RBA oraz numery seryjne woluminów (które można znaleźć w programie narzędziowym mapy dziennika wydruku, patrz [Program narzędziowy do mapowania dziennika drukowania](#)).

Aby zmienić hasło dla nowych zestawów danych dziennika archiwalnego, należy użyć następującej komendy:

```
ARCHIVE PASSWORD= password
```

Aby zatrzymać umieszczanie haseł w nowych zestawach danych dziennika archiwalnego, należy użyć:

ARCHIVE NOPASSWD

Uwaga: Funkcji programu narzędziowego ARCHIVE należy używać tylko wtedy, gdy zewnętrzny menedżer zabezpieczeń nie jest dostępny.

z/OS

Zmiana kwalifikatora wysokiego poziomu (HLQ) dla dzienników i BSDS

Ten temat zawiera opis procedury wymaganej do zmiany kwalifikatora wysokiego poziomu (HLQ).

Zanim rozpocznie

Menedżer kolejek musi zostać zakończony normalnie przed skopiowaniem dowolnego z dzienników lub zestawów danych do nowych zestawów danych. Ma to na celu zapewnienie spójności danych i podczas restartowania nie jest wymagane żadne działanie.

O tym zadaniu

To zadanie udostępnia informacje o tym, jak zmienić HLQ dla dzienników i BSDS. W tym celu wykonaj następujące działania:

Procedura

1. Uruchom program narzędziowy do drukowania dziennika CSQJU004 , aby zarejestrować informacje o zestawie danych dziennika. Te informacje są potrzebne później.
2. Istnieją dwie możliwości:
 - a) uruchomienie tworzenia i odtwarzania kopii zapasowych DSS wraz z zmianą nazwy w zestawach danych dziennika i BSDS, które mają być przemianowane, lub
 - b) użyj opcji AMS DEFINE i REPRO, aby utworzyć zestawy danych HLQ i skopiować dane ze starych zestawów danych.
3. Zmodyfikuj procedury MSTR i CHIN w taki sposób, aby wskazywały na nowe zestawy danych.
4. Usuń stare informacje z dziennika w nowej kopii BSDS za pomocą CSQJU003.
5. Zdefiniuj nowe zestawy danych dziennika do nowego BSDS, korzystając z funkcji NEWLOG CSQJU003. Wszystkie informacje o każdym dzienniku należy przechowywać w taki sam sposób, poza HLQ.
6. Nowe BSDS powinny odzwierciedlać te same informacje, które zostały zarejestrowane dla starych dzienników w starych BSDS.
HLQ powinna być jedyną rzeczą, która się zmienia.

Co dalej

Przed uruchomieniem menedżera kolejek porównaj dane wyjściowe CSQJU004 dla starego i nowego BSDS, aby upewnić się, że wyglądały dokładnie takie same (z wyjątkiem HLQs).

Uwaga: Podczas wykonywania tych operacji należy zachować ostrożność. Niepoprawne działania mogą prowadzić do sytuacji niemożliwych do naprawienia. Sprawdź dane wyjściowe narzędzia PRINT LOG MAP UTILITY i upewnij się, że uwzględniono wszystkie informacje wymagane do odtwarzania lub restartowania.

z/OS

Odtwarzanie BSDS

Jeśli program IBM MQ działa w trybie podwójnego BSDS i jeden BSDS zostanie uszkodzony, wymuszenie IBM MQ w trybie pojedynczego BSDS spowoduje, że program IBM MQ będzie działał bez problemu (do następnego restartu).

Aby przywrócić środowisko do podwójnego trybu BSDS:

1. Użyj programu Access Method Services, aby zmienić nazwę uszkodzonego BSDS lub usunąć uszkodzone BSDS i zdefiniować nowe BSDS o tej samej nazwie, co uszkodzony BSDS. Przykładowe instrukcje sterujące można znaleźć w zadaniu CSQ4BREC w pliku thlqual.SCSQPROC.
2. Wydadaj komendę IBM MQ RECOVER BSDS, aby utworzyć kopię poprawnego typu BSDS w nowo przydzielonym zestawie danych i przywrócić podwójny tryb BSDS.

Jeśli produkt IBM MQ działa w trybie pojedynczego BSDS i jeśli BSDS jest uszkodzony lub jeśli program IBM MQ działa w trybie podwójnego BSDS i oba BSDSs są uszkodzone, menedżer kolejek zostanie zatrzymany i nie zostanie zrestartowany, dopóki zestawy danych BSDS nie zostaną naprawione. W tym przypadku:

1. Znajdź plik BSDS powiązany z najnowszym zestawem danych dziennika archiwalnego. Nazwa zestawu danych najnowszego dziennika archiwalnego pojawia się w protokole zadania w ostatnim wystąpieniu komunikatu CSQJ003I, co oznacza, że przetwarzanie offload zostało pomyślnie zakończone. W ramach przygotowań do dalszej części tej procedury, dobrym zwyczajem jest zachowanie dziennika wszystkich udanych archiwów, które zostały odnotowane przez ten komunikat:

- Jeśli dzienniki archiwalne znajdują się w pamięci DASD, BSDS jest przydzielany na dowolnym dostępnym urządzeniu DASD. Nazwa BSDS jest taka sama jak odpowiadająca jej nazwa zestawu danych dziennika archiwalnego. Zmień tylko pierwszą literę ostatniego kwalifikatora, od A do B, tak jak w tym przykładzie:

Nazwa dziennika archiwalnego

CSQ.ARCHLOG1. **A** 0000001

Nazwa kopii BSDS

CSQ.ARCHLOG1. **B** 0000001

- Jeśli dzienniki archiwalne znajdują się na taśmie, BSDS jest pierwszym zestawem danych pierwszego woluminu dziennika archiwalnego. BSDS nie jest powtarzane w późniejszych woluminach.
2. Jeśli najnowszy zestaw danych dziennika archiwalnego nie ma kopii BSDS (na przykład, ponieważ wystąpił błąd podczas odciażania go), znajdź wcześniejszą kopię BSDS z wcześniejszego przetwarzania odciażania.
 3. Zmień nazwę *uszkodzonych* BSDSs za pomocą komendy Access Method Services ALTER z opcją NEWNAME. Jeśli chcesz usunąć uszkodzone BSDS, użyj komendy Access Method Services DELETE. Dla każdego uszkodzonego BSDS należy użyć funkcji Access Method Services, aby zdefiniować nowy zestaw danych BSDS jako zastępujący zestaw danych. Zadanie CSQ4BREC w pliku thlqual.SCSQPROC zawiera instrukcje kontroli usług metod dostępu w celu zdefiniowania nowego BSDS.
 4. Użyj komendy REPRO usług metod dostępu, aby skopiować BSDS z dziennika archiwalnego do jednego z zastępczych BSDSs, które zostały zdefiniowane w kroku [“3” na stronie 390](#). Nie należy kopiować żadnych danych do drugiego zastąpienia BSDS, należy to zrobić w kroku [“5” na stronie 392](#).

- a. Wydrukuj zawartość zastępczego BSDS.

Użyj programu narzędziowego do tworzenia map dzienników (CSQJU004) w celu wydrukowania treści zastępczych BSDS. Umożliwia to przejrzanie treści zastępczego BSDS przed kontynuowaniem pracy z odzyskiwaniem.

- b. Zaktualizuj spis zasobów zestawu danych dziennika archiwalnego w zastępczym BSDS.

Przeanalizuj dane wyjściowe z programu narzędziowego do odwzorowywania dzienników wydruków i sprawdź, czy zastępujący BSDS nie zawiera rekordu dziennika archiwalnego, z którego kopiowany został BSDS. Jeśli zamiana BSDS jest starą kopią, jej stan magazynowy może nie zawierać wszystkich zestawów danych dziennika archiwalnego, które zostały utworzone ostatnio. Spis zasobów BSDS w zestawach danych dziennika archiwalnego musi zostać zaktualizowany w celu odzwierciedlenia bieżącego spisu zasobów podsystemu.

Użyj instrukcji zmiany magazynu dziennika (CSQJU003) NEWLOG, aby zaktualizować zastępcze BSDS, dodając rekord dziennika archiwalnego, z którego kopiowany został BSDS. Jeśli zestaw danych dziennika archiwalnego jest chroniony hasłem, należy użyć opcji PASSWORD funkcji NEWLOG. Ponadto, jeśli zestaw danych dziennika archiwalnego jest wpisany do katalogu, upewnij się, że opcja CATALOG funkcji NEWLOG jest poprawnie ustawiona na CATALOG=YES. Użyj instrukcji

NEWLOG, aby dodać dodatkowe zestawy danych dziennika archiwalnego, które zostały utworzone później niż w kopii BSDS.

c. Zaktualizuj hasła w zastępczym BSDS.

BSDS zawiera hasła dla zestawów danych dziennika archiwalnego oraz dla aktywnych zestawów danych dziennika. Aby upewnić się, że hasła w zastępczym BSDS odzwierciedlają bieżące hasła używane przez instalację, należy użyć funkcji programu narzędziowego ARCHIVE dla dziennika zmian z opcją PASSWORD.

d. Zaktualizuj spis aktywnych zestawów danych dziennika w zastępczym BSDS.

W nietypowych okolicznościach instalacja mogła zostać dodana, usunięta lub nazwa aktywnego zestawu danych dziennika, ponieważ kopiowanie BSDS zostało skopiowane. W takim przypadku zastąpienie BSDS nie odzwierciedla rzeczywistej liczby lub nazw aktywnych zestawów danych dziennika, które są obecnie używane przez instalację.

Jeśli konieczne jest usunięcie aktywnego zestawu danych dziennika z wymiany zasobów dziennika BSDS, należy użyć funkcji DELETE programu narzędziowego do spisywania zasobów dziennika.

Jeśli konieczne jest dodanie aktywnego zestawu danych dziennika do wymiany zasobów dziennika BSDS, należy użyć funkcji NEWLOG programu narzędziowego do wymiany dziennika zmian. Upewnij się, że zakres RBA został poprawnie określony w funkcji NEWLOG. Jeśli zestaw danych aktywnego dziennika jest chroniony hasłem, należy użyć opcji PASSWORD.

Jeśli konieczna jest zmiana nazwy aktywnego zestawu danych dziennika w zastępczym zapasu dziennika BSDS, należy użyć funkcji DELETE programu narzędziowego do spisywania zasobów dziennika, a następnie funkcji NEWLOG. Upewnij się, że zakres RBA został poprawnie określony w funkcji NEWLOG. Jeśli zestaw danych aktywnego dziennika jest chroniony hasłem, należy użyć opcji PASSWORD.

e. Zaktualizuj aktywne zakresy RBA dziennika w zastępczym BSDS.

Później, gdy menedżer kolejek zostanie zrestartowany, porównuje RBAs aktywnych zestawów danych dziennika wymienionych w BSDS z RBAs znalezionym w rzeczywistych zestawach danych aktywnego dziennika. Jeśli wartość RBAs nie jest akceptowana, menedżer kolejek nie zostanie zrestartowany. Problem jest powiększony, gdy używana jest stara kopia BSDS. Aby rozwiązać ten problem, należy użyć programu narzędziowego do spisywania zasobów dziennika zmian (CSQJU003) w celu dostosowania strategii RBAs znalezionych w BSDS przy użyciu RBAs w rzeczywistych zestawach danych aktywnego dziennika. W tym celu wykonaj następujące czynności:

- Za pomocą programu narzędziowego do rejestrowania rekordów dziennika (CSQ1LOGP) do wydrukowania raportu podsumowujących aktywny zestaw danych dziennika. Powoduje to wyświetlenie uruchamiania i kończenia RBAs.
- Porównywanie rzeczywistych zakresów RBA z zakresami RBA, które zostały właśnie wydrukowane, gdy znane są dane RBAs wszystkich aktywnych zestawów danych dziennika.

Jeśli zakresy RBA są równe dla wszystkich aktywnych zestawów danych dziennika, można przejść do następnego kroku odtwarzania bez żadnych dodatkowych prac.

Jeśli zakresy RBA nie są równe, dopasuj wartości w BSDS, aby odzwierciedlić rzeczywiste wartości. Dla każdego aktywnego zestawu danych dziennika, który wymaga dopasowania zakresu RBA, należy użyć funkcji zmiany w programie narzędziowym spisywania zasobów dziennika DELETE, aby usunąć zestaw danych aktywnego dziennika z magazynu w zastępczym BSDS. Następnie należy użyć funkcji NEWLOG, aby ponownie zdefiniować zestaw danych aktywnego dziennika do BSDS. Jeśli zestawy danych aktywnego dziennika są zabezpieczone hasłem, należy użyć opcji PASSWORD funkcji NEWLOG.

f. Jeśli dla każdej kopii aktywnego dziennika określono tylko dwa aktywne zestawy danych dziennika, IBM MQ może mieć trudności podczas restartu menedżera kolejek. Problem może wystąpić, gdy jeden z aktywnych zestawów danych dziennika jest pełny i nie został on odciążony, podczas gdy drugi aktywny zestaw danych dziennika jest bliski zapełnienia. W takim przypadku należy dodać

nowy zestaw danych aktywnego dziennika dla każdej kopii aktywnego dziennika i zdefiniować każdy nowy aktywny zestaw danych dziennika w zastępczym magazynie dziennika BSDS.

Użyj komendy Access Method Services DEFINE, aby zdefiniować nowy aktywny zestaw danych dziennika dla każdej kopii aktywnego dziennika i użyć funkcji NEWLOG programu narzędziowego do wymiany dziennika zmian, aby zdefiniować nowe aktywne zestawy danych dziennika w zastępczym BSDS. Nie ma potrzeby określania zakresów RBA w instrukcji NEWLOG. Jeśli jednak aktywne zestawy danych dziennika są zabezpieczone hasłem, należy użyć opcji PASSWORD funkcji NEWLOG. Przykładowe instrukcje sterujące umożliwiające wykonanie tego zadania można znaleźć w zadaniu CSQ4LREC w pliku thlqual.SCSQPROC.

5. Skopiuj zaktualizowane BSDS do drugiego nowego zestawu danych BSDS. Pliki BSDSs są teraz identyczne.

Użyj programu narzędziowego do tworzenia map dzienników (CSQJU004) w celu wydrukowania zawartości drugiego zamiennika BSDS w tym punkcie.

6. Więcej informacji na temat utraty bieżącego aktywnego zestawu danych dziennika zawiera sekcja [Problemy z dziennikiem aktywnym](#).
7. Zrestartuj menedżer kolejek przy użyciu nowo skonstruowanych BSDS. IBM MQ określa bieżącą wartość RBA oraz informacje o tym, jakie dzienniki aktywne muszą zostać zarchiwizowane.

Zarządzanie zestawami stron

Ten temat zawiera informacje na temat zarządzania zestawami stron powiązanych z menedżerem kolejek.

W tej sekcji opisano sposób dodawania, kopiowania i ogólnie zarządzania zestawami stron powiązanych z menedżerem kolejek. Zawiera on następujące sekcje:

- [“Jak zmienić kwalifikator wysokiego poziomu \(HLQ\) dla zestawów stron” na stronie 392](#)
- [“Jak dodać zestaw stron do menedżera kolejek” na stronie 393](#)
- [“Co zrobić, gdy jeden z zestawów stron stanie się pełny” na stronie 393](#)
- [“Jak równoważyć obciążenia na zestawach stron” na stronie 394](#)
- [Jak zwiększyć rozmiar zestawu stron](#)
- [“Jak zmniejszyć zestaw stron” na stronie 398](#)
- [“Jak ponownie wprowadzić zestaw stron” na stronie 398](#)
- [“Tworzenie kopii zapasowych i odtwarzanie zestawów stron” na stronie 399](#)
- [“Usuwanie zestawów stron” na stronie 403](#)
- [“Tworzenie kopii zapasowych i odtwarzanie kolejek za pomocą komendy CSQUTIL” na stronie 403](#)

Sekcja [Zestawy stron](#) zawiera opis zestawów stron, klas pamięci masowej, buforów i pul buforów oraz niektórych uwag dotyczących wydajności, które mają zastosowanie.

Jak zmienić kwalifikator wysokiego poziomu (HLQ) dla zestawów stron

To zadanie zawiera informacje na temat sposobu zmiany wartości HLQ dla zestawów stron. Aby wykonać tę czynność, wykonaj następujące czynności:

1. Zdefiniuj nowe zestawy stron HLQ.
2. Jeśli przydział wielkości jest taki sam, jak stary zestaw stron, skopiuj istniejący zestaw stron za pomocą komendy REPRO do pustych nowych zestawów stron HLQ. W przypadku zwiększania rozmiaru zestawów stron należy użyć funkcji FORMAT CSQUTIL, aby sformatować zestaw stron docelowych. Więcej informacji na ten temat zawiera sekcja [Formatowanie zestawów stron \(FORMAT\)](#).

3. Użyj funkcji COPYPAGE programu CSQUTIL, aby skopiować wszystkie komunikaty ze strony źródłowej ustawionej na docelowy zestaw stron. Więcej informacji na ten temat zawiera sekcja [Expanding a page set \(COPYPAGE\)](#).
4. Zmień instrukcję DD CSQP00xx w procedurze menedżera kolejek w taki sposób, aby wskazywała na nowe zestawy stron HLQ.

Zrestartuj menedżer kolejek i zweryfikuj zmiany w zestawach stron.

Jak dodać zestaw stron do menedżera kolejek

W tym opisie założono, że istnieje już menedżer kolejek, który jest już uruchomiony. Może być konieczne dodanie zestawu stron, jeśli na przykład menedżer kolejek musi radzić sobie z nowymi aplikacjami przy użyciu nowych kolejek.

Aby dodać nowy zestaw stron, wykonaj następującą procedurę:

1. Zdefiniuj i sformatuj nowy zestaw stron. Przykładowego kodu JCL można użyć jako podstawy do użycia w pliku thlqual.SCSQPROC(CSQ4PAGE). Więcej informacji na ten temat zawiera sekcja [Formatowanie zestawów stron \(FORMAT\)](#).
Zadbaj o to, aby nie formatować żadnych zestawów stron, które są w użyciu, chyba że jest to, co zamierzasz. Jeśli tak, użyj opcji FORCE funkcji programu narzędziowego FORMAT.
2. Aby powiązać zestaw stron z pulą buforów, należy użyć komendy DEFINE PSID z opcją DSN.
3. Dodaj odpowiednie definicje klas pamięci masowej dla zestawu stron, wydając komendy DEFINE STGCLASS.
4. Opcjonalnie, aby udokumentować sposób konfigurowania menedżera kolejek:
 - a. Dodaj nowy zestaw stron do procedury uruchomionego zadania dla menedżera kolejek.
 - b. Dodaj definicję dla nowego zestawu stron do zestawu danych inicjowania CSQINP1 .
 - c. Dodaj definicję dla nowej klasy pamięci masowej do elementu zestawu danych inicjowania CSQ4INYP .

Szczegółowe informacje na temat komend DEFINE PSID i DEFINE STGCLASS można znaleźć w sekcji [DEFINE PSID i DEFINE STGCLASS](#).

Co zrobić, gdy jeden z zestawów stron stanie się pełny

Informacje na temat wykorzystania zestawów stron można znaleźć za pomocą komendy IBM MQ DISPLAY USAGE. Na przykład komenda:

```
DISPLAY USAGE PSID(03)
```

Wyświetla bieżący stan zestawu stron 03. To informuje o liczbie wolnych stron, jakie posiada ten zestaw stron.

Jeśli zdefiniowano dodatkowe przydziały dla zestawów stron, są one dynamicznie rozszerzane za każdym razem, gdy wypełniają je. W końcu używane są wszystkie dodatkowe obszary lub nie jest dostępna żadna dodatkowa przestrzeń dyskowa. Jeśli tak się stanie, aplikacja otrzymuje kod powrotu MQRC_STORAGE_MEDIUM_FULL.

Jeśli aplikacja odbierze kod powrotu MQRC_STORAGE_MEDIUM_FULL z wywołania MQI, jest to wyraźne wskazanie, że na zestawie stron nie ma wystarczającej ilości miejsca. Jeśli problem będzie się powtarzał lub prawdopodobnie będzie się powtarzał, należy zrobić coś, aby go rozwiązać.

Można podejść do tego problemu na wiele sposobów:

- Równoważenie obciążenia między zestawami stron przez przenoszenie kolejek z jednej strony ustawionej na inną.
- Rozwiń zestaw stron. Instrukcje na ten temat zawiera sekcja [“Jak zwiększyć rozmiar zestawu stron”](#) na stronie 396 .
- Zdefiniuj ponownie zestaw stron tak, aby mógł on rozwinąć się poza 4 GB do maksymalnej wielkości 64 GB. Więcej informacji na ten temat zawiera sekcja [Definiowanie zestawu stron do wielkości większej niż 4 GB](#) .

Jak równoważyć obciążenia na zestawach stron

Równoważenie obciążenia w zestawach stron oznacza przenoszenie komunikatów powiązanych z jedną lub większą liczbą kolejek z jednej strony ustawionej na inny, mniej używany, zestaw stron. Tej techniki należy użyć, jeśli nie jest praktyczny, aby rozwinąć zestaw stron.

Aby określić, które kolejki korzystają z zestawu stron, należy użyć odpowiednich komend produktu IBM MQ . Na przykład, aby dowiedzieć się, które kolejki są odwzorowane na zestaw stron 02, najpierw należy sprawdzić, które klasy pamięci masowej są odwzorowane na zestaw stron 02, za pomocą komendy:

```
DISPLAY STGCLASS(*) PSID(02)
```

Następnie użyj następującej komendy, aby dowiedzieć się, które kolejki używają tej klasy pamięci masowej:

```
DISPLAY QUEUE(*) TYPE(QLOCAL) STGCLASS
```

Przenoszenie kolejki niewspółużytkowanej

Aby przenieść kolejki i ich komunikaty z jednego zestawu stron do innego, należy użyć komendy MQSC MOVE QLOCAL (opisanej w sekcji [MOVE QLOCAL](#)). Po zidentyfikowaniu kolejki lub kolejek, które mają zostać przeniesione do nowego zestawu stron, należy wykonać następującą procedurę dla każdej z tych kolejek:

1. Upewnij się, że kolejka, która ma zostać przeniesiona, nie jest używana przez żadne aplikacje (czyli wartości IPPROCS i OPPROCS z komendy DISPLAY QSTATUS są równe zero) i że nie ma ona niezatwierdzonych komunikatów (wartość UNCOM komendy DISPLAY QSTATUS nie ma wartości NO).

Uwaga: Jedynym sposobem zapewnienia, że ten stan będzie kontynuowany, jest tymczasowe zmiana autoryzacji zabezpieczeń kolejki. Więcej informacji na ten temat zawiera sekcja [Profile dla zabezpieczeń kolejki](#) .

Jeśli nie jest to możliwe, późniejsze etapy tej procedury mogą się nie powieść, jeśli aplikacje zaczną korzystać z kolejki pomimo kroków zapobiegawczych, takich jak ustawienie PUT (DISABLED). Ta procedura nie może jednak nigdy utracić komunikatów.

2. Zapobiegaj przenoszeniu komunikatów do kolejki przez aplikacje, modyfikując definicję kolejki, aby wyłączyć MQPUT . Zmień definicję kolejki na PUT (DISABLED).
3. Zdefiniuj kolejkę tymczasową o takich samych atrybutach, jak kolejka, która jest przenoszona, za pomocą komendy:

```
DEFINE QL(TEMP_QUEUE) LIKE(Queue_to_move) PUT(ENABLED) GET(ENABLED)
```

Uwaga: Jeśli ta kolejka tymczasowa już istnieje z poprzedniego uruchomienia, usuń ją przed zdefiniowaniem.

4. Przenieś komunikaty do kolejki tymczasowej za pomocą następującej komendy:

```
MOVE QLOCAL(Queue_To_Move) TOQLOCAL(Temp_Queue)
```

5. Usuń kolejkę, która jest przenoszona, używając komendy:

```
DELETE QLOCAL(Queue_To_Move)
```

6. Zdefiniuj nową klasę pamięci masowej, która jest odwzorowywać na wymagany zestaw stron, na przykład:

```
DEFINE STGCLASS(NEW) PSID(nn)
```

Dodaj nową definicję klasy pamięci masowej do zestawów danych CSQINP2 , które są gotowe do następnego restartu menedżera kolejek.

7. Ponownie zdefiniuj przenoszony kolejkę, zmieniając atrybut klasy pamięci masowej:

```
DEFINE QL(Queue_To_Move) LIKE(Temp_Queue) STGCLASS(NEW)
```

Gdy kolejka jest ponownie zdefiniowana, jest ona oparta na kolejce tymczasowej utworzonej w kroku "3" na stronie 394.

8. Przenieś komunikaty z powrotem do nowej kolejki za pomocą komendy:

```
MOVE QLOCAL(Temp) TOQLOCAL(Queue_To_Move)
```

9. Kolejka utworzona w kroku "3" na stronie 394 nie jest już wymagana. Aby go usunąć, użyj następującej komendy:

```
DELETE QL(Temp_Queue)
```

10. Jeśli przenoszona kolejka została zdefiniowana w zestawach danych CSQINP2 , zmień atrybut STGCLASS odpowiedniej komendy DEFINE QLOCAL w zestawach danych CSQINP2 . Dodaj słowo kluczowe REPLACE w taki sposób, aby istniejąca definicja kolejki została zastąpiona.

Rysunek 46 na stronie 396 przedstawia wyodrębnienie z zadania równoważenia obciążenia.

```

//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=th1qua1.SCSQANLE,DISP=SHR
// DD DSN=th1qua1.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
ALTER QL(Queue_TO_MOVE) PUT(DISABLED)
DELETE QL(TEMP_QUEUE) PURGE
DEFINE QL(TEMP_QUEUE) LIKE(Queue_TO_MOVE) PUT(ENABLED) GET(ENABLED)
MOVE QLOCAL(Queue_TO_MOVE) TOQLOCAL(TEMP_QUEUE)
DELETE QL(Queue_TO_MOVE)
DEFINE STGCLASS(NEW) PSID(2)
DEFINE QL(Queue_TO_MOVE) LIKE(TEMP_QUEUE) STGCLASS(NEW)
MOVE QLOCAL(TEMP_QUEUE) TOQLOCAL(Queue_TO_MOVE)
DELETE QL(TEMP_QUEUE)
/*

```

Rysunek 46. Wyodrębnij z zadania równoważenia obciążenia dla zestawu stron

Jak zwiększyć rozmiar zestawu stron

Początkowo można przydzielić zestaw stron większy niż 4 GB, patrz sekcja [Definiowanie zestawu stron do wielkości większej niż 4 GB](#)

Zestaw stron można zdefiniować, aby był automatycznie rozwijany w miarę zapetnia się przez określenie opcji EXPAND (SYSTEM) lub EXPAND (USER). Jeśli zestaw stron został zdefiniowany z parametrem EXPAND (NONE), można go rozwinąć na jeden z dwóch sposobów:

- Zmień jego definicję, aby umożliwić automatyczną ekspansję. Więcej informacji na ten temat zawiera sekcja [Zmiana zestawu stron w celu umożliwienia automatycznego rozszerzania](#) .
- Utwórz nowy, większy zestaw stron i skopiuj komunikaty ze starej strony ustawionej na nową. Więcej informacji na ten temat zawiera sekcja [Przenoszenie komunikatów do nowego, większego zestawu stron](#) .

Definiowanie zestawu stron o wielkości większej niż 4 GB

Produkt IBM MQ może korzystać z zestawu stron o wielkości do 64 GB, pod warunkiem że zestaw danych jest zdefiniowany z rozszerzoną adresowalnością do VSAM. Rozszerzona adresowalność to atrybut, który jest nadawany przez klasę danych SMS. W przykładzie przedstawionym w poniższym przykładowym JCL klasa zarządzania 'EXTENDED' jest definiowana jako SMS z 'Rozszerzona adresowalność'. Jeśli istniejący zestaw stron nie jest obecnie zdefiniowany jako mający rozszerzoną adresowalność, należy użyć następującej metody, aby przeprowadzić migrację do rozszerzonego zestawu danych formatu adresowalności.

1. Zatrzymaj menedżer kolejek.
2. Użyj usług metod dostępu, aby zmienić nazwę istniejącego zestawu stron.
3. Zdefiniuj docelowy zestaw stron, taki sam rozmiar, jak istniejący zestaw stron, ale z DATACLAS (EXTENDED).

Uwaga: Zestawy danych w formacie rozszerzalnym muszą być zarządzane przez SMS. Są to mechanizmy żądania rozszerzonego formatu dla zestawów danych VSAM:

- Użycie klasy danych, która ma wartość DSNTYPE EXT, oraz podparametr R lub P, aby wskazać, że jest to wymagane lub preferowane.
- Kodowanie DSNTYPE=EXTREQ (format rozszerzony jest wymagane) lub DSNTYPE=EXTPREF (preferowany jest rozszerzony format) w instrukcji DD.
- Kodowanie parametru LIKE= w instrukcji DD w celu odwołania się do istniejącego zestawu danych w formacie rozszerzalnym.

Więcej informacji na ten temat zawiera sekcja [Ograniczenia w definiowaniu rozszerzonych zestawów danych](#).

4. Użyj funkcji COPYPAGE programu CSQUTIL, aby skopiować wszystkie komunikaty ze strony źródłowej ustawionej na docelowy zestaw stron. Więcej informacji na ten temat zawiera sekcja [Rozwijanie zestawu stron \(COPYPAGE\)](#).
5. Zrestartuj menedżer kolejek.
6. Zmień zestaw stron, aby użyć rozszerzenia systemu, aby zezwolić na kontynuację wzrostu poza jego bieżącą alokacją.

W poniższym języku JCL przedstawiono przykładowe komendy programu Access Method Services:

```
//S1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
ALTER 'VICY.CSQ1.PAGE01' -
NEWNAME('VICY.CSQ1.PAGE01.OLD')
ALTER 'VICY.CSQ1.PAGE01.DATA' -
NEWNAME('VICY.CSQ1.PAGE01.DATA.OLD')
DEFINE CLUSTER (NAME('VICY.CSQ1.PAGE01') -
MODEL('VICY.CSQ1.PAGE01.OLD') -
DATACLAS(EXTENDED))
/*
```

Zmiana zestawu stron w taki sposób, aby umożliwić automatyczne rozszerzanie

Użyj komendy ALTER PSID z opcjami EXPAND (USER) lub EXPAND (SYSTEM). Ogólne informacje na temat rozwijania zbiorów stron zawiera sekcja [ALTER PSID](#) i [Expanding a page set \(COPYPAGE\)](#).

Przenoszenie wiadomości do nowego, większego zestawu stron

Ta technika obejmuje zatrzymywanie i restartowanie menedżera kolejek. Spowoduje to usunięcie wszystkich nietrwałych komunikatów, które nie znajdują się w kolejkach współużytkowanych w czasie restartu. Jeśli istnieją komunikaty nietrwałe, które nie mają być usuwane, należy zamiast tego użyć równoważenia obciążenia. Szczegółowe informacje na ten temat zawiera sekcja [“Jak równoważyć obciążenia na zestawach stron”](#) na stronie 394. W tym opisie zestaw stron, który ma zostać rozwinięty, jest określany mianem zestawu stron *źródło*. Nowy, większy zestaw stron jest określany jako zestaw stron *miejsce docelowe*.

Wykonaj następujące kroki:

1. Zatrzymaj menedżer kolejek.
2. Zdefiniuj docelowy zestaw stron, upewniając się, że jest on większy niż źródłowy zestaw stron, z większą dodatkową wartością przydziału.
3. Aby sformatować docelowy zestaw stron, należy użyć funkcji FORMAT CSQUTIL. Więcej informacji na ten temat zawiera sekcja [Formatowanie zestawów stron \(FORMAT\)](#).
4. Użyj funkcji COPYPAGE programu CSQUTIL, aby skopiować wszystkie komunikaty ze strony źródłowej ustawionej na docelowy zestaw stron. Więcej informacji na ten temat zawiera sekcja [Rozwijanie zestawu stron \(COPYPAGE\)](#).
5. Zrestartuj menedżer kolejek przy użyciu docelowego zestawu stron, wykonując jedną z następujących czynności:
 - Zmień procedurę uruchomionego zadania menedżera kolejek w taki sposób, aby odwoła się do docelowego zestawu stron.
 - Użyj usług metod dostępu, aby usunąć zestaw stron źródłowych, a następnie zmień nazwę docelowego zestawu stron, nadając mu taką samą nazwę, jak nazwa źródłowego zestawu stron.

Uwaga:

Przed usunięciem zestawu stron produktu IBM MQ należy upewnić się, że wykonano wymagane kopie zapasowe.

Jak zmniejszyć zestaw stron

Przed użyciem menedżera kolejek należy uniemożliwić wszystkim użytkownikom, innym niż administrator produktu IBM MQ, korzystanie z menedżera kolejek. Na przykład: zmieniając ustawienia zabezpieczeń dostępu.

Jeśli istnieje duży zestaw stron, który jest w większości pusty (tak jak to pokazano w komendzie DISPLAY USAGE), można zmniejszyć jego wielkość. Procedura ta obejmuje korzystanie z funkcji COPY, FORMAT i LOAD programu CSQUTIL (patrz [Program narzędziowy IBM MQ](#)). Ta procedura nie działa dla zestawu stron zero (0), ponieważ nie jest on praktyczny w celu zmniejszenia wielkości tego zestawu stron. Jedynym sposobem na to jest ponowne zainicjowanie menedżera kolejek (patrz [“Reinicjowanie menedżera kolejek”](#) na stronie 423). Warunkiem wstępnym tej procedury jest wypróbowanie i usunięcie wszystkich użytkowników z systemu, tak aby wszystkie jednostki UOWs zostały zakończone, a zestawy stron były spójne.

1. Aby zatrzymać menedżer kolejek, należy użyć komendy STOP QMGR z atrybutem QUIESCE lub FORCE.
2. Uruchom funkcję SCOPY dla CSQUTIL z opcją PSID, aby skopiować wszystkie dane komunikatu z dużego zestawu stron i zapisać je w sekwencyjnym zestawie danych.
3. Zdefiniuj nowy, mniejszy zestaw danych zestawu stron, który zastąpi duży zestaw stron.
4. Uruchom funkcję FORMAT TYPE (NEW) w CSQUTIL dla zestawu stron utworzonego w kroku [“3”](#) na stronie 398.
5. Zrestartuj menedżer kolejek przy użyciu zestawu stron utworzonego w kroku [“3”](#) na stronie 398.
6. Uruchom funkcję LOAD CSQUTIL, aby ponownie załadować wszystkie komunikaty zapisane w kroku [“2”](#) na stronie 398.
7. Zezwól wszystkim użytkownikom na dostęp do menedżera kolejek.
8. Usuń stary duży zestaw stron.

Jak ponownie wprowadzić zestaw stron

W niektórych scenariuszach przydatna jest możliwość ponownego wprowadzenia starej strony do menedżera kolejek w trybie z połączeniem. Jeśli nie zostanie podjęte konkretne działanie, po wprowadzeniu starego zestawu stron menedżer kolejek będzie rozpoznawać, że RBA odtwarzania zestawu stron zapisanej w samym zestawie stron oraz w rekordach punktów kontrolnych jest stara i w związku z tym automatycznie rozpocznie odtwarzanie nośnika dla zestawu stron, aby było ono aktualne.

Takie odtwarzanie nośników może być wykonywane tylko przy restarcie menedżera kolejek i może zająć dużo czasu, zwłaszcza jeśli dzienniki archiwalne przechowywane na taśmie muszą być odczytane. Jednak zwykle w tym przypadku zestaw stron był w trybie bez połączenia dla tego okresu, a więc dziennik nie zawiera żadnych informacji istotnych dla odtwarzania zestawu stron.

Dostępne są następujące trzy opcje:

Umożliwia wykonanie pełnego odtwarzania nośnika.

1. Zatrzymaj menedżer kolejek.
2. Upewnij się, że definicje są dostępne dla strony ustawionej zarówno w uruchomionej procedurze zadania dla menedżera kolejek, jak i w zestawie danych inicjowania CSQINP1.
3. Zrestartuj menedżer kolejek.

Zezwól na zniszczenie wszystkich komunikatów znajdujących się na stronie.

Ten wybór jest przydatny w przypadku, gdy zestaw stron jest odłączony od dłuższego czasu (na przykład kilka miesięcy) i został on teraz postanowiony ponownie wykorzystać go w innym celu.

1. Formatuj zestaw stron przy użyciu funkcji FORMAT CSQUTIL z opcją TYPE (NEW).
2. Dodaj definicje dla strony ustawionej zarówno do procedury uruchomionego zadania dla menedżera kolejek, jak i do zestawu danych inicjowania CSQINP1.
3. Zrestartuj menedżer kolejek.

Użycie opcji TYPE (NEW) dla formatowania powoduje skasowanie bieżącej zawartości zestawu stron i nakazuje menedżerowi kolejek ignorowanie informacji historycznych w punkcie kontrolnym na temat zestawu stron.

Przeniesienie zestawu stron w tryb z połączeniem, unikając procesu odtwarzania nośnika.

Tej techniki należy używać tylko wtedy, gdy użytkownik ma pewność, że zestaw stron został odłączony od momentu czyszczenia menedżera kolejek. Wybór ten jest najbardziej odpowiedni w przypadku, gdy zestaw stron był w trybie bez połączenia przez krótki okres, zwykle z powodu problemów z działaniem, takich jak tworzenie kopii zapasowej podczas uruchamiania menedżera kolejek.

1. Sformatuj zestaw stron przy użyciu funkcji FORMAT CSQUTIL z opcją TYPE (REPLACE).
2. Dodaj ponownie zestaw stron z powrotem do menedżera kolejek za pomocą komendy DEFINE PSID z opcją DSN lub zezwoli na dodanie jej przy restarcie menedżera kolejek.

Użycie opcji TYPE (REPLACE) w celu formatowania sprawdzenia, czy zestaw stron został poprawnie zamknięty przez menedżer kolejek, i zaznacza go w taki sposób, aby odtwarzanie nośników nie było wykonywane. Żadne inne zmiany nie zostaną wprowadzone do zawartości zestawu stron.

Tworzenie kopii zapasowych i odtwarzanie zestawów stron

Dostępne są różne mechanizmy tworzenia kopii zapasowej i odtwarzania. Ten temat zawiera informacje na temat tych mechanizmów.

W tej sekcji opisano następujące tematy:

- [“Tworzenie punktu odtwarzania dla zasobów niewspółużytkowanych” na stronie 399](#)
- [“Tworzenie kopii zapasowych zestawów stron” na stronie 400](#)
- [“Odzyskiwanie zbiorów stron” na stronie 401](#)
- [Usuwanie zestawów stron](#)

Informacje na temat sposobu tworzenia punktu odtwarzania dla zasobów współużytkowanych zawiera sekcja [“Odtwarzanie współużytkowanych kolejek” na stronie 408](#).

Tworzenie punktu odtwarzania dla zasobów niewspółużytkowanych

Program IBM MQ może odtwarzać obiekty i nietrwałe komunikaty trwałe do ich bieżącego stanu, jeśli oba te komunikaty:

1. Istnieją kopie zestawów stron z wcześniejszego punktu.
2. Wszystkie dzienniki produktu IBM MQ są dostępne, aby można było wykonać odtwarzanie z tego punktu.

Reprezentują one punkt odtwarzania dla zasobów niewspółużytkowanych.

Zarówno obiekty, jak i komunikaty są przechowywane w zestawach stron. Na tym samym zestawie stron może istnieć wiele obiektów i komunikatów z różnych kolejek. Dla celów odtwarzania obiekty i komunikaty nie mogą być kopiowane w izolacji, dlatego należy utworzyć kopię zapasową zestawu stron, aby zapewnić poprawne odtworzenie danych.

Dziennik odtwarzania produktu IBM MQ zawiera zapis wszystkich trwałych komunikatów i zmian wprowadzonych w obiektach. Jeśli IBM MQ nie powiedzie się (na przykład z powodu błędu we/wy w zestawie stron), można odtworzyć zestaw stron, odtwarzając kopię zapasową i restartując menedżer kolejek. Program IBM MQ wprowadza zmiany w dzienniku do zestawu stron z punktu kopii zapasowej.

Istnieją dwa sposoby tworzenia punktu odtwarzania:

Pełna kopia zapasowa

Zatrzymaj menedżer kolejek, który wymusza wszystkie aktualizacje w zestawach stron.

Pozwala to na restartowanie z punktu odtwarzania, przy użyciu tylko zestawów danych, dla których utworzono kopie zapasowe, oraz z dzienników, które są od tego momentu.

Kopia zapasowa rozmytego

Należy wykonać *rozmyte* kopie zapasowe zestawów stron bez zatrzymywania menedżera kolejek.

Jeśli używana jest ta metoda, a powiązane z nimi dzienniki zostaną później uszkodzone lub utracone, nie będzie można użyć kopii zapasowej zestawu stron rozmytych do odtworzenia. Dzieje się tak dlatego, że kopie zapasowe zestawów stron rozmytych zawierają niespójny widok stanu menedżera kolejek i są zależne od dostępnych dzienników. Jeśli dzienniki nie są dostępne, należy powrócić do ostatniego zestawu kopii zapasowych zestawu stron, które zostały pobrane w czasie, gdy podsystem był nieaktywny (Metoda 1) i akceptować utratę danych z tego czasu.

Metoda 1: Pełna kopia zapasowa

Ta metoda polega na zamknięciu menedżera kolejek w dół. Wymusza to wszystkie aktualizacje zestawów stron tak, aby zestawy stron były w stanie spójnym.

1. Zatrzymaj wszystkie aplikacje produktu IBM MQ, które korzystają z menedżera kolejek (co pozwala im na wykonanie pierwszego). Można to zrobić poprzez zmianę zabezpieczeń dostępu lub ustawień kolejek, na przykład.
2. Po zakończeniu działania wszystkich działań wyświetl i rozwiąż wszelkie wątpliwe jednostki odzyskiwania. (Użyj komend DISPLAY CONN i RESOLVE INDOUBT, zgodnie z opisem w sekcji DISPLAY CONN i RESOLVE INDOUBT).

Powoduje to, że strona jest spójna. Jeśli tego nie zrobisz, zestawy stron mogą nie być spójne, a operacja tworzenia kopii zapasowej rozmyta jest efektywnie.

3. Wydadź komendę ARCHIVE LOG, aby upewnić się, że najnowsze dane dziennika są zapisywane w zestawach danych dziennika.
4. Wydadź komendę STOP QMGR MODE (QUIESCE). Zanotuj najniższą wartość RBA w komunikatach CSQI024I lub CSQI025I (więcej informacji na ten temat można znaleźć w sekcji CSQI024I i CSQI025I). Zestawy danych dziennika należy przechowywać począwszy od wartości wskazanej przez wartość RBA, aż do bieżącego zestawu danych dziennika.
5. Utwórz kopie zapasowe wszystkich zestawów stron menedżera kolejek (patrz “Tworzenie kopii zapasowych zestawów stron” na stronie 400).

Metoda 2: kopia zapasowa rozmytego

Ta metoda nie obejmuje wyłączania menedżera kolejek w dół. Z tego powodu aktualizacje mogą znajdować się w wirtualnych buforach pamięci masowej podczas procesu tworzenia kopii zapasowej. Oznacza to, że zestawy stron nie są w stanie spójnym i mogą być używane tylko do odtwarzania wraz z dziennikami.

1. Wprowadź komendę DISPLAY USAGE TYPE (ALL), a następnie zapisz wartość RBA w komunikatach CSQI024I lub CSQI025I (więcej informacji na ten temat zawiera sekcja CSQI024I i CSQI025I).
2. Utwórz kopie zapasowe zestawów stron (patrz “Tworzenie kopii zapasowych zestawów stron” na stronie 400).
3. Wydadź komendę ARCHIVE LOG, aby upewnić się, że najnowsze dane dziennika są zapisywane w zestawach danych dziennika. Aby restartować z punktu odtwarzania, należy zachować zestawy danych dziennika, rozpoczynając od zestawu danych dziennika wskazanego przez wartość RBA, aż do bieżącego zestawu danych dziennika.

Tworzenie kopii zapasowych zestawów stron

Aby odtworzyć zestaw stron, program IBM MQ musi wiedzieć, jak daleko w dzienniku należy przejść. Produkt IBM MQ przechowuje numer RBA dziennika w zerowej stronie każdego zestawu stron, nazywał się *numerem kolejnym dziennika odtwarzania* (LSN). Jest to początkowy adres RBA w dzienniku, z którego program IBM MQ może odtworzyć zestaw stron. Podczas tworzenia kopii zapasowej zestawu stron ten numer jest również kopiowany.

Jeśli kopia jest później używana do odtwarzania zestawu stron, program IBM MQ musi mieć dostęp do wszystkich rekordów dziennika z tej wartości RBA do bieżącego RBA. Oznacza to, że należy zachować wystarczającą ilość rekordów dziennika, aby umożliwić IBM MQ odtwarzanie z najstarszej kopii zapasowej zestawu stron, który ma być zachowany.

Użyj funkcji ADRDSSU COPY, aby skopiować zestawy stron.

Więcej informacji na ten temat zawiera dokumentacja [COPY DATASET Command Syntax for Logical Data Set](#).

Na przykład:

```
//STEP2 EXEC PGM=ADRDSSU,REGION=6M
//SYSPRINT DD SYSOUT=H
//SYSIN DD *
COPY -
DATASET(INCLUDE(SCENDATA.MQPA.PAGESET.*)) -
RENAMEU(SCENDATA.MQPA.PAGESET.** ,SCENDATA.MQPA.BACKUP1.** ) -
SPHERE -
REPUNC -
FASTREPLICATION(PREF ) -
CANCELERROR -
TOL(ENQF)
/*
//
```

Jeśli zestaw stron jest kopiowany podczas działania menedżera kolejek, należy użyć programu narzędziowego do kopiowania, który najpierw kopiuje stronę zerową zestawu stron. Jeśli tego nie zrobisz, możesz uszkodzić dane w zestawie stron.

Jeśli proces dynamicznego rozwijania zestawu stron zostanie przerwany, na przykład przez zasilanie utraconego systemu, nadal można użyć komendy ADRDSSU, aby utworzyć kopię zapasową zestawu stron.

Jeśli zostanie wykonana usługa Access Method Services IDCAMS LISTCAT ENT('page set data set name') ALLOC, użytkownik zobaczy, że wartość HI-ALLOC-RBA jest większa niż HI-USED-RBA.

Następnym razem, gdy ten zestaw stron zostanie zapełniony, zostanie on ponownie rozszerzony, o ile to możliwe, a strony między wysokim zużytym RBA i najwyższym przydzielonym RBA są używane, wraz z innym nowym zakresem.

Tworzenie kopii zapasowej definicji obiektów

Należy również utworzyć kopię zapasową kopii definicji obiektów. W tym celu należy użyć funkcji MAKEDEF funkcji CSQUTIL COMMAND (opisanej w sekcji [Wydawanie komend do IBM MQ \(COMMAND\)](#)).

Utwórz kopię zapasową definicji obiektów za każdym razem, gdy zostanie utworzona kopia zapasowa menedżera kolejek, i zachowaj najnowszą wersję.

Odzyskiwanie zbiorów stron

Jeśli menedżer kolejek zakończył działanie z powodu niepowodzenia, menedżer kolejek może zostać normalnie zrestartowany przy użyciu wszystkich operacji odtwarzania wykonywanych podczas restartu. Takie odtwarzanie nie jest jednak możliwe, jeśli żaden z zestawów stron lub zestawów danych dziennika nie jest dostępny. Zakres, w jakim można teraz odzyskać, zależy od dostępności kopii zapasowych zestawów stron i zestawów danych dziennika.

Aby zrestartować system z punktu odtwarzania, należy:

- Kopia zapasowa zestawu stron, który ma zostać odzyskany.
- Jeśli został użyty proces tworzenia kopii zapasowej "fuzzy" opisany w sekcji "Metoda 2: kopia zapasowa rozmytego" na stronie 400, zestaw danych dziennika zawierający zarejestrowaną wartość RBA, zestaw danych dziennika, który został wykonany przez komendę ARCHIVE LOG, oraz wszystkie zestawy danych dziennika między tymi komendami.

- Jeśli użyto pełnej kopii zapasowej, ale nie ma zestawów danych dziennika następujących po wykonaniu komendy ARCHIVE LOG, **nie** należy uruchamiać funkcji FORMAT TYPE (REPLACE) programu narzędziowego CSQUTIL dla wszystkich zestawów stron.

Aby odtworzyć zestaw stron do bieżącego stanu, należy mieć także wszystkie zestawy danych dziennika i rekordy od czasu wykonania komendy ARCHIVE LOG.

Istnieją dwie metody odtwarzania zestawu stron. Aby można było użyć jednej z metod, menedżer kolejek musi zostać zatrzymany.

Odtwarzanie proste

Jest to prostsza metoda, która jest odpowiednia dla większości sytuacji odzyskiwania.

1. Usuń zestaw stron, który ma zostać odtworzony z kopii zapasowej.
2. Użyj funkcji ADRDSSU COPY, aby odtworzyć zestaw stron z kopii zapasowej.

Alternatywnie można zmienić nazwę kopii zapasowej na nazwę oryginalną lub zmienić instrukcję DD CSQP00xx w procedurze menedżera kolejek w taki sposób, aby wskazywała na zestaw stron kopii zapasowych. Jeśli jednak zostanie utracony lub uszkodzony zestaw stron, kopia zapasowa nie będzie już miała miejsca do odtworzenia.

3. Zrestartuj menedżer kolejek.
4. Po pomyślnym zrestartowaniu menedżera kolejek można zrestartować aplikacje.
5. Przywróć normalne procedury tworzenia kopii zapasowej dla odtwarzanej strony.

Odtwarzanie zaawansowane

Ta metoda zapewnia korzyści z wydajności, jeśli istnieje duży zestaw stron do odtworzenia, lub jeśli od czasu wykonania ostatniej kopii zapasowej jest dużo aktywności na stronie. Wymaga to jednak bardziej ręcznej interwencji niż metoda prosta, co może zwiększyć ryzyko wystąpienia błędu oraz czas na wykonanie odzyskiwania.

1. Usuń i ponownie zdefiniuj zestaw stron, który ma zostać odtworzony z kopii zapasowej.
2. Użyj komendy ADRDSSU, aby skopiować kopię zapasową zestawu stron do nowego zestawu stron. Zdefiniuj nowy zestaw stron o drugorzędnej wartości przydziału, aby można go było dynamicznie rozwijać.

Alternatywnie można zmienić nazwę kopii zapasowej na nazwę oryginalną lub zmienić instrukcję DD CSQP00xx w procedurze menedżera kolejek w taki sposób, aby wskazywała na zestaw stron kopii zapasowych. Jeśli jednak zostanie utracony lub uszkodzony zestaw stron, kopia zapasowa nie będzie już miała miejsca do odtworzenia.

3. Zmień definicje CSQINP1 dla menedżera kolejek, aby pula buforów powiązana z zestawem stron była odtwarzana tak, jak to możliwe. Dzięki czemu pula buforów jest duża, można zachować wszystkie zmienione strony rezydentne w puli buforów i zmniejszyć ilość operacji we/wy do zestawu stron.
4. Zrestartuj menedżer kolejek.
5. Po pomyślnym zrestartowaniu menedżera kolejek zatrzymaj go (przy użyciu wyciszenia), a następnie zrestartuj go, używając normalnej definicji puli buforów dla tego zestawu stron. Po pomyślnym zakończeniu drugiego restartu można zrestartować aplikacje.
6. Przywróć normalne procedury tworzenia kopii zapasowej dla odtwarzanej strony.

Co się dzieje, gdy menedżer kolejek jest restartowany

Po zrestartowaniu menedżera kolejek wszystkie zmiany wprowadzone w zestawie stron, które są zarejestrowane w dzienniku, zaczynają się od punktu restartu dla zestawu stron. Produkt IBM MQ może odtworzyć wiele zestawów stron w ten sposób. Zestaw stron jest dynamicznie rozwijany, jeśli jest wymagany, podczas odtwarzania nośnika.

Podczas restartu program IBM MQ określa dziennik RBA, od którego ma zostać rozpoczęty, od najniższej wartości z następujących:

- Numer LSN odtwarzania z rekordu dziennika punktu kontrolnego dla każdego zestawu stron.

- Odzyskaj numer LSN od strony zero w każdym zestawie stron.
- RBA najstarszej niekompletnej jednostki odzyskiwania w systemie w momencie tworzenia kopii zapasowej.

Wszystkie definicje obiektów są przechowywane na stronie zerowej. Komunikaty mogą być zapisywane na dowolnym dostępnym zestawie stron.

Uwaga: Nie można zrestartować menedżera kolejek, jeśli zerowy zestaw stron nie jest dostępny.

Usuwanie zestawów stron

Zestaw stron można usunąć za pomocą komendy DELETE PSID. Szczegółowe informacje na temat tej komendy zawiera sekcja [DELETE PSID](#).

Nie można usunąć zestawu stron, do którego nadal odwołuje się dowolna klasa pamięci masowej. Użyj komendy DISPLAY STGCLASS, aby dowiedzieć się, które klasy pamięci masowej odwołują się do zestawu stron.

Zestaw danych jest deprzydzielony z programu IBM MQ, ale nie został usunięty. Pozostaje on dostępny do wykorzystania w przyszłości lub można go usunąć za pomocą narzędzi z/OS.

Usuń zestaw stron z uruchomionej procedury zadania dla menedżera kolejek.

Usuń definicję zestawu stron z zestawu danych inicjowania CSQINP1.

Tworzenie kopii zapasowych i odtwarzanie kolejek za pomocą komendy CSQUTIL

W tym temacie opisano dodatkowe informacje na temat tworzenia kopii zapasowej i odtwarzania za pomocą komendy CSQUTIL.

Można użyć funkcji programu narzędziowego CSQUTIL do tworzenia kopii zapasowych i odtwarzania kolejek. Aby utworzyć kopię zapasową kolejki, należy użyć funkcji COPY lub SCOPY, aby skopiować komunikaty z kolejki do zestawu danych. Aby odtworzyć kolejkę, należy użyć funkcji uzupełniających LOAD lub SLOAD. Więcej informacji na ten temat zawiera sekcja [Program narzędziowy IBM MQ](#).

Zarządzanie pulami buforów

W tym temacie opisano, czy pule buforów mają zostać zmienione, czy też mają zostać usunięte.

W tej sekcji opisano sposób modyfikowania i usuwania pul buforów. Zawiera on następujące sekcje:

- [“Jak zmienić liczbę buforów w puli buforów”](#) na stronie 404
- [“Jak usunąć pulę buforów”](#) na stronie 404

Pule buforów są definiowane podczas inicjowania menedżera kolejek przy użyciu komend DEFINE BUFFPOOL wydanych na podstawie wejściowego zestawu danych inicjowania CSQINP1. Ich atrybuty można zmieniać w odpowiedzi na wymagania biznesowe, gdy menedżer kolejek jest uruchomiony, korzystając z procesów szczegółowo określonych w tym temacie. Menedżer kolejek rejestruje bieżące atrybuty puli buforów w rekordach dziennika punktu kontrolnego. Są one automatycznie odtwarzane przy kolejnym restarcie menedżera kolejek, chyba że definicja puli buforów w CSQINP1 zawiera atrybut REPLACE.

Użyj komendy [DISPLAY USAGE](#), aby wyświetlić bieżące atrybuty buforu.

Pule buforów można również definiować dynamicznie za pomocą komendy [DEFINE PSID](#) z opcją DSN.

W przypadku dynamicznej zmiany pul buforów należy również zaktualizować ich definicje w zestawie danych inicjowania CSQINP1.

Sekcja Planowanie w systemie z/OS zawiera opis zestawów stron, klas pamięci masowej, buforów i pul buforów oraz niektórych uwag dotyczących wydajności, które mają zastosowanie.

Uwaga: Pule buforów wykorzystują znaczącą pamięć masową. Po zwiększeniu wielkości puli buforów lub zdefiniowaniu nowej puli buforów upewnij się, że dostępna jest wystarczająca ilość pamięci masowej. Więcej informacji na ten temat zawiera sekcja [Przestrzeń pamięci dla przestrzeni adresowej](#).

Jak zmienić liczbę buforów w puli buforów

Jeśli pula buforów jest za mała, może to spowodować, że w konsoli zostanie wyświetlony komunikat `CSQP020E`, można przydzielić do niego więcej buforów przy użyciu komendy `ALTER BUFFPOOL` w następujący sposób:

1. Określ, ile miejsca jest dostępne dla nowych buforów, przeglądając komunikaty `CSQY220I` w dzienniku. Dostępne miejsce jest raportowane w MB. Ponieważ bufor ma wielkość 4 kB, każdy MB wolnego miejsca pozwala na przydzielenie 256 buforów. Nie należy przydzielać wszystkich wolnych przestrzeni do buforów, ponieważ niektóre z nich są wymagane do wykonywania innych zadań.

Jeśli pula buforów korzysta ze stałych stron o wielkości 4 kB, to znaczy, że jej atrybut `PAGECLAS` ma wartość `FIXED4KB`, należy upewnić się, że na partycji `LPAR` dostępna jest wystarczająca ilość pamięci rzeczywistej.

2. Jeśli zgłoszona ilość wolnego miejsca jest niewystarczająca, zwolnij niektóre bufory z innej puli buforów przy użyciu komendy

```
ALTER BUFFPOOL(buf-pool-id) BUFFERS(integer)
```

gdzie *id-puli-buf* to pula buforów, z której ma zostać odzyskane miejsce, a *liczba całkowita* to nowa liczba buforów, które mają zostać przydzielone do tej puli buforów, która musi być mniejsza niż pierwotna liczba przydzielonych do niej buforów.

3. Dodaj bufory do puli buforów, która ma zostać rozwinięta przy użyciu komendy

```
ALTER BUFFPOOL(buf-pool-id) BUFFERS(integer)
```

gdzie *id-puli-buf* to pula buforów, która ma zostać rozwinięta, a *liczba całkowita* to nowa liczba buforów, które mają zostać przydzielone do tej puli buforów, która musi być większa niż pierwotna liczba przydzielonych do niej buforów.

Jak usunąć pulę buforów

Jeśli pula buforów nie jest już używana przez żadne zestawy stron, należy ją usunąć, aby zwolnić pamięć wirtualną przydzieloną do tej puli pamięci masowej.

Pulę buforów należy usunąć za pomocą komendy `DELETE BUFFPOOL`. Wykonanie komendy nie powiedzie się, jeśli wszystkie zestawy stron korzystają z tej puli buforów.

Informacje na temat usuwania zestawów stron zawiera sekcja [“Usuwanie zestawów stron”](#) na stronie 403.

Zarządzanie grupami współużytkowania kolejek i współużytkowanymi kolejkami w systemie z/OS

Produkt IBM MQ może korzystać z różnych typów zasobów współużytkowanych, takich jak grupy współużytkowania kolejek, kolejki współużytkowane i narzędzie CF. W tym temacie opisano procedury niezbędne do zarządzania tymi zasobami współużytkowanymi.

Ta sekcja zawiera informacje na temat następujących tematów:

- [“Zarządzanie grupami współużytkowania kolejek”](#) na stronie 405
- [“Zarządzanie współużytkowanymi kolejkami”](#) na stronie 408
- [“Zarządzanie obiektami grupy”](#) na stronie 413

- [“Zarządzanie narzędziem CF” na stronie 413](#)

Zarządzanie grupami współużytkowania kolejek

Menedżer kolejek można dodawać do grupy współużytkowania kolejek (QSG) lub usuwać je, a także zarządzać powiązаныmi tabelami Db2 .

W tym temacie przedstawiono sekcje dotyczące następujących zadań:

- [“Konfigurowanie grupy współużytkowania kolejek” na stronie 405](#)
- [“Dodawanie menedżera kolejek do grupy współużytkowania kolejek” na stronie 406](#)
- [“Usuwanie menedżera kolejek z grupy współużytkowania kolejek” na stronie 407](#)
- [“Usuwanie grupy współużytkowania kolejek z tabel Db2” na stronie 407](#)
- [“Sprawdzanie poprawności spójności definicji Db2” na stronie 408](#)

Konfigurowanie grupy współużytkowania kolejek

Każda grupa współużytkowania kolejek ma nazwę maksymalnie cztery znaki. Nazwa taka musi być unikalna w danej sieci i nie może być identyczna z nazwą menedżera kolejek.

Aby skonfigurować grupę współużytkowania kolejek, wykonaj następujące kroki:

1. Jeśli jest to pierwsza grupa współużytkowania kolejek, w której ma być używana grupa współużytkowania danych produktu Db2 , [skonfiguruj środowisko produktu Db2](#).
2. [Skonfiguruj narzędzie CF](#).
3. Dodaj grupę współużytkowania kolejek do tabel produktu Db2 . Użyj funkcji ADD QSG programu narzędziowego grupy współużytkowania kolejki (CSQ5PQSG). Ten program jest opisany w sekcji [Program narzędziowy grupy współużytkowania kolejek](#). Przykład można znaleźć w pliku `thlqual.SCSQPROC(CSQ45AQS)`.
4. Dodaj menedżera kolejek do grupy współużytkowania kolejek, wykonując kroki opisane w sekcji [“Dodawanie menedżera kolejek do grupy współużytkowania kolejek” na stronie 406](#) .
5. Zdefiniuj struktury aplikacji na serwerze IBM MQ , wykonując kroki opisane w sekcji [“Dodawanie struktury narzędzia CF” na stronie 413](#).
6. Jeśli jest to wymagane, [przeprowadź migrację kolejek niewspółużytkowanych do współużytkowanych kolejek](#).
7. Aby uzyskać dostępność, należy utworzyć kanały współużytkowane do i z grupy współużytkowania kolejek.
 - W przypadku połączeń z grupą współużytkowania kolejek:
 - Skonfiguruj gniazdo VIPA lub router sprzętu w celu rozłożenia obciążenia między dostępnymi menedżerami kolejek w QSG.
 - Zdefiniuj kanał odbiorczy z QSGDISP (GROUP), aby upewnić się, że definicja kanału jest dostępna we wszystkich menedżerach kolejek w QSG.
 - Uruchom program nasłuchujący z INDISP (GROUP), dla każdego menedżera kolejek, dla połączeń kanału MCA w QSG. Połączenia klienckie z QSG powinny nadal łączyć się z programem nasłuchującym uruchomionym z programem INDISP (QMGR).
 - Zmień aplikacje tak, aby nawiązały połączenie przy użyciu nazwy QSG, a nie określonej nazwy menedżera kolejek.
 - Upewnij się, że reguły uwierzytelniania kanału we wszystkich menedżerach kolejek w QSG są takie same, aby umożliwić aplikacjom łączenie się z dowolnym menedżerem kolejek w QSG.
 - Dla połączeń wychodzących z grupy współużytkowania kolejek:
 - Zdefiniuj współużytkowaną kolejkę transmisji.
 - Zdefiniuj kanał wychodzący z QSGDISP (GROUP) i DEFCDISP (SHARED).

Jeśli istniejący kanał jest przekształcany w kanał współużytkowany, może być konieczne wydanie komendy `RESET CHANNEL` przed uruchomieniem kanału, ponieważ kolejka synchronizacji używana przez kanał zostanie zmieniona.

Dodawanie menedżera kolejek do grupy współużytkowania kolejek

Menedżer kolejek może zostać dodany do istniejącej grupy współużytkowania kolejek.

Należy pamiętać, że:

- Aby można było dodać do niej menedżery kolejek, musi istnieć grupa współużytkowania kolejek.
- Menedżer kolejek może być elementem tylko jednej grupy współużytkowania kolejek.

Aby dodać menedżera kolejek do grupy współużytkowania kolejek, wykonaj następujące kroki:

1. Wykonaj zadania w sekcji zaimplementuj elementy sterujące zabezpieczeń ESM dla grupy współużytkowania kolejek, aby nadać odpowiednie uprawnienia dostępu do identyfikatorów użytkowników menedżera kolejek i inicjatora kanału.
2. Jeśli grupa współużytkowania kolejki ma struktury CF skonfigurowane na potrzeby przenoszenia danych do SMDS, wykonaj zadania w sekcji konfigurowanie środowiska SMDS.
3. Zatrzymaj menedżer kolejek.
4. Należy użyć funkcji `ADD QMGR` programu narzędziowego grupy współużytkowania kolejki (`CSQ5PQSG`). Ten program jest opisany w sekcji Program narzędziowy grupy współużytkowania kolejki. Przykład można znaleźć w pliku `thlqual.SCSQPROC(CSQ45AQM)`.
5. Zmień moduł parametru systemowego, aby dodać dane grupy współużytkowania kolejek:
 - a. Zmodyfikuj `CSQ6SYSP`, aby określić parametr `QSGDATA`. Więcej informacji na ten temat zawiera sekcja za pomocą komendy CSQ6SYSP.
 - b. Zmontuj i dociążaj moduł parametrów systemowych. Dla modułu ładowania można użyć innej nazwy.
 - c. Zmień proces uruchamiania w taki sposób, aby był używany nowy moduł.
6. Skopiuj i dopasuj przykładowy element `thlqual.SCSQPROC(CSQ4INSS)`, który definiuje wymagane struktury systemu CF i kolejki `SYSTEM`. Dodaj dostosowany element do definicji danych `CSQINP2` w kodzie `JCL` uruchamiania menedżera kolejek.
7. Zrestartuj menedżer kolejek przy użyciu modułu parametru systemowego grupy współużytkowania kolejki.
8. Opcjonalnie można przeprowadzić migrację do profilu zabezpieczeń z przedrostkiem nazwy grupy współużytkowania kolejek, a nie z nazwy menedżera kolejek.
9. Jeśli do połączeń z QSG używane są kanały współużytkowane, należy utworzyć reguły uwierzytelniania kanału, które są kopią lustrzaną tych z innych menedżerów kolejek w QSG, aby umożliwić aplikacjom nawiązać połączenie z dowolnym menedżerem kolejek w QSG.
10. Opcjonalnie wykonaj jedną z następujących czynności, aby zezwolić aplikacjom połączonym z menedżerem kolejek w QSG na umieszczanie komunikatów w kolejkach udostępnianych przez inne menedżery kolejek w QSG:
 - Włącz kolejkowanie wewnątrz grupy, wydając komendę `ALTER QMGR IGQ (ENABLED)`.
 - Zdefiniuj kolejki transmisji i kanały do innych menedżerów kolejek w QSG. Definiowanie kolejek transmisji o takiej samej nazwie, jak docelowe menedżery kolejek, pozwala uniknąć konieczności definiowania kolejek zdalnych i aliasów menedżera kolejek.

Uwaga: Aby dodać menedżera kolejek do istniejącej grupy współużytkowania kolejek, w której znajdują się menedżery kolejek działające we wcześniejszych wersjach produktu IBM MQ, należy najpierw zastosować poprawkę PTF dla najwyższej wersji produktu IBM MQ w grupie do każdego menedżera kolejek wcześniejszych wersji w grupie.

Usuwanie menedżera kolejek z grupy współużytkowania kolejek

Menedżer kolejek można usunąć z grupy współużytkowania kolejek tylko wtedy, gdy protokoły menedżera kolejek nie są wymagane przez inny proces, a wszystkie SMDS należące do tego menedżera kolejek są puste.

Dzienniki są potrzebne, jeśli zawierają:

- Najnowsza kopia zapasowa jednej z struktur aplikacji CF (Coupling Facility) używanych przez grupę współużytkowania kolejek
- Dane wymagane przez proces odtwarzania w przyszłości, tj. menedżer kolejek, użył odtwarzalnej struktury od czasu opisanego przez wartość ostatniego okresu wykluczenia z kopii zapasowej.

Jeśli albo oba te punkty mają zastosowanie, albo SMDS, którego właścicielem jest menedżer kolejek, zawiera komunikaty, menedżer kolejek nie może zostać usunięty. Aby określić, które dzienniki menedżerów kolejek są potrzebne w przypadku przyszłego procesu odtwarzania, należy użyć komendy MQSC DISPLAY CFSTATUS z opcją TYPE (BACKUP) (szczegółowe informacje na temat tej komendy zawiera sekcja [DISPLAY CFSTATUS](#)).

Aby usunąć menedżera kolejek z grupy współużytkowania kolejek, wykonaj następujące kroki:

1. Zatrzymaj wszystkie aplikacje połączone z menedżerem kolejek, które umieszczają komunikaty w kolejkach współużytkowanych.
2. Rozstrzygnij wszystkie niepewne jednostki pracy związane z tym menedżerem kolejek.
3. Określ, czy we wszystkich SMDS należących do menedżera kolejek znajdują się komunikaty, wydając komendę DISPLAY USAGE TYPE (SMDS).
4. Jeśli istnieją odciążone komunikaty dla dowolnej struktury aplikacji, poczekaj, aż te komunikaty zostaną pobrane z kolejki. Liczba przetadowanych komunikatów zgłaszanych przez komendę DISPLAY USAGE TYPE (SMDS) powinna wynosić zero przed kontynuowaniem.
5. Zamknij menedżer kolejek w trybie czyszczonym, używając trybu STOP QMGR MODE (QUIESCE).
6. Poczekaj, aż odstęp czasu będzie równy co najmniej wartości parametru EXCLINT, który zostanie podany w komendzie BACKUP CFSTRUCT w następnym kroku.
7. W innym menedżerze kolejek należy uruchomić kopię zapasową struktury CF dla każdej odtwarzalnej struktury CF, używając komendy MQSC BACKUP CFSTRUCT i określając wartość EXCLINT zgodnie z wymaganiami w poprzednim kroku.
8. Upewnij się, że dzienniki menedżera kolejek nie są potrzebne do odtworzenia struktur systemu CF, poprzez sprawdzenie danych wyjściowych komendy DISPLAY CFSTATUS (*) TYPE (BACKUP).
9. Użyj funkcji REMOVE QMGR w programie narzędziowym CSQ5PQSG, aby usunąć menedżera kolejek z grupy współużytkowania kolejek. Ten program jest opisany w sekcji [Program narzędziowy grupy współużytkowania kolejek](#). Przykład można znaleźć w pliku thlqual.SCSQPROC(CSQ45RQM).
10. Przed zrestartowaniem menedżera kolejek należy zresetować parametr systemowy QSGDATA na wartość domyślną, a następnie ponownie utworzyć moduł parametrów systemowych. Informacje na temat sposobu dostosowania parametrów systemowych zawiera sekcja [Korzystanie z komendy CSQ6SYSP](#).

Należy zwrócić uwagę, że podczas usuwania ostatniego menedżera kolejek w grupie współużytkowania kolejek należy użyć opcji FORCE, a nie REMOVE. Spowoduje to usunięcie menedżera kolejek z grupy współużytkowania kolejek, podczas gdy nie jest wykonywane sprawdzanie spójności dzienników menedżera kolejek, które są wymagane do odtwarzania. Tę operację należy wykonać tylko wtedy, gdy usuwana jest grupa współużytkowania kolejek.

Usuwanie grupy współużytkowania kolejek z tabel Db2

Aby usunąć grupę współużytkowania kolejek z tabel produktu Db2, należy użyć funkcji REMOVE QSG programu narzędziowego grupy współużytkowania kolejki (CSQ5PQSG). Ten program jest opisany

w sekcji [Program narzędziowy grupy współużytkowania kolejki](#). Przykład można znaleźć w pliku `thlqual.SCSQPROC(CSQ45RQS)`.

Grupę współużytkowania kolejek można usunąć tylko ze wspólnych tabel grup współużytkowania danych produktu Db2 po usunięciu wszystkich menedżerów kolejek z grupy współużytkowania kolejek (zgodnie z opisem w sekcji [“Usuwanie menedżera kolejek z grupy współużytkowania kolejek”](#) na stronie 407).

Po usunięciu rekordu grupy współużytkowania kolejki z tabeli administracyjnej grupy współużytkowania kolejki wszystkie obiekty i informacje administracyjne odnoszące się do tej grupy współużytkowania kolejki są usuwane z innych tabel programu IBM MQ Db2. Obejmuje to informacje o kolejce współużytkowanej i obiekcie grupy.

Sprawdzanie poprawności spójności definicji Db2

Problemy z kolejkami współużytkowanymi w grupie współużytkowania kolejek mogą wystąpić, jeśli definicje obiektów produktu Db2 mają, z dowolnej przyczyny, niespójne.

Aby sprawdzić spójność definicji obiektów Db2 dla menedżerów kolejek, struktur CF i kolejek współużytkowanych, należy użyć funkcji VERIFY QSG programu narzędziowego grupy współużytkowania kolejki (CSQ5PQSG). Ten program jest opisany w sekcji [Program narzędziowy grupy współużytkowania kolejek](#).

Zarządzanie współużytkowanymi kolejkami

Ten temat zawiera informacje na temat odtwarzania, przenoszenia i migrowania współużytkowanych kolejek.

W tej sekcji opisano następujące zadania:

- [“Odtwarzanie współużytkowanych kolejek”](#) na stronie 408
- [“Przenoszenie kolejek współużytkowanych”](#) na stronie 409
- [“Migrowanie kolejek niewspółużytkowanych do współużytkowanych kolejek”](#) na stronie 411
- [Zawieszanie połączenia Db2](#)

Odtwarzanie współużytkowanych kolejek

Produkt IBM MQ może odzyskać trwałe komunikaty w kolejkach współużytkowanych, jeśli wszystkie:

- Wykonano kopie zapasowe struktur CF zawierających komunikaty.
- Wszystkie dzienniki dla wszystkich menedżerów kolejek w grupie współużytkowania kolejek są dostępne, aby można było wykonać odtwarzanie z punktu, w którym są wykonywane kopie zapasowe.
- Produkt Db2 jest dostępny, a tabela kopii zapasowej struktury jest nowsza niż najnowsza kopia zapasowa struktury CF.

Komunikaty w kolejce współużytkowanej są przechowywane w strukturze CF (Coupling Facility). Komunikaty trwałe mogą być umieszczane w kolejkach współużytkowanych i jak komunikaty trwałe w niewspółużytkowanych kolejkach, są one kopiowane do dziennika menedżera kolejek. Udostępniono komendy MQSC [BACKUP CFSTRUCT](#) i [RECOVER CFSTRUCT](#), aby umożliwić odtwarzanie struktury systemu CF w mało prawdopodobnym przypadku awarii narzędzia CF. W takich okolicznościach wszystkie nietrwałe komunikaty zapisane w strukturze, której dotyczy problem, są tracone, ale trwałe komunikaty mogą być odtwarzane. Wszelkie dalsze działania aplikacji korzystające ze struktury są zapobiegane do czasu odtworzenia struktury.

Aby włączyć odtwarzanie, należy często utworzyć kopię zapasową struktur listy narzędzia CF przy użyciu komendy MQSC [BACKUP CFSTRUCT](#). Komunikaty w strukturze CF są zapisywane w aktywnym zestawie danych dziennika menedżera kolejek, w którym ma zostać utworzona kopia zapasowa. Zapisuje ona rekord kopii zapasowej do programu Db2: nazwa kopii zapasowej struktury CF, nazwa menedżera kolejek wykonanego dla kopii zapasowej, zakres RBA dla tej kopii zapasowej w dzienniku menedżera kolejek oraz czas utworzenia kopii zapasowej. Należy utworzyć kopię zapasową struktur listy CF, nawet jeśli

nie są aktywnie używane kolejki współużytkowane, na przykład w przypadku skonfigurowania grupy współużytkowania kolejki, która ma zamiar używać go w przyszłości.

Strukturę systemu CF można odzyskać, wydając komendę MQSC RECOVER CFSTRUCT do menedżera kolejek, który może wykonać odtwarzanie. W grupie współużytkowania kolejek można użyć dowolnego menedżera kolejek. Można określić pojedynczą strukturę CF, która ma być odzyskana, lub można odtworzyć kilka struktur systemu CF równocześnie.

Jak zauważono wcześniej, ważne jest częste tworzenie kopii zapasowych struktur listy CF, w przeciwnym razie odzyskanie struktury CF może zająć dużo czasu. Co więcej, proces odtwarzania nie może zostać anulowany.

Definicja kolejki współużytkowanej jest przechowywana w bazie danych Db2 i dlatego może być odzyskana w razie potrzeby przy użyciu standardowych procedur bazy danych Db2 . Więcej informacji na ten temat zawiera sekcja [Kolejki współużytkowane i grupy współużytkowania kolejek](#) .

Przenoszenie kolejek współużytkowanych

W tej sekcji opisano sposób równoważenia obciążenia poprzez przeniesienie współużytkowanej kolejki z jednej struktury narzędzia CF do innej. Opisano także sposób przenoszenia niewspółużytkowanej kolejki do kolejki współużytkowanej oraz przenoszenia kolejki współużytkowanej do kolejki niewspółużytkowanej.

W przypadku przenoszenia kolejki należy zdefiniować kolejkę tymczasową jako część procedury. Jest to spowodowane tym, że każda kolejka musi mieć unikalną nazwę, więc nie można mieć dwóch kolejek o tej samej nazwie, nawet jeśli kolejki mają różne dyspozycje kolejki. Produkt IBM MQ toleruje dwie kolejki o tej samej nazwie (jak w kroku [“2”](#) na stronie [409](#)), ale nie można używać kolejek.

- Przenoszenie kolejki z jednej struktury narzędzia CF do innej
- Przenoszenie kolejki niewspółużytkowanej do kolejki współużytkowanej
- Przenoszenie kolejki współużytkowanej do kolejki niewspółużytkowanej

Przenoszenie kolejki z jednej struktury narzędzia CF do innej

Aby przenieść kolejki i ich komunikaty z jednej struktury CF do innej, należy użyć komendy MQSC [MOVE QLOCAL](#) . Po zidentyfikowaniu kolejki lub kolejek, które mają zostać przeniesione do nowej struktury systemu CF, należy wykonać następującą procedurę, aby przenieść każdą kolejkę:

1. Upewnij się, że kolejka, która ma zostać przeniesiona, nie jest używana przez żadne aplikacje, to znaczy atrybuty kolejki IPPROCS i OPPROCS są równe zero we wszystkich menedżerach kolejek w grupie współużytkowania kolejek.
2. Zapobiegaj przenoszeniu komunikatów do kolejki przez aplikacje, modyfikując definicję kolejki, aby wyłączyć MQPUT . Zmień definicję kolejki na PUT (DISABLED).
3. Zdefiniuj kolejkę tymczasową o tych samych atrybutach, co kolejka, która jest przenoszona za pomocą następującej komendy:

```
DEFINE QL(TEMP_QUEUE) LIKE(Queue_To_Move) PUT(ENABLED) GET(ENABLED) QSGDISP(QMGR)
```

Uwaga: Jeśli ta kolejka tymczasowa istnieje z poprzedniego uruchomienia, usuń ją przed zdefiniowaniem.

4. Przenieś komunikaty do kolejki tymczasowej za pomocą następującej komendy:

```
MOVE QLOCAL(Queue_To_Move) TOQLOCAL(TEMP_QUEUE)
```

5. Usuń kolejkę, która jest przenoszona, używając komendy:

```
DELETE QLOCAL(Queue_To_Move)
```

6. Zmień definicję kolejki, która jest przenoszona, zmieniając atrybut CFSTRUCT, używając następującej komendy:

```
DEFINE QL(Queue_To_Move) LIKE(TEMP_Queue) CFSTRUCT(NEW) QSGDISP(SHARED)
```

Gdy kolejka jest ponownie zdefiniowana, jest ona oparta na kolejce tymczasowej utworzonej w kroku “3” na stronie 409.

7. Przenieś komunikaty z powrotem do nowej kolejki za pomocą komendy:

```
MOVE QLOCAL(TEMP) TOQLOCAL(Queue_To_Move)
```

8. Kolejka utworzona w kroku “3” na stronie 409 nie jest już wymagana. Aby go usunąć, użyj następującej komendy:

```
DELETE QL(TEMP_Queue)
```

9. Jeśli przenoszona kolejka została zdefiniowana w zestawach danych CSQINP2, zmień atrybut CFSTRUCT odpowiedniej komendy DEFINE QLOCAL w zestawach danych CSQINP2. Dodaj słowo kluczowe REPLACE w taki sposób, aby istniejąca definicja kolejki została zastąpiona.

Rysunek 47 na stronie 410 przedstawia przykładowe zadanie przenoszenia kolejki z jednej struktury CF do innej.

```
//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=thlqual.SCSQANLE,DISP=SHR
// DD DSN=thlqual.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
ALTER QL(Queue_To_Move) PUT(DISABLED)
DELETE QL(TEMP_Queue) PURGE
DEFINE QL(TEMP_Queue) LIKE(Queue_To_Move) PUT(ENABLED) GET(ENABLED) QSGDISP(QMGR)
MOVE QLOCAL(Queue_To_Move) TOQLOCAL(TEMP_Queue)
DELETE QL(Queue_To_Move)
DEFINE QL(Queue_To_Move) LIKE(TEMP_Queue) CFSTRUCT(NEW) QSGDISP(SHARED)
MOVE QLOCAL(TEMP_Queue) TOQLOCAL(Queue_To_Move)
DELETE QL(TEMP_Queue)
/*
```

Rysunek 47. Przykładowe zadanie przenoszenia kolejki z jednej struktury CF do innej

Przenoszenie kolejki niewspółużytkowanej do kolejki współużytkowanej

Procedura przenoszenia niewspółużytkowanej kolejki do kolejki współużytkowanej jest taka, jak procedura przenoszenia kolejki z jednej struktury CF do innej (patrz “Przenoszenie kolejki z jednej struktury narzędzia CF do innej” na stronie 409). Rysunek 48 na stronie 411 daje przykładowe zadanie do wykonania tego zadania.

Uwaga: Należy pamiętać, że komunikaty w kolejkach współużytkowanych podlegają pewnym ograniczeniom dotyczącym maksymalnej wielkości komunikatu, trwałości komunikatów i typu indeksu kolejki, dzięki czemu możliwe jest nie przeniesienie niektórych niewspółużytkowanych kolejek do kolejki współużytkowanej.

```
//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=thlqua1.SCSQANLE,DISP=SHR
// DD DSN=thlqua1.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
ALTER QL(Queue_TO_MOVE) PUT(DISABLED)
DELETE QL(TEMP_QUEUE) PURGE
DEFINE QL(TEMP_QUEUE) LIKE(Queue_TO_MOVE) PUT(ENABLED) GET(ENABLED)
MOVE QLOCAL(Queue_TO_MOVE) TOQLOCAL(TEMP_QUEUE)
DELETE QL(Queue_TO_MOVE)
DEFINE QL(Queue_TO_MOVE) LIKE(TEMP_QUEUE) CFSTRUCT(NEW) QSGDISP(SHARED)
MOVE QLOCAL(TEMP_QUEUE) TOQLOCAL(Queue_TO_MOVE)
DELETE QL(TEMP_QUEUE)
/*
```

Rysunek 48. Przykładowe zadanie przenoszenia niewspółużytkowanej kolejki do współużytkowanej kolejki

Przenoszenie kolejki współużytkowanej do kolejki niewspółużytkowanej

Procedura przenoszenia kolejki współużytkowanej do kolejki niewspółużytkowanej jest taka, jak procedura przenoszenia kolejki z jednej struktury CF do innej (patrz [“Przenoszenie kolejki z jednej struktury narzędzia CF do innej”](#) na stronie 409).

Rysunek 49 na stronie 411 daje przykładowe zadanie do wykonania tego zadania.

```
//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=thlqua1.SCSQANLE,DISP=SHR
// DD DSN=thlqua1.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
ALTER QL(Queue_TO_MOVE) PUT(DISABLED)
DELETE QL(TEMP_QUEUE) PURGE
DEFINE QL(TEMP_QUEUE) LIKE(Queue_TO_MOVE) PUT(ENABLED) GET(ENABLED) QSGDISP(QMGR)
MOVE QLOCAL(Queue_TO_MOVE) TOQLOCAL(TEMP_QUEUE)
DELETE QL(Queue_TO_MOVE)
DEFINE QL(Queue_TO_MOVE) LIKE(TEMP_QUEUE) STGCLASS(NEW) QSGDISP(QMGR)
MOVE QLOCAL(TEMP_QUEUE) TOQLOCAL(Queue_TO_MOVE)
DELETE QL(TEMP_QUEUE)
/*
```

Rysunek 49. Przykładowe zadanie przenoszenia współużytkowanej kolejki do kolejki niewspółużytkowanej

Migrowanie kolejek niewspółużytkowanych do współużytkowanych kolejek

Istnieją dwa etapy migrowania niewspółużytkowanych kolejek do współużytkowanych kolejek:

- Migrowanie pierwszego (lub tylko) menedżera kolejek w grupie współużytkowania kolejek
- Migrowanie wszystkich innych menedżerów kolejek w grupie współużytkowania kolejek

Migrowanie pierwszego (lub tylko) menedżera kolejek w grupie współużytkowania kolejek

Rysunek 48 na stronie 411 przedstawia przykładowe zadanie przeniesienia niewspółużytkowanej kolejki do kolejki współużytkowanej. Należy to zrobić dla każdej kolejki, która wymaga migracji.

Uwaga:

1. Komunikaty w kolejkach współużytkowanych podlegają pewnym ograniczeniom dotyczącym maksymalnej wielkości komunikatu, trwałości komunikatów i typu indeksu kolejki, dzięki czemu możliwe jest nie przeniesienie niektórych niewspółużytkowanych kolejek do kolejki współużytkowanej.
2. W przypadku kolejek współużytkowanych należy użyć poprawnego typu indeksu. Jeśli kolejka transmisji jest migrowane do współużytkowanej kolejki, typem indeksu musi być MSGID.

Jeśli kolejka jest pusta lub nie ma potrzeby przechowywania na nim komunikatów, migracja kolejki jest prostsza. Rysunek 50 na stronie 412 przedstawia przykładowe zadanie, które ma być używane w tych okolicznościach.

```
//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=thlqua1.SCSQANLE,DISP=SHR
// DD DSN=thlqua1.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
DELETE QL(TEMP_QUEUE) PURGE
DEFINE QL(TEMP_QUEUE) LIKE(Queue_TO_MOVE) PUT(ENABLED) GET(ENABLED)
DELETE QL(Queue_TO_MOVE)
DEFINE QL(Queue_TO_MOVE) LIKE(TEMP_QUEUE) CFSTRUCT(NEW) QSGDISP(SHARED)
DELETE QL(TEMP_QUEUE)
/*
```

Rysunek 50. Przykładowe zadanie przenoszenia niewspółużytkowanej kolejki bez komunikatów do kolejki współużytkowanej

Migrowanie wszystkich innych menedżerów kolejek w grupie współużytkowania kolejek

1. Dla każdej kolejki, która nie ma tej samej nazwy, co istniejąca kolejka współużytkowana, przenieś ją zgodnie z opisem w sekcji Rysunek 48 na stronie 411 lub Rysunek 50 na stronie 412.
2. W przypadku kolejek o tej samej nazwie co istniejąca kolejka współużytkowana, przenieś komunikaty do kolejki współużytkowanej za pomocą komend przedstawionych w sekcji Rysunek 51 na stronie 412.

```
MOVE QLOCAL(Queue_TO_MOVE) QSGDISP(QMGR) TOQLOCAL(Queue_TO_MOVE)
DELETE QLOCAL(Queue_TO_MOVE) QSGDISP(QMGR)
```

Rysunek 51. Przenoszenie komunikatów z kolejki niewspółużytkowanej do istniejącej kolejki współużytkowanej

Zawieszanie połączenia z serwerem Db2

Aby zastosować konserwację lub usługę do tabel lub pakietów produktu Db2 związanych z kolejkami współużytkowanymi bez zatrzymywania menedżera kolejek, należy tymczasowo odłączyć menedżery kolejek w grupie współużytkującej dane (DSG) z produktu Db2.

W tym celu:

1. Użyj komendy MQSC `SUSPEND QMGR FACILITY (Db2)`.

2. Wykonaj powiązanie.
3. Ponownie nawiąże połączenie z programem Db2 przy użyciu komendy MQSC `RESUME QMGR FACILITY (Db2)`

Należy zauważyć, że istnieją ograniczenia dotyczące korzystania z tych komend.



Ostrzeżenie: Gdy połączenie Db2 jest zawieszona, następujące operacje nie będą dostępne. W związku z tym należy wykonać tę pracę w czasie, gdy przedsiębiorstwo jest najmniej zajęte.

- Dostęp do obiektów kolejki współużytkowanej na potrzeby administrowania (definiowanie, usuwanie, zmiana)
- Uruchamianie kanałów współużytkowanych
- Zapisywanie komunikatów w produkcie Db2
- Utwórz kopię zapasową lub odzyskaj CFSTRUCT

Zarządzanie obiektami grupy

Ten temat zawiera informacje na temat pracy z obiektami grupy.

Produkt IBM MQ automatycznie kopiuje definicję obiektu grupy do zestawu stron zero dla każdego menedżera kolejek, który go używa. Kopię definicji można tymczasowo zmienić, a program IBM MQ umożliwi odświeżenie kopii zestawu stron z kopii repozytorium. Program IBM MQ zawsze próbuje odświeżyć kopie zestawu stron z kopii repozytorium podczas uruchamiania (w przypadku obiektów kanałów, jest to wykonywane po restarcie inicjatora kanału). Zapewnia to, że kopie zestawu stron odzwierciedlają wersję w repozytorium, w tym wszelkie zmiany, które zostały wprowadzone w przypadku, gdy menedżer kolejek był nieaktywny.

Istnieją okoliczności, w których odświeżanie nie jest wykonywane, na przykład:

- Jeśli kopia kolejki jest otwarta, odświeżenie, które spowodowałoby zmianę użycia kolejki, nie powiedzie się.
- Jeśli kopia kolejki ma na niej komunikaty, odświeżenie, które spowodowałoby usunięcie tej kolejki, nie powiedzie się.

W tych okolicznościach odświeżanie nie jest wykonywane dla tej kopii, ale jest wykonywane na kopiach we wszystkich innych menedżerach kolejek. Sprawdź i rozwiążaj problemy związane z kopiowaniem obiektów po dodaniu, zmianie lub usunięciu obiektu grupy, a także w menedżerze kolejek lub restarcie inicjatora kanału.

Zarządzanie narzędziem CF

W tym temacie opisano sposób dodawania lub usuwania struktur CF (Coupling Facility).

W tej sekcji opisano następujące zadania:

- [“Dodawanie struktury narzędzia CF” na stronie 413](#)
- [“Usuwanie struktury narzędzia CF” na stronie 414](#)

Dodawanie struktury narzędzia CF

Aby dodać strukturę narzędzia CF, wykonaj następującą procedurę:

1. Zdefiniuj strukturę CF w zestawie danych strategii CFRM. Informacje na temat konfigurowania narzędzia CF w sekcji [Konfigurowanie narzędzia CF](#) zawierają opis reguł dotyczących nazewnictwa struktur narzędzia CF oraz sposobu definiowania struktur w zestawie danych strategii CFRM.
2. Jeśli chcesz skonfigurować strukturę, aby odciażać dane komunikatu do zestawów danych SMDS, przydziel i preformatuj zestawy danych. Szczegółowe informacje na ten temat zawiera sekcja [Tworzenie współużytkowanego zestawu danych komunikatów](#).
3. Zdefiniuj strukturę na wartość IBM MQ, używając komendy `DEFINE CFSTRUCT`.

Usuwanie struktury narzędzia CF

Aby usunąć strukturę narzędzia CF, wykonaj następującą procedurę:

1. Użyj następującej komendy, aby uzyskać listę wszystkich kolejek za pomocą struktury narzędzia CF, które mają zostać usunięte:

```
DISPLAY QUEUE(*) QSGDISP(SHARED) CFSTRUCT(structure-name)
```

2. Usuń wszystkie kolejki, które korzystają ze struktury.
3. Usuń strukturę CF z programu IBM MQ , używając komendy `DELETE CFSTRUCT` .
4. Jeśli struktura została skonfigurowana w celu odciążania danych komunikatu do SMDS, należy usunąć SMDS.
5. Usuń definicję struktury z zestawu danych strategii CFRM i uruchom program narzędziowy IXCMIAPU. (Jest to odwrotne zadanie dostosowywania, które jest skonfigurowane w narzędziu CF, opisane w sekcji [Konfigurowanie narzędzia CF](#)).

Strojenie monitorowania listy narzędzia CF

Ten temat zawiera informacje na temat monitorowania listy narzędzia CF.

Monitorowanie listy narzędzia CF jest używane do monitorowania stanu struktur list zawierających współużytkowane kolejki produktu IBM MQ . Gdy komunikat jest dodawany do kolejki współużytkowanej, a przejścia głębokości kolejki od zera do wartości niezerowej, system CF powiadamia wszystkie menedżery kolejek w grupie współużytkowania kolejek. Po powiadomieniu menedżerów kolejek może być wykonywana pewna liczba działań, w tym powiadamianie monitorów wyzwalaczy, które używają TRIGGER (FIRST), lub aplikacji, które wykonują operacje get-wait.

Domyślnie system CF powiadamia wszystkie menedżery kolejek w grupie współużytkowania kolejek w tym samym czasie. W niektórych konfiguracjach może to być przyczyną problemów, takich jak:

- Skośny rozkład obciążenia, w którym duży procent komunikatów trafia do konkretnego menedżera kolejek w grupie współużytkowania kolejek, często menedżera kolejek działającego na najszybszej partycji LPAR lub który znajduje się najbliżej systemu CF, lub
- Duża liczba pobrań nie powiodła się, co spowodowało zmarnowanie czasu pracy procesora.

Program z/OS V2R3 wprowadza nowy atrybut menedżera zasobów narzędzia CF o nazwie **KEYRNOTIFYDELAY**, który może być używany z strukturami listy zawierającymi współużytkowane kolejki (to znaczy struktury aplikacji, a nie strukturę administracyjną), a które mogą, dla niektórych obciążeń, zminimalizować skutki skośowania obciążenia i pustych wywołań MQGET lub pustych wywołań MQGET.

KEYRNOTIFYDELAY można ustawić tylko w strukturach w systemie CF, działającym na poziomie CFLEVEL 22 lub wyższym.

Jego wartość musi być jedną z siedmiu cyfr dziesiętnych, w zakresie od 0 do 1 000 000 mikrosekund. W przypadku ustawienia wartości niezerowej i głębokości przejść kolejki od zera do wartości niezerowej system CF wybiera jeden menedżer kolejek z grupy współużytkowania kolejek i powiadamia o tym menedżera kolejek przed wszystkimi pozostałymi menedżerami kolejek w grupie.

Menedżer kolejek jest wybierany w trybie karuzelowym. Jeśli wybrany menedżer kolejek nie przetworzy komunikatu w przedziale czasu opisanym przez program **KEYRNOTIFYDELAY** , zostaną również powiadomione wszystkie pozostałe menedżery kolejek w grupie współużytkowania kolejek.

Więcej informacji na temat produktu **KEYRNOTIFYDELAY** znajduje się w następującym miejscu: [Understanding Keyrange Monitoring Notification Delay](#).

Należy zauważyć, że istnieją dwa podobne atrybuty CFRM o nazwach **LISTNOTIFYDELAY** i **SUBNOTIFYDELAY**. Żadne z nich nie ma mierzalnego wpływu na obciążenie IBM MQ .

W tym temacie opisano mechanizmy odtwarzania i restartowania używane przez produkt IBM MQ.

restartowanie IBM MQ

Po zakończeniu działania menedżera kolejek istnieją różne procedury restartowania, które są wymagane w zależności od tego, jak menedżer kolejek zakończył działanie. W tym temacie opisano różne procedury restartowania, których można użyć.

Ten temat zawiera informacje na temat restartowania menedżera kolejek w następujących okolicznościach:

- [“Restartowanie po normalnym zamknięciu systemu” na stronie 415](#)
- [“Restartowanie po nieprawidłowym zakończeniu” na stronie 415](#)
- [“Restartowanie, jeśli utracono zestawy stron” na stronie 415](#)
- [“Restartowanie, jeśli utracono zestawy danych dziennika” na stronie 416](#)
- [Restartowanie, jeśli utracieś struktury CF](#)

Restartowanie po normalnym zamknięciu systemu

Jeśli menedżer kolejek został zatrzymany za pomocą komendy STOP QMGR, system zakończy pracę w uporządkowany sposób i przed zatrzymaniem podejmie punkt kontrolny zakończenia. Po zrestartowaniu menedżera kolejek używa on informacji z systemowego punktu kontrolnego i dziennika odtwarzania w celu określenia statusu systemu podczas jego zamknięcia.

Aby zrestartować menedżer kolejek, należy wprowadzić komendę START QMGR zgodnie z opisem w sekcji [“Uruchamianie i zatrzymywanie menedżera kolejek w systemie z/OS” na stronie 342](#).

Restartowanie po nieprawidłowym zakończeniu

Program IBM MQ automatycznie wykrywa, czy restart następuje po normalnym zamknięciu systemu, czy też nieprawidłowym zakończeniu.

Uruchomienie menedżera kolejek po jego nieprawidłowym zakończeniu jest inne niż uruchomienie komendy po wydaniu komendy STOP QMGR. Jeśli menedżer kolejek zakończy działanie w sposób nieprawidłowy, kończy działanie bez możliwości zakończenia pracy lub wykonania punktu kontrolnego zakończenia.

Aby zrestartować menedżer kolejek, należy wprowadzić komendę START QMGR zgodnie z opisem w sekcji [“Uruchamianie i zatrzymywanie menedżera kolejek w systemie z/OS” na stronie 342](#). Po zrestartowaniu menedżera kolejek po nieprawidłowym zakończeniu odświeża on swoją wiedzę o jego statusie podczas kończenia pracy z użyciem informacji w dzienniku i powiadamia użytkownika o statusie różnych zadań.

Zwykle proces restartowania rozwiązuje wszystkie niespójne stany. Jednak w niektórych przypadkach konieczne jest podjęcie konkretnych kroków w celu usunięcia niespójności. Jest to opisane w sekcji [“Ręczne odzyskiwanie jednostek pracy” na stronie 429](#).

Restartowanie, jeśli utracono zestawy stron

Jeśli utracono zestawy stron, należy je odtworzyć z kopii zapasowych przed zrestartowaniem menedżera kolejek. Jest to opisane w sekcji [“Tworzenie kopii zapasowych i odtwarzanie zestawów stron” na stronie 399](#).

Restartowanie menedżera kolejek w tych okolicznościach może zająć dużo czasu ze względu na czas potrzebny na odtwarzanie nośników.

Restartowanie, jeśli utracono zestawy danych dziennika

Jeśli po zatrzymaniu menedżera kolejek (za pomocą komendy STOP QMGR) obie kopie dziennika zostaną utracone lub uszkodzone, można zrestartować menedżer kolejek udostępniający spójny zestaw zestawów stron (tworzony przy użyciu metody [Metoda 1: pełna kopia zapasowa](#)).

Wykonaj następującą procedurę:

1. Zdefiniuj nowe zestawy stron tak, aby odpowiadała każdemu istniejącym zestawowi stron w menedżerze kolejek. Więcej informacji na temat definicji zestawu stron można znaleźć w sekcji [Czynność 15: Definiowanie zestawów stron](#).
- Upewnij się, że każdy nowy zestaw stron jest większy niż odpowiedni zestaw stron źródłowych.
2. Aby sformatować docelowy zestaw stron, należy użyć funkcji FORMAT CSQUTIL. Więcej informacji na ten temat zawiera sekcja [Formatowanie zestawów stron](#).
3. Użyj funkcji RESETPAGE programu CSQUTIL w celu skopiowania istniejących zestawów stron lub zresetowania ich w miejscu, a następnie zresetuj wartość RBA dziennika na każdej stronie. Więcej informacji na temat tej funkcji zawiera sekcja [Kopiowanie zestawu stron i resetowanie dziennika](#).
4. Ponownie zdefiniuj zestawy danych dziennika menedżera kolejek i BSDS za pomocą komendy CSQJU003 (patrz [Program narzędziowy do rejestrowania dziennika zmian](#)).
5. Zrestartuj menedżer kolejek przy użyciu nowych zestawów stron. Aby to zrobić, wykonaj jedną z następujących czynności:
 - Zmień procedurę uruchomionego zadania menedżera kolejek w taki sposób, aby odwoła się do nowych zestawów stron. Więcej informacji na ten temat zawiera sekcja [Czynność 6: Tworzenie procedur dla menedżera kolejek produktu IBM MQ](#).
 - Użyj usług metod dostępu, aby usunąć stare zestawy stron, a następnie zmień nazwy nowych zestawów stron, nadając im takie same nazwy, jak stare zestawy stron.

Uwaga: Przed usunięciem zestawu stron IBM MQ należy upewnić się, że wykonano wymagane kopie zapasowe.

Jeśli menedżer kolejek jest elementem grupy współużytkowania kolejek, to definicje obiektów GROUP i SHARED nie są zazwyczaj naruszane przez utracone lub uszkodzone dzienniki. Jeśli jednak wszystkie komunikaty kolejki współużytkowanej są zaangażowane w jednostkę pracy, która została pokryta przez utracone lub uszkodzone dzienniki, wpływ na takie niezatwierdzone komunikaty jest nieprzewidywalny.

Uwaga: Jeśli dzienniki są uszkodzone, a menedżer kolejek jest elementem grupy współużytkowania kolejek, to możliwość odtwarzania współużytkowanych trwałych komunikatów może zostać utracona. Wydadaj komendę BACKUP CFSTRUCT natychmiast dla innego aktywnego menedżera kolejek w grupie współużytkowania kolejki dla wszystkich struktur CF z atrybutem RECOVER (YES).

Restartowanie, jeśli utraciłeś struktury CF

Nie trzeba restartować, jeśli struktura CF jest tracona, ponieważ menedżer kolejek nie kończy działania.

Alternatywne odtwarzanie ośrodka w systemie z/OS

Można odtworzyć pojedynczy menedżer kolejek lub grupę współużytkowania kolejek lub rozważyć zapis lustrzany dysków.

Więcej informacji można znaleźć w następujących sekcjach:

- [Odtwarzanie pojedynczego menedżera kolejek w ośrodku alternatywnym](#)
- [Odtwarzanie grupy współużytkowania kolejek.](#)
 - [Odtwarzanie nośników struktury CF](#)
 - [Tworzenie kopii zapasowej grupy współużytkowania kolejek w ośrodku podstawowym](#)
 - [Odtwarzanie grupy współużytkowania kolejek w ośrodku alternatywnym](#)
- [Korzystanie z zapisu lustrzanego na dysku](#)

Odtwarzanie pojedynczego menedżera kolejek w ośrodku alternatywnym

Jeśli dojdzie do całkowitej utraty centrum przetwarzania danych IBM MQ, można je odtworzyć w innym menedżerze kolejek lub grupie współużytkownika kolejek w ośrodku zapasowym. (W przypadku alternatywnej procedury odtwarzania serwisu dla grupy współużytkownika kolejek należy zapoznać się z [“Odtwarzanie grupy współużytkownika kolejek w ośrodku alternatywnym”](#) na stronie 420).

Aby można było odtworzyć dane w innym menedżerze kolejek w ośrodku odtwarzania, należy regularnie tworzyć kopie zapasowe zestawów stron i dzienników. Podobnie jak w przypadku wszystkich operacji odtwarzania danych, celem odzyskiwania po awarii jest utrata jako małych danych, przetwarzania obciążenia (aktualizacji) i czasu, jak to możliwe.

W ośrodku zapasowym:

- Menedżery kolejek odtwarzania **muszą** mieć takie same nazwy, jak utracone menedżery kolejek.
- Moduł parametrów systemowych (na przykład CSQZPARM) używany w każdym menedżerze kolejek odtwarzania musi zawierać te same parametry, co odpowiadający mu utracony menedżer kolejek.

Po wykonaniu tej czynności należy ponownie ustanowić wszystkie menedżery kolejek zgodnie z opisem w poniższej procedurze. Może to być używane do odtwarzania po awarii w ośrodku zapasowym dla pojedynczego menedżera kolejek. Zakłada się, że wszystkie dostępne są następujące elementy:

- Kopie dzienników archiwalnych i BSDSs utworzone przez normalne działanie w ośrodku podstawowym (dzienniki aktywne zostaną utracone wraz z menedżerem kolejek w ośrodku podstawowym).
- Kopie zestawów stron z menedżera kolejek w ośrodku podstawowym są takie same lub starsze niż najnowsze dostępne kopie dzienników archiwalnych.

W przypadku dzienników aktywnych i archiwalnych można użyć podwójnego rejestrowania, w którym to przypadku należy zastosować aktualizacje BSDS do obu kopii:

1. Zdefiniuj nowe zestawy danych zestawu stron i załaduj je wraz z danymi w kopiach zestawów stron z ośrodka podstawowego.
2. Zdefiniuj nowe aktywne zestawy danych dziennika.
3. Zdefiniuj nowy zestaw danych BSDS i użyj programu Access Method Services REPRO, aby skopiować do niej *najnowsze* zarchiwizowane BSDS.
4. Użyj programu narzędziowego do tworzenia map dzienników CSQJU004, aby wydrukować informacje z najnowszego BSDS. W czasie, gdy ten BSDS został zarchiwizowany, ostatnio zarchiwizowany dziennik zostałby obciążony jako aktywny dziennik i nie jest wyświetlany jako zarchiwizowany dziennik. Zapisz plik STARTRBA i ENDRBA tego dziennika.
5. Użyj programu narzędziowego do spisywania zasobów dziennika zmian CSQJU003, aby zarejestrować ten najnowszy zestaw danych dziennika archiwalnego w BSDS, które właśnie zostały odtworzone, za pomocą komendy STARTRBA i ENDRBA odnotowanych w kroku [“4”](#) na stronie 417.
6. Użyj opcji DELETE CSQJU003, aby usunąć wszystkie aktywne informacje dziennika z BSDS.
7. Użyj opcji NEWLOG CSQJU003, aby dodać aktywne dzienniki do BSDS, nie określaj STARTRBA ani ENDRBA.
8. Użyj komendy CSQJU003, aby dodać rekord sterowania restartowaniem do BSDS. Podaj CRESTART CREATE, ENDRBA=highrba, gdzie highrba jest wysokim RBA najnowszego dostępnego dziennika archiwalnego (znalezionego w kroku [“4”](#) na stronie 417), plus 1.

BSDS opisuje teraz wszystkie aktywne dzienniki jako puste, wszystkie zarchiwizowane dzienniki, które są dostępne, i nie ma punktów kontrolnych poza końcem dzienników.

9. Zrestartuj menedżer kolejek za pomocą komendy START QMGR. Podczas inicjowania generowany jest komunikat odpowiedzi operatora, taki jak następujący:

```
CSQJ245D +CSQ1 RESTART CONTROL INDICATES TRUNCATION AT RBA highrba.  
REPLY Y TO CONTINUE, N TO CANCEL
```

Wpisz Y , aby uruchomić menedżer kolejek. Menedżer kolejek zostanie uruchomiony i odzyskuje dane aż do ENDRBA określonego w instrukcji CRESTART.

Informacje na temat używania programów narzędziowych CSQJU003 i CSQJU004 zawiera sekcja Korzystanie z programów narzędziowych produktu IBM MQ .

W poniższym przykładzie przedstawiono przykładowe instrukcje wejściowe dla CSQJU003 dla kroków 6, 7 i 8:

```
* Step 6
DELETE DSNAME=MQM2.LOGCOPY1.DS01
DELETE DSNAME=MQM2.LOGCOPY1.DS02
DELETE DSNAME=MQM2.LOGCOPY1.DS03
DELETE DSNAME=MQM2.LOGCOPY1.DS04
DELETE DSNAME=MQM2.LOGCOPY2.DS01
DELETE DSNAME=MQM2.LOGCOPY2.DS02
DELETE DSNAME=MQM2.LOGCOPY2.DS03
DELETE DSNAME=MQM2.LOGCOPY2.DS04

* Step 7
NEWLOG DSNAME=MQM2.LOGCOPY1.DS01,COPY1
NEWLOG DSNAME=MQM2.LOGCOPY1.DS02,COPY1
NEWLOG DSNAME=MQM2.LOGCOPY1.DS03,COPY1
NEWLOG DSNAME=MQM2.LOGCOPY1.DS04,COPY1
NEWLOG DSNAME=MQM2.LOGCOPY2.DS01,COPY2
NEWLOG DSNAME=MQM2.LOGCOPY2.DS02,COPY2
NEWLOG DSNAME=MQM2.LOGCOPY2.DS03,COPY2
NEWLOG DSNAME=MQM2.LOGCOPY2.DS04,COPY2

* Step 8
CRESTART CREATE,ENDRBA=063000
```

Rzeczy, które należy uwzględnić podczas restartowania inicjatora kanału w ośrodku odtwarzania, są podobne do tych, które mają miejsce podczas używania ARM do restartowania inicjatora kanału na innym obrazie produktu z/OS . Więcej informacji zawiera temat “Używanie ARM w sieci IBM MQ” na stronie 427. Strategia odtwarzania powinna również obejmować odzyskiwanie bibliotek produktu IBM MQ oraz środowisk programowania aplikacji, które korzystają z produktu IBM MQ (na przykład CICS).

Inne funkcje programu narzędziowego do spisywania zasobów dziennika zmian (CSQJU003) mogą być również używane w scenariuszach odtwarzania po awarii. Funkcja HIGHRBA umożliwia aktualizację najwyższych wartości RBA zapisanych i najwyższych RBA przesuniętych wartości w ramach zestawu danych programu startowego. Funkcja CHECKPT umożliwia dodawanie nowych rekordów kolejek punktów kontrolnych lub usuwanie istniejących rekordów kolejek punktów kontrolnych w BSDS.

Uwaga: Te funkcje mogą mieć wpływ na integralność danych produktu IBM MQ . Używaj ich tylko w scenariuszach odtwarzania po awarii pod kierunkiem personelu serwisu IBM .

Techniki szybkiego kopiowania

Jeśli kopie wszystkich zestawów stron i dzienników są wykonywane w czasie, gdy menedżer kolejek jest zamrożony, kopie te będą spójnym zestawem, który może zostać użyty do zrestartowania menedżera kolejek w ośrodku alternatywnym. Zwykle umożliwiają one znacznie szybsze restartowanie menedżera kolejek, ponieważ ma być wykonywane małe odtwarzanie nośników.

Użyj komendy SUSPEND QMGR LOG, aby zamrozić menedżera kolejek. Ta komenda powoduje opróżnienie pul buforów do zestawów stron, wykonanie punktu kontrolnego i zatrzymanie dalszych działań zapisu w dzienniku. Po zawieszeniu działania zapisu w dzienniku menedżer kolejek jest w rzeczywistości zamrożony do czasu wydania komendy RESUME QMGR LOG. Podczas gdy menedżer kolejek jest zamrożony, zestawy stron i dzienniki mogą być kopiowane.

Korzystając z narzędzi do kopiowania, takich jak FLASHCOPY lub SNAPSHOT, aby szybko kopiować zestawy stron i dzienniki, czas, w którym menedżer kolejek jest zamrożony, może być zmniejszony do minimum.

Jednak w obrębie grupy współużytkowania kolejek komenda SUSPEND QMGR LOG może nie być tak dobrym rozwiązaniem. Aby możliwe było odtworzenie kopii dzienników, wszystkie kopie dzienników

muszą zawierać ten sam punkt w czasie odtwarzania, co oznacza, że komenda SUSPEND QMGR LOG musi być uruchomiona jednocześnie we wszystkich menedżerach kolejek w grupie współużytkowania kolejki, dlatego cała grupa współużytkowania kolejki zostanie przez pewien czas zamrożona.

Odtwarzanie grupy współużytkowania kolejek

W przypadku awarii ośrodka podstawowego można zrestartować grupę współużytkowania kolejek w ośrodku zdalnym przy użyciu zestawów danych kopii zapasowych z ośrodka podstawowego. Aby odzyskać grupę współużytkowania kolejek, należy skoordynować odtwarzanie we wszystkich menedżerach kolejek w grupie współużytkowania kolejek i skoordynować z innymi zasobami, przede wszystkim z produktem Db2. W tej sekcji opisano szczegółowo te zadania.

- [Odtwarzanie nośników struktury CF](#)
- [Tworzenie kopii zapasowej grupy współużytkowania kolejek w ośrodku podstawowym](#)
- [Odtwarzanie grupy współużytkowania kolejek w ośrodku alternatywnym](#)

Odtwarzanie nośnika struktury CF

Odtwarzanie nośników struktury CF używanej do przechowywania trwałych komunikatów w kolejce współużytkowanej polega na utworzeniu kopii zapasowej nośnika, który może być odtwarzający, odzyskany przez aplikację zarejestrowanych aktualizacji. Należy okresowo tworzyć kopie zapasowe struktur systemu CF przy użyciu komendy MQSC BACKUP CFSTRUCT. Wszystkie aktualizacje w kolejkach współużytkowanych (MQGETi MQPUT) są zapisywane w dzienniku menedżera kolejek, w którym wykonywana jest aktualizacja. Aby przeprowadzić odtwarzanie nośnika w strukturze CF, należy zastosować zarejestrowane aktualizacje dla tej kopii zapasowej z dzienników wszystkich menedżerów kolejek, które użyły tej struktury CF. Gdy używana jest komenda MQSC RECOVER CFSTRUCT, produkt IBM MQ automatycznie scala dzienniki z odpowiednich menedżerów kolejek i stosuje aktualizacje do najnowszej kopii zapasowej.

Kopia zapasowa struktury CF jest zapisywana w dzienniku menedżera kolejek, który przetworzył komendę BACKUP CFSTRUCT, dlatego nie ma żadnych dodatkowych zestawów danych, które mają być gromadzone i transportowane do ośrodka alternatywnego.

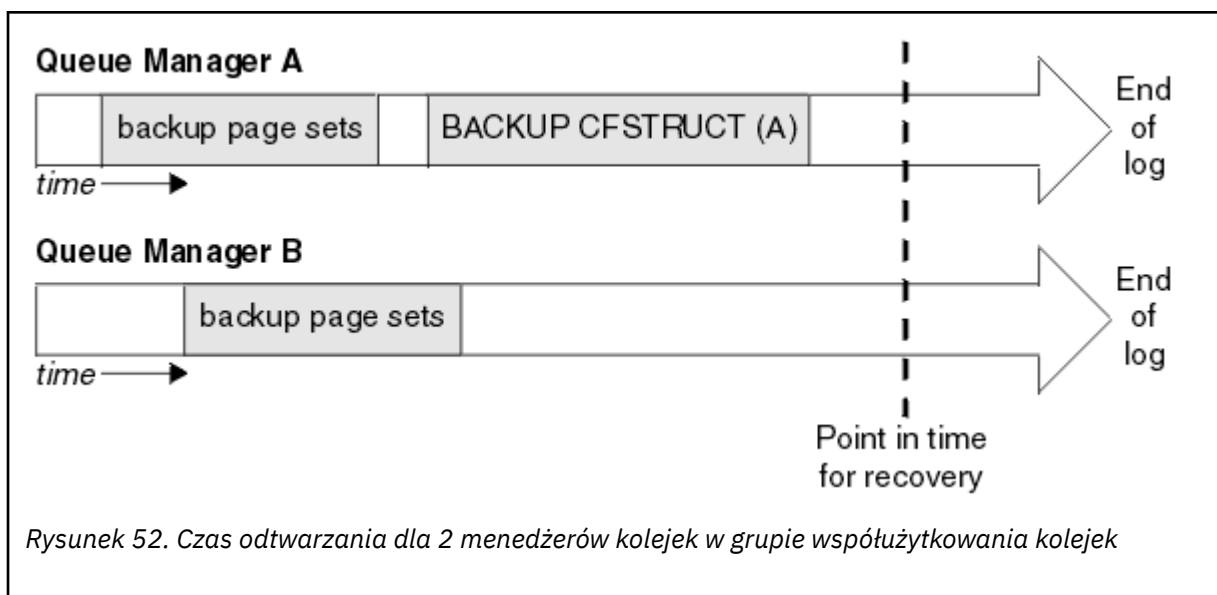
Tworzenie kopii zapasowej grupy współużytkowania kolejek w ośrodku podstawowym

W ośrodku podstawowym konieczne jest regularne tworzenie spójnego zestawu kopii zapasowych, które mogą być używane w przypadku awarii w celu odbudowania grupy współużytkowania kolejek w ośrodku alternatywnym. W przypadku pojedynczego menedżera kolejek odtwarzanie może mieć miejsce w dowolnym momencie, zwykle na końcu dzienników dostępnych w ośrodku zdalnym. Jednak w przypadku, gdy komunikaty trwałe zostały zapisane w kolejce współużytkowanej, dzienniki wszystkich menedżerów kolejek w grupie współużytkowania kolejek muszą zostać scalone w celu odtworzenia kolejek współużytkowanych, ponieważ każdy menedżer kolejek w grupie współużytkowania kolejek mógł przeprowadzać aktualizacje (MQPUT lub MQGET) w kolejce.

Aby możliwe było odtwarzanie grupy współużytkowania kolejek, należy ustanowić punkt w czasie, który znajduje się w zakresie dziennika danych dziennika wszystkich menedżerów kolejek. Ponieważ jednak możliwe jest tylko **przekazanie** odtwarzania nośników z dziennika, ten punkt w czasie musi znajdować się po wydaniu komendy BACKUP CFSTRUCT i po wykonaniu wszystkich kopii zapasowych zestawu stron. (Zwykle punkt w czasie odtwarzania może odpowiadać końcowi dnia biznesowego lub tygodnia).

Na poniższym diagramie przedstawiono linie czasu dla dwóch menedżerów kolejek w grupie współużytkowania kolejek. Dla każdego menedżera kolejek wykonywane są rozmyte kopie zapasowe zestawów stron (patrz Metoda 2: kopia zapasowa rozmyta). W menedżerze kolejek A została wydana komenda BACKUP CFSTRUCT. Następnie komenda ARCHIVE LOG jest wydawana w każdym menedżerze kolejek w celu obciążenia aktywnego dziennika i skopiowania go na nośnik w trybie bez połączenia z menedżera kolejek, który może zostać przetransportowany do ośrodka alternatywnego. Koniec dziennika określa czas, w którym została wydana komenda ARCHIVE LOG, i dlatego określa

zakres danych dziennika, które są zwykle dostępne w ośrodku alternatywnym. Punkt w czasie odtwarzania musi znajdować się między końcem dowolnego zestawu stron lub kopii zapasowych struktury systemu CF, a najwcześniejszym końcem dziennika dostępnym w ośrodku alternatywnym.



IBM MQ rejestruje informacje powiązane z kopiami zapasowymi struktury CF w tabeli w produkcji Db2. W zależności od wymagań użytkownik może chcieć skoordynować punkt w czasie w celu odtworzenia IBM MQ z tym, że w przypadku produktu Db2 może być wystarczające do wykonania kopii tabeli IBM MQ CSQ.ADMIN_B_STRBACKUP po zakończeniu wykonywania komend BACKUP CFSTRUCT.

Aby przygotować się do odtwarzania:

1. Utwórz kopie zapasowe zestawu stron dla każdego menedżera kolejek w grupie współużytkowania kolejek.
2. Wydadź komendę BACKUP CFSTRUCT dla każdej struktury CF z atrybutem RECOVER (YES). Te komendy można wydać z jednego menedżera kolejek lub z różnych menedżerów kolejek w grupie współużytkowania kolejek w celu zrównoważenia obciążenia.
3. Po zakończeniu tworzenia kopii zapasowych wydaj komendę ARCHIVE LOG, aby przetączyć aktywny dziennik i utworzyć kopie dzienników i plików BSDSs każdego menedżera kolejek w grupie współużytkowania kolejek.
4. Należy transportować kopie zapasowe zestawów stron, zarchiwizowane dzienniki, zarchiwizowane BSDS wszystkich menedżerów kolejek w grupie współużytkowania kolejek oraz wybrane informacje o kopii zapasowej Db2 poza siedzibą firmy.

Odtwarzanie grupy współużytkowania kolejek w ośrodku alternatywnym

Aby możliwe było odzyskanie grupy współużytkowania kolejek, należy przygotować środowisko:

1. Jeśli podczas instalowania grupy współużytkowania kolejek w narzędziu CF znajdują się stare informacje na temat narzędzia CF, należy najpierw wyczyścić ten element:

Uwaga: Jeśli w narzędziu CF nie ma starych informacji, można pominąć ten krok.

- a. Wprowadź następującą komendę z/OS, aby wyświetlić struktury CF dla tej grupy współużytkowania kolejek:

```
D XCF,STRUCTURE,STRNAME= qsgname
```

- b. Dla wszystkich struktur, które rozpoczynają się od nazwy grupy współużytkowania kolejek, użyj komendy z/OS SETXCF FORCE CONNECTION, aby wymusić połączenie z tymi strukturami:

```
SETXCF FORCE,CONNECTION,STRNAME= strname,CONNAME=ALL
```

- c. Usuń wszystkie struktury CF za pomocą następującej komendy dla każdej struktury:

```
SETXCF FORCE,STRUCTURE,STRNAME= strname
```

2. Odtwórz systemy Db2 i grupy współużytkowania danych.
3. Odzyskaj CSQ.ADMIN_B_STRBACKUP, która zawiera informacje na temat najnowszych kopii zapasowych utworzonych w serwisie głównym.

Uwaga: Ważne jest, aby tabela STRBACKUP zawierała najnowsze informacje o kopii zapasowej struktury. Starsze informacje dotyczące tworzenia kopii zapasowej struktury mogą wymagać zbiorów danych, które zostały usunięte w wyniku uzyskania informacji podanych przez ostatnią komendę DISPLAY USAGE TYPE (DATASET), co oznaczałoby, że odzyskana struktura CF nie zawiera dokładnych informacji.

4. Uruchom komendę ADD QMGR w programie narzędziowym CSQ5PQSG dla każdego menedżera kolejek w grupie współużytkowania kolejki. Spowoduje to utworzenie pozycji grupy XCF dla każdego menedżera kolejek.

Gdy program narzędziowy jest uruchamiany w tym scenariuszu, następujące komunikaty są normalne:

```
CSQU566I Unable to get attributes for admin structure, CF not found  
or not allocated  
CSQU546E Unable to add QMGR queue_manager_name entry,  
already exists in DB2 table CSQ.ADMIN_B_QMGR  
CSQU148I CSQ5PQSG Utility completed, return code=4
```

Aby odzyskać menedżery kolejek w grupie współużytkowania kolejek:

1. Zdefiniuj nowe zestawy danych zestawu stron i załaduj je wraz z danymi w kopiach zestawów stron z ośrodka podstawowego.
2. Zdefiniuj nowe aktywne zestawy danych dziennika.
3. Zdefiniuj nowy zestaw danych BSDS i użyj programu Access Method Services REPRO, aby skopiować do niej *najnowsze* zarchiwizowane BSDS.
4. Użyj programu narzędziowego do tworzenia map dzienników CSQJU004, aby wydrukować informacje z najnowszego BSDS. W czasie, gdy ten BSDS został zarchiwizowany, ostatnio zarchiwizowany dziennik zostałby obciążony jako aktywny dziennik i nie jest wyświetlany jako zarchiwizowany dziennik. Zanotuj wartości parametrów STARTRBA, STARTLRSN, ENDRBA i ENDLRSN tego dziennika.
5. Użyj programu narzędziowego do spisywania zasobów dziennika zmian CSQJU003, aby zarejestrować ten najnowszy zestaw danych dziennika archiwalnego w BSDS, który właśnie został odtworzony, z użyciem wartości zapisanych w kroku "4" na stronie 421.
6. Użyj opcji DELETE CSQJU003, aby usunąć wszystkie aktywne informacje dziennika z BSDS.
7. Użyj opcji NEWLOG CSQJU003, aby dodać aktywne dzienniki do BSDS, nie określaj STARTRBA ani ENDRBA.
8. Oblicz wartość *recoverylrsn* dla grupy współużytkowania kolejek. Parametr *recoverylrsn* jest najniższym z ENDLRSNs we wszystkich menedżerach kolejek w grupie współużytkowania kolejek (zapisanej w kroku "4" na stronie 421) minus 1. Na przykład, jeśli w grupie współużytkowania kolejek istnieją dwa menedżery kolejek, a ENDLRSN dla jednego z nich to B713 3C72 22C5, a dla drugiego to B713 3D45 2123, to *recoverylrsn* to B713 3C72 22C4.
9. Użyj komendy CSQJU003, aby dodać rekord sterowania restartowaniem do BSDS. Podaj:

```
CRESTART CREATE,ENDLRSN= recoverylrsn
```

gdzie `recoveryLrsn` jest wartością zapisaną w kroku "8" na stronie 421.

BSDS opisuje teraz wszystkie aktywne dzienniki jako puste, wszystkie zarchiwizowane dzienniki, które są dostępne, i nie ma punktów kontrolnych poza końcem dzienników.

Należy dodać rekord CRESTART do BSDS dla każdego menedżera kolejek w ramach grupy współużytkowania kolejki.

10. Zrestartuj każdy menedżer kolejek w grupie współużytkowania kolejek za pomocą komendy START QMGR. Podczas inicjowania generowany jest komunikat odpowiedzi operatora, taki jak następujący:

```
CSQJ245D +CSQ1 RESTART CONTROL INDICATES TRUNCATION AT RBA highrba.  
REPLY Y TO CONTINUE, N TO CANCEL
```

Odpowiedz Y , aby uruchomić menedżer kolejek. Menedżer kolejek zostanie uruchomiony i odzyskuje dane aż do ENDRBA określonego w instrukcji CRESTART.

W przypadku systemu IBM WebSphere MQ 7.0.1 i jego nowszych wersji uruchomiony pierwszy menedżer kolejek może odbudować partycje struktury administracyjnej dla innych członków grupy współużytkowania kolejek, jak również własne, i nie jest już konieczne restartowanie każdego menedżera kolejek w grupie współużytkowania kolejek na tym etapie.

11. Gdy dane struktury administracyjnej dla wszystkich menedżerów kolejek zostały odbudowane, wydaj komendę RECOVER CFSTRUCT dla każdej struktury aplikacji CF.

Jeśli komenda RECOVER CFSTRUCT zostanie wydana dla wszystkich struktur w pojedynczym menedżerze kolejek, proces scalania dzienników zostanie przeprowadzony tylko raz, to jest szybsze niż wydanie komendy w innym menedżerze kolejek dla każdej struktury CF, w której każdy menedżer kolejek musi wykonać krok scalania dziennika.

Gdy przetwarzanie warunkowego restartu jest używane w grupie współużytkowania kolejek, w produkcie IBM WebSphere MQ 7.0.1 i nowszych menedżerach kolejek, wykonujących odbudowywanie węzłów sieci, należy sprawdzić, czy elementy BSDS są takie same, jak CRESTART LRSN, co własne. Ma to na celu zapewnienie integralności odbudowanej struktury administracyjnej. Dlatego ważne jest zrestartowanie innych węzłów w QSG, dzięki czemu będą mogli przetwarzać własne informacje CRESTART, przed kolejnym bezwarunkowym restartowaniem dowolnego członka grupy.

Używanie zapisu lustrzanego na dysku

W wielu instalacjach używane są obecnie technologie zapisu lustrzanego dysków, takie jak IBM Metro Mirror (dawniej PPRC), w celu wykonywania synchronicznych kopii zestawów danych w ośrodku alternatywnym. W takich sytuacjach wiele kroków szczegółowych staje się niepotrzebne, ponieważ zestawy stron produktu IBM MQ i dzienniki w ośrodku alternatywnym są w rzeczywistości identyczne z tymi, które znajdują się w ośrodku podstawowym. W przypadku korzystania z takich technologii, kroki w celu zrestartowania grupy współużytkowania kolejek w ośrodku alternatywnym można podsumować jako:

- Wyczyść struktury CF IBM MQ w ośrodku alternatywnym. (Te często zawierają informacje uzupełniające z poprzednich operacji odzyskiwania po awarii).
- Odtwarzanie systemów Db2 i wszystkich tabel w bazie danych używanej przez grupę współużytkowania kolejek produktu IBM MQ .
- Zrestartuj menedżery kolejek. Przed IBM WebSphere MQ 7.0.1 konieczne jest zrestartowanie każdego menedżera kolejek zdefiniowanego w grupie współużytkowania kolejki, ponieważ każda kolejka zarządza odzyskiwaniem własnej partycji struktury administracyjnej podczas restartu menedżera kolejek. Po zrestartowaniu każdego menedżera kolejek te, które nie znajdują się w głównej partycji LPAR, mogą zostać ponownie zamknięte. W przypadku systemu IBM WebSphere MQ 7.0.1 i nowszych menedżer kolejek, który uruchomił pierwszy menedżer kolejek, odbudowuje partycje struktury administracyjnej dla innych członków grupy współużytkowania kolejek, jak również własne, i nie jest już konieczne restartowanie każdego menedżera kolejek w grupie współużytkowania kolejek.

- Po odbudowaniu struktury administracyjnej odzyskaj struktury aplikacji.

Reinicjowanie menedżera kolejek

Jeśli menedżer kolejek został zakończony nieprawidłowo, może nie być w stanie go zrestartować. Może to być spowodowane tym, że zestawy stron lub dzienniki zostały utracone, obcięte lub uszkodzone. W takim przypadku może być konieczne ponowne zainicjowanie menedżera kolejek (wykonanie zimnego startu).

Uwaga

Jeśli menedżer kolejek nie może zostać zrestartowany w inny sposób, należy wykonać tylko zimny start. Wykonanie zimnego startu umożliwi odtworzenie menedżera kolejek i definicji obiektów. **Nie** będzie możliwe odtworzenie danych komunikatu. Należy sprawdzić, czy żaden z pozostałych scenariuszy restartu opisanych w tym temacie nie działa przed wykonaniem przez użytkownika.

Po zrestartowaniu wszystkie obiekty produktu IBM MQ są zdefiniowane i są dostępne do użycia, ale nie ma danych komunikatu.

Uwaga: Nie należy reinicjować menedżera kolejek w czasie, gdy jest on częścią klastra. Najpierw należy usunąć menedżera kolejek z klastra (przy użyciu komend RESET CLUSTER w innych menedżerach kolejek w klastrze), a następnie ponownie zainicjować go, a następnie ponownie wprowadzić do klastra jako nowy menedżer kolejek.

Jest to spowodowane tym, że podczas ponownego inicjowania identyfikator menedżera kolejek (QMID) jest zmieniany, dlatego każdy obiekt klastra o starym identyfikatorze menedżera kolejek musi zostać usunięty z klastra.

Więcej informacji na ten temat zawierają następujące sekcje:

- [Reinicjowanie menedżera kolejek, który nie znajduje się w grupie współużytkowania kolejek](#)
- [Reinicjowanie menedżerów kolejek w grupie współużytkowania kolejek](#)

Reinicjowanie menedżera kolejek, który nie znajduje się w grupie współużytkowania kolejek

Aby reinicjować menedżera kolejek, należy wykonać następującą procedurę:

1. Przygotuj instrukcje definicji obiektu, które mają być używane przy restartowaniu menedżera kolejek. W tym celu wykonaj jedną z następujących czynności:
 - Jeśli zestaw stron zero jest dostępny, należy użyć funkcji CSQUTIL SDEFS (patrz sekcja [Tworzenie listy komend definiowania IBM MQ](#)). Należy uzyskać definicje dla wszystkich typów obiektów (obiekty informacji uwierzytelniających, struktury CF, kanały, listy nazw, procesy, kolejki i klasy pamięci masowej).
 - Jeśli zestaw stron zero nie jest dostępny, należy użyć definicji od ostatniego utworzenia kopii zapasowej definicji obiektów.
2. Zdefiniuj ponownie zestawy danych menedżera kolejek (nie wykonuj tego czasu, dopóki nie zostanie ukończony krok "1" na stronie 423).

Więcej informacji na ten temat zawiera sekcja [Tworzenie zestawów danych startowych i zestawów danych dziennika](#) oraz [definiowanie zestawów stron](#).

3. Zrestartuj menedżer kolejek przy użyciu nowo zdefiniowanych i zainicjowanych zestawów danych dziennika, BSDS i zestawów stron. Należy użyć instrukcji wejściowych definicji obiektu, które zostały utworzone w kroku "1" na stronie 423 jako dane wejściowe w zestawie danych wejściowych inicjowania CSQINP2.

Reinicjowanie menedżerów kolejek w grupie współużytkowania kolejek

W grupie współużytkowania kolejek reinicjowanie menedżera kolejek jest bardziej złożone. Może być konieczne ponowne zainicjowanie jednego lub większej liczby menedżerów kolejek ze względu na problemy związane z zestawem stron lub problemami z dziennikami, ale mogą wystąpić problemy z produktem Db2 lub narzędziem sprzęgającym, z którym należy się zmierzyć. W związku z tym istnieje wiele alternatyw:

zimny start

Reinicjowanie całej grupy współużytkowania kolejek obejmuje wymuszenie wszystkich struktur narzędzi CF, usunięcie wszystkich definicji obiektów dla grupy współużytkowania kolejek z programu Db2, usunięcie lub ponowne zdefiniowanie dzienników i BSDS oraz formatowanie zestawów stron dla wszystkich menedżerów kolejek w grupie współużytkowania kolejek.

Zachowane definicje współużytkowane

Usuń lub ponownie zdefiniuj dzienniki i BSDS, sformatuj zestawy stron dla wszystkich menedżerów kolejek w grupie współużytkowania kolejek i wymuszaj wszystkie struktury narzędzi CF. Po restarcie wszystkie komunikaty zostaną usunięte. Menedżery kolejek ponownie tworzą obiekty COPY, które odpowiadają obiektom GROUP, które nadal istnieją w bazie danych Db2. Wszystkie kolejki współużytkowane nadal istnieją i mogą być używane.

Ponowne zainicjowanie pojedynczego menedżera kolejek

Usuń lub ponownie zdefiniuj dzienniki i BSDS, a także sformatuj zestawy stron dla pojedynczego menedżera kolejek (usuwa wszystkie swoje prywatne obiekty i komunikaty). Po restarcie menedżer kolejek ponownie tworzy obiekty COPY, które odpowiadają obiektom GROUP, które nadal istnieją w bazie danych Db2. Wszystkie kolejki współużytkowane nadal istnieją, podobnie jak komunikaty na nich, i mogą być używane.

Odtwarzanie grupy współużytkowania kolejek w czasie

Jest to alternatywny scenariusz odtwarzania po awarii w ośrodku.

Obiekty współużytkowane są odzyskiwane do punktu w czasie osiąganym przez odtwarzanie Db2 (opisane w [awarii systemu Db2](#)). Każdy menedżer kolejek może być odzyskany do punktu w czasie, który jest możliwy do osiągnięcia z kopii zapasowych dostępnych w ośrodku alternatywnym.

Komunikaty trwałe mogą być używane w grupach współużytkowania kolejek i mogą być odtwarzane przy użyciu komendy MQSC RECOVER CFSTRUCT. Należy zauważyć, że ta komenda odtwarza dane do czasu niepowodzenia. Nie ma jednak odtwarzania nietrwałych komunikatów kolejki współużytkowanej. Są one tracone, chyba że kopie zapasowe zostały wykonane niezależnie, korzystając z funkcji COPY programu narzędziowego CSQUTIL.

Odtwarzanie każdego menedżera kolejek do tego samego punktu w czasie nie jest konieczne, ponieważ nie istnieją wzajemne zależności między obiektami lokalnymi w różnych menedżerach kolejek (które są rzeczywiście odtwarzane) i resynchronizacja menedżera kolejek z programem Db2 przy restarcie tworzy lub usuwa obiekty COPY w zależności od potrzeb w menedżerze kolejek w oparciu o menedżera kolejek.

Korzystanie z programu z/OS Automatic Restart Manager (ARM)

W tym temacie opisano sposób, w jaki można używać ARM do automatycznego restartowania menedżerów kolejek.

Ta sekcja zawiera informacje na temat następujących tematów:

- [“Co to jest ARM?” na stronie 424](#)
- [“Strategie ARM” na stronie 425](#)
- [“Używanie ARM w sieci IBM MQ” na stronie 427](#)

Co to jest ARM?

Program z/OS Automatic Restart Manager (ARM) to funkcja odtwarzania systemu z/OS, która może poprawić dostępność menedżerów kolejek. Jeśli zadanie lub zadanie nie powiedzie się lub system,

na którym jest uruchomiony, menedżer ARM może zrestartować zadanie lub zadanie bez interwencji operatora.

Jeśli menedżer kolejek lub inicjator kanału nie powiedzie się, menedżer ARM zrestartuje go na tym samym obrazie produktu z/OS . Jeśli z/OS, a więc cała grupa powiązanych podsystemów i aplikacji nie powiedzie się, ARM może automatycznie zrestartować wszystkie systemy, które nie powiodły się, w predefiniowanym porządku, na innym obrazie z/OS w syspleksie. Jest to nazywane *restartowaniem systemu międzysystemowego*.

Zrestartuj inicjator kanału przez ARM tylko w wyjątkowych okolicznościach. Jeśli menedżer kolejek jest restartowany przez menedżera ARM, zrestartuj inicjator kanału z zestawu danych inicjowania CSQINP2 (patrz [“Używanie ARM w sieci IBM MQ”](#) na stronie 427).

Za pomocą menedżera ARM można zrestartować menedżer kolejek na innym obrazie produktu z/OS w środowisku sysplex w przypadku niepowodzenia programu z/OS . Implikacje dla sieci związane z restartowaniem IBM MQ ARM na innym obrazie z/OS są opisane w [“Używanie ARM w sieci IBM MQ”](#) na stronie 427.

Aby włączyć automatyczne restartowanie:

- Ustaw zestaw danych pary ARM.
- Zdefiniuj działania automatycznego restartowania, które mają być wykonywane przez program z/OS w *strategii ARM*.
- Uruchom strategię ARM.

Ponadto program IBM MQ musi zarejestrować się w menedżerze ARM podczas uruchamiania (dzieje się to automatycznie).

Uwaga: Aby automatycznie zrestartować menedżery kolejek w różnych obrazach produktu z/OS , należy zdefiniować każdy menedżer kolejek jako podsystem w każdym obrazie produktu z/OS , w którym ten menedżer kolejek może zostać zrestartowany, z unikalną czterema unikalnymi nazwą podsystemu znaków sysplex.

Zestawy danych pary ARM

Należy upewnić się, że zdefiniowano kilka zestawów danych wymaganych dla ARM oraz że są one aktywne i aktywne przed uruchomieniem dowolnego menedżera kolejek, dla którego ma być obsługiwana obsługa ARM. IBM MQ automatyczna rejestracja ARM kończy się niepowodzeniem, jeśli para zestawów danych nie jest dostępna podczas uruchamiania menedżera kolejek. W takiej sytuacji program IBM MQ zakłada, że brak zestawu danych dla pary oznacza, że nie ma potrzeby obsługi menedżera ARM, a inicjowanie jest kontynuowane.

Informacje na temat zestawów danych pary ARM zawiera publikacja [z/OS MVS Setting up a Sysplex](#) .

Strategie ARM

Strategie automatycznego restartu menedżera są regułami zdefiniowanymi przez użytkownika, które sterują funkcjami ARM, które mogą sterować restartami menedżera kolejek.

Funkcje ARM są sterowane przez zdefiniowaną przez użytkownika *strategię ARM*. Każdy obraz z/OS , na którym uruchomiona jest instancja menedżera kolejek, która ma zostać zrestartowana przez ARM, musi być połączona z zestawem danych pary ARM z aktywną strategią ARM.

Produkt IBM udostępnia domyślną strategię ARM. Można zdefiniować nowe strategie lub przestonić wartości domyślne strategii za pomocą *programu narzędziowego danych administracyjnych* (IXCMIAPU) udostępnionego razem z produktem z/OS. *z/OS MVS Konfigurowanie syspleksu* opisuje ten program narzędziowy i zawiera szczegółowe informacje na temat definiowania strategii ARM.

[Rysunek 53](#) na stronie 426 przedstawia przykład strategii ARM. Ta przykładowa strategia restartuje menedżera kolejek w syspleksie, jeśli menedżer kolejek się nie powiódł, albo cały system nie powiódł się.

```
//IXCMIAPU EXEC PGM=IXCMIAPU,REGION=2M
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DATA TYPE(ARM)
DEFINE POLICY NAME(ARMPOL1) REPLACE(YES)
RESTART_GROUP(DEFAULT)
ELEMENT(*)
RESTART_ATTEMPTS(0) /* Jobs not to be restarted by ARM */
RESTART_GROUP(GROUP1)
ELEMENT(SYSMQMGRMQ*) /* These jobs to be restarted by ARM */
/*
```

Rysunek 53. Przykładowa strategia ARM

Więcej informacji na ten temat zawierają następujące sekcje:

- [Definiowanie strategii ARM](#)
- [Aktywowanie strategii ARM](#)
- [Rejestrowanie w menedżerze ARM](#)

Definiowanie strategii ARM

Skonfiguruj strategię ARM w następujący sposób:

- Zdefiniuj RESTART_GROUPS dla każdej instancji menedżera kolejek, która zawiera również wszystkie podsystemy CICS lub IMS, które łączą się z tą instancją menedżera kolejek. Jeśli używana jest konwencja nazewnictwa podsystemu, może być w stanie użyć znaku '?' oraz znaki '*' w nazwach elementów w celu zdefiniowania RESTART_GROUPS z minimalnym nakładem pracy definicji.
- Podaj TERMTYPE (ELEMTERM) dla inicjatorów kanału, aby wskazać, że zostaną one zrestartowane tylko wtedy, gdy inicjator kanału nie powiódł się i obraz z/OS nie uległ awarii.
- Podaj TERMTYPE (ALLTERM) dla menedżerów kolejek, aby wskazać, że zostaną zrestartowane, jeśli menedżer kolejek uległ awarii lub obraz z/OS nie powiódł się.
- Określ parametr RESTART_METHOD (BOTH, PERSIST) dla obu menedżerów kolejek i inicjatorów kanałów. Informuje on ARM o konieczności restartu za pomocą skryptu JCL, który został zapisany (po rozwiązaniu symboli systemowych) podczas ostatniego uruchomienia. Informuje on ARM, aby zrobił to niezależnie od tego, czy dany element nie powiódł się, czy obraz z/OS nie powiódł się.
- Akceptuj wartości domyślne dla wszystkich pozostałych opcji strategii ARM.

Aktywowanie strategii ARM

Aby uruchomić strategię automatycznego zarządzania restartowaniem, wydaj następującą komendę z/OS:

```
SETXCF START,POLICY,TYPE=ARM,POLNAME= mypoł
```

Po uruchomieniu strategii wszystkie systemy połączone z zestawem danych pary ARM korzystają z tej samej aktywnej strategii.

Aby wyłączyć automatyczne restarty, należy użyć komendy SETXCF STOP.

Rejestrowanie w menedżerze ARM

IBM MQ rejestruje się automatycznie jako *element ARM* podczas uruchamiania menedżera kolejek (z uwzględnieniem dostępności ARM). Deregisters jest rejestrowany w fazie zamykania, o ile nie jest to wymagane.

Po uruchomieniu menedżer kolejek określa, czy ARM jest dostępny. Jeśli jest, produkt IBM MQ rejestruje się przy użyciu nazwy SYSMQMGR *ssid*, gdzie *ssid* jest nazwą menedżera kolejek o czterech znakach, a parametr SYSMQMGR jest typem elementu.

Komendy STOP QMGR MODE (QUIESCE) i STOP QMGR MODE (FORCE) wyrejestrowują menedżer kolejek z menedżera ARM (jeśli został on zarejestrowany przy uruchamianiu menedżera ARM). Zapobiega to restartowaniu menedżera kolejek w tym menedżerze kolejek. Komenda STOP QMGR MODE (RESTART) nie wyrejestrowuje menedżera kolejek z menedżera ARM, dlatego jest ona zakwalifikowana do natychmiastowego automatycznego restartu.

Każdy obszar adresowy inicjatora kanału określa, czy ARM jest dostępny, a jeśli tak, to jest rejestrowana z nazwą elementu SYSMQCH *ssid*, gdzie *ssid* jest nazwą menedżera kolejek, a SYSMQCH jest typem elementu.

Inicjator kanału jest zawsze wyrejestrowany z ARM, gdy zatrzymuje się normalnie, i pozostaje zarejestrowany tylko wtedy, gdy kończy się nieprawidłowo. Jeśli menedżer kolejek nie powiedzie się, inicjator kanału jest zawsze wyrejestrowany.

z/OS Używanie ARM w sieci IBM MQ

Menedżer kolejek można skonfigurować w taki sposób, aby inicjatory kanału i powiązane obiekty nastuchiwania były uruchamiane automatycznie po zrestartowaniu menedżera kolejek.

Aby zapewnić w pełni automatyczne restartowanie menedżera kolejek na tym samym obrazie z/OS dla protokołów komunikacyjnych LU 6.2 i TCP/IP:

- Uruchom programy nasłuchujące automatycznie, dodając odpowiednią komendę START LISTENER do zestawu danych CSQINPX.
- Uruchom inicjator kanału automatycznie, dodając odpowiednią komendę START CHINIT do zestawu danych CSQINP2 .

Informacje o restartowaniu menedżera kolejek za pomocą protokołu TCP/IP lub LU6.2 można znaleźć w sekcji

- [“Restartowanie na innym obrazie z/OS przy użyciu protokołu TCP/IP” na stronie 427](#)
- [“Restartowanie na innym obrazie z/OS z jednostką logiczną 6.2” na stronie 429](#)

Informacje na temat zestawów danych CSQINP2 i CSQINPX znajdują się w sekcji [Czynność 13: Dostosowanie zestawów danych wejściowych inicjowania](#) .

Restartowanie na innym obrazie z/OS przy użyciu protokołu TCP/IP

Jeśli jako protokół komunikacyjny używany jest protokół TCP/IP, a użytkownik korzysta z wirtualnych adresów IP, można skonfigurować je do odtwarzania na innych obrazach produktu z/OS , umożliwiając kanały łączące się z tym menedżerem kolejek w celu ponownego nawiązania połączenia bez żadnych zmian. W przeciwnym razie można ponownie przydzielić adres TCP/IP po przeniesieniu menedżera kolejek do innego obrazu z/OS tylko wtedy, gdy używane są klastry, lub jeśli połączenie z grupą współużytkownika kolejek jest nawiązane za pomocą nazwy grupy logicznej systemu DNS (WLM).

- [Podczas korzystania z technologii klastrowej](#)
- [Podczas nawiązywania połączenia z grupą współużytkownika kolejek](#)

Podczas używania technologii klastrowej

z/OS ARM odpowiada na awarię systemu, restartując menedżer kolejek na innym obrazie z/OS w tym samym syspleksie; ten system ma inny adres TCP/IP do oryginalnego obrazu z/OS . Poniżej wyjaśniono, w jaki sposób można użyć klastrów produktu IBM MQ w celu ponownego przypisania adresu TCP/IP menedżera kolejek po jego przeniesieniu przez restart ARM do innego obrazu z/OS .

Gdy menedżer kolejek klienta wykryje awarię menedżera kolejek (jako awarię kanału), odpowie poprzez przeniesienie odpowiednich komunikatów w kolejce transmisji klastra do innego menedżera kolejek serwera, który udostępnia inną instancję docelowej kolejki klastra. Nie można jednak ponownie przydzielić komunikatów, które są powiązane z pierwotnym serwerem przez ograniczenia powinowactwa lub komunikaty, które są wątpliwe, ponieważ menedżer kolejek serwera nie powiódł się podczas przetwarzania zakończenia przetwarzania wsadowego. Aby przetworzyć te komunikaty, wykonaj następujące czynności:

1. Przydziel inną nazwę kanału odbierającego klastry i inny port TCP/IP do każdego menedżera kolejek produktu z/OS . Każdy menedżer kolejek musi mieć inny port, aby dwa systemy mogły współużytkować pojedynczy stos TCP/IP na obrazie produktu z/OS . Jednym z nich jest menedżer kolejek, który został pierwotnie uruchomiony na tym obrazie produktu z/OS , a drugi to menedżer kolejek, który zostanie zrestartowany ARM na tym obrazie produktu z/OS po wystąpieniu awarii systemu. Skonfiguruj każdy port na każdym obrazie produktu z/OS , tak aby ARM mógł zrestartować dowolny menedżer kolejek w dowolnym obrazie produktu z/OS .
2. Utwórz inny plik wejściowy komendy inicjatora kanału (CSQINPX) dla każdego menedżera kolejek i kombinacji obrazu z/OS , który ma być przywoływany podczas uruchamiania inicjatora kanału.

Każdy plik CSQINPX musi zawierać komendę START LISTENER PORT (port) specyficzną dla tego menedżera kolejek, a także komendę ALTER CHANNEL dla kanału odbiorczego klastra specyficznego dla tego menedżera kolejek i kombinacji obrazu z/OS . Komenda ALTER CHANNEL musi ustawić nazwę połączenia na nazwę TCP/IP obrazu z/OS , na którym jest restartowany. Musi ona zawierać numer portu specyficzny dla zrestartowanego menedżera kolejek jako część nazwy połączenia.

Kod JCL uruchamiania każdego menedżera kolejek może mieć stałą nazwę zestawu danych dla tego pliku CSQINPX, a każdy obraz z/OS musi mieć inną wersję każdego pliku CSQINPX na niewspółużytkowanym woluminie DASD.

Jeśli wystąpi restart ARM, program IBM MQ zareklamuje zmienioną definicję kanału do repozytorium klastra, co z kolei opublikuje je wszystkim menedżerom kolejek klienta, które wyraziły zainteresowanie menedżerem kolejek serwera.

Menedżer kolejek klienta traktuje awarię menedżera kolejek serwera jako awarię kanału i próbuje zrestartować kanał, który uległ awarii. Gdy menedżer kolejek klienta dowiaduje się o nowej nazwie połączenia z serwerem, restartowanie kanału ponownie łączy menedżera kolejek klienta z restartowanym menedżerem kolejek serwera. Menedżer kolejek klienta może następnie resynchronizować swoje komunikaty, rozwiązywać wszelkie wątpliwe komunikaty w kolejce transmisji menedżera kolejek klienta, a normalne przetwarzanie może być kontynuowane.

Podczas nawiązywania połączenia z grupą współużytkowania kolejek

Podczas nawiązywania połączenia z grupą współużytkowania kolejek za pomocą nazwy grupy logicznej systemu DNS (Dynamic Domain Name System) nazwa połączenia w definicji kanału określa nazwę grupy logicznej grupy współużytkowania kolejki, a nie nazwę hosta lub adres IP komputera fizycznego. Gdy kanał jest uruchamiany, łączy się on z dynamicznym serwerem DNS, a następnie jest połączony z jednym z menedżerów kolejek w grupie współużytkowania kolejek. Ten proces został wyjaśniony w sekcji [Konfigurowanie komunikacji dla produktu IBM MQ for z/OS przy użyciu grup współużytkowania kolejek](#).

W mało prawdopodobnym przypadku wystąpienia awarii obrazu występuje jedna z następujących sytuacji:

- Menedżery kolejek na uszkodzonym obrazie de-register z dynamicznego serwera DNS działającego w systemie sysplex. Kanał odpowiada na niepowodzenie połączenia przez wprowadzenie stanu RETRYING, a następnie łączy się z dynamicznym systemem DNS działającym na syspleksie. Dynamiczny serwer DNS przydziela żądanie przychodzące do jednego z pozostałych elementów grupy współużytkowania kolejek, która nadal działa na pozostałych obrazach.
- Jeśli żaden inny menedżer kolejek w grupie współużytkowania kolejek nie jest aktywny, a menedżer ARM restartuje menedżer kolejek i inicjator kanału na innym obrazie, program nasłuchujący grupy

zarejestruje się z dynamicznym serwerem DNS z tego nowego obrazu. Oznacza to, że nazwa grupy logicznej (z pola nazwy połączenia kanału) łączy się z dynamicznym serwerem DNS, a następnie jest połączona z tym samym menedżerem kolejek, a teraz działa na innym obrazie. Definicja kanału nie była wymagana do zmiany.

W odniesieniu do tego typu odzysku należy odnotować następujące punkty:

- W systemie z/OS dynamiczny serwer DNS jest uruchamiany na jednym z obrazów z/OS w syspleksie. Jeśli ten obraz nie powiedzie się, dynamiczny serwer DNS musi zostać skonfigurowany tak, aby w syspleksie działał dodatkowy serwer nazw, działający jako alternatywa dla podstawowego serwera nazw. Informacje na temat podstawowych i dodatkowych dynamicznych serwerów DNS można znaleźć w podręczniku *OS/390 SecureWay CS IP Configuration*.
- Program nasłuchujący grupy TCP/IP mógł zostać uruchomiony na określonym adresie IP, który może nie być dostępny w tym obrazie produktu z/OS. W takim przypadku może być konieczne uruchomienie nasłuchiwanie na innym adresie IP na nowym obrazie. Jeśli używane są wirtualne adresy IP, można skonfigurować je do odtwarzania na innych obrazach z/OS, tak aby nie były wymagane żadne zmiany komendy START LISTENER.

Restartowanie na innym obrazie z/OS z jednostką logiczną 6.2

Jeśli używane są tylko protokoły komunikacyjne LU 6.2, wykonaj następującą procedurę, aby włączyć ponowne połączenie sieciowe po automatycznym zrestartowaniu menedżera kolejek na innym obrazie z/OS w obrębie syspleksu:

- Zdefiniuj każdego menedżera kolejek w syspleksie z unikalną nazwą podsystemu.
- Zdefiniuj każdy inicjator kanału w syspleksie z unikalną nazwą LUNAME. Wartość ta jest określona zarówno w atrybutach menedżera kolejek, jak i w komendzie START LISTENER.

Uwaga: Nazwa LUNAME jest nazwą pozycji w tabeli po stronie APPC, która z kolei odwzorowuje tę wartość na rzeczywistą wartość LUNAME.

- Skonfiguruj współużytkowaną tabelę po stronie APPC, która jest przywoływana przez każdy obraz z/OS w syspleksie. Powinna ona zawierać pozycję dla każdego obiektu LUNAME inicjatora kanału. Więcej informacji na ten temat zawiera sekcja *z/OS MVS Planowanie: zarządzanie APPC/MVS*.
- Skonfiguruj element APPCPM xx systemu SYS1.PARMLIB dla każdego inicjatora kanału w syspleksie, który ma zawierać LUADD, aby aktywować pozycję tabeli po stronie APPC dla tego inicjatora kanału. Elementy te powinny być współużytkowane przez każdy obraz z/OS. Odpowiedni SYS1.PARMLIB jest aktywowany przez komendę z/OS SET APPC= xx, która jest automatycznie uruchamiana podczas restartu menedżera ARM menedżera kolejek (i jego inicjatora kanału) na innym obrazie produktu z/OS, zgodnie z opisem w poniższym tekście.
- Użyj atrybutu menedżera kolejek LU62ARM, aby określić przyrostek xx dla tego SYS1.PARMLIB dla każdego inicjatora kanału. Powoduje to, że inicjator kanału musi wydać wymaganą komendę z/OS SET APPC= xx, aby aktywować jej nazwę LUNAME.

Zdefiniuj strategię ARM w taki sposób, aby restartuje inicjator kanału tylko wtedy, gdy nie powiedzie się, podczas gdy jego obraz z/OS pozostaje w górę. ID użytkownika powiązany z przestrzenią adresową XCFAS musi być autoryzowany do wydania komendy IBM MQ START CHINIT. Nie należy restartować inicjatora kanału automatycznie, jeśli jego obraz z/OS również nie powiedzie się, zamiast tego użyj komend w zestawach danych CSQINP2 i CSQINPX, aby uruchomić inicjator kanału i obiekty nasłuchiwanie.

Ręczne odzyskiwanie jednostek pracy

Jednostki pracy CICS, IMS, RRS lub inne menedżery kolejek można odzyskać ręcznie w grupie współużytkowania kolejek. Za pomocą komend menedżera kolejek można wyświetlić status jednostek pracy powiązanych z każdym połączeniem z menedżerem kolejek.

Ten temat zawiera informacje na temat następujących tematów:

- [“Wyświetlanie połączeń i wątków” na stronie 430](#)

- [“Ręczne odtwarzanie CICS jednostek odzyskiwania” na stronie 430](#)
- [“Ręczne odtwarzanie IMS jednostek odzyskiwania” na stronie 434](#)
- [“Ręczne odzyskiwanie jednostek RRS odzyskiwania” na stronie 436](#)
- [“Odzyskiwanie jednostek odtwarzania w innym menedżerze kolejek w grupie współużytkowania kolejek” na stronie 436](#)

Wyświetlanie połączeń i wątków

Za pomocą komendy `DISPLAY CONN` można uzyskać informacje na temat połączeń z menedżerami kolejek i powiązanych z nimi jednostkami pracy. Użytkownik może wyświetlić aktywne jednostki pracy, aby zobaczyć, co się obecnie dzieje, lub sprawdzić, co należy zakończyć, aby umożliwić menedżerowi kolejek zamknięcie systemu, a także wyświetlić nierozstrzygnięte jednostki pracy, które mają pomóc w odzyskaniu.

Aktywne jednostki pracy

Aby wyświetlić tylko aktywne jednostki pracy, należy użyć

```
DISPLAY CONN(*) WHERE(UOWSTATE EQ ACTIVE)
```

Nierozstrzygnięte jednostki pracy

Nierozstrzygnięta jednostka pracy, znana również jako wątek wątpliwy, jest w drugiej części operacji zatwierdzania dwufazowego. Zasoby są przechowywane w IBM MQ w jego imieniu. Aby wyświetlić nierozstrzygnięte jednostki pracy, użyj

```
DISPLAY CONN(*) WHERE(UOWSTATE EQ UNRESOLVED)
```

Konieczna jest interwencja zewnętrzna w celu rozwiązania statusu nierozstrzygniętych jednostek pracy. Może to dotyczyć tylko uruchamiania koordynatora odtwarzania (CICS, IMS lub RRS) lub może wymagać więcej, zgodnie z opisem w poniższych sekcjach.

Ręczne odtwarzanie CICS jednostek odzyskiwania

W tym temacie opisano, co się dzieje, gdy adapter CICS zostanie zrestartowany, a następnie wyjaśnia sposób postępowania z dowolnymi nierozstrzygniętą jednostką odtwarzania.

Co się dzieje, gdy adapter CICS zostanie zrestartowany

Za każdym razem, gdy połączenie jest zerwane, adapter musi przejść przez *fazę restartu* podczas *ponownego łączenia procesu*. Faza restartowania resynchronizuje zasoby. Resynchronizacja między CICS a IBM MQ umożliwia identyfikowanie i rozstrzygnięcie w wątpliwych jednostkach pracy.

Resynchronizacja może być spowodowana przez:

- Jawne żądanie z rozproszonego komponentu kolejkowania
- Żądanie niejawne, gdy połączenie jest nawiązane z IBM MQ

Jeśli resynchronizacja jest spowodowana połączeniem z serwerem IBM MQ, sekwencja zdarzeń jest następująca:

1. Proces połączenia pobiera listę identyfikatorów jednostek pracy (UOW) z listy wątpliwych jednostek pracy z produktu IBM MQ.
2. Identyfikatory UOW są wyświetlane na konsoli w komunikatach CSQC313I .
3. Identyfikatory UOW są przekazywane do produktu CICS.
4. CICS inicjuje zadanie resynchronizacji (CRSY) dla każdej wątpliwej identyfikatora UOW.

5. Wynik zadania dla każdej wątpliwej jednostki UOW jest wyświetlany na konsoli.

Należy sprawdzić komunikaty wyświetlane w procesie połączenia:

CSQC313I

Wskazuje, że UOW jest wątpliwy.

CSQC400I

Identyfikuje jednostkę pracy, po której następuje jeden z następujących komunikatów:

- CSQC402I lub CSQC403I pokazuje, że jednostka pracy została pomyślnie rozstrzygnięta (zatwierdzona lub wycofana).
- Wartości CSQC404E, CSQC405E, CSQC406E lub CSQC407E wskazują, że jednostka pracy nie została rozstrzygnięta.

CSQC409I

Wskazuje, że wszystkie jednostki UOWs zostały rozstrzygnięte pomyślnie.

CSQC408I

Wskazuje, że nie wszystkie jednostki UOWs zostały rozstrzygnięte pomyślnie.

CSQC314I

Ostrzega, że identyfikatory UOW podświetlone znakiem * nie są rozstrzygane automatycznie. Te jednostki UOWs muszą zostać jawnie rozstrzygnięte przez rozproszony komponent kolejkowania po jego zrestartowaniu.

Rysunek 54 na stronie 431 przedstawia przykładowy zestaw komunikatów o restarcie wyświetlanych w konsoli z/OS .

```
CSQ9022I +CSQ1 CSQYASCP ' START QMGR' NORMAL COMPLETION
+CSQC323I VICIC1 CSQCQCON CONNECT received from TERMID=PB62 TRANID=CKCN
+CSQC303I VICIC1 CSQCCON CSQCSERV loaded. Entry point is 850E8918
+CSQC313I VICIC1 CSQCCON UOWID=VICIC1.A6E5A6F0E2178D25 is in doubt
+CSQC313I VICIC1 CSQCCON UOWID=VICIC1.A6E5A6F055B2AC25 is in doubt
+CSQC313I VICIC1 CSQCCON UOWID=VICIC1.A6E5A6EFFF60D425 is in doubt
+CSQC313I VICIC1 CSQCCON UOWID=VICIC1.A6E5A6F07AB56D22 is in doubt
+CSQC307I VICIC1 CSQCCON Successful connection to subsystem VC2
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008BAD18) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008BAA10) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008BA708) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CAE88) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CAB80) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CA878) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CA570) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CA268) connect
successful
+CSQC403I VICIC1 CSQCTRUE Resolved BACKOUT for
+CSQC400I VICIC1 CSQCTRUE UOWID=VICIC1.A6E5A6F0E2178D25
+CSQC403I VICIC1 CSQCTRUE Resolved BACKOUT for
+CSQC400I VICIC1 CSQCTRUE UOWID=VICIC1.A6E5A6F055B2AC25
+CSQC403I VICIC1 CSQCTRUE Resolved BACKOUT for
+CSQC400I VICIC1 CSQCTRUE UOWID=VICIC1.A6E5A6F07AB56D22
+CSQC403I VICIC1 CSQCTRUE Resolved BACKOUT for
+CSQC400I VICIC1 CSQCTRUE UOWID=VICIC1.A6E5A6EFFF60D425
+CSQC409I VICIC1 CSQCTRUE Resynchronization completed successfully
```

Rysunek 54. Przykładowe komunikaty o restarcie

Łączna liczba komunikatów CSQC313I powinna być równa łącznej liczbie komunikatów CSQC402I plus CSQC403I . Jeśli sumy nie są równe, istnieją jednostki UOWs, których proces połączenia nie może rozwiązać. Te jednostki UOWs, które nie mogą zostać rozwiązane, są spowodowane problemami z produktem CICS (na przykład zimnym startem) lub z produktem IBM MQ, albo dystrybucją

kolejkowania. Po rozwiązaniu tych problemów można zainicjować kolejną resynchronizację, rozłączając się, a następnie ponownie nawiązując połączenie.

Alternatywnie można rozwiązać każdą znakomitą jednostkę pracy samodzielnie, używając komendy RESOLVE INDOUBT i identyfikatora jednostki pracy wyświetlonej w komunikacie CSQC400I. Następnie należy zainicjować rozłączenie i połączenie w celu wyczyszczenia *jednostki deskryptorów odtwarzania* w produkcie CICS. Aby ręcznie rozwiązać UOWs, należy znać poprawny wynik pracy jednostki pracy.

Wszystkie komunikaty, które są powiązane z nierozstrzygniętymi UOWami, są zablokowane przez produkt IBM MQ i nie mają do nich dostępu żadne zadanie wsadowe, TSO ani zadanie CICS .

Jeśli CICS nie powiedzie się, a wymagany jest restart awaryjny, *nie* udostępni wartości GENERIC APPLID w systemie CICS . Jeśli zostanie nawiązane połączenie z serwerem IBM MQ, nie będzie można zagwarantować integralności danych z produktem IBM MQ . Wynika to z faktu, że program IBM MQ traktuje nową instancję produktu CICS jako inną CICS (ponieważ APPLID jest inna). Rozstrzygnięcie wątpliwe jest następnie oparte na niewłaściwym dzienniku produktu CICS .

Sposób ręcznego rozwiązywania jednostek odzyskiwania CICS

Jeśli adapter zakończy działanie nieprawidłowo, CICS i IBM MQ będą tworzyć listy wątpliwe, albo dynamicznie, albo podczas restartu, w zależności od tego, który podsystem spowodował zakończenie.

Uwaga: If you use the DFH\$INDB sample program to show units of work, you might find that it does not always show IBM MQ UOWs correctly.

Gdy program CICS łączy się z produktem IBM MQ, może istnieć co najmniej jedna jednostka odzyskiwania, która nie została rozstrzygnięta.

Do konsoli zostanie wysłany jeden z następujących komunikatów:

- CSQC404E
- CSQC405E
- CSQC406E
- CSQC407E
- CSQC408I

Szczegółowe informacje na temat tych komunikatów można znaleźć w komunikatach [Komunikaty adaptera i mostu CICS](#) .

Program CICS zachowuje szczegółowe informacje o jednostkach odzyskiwania, które nie zostały rozstrzygnięte podczas uruchamiania połączenia. Pozycja jest czyszczona, gdy nie jest już wyświetlana na liście prezentowanej przez produkt IBM MQ.

Any units of recovery that CICS cannot resolve must be resolved manually using IBM MQ commands. Ta procedura ręczna jest rzadko używana w instalacji, ponieważ jest ona wymagana tylko w przypadku, gdy błędy operacyjne lub problemy z oprogramowaniem uniemożliwiły automatyczne rozstrzygnięcie. *Należy zbadać wszelkie niespójności znalezione podczas rozstrzygnięcia wątpliwych wątpliwości.*

Aby rozwiązać jednostki odzyskiwania:

1. Uzyskaj listę jednostek odtwarzania z produktu IBM MQ , korzystając z następującej komendy:

```
+CSQ1 DISPLAY CONN( * ) WHERE(UOWSTATE EQ UNRESOLVED)
```

Zostanie wyświetlony następujący komunikat:


```

CSQM201I +CSQ1 CSQMDRTC DISPLAY CONN DETAILS
CONN (BC85772CBE3E0001)
EXTCONN (C3E2D8C3C7D9F0F9404040404040)
TYPE (CONN)
CONNOPTS (
MQCNO_STANDARD_BINDING
)
UOWLOGDA (2005-02-04)
UOWLOGTI (10.17.44)
UOWSTDA (2005-02-04)
UOWSTTI (10.17.44)
UOWSTATE (UNRESOLVED)
NID (IYRCSQ1 .BC8571519B60222D)
EXTURID (BC8571519B60222D)
QMURID (0000002BDA50)
URTYPE (CICS)
USERID (MQTEST)
APPLTAG (IYRCSQ1)
ASID (0000)
APPLTYPE (CICS)
TRANSID (GP02)
TASKNO (0000096)
END CONN DETAILS

```

W przypadku połączeń produktu CICS wartość NID składa się z identyfikatora applid produktu CICS i unikalnego numeru udostępnionego przez produkt CICS w momencie zapisywania pozycji dziennika punktu synchronizacji. Ten unikalny numer jest zapisywany w rekordach zapisanych zarówno w dzienniku systemowym CICS, jak i w dzienniku IBM MQ w czasie przetwarzania w punkcie synchronizacji. Ta wartość jest określana w produkcie CICS jako *znacznik odtwarzania*.

2. Zeskanuj dziennik CICS pod kątem pozycji związanych z określoną jednostką odzyskiwania.

Poszukaj rekordu PREPARE dla instalacji związanej z zadaniem, w której pole tokenu odtwarzania (JCSRMTKN) jest równe wartości uzyskanej z identyfikatora sieci. Identyfikator sieci jest dostarczany przez produkt IBM MQ w danych wyjściowych komendy DISPLAY CONN.

Rekord PREPARE w dzienniku CICS dla jednostek odzyskiwania udostępnia numer zadania CICS. Wszystkie pozostałe wpisy w dzienniku dla tego zadania CICS mogą być umieszczone przy użyciu tej liczby.

Podczas skanowania dziennika można użyć programu CICS do drukowania dziennika DFHJUP. Szczegółowe informacje na temat korzystania z tego programu można znaleźć w publikacji *CICS Operations and Utilities Guide*.

3. Zeskanuj dziennik IBM MQ pod kątem rekordów z NID związaną z określoną jednostką odzyskiwania. Następnie należy użyć identyfikatora URID z tego rekordu, aby uzyskać pozostałe rekordy dziennika dla tej jednostki odzyskiwania.

Podczas skanowania dziennika produktu IBM MQ należy pamiętać, że komunikat o uruchomieniu CSQJ001I programu IBM MQ udostępnia uruchomienie RBA dla tej sesji.

Do tego celu można użyć programu do obsługi rekordów protokołu drukowania (CSQ1LOGP).

4. W razie potrzeby należy wykonać wątpliwe rozwiązanie w produkcie IBM MQ.

IBM MQ można nakazywać, aby wykonać działanie odzyskiwania dla jednostki odzyskiwania przy użyciu komendy IBM MQ [RESOLVE INDOUBT](#).

Aby odtworzyć wszystkie wątki powiązane z konkretną *nazwą połączenia*, należy użyć opcji NID (*).

Komenda generuje jeden z następujących komunikatów pokazujący, czy wątek został zatwierdzony, czy wycofany:

```

CSQV414I +CSQ1 THREAD network-id COMMIT SCHEDULED
CSQV415I +CSQ1 THREAD network-id ABORT SCHEDULED

```

When performing in-doubt resolution, CICS and the adapter are not aware of the commands to IBM MQ to commit or back out units of recovery, because only IBM MQ resources are affected. Jednak program CICS przechowuje szczegółowe informacje o wątkach wątpliwych, które nie mogły zostać rozstrzygnięte przez produkt IBM MQ. Te informacje są usuwane albo gdy prezentowana lista jest pusta, albo gdy lista nie zawiera jednostki odzyskiwania, w której CICS ma szczegóły.

Ręczne odtwarzanie IMS jednostek odzyskiwania

W tym temacie opisano, co się dzieje, gdy adapter IMS zostanie zrestartowany, a następnie wyjaśnia sposób postępowania z dowolnymi nierozstrzygniętą jednostką odtwarzania.

Co się dzieje, gdy adapter IMS zostanie zrestartowany

Po zrestartowaniu połączenia z serwerem IBM MQ , po zrestartowaniu menedżera kolejek lub komendzie IMS /START SUBSYS, program IMS inicjuje następujący proces resynchronizacji:

1. IMS przedstawia listę identyfikatorów jednostek pracy (UOW), które są w wątpliwość co najmniej raz w przypadku adaptera IBM MQ IMS z parametrem rozstrzygnięcia zatwierdzenia lub wycofania.
2. Adapter IMS przekazuje żądanie rozwiązania do produktu IBM MQ i raportuje wynik z powrotem do produktu IMS.
3. Po przetworzeniu wszystkich żądań rozwiązania IMS adapter IMS pobiera z listy IBM MQ listę wszystkich jednostek UOWs, które IBM MQ nadal mają wątpliwości, które zostały zainicjowane przez system IMS . Są one zgłaszane do głównego terminalu IMS w komunikacie CSQQ008I.

Uwaga: Chociaż jednostka pracy jest wątpliwa, wszystkie powiązane komunikaty produktu IBM MQ są zablokowane przez produkt IBM MQ i nie są dostępne dla żadnej aplikacji.

Sposób ręcznego rozwiązywania jednostek odzyskiwania IMS

Gdy program IMS łączy się z serwerem IBM MQ, produkt IBM MQ może mieć jedną lub większą liczbę wątpliwych jednostek odzyskiwania, które nie zostały rozstrzygnięte.

Jeśli IBM MQ ma wątpliwe jednostki odzyskiwania, które nie zostały przetłumaczone przez produkt IMS , na głównym terminalu IMS zostanie wygenerowany następujący komunikat:

```
CSQQ008I nn units of recovery are still in doubt in queue manager qmgr-name
```

Jeśli ten komunikat został wygenerowany, program IMS był zimny lub został uruchomiony z niekompletną taśmą dzienną. Ten komunikat może być również generowany, jeśli program IBM MQ lub IMS zakończy działanie w sposób nieprawidłowy z powodu błędu oprogramowania lub innej awarii podsystemu.

Po odebraniu komunikatu CSQQ008I :

- Połączenie pozostaje aktywne.
- Aplikacje produktu IMS mogą nadal uzyskiwać dostęp do zasobów produktu IBM MQ .
- Niektóre zasoby produktu IBM MQ pozostają zablokowane.

Jeśli wątek wątpliwy nie zostanie rozwiązany, kolejki komunikatów programu IMS mogą zostać rozpoczęte. Jeśli kolejki produktu IMS zapełnią pojemność, program IMS kończy działanie. Należy pamiętać o tym potencjalnym trudnościach i należy monitorować produkt IMS do czasu, aż wątpliwe jednostki odtwarzania zostaną w pełni rozstrzygnięte.

Procedura odzyskiwania

Aby odtworzyć jednostki pracy produktu IMS , wykonaj następującą procedurę:

1. Wymuś zamknięcie dziennika IMS za pomocą komendy /SWI OLDS, a następnie zarchiwizowanie dziennika produktu IMS . Użyj programu narzędziowego DFSERA10, aby wydrukować rekordy

z poprzedniej taśmy protokołu IMS . Rekordy dziennika typu X ' 3730 ' wskazują żądanie zatwierdzenia phase-2 , a w rekordach dziennika typu X ' 38 ' -żądanie przerwania. Zanotuj żądane działanie dla ostatniej transakcji w każdym regionie zależnym.

2. Uruchom zadanie wsadowe DL/I, aby wycofać wszystkie zaangażowane PSB, które nie osiągnęły punktu zatwierdzenia. Proces może zająć trochę czasu, ponieważ transakcje są nadal przetwarzane. Może również zablokować pewną liczbę rekordów, co może wpłynąć na pozostałą część przetwarzania i pozostałe kolejki komunikatów.
3. Aby utworzyć listę niepewnych jednostek odtwarzania z produktu IBM MQ , należy użyć następującej komendy:

```
+CSQ1 DISPLAY CONN(*) WHERE(UOWSTATE EQ UNRESOLVED)
```

Zostanie wyświetlony następujący komunikat:

```
CSQM201I +CSQ1 CSQMDRTC DISPLAY CONN DETAILS
CONN(BC45A794C4290001)
EXTCONN(C3E2D8C3E2C5C3F240404040404040)
TYPE(CONN)
CONNOPTS(
MQCNO_STANDARD_BINDING
)
UOWLOGDA(2005-02-15)
UOWLOGTI(16.39.43)
UOWSTDA(2005-02-15)
UOWSTTI(16.39.43)
UOWSTATE(UNRESOLVED)
NID(IM8F .BC45A794D3810344)
EXTURID(
0000052900000000
)
QMURID(00000354B76E)
URTYPE(IMS)
USERID(STCPI)
APPLTAG(IM8F)
ASID(0000)
APPLTYPE(IMS)
PSTID(0004)
PSBNAME(GP01MPP)
```

W przypadku systemu IMS wartość NID składa się z nazwy połączenia IMS i unikalnego numeru udostępnionego przez produkt IMS. Wartość ta jest określana w produkcie IMS jako *znacznik odtwarzania*. Aby uzyskać więcej informacji, zapoznaj się z *Podręcznikiem dostosowywania produktu IMS*.

4. Porównaj wartości NID (IMSID plus OASN w postaci szesnastkowej) wyświetlone w komunikatach DISPLAY THREAD z wartością OASNs (4 bajty dziesiętne) wyświetlonym w danych wyjściowych DFSERA10 . Zdecyduj, czy zatwierdzić, czy wycofać zmiany.
5. Wykonaj wątpliwe rozwiązanie w programie IBM MQ za pomocą komendy RESOLVE INDOUBT , w następujący sposób:

```
RESOLVE INDOUBT( connection-name )
ACTION(COMMIT|BACKOUT)
NID( network-id )
```

Aby odtworzyć wszystkie wątki powiązane z *nazwą połączenia*, należy użyć opcji NID (*). Komenda powoduje wykonanie jednego z następujących komunikatów w celu wskazania, czy wątek został zatwierdzony, czy wycofany:

```
CSQV414I THREAD network-id COMMIT SCHEDULED
CSQV415I THREAD network-id BACKOUT SCHEDULED
```

When performing in-doubt resolution, IMS and the adapter are not aware of the commands to IBM MQ to commit or back out in-doubt units of recovery because only IBM MQ resources are affected.

z/OS Ręczne odzyskiwanie jednostek RRS odzyskiwania

W tym temacie opisano sposób określania, czy istnieją wątpliwe jednostki odzyskiwania RRS, a także sposób ręcznego rozwiązywania tych jednostek odzyskiwania.

Gdy produkt RRS łączy się z serwerem IBM MQ, produkt IBM MQ może mieć jedną lub więcej wątpliwych jednostek odtwarzania, które nie zostały rozstrzygnięte. Jeśli w produkcie IBM MQ istnieją wątpliwe jednostki odtwarzania, które nie zostały rozstrzygane przez usługę RRS, w konsoli produktu z/OS jest generowany jeden z następujących komunikatów:

- CSQ3011I
- CSQ3013I
- CSQ3014I
- CSQ3016I

Zarówno produkt IBM MQ, jak i usługa RRS udostępniają narzędzia do wyświetlania informacji o wątpliwych jednostkach odzyskiwania i technikach ich ręcznego rozwiązywania.

W programie IBM MQ należy użyć komendy DISPLAY CONN, aby wyświetlić informacje o wątkach wątpliwych IBM MQ. Dane wyjściowe komendy obejmują jednostkę RRS identyfikatorów odzyskiwania dla tych wątków programu IBM MQ, które mają RRS jako koordynator. Można to wykorzystać do określenia wyniku jednostki odzyskiwania.

Użyj komendy RESOLVE INDOUBT, aby ręcznie rozstrzygać wątek wątpliwy IBM MQ. Ta komenda może zostać użyta do zatwierdzenia lub usunięcia jednostki odzyskiwania po określeniu, jaka jest poprawna decyzja.

z/OS Odzyskiwanie jednostek odtwarzania w innym menedźerze kolejek w grupie współużytkowania kolejek

W tym temacie opisano sposób identyfikowania i ręcznego odtwarzania jednostek odtwarzania w innych menedżerach kolejek w grupie współużytkowania kolejek.

Jeśli menedżer kolejek, który jest elementem grupy współużytkowania kolejek, nie powiedzie się i nie może zostać zrestartowany, inne menedżery kolejek w grupie mogą wykonać odtwarzanie równorzędne i przejąć od niego działanie. Jednak menedżer kolejek może mieć wątpliwe jednostki odzyskiwania, które nie mogą zostać rozstrzygnięte przez odtwarzanie równorzędne, ponieważ końcowa dyspozycja tej jednostki odzyskiwania jest znana tylko dla menedżera kolejek zakończonych niepowodzeniem. Te jednostki odzyskiwania są rozstrzygane, gdy menedżer kolejek zostanie ostatecznie zrestartowany, ale do tego czasu pozostają one wątpliwe.

Oznacza to, że niektóre zasoby (na przykład komunikaty) mogą być zablokowane, co powoduje, że są one niedostępne dla innych menedżerów kolejek w grupie. W takiej sytuacji można użyć komendy DISPLAY THREAD w celu wyświetlenia tych jednostek pracy w nieaktywnym menedźerze kolejek. Jeśli te jednostki odzyskiwania mają zostać rozstrzygane ręcznie w celu udostępnienia komunikatów innym menedżerom kolejek w grupie, można użyć komendy RESOLVE INDOUBT.

Po wywołaniu komendy DISPLAY THREAD w celu wyświetlenia jednostek odzyskiwania, które są wątpliwe, można użyć słowa kluczowego QMNAME, aby określić nazwę nieaktywnego menedżera kolejek. Na przykład, jeśli wydajesz następującą komendę:

```
+CSQ1 DISPLAY THREAD(*) TYPE(INDOUBT) QMNAME(QM01)
```

Użytkownik otrzymuje następujące komunikaty:

```
CSQV436I +CSQ1 INDOUBT THREADS FOR QM01 -
NAME  THREAD-XREF  URID  NID
USER1  00000000000000000000000000000000 CSQ:0001.0
USER2  00000000000000000000000000000000 CSQ:0002.0
DISPLAY THREAD REPORT COMPLETE
```

Jeśli określony menedżer kolejek jest aktywny, program IBM MQ nie zwraca informacji o wątkach wątpliwych, ale wydaje następujący komunikat:

```
CSQV435I CANNOT USE QMNAME KEYWORD, QM01 IS ACTIVE
```

Użyj komendy IBM MQ RESOLVE INDOUBT, aby ręcznie rozstrzygnąć wątki wątpliwe. Użyj słowa kluczowego QMNAME, aby określić nazwę nieaktywnego menedżera kolejek w komendzie.

Ta komenda może zostać użyta do zatwierdzenia lub odzyskania jednostki odzyskiwania. Komenda rozwiązuje tylko współużytkowaną część jednostki odzyskiwania; wszystkie komunikaty lokalne nie mają wpływu na działanie i pozostają zablokowane do momentu zrestartowania menedżera kolejek lub ponownego nawiązania połączenia z CICS, IMS lub RRS.

IBM MQ and IMS

Produkt IBM MQ udostępnia dwa komponenty do współpracy z produktem IMS, adapterem IBM MQ - IMS oraz mostem IBM MQ - IMS. Te komponenty są powszechnie nazywane adapterem IMS i mostem IMS.

Działanie adaptera IMS

W tym temacie opisano sposób działania adaptera IMS, który łączy produkt IBM MQ z systemami IMS.

Uwaga: Adapter IMS nie uwzględnia żadnych operacji i paneli sterujących.

Ten temat zawiera następujące sekcje:

- [“Sterowanie połączeniami produktu IMS” na stronie 437](#)
- [“Nawiązanie połączenia z regionu sterującego IMS” na stronie 438](#)
- [“Wyświetlanie w wątpliwych jednostkach odzyskiwania” na stronie 440](#)
- [“Sterowanie połączeniami regionu zależnego IMS” na stronie 442](#)
- [“odłączanie odIMS” na stronie 444](#)
- [“Sterowanie monitorem wyzwalacza IMS” na stronie 444](#)

Sterowanie połączeniami produktu IMS

W tym temacie opisano komendy operatora IMS, które sterują i monitorują połączenie z programem IBM MQ.

Produkt IMS udostępnia następujące komendy operatora do sterowania i monitorowania połączenia z programem IBM MQ:

/CHANGE SUBSYS

Usuwa wątpliwe jednostki odzyskiwania z programu IMS.

/DISPLAY OASN SUBSYS

Wyświetla oczekujące elementy odtwarzania.

/DISPLAY SUBSYS

Wyświetla status połączenia i działanie wątku.

/START SUBSYS

Łączy region sterujący IMS z menedżerem kolejek.

/STOP SUBSYS

Rozłącza produkt IMS z menedżera kolejek.

/TRACE

Steruje śledzeniem IMS .

Więcej informacji na temat tych komend znajduje się w podręczniku *IMS/ESA Operator's Reference* dla używanego poziomu produktu IMS .

Odpowiedzi komend IMS są wysyłane do terminalu, z którego została wydana komenda. Autoryzacja do wydawania komend IMS jest oparta na zabezpieczeniach produktu IMS .

z/OS Nawiązanie połączenia z regionu sterującego IMS

Ten temat zawiera informacje na temat mechanizmów dostępnych do łączenia się z produktem IMS z produktem IBM MQ.

Produkt IMS tworzy jedno połączenie z regionu sterującego do każdego menedżera kolejek, który używa produktu IMS. Aby można było nawiązać połączenie w jeden z następujących sposobów, należy włączyć opcję IMS :

- Automatycznie podczas:
 - Inicjowanie zimnego startu.
 - Ciepłe uruchomienie produktu IMS, jeśli połączenie IBM MQ było aktywne, gdy program IMS został wyłączony.
- W odpowiedzi na komendę IMS :

```
/START SUBSYS sysid
```

gdzie *sysid* jest nazwą menedżera kolejek.

Komenda może zostać wydana niezależnie od tego, czy menedżer kolejek jest aktywny.

Połączenie nie zostanie nawiązane, dopóki nie zostanie wykonane pierwsze wywołanie funkcji API MQ do menedżera kolejek. Do tego czasu komenda IMS /DIS SUBSYS wyświetla status 'NOT CONN'.

Kolejność uruchamiania produktu IMS i menedżera kolejek nie jest znacząca.

Produkt IMS nie może ponownie włączyć połączenia z menedżerem kolejek automatycznie, jeśli menedżer kolejek został zatrzymany za pomocą komendy STOP QMGR, komendy IMS /STOP SUBSYS lub nieprawidłowego zakończenia. W związku z tym należy nawiązać połączenie za pomocą komendy IMS /START SUBSYS.

Inicjowanie adaptera i nawiąże połączenie z menedżerem kolejek

Adapter jest zestawem modułów załadowanych do obszaru sterującego IMS i regionów zależnych za pomocą zewnętrznego narzędzia dołączania podsystemu IMS .

Ta procedura inicjuje adapter i łączy się z menedżerem kolejek:

1. Odczytaj element podsystemu (SSM) z produktu IMS.PROCLIB. Wybrany SSM jest parametrem EXEC IMS . Istnieje jedna pozycja w elemencie dla każdego menedżera kolejek, z którym program IMS może się połączyć. Każda pozycja zawiera informacje sterujące dotyczące adaptera IBM MQ .
2. Załaduj adapter IMS .

Uwaga: Program IMS ładuje jedną kopię modułów adaptera dla każdej instancji produktu IBM MQ , która jest zdefiniowana w elemencie SSM.

3. Dołącz zadanie zewnętrznego podsystemu dla IBM MQ.

4. Uruchom adapter z parametrem CTL EXEC (IMSID) jako nazwą połączenia.

Proces jest taki sam, niezależnie od tego, czy połączenie jest częścią inicjowania, czy wynikiem komendy / START SUBSYS komendy IMS .

Jeśli menedżer kolejek jest aktywny, gdy program IMS próbuje nawiązać połączenie, wysyłane są następujące komunikaty:

- do konsoli z/OS :

```
DFS3613I ESS TCB INITIALIZATION COMPLETE
```

- do głównego terminala IMS :

```
CSQQ000I IMS/TM imsid connected to queue manager ssnm
```

Gdy program IMS próbuje nawiązać połączenie, a *menedżer kolejek nie jest aktywny*, następujące komunikaty są wysyłane do głównego terminala IMS za każdym razem, gdy aplikacja wykonuje wywołanie MQI:

```
CSQQ001I IMS/TM imsid not connected to queue manager ssnm.  
Notify message accepted  
DFS3607I MQM1 SUBSYSTEM ID EXIT FAILURE, FC = 0286, RC = 08,  
JOBNAME = IMSEMPR1
```

Jeśli komunikaty DFS3607I zostaną wyświetlone po uruchomieniu połączenia z produktem IMS lub podczas uruchamiania systemu, oznacza to, że menedżer kolejek nie jest dostępny. Aby zapobiec generowaniu dużej liczby komunikatów, należy wykonać jedną z następujących czynności:

1. Uruchom odpowiedni menedżer kolejek.
2. Wydadź komendę IMS :

```
/STOP SUBSYS
```

Oznacza to, że program IMS nie będzie się spodziewać połączenia z menedżerem kolejek.

Jeśli nie zostanie to wykonane, komunikat DFS3607I i powiązany komunikat CSQQ001I są wysyłane za każdym razem, gdy zadanie jest zaplanowane w regionie i za każdym razem, gdy żądanie połączenia z menedżerem kolejek jest wykonywane przez aplikację.

Załącznik do wątku

W regionie MPP lub IFP program IMS tworzy połączenie z wątkiem, gdy pierwszy program użytkowy jest zaplanowany w tym regionie, nawet jeśli ten program użytkowy nie wywoła wywołania IBM MQ . W regionie BMP połączenie z wątkiem jest nawiązywać, gdy aplikacja tworzy swoje pierwsze wywołanie produktu IBM MQ (MQCONN lub MQCONNX). Ten wątek jest zachowywany przez cały czas trwania regionu lub do momentu zatrzymania połączenia.

Zarówno dla regionów sterowanych komunikatami, jak i regionów niesterowanych komunikatami, identyfikator odwołania do wątku odtwarzania, *Thread-xref*, powiązany z wątkiem to:

```
PSTid + PSBname
```

gdzie:

PSTid

Identyfikator regionu tabeli specyfikacji partycji

PSBname

Nazwa bloku specyfikacji programu

Identyfikatory połączeń mogą być używane jako unikalne identyfikatory w komendach IBM MQ , w których to przypadku produkt IBM MQ automatycznie wstawia te identyfikatory do dowolnego komunikatu operatora, który generuje.

Wyświetlanie w wątpliwych jednostkach odzyskiwania

Użytkownik może wyświetlać wątpliwość jednostek odzyskiwania i próbować je odzyskać.

Kroki operacyjne używane do wyświetlania i odtwarzania w wątpliwych jednostkach odzyskiwania w tym temacie dotyczą tylko stosunkowo prostych obserwacji. Jeśli menedżer kolejek zostanie nieprawidłowo zakończony podczas połączenia z serwerem IMS, produkt IMS może zatwierdzić lub wycofać pracę bez tego, że produkt IBM MQ jest tego świadomy. Po zrestartowaniu menedżera kolejek ta praca jest *wątpliwa*. Należy podjąć decyzję w sprawie statusu utworu.

Aby wyświetlić listę wątpliwych jednostek odzyskiwania, wydaj komendę:

```
+CSQ1 DISPLAY CONN(*) WHERE(UOWSTATE EQ UNRESOLVED)
```

Produkt IBM MQ odpowiada na komunikat podobny do następującego:

```
CSQM201I +CSQ1 CSQMDRTC DIS CONN DETAILS
CONN(BC0F6125F5A30001)
EXTCONN(C3E2D8C3C3E2D8F140404040404040)
TYPE(CONN)
CONNOPTS(
MQCNO_STANDARD_BINDING
)
UOWLOGDA(2004-11-02)
UOWLOGTI(12.27.58)
UOWSTDA(2004-11-02)
UOWSTTI(12.27.58)
UOWSTATE(UNRESOLVED)
NID(CSQ1CHIN.BC0F5F1C86FC0766)
EXTURID(000000000000001F000000007472616E5F6964547565204E6F762020...)
QMURID(0000000026232)
URTYPE(XA)
USERID( )
APPLTAG(CSQ1CHIN)
ASID(0000)
APPLTYPE(CHINIT)
CHANNEL( )
CONNNAME( )
END CONN DETAILS
```

Wyjaśnienie atrybutów w tym komunikacie można znaleźć w opisie komendy [DISPLAY CONN](#) .

Odzyskiwanie w wątpliwych jednostkach odzyskiwania

Aby odtworzyć wątpliwe jednostki odzyskiwania, wydaj następującą komendę:


```
+CSQ1 RESOLVE INDOUBT( connection-name ) ACTION(COMMIT|BACKOUT)
NID( net-node.number )
```

gdzie:

connection-name

Identyfikator systemu IMS .

DZIAŁANIE

Wskazuje, czy jednostka odtwarzania ma być zatwierdzana (COMMIT), czy wycofana (BACKOUT).

net-node.number

Powiązany net-node.number.

Po wydaniu komendy RESOLVE INDOUBT, zostanie wyświetlony jeden z następujących komunikatów:

```
CSQV414I +CSQ1 THREAD network-id COMMIT SCHEDULED
CSQV415I +CSQ1 THREAD network-id BACKOUT SCHEDULED
```

Rozstrzygnięcie pozycji odzyskiwania reszt

W danym momencie program IMS tworzy listę pozostałych pozycji odtwarzania (RREs). RREs to jednostki odzyskiwania, w których IBM MQ może mieć wątpliwości. Pojawiają się one w kilku sytuacjach:

- Jeśli menedżer kolejek nie jest aktywny, produkt IMS ma RREs, których nie można rozstrzygnąć, dopóki menedżer kolejek nie będzie aktywny. Te RREs nie są problemem.
- Jeśli menedżer kolejek jest aktywny i połączony z produktem IMS, a program IMS wycofuje prace, które zostały zatwierdzone przez produkt IBM MQ , adapter IMS generuje komunikat CSQQ010E. Jeśli dane w tych dwóch systemach muszą być spójne, występuje problem. Więcej informacji na temat rozwiązywania tego problemu zawiera sekcja [“Ręczne odtwarzanie IMS jednostek odzyskiwania” na stronie 434.](#)
- Jeśli menedżer kolejek jest aktywny i połączony z programem IMS, nadal mogą istnieć elementy RRES, nawet jeśli żadne komunikaty nie poinformowały użytkownika o tym problemie. Po nawiązaniu połączenia IBM MQ z produktem IMS można wydać następującą komendę IMS , aby sprawdzić, czy wystąpił problem:

```
/DISPLAY OASN SUBSYS sysid
```

Aby wyczyścić RRE, wydaj jedną z następujących komend IMS :

```
/CHANGE SUBSYS sysid RESET
/CHANGE SUBSYS sysid RESET OASN nnnn
```

gdzie *nnnn* to początkowy numer kolejny aplikacji wymieniony w odpowiedzi na komendę DISPLAY +CSQ1 . Jest to numer harmonogramu instancji programu, który jest umieszczany w sekwencji wywołań tego programu od czasu ostatniego zimnego startu programu IMS . Produkt IMS nie może mieć dwóch wątpliwych jednostek odtwarzania o tym samym numerze harmonogramu.

Te komendy resetują status IMS ; nie powodują one żadnej komunikacji z produktem IBM MQ.

Sterowanie połączeniami regionu zależnego IMS

Użytkownik może sterować, monitorować i w razie potrzeby przerywać połączenia między programem IMS i produktem IBM MQ.

Sterowanie połączeniami regionu zależnego od produktu IMS obejmuje następujące działania:

- [Nawiązywanie połączenia z regionów zależnych](#)
- [Opcje błędu regionu](#)
- [Monitorowanie aktywności połączeń](#)
- [Odłączanie od regionów zależnych](#)

Łączenie z regionów zależnych

Adapter IMS używany w regionie sterującym jest również ładowany do regionów zależnych. Połączenie jest nawiązywane z każdego regionu zależnego do produktu IBM MQ. To połączenie jest używane do koordynowania zobowiązań dotyczących pracy IBM MQ i IMS . Aby zainicjować i nawiązać połączenie, program IMS wykonuje następujące czynności:

1. Odczytuje podzbiór podsystemu (SSM) z IMS.PROCLIB.

Podzbiór podsystemu może być określony w parametrze EXEC regionu zależnego. Jeśli nie zostanie ona określona, używany jest region sterujący SSM. Jeśli region nigdy nie nawiąże połączenia z produktem IBM MQ, aby uniknąć ładowania adaptera, należy określić element bez żadnych pozycji.

2. Ładuje adapter IBM MQ .

W przypadku programu komunikatu wsadowego ładowanie nie jest wykonywane do momentu wydania przez aplikację pierwszej komendy przesyłania komunikatów. W tym czasie program IMS próbuje nawiązać połączenie.

W przypadku regionu programu przetwarzania komunikatów lub regionu fast-path programu IMS podejmowana jest próba zainicjowania regionu.

Opcje błędów regionu

Jeśli menedżer kolejek nie jest aktywny lub jeśli zasoby nie są dostępne, gdy pierwsza komenda przesyłania komunikatów jest wysyłana z programów aplikacji, to działanie jest zależne od opcji błędu określonej w pozycji SSM. Dostępne są następujące opcje:

R

Odpowiedni kod powrotu jest wysyłany do aplikacji.

Q

Aplikacja kończy działanie nieprawidłowo z kodem nieprawidłowego zakończenia U3051. Komunikat wejściowy zostanie ponownie umieszczony w kolejce.

A

Aplikacja kończy działanie nieprawidłowo z kodem abend U3047. Komunikat wejściowy jest odrzucony.

Monitorowanie aktywności połączeń

Wątek jest ustanawiany z regionu zależnego, gdy aplikacja tworzy pierwsze pomyślne żądanie produktu IBM MQ . Informacje o połączeniach i aplikacjach, które są obecnie używane przez użytkownika, można wyświetlić za pomocą następującej komendy z programu IBM MQ:

```
+CSQ1 DISPLAY CONN(*) ALL
```

Komenda generuje komunikat podobny do następującego:

```
CONN(BC45A794C4290001)
EXTCONN(C3E2D8C3C3E2D8F140404040404040)
TYPE(CONN)
CONNOPTS(
MQCNO_STANDARD_BINDING
)
UOWLOGDA(2004-12-15)
UOWLOGTI(16.39.43)
UOWSTDA(2004-12-15)
UOWSTTI(16.39.43)
UOWSTATE(ACTIVE)
NID( )
EXTURID(
0000052900000000
)
QMURID(00000354B76E)
URTYPE(IMS)
USERID(STCPI)
APPLTAG(IM8F)
ASID(0049)
APPLTYPE(IMS)
PSTID(0004)
PSBNAME(GP01MPP)
```

W przypadku regionu sterującego *thread-xref* jest wartością specjalną CONTROL. W przypadku regionów zależnych jest to identyfikator PSTid konkatenowany z nazwą PSBname. *auth-id* jest to pole użytkownika z karty pracy lub identyfikator z tabeli procedur uruchomionych w programie z/OS .

Wyjaśnienie wyświetlanej listy znajduje się w opisie komunikatu CSQV402I w dokumentacji produktu [IBM MQ for z/OS](#) : komunikaty, zakończenie i kody przyczyny .

Produkt IMS udostępnia komendę wyświetlania do monitorowania połączenia z produktem IBM MQ. Pokazuje on, który program jest aktywny dla każdego połączenia regionu zależnego, nazwy użytkownika LTERM i statusu połączenia regionu sterowania. Komenda jest następująca:

```
/DISPLAY SUBSYS name
```

Status połączenia między IMS a IBM MQ jest wyświetlany jako jeden z następujących:

```
CONNECTED
NOT CONNECTED
CONNECT IN PROGRESS
STOPPED
STOP IN PROGRESS
INVALID SUBSYSTEM NAME= name
SUBSYSTEM name NOT DEFINED BUT RECOVERY OUTSTANDING
```

Status wątku z każdego regionu zależnego to jeden z następujących:

```
CONN
CONN, ACTIVE (includes LTERM of user)
```

Rozłączanie z regionami zależnymi

Aby zmienić wartości w elemencie SSM produktu IMS.PROCLIB, należy odłączyć region zależny. Aby to zrobić, należy:

1. Wydadz komendę IMS :

```
/STOP REGION
```

2. Zaktualizuj element SSM.

3. Wydadz komendę IMS :

```
/START REGION
```

odłączanie odIMS

Połączenie zostanie zakończone, gdy produkt IMS lub menedżer kolejek zakończy działanie. Alternatywnie, główny operator terminalu IMS może jawnie przerwać połączenie.

Aby zakończyć połączenie między IMS i IBM MQ, należy użyć następującej komendy IMS :

```
/STOP SUBSYS sysid
```

Komenda wysłała następujący komunikat do terminalu, który go wydał, zazwyczaj główny operator terminalu (MTO):

```
DFS058I STOP COMMAND IN PROGRESS
```

Komenda IMS :

```
/START SUBSYS sysid
```

jest wymagane do ponownego nawiązania połączenia.

Uwaga: Komenda IMS /STOP SUBSYS nie jest zakończona, jeśli uruchomiony jest monitor wyzwalacza IMS .

Sterowanie monitorem wyzwalacza IMS

Istnieje możliwość użycia transakcji CSQQTRMN do zatrzymania i uruchomienia monitora wyzwalacza IMS .

Monitor wyzwalacza IMS (transakcja CSQQTRMN) jest opisany w sekcji [Konfigurowanie monitora wyzwalacza IMS](#).

Aby sterować monitorem wyzwalacza IMS , patrz:

- [Uruchamianie CSQQTRMN](#)

- Zatrzymywanie CSQQTRMN

Uruchamianie CSQQTRMN

1. Uruchom BMP zorientowany na zadanie, który uruchamia program CSQQTRMN dla każdej kolejki inicjuj, która ma być monitorowana.
2. Zmodyfikuj zadanie wsadowe JCL w taki sposób, aby dodać nazwę DDname CSQOUT1 , która wskazuje zestaw danych zawierający następujące informacje:

```
QMGRNAME=q_manager_name      Comment: queue manager name
INITQUEUEUENAME=init_q_name    Comment: initiation queue name
LTERM=lterm                    Comment: LTERM to remove error messages
CONSOLEMESSAGES=YES           Comment: Send error messages to console
```

gdzie:

q_manager_name	Nazwa menedżera kolejek (jeśli jest to pole puste, przyjmowana jest wartość domyślna nominowana w CSQQDEFV)
init_q_name	Nazwa kolejki inicjuj, która ma być monitorowana
lterm	Nazwa LTERM produktu IMS dla miejsca docelowego komunikatów o błędach (jeśli jest to pole puste, wartością domyślną jest MASTER).
CONSOLEMESSAGES= YES	Żąda, aby komunikaty wysłane do nominowanego IMS LTERM zostały również wysłane do konsoli z/OS . Jeśli ten parametr zostanie pominięty lub został niepoprawnie zapisany, wartością domyślną NIE będzie wysyłanie komunikatów do konsoli.

3. Dodaj nazwę DD CSQOUT2 , jeśli chcesz, aby wydrukowany raport przetwarzania danych wejściowych CSQOUT1 był przetwarzany.

Uwaga:

1. Zestaw danych CSQOUT1 jest zdefiniowany z LRECL=80. Inne informacje DCB są pobierane z zestawu danych. Baza danych DCB dla zestawu danych CSQOUT2 to RECFM=VBA i LRECL=125.
2. Dla każdego rekordu można umieścić tylko jedno słowo kluczowe. Wartość słowa kluczowego jest oddzielona od pierwszego odstępów po słowie kluczowym; oznacza to, że można uwzględnić komentarze. Gwiazdka w kolumnie 1 oznacza, że cały rekord wejściowy jest komentarzem.
3. W przypadku błędnego zakłęcia słów kluczowych QMGRNAME lub LTERM , CSQQTRMN używa wartości domyślnej dla tego słowa kluczowego.
4. Upewnij się, że podsystem został uruchomiony w IMS (za pomocą komendy /START SUBSYS) przed przestaniem zadania monitora wyzwalacza BMP. Jeśli nie jest on uruchomiony, zadanie monitora wyzwalacza kończy pracę z kodem abend U3042.

Zatrzymywanie CSQQTRMN

Po uruchomieniu CSQQTRMN działa do momentu, gdy połączenie między IBM MQ i IMS zostanie zerwane ze względu na jedno z następujących zdarzeń:

- zakończenie menedżera kolejek
- IMS zamykanie

lub została wprowadzona komenda z/OS STOP **jobname** .

W tej sekcji przedstawiono opis komend produktu IMS , których można użyć do sterowania mostem IMS .

Brak komend IBM MQ do sterowania mostem IBM MQ-IMS . Można jednak zatrzymać dostarczanie komunikatów do programu IMS w następujący sposób:

- W przypadku kolejek niewspółużytkowanych, za pomocą komendy ALTER QLOCAL (xxx) GET (DISABLED) dla wszystkich kolejek mostów.
- W przypadku kolejek klastrowych za pomocą komendy SUSPEND QMGR CLUSTER (xxx). Jest to skuteczne tylko wtedy, gdy inny menedżer kolejek obsługuje również kolejkę mostu klastrowego.
- W przypadku kolejek klastrowych za pomocą komendy SUSPEND QMGR FACILITY (IMSBRIDGE). Żadne dodatkowe komunikaty nie są wysyłane do produktu IMS, ale odpowiedzi dotyczące wszystkich zaległych transakcji są odbierane z produktu IMS.

Aby ponownie uruchomić wysyłanie komunikatów do produktu IMS , wydaj komendę RESUME QMGR FACILITY (IMSBRIDGE).

Aby wyświetlić, czy most jest zawieszony, można również użyć komendy MQSC DISPLAY SYSTEM.

Szczegółowe informacje na temat tych komend zawiera sekcja [Komendy MQSC](#) .

Więcej informacji na ten temat zawiera:

- [“Uruchamianie i zatrzymywanie mostu IMS” na stronie 446](#)
- [“Sterowanie połączeniami produktu IMS” na stronie 446](#)
- [Sterowanie kolejkami mostu](#)
- [“Resynchronizacja mostu IMS” na stronie 448](#)
- [Praca z nazwami tppe](#)
- [Usuwanie komunikatów z produktu IMS](#)
- [Usuwanie potoków](#)
- [“IMS Utrata ważności transakcji” na stronie 450](#)

Uruchamianie i zatrzymywanie mostu IMS

Uruchom most IBM MQ , uruchamiając OTMA. Użyj komendy IMS :

```
/START OTMA
```

lub uruchom go automatycznie, podając parametr OTMA=YES w parametrach systemowych IMS . Jeśli OTMA jest już uruchomione, most zostanie uruchomiony automatycznie po zakończeniu uruchamiania menedżera kolejek. Po uruchomieniu OTMA generowany jest komunikat zdarzenia IBM MQ .

Użyj komendy IMS :

```
/STOP OTMA
```

w celu zatrzymania komunikacji OTMA. Po wydaniu tej komendy generowany jest komunikat zdarzenia IBM MQ .

Sterowanie połączeniami produktu IMS

Produkt IMS udostępnia te komendy operatora do sterowania i monitorowania połączenia z produktem IBM MQ:

/DEQUEUE TMEMBER *tmember* TPIPE *tpipe*

Usuwa komunikaty z potoku Tpipe. Aby usunąć wszystkie komunikaty lub opcję PURGE1 , aby usunąć tylko pierwszy komunikat, należy określić wartość PURGE.

/DISPLAY OTMA

Wyświetla podsumowanie informacji na temat serwera i klientów OTMA oraz statusu klienta.

/DISPLAY TMEMBER *nazwa*

Wyświetla informacje na temat klienta OTMA.

/DISPLAY TRACE TMEMBER *nazwa*

Wyświetla informacje na temat śledzonych elementów.

/BEZPIECZNA OTMA

Ustawia opcje zabezpieczeń.

/START OTMA

Umożliwia komunikację za pośrednictwem OTMA.

/START TMEMBER *tmember* TPIPE *tpipe*

Uruchamia nazwaną Tpipe.

/STOP OTMA

Zatrzymuje komunikację za pośrednictwem OTMA.

/STOP TMEMBER *tmember* TPIPE *tpipe*

Zatrzymuje nazwaną Tpipe.

/TRACE

Steruje śledzeniem IMS .

Więcej informacji na temat tych komend znajduje się w podręczniku *Informacje o operatorach produktu IMS/ESA* dla używanego poziomu produktu IMS .

Odpowiedzi komend IMS są wysyłane do terminalu, z którego została wydana komenda. Autoryzacja do wydawania komend IMS jest oparta na zabezpieczeniach produktu IMS .

Sterowanie kolejkami mostów

Aby przerwać komunikację z menedżerem kolejek o nazwie elementu *tmember* XCF przez most, należy wprowadzić następującą komendę IMS :

```
/STOP TMEMBER tmember TPIPE ALL
```

Aby wznowić komunikację, wydaj następującą komendę IMS :

```
/START TMEMBER tmember TPIPE ALL
```

Potoki dla kolejki mogą być wyświetlane za pomocą komendy MQ DISPLAY QUEUE.

Aby zatrzymać komunikację z menedżerem kolejek w pojedynczym potoku Tpipe, należy wprowadzić następującą komendę IMS :

```
/STOP TMEMBER tmember TPIPE tpipe
```

Dla każdej aktywnej kolejki mostu tworzone są dwa lub dwa potoków, więc wydanie tej komendy spowoduje zatrzymanie komunikacji z kolejką IBM MQ . Aby wznowić komunikację, użyj następującej komendy IMS :

```
/START TMEMBER tmember TPIPE tpipe
```

Alternatywnie można zmienić atrybuty kolejki produktu IBM MQ , aby została ona zablokowana.

Resynchronizacja mostu IMS

Most IMS jest automatycznie restartowany za każdym razem, gdy restartowany jest menedżer kolejek, IMSlub OTMA.

Pierwszym zadaniem realizowanego przez most IMS jest resynchronizacja z produktem IMS. Obejmuje to IBM MQ i IMS numery porządkowe we wszystkich synchronizowanych systemach Tpipe. Zsynchronizowana potok Tpipe jest używana, gdy komunikaty trwałe są wysyłane do programu IMS z kolejki mostu IBM MQ - IMS przy użyciu trybu kontroli transakcji zero (commit-then-send).

Jeśli most nie może resynchronizować z produktem IMS, kod rozpoznania IMS jest zwracany w komunikacie CSQ2023E , a połączenie z OTMA zostało zatrzymane. Jeśli most nie może resynchronizować z pojedynczym rpipe IMS Tpipe, kod rozpoznania IMS jest zwracany w komunikacie CSQ2025E , a Tpipe jest zatrzymany. Jeśli Tpipe jest zimny, odtwarzalne numery kolejne są automatycznie resetowane do 1.

Jeśli most wykryje niezgodne numery kolejne podczas resynchronizacji z potok Tpipe, zostanie wyświetlony komunikat CSQ2020E . Użyj komendy IBM MQ RESET TPIPE, aby zainicjować resynchronizację za pomocą potoku IMS Tpipe. Należy podać nazwę i nazwę podzbioru XCF oraz nazwę Tpipe. Informacje te są udostępniane przez komunikat.

Można również określić:

- Nowy odtwarzalny numer kolejny, który ma zostać ustawiony w potoku Tpipe dla komunikatów wysyłanych przez produkt IBM MQ, i który ma zostać ustawiony jako numer kolejny odbieranego partnera. Jeśli ta opcja nie zostanie określona, numer kolejny odbieranego partnera zostanie ustawiony na bieżący numer kolejny wysyłania IBM MQ .
- Nowy odtwarzalny numer kolejny, który ma zostać ustawiony w potoku Tpipe dla komunikatów odebranych przez produkt IBM MQ i który ma zostać ustawiony jako numer kolejny w kolejności wysyłania partnera. Jeśli ta opcja nie zostanie określona, numer kolejny wysyłania partnera jest ustawiany na bieżący numer kolejny odbioru IBM MQ .

Jeśli istnieje nierozstrzygnięta jednostka odtwarzania powiązana z Tpipe, jest ona również powiadamiana w komunikacie. Komenda IBM MQ RESET TPIPE służy do określania, czy jednostka odtwarzania ma być zatwierdzona, czy też wycofana. Jeśli jednostka odtwarzania zostanie zatwierdzona, zadanie wsadowe komunikatów zostało już wysłane do programu IMS i zostanie usunięte z kolejki mostu. Jeśli jednostka odtwarzania jest wycofana, komunikaty są zwracane do kolejki mostu w celu późniejszego wysłania do programu IMS.

Tryb kontroli transakcji 1 (send-then-commit) Trury nie są synchronizowane.

Uwagi dotyczące transakcji w trybie kontroli transakcji 1

W programie IMS transakcje w trybie kontroli transakcji 1 (CM1) wysyłają swoje odpowiedzi wyjściowe przed punktem synchronizacji.

Transakcja CM1 może nie być w stanie wysłać odpowiedzi, na przykład dlatego, że:

- Tpipe, na którym ma zostać wysłana odpowiedź, jest zatrzymany
- Zatrzymano OTMA
- Klient OTMA (czyli menedżer kolejek) odszedł od siebie
- Kolejka odpowiedzi i kolejka niedostarczonych komunikatów są niedostępne.

Z tych powodów aplikacja IMS wysyła pseudo-abkońskie komunikaty o kodzie U0119. Transakcja i program IMS nie są zatrzymani w tym przypadku.

Przyczyny te często uniemożliwiają wysyłanie komunikatów do produktu IMS, jak również odpowiedzi dostarczane z produktu IMS. Abend U0119 może wystąpić, jeśli:

- Menedżer kolejek (Tpipe, OTMA) lub menedżer kolejek został zatrzymany, gdy komunikat znajduje się w katalogu IMS .
- IMS odpowiedzi na inny Tpipe w komunikacie przychodzącym oraz że Tpipe jest zatrzymany
- IMS odpowiada na innego klienta OTMA, a ten klient jest niedostępny.

Za każdym razem, gdy wystąpi błąd U0119 , zarówno komunikat przychodzący do programu IMS , jak i komunikaty odpowiedzi do programu IBM MQ są tracone. Jeśli dane wyjściowe transakcji CMO nie mogą zostać dostarczone z któregośkolwiek z tych powodów, zostanie umieszczone w kolejce Tpipe w produkcie IMS.

Praca z nazwami tpipe

Wiele z komend używanych do sterowania mostem IBM MQ - IMS wymaga nazwy *tpipe* . W tym temacie opisano sposób znajdowania dalszych szczegółów nazwy *tpipe* .

Dla wielu komend, które sterują mostem IBM MQ - IMS , potrzebne są nazwy *tpipe* . Nazwy *tpipe* można pobrać z komendy DISPLAY QUEUE, a następnie zwrócić uwagę na następujące punkty:

- Nazwy *tpipe* są przypisywane po zdefiniowaniu kolejki lokalnej
- W kolejce lokalnej podano dwie nazwy *tpipe*, jeden do synchronizacji i drugi dla niesynchronizowanych
- Nazwy *tpipe* nie będą znane IMS , dopóki nie zostanie określona komunikacja między IMS i IBM MQ specyficzną dla danej kolejki lokalnej.
- Aby można było korzystać z mostu *tpipe* w celu użycia przez most IBM MQ - IMS , powiązana z nim kolejka musi być przypisana do klasy pamięci masowej, która ma poprawne pola grupy XCF i nazwy elementu.

Usuwanie komunikatów z produktu IMS

Jeśli element Tmember/Tpipe jest zatrzymany, można usunąć komunikat, który jest przeznaczony dla produktu IBM MQ za pośrednictwem mostu IMS . Aby usunąć jeden komunikat dla menedżera kolejek z elementem XCF o nazwie *tmember*, należy wprowadzić następującą komendę IMS :

```
/DEQUEUE TMEMBER tmember TPIPE tpipe PURGE1
```

Aby usunąć wszystkie komunikaty w potoku Tpipe, wydaj następującą komendę IMS :

```
/DEQUEUE TMEMBER tmember TPIPE tpipe PURGE
```

Usuwanie potoków

Nie można samodzielnie usunąć potoków IMS . Są one usuwane przez program IMS w następujących momentach:

- Synchronizowane potoków są usuwane, gdy program IMS jest zimny.
- Niesynchronizowane potoki są usuwane po zrestartowaniu programu IMS .

IMS Utrata ważności transakcji

Czas utraty ważności jest powiązany z transakcją. Z każdym komunikatem IBM MQ może być powiązany czas utraty ważności. Okres ważności jest przekazywany z aplikacji do produktu IBM MQ przy użyciu deskryptora MQMD.Expiry. Czas to czas trwania komunikatu przed jego wygaśnięciem, wyrażony jako wartość w dziesiątych częściach sekundy. Próba wykonania operacji MQGET komunikatu, później niż utraciła ważność, powoduje usunięcie komunikatu z kolejki i wykonane przetwarzanie utraty ważności. Czas utraty ważności jest zmniejszany w miarę przepływu komunikatów między menedżerami kolejek w sieci IBM MQ. Gdy komunikat IMS jest przekazywany przez most IMS do OTMA, pozostały czas utraty ważności komunikatu jest przekazywany do OTMA jako czas utraty ważności transakcji.

Jeśli dla transakcji określono czas utraty ważności, OTMA traci ważność transakcji wejściowych w trzech różnych miejscach w produkcie IMS:

- komunikat wejściowy odbierający z XCF
- czas kolejkowania komunikatu wejściowego
- czas GU aplikacji

Po czasie GU nie jest wykonywane żadne przedawnienie.

Transakcja EXPRTIME może być udostępniana przez:

- IMS Definicja transakcji
- Nagłówek komunikatu OTMA produktu IMS
- Wyjście użytkownika programu IMS DFSINSX0
- Komendy IMS CREATE lub UPDATE TRAN

IMS wskazuje, że transakcja utraciła ważność, abending transakcję z 0243, a następnie wydanie komunikatu. Komunikat ten ma wartość DFS555I w środowisku bez współużytkowania kolejek lub DFS2224I w środowisku kolejek współużytkowanych.

z/OS

Operacyjne Advanced Message Security w systemie z/OS

Przestrzeń adresowa Advanced Message Security akceptuje komendy za pomocą komendy z/OS MODIFY.

Aby wprowadzić komendy dla przestrzeni adresowej Advanced Message Security, należy użyć komendy z/OS MODIFY.

Na przykład składnia

```
F qmgr AMSM, cmd
```

Akceptowane są następujące komendy MODIFY:

Komenda	Opcja	Opis
WYŚWIETL		Wyświetl informacje o wersji
ODŚWIEŻ	KEYRING strategia ALL	Odśwież certyfikaty kluczy, strategie bezpieczeństwa lub oba te certyfikaty.
KONTROLA SMFAUDIT	SUKCES FAILURE ALL	Określa, czy kontrola SMF jest wymagana, gdy AMS pomyślnie zabezpiecza/unchroni komunikaty, gdy AMS nie zabezpiecza/nie chroni komunikatów, ani obu tych elementów.

Tabela 27. Komendy MODIFY przestrzeni adresowej Advanced Message Security (kontynuacja)

Komenda	Opcja	Opis
TYP_MFP	0 - 255	Ustaw typ rekordu SMF, który ma być generowany, gdy AMS zabezpieczy/unchroni komunikaty. Aby wyłączyć kontrolę SMF, należy określić typ rekordu o wartości 0.

Uwaga: Aby określić opcję, musi ona być oddzielona przecinkiem. Na przykład:

```
F qmgIAMS,REFRESH KEYRING
F qmgIAMS,SMFAUDIT ALL
F qmgIAMS,SMFTYPE 180
```

Komenda REFRESH.

Aplikacja, która wywołała wywołanie MQOPEN, poda zmiany. Istniejące aplikacje nadal korzystają z opcji, z których ta aplikacja otworzyła kolejkę. Aby pobrać zmiany, aplikacja musi zamknąć i ponownie otworzyć kolejkę.

z/OS

V 9.0.1

Dostawca usług IBM MQ for z/OS dla produktu z/OS

Connect

Produkt IBM MQ for z/OS Service Provider for z/OS Connect (MQ Service Provider) przetwarza żądania przychodzące za pośrednictwem produktu z/OS Connect. Produkt MQ Service Provider umożliwia aplikacjom REST rozpoznanie interakcji z zasobami aplikacyjnymi z/OS, które są ujawniane przy użyciu kolejek i tematów produktu IBM MQ for z/OS. Można to osiągnąć bez obaw związanych z kodowaniem wymagającym do korzystania z asynchronicznego przesyłania komunikatów.

Ważne: Produkt z/OS Connect EE w wersji 3.0.21.0 i nowszych dostarcza rozszerzoną wersję produktu MQ Service Provider, która obsługuje pliki archiwum usług. Należy przeprowadzić migrację do tej wersji produktu z/OS Connect EE i użyć wbudowanej wersji MQ Service Provider, a nie dostawcy usług dostarczanego z produktem IBM MQ for z/OS.

W informacjach dotyczących produktu z/OS Connect EE w produkcie IBM Documentationscenariusze szybkiego startu dla produktu MQ Service Provider w produkcie z/OS Connect EE są udostępniane w sekcji [Scenariusze szybkiego startu](#), a szczegółowe informacje dodatkowe są dostępne w sekcji [Korzystanie z dostawcy usług IBM MQ](#).

Ta sekcja składa się z następujących informacji:

Informacje pokrewne

Film wideo: [IBM MQ Service Provider for z/OS Connect \(YouTube\)](#)

z/OS

V 9.0.1

Produkt IBM MQ for z/OS Service Provider for z/OS

Connect-przegląd

Przegląd produktu IBM MQ for z/OS Service Provider for z/OS Connect, omówienie stosowanych zasad i komend używanych przez dostawcę usług.

Ważne: Produkt z/OS Connect EE w wersji 3.0.21.0 i nowszych dostarcza rozszerzoną wersję produktu MQ Service Provider, która obsługuje pliki archiwum usług. Należy przeprowadzić migrację do tej wersji produktu z/OS Connect EE i użyć wbudowanej wersji MQ Service Provider, a nie dostawcy usług dostarczanego z produktem IBM MQ for z/OS.

W informacjach dotyczących produktu z/OS Connect EE w produkcie IBM Documentationscenariusze szybkiego startu dla produktu MQ Service Provider w produkcie z/OS Connect EE są udostępniane w sekcji [Scenariusze szybkiego startu](#), a szczegółowe informacje dodatkowe są dostępne w sekcji [Korzystanie z dostawcy usług IBM MQ](#).

Produkt MQ Service Provider obsługuje dwie wersje produktu z/OS Connect:

IBM z/OS Connect wersja 1 (z/OS Connect V1)

Jest to komponent produktu WebSphere Liberty Profile (WLP) w systemie z/OS, dostępny bez dodatkowych kosztów, który prezentuje zasoby aplikacyjne produktu z/OS jako interfejs REST i umożliwia aplikacjom zdalnym, takim jak te działające na urządzeniu przenośnym, wysyłanie tych danych w formacie JSON.

Więcej informacji na ten temat zawiera sekcja [Przegląd produktu IBM z/OS Connect](#).

Ważne:

1. Produkt MQ Service Provider for IBM z/OS Connect V1 wymaga adaptera zasobów IBM WebSphere MQ 7.5 . Data zakończenia usługi wynosi 30th kwietnia 2018 r.

Po tym momencie program MQ Service Provider nie będzie obsługiwany za pomocą programu IBM z/OS Connect V1 , a zamiast tego należy użyć programu IBM z/OS Connect EE.

Z tego powodu produkt IBM z/OS Connect V1 jest lepiej umieszczany w celach programistycznych i dowodowych, a nie w środowiskach produkcyjnych.

2. MQ Service Provider obsługuje tylko kod produktu z/OS Connect V1 , który jest dostarczany w produkcie WLP 8.5.5.9 i nowszym.

IBM z/OS Connect Enterprise Edition (z/OS Connect EE)

Jest to osobny, wymagalny produkt oparty na produkcie WLP, który udostępnia całą funkcję w programie IBM z/OS Connect V1, a także szereg udoskonaleń, w tym narzędzia służące do generowania interfejsów API zgodnych ze specyfikacją REST-edytora API i integracja z produktem IBM API Connect.

Więcej informacji na ten temat zawiera sekcja [IBM z/OS Connect EE](#).

Ważne: Produkt MQ Service Provider obsługuje tylko produkt z/OS Connect EE V2.0.3.0 (APAR [PI66869](#)) lub nowszy.

Produkt MQ Service Provider udostępnia tę samą funkcję, niezależnie od tego, która wersja programu z/OS Connect jest używana. Wszystkie możliwości produktu z/OS Connect EE, takie jak edytor API, są obsługiwane przez produkt MQ Service Provider.

W całej dokumentacji termin *z/OS Connect* jest używany do odwołania się do obu wersji, o ile nie ma potrzeby odwołania się do konkretnej wersji, w takim przypadku używana jest jedna z nazw zdefiniowanych w poprzednim tekście.

Zasoby aplikacyjne produktu z/OS ujawnione przez produkt z/OS Connect nie muszą rozumieć usług REST ani JSON, ponieważ produkt z/OS Connect odwzorowuje wywołania usługi REST na wywołania lokalne, a także udostępnia transformacje do konwersji między obiektami JSON i lokalnymi strukturami danych, takimi jak książki kopii w języku COBOL.

Produkt MQ Service Provider jest dostępny jako osobny składnik WLP, który można zainstalować w odpowiedniej instalacji produktu z/OS Connect. Należy zauważyć, że większość prac związanych z konfiguracją produktu MQ Service Provider jest oparta na konfiguracji, dzięki czemu aplikacja REST nie musi mieć informacji na temat produktu IBM MQ. Dostęp do elementów, takich jak pola MQMD, jest jednak dostępny dla bardziej zaawansowanych aplikacji.

Produkt MQ Service Provider prezentuje IBM MQ kolejki i tematy jako [“Usługi” na stronie 453](#).

Obsługiwane wersje produktu IBM MQ

Produkt MQ Service Provider jest obsługiwany w produkcie IBM MQ for z/OS 8.0 i nowszych wersjach.

Wersje wcześniejsze niż IBM MQ 9.0.1 nie będą dostarczane razem z serwerem MQ Service Provider. Zamiast tego należy zapoznać się z [“Uzyskiwanie MQ Service Provider” na stronie 455](#) na temat sposobu uzyskiwania MQ Service Provider.

Dokumentacja opisana w sekcjach z/OS Connect produktu IBM Documentation jest jednakowo stosowana do wszystkich obsługiwanych wersji produktu IBM MQ.

Inne zagadnienia

Produkt MQ Service Provider korzysta z:

- HTTP GET dla wywołania MQGET, który nie jest destruktywny
- HTTP DELETE dla destrukcyjnego wywołania MQGET
- HTTP POST dla wywołania MQPUT

Są to te same czasowniki używane przez most IBM MQ dla protokołu HTTP. Podobnie w produkcji MQ Service Provider nagłówki HTTP są używane jako sposób określania różnych elementów, takich jak wartości MQMD.

Więcej informacji na temat wywołań HTTP zawiera sekcja [“Czasowniki obsługiwane przez produkt MQ Service Provider”](#) na stronie 454

Należy zauważyć, że dane JSON są jedynymi danymi, które są reprezentowane przez dane JSON. Oznacza to, że aplikacje nie muszą być narażone na kontakt z koncepcjami produktu IBM MQ, jeśli są potrzebne tylko dla danych komunikatu.

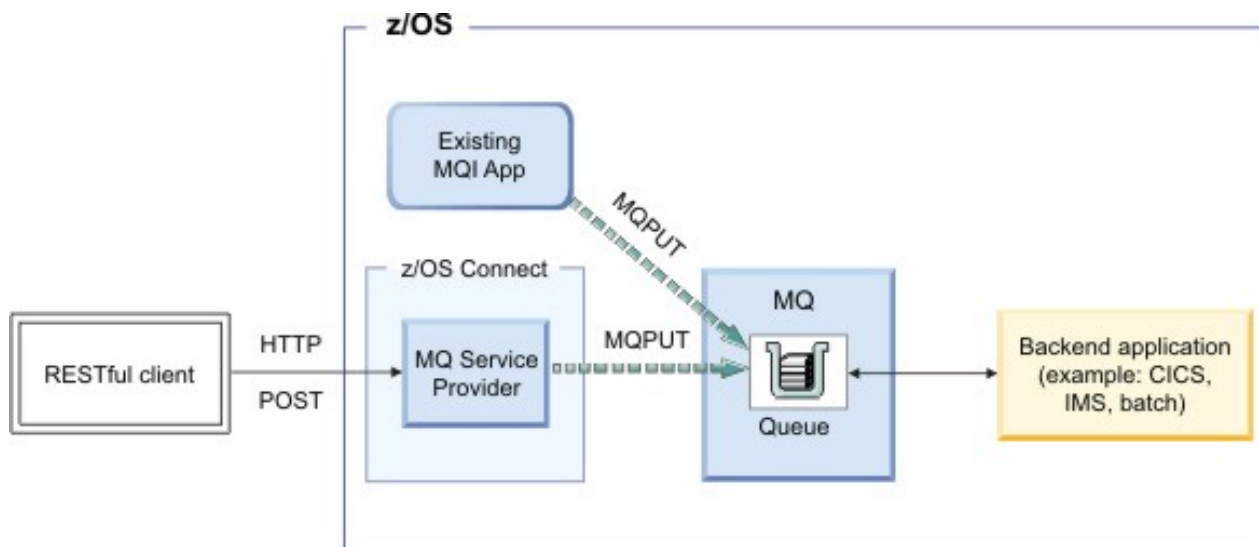
Produkt MQ Service Provider korzysta z obsługi adaptera zasobów produktu IBM MQ w produkcji WLP, a zatem jest oparty na usłudze JMS.

z/OS V 9.0.1 Usługi

Produkt MQ Service Provider prezentuje IBM MQ kolejki i tematy oraz aplikacje, które są za nimi, jako usługi. Istnieją dwa rodzaje usług: jednokierunkowy i dwukierunkowy, które opisano w tej sekcji.

usługa jednokierunkowa

Usługa jednokierunkowa może być używana do udostępniania interfejsu API zgodnego ze specyfikacją REST u góry pojedynczej kolejki lub tematu produktu IBM MQ. Klienci zgodne ze specyfikacją REST mogą wysłać żądania HTTP POST z ładunkiem JSON do usługi jednokierunkowej, a usługa pobiera ładunek i wysyła komunikat, z ładunkiem jako treść komunikatu, do docelowej kolejki lub tematu.



Rysunek 55. usługa jednokierunkowa

Usługi jednokierunkowe umożliwiają również wydawanie żądań HTTP DELETE i HTTP GET dla kolejek produktu IBM MQ.

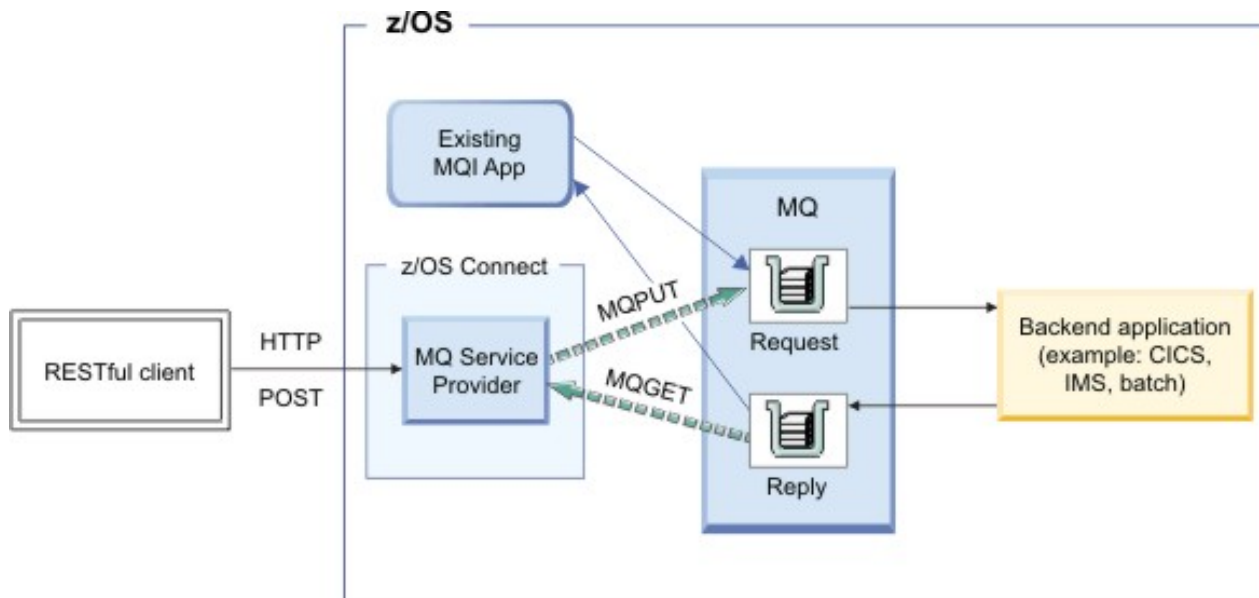
Żądanie HTTP DELETE powoduje destrukcyjne pobranie dostępnego komunikatu z kolejki. HTTP GET powoduje przejście do przeglądania pierwszego dostępnego komunikatu z kolejki.

Uwaga: Dwa wywołania HTTP GET zwracają ten sam komunikat, chyba że wystąpiło inne działanie, na przykład żądanie HTTP DELETE lub utrata ważności komunikatu, w celu usunięcia komunikatu z kolejki.

Treść komunikatu jest zwracana do klienta zgodnego ze specyfikacją REST w postaci JSON. Jeśli treść komunikatu nie jest jeszcze w formacie JSON, na przykład struktura copybook języka COBOL, produkt z/OS Connect można skonfigurować w taki sposób, aby przekształcał go w format JSON przy użyciu transformacji danych.

usługa dwukierunkowa

Usługa dwukierunkowa umożliwia klientowi RESTful wykonywanie przesyłania komunikatów żądanie-odpowiedź na parę kolejek.



Rysunek 56. usługa dwukierunkowa

Klient wysyła żądanie HTTP POST określające ładunek JSON. Usługa pobiera ładunek, opcjonalnie przekształca go w inny format, taki jak struktura copybook języka COBOL, i wysyła go jako komunikat do kolejki żądań.

Aplikacja zaplecza konsumuje komunikat, przetwarza je i generuje odpowiedź, która jest umieszczana w kolejce odpowiedzi. Usługa lokalizuje ten komunikat, pobiera jej ładunek, opcjonalnie przekształca ją w format JSON i zwraca ją jako treść odpowiedzi HTTP POST.

z/OS V 9.0.1 Czasowniki obsługiwane przez produkt MQ Service Provider

Produkt MQ Service Provider obsługuje żądania HTTP GET, HTTP DELETE i HTTP POST, które są używane w moście produktu IBM MQ dla protokołu HTTP, w tym samym znaczeniu. Użycie tych komend zezwala na podstawową interakcję z produktem IBM MQ for z/OS, a skład szeregu tych komend umożliwia ujawnianie bardziej skomplikowanych funkcji.

Podczas wykonywania operacji HTTP GET lub HTTP DELETE MQ Service Provider ignoruje wszystkie treści podane w żądaniu.

Jeśli operacja HTTP GET lub HTTP DELETE zakończy się pomyślnie:

- Ładunek komunikatu jest zwracany w treści odpowiedzi HTTP, a kod statusu HTTP to 200.
- Komunikat bez ładunku jest zwracany bez treści HTTP. Jeśli nie jest dostępny żaden komunikat, nie jest zwracany żaden treść HTTP, a kod statusu HTTP to 204 (Brak treści).



Ostrzeżenie: Program z/OS Connect oczekuje, że ładunki JSON zawsze będą w postaci obiektu. Jest to minimalny poprawny ładunek, który jest `{}`.

Podczas wykonywania testu POST HTTP jedyne oczekiwane parametry to ładunek komunikatu, który będzie znajdować się w treści HTTP, a także różne elementy opisane w elemencie `mzqOSConnectService`.

Jeśli test POST powiedzie się, wykonaj następujące czynności:

usługa jednokierunkowa

Zwraca kod HTTP 204 (Brak treści) wraz z pustym ładunkiem.

usługa dwukierunkowa

Zwraca treść komunikatu odpowiedzi w formacie JSON lub, jeśli nie jest dostępny żaden komunikat odpowiedzi, pustą treść odpowiedzi z kodem odpowiedzi HTTP 204 (Brak treści).

Nagłówki HTTP

W przypadku wysyłania żądania HTTP POST do usługi jednokierunkowej lub dwukierunkowej treść HTTP musi być poprawnym obiektem JSON. Należy określić:

- Nagłówek HTTP produktu Content-Type=application/json do żądania HTTP_POST
- Kodowanie UTF-8

Można również określić liczbę opcjonalnych nagłówków HTTP, w zależności od użytego czasownika HTTP, w celu dopasowania zachowania MQ Service Provider. Więcej informacji na ten temat zawiera sekcja [nagłówki HTTP, które mogą być używane z produktem MQ Service Provider](#).

Kody HTTP

Jeśli program MQ Service Provider wykryje błąd, usługa ustawia kod statusu HTTP w zakresie 400-599. W przeciwnym razie opisane wcześniej kody statusu są zawsze zwracane.

Błąd po stronie klienta

Jeśli w aplikacji wywołującej przekazano niepoprawne dane, do programu wywołującego zwracany jest kod statusu HTTP z zakresu od 400 do 499. Dokładny kod statusu zależy od błędu.

Jeśli niepoprawne dane zostały wykryte przez zgłoszony wyjątek, dane śledzenia stosu w formacie JSON są udostępniane na przykład na stronie serwera.

Błąd po stronie serwera

Jeśli program MQ Service Provider doświadczy nieoczekiwanego błędu, stos wywołań zostanie przekształconego w format JSON i zostanie zwrócony do programu wywołującego z kodem statusu HTTP 500 (Wewnętrzny błąd serwera). Odpowiednia diagnostyka zostanie również zapisana w dziennikach produktu z/OS Connect.

Uproszczonym przykładowym ładunkiem odpowiedzi błędu jest:

```
{
  "errorText": "CSQZ0006E: An unexpected JMSEException occurred while processing a request
for the 'mq7' service. ",
  "exceptionData": {
    "exceptions": [
      {
        "stackTrace": [
          "com.ibm.msg.client.jms.DetailedInvalidDestinationException: JMSWMQ2008:
Failed to open MQ queue 'ThisQueueDoesNotExist'.",
          <further content removed for brevity>
        ]
      },
      {
        "stackTrace": [
          "com.ibm.mq.MQException: JMSMQ0001: WebSphere MQ call failed with
compcode '2' ('MQCC_FAILED') reason '2085' ('MQRC_UNKNOWN_OBJECT_NAME').",
          <further content removed for brevity>
        ]
      }
    ]
  }
}
```

Uzyskiwanie MQ Service Provider

Produkt MQ Service Provider jest obsługiwany w celu użycia w produkcie IBM MQ for z/OS 8.0 i nowszych. Istnieją dwa sposoby uzyskania MQ Service Provider do użycia z dowolną obsługiwaną wersją produktu IBM MQ.

1. [“Obtaining the MQ Service Provider from the Unix Systems Services Components feature of IBM MQ 9.0.1 \(or later\)” na stronie 456](#)
2. [“Uzyskiwanie MQ Service Provider z serwisu Fix Central” na stronie 456](#)

Obtaining the MQ Service Provider from the Unix Systems Services Components feature of IBM MQ 9.0.1 (or later)

Zainstaluj składnik zgodnie z instrukcjami w katalogu programu. MQ Service Provider będzie znajdować się w katalogu *PATHPREFIX/zosconnect* w systemie plików, gdzie *PATHPREFIX* jest wartością wybraną w czasie instalacji. By default, for IBM MQ 9.0.1, *PATHPREFIX* is */usr/lpp/mqm/V9R0M1*.

Uzyskiwanie MQ Service Provider z serwisu Fix Central

Przejdź do serwisu [Fix Central](#) i znajdź produkt MQ Service Provider , wyszukując w systemie *IBM-MQ-zOSConnect-ServiceProvider*.

Powinno to znaleźć plik tar, na przykład *9.0.1-IBM-MQ-zOSConnect-ServiceProvider.tar*, który może zostać pobrany na stację roboczą.



Ostrzeżenie: Ten sam plik tar jest również używany razem z produktem IBM MQ 8.0 . Nie ma osobnego pliku tar dla IBM MQ 8.0.

Po pobraniu przenieś plik tar do katalogu *PATHPREFIX* na odpowiedniej partycji LPAR z/OS . Katalog ten powinien znajdować się w dowolnym miejscu, który jest dostępny dla produktu z/OS Connect.

Wyodrębnij pliki, wydając następującą komendę:

```
tar -oxvf 9.0.1-IBM-MQ-zOSConnect-ServiceProvider.tar
```

Zmień uprawnienia struktury katalogów w odpowiedni sposób dla przedsiębiorstwa.

Struktura katalogów

Oba sposoby uzyskiwania MQ Service Provider tworzą tę samą strukturę katalogów. Katalog główny tej struktury jest określany jako *MQSP_ROOT* w całej dokumentacji, gdzie *MQSP_ROOT* to *PATHPREFIX/zosconnect*.

Poniżej *MQSP_ROOT* znajdują się następujące pliki i katalogi:

mqzosconnect.properties	Properties file that can be copied into z/OS Connect
v1.0/ Connect V1	Directory containing MQ Service Provider for z/OS
lib/	
com.ibm.mq.zosconnect_1.0.0.jar	Feature jar file for MQ Service Provider for z/OS
Connect V1	
features/	
zosConnectMQ-1.0.mf	Feature manifest for MQ Service Provider for z/OS
Connect V1	
v2.0/ Connect EE	Directory containing MQ Service Provider for z/OS
lib/	
com.ibm.mq.zosconnect_2.0.0.jar	Feature jar file for MQ Service Provider for z/OS
Connect EE	
features/	
zosConnectMQ-2.0.mf	Feature manifest for MQ Service Provider for z/OS
Connect EE	

Uwagi dotyczące transakcji

Protokół HTTP nie jest protokołem transakcyjnym, dlatego nie jest możliwa koordynacja transakcyjna operacji przesyłania komunikatów wykonywanych przez produkt MQ Service Provider .

Ma to następujące konsekwencje:

- Jeśli metoda HTTP POST jest nawiązana do usługi jednokierunkowej, a połączenie nie powiedzie się przed odebraniem odpowiedzi HTTP przez klienta, klient nie może od razu stwierdzić, czy komunikat został wysłany do skonfigurowanej kolejki lub tematu, czy też nie.
- Jeśli do usługi jednokierunkowej zostanie wykonana operacja HTTP DELETE, a połączenie nie powiedzie się, zanim klient otrzyma odpowiedź HTTP, komunikat mógł zostać zniszczony z kolejki i utracony, ponieważ nie ma możliwości wycofywania destrukcyjnego powrotu.
- Jeśli do usługi dwukierunkowej zostanie wykonana operacja HTTP POST, a połączenie nie powiedzie się przed odebraniem odpowiedzi HTTP przez klienta, klient nie może stwierdzić, gdzie wystąpiła awaria. Komunikat żądania mógł zostać wysłany do kolejki żądań lub komunikat odpowiedzi mógł zostać wysłany z kolejki odpowiedzi i utracony.
- Nie istnieje sposób, aby skoordynować wynik wielu komend HTTP, zarówno jednokierunkowych, jak i dwukierunkowych.

IBM z/OS Connect EE - MQ Service Provider setup procedure

Przegląd komponentów, które należy zainstalować, aby umożliwić skonfigurowanie produktu MQ Service Provider w systemie IBM z/OS Connect EE.

O tym zadaniu

Wykonaj następujące czynności, w kolejności pokazanej w celu zainstalowania produktu MQ Service Provider i wszystkich jego wymagań wstępnych.

Instalowanie produktu IBM z/OS Connect EE

Wymagania i procedura instalowania produktu IBM z/OS Connect EE.

Zanim rozpoczniesz

Upewnij się, że masz dostęp do aktualizacji wbudowanego produktu WLP produktu z/OS Connect EE V2.

O tym zadaniu

Ta procedura służy do konfigurowania serwera z/OS Connect EE, który jest odpowiedni do użycia z serwerem MQ Service Provider. Jeśli istnieje już serwer, który został skonfigurowany w ten sam sposób, który może zostać użyty.

Procedura

1. Utwórz serwer, korzystając z instrukcji opisanych w następującym temacie: [Tworzenie serwera z/OS Connect EE](#)
2. Włącz autoryzowaną usługę TXRRS, postępując zgodnie z procedurą szczegółowo w sekcji [Konfigurowanie procesu Liberty Angel i autoryzowanych usług systemu z/OS](#).
3. Sprawdź, czy autoryzowana usługa TXRRS została poprawnie skonfigurowana przez:
 - a) Uruchamianie serwera.
Postępuj zgodnie z procedurą w sekcji [Uruchamianie i zatrzymywanie produktu z/OS Connect EE](#).
 - b) Wyświetl dzienniki serwera, które znajdują się pod adresem:

```
/var/zosconnect/servers/server_name/logs/messages.log
```

Są to pliki ASCII, które powinny zawierać dane wyjściowe podobne do poniższych:

```
A CWWKE0001I: The server test has been launched.
```

```
I CWWKB0103I: Authorized service group LOCALCOM is available.
I CWWKB0103I: Authorized service group PRODMGR is available.
I CWWKB0103I: Authorized service group SAFCREED is available.
I CWWKB0103I: Authorized service group TXRRS is available.
I CWWKB0103I: Authorized service group WOLA is available.
I CWWKB0103I: Authorized service group ZOSDUMP is available.
I CWWKB0103I: Authorized service group ZOSWLM is available.
I CWWKB0103I: Authorized service group CLIENT.WOLA is available.
I CWWKB0108I: IBM CORP product z/OS Connect version 02.00 successfully registered with z/OS
```

Sprawdź dane wyjściowe, aby sprawdzić, czy autoryzowana grupa usług TXRRS jest dostępna. W powyższym przykładzie wiersz tekstu pogrubionego pokazuje, że grupa usług jest dostępna.

Serwer MQ Service Provider wymaga, aby ta grupa usług była dostępna.

c) Zatrzymywanie serwera.

Postępuj zgodnie z procedurą w sekcji [Uruchamianie i zatrzymywanie serwera z poziomu wiersza komend](#)

Wyniki

Produkt z/OS Connect EE został pomyślnie zainstalowany.

Co dalej

Zainstaluj adapter zasobów produktu IBM MQ .

Zadania pokrewne

[“IBM z/OS Connect EE - MQ Service Provider setup procedure” na stronie 457](#)

Przegląd komponentów, które należy zainstalować, aby umożliwić skonfigurowanie produktu MQ Service Provider w systemie IBM z/OS Connect EE.

[“Instalowanie adaptera zasobów produktu IBM MQ” na stronie 458](#)

Funkcja dostawcy przesyłania komunikatów produktu IBM MQ w produkcie IBM z/OS Connect EE korzysta z komponentu IBM MQ zwanego adapterem zasobów produktu IBM MQ . Adapter zasobów jest dostarczany razem z produktem IBM MQ w ramach składnika IBM MQ for z/OS Unix System Services Components.

Instalowanie adaptera zasobów produktu IBM MQ

Funkcja dostawcy przesyłania komunikatów produktu IBM MQ w produkcie IBM z/OS Connect EE korzysta z komponentu IBM MQ zwanego adapterem zasobów produktu IBM MQ . Adapter zasobów jest dostarczany razem z produktem IBM MQ w ramach składnika IBM MQ for z/OS Unix System Services Components.

Zanim rozpocznie

Należy upewnić się, że zainstalowano składnik IBM MQ for z/OS Unix System Services Components dla wersji menedżera kolejek, z którym ma zostać nawiązane połączenie.

Jeśli ma zostać nawiązane połączenie z wieloma wersjami menedżera kolejek, należy użyć tej opcji z najnowszej wersji.

O tym zadaniu

Ta procedura umożliwi zainstalowanie adaptera zasobów produktu IBM MQ w produkcie IBM z/OS Connect EE.

Procedura

1. Znajdź katalog komponentów usług systemowych Unix w wersji produktu IBM MQ , z którą ma zostać nawiązane połączenie.

Na przykład ten katalog może mieć nazwę `/usr/lpp/mqm/V9R0M1/`. Katalog ten powinien zawierać podkatalog `java/lib`, który zawiera wiele bibliotek rodzimych (pliki `.so`).



Ostrzeżenie: Jeśli ma zostać nawiązane połączenie z wieloma wersjami menedżera kolejek, należy użyć najnowszej wersji.

2. Otwórz do edycji plik `server.xml` serwera z/OS Connect EE.

Dodaj następujące wiersze:

```
<variable name="wmqJmsClient.rar.location"
  value="MQJAVA_LIB_DIR/jca/wmq.jmsra.rar"/>
<wmqJmsClient nativeLibraryPath="MQJAVA_LIB_DIR"/>
```

gdzie `MQJAVA_LIB_DIR` jest wyłaczany z katalogu znajdującego się w kroku “1” na stronie 458, na przykład `/usr/lpp/mqm/V9R0M1/java/lib`.

- b. Zapisz zmiany.

W pierwszym wierszu znajduje się informacja IBM z/OS Connect EE, gdzie znajduje się adapter zasobów IBM MQ.

Drugi wiersz zawiera informacje na temat adaptera zasobów produktu IBM MQ, w którym znajdują się biblioteki rodzime używane dla połączeń powiązań z produktem IBM MQ.

Więcej informacji na ten temat zawiera sekcja [KlientwmqJmsi Wdrażanie aplikacji JMS na serwerze Liberty w celu korzystania z dostawcy przesyłania komunikatów produktu IBM MQ](#).

3. Skonfiguruj bibliotekę STEPLIB serwera w celu uwzględnienia bibliotek produktu IBM MQ.

Należy to zrobić tak, aby rodzime biblioteki znajdujące się w kroku “1” na stronie 458 mogły nawiązać połączenie z menedżerem kolejek.

Zwykle dotyczy to edycji JCL, który uruchamia serwer, w celu uwzględnienia następujących elementów:

```
//STEPLIB DD DSN=HLQ.SCSQAUTH,DISP=SHR
// DD DSN=HLQ.SCSQANLE,DISP=SHR
```

gdzie `HLQ` to kwalifikator wysokiego poziomu dla zestawów danych zawierających instalację produktu IBM MQ.

Jeśli ma zostać nawiązane połączenie z wieloma wersjami produktu IBM MQ z tego samego serwera, należy użyć zestawów danych z najnowszej wersji.

Wyniki

Adapter zasobów produktu IBM MQ został częściowo zainstalowany.

Należy pamiętać, że adapter zasobów produktu IBM MQ nie zostanie w pełni zainstalowany, dopóki nie zostanie włączona jego funkcja nadrzędna (`wmqJmsClient-2.0`), zgodnie z opisem w sekcji [“Włączanie produktu z/OS Connect i produktu MQ Service Provider”](#) na stronie 460.

Co dalej

Install the MQ Service Provider into IBM z/OS Connect EE.

Zadania pokrewne

[“IBM z/OS Connect EE - MQ Service Provider setup procedure”](#) na stronie 457

Przegląd komponentów, które należy zainstalować, aby umożliwić skonfigurowanie produktu MQ Service Provider w systemie IBM z/OS Connect EE.

[“Instalowanie produktu IBM z/OS Connect EE”](#) na stronie 457

Wymagania i procedura instalowania produktu IBM z/OS Connect EE.

[“Installing the MQ Service Provider into IBM z/OS Connect EE”](#) na stronie 460

You must install the MQ Service Provider into IBM z/OS Connect EE before it is used.

Installing the MQ Service Provider into IBM z/OS Connect EE

You must install the MQ Service Provider into IBM z/OS Connect EE before it is used.

O tym zadaniu

W tym celu należy wykonać następującą procedurę:

Procedura

1. Znajdź katalog rozszerzeń produktu IBM z/OS Connect EE.
W przypadku instalacji typowej jest to katalog `/var/zosconnect/v2r0/extensions` opisany w sekcji [Konfigurowanie katalogu rozszerzeń produktu](#).
2. Skopiuj `MQSP_ROOT/mqzosconnect.properties` do katalogu znajdującego się w kroku [“1”](#) na stronie 460.
3. Edytuj skopiowany plik.
Należy zauważyć, że ten plik jest plikiem ASCII. Zmień `PATH_TO_INSTALL` na `MQSP_ROOT/v2.0` i zapisz zmiany.



Ostrzeżenie: W kroku [“3”](#) na stronie 460 założono, że ID użytkownika, na którym działa serwer, ma prawo do odczytu struktury katalogów `MQSP_ROOT`. Jeśli tak nie jest, należy dodać odpowiedni dostęp do identyfikatora użytkownika lub przenieść zawartość produktu `MQSP_ROOT` do miejsca, w którym znajduje się odpowiedni dostęp.

Wyniki

Zostały zainstalowane wszystkie wymagania wstępne dla MQ Service Provider.

Co dalej

Teraz konieczne jest włączenie opcji MQ Service Provider i z/OS Connect.

Zadania pokrewne

[“IBM z/OS Connect EE - MQ Service Provider setup procedure”](#) na stronie 457

Przegląd komponentów, które należy zainstalować, aby umożliwić skonfigurowanie produktu MQ Service Provider w systemie IBM z/OS Connect EE.

[“Instalowanie adaptera zasobów produktu IBM MQ”](#) na stronie 458

Funkcja dostawcy przesyłania komunikatów produktu IBM MQ w produkcie IBM z/OS Connect EE korzysta z komponentu IBM MQ zwanego adapterem zasobów produktu IBM MQ. Adapter zasobów jest dostarczany razem z produktem IBM MQ w ramach składnika IBM MQ for z/OS Unix System Services Components.

[“Włączanie produktu z/OS Connect i produktu MQ Service Provider”](#) na stronie 460

Co należy zrobić, aby włączyć program z/OS Connect i produkt MQ Service Provider.

Włączanie produktu z/OS Connect i produktu MQ Service Provider

Co należy zrobić, aby włączyć program z/OS Connect i produkt MQ Service Provider.

Zanim rozpoczniesz

Upewnij się, że wykonano procedury:

- [“Instalowanie produktu IBM z/OS Connect EE”](#) na stronie 457
- [“Instalowanie adaptera zasobów produktu IBM MQ”](#) na stronie 458
- [“Installing the MQ Service Provider into IBM z/OS Connect EE”](#) na stronie 460

O tym zadaniu

Ta procedura umożliwia zarówno produkt z/OS Connect, jak i produkt MQ Service Provider.

Procedura

1. Przeprowadź edycję `server.xml` utworzonego serwera Connect EE produktu z/OS i zastąp cały element **featureManager** następującymi liniami:

```
<featureManager>
  <feature>zosconnect:zosconnect-2.0</feature>
  <feature>appSecurity-2.0</feature>
  <feature>jms-2.0</feature>
  <feature>mqzosconnect:zosConnectMQ-2.0</feature>
  <feature>wmqJmsClient-2.0</feature>
  <feature>zosTransaction-1.0</feature>
</featureManager>
```



Ostrzeżenie: Zmień te wpisy, tylko jeśli jeszcze nie istnieją.

2. Skonfiguruj zabezpieczenia produktu z/OS Connect.
Szczegółowe informacje na temat sposobu przeprowadzania tej procedury można znaleźć w sekcji [Konfigurowanie zabezpieczeń dla produktu z/OS Connect EE](#).
3. Uruchom serwer.

Co dalej

Sprawdź, czy program z/OS Connect jest poprawnie skonfigurowany.

Zadania pokrewne

“IBM z/OS Connect EE - MQ Service Provider setup procedure” na stronie 457

Przegląd komponentów, które należy zainstalować, aby umożliwić skonfigurowanie produktu MQ Service Provider w systemie IBM z/OS Connect EE.

“Sprawdzanie, czy program z/OS Connect jest poprawnie skonfigurowany” na stronie 461

Sposób sprawdzania, czy program z/OS Connect jest poprawnie skonfigurowany.

Sprawdzanie, czy program z/OS Connect jest poprawnie skonfigurowany

Sposób sprawdzania, czy program z/OS Connect jest poprawnie skonfigurowany.

Zanim rozpoczniesz

Upewnij się, że procedura została wykonana szczegółowo w podręczniku [“Włączanie produktu z/OS Connect i produktu MQ Service Provider”](#) na stronie 460.

O tym zadaniu

Produkt z/OS Connect udostępnia interfejs API zgodny ze specyfikacją REST, który może być używany do wysyłania zapytań dotyczących zainstalowanych usług i wykonywania działań administracyjnych, takich jak zatrzymywanie i uruchamianie tych usług.

Procedura

1. Wydadaj komendę HTTP GET dla produktu z/OS Connect, aby wysłać zapytanie do listy aktualnie zainstalowanych usług.
W tym celu należy użyć przeglądarki WWW, wprowadzając adres URL formularza.

```
https://HOST_NAME:HTTPS_PORT/zosConnect/services
```

gdzie `NAZWA_HOSTA` i `PORT_HTTPS_PORT` to wartości wprowadzone w kroku "1" na stronie 457 produktu "Instalowanie produktu IBM z/OS Connect EE" na stronie 457.

Na przykład składnia

```
https://yourdomainname:12342/zosConnect/services
```

2. Wprowadź ID użytkownika i hasło po wyświetleniu zapytania przez przeglądarkę.

Są to wartości wprowadzone w elemencie użytkownika w kroku "2" na stronie 461 produktu "Włączanie produktu z/OS Connect i produktu MQ Service Provider" na stronie 460.

Wyniki

Wynikiem tego jest następująca odpowiedź JSON wskazująca, że program z/OS Connect jest uruchomiony, ale nie ma zainstalowanych żadnych usług. Należy zwrócić uwagę, że jeśli używany jest istniejący serwer z/OS Connect, który miał zainstalowane usługi, to zostaną wyświetlone te usługi.

```
-----  
{  
  "zosConnectServices": [  ]  
}
```

Co dalej

[Skonfiguruj prostą usługę IBM MQ w programie z/OS Connect EE.](#)

Zadania pokrewne

["IBM z/OS Connect EE - MQ Service Provider setup procedure" na stronie 457](#)

Przegląd komponentów, które należy zainstalować, aby umożliwić skonfigurowanie produktu MQ Service Provider w systemie IBM z/OS Connect EE.

Konfigurowanie prostego sposobu obsługi usługi MQ Service Provider w produkcie z/OS Connect EE

Aby skonfigurować prostą usługę MQ Service Provider jednokierunkową, należy wykonać następujące kroki.

Zanim rozpoczniesz

Upewnij się, że wszystkie komponenty zostały poprawnie skonfigurowane, zgodnie z opisem w sekcji [Włączanie funkcji z/OS Connect i MQ Service Provider](#) oraz [Sprawdzanie, czy produkt z/OS Connect jest poprawnie skonfigurowany](#)

Procedura

1. Utwórz kolejkę o nazwie `ONE_WAY_QUEUE` w docelowym menedżerze kolejek produktu z/OS, używając komend `MQSC` lub `IBM MQ Explorer`.
2. Zdefiniuj fabrykę połączeń dostawcy przesyłania komunikatów produktu IBM MQ i kolejkę.

W tym celu należy dodać następujące elementy (w dolnej części `server.xml`) serwera z/OS Connect EE, który został utworzony, ale wewnątrz elementu serwera.

```
-----  
<jmsConnectionFactory id="cf1" jndiName="jms/cf1" connectionManagerRef="ConMgr1">  
  <properties.wmqJms  
    transportType="BINDINGS"  
    queueManager="MQ21" />  
</jmsConnectionFactory>  
  
<connectionManager id="ConMgr1" maxPoolSize="5" />
```

```
<jmsQueue id="q1" jndiName="jms/d1">
  <properties.wmqJms
    baseQueueName="ONE_WAY_QUEUE"/>
</jmsQueue>
```

Uwagi:

- a. Zmień wartość atrybutu **queueManager** na poprawną docelową nazwę menedżera kolejek.
- b. Używany jest **transportType** powiązań. Oznacza, że do komunikacji z menedżerem kolejek używany jest połączenie międzypamięciowe. Jest to jedyna obsługiwana **transportType** podczas korzystania z MQ Service Provider.
3. Zdefiniuj prostą usługę jednokierunkową IBM MQ , dodając następujące elementy do `server.xml` serwera z/OS Connect EE, który został utworzony, ale wewnątrz elementu serwera.

```
<zosconnect_zosConnectService id="zosconnMQ1"
  invokeURI="/oneWay"
  serviceName="oneWay"
  serviceRef="oneWay" />

<mqzosconnect_mqzOSConnectService id="oneWay "
  connectionFactory="jms/cf1"
  destination="jms/d1"/>
```

Element `zosConnectService` definiuje nową usługę dla programu z/OS Connect za pomocą konsoli **serviceName** `oneWay`. W:

- Atrybut **invokeURI** ułatwia wywoływanie usługi.
- Atrybut **serviceRef** musi być zgodny z atrybutem identyfikatora dostawcy usług produktu z/OS Connect, który w tym przypadku jest udostępniany przez element `mqzOSConnectService`.

Element `mqzOSConnectService` definiuje pojedynczą instancję usługi udostępnianej przez MQ Service Provider.

Atrybuty **connectionFactory** i **destination** zawierają informacje o tym, w jaki sposób można znaleźć fabrykę połączeń dostawcy przesyłania komunikatów produktu IBM MQ oraz kolejkę.

Szczegółowe informacje na temat atrybutów w tej strukturze można znaleźć w sekcji [ElementmqzOSConnectService](#).

Wyniki


Ustawiłeś prostą usługę jednokierunkową.

Co dalej

Należy przetestować usługę.

Zadania pokrewne

“Testowanie usługi jednokierunkowej w produkcie z/OS Connect EE” na stronie 463
Seria kroków, które zapewniają pracę w jednokierunkowy sposób.

 *Testowanie usługi jednokierunkowej w produkcie z/OS Connect EE*
Seria kroków, które zapewniają pracę w jednokierunkowy sposób.

Zanim rozpoczniesz

Upewnij się, że pomyślnie ukończono “[Konfigurowanie prostego sposobu obsługi usługi MQ Service Provider w produkcie z/OS Connect EE](#)” na stronie 462.

Procedura

1. Sprawdź, czy program z/OS Connect rozpoznaje nową usługę.

W tym celu należy ponownie wykonać kroki opisane w sekcji [Sprawdzanie, czy produkt z/OS Connect jest poprawnie skonfigurowany](#).

Ponieważ istnieje już zdefiniowana usługa, należy zobaczyć coś podobnego do następującego:

```
-----  
{  
  "zosConnectServices": [  
    {  
      "ServiceName": "oneWay",  
      "ServiceDescription": "DATA_UNAVAILABLE",  
      "ServiceProvider": "IBM MQ for z/OS service provider for IBM z/OS Connect EE V2.0",  
      "ServiceURL": "https://yourdomainname:12342/zosConnect/services/oneWay"  
    }  
  ]  
}
```

Uwaga: Wartość **ServiceDescription** jest zwracana jako DATA_UNAVAILABLE, ponieważ atrybut **serviceDescription** nie został określony w elemencie zosConnectService.

2. Wprowadź żądanie HTTP POST do usługi. Spowoduje to wysłanie nowego komunikatu do kolejki ONE_WAY_QUEUE z tym, który ładunek został określony w żądaniu POST.

Dobrym sposobem wykonania tego rodzaju testów jest użycie czegoś takiego, jak wtyczka [Zaawansowana klient REST \(ARC\)](#) w przeglądarce Google Chrome.

- a) Użyj ARC, aby wprowadzić żądanie HTTP POST dla następującego adresu URL:

```
https://HOST_NAME:HTTPS_PORT/oneWay
```

- b) Określ poprawny format JSON dla treści, na przykład:

```
{"name1" : "value1", "name2" : "value2"}
```

Należy określić dwa nagłówki:

- i) Autoryzacja = Basic *ENCODED_USERID_PASSWORD*

gdzie *ENCODED_USERID_PASSWORD* jest zakodowanym identyfikatorem użytkownika i hasłem zakodowanym w wersji 64 zgodnie z opisem w sekcji [Podstawowy schemat uwierzytelniania](#).

- ii) Content-Type = application/json

Pierwszy nagłówek jest używany do logowania się do produktu z/OS Connect i jest zakodowanym formularzem identyfikatora użytkownika i hasła używanym w sekcji [Włączanie programu z/OS Connect](#) i serwera MQ Service Provider.

Drugi nagłówek informuje serwer o tym, że ładunek jest obiektem JSON, który jest jedyną obsługiwaną wartością podczas wysyłania wywołań HTTP POST do MQ Service Provider.

Jeśli ten nagłówek nie zostanie określony, zostanie zgłoszony błąd HTTP 415, który wskazuje nieobsługiwane wyniki typu nośnika.

Wysłanie żądania HTTP POST do jednokierunkowej usługi powoduje, że zwracany jest pusty treść odpowiedzi HTTP i kod odpowiedzi HTTP 204 (bez treści).

ARC generuje komunikat informujący o tym, że odpowiedź nie zawiera żadnych danych.

Aby sprawdzić, czy komunikat został umieszczony, użyj narzędzia, takiego jak Eksplorator IBM MQ, w celu przeglądania zawartości kolejki ONE_WAY_QUEUE. W kolejce powinien być wyświetlany komunikat IBM MQ o formacie MQSTR z ładunkiem JSON, który został wysłany.

3. Użyj ARC, aby wydać komendę HTTP DELETE dla następującego adresu URL:

```
https://HOST_NAME:HTTPS_PORT/oneWay
```


To niszczy przekaz, który został wysłany do usługi w jedną stronę.

Powinno to dać treść odpowiedzi, która została pierwotnie wysłana. Za pomocą programu IBM MQ Explorer można potwierdzić, że komunikat został zniszczony w sposób destruktywny.

Zadania pokrewne

[“Konfigurowanie prostego sposobu obsługi usługi MQ Service Provider w produkcie z/OS Connect EE” na stronie 462](#)

Aby skonfigurować prostą usługę MQ Service Provider jednokierunkową, należy wykonać następujące kroki.

Konfigurowanie dwóch prostych sposobów usługi produktu IBM MQ w produkcie z/OS Connect EE

W jaki sposób można skonfigurować dwukierunkową usługę IBM MQ w programie z/OS Connect EE, na podstawie istniejącej pary kolejek produktu IBM MQ o nazwach MQ_REQUEST_Q i MQ_REPLY_Q.

Zanim rozpocznie

Upewnij się, że wszystkie komponenty zostały poprawnie skonfigurowane zgodnie z opisem w sekcji [“Włączanie produktu z/OS Connect i produktu MQ Service Provider” na stronie 472](#) i [“Sprawdzanie, czy program z/OS Connect jest poprawnie skonfigurowany” na stronie 473](#).

O tym zadaniu

Jest to nieco bardziej związane z konfigurowaniem usługi jednokierunkowej, ponieważ wymaga aplikacji zaplecza do korzystania z komunikatów żądań wysyłanych za pomocą programu z/OS Connect i generowania komunikatów odpowiedzi.

W tym zadaniu przyjęto założenie, że kolejki MQ_REQUEST_Q i MQ_REPLY_Q są używane przez istniejącą aplikację zaplecza, na przykład transakcję CICS lub IMS, która pobiera komunikat żądania z MQ_REQUEST_Q i generuje komunikat odpowiedzi umieszczany w komendzie MQ_REPLY_Q.

Procedura

1. Zdefiniuj fabrykę połączeń dostawcy przesyłania komunikatów produktu IBM MQ oraz dwie kolejki dostawcy przesyłania komunikatów produktu IBM MQ na serwerze z/OS Connect EE.

Kolejki dostawcy przesyłania komunikatów produktu IBM MQ są odwzorowywać na istniejące kolejki MQ_REQUEST_Q i MQ_REPLY_Q. W tym celu należy dodać następujące elementy (w dolnej części `server.xml`) serwera z/OS Connect EE, który został utworzony, ale wewnątrz elementu serwera.

```
<jmsConnectionFactory id="cf2" jndiName="jms/cf2"
  connectionManagerRef="ConMgr2">
  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21"/>
</jmsConnectionFactory>

<connectionManager id="ConMgr2" maxPoolSize="5"/>

<jmsQueue id="requestQueue" jndiName="jms/requestQueue">
  <properties.wmqJms baseQueueName="MQ_REQUEST_Q"/>
</jmsQueue>

<jmsQueue id="replyQueue" jndiName="jms/replyQueue">
  <properties.wmqJms baseQueueName="MQ_REPLY_Q"/>
</jmsQueue>
```

Uwagi:

- a. Zmień wartość atrybutu `queueManager` na poprawną docelową nazwę menedżera kolejek.

- b. Może być konieczne przeprowadzenie dalszej konfiguracji tych elementów w zależności od rodzaju aplikacji zaplecza. Więcej informacji na ten temat zawiera sekcja [“Konfigurowanie MQ Service Provider”](#) na stronie 478 .
 - c. Może być konieczne dostosowanie atrybutu **maxPoolSize** w zależności od liczby współbieżnych żądań. Więcej informacji na temat tego atrybutu zawiera sekcja [Konfigurowanie zestawiania połączeń dla połączeń JMS](#).
2. Zdefiniuj dwukierunkową usługę IBM MQ , dodając następujące elementy do `server.xml` serwera z/OS Connect EE, który został utworzony, ale wewnątrz elementu serwera.

```

-----
<zoscconnect_zosConnectService id="zosconnMQ2"
    invokeURI="/twoWay"
    serviceName="twoWay"
    serviceRef="twoWay" />

<mqzosconnect_mqzOSConnectService id="twoWay"
    connectionFactory="jms/cf2"
    destination="jms/requestQueue"
    replyDestination="jms/replyQueue"
    waitInterval="10000" />
-----

```

Uwagi:

- a. Może być konieczne przeprowadzenie dalszej konfiguracji tych elementów w zależności od rodzaju aplikacji zaplecza. Na przykład może być potrzebna transformacja danych. Więcej informacji na ten temat zawiera sekcja [“Konfigurowanie MQ Service Provider”](#) na stronie 478 .
- b. Może być konieczne dostosowanie wartości **waitInterval**. Więcej informacji na ten temat zawiera sekcja [waitInterval](#) .
- c. W tym przykładzie założono, że komunikaty umieszczone w miejscu docelowym odpowiedzi przez aplikację zaplecza mają identyfikator korelacji, który jest taki sam, jak identyfikator komunikatu w miejscu docelowym żądania.

Jeśli tak się nie stanie, należy dodać atrybut **replySelection** do elementu `mqzosconnect_mqzOSConnectService` i odpowiednio ustawić wartość.

Więcej informacji na ten temat zawiera sekcja [replySelection](#) .

Wyniki

W programie z/OS Connect EE została ustawiona usługa dwukierunkowa, która może być używana z istniejącą aplikacją zaplecza, która korzysta z pary kolejek produktu IBM MQ .

Co dalej

Można wywołać dwukierunkową usługę, wydając komendę HTTP POST na adres URL `https://host_name:port_no/twoWay`.

Kroki, które należy wykonać, są podobne do kroków opisanych w sekcji [“Testowanie usługi jednokierunkowej w produkcie z/OS Connect EE”](#) na stronie 463, z tym wyjątkiem, że obsługiwane jest tylko komendy HTTP POST i należy wysłać dane ładunku JSON, które są odpowiednie dla istniejącej aplikacji zaplecza.

Zadania pokrewne

[“Konfigurowanie prostego sposobu obsługi usługi MQ Service Provider w produkcie z/OS Connect EE”](#) na stronie 462

Aby skonfigurować prostą usługę MQ Service Provider jednokierunkową, należy wykonać następujące kroki.

Generowanie pliku archiwum usługi (SAR)

Przed utworzeniem interfejsu API za pomocą programu z/OS Connect EE należy wygenerować plik SAR.

O tym zadaniu

Można to osiągnąć za pomocą programu IBM MQ, korzystając z procesu CICS-WOLA opisanego w informacjach dotyczących produktu *z/OS Connect EE* w produkcie IBM Documentation. Więcej informacji na ten temat zawiera sekcja [Generowanie archiwum usług z struktury copybook programu CICS w języku COBOL](#).

Ważne: Proces opisany w poniższym tekście jest opcjonalny. Jeśli nie chcesz korzystać z udogodnień udostępnianych przez edytor API, to nie potrzebujesz pliku SAR. Można jednak użyć schematów JSON i plików powiązań opisanych w poprzednim odsyłaczu, aby korzystać z obsługi transformacji danych udostępnianej przez produkt *z/OS Connect EE*.

Proces opisany w poprzednim odsyłaczu korzysta z programu narzędziowego **BAQLS2JS** udostępnianego razem z programem *z/OS Connect EE*.

Ten program narzędziowy:

1. Generuje schematy JSON i pliki powiązań, które mogą być używane z funkcją transformacji danych udostępnianą w produkcie *z/OS Connect EE*.

Aby skonfigurować te pliki w programie *z/OS Connect EE*, należy ustawić je w konfiguracji za pomocą elementu `zosconnect_zosConnectDataXform`.

Więcej informacji na ten temat zawiera sekcja [“Korzystanie z transformacji danych za pomocą MQ Service Provider”](#) na stronie 482.

2. Generuje plik SAR zawierający schematy JSON z poprzedniego kroku oraz odwołanie do atrybutu **serviceName** konkretnej instancji usługi Connect EE produktu *z/OS*.

Odwołanie jest dostarczane do produktu **BAQLS2JS** przy użyciu parametru SERVICE-NAME.

Na przykład, jeśli atrybut SERVICE-NAME=MQ1WayBackend jest ustawiony jako atrybut na wartość **BAQLS2JS**, produkt **BAQLS2JS** wygeneruje SAR, który może być używany z instancją usługi IBM MQ skonfigurowaną w następujący sposób:

```
<zosconnect_zosConnectService id="MQ1WayService"
  serviceName="MQ1WayBackend"
  serviceRef="mq1way"
  invokeURI="/mq1way"
  dataXformRef="xformJSON2Byte"/>

<mqzosconnect_mqzOSConnectService id="mq1way"
  connectionFactory="jms/cf1"
  destination="jms/oneWayRequestQ"/>
```

Wyniki

Po wygenerowaniu plik SAR może zostać zaimportowany do edytora API w celu wygenerowania interfejsu API. Więcej informacji na ten temat zawiera sekcja [Tworzenie interfejsu API za pomocą edytora API produktu z/OS Connect EE](#).

Po utworzeniu interfejsu API w edytorze interfejsów API generowane jest archiwum API (API Archive-AAR), które następnie można wdrożyć w programie *z/OS Connect EE*, gdzie będzie działać w odniesieniu do usługi, do której odwołuje się usługa-NAME przypisywana w produkcie **BAQLS2JS**. Więcej informacji na ten temat zawiera sekcja [Wdrażanie interfejsu API na serwerze z/OS Connect EE](#).

Zadania pokrewne

[“Konfigurowanie prostego sposobu obsługi usługi MQ Service Provider w produkcie z/OS Connect EE”](#) na stronie 462

Aby skonfigurować prostą usługę MQ Service Provider jednokierunkową, należy wykonać następujące kroki.

[“Konfigurowanie dwóch prostych sposobów usługi produktu IBM MQ w produkcie z/OS Connect EE”](#) na stronie 465

W jaki sposób można skonfigurować dwukierunkową usługę IBM MQ w programie z/OS Connect EE, na podstawie istniejącej pary kolejek produktu IBM MQ o nazwach MQ_REQUEST_Q i MQ_REPLY_Q.

IBM z/OS Connect V1 - procedura konfiguracji MQ Service Provider

Przegląd komponentów, które należy zainstalować, aby umożliwić skonfigurowanie produktu MQ Service Provider na serwerze IBM z/OS Connect V1 .

O tym zadaniu

Wykonaj następujące czynności, w kolejności pokazanej w celu zainstalowania produktu MQ Service Provider i wszystkich jego wymagań wstępnych.

instalowanie WebSphere Application Server Liberty

Wymagania i procedura instalowania produktu WebSphere Application Server Liberty (WLP).

Zanim rozpoczniesz

Upewnij się, że masz dostęp do wersji produktu WLP dla produktu z/OS , która zawiera produkt z/OS Connect, na przykład 16.0.0.2.

O tym zadaniu

Ta procedura służy do instalowania produktu WLP i konfigurowania serwera.

Procedura

1. Zainstaluj produkt WLP, postępując zgodnie z procedurą szczegółową w sekcji [Instalowanie serwera Liberty za pomocą programu Installation Manager](#).
Należy zauważyć, że w pozostałej części instrukcji struktura katalogów jest określana jako *WLP_ROOT*.
2. Utwórz nowy serwer, postępując zgodnie z procedurą szczegółową w sekcji [Ręczne tworzenie serwera Liberty](#) .

Zapoznaj się z opisem w sekcji [Konfigurowanie produktu z/OS Connect](#) , która opisuje sposób włączania produktu z/OS Connect na utworzonym serwerze.



Ostrzeżenie: Nie ma potrzeby wykonywania opcjonalnych kroków programu *WebSphere zoptymalizowane adaptory lokalne (WOLA)* .

3. Włącz autoryzowaną usługę TXRRS, wykonując procedurę szczegółową w sekcji [Włączanie autoryzowanych usług systemu z/OS na serwerze Liberty dla systemu z/OS](#) .
4. Sprawdź, czy autoryzowana usługa TXRRS została poprawnie skonfigurowana przez:
 - a) Uruchamianie serwera.
Postępuj zgodnie z procedurą w sekcji [Uruchamianie i zatrzymywanie serwera z poziomu wiersza komend](#)
 - b) Wyświetl dzienniki serwera, które znajdują się pod adresem:

```
WLP_ROOT/usr/servers/server_name/logs/message.log
```

Są to pliki ASCII, które powinny zawierać dane wyjściowe podobne do poniższych:

```
A CWWKE0001I: The server server1 has been launched.
I CWWKB0103I: Authorized service group LOCALCOM is available.
I CWWKB0103I: Authorized service group PRODMGR is available.
I CWWKB0103I: Authorized service group SAFCREd is available.
I CWWKB0103I: Authorized service group TXRRS is available.
I CWWKB0103I: Authorized service group WOLA is available.
```

```
I CWWKB0103I: Authorized service group ZOSDUMP is available.
I CWWKB0103I: Authorized service group ZOSWLM is available.
I CWWKB0103I: Authorized service group CLIENT.WOLA is available.
I CWWKB0108I: IBM CORP product WAS FOR Z/OS version 16.0 successfully registered with z/OS.
```

Sprawdź dane wyjściowe, aby sprawdzić, czy autoryzowana grupa usług TXRRS jest dostępna. W powyższym przykładzie wiersz tekstu pogrubionego pokazuje, że grupa usług jest dostępna.

Serwer MQ Service Provider wymaga, aby ta grupa usług była dostępna.

c) Zatrzymywanie serwera.

Postępuj zgodnie z procedurą w sekcji [Uruchamianie i zatrzymywanie serwera z poziomu wiersza komend](#)

Wyniki

Pakiet WLP został pomyślnie zainstalowany.

Co dalej

Zainstaluj adapter zasobów produktu IBM MQ .

Zadania pokrewne

[“IBM z/OS Connect V1 -procedura konfiguracji MQ Service Provider” na stronie 468](#)

Przegląd komponentów, które należy zainstalować, aby umożliwić skonfigurowanie produktu MQ Service Provider na serwerze IBM z/OS Connect V1 .

[“Instalowanie adaptera zasobów produktu IBM MQ” na stronie 469](#)

Funkcja dostawcy przesyłania komunikatów produktu IBM MQ w produkcie WLP korzysta z komponentu IBM MQ zwanego adapterem zasobów produktu IBM MQ . Ze względu na kompatybilność użytkownicy produktu z/OS Connect V1 muszą korzystać z adaptera zasobów produktu IBM WebSphere MQ 7.5 , niezależnie od wersji menedżera kolejek, z którym łączy się on.

Instalowanie adaptera zasobów produktu IBM MQ

Funkcja dostawcy przesyłania komunikatów produktu IBM MQ w produkcie WLP korzysta z komponentu IBM MQ zwanego adapterem zasobów produktu IBM MQ . Ze względu na kompatybilność użytkownicy produktu z/OS Connect V1 muszą korzystać z adaptera zasobów produktu IBM WebSphere MQ 7.5 , niezależnie od wersji menedżera kolejek, z którym łączy się on.

Zanim rozpocznie

Przejdź do sekcji [Uzyskiwanie adaptera zasobów IBM MQ dla produktu WLP](#), który zawiera opis odsyłacza do serwisu Fix Centrali pobierz najnowszy adapter zasobów.

Adapter zasobów będzie miał postać: `7.5.0.Fix_level-WS-MQ-Java-InstallRA.jar`. Uwaga, to jest ten plik jar, do którego odnosi się procedura.

O tym zadaniu

Ta procedura służy do instalowania adaptera zasobów produktu IBM MQ w produkcie WLP.

Procedura

- Wyodrębnij zawartość pliku JAR, przejdź do katalogu `wmq` i Prześlij plik `wmq.jmsra.rar` w pliku binarnym do odpowiedniego katalogu w docelowym systemie plików.
W dalszej części tych instrukcji katalog ten jest określany jako `MQRA_ROOT` .
- Znajdź katalog komponentów usług systemowych Unix w docelowym menedżerze kolejek.
Na przykład ten katalog może mieć nazwę `/mqm/V9R0M1/`. Katalog ten powinien zawierać podkatalog `java/lib` , który zawiera wiele bibliotek rodzimych (pliki `.so`).
- Edytuj plik `server.xml` serwera.

Dodaj następujące wiersze:

```
<variable name="wmqJmsClient.rar.location"
  value="MQRA_ROOT/wmq.jmsra.rar"/>
<wmqJmsClient nativeLibraryPath="MQJAVA_LIB_DIR" />
```

gdzie *MQJAVA_LIB_DIR* to katalog znajdujący się w kroku “2” na stronie 469, na przykład `/usr/lpp/mqm/V9R0M1/java/lib`.

b. Zapisz zmiany.

W pierwszym wierszu znajduje się *WLP*, gdzie znajduje się adapter zasobów IBM MQ .

Drugi wiersz zawiera informacje na temat adaptera zasobów produktu IBM MQ , w którym znajdują się biblioteki rodzime używane dla połączeń powiązań z produktem IBM MQ .

Więcej informacji na ten temat zawiera sekcja [KlientwmqJmsi Wdrażanie aplikacji JMS na serwerze Liberty w celu korzystania z dostawcy przesyłania komunikatów produktu IBM MQ](#) .

4. Skonfiguruj STEPLIB.

Należy to zrobić tak, aby rodzime biblioteki znajdujące się w kroku “2” na stronie 469 mogły nawiązać połączenie z menedżerem kolejek.

Uruchom następującą komendę w powłoce użytej do uruchomienia serwera *WLP*:

```
export STEPLIB=HLQ.SCSQAUTH:HLQ.SCSQANLE
```

gdzie *HLQ* to kwalifikator wysokiego poziomu dla zestawów danych zawierających instalację produktu IBM MQ .

Wyniki

Adapter zasobów produktu IBM MQ został częściowo zainstalowany. Należy pamiętać, że adapter zasobów produktu IBM MQ nie zostanie w pełni zainstalowany, dopóki nie zostanie włączona jego funkcja nadrzędna (*wmqJmsClient-1.1*).

Co dalej

Zainstaluj produkt MQ Service Provider w produkcie *WLP*.

Zadania pokrewne

“[IBM z/OS Connect V1 -procedura konfiguracji MQ Service Provider](#)” na stronie 468

Przegląd komponentów, które należy zainstalować, aby umożliwić skonfigurowanie produktu MQ Service Provider na serwerze IBM z/OS Connect V1 .

“[instalowanieWebSphere Application Server Liberty](#)” na stronie 468

Wymagania i procedura instalowania produktu WebSphere Application Server Liberty (*WLP*).

“[Instalowanie produktu MQ Service Provider w produkcie WLP](#)” na stronie 470

Produkt MQ Service Provider należy zainstalować w produkcie *WLP*, zanim zostanie on użyty.

 **Instalowanie produktu MQ Service Provider w produkcie WLP**

Produkt MQ Service Provider należy zainstalować w produkcie *WLP*, zanim zostanie on użyty.

Zanim rozpocznie

Uzyskaj i zainstaluj produkt MQ Service Provider, korzystając z informacji zawartych w produkcie [“Uzyskiwanie MQ Service Provider”](#) na stronie 455.

O tym zadaniu

Produkt MQ Service Provider jest standardowym składnikiem produktu WLP, który można zainstalować przy użyciu jednej z metod opisanych w sekcji [Pakowanie i instalowanie składników Liberty](#).

Informacje na temat wartości `#{wlp.install.dir}` i `#{wlp.user.dir}` można znaleźć w sekcji [Położenie katalogu i właściwości](#).

W poniższym tekście przedstawiono przegląd metod.

Procedura

1. Zainstaluj składnik w jądrze WLP.



Ostrzeżenie: Ta metoda wymaga dostępu do zapisu do katalogu instalacyjnego produktu WLP. Jeśli użytkownik nie ma uprawnień do zapisu w katalogu instalacyjnym produktu WLP, opcja “2” na stronie 471 może być bardziej odpowiednia.

W tym celu należy skopiować:

- a) `MQSP_ROOT/v1.0/lib/com.ibm.mq.zosconnect_1.0.0.jar` do `#{wlp.install.dir}/lib`.
 - b) `MQSP_ROOT/v1.0/lib/features/zosConnectMQ-1.0.mf` do `#{wlp.install.dir}/lib/features`.
2. Zainstaluj składnik w konfiguracji użytkownika.
Aby wykonać tę kopię:
 - a) `MQSP_ROOT/v1.0/lib/com.ibm.mq.zosconnect_1.0.0.jar` do `{wlp.user.dir}/extension/lib`.
 - b) `MQSP_ROOT/v1.0/lib/features/zosConnectMQ-1.0.mf` do `{wlp.user.dir}/extension/lib/features`.
 3. Zainstaluj składnik jako rozszerzenie produktu.

W tym celu:

- a) Skopiuj `MQSP_ROOT/mqzosconnect.properties` do `#{wlp.install.dir}/etc/extensions`.
- b) Edytuj skopiowany plik.
Należy zauważyć, że ten plik jest plikiem ASCII. Zmień `PATH_TO_INSTALL` na `MQSP_ROOT/v1.0` i zapisz zmiany.

Wyniki

Zostały zainstalowane wszystkie wymagania wstępne dla MQ Service Provider.

Uwaga: Podejście używane do instalowania produktu MQ Service Provider wpływa na sposób, w jaki jest on przywołany w produkcie `server.xml`. Przykłady są przedstawione w sekcji [“Włączanie produktu z/OS Connect i produktu MQ Service Provider”](#) na stronie 472.

Co dalej

Teraz konieczne jest włączenie opcji MQ Service Provider i z/OS Connect.

Zadania pokrewne

[“IBM z/OS Connect V1 -procedura konfiguracji MQ Service Provider”](#) na stronie 468

Przegląd komponentów, które należy zainstalować, aby umożliwić skonfigurowanie produktu MQ Service Provider na serwerze IBM z/OS Connect V1.

[“Instalowanie adaptera zasobów produktu IBM MQ”](#) na stronie 469

Funkcja dostawcy przesyłania komunikatów produktu IBM MQ w produkcie WLP korzysta z komponentu IBM MQ zwanego adapterem zasobów produktu IBM MQ. Ze względu na kompatybilność użytkownicy

produktu z/OS Connect V1 muszą korzystać z adaptera zasobów produktu IBM WebSphere MQ 7.5 , niezależnie od wersji menedżera kolejek, z którym łączy się on.

“[Włączanie produktu z/OS Connect i produktu MQ Service Provider](#)” na stronie 472
Co należy zrobić, aby włączyć program z/OS Connect i produkt MQ Service Provider.

Włączanie produktu z/OS Connect i produktu MQ Service Provider

Co należy zrobić, aby włączyć program z/OS Connect i produkt MQ Service Provider.

Zanim rozpoczniesz

Upewnij się, że wykonano procedury:

- [“instalowanie WebSphere Application Server Liberty”](#) na stronie 468
- [“Instalowanie adaptera zasobów produktu IBM MQ”](#) na stronie 469
- [“Instalowanie produktu MQ Service Provider w produkcie WLP”](#) na stronie 470

O tym zadaniu

Ta procedura umożliwi zarówno produkt z/OS Connect, jak i produkt MQ Service Provider.

Procedura

1. Zmodyfikuj utworzony `server.xml` i zastąp cały element **featureManager** tymi wierszami.



Ostrzeżenie: Dokładny format linii `<feature>zosConnectMQ-1.0</feature>` zależy od metody użytej do zainstalowania MQ Service Provider, która została opisana w [“Instalowanie produktu MQ Service Provider w produkcie WLP”](#) na stronie 470.

W przypadku użycia:

- Metoda [“1”](#) na stronie 471, linia jest wyświetlana jako `<feature>zosConnectMQ-1.0</feature>`.
- Metoda [“2”](#) na stronie 471, linia jest wyświetlana jako `<feature>usr:zosConnectMQ-1.0</feature>`.
- Metoda [“3”](#) na stronie 471, linia jest wyświetlana jako `<feature>mqzosconnect:zosConnectMQ-1.0</feature>`.

Należy również zmienić następujące wpisy, tylko jeśli jeszcze nie istnieją.

```
<featureManager>
<feature>zosConnect-1.0</feature>
<feature>appSecurity-2.0</feature>
<feature>zosConnectMQ-1.0</feature>
<feature>wmqJmsClient-1.1</feature>
<feature>zosTransaction-1.0</feature>
</featureManager>
```

2. Skonfiguruj zabezpieczenia produktu z/OS Connect.

Szczegółowe informacje na temat sposobu przeprowadzania tej procedury można znaleźć w sekcji [Konfigurowanie zabezpieczeń dla produktu z/OS Connect](#).

3. Uruchom serwer.

Postępuj zgodnie z procedurą szczegółową w sekcji [Uruchamianie i zatrzymywanie serwera z poziomu wiersza komend](#).

Co dalej

Sprawdź, czy program z/OS Connect jest poprawnie skonfigurowany.

Zadania pokrewne

“IBM z/OS Connect V1 -procedura konfiguracji MQ Service Provider” na stronie 468

Przegląd komponentów, które należy zainstalować, aby umożliwić skonfigurowanie produktu MQ Service Provider na serwerze IBM z/OS Connect V1 .

“Sprawdzanie, czy program z/OS Connect jest poprawnie skonfigurowany” na stronie 473

Sposób sprawdzania, czy program z/OS Connect jest poprawnie skonfigurowany.

Sprawdzanie, czy program z/OS Connect jest poprawnie skonfigurowany

Sposób sprawdzania, czy program z/OS Connect jest poprawnie skonfigurowany.

Zanim rozpoczniesz

Upewnij się, że procedura została wykonana szczegółowo w podręczniku [“Włączanie produktu z/OS Connect i produktu MQ Service Provider”](#) na stronie 472 .

O tym zadaniu

Produkt z/OS Connect udostępnia interfejs API zgodny ze specyfikacją REST, który może być używany do wysyłania zapytań dotyczących zainstalowanych usług i wykonywania działań administracyjnych, takich jak zatrzymywanie i uruchamianie tych usług.

Procedura

1. Wydadź komendę HTTP GET dla produktu z/OS Connect, aby wysłać zapytanie do listy aktualnie zainstalowanych usług.

W tym celu należy użyć przeglądarki WWW, wprowadzając adres URL formularza.

```
https://HOST_NAME:HTTPS_PORT/zosConnect/services
```

gdzie *NAZWA_HOSTA* i *PORT_HTTPS_PORT* to wartości wprowadzone w kroku “2” na stronie 468 produktu [“instalowanieWebSphere Application Server Liberty”](#) na stronie 468.

Na przykład składnia

```
https://yourdomainname:12342/zosConnect/services
```

2. Wprowadź ID użytkownika i hasło po wyświetleniu zapytania przez przeglądarkę.

Są to wartości wprowadzone w elemencie użytkownika w kroku “2” na stronie 472 produktu [“Włączanie produktu z/OS Connect i produktu MQ Service Provider”](#) na stronie 472.

Wyniki

Wynikiem tego jest następująca odpowiedź JSON wskazująca, że produkt z/OS Connect jest uruchomiony, ale nie ma zainstalowanych żadnych usług:

```
-----  
{  
  "zosConnectServices": [  ]  
}
```

Co dalej

“Konfigurowanie prostej usługi o jedną stronę MQ Service Provider na serwerze z/OS Connect V1” na stronie 474

Zadania pokrewne

“IBM z/OS Connect V1 -procedura konfiguracji MQ Service Provider” na stronie 468

Przegląd komponentów, które należy zainstalować, aby umożliwić skonfigurowanie produktu MQ Service Provider na serwerze IBM z/OS Connect V1 .

Konfigurowanie prostej usługi o jedną stronę MQ Service

Provider na serwerze z/OS Connect V1

Aby skonfigurować prostą usługę MQ Service Provider jednokierunkową, należy wykonać następujące kroki.

Zanim rozpoczniesz

Upewnij się, że wszystkie komponenty zostały poprawnie skonfigurowane, zgodnie z opisem w sekcji [Włączanie funkcji z/OS Connect i MQ Service Provider](#) oraz [Sprawdzanie, czy produkt z/OS Connect jest poprawnie skonfigurowany](#)

Procedura

1. Utwórz kolejkę o nazwie ONE_WAY_QUEUE w docelowym menedżerze kolejek produktu z/OS , używając komend MQSC lub IBM MQ Explorer.
2. Zdefiniuj fabrykę połączeń dostawcy przesyłania komunikatów produktu IBM MQ i kolejkę.
W tym celu należy dodać następujące elementy w dolnej części składnika `server.xml`, ale wewnątrz elementu serwera.

```
-----  
<jmsConnectionFactory id="cf1" jndiName="jms/cf1" connectionManagerRef="ConMgr1">  
  <properties.wmqJms  
    transportType="BINDINGS"  
    queueManager="MQ21" />  
</jmsConnectionFactory>  
  
<connectionManager id="ConMgr1" maxPoolSize="5" />  
  
<jmsQueue id="q1" jndiName="jms/d1">  
  <properties.wmqJms  
    baseQueueName="ONE_WAY_QUEUE" />  
</jmsQueue>  
-----
```

Uwagi:

- a. Zmień wartość atrybutu **queueManager** na poprawną docelową nazwę menedżera kolejek.
 - b. Używany jest **transportType** powiązań. Oznacza, że do komunikacji z menedżerem kolejek używany jest połączenie międzypamięciowe. Jest to jedyna obsługiwana **transportType** podczas korzystania z MQ Service Provider.
3. Zdefiniuj prostą usługę jednokierunkową MQ Service Provider , dodając następujące elementy do składnika `server.xml`, ale wewnątrz elementu serwera.

```
-----  
<zosConnectService id="zosconnMQ1"  
  invokeURI="/oneWay"  
  serviceName="oneWay"  
  serviceRef="oneWay" />  
  
<mqzOSConnectService id="oneWay"  
  connectionFactory="jms/cf1"  
  destination="jms/d1" />  
-----
```

Element `zosConnectService` definiuje nową usługę dla programu z/OS Connect za pomocą konsoli **serviceName** `oneWay`. W:

- Atrybut **invokeURI** ułatwia wywoływanie usługi.

- Atrybut **serviceRef** musi być zgodny z atrybutem identyfikatora dostawcy usług produktu z/OS Connect, który w tym przypadku jest udostępniany przez element `mzqOSConnectService`.

Element `mzqOSConnectService` definiuje pojedynczą instancję usługi udostępnianej przez MQ Service Provider.

Atrybuty **connectionFactory** i **destination** zawierają informacje o tym, w jaki sposób można znaleźć fabrykę połączeń dostawcy przesyłania komunikatów produktu IBM MQ oraz kolejkę.

Szczegółowe informacje na temat atrybutów w tej strukturze można znaleźć w sekcji [ElementmzqOSConnectService](#).

Wyniki


Ustawiłeś prostą usługę jednokierunkową.

Co dalej

Należy przetestować usługę.

Zadania pokrewne

“[Testowanie jednokierunkowej usługi na serwerze z/OS Connect V1](#)” na stronie 475
Seria kroków, które zapewniają pracę w jednokierunkowy sposób.

 [Testowanie jednokierunkowej usługi na serwerze z/OS Connect V1](#)
Seria kroków, które zapewniają pracę w jednokierunkowy sposób.

Zanim rozpoczniesz

Upewnij się, że pomyślnie ukończono “[Konfigurowanie prostej usługi o jedną stronę MQ Service Provider na serwerze z/OS Connect V1](#)” na stronie 474.

Procedura

1. Sprawdź, czy program z/OS Connect rozpoznaje nową usługę.

W tym celu należy ponownie wykonać kroki opisane w sekcji [Sprawdzanie, czy produkt z/OS Connect jest poprawnie skonfigurowany](#).

Ponieważ istnieje już zdefiniowana usługa, należy zobaczyć coś podobnego do następującego:

```
-----
{
  "zosConnectServices": [
    {
      "ServiceName": "oneWay",
      "ServiceDescription": "DATA_UNAVAILABLE",
      "ServiceProvider": "IBM MQ for z/OS service provider for IBM z/OS Connect" V1.0,
      "ServiceURL": "https://yourdomainname:12342/zosConnect/services/oneWay"
    }
  ]
}
-----
```

Uwaga: Wartość **ServiceDescription** jest zwracana jako DATA_UNAVAILABLE, ponieważ atrybut **serviceDescription** nie został określony w elemencie `zosConnectService`.

2. Wprowadź żądanie HTTP POST do usługi. Spowoduje to wysłanie nowego komunikatu do kolejki ONE_WAY_QUEUE z tym, który ładunek został określony w żądaniu POST.

Dobrym sposobem wykonania tego rodzaju testów jest użycie czegoś takiego, jak wtyczka [Zaawansowana klient REST \(ARC\)](#) w przeglądarce Google Chrome.

- a) Użyj ARC, aby wprowadzić żądanie HTTP POST dla następującego adresu URL:

```
https://HOST_NAME:HTTPS_PORT/oneWay
```

b) Określ poprawny format JSON dla treści, na przykład:

```
{"name1" : "value1", "name2" : "value2"}
```

Należy określić dwa nagłówki:

i) Autoryzacja = Basic *ENCODED_USERID_PASSWORD*

gdzie *ENCODED_USERID_PASSWORD* jest zakodowanym identyfikatorem użytkownika i hasłem zakodowanym w wersji 64 zgodnie z opisem w sekcji [Podstawowy schemat uwierzytelniania](#).

ii) Content-Type = application/json

Pierwszy nagłówek jest używany do logowania się do programu z/OS Connect i jest zakodowanym formularzem identyfikatora użytkownika i hasła używanego w produkcie [“Włączanie produktu z/OS Connect i produktu MQ Service Provider”](#) na stronie 472.

Drugi nagłówek informuje serwer o tym, że ładunek jest obiektem JSON, który jest jedyną obsługiwaną wartością podczas wysyłania wywołań HTTP POST do MQ Service Provider.

Jeśli ten nagłówek nie zostanie określony, zostanie zgłoszony błąd HTTP 415, który wskazuje nieobsługiwane wyniki typu nośnika.

Wysłanie żądania HTTP POST do jednokierunkowej usługi powoduje, że zwracany jest pusty treść odpowiedzi HTTP i kod odpowiedzi HTTP 204 (bez treści).

ARC generuje komunikat informujący o tym, że odpowiedź nie zawiera żadnych danych.

Aby sprawdzić, czy komunikat został umieszczony, użyj narzędzia, takiego jak Eksplorator IBM MQ, w celu przeglądania zawartości kolejki ONE_WAY_QUEUE. W kolejce powinien być wyświetlany komunikat IBM MQ o formacie MQSTR z ładunkiem JSON, który został wysłany.

3. Użyj ARC, aby wydać komendę HTTP DELETE dla następującego adresu URL:

```
https://HOST_NAME:HTTPS_PORT/oneWay
```

To niszczy przekaz, który został wysłany do usługi w jedną stronę.

Powinno to dać treść odpowiedzi, która została pierwotnie wysłana. Za pomocą programu IBM MQ Explorer można potwierdzić, że komunikat został zniszczony w sposób destruktywny.

Zadania pokrewne

[“Konfigurowanie prostej usługi o jedną stronę MQ Service Provider na serwerze z/OS Connect V1”](#) na stronie 474

Aby skonfigurować prostą usługę MQ Service Provider jednokierunkową, należy wykonać następujące kroki.

Konfigurowanie dwóch prostych sposobów usługi IBM MQ w produkcie z/OS Connect w wersji 1

W jaki sposób można skonfigurować dwukierunkową usługę IBM MQ w programie z/OS Connect V1, na podstawie istniejącej pary kolejek produktu IBM MQ o nazwach MQ_REQUEST_Q i MQ_REPLY_Q.

Zanim rozpocznesz

Upewnij się, że wszystkie komponenty zostały poprawnie skonfigurowane zgodnie z opisem w sekcji [“Włączanie produktu z/OS Connect i produktu MQ Service Provider”](#) na stronie 472 i [“Sprawdzanie, czy program z/OS Connect jest poprawnie skonfigurowany”](#) na stronie 473.

O tym zadaniu

Jest to nieco bardziej związane z konfigurowaniem usługi jednokierunkowej, ponieważ wymaga aplikacji zaplecza do korzystania z komunikatów żądań wysyłanych za pomocą produktu z/OS Connect i generowania komunikatów odpowiedzi.

W tym zadaniu przyjęto założenie, że kolejki MQ_REQUEST_Q i MQ_REPLY_Q są używane przez istniejącą aplikację zaplecza, na przykład transakcję CICS lub IMS, która pobiera komunikat żądania z MQ_REQUEST_Q i generuje komunikat odpowiedzi umieszczony w komendzie MQ_REPLY_Q.

Procedura

1. Zdefiniuj fabrykę połączeń dostawcy przesyłania komunikatów produktu IBM MQ oraz dwie kolejki dostawcy przesyłania komunikatów produktu IBM MQ na serwerze z/OS Connect V1.
Kolejki dostawcy przesyłania komunikatów produktu IBM MQ są odwzorowywać na istniejące kolejki MQ_REQUEST_Q i MQ_REPLY_Q. W tym celu należy dodać następujące elementy w dolnej części składnika `server.xml`, ale wewnątrz elementu serwera.

```
<jmsConnectionFactory id="cf2" jndiName="jms/cf2"
  connectionManagerRef="ConMgr2">
  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21"/>
</jmsConnectionFactory>

<connectionManager id="ConMgr2" maxPoolSize="5"/>

<jmsQueue id="requestQueue" jndiName="jms/requestQueue">
  <properties.wmqJms baseQueueName="MQ_REQUEST_Q"/>
</jmsQueue>

<jmsQueue id="replyQueue" jndiName="jms/replyQueue">
  <properties.wmqJms baseQueueName="MQ_REPLY_Q"/>
</jmsQueue>
```

Uwagi:

- a. Zmień wartość atrybutu **queueManager** na poprawną docelową nazwę menedżera kolejek.
 - b. Może być konieczne przeprowadzenie dalszej konfiguracji tych elementów w zależności od rodzaju aplikacji zaplecza. Więcej informacji na ten temat zawiera sekcja [“Konfigurowanie MQ Service Provider”](#) na stronie 478.
 - c. Może być konieczne dostosowanie atrybutu **maxPoolSize** w zależności od liczby współbieżnych żądań. Więcej informacji na temat tego atrybutu zawiera sekcja [Konfigurowanie zestawiania połączeń dla połączeń JMS](#).
2. Zdefiniuj dwukierunkową usługę IBM MQ, dodając następujące elementy do składnika `server.xml`, ale wewnątrz elementu serwera.

```
-----
<zosConnectService id="zosconnMQ2"
  invokeURI="/twoWay"
  serviceName="twoWay"
  serviceRef="twoWay"/>

<mqz0SConnectService id="twoWay"
  connectionFactory="jms/cf2"
  destination="jms/requestQueue"
  replyDestination="jms/replyQueue"
  waitInterval="10000"/>
-----
```

Uwagi:

- a. Może być konieczne przeprowadzenie dalszej konfiguracji tych elementów w zależności od rodzaju aplikacji zaplecza. Na przykład może być potrzebna transformacja danych. Więcej informacji na ten temat zawiera sekcja [“Konfigurowanie MQ Service Provider”](#) na stronie 478.

- b. Może być konieczne dostosowanie wartości **waitInterval**. Więcej informacji na ten temat zawiera sekcja **waitInterval**.
- c. W tym przykładzie założono, że komunikaty umieszczone w miejscu docelowym odpowiedzi przez aplikację zaplecza mają identyfikator korelacji, który jest taki sam, jak identyfikator komunikatu w miejscu docelowym żądania.

Jeśli tak się nie stanie, należy dodać atrybut **replySelection** do elementu `mzqOSConnectService` i odpowiednio ustawić wartość.

Więcej informacji na ten temat zawiera sekcja **replySelection**.

Wyniki

Usługa dwukierunkowa została ustawiona w programie z/OS Connect V1, która może być używana z istniejącą aplikacją zaplecza, która korzysta z pary kolejek produktu IBM MQ.

Co dalej

Można wywołać dwukierunkową usługę, wydając komendę HTTP POST na adres URL `https://host_name:port_no/twoWay`.

Kroki, które należy wykonać, są podobne do kroków opisanych w sekcji [“Testowanie jednokierunkowej usługi na serwerze z/OS Connect V1”](#) na stronie 475, z tym wyjątkiem, że obsługiwane jest tylko komendy HTTP POST i należy wysłać dane ładunku JSON, które są odpowiednie dla istniejącej aplikacji zaplecza.

Zadania pokrewne

[“Konfigurowanie prostej usługi o jedną stronę MQ Service Provider na serwerze z/OS Connect V1”](#) na stronie 474

Aby skonfigurować prostą usługę MQ Service Provider jednokierunkową, należy wykonać następujące kroki.

Konfigurowanie MQ Service Provider

Przed użyciem produktu MQ Service Provider należy wziąć pod uwagę różne obszary, które należy wziąć pod uwagę.

W tej sekcji zawarto następujące informacje:

- [“Uwagi dotyczące zabezpieczeń w przypadku korzystania z produktu MQ Service Provider”](#) na stronie 478
- [“Korzystanie z transformacji danych za pomocą MQ Service Provider”](#) na stronie 482
- [“Praca z aplikacjami MQI”](#) na stronie 487

Uwagi dotyczące zabezpieczeń w przypadku korzystania z produktu MQ Service Provider

Istnieją dwa zestawy uwag dotyczących zabezpieczeń dla produktu MQ Service Provider.

- Określa, czy dany użytkownik może wysyłać żądania RESTful do usług produktu MQ Service Provider ujawnionych przez produkt z/OS Connect.
- Określa, czy dana usługa produktu MQ Service Provider jest autoryzowana do uzyskiwania dostępu do menedżera kolejek i jego zasobów.

Określa, czy dany użytkownik może wysyłać żądania zgodne ze specyfikacją REST do usług produktu MQ Service Provider ujawnionych za pomocą programu z/OS Connect

Jest to kontrolowane przez konfigurację w pliku `server.xml` produktu z/OS Connect.

Informacje na temat produktu z/OS Connect EE zawiera sekcja [Konfigurowanie zabezpieczeń dla produktu z/OS Connect EE](#).

W przypadku produktu z/OS Connect V1 należy zapoznać się z informacjami w sekcji [Konfigurowanie zabezpieczeń dla produktu z/OS Connect](#).

Określa, czy dana usługa produktu MQ Service Provider jest autoryzowana do uzyskiwania dostępu do menedżera kolejek i jego zasobów.

Produkt MQ Service Provider jest zasadniczo aplikacją JMS opartą na dostawcy przesyłania komunikatów produktu IBM MQ, który jest dostarczany z produktem WLP, który łączy się z jednym lub większą liczbą menedżerów kolejek za pomocą połączeń w trybie powiązań.

W rezultacie produkt MQ Service Provider może być zabezpieczony w taki sam sposób, jak aplikacje, które współużytkują te cechy. W tym temacie opisano te podobieństwa, jak również wskazują różnice.

Ponieważ aplikacja MQ Service Provider jest aplikacją powiązań, łączy się ona z menedżerem kolejek i przedstawia ID użytkownika i opcjonalne hasło. Poprawność tych elementów jest opcjonalnie sprawdzana przy użyciu opcji [Uwierzytelnianie połączenia](#).

Jeśli jest to poprawne, użytkownik może się łączyć w zależności od konfiguracji zabezpieczeń menedżera kolejek. Więcej informacji na ten temat zawiera sekcja [Konfigurowanie zabezpieczeń w systemie z/OS](#).

Atrybuty konfiguracji elementu `mzqOSConnectService` oraz podelement `properties.wmqJMS` elementu **jmsConnectFactory** w produkcie `server.xml` mają wpływ na to, który identyfikator użytkownika i opcjonalne hasło są wyświetlane w menedżerze kolejek.

Poszczególne możliwości są szczegółowo opisane w poniższej tabeli.



Ostrzeżenie: Wśród właściwości konfiguracji zabezpieczeń istnieje kolejność wykonywania operacji. Atrybuty `mzqOSConnectService` przesłaniają atrybuty `properties.wmqJMS`, a atrybut `useCallerPrincipal` przesłania wszystkie inne atrybuty.

Oba atrybuty hasła mogą być zapisane w formacie zwykłego tekstu lub w formacie zakodowanym. Należy użyć zakodowanego formatu, ponieważ każdy, kto ma dostęp do `server.xml`, może wyświetlać hasło w postaci jawnego tekstu.

Obie wersje produktu z/OS Connect są dostarczane z narzędziem o nazwie **securityUtility**, które mogą być używane do kodowania haseł. Więcej informacji na ten temat zawiera sekcja [Komenda Liberty: securityUtility](#).

Element usługi <code>mzqOSConnect</code>		Element <code>properties.wmqJMS</code>		Wynik
Nazwa użytkownika <code>useCaller</code>	Nazwa użytkownika i hasło	<code>userName</code>	Hasło	
nie set/false	nieokreślone/puste	nieokreślone/puste	nieokreślone/puste	Nazwa użytkownika powiązana z przestrzenią adresową programu z/OS Connect jest prezentowana menedżerowi kolejek w celu autoryzacji i uwierzytelniania. Nie jest wyświetlane żadne hasło.
nie set/false	nieokreślone/puste	zbiór	nieokreślone/puste	Nazwa użytkownika w elemencie <code>properties.wmqJMS</code> jest prezentowana menedżerowi kolejek w celu autoryzacji i uwierzytelniania. Nie jest wyświetlane żadne hasło.

Tabela 28. Autoryzacje usług (kontynuacja)

Element usługi mqzOSConnect		Element properties.wmqJms		Wynik
nie set/false	nieokreślone/puste	zbiór	zbiór	Nazwa użytkownika i hasło w elemencie properties.wmqJms są wyświetlane w menedźerze kolejek w celu autoryzacji i uwierzytelniania.
nie set/false	obie wartości są ustawione	ignorowana, jeśli ustawiono	ignorowana, jeśli ustawiono	Nazwa użytkownika i hasło w elemencie mqzOSConnectService są wyświetlane w menedźerze kolejek w celu autoryzacji i uwierzytelniania.
Prawda	ignorowana, jeśli ustawiono	ignorowana, jeśli ustawiono	ignorowana, jeśli ustawiono	Nazwa użytkownika, który został uwierzytelniony w programie z/OS Connect, jest podejmowana, a jego nazwa użytkownika jest prezentowana menedźerowi kolejek w celu autoryzacji i uwierzytelniania. Nie jest wyświetlane żadne hasło. Informacje na temat konfigurowania zabezpieczeń produktu z/OS Connect EE znajduje się w tym miejscu ¹ . Informacje na temat produktu z/OS Connect Version 1 znajdują się w tym miejscu: ² .

Uwagi:

1. [Konfigurowanie zabezpieczeń dla produktu z/OS Connect EE](#)
2. [Konfigurowanie zabezpieczeń dla produktu z/OS Connect](#)

Przykłady

Spowoduje to wyświetlenie jednokierunkowej usługi, w której zarówno fabryka połączeń, jak i mqzOSConnectService, określają zarówno **userID**, jak i **password**. W obu przypadkach hasła są kodowane. Używane są **userID** i **password** zdefiniowane w definicji mqzOSConnectService.

z/OS Connect V1

```
<zosConnectService
id="samplezOSConnectService1"
invokeURI="/samplezOSConnectService1"
serviceName="samplezOSConnectService1_name"
serviceRef="samplezOSConnectService1_MQ"/>

<mqzOSConnectService
id="samplezOSConnectService1_MQ"
connectionFactory="jms/sampleCF1"
userName="bill"
password="{aes}AJ+DdZ+1u0KEG5KIwUz4LvHBAQ8nTd3y8K8HAI+48Tt"
destination="jms/sampleQ1"/>

<jmsConnectionFactory
id="sampleCF1"
jndiName="jms/sampleCF1"
connectionManagerRef="sampleCF2ConnectionManager1">

<properties.wmqJms
transportType="BINDINGS"
queueManager="MQ21"
```



```

userName="matt"
password="{xor}GBMeEg9uERg=" />
</jmsConnectionFactory>

<jmsQueue
id="sampleQ1"
jndiName="jms/sampleQ1">

<properties.wmqJms
baseQueueName="SampleQ1" />
</jmsQueue>

```

z/OS Connect EE

```

<zosconnect_zosConnectService
id="samplezOSConnectService1"
invokeURI="/samplezOSConnectService1"
serviceName="samplezOSConnectService1_name"
serviceRef="samplezOSConnectService1_MQ"/>

<mqzosconnect_mqzOSConnectService
id="samplezOSConnectService1_MQ"
connectionFactory="jms/sampleCF1"
userName="bill"
password="{aes}AJ+DdZ+1u0KEG5KIwUz4LvHBAQ8nTd3y8K8HAI+48Tt"
destination="jms/sampleQ1"/>

<jmsConnectionFactory
id="sampleCF1"
jndiName="jms/sampleCF1"
connectionManagerRef="sampleCF2ConnectionManager1">

<properties.wmqJms
transportType="BINDINGS"
queueManager="MQ21"
userName="matt"
password="{xor}GBMeEg9uERg=" />
</jmsConnectionFactory>

<jmsQueue
id="sampleQ1"
jndiName="jms/sampleQ1">

<properties.wmqJms
baseQueueName="SampleQ1" />
</jmsQueue>

```

W tym przykładzie przedstawiono definicję usługi jednego sposobu, która używa tych samych obiektów JMS (kolejka sampleQ1 i fabryka połączeń sampleCF1) jako pierwszy przykład. Jednostka główna, która jest uwierzytelniona w programie z/OS Connect, jest prezentowana w menedżerze kolejek, ponieważ określono wartość **useCallerPrincipal="true"**.

z/OS Connect V1

```

<zosConnectService
id="samplezOSConnectService2"
invokeURI="/samplezOSConnectService2"
serviceName="samplezOSConnectService2_name"
serviceRef="samplezOSConnectService2_MQ"/>

<mqzOSConnectService
id="samplezOSConnectService2_MQ"
connectionFactory="jms/sampleCF1"
destination="jms/sampleQ1"
useCallerPrincipal="true"/>

```

z/OS Connect EE

```

<zosconnect_zosConnectService
id="samplezOSConnectService2"
invokeURI="/samplezOSConnectService2"
serviceName="samplezOSConnectService2_name"

```

```
serviceRef="samplezOSConnectService2_MQ"/>
```

```
<mqzosconnect_mqzOSConnectService  
id="samplezOSConnectService2_MQ"  
connectionFactory="jms/sampleCF1"  
destination="jms/sampleQ1"  
useCallerPrincipal="true"/>
```

Korzystanie z transformacji danych za pomocą MQ Service

Provider

Produkt z/OS Connect umożliwia transformowanie danych JSON w dowolny format przed wywołaniem zasobu z/OS zaplecza, takiego jak transakcja CICS, a następnie transformowanie odpowiedzi z zasobu aplikacyjnego z/OS z powrotem do formatu JSON.

Możliwość ta jest udostępniana przez dołączonych dostawców transformacji danych. Wbudowany dostawca umożliwia transformowanie obiektów JSON w struktury języka COBOL, PLI lub C lub ich transformowanie.

Produkt MQ Service Provider współpracuje z transformacjami danych programu z/OS Connect, ale istnieje kilka uwag, które należy wziąć pod uwagę.

Więcej informacji na temat transformacji danych programu z/OS Connect

Informacje na temat produktu z/OS Connect Enterprise Edition zawiera sekcja [Definiowanie transformatorów danych](#).

W przypadku produktu z/OS Connect, wersja 1, należy zapoznać się z [Definiowanie transformacji ładunku komunikatu produktu z/OS Connect](#).

Wysyłanie komunikatów do programu IBM MQ

Gdy usługa MQ Service Provider jest skonfigurowana z transformacją danych i ma zamiar wysłać komunikat do menedżera kolejek (na przykład, jeśli jest to usługa jednokierunkowa lub dwukierunkowa, która odebrała żądanie HTTP POST zawierające JSON), MQ Service Provider wykonuje następujące kroki:

1. Pobiera ładunek JSON z żądania HTTP.
2. Przekazuje ładunek do produktu z/OS Connect w celu przeprowadzenia transformacji danych z formatu JSON do tablicy bajtów.
3. Pobiera wynik transformacji danych i wysyła je do menedżera kolejek w postaci pliku JMS `BytesMessage`.

Domyślnie wysyłany komunikat ma puste pole MQMD **Format**. W wielu przypadkach jest to nieodpowiednie, aby można było ustawić odpowiednią wartość atrybutu `mqmdFormat` w elemencie `mqzOSConnectService`.

Wbudowana obsługa transformacji danych zawsze generuje dane wyjściowe o identyfikatorze CCSID 37. Informacje te należy określić w polu MQMD `CodedCharSetId`. W przeciwnym razie aplikacja pobierająca z kolejki może nie dekodować komunikatu. W tym celu należy ustawić atrybut CCSID w kolejce przywoływanej przez element usługi `mqzOSConnectService`.

W poniższej przykładowej konfiguracji przedstawiono odpowiednią konfigurację dla usługi jednokierunkowej, która będzie używana do wysyłania komunikatów do kolejki o nazwie `SampleQ1`.

Komunikaty mają być wysyłane z polem MQMD `Format` o wartości `AFORMAT` i polem `CodedCharSetId` o wartości 37.

Element `zosConnectDataXform` informuje program z/OS Connect o miejscu, w którym można znaleźć konfigurację dla transformacji danych, a następnie określa, że jest on używany z atrybutem `dataXformRef` elementu `zosConnectService`.

Informacje na temat konfiguracji dla usługi dwóch sposobów można znaleźć w sekcji [“Odbieranie komunikatów z produktu IBM MQ”](#) na stronie 483.

```
<jmsConnectionFactory
  id="sampleCF1"
  jndiName="jms/sampleCF1"
  connectionManagerRef="sampleCF1ConnectionManager">

  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21" />
</jmsConnectionFactory>

<connectionManager
  id="sampleCF1ConnectionManager"
  maxPoolSize="5" />

<jmsQueue id="sampleQ1"
  jndiName="jms/sampleQ1">

  <properties.wmqJms
    baseQueueName="SampleQ1"
    CCSID="37" />
</jmsQueue>

<zosConnectService
  id="samplezOSConnectService1"
  invokeURI="/samplezOSConnectService1"
  serviceName="samplezOSConnectService1_name"
  serviceRef="samplezOSConnectService1_MQ"
  dataXformRef="xformJSON2Byte" />

<mqzOSConnectService
  id="samplezOSConnectService1_MQ"
  connectionFactory="jms/sampleCF1"
  mqmdFormat="AFORMAT"
  destination="jms/sampleQ1" />

<zosConnectDataXform id="xformJSON2Byte"
  bindFileLoc="/XFORM_ROOT/bindfiles" bindFileSuffix=".bnd"
  requestSchemaLoc="/XFORM_ROOT/json" requestSchemaSuffix=".json"
  responseSchemaLoc="/XFORM_ROOT/json"
  responseSchemaSuffix=".json" />
```

Odbieranie komunikatów z produktu IBM MQ

Jeśli instancja produktu MQ Service Provider jest skonfigurowana z transformacją danych i ma zostać odebrany komunikat z menedżera kolejek (na przykład, jeśli jest to usługa jednokierunkowa aktywacja HTTP GET lub DELETE albo usługa dwukierunkowa, która odebrała żądanie HTTP POST), MQ Service Provider wykonuje następujące kroki.



Ostrzeżenie: Jeśli używana jest usługa dwudrożna, usługa ta będzie już wykonana kroki opisane w sekcji [“Wysyłanie komunikatów do programu IBM MQ”](#) na stronie 482.

1. Pobiera komunikat z kolejki.
2. Sprawdza, czy komunikat jest JMS **BytesMessage**, czy JMS **TextMessage**. Jeśli komunikat nie jest wyświetlany, generowany jest błąd i jest on zwracany do programu wywołującego. "
3. Przekazuje ładunek komunikatu do programu z/OS Connect w celu przeprowadzenia transformacji danych z tablicy bajtów do formatu JSON.
4. Pobiera wynik transformacji danych i zwraca ją w odpowiedzi na odpowiedź metody HTTP.

W zależności od typu odebranego komunikatu może być konieczne wykonanie dodatkowej konfiguracji. Jest to spowodowane tym, że program MQ Service Provider musi przekształcić odebrany ładunek komunikatu w poprawny format w celu przejścia do skonfigurowanej transformacji danych.

Domyślna transformacja danych oczekuje, że ładunek ma być w identyfikatorze CCSID 37, jednak zasób aplikacyjny z/OS może nie generować komunikatów w tym identyfikatorze CCSID.

Wymagana konfiguracja zależy od tego, czy odebrano **BytesMessage**, czy **TextMessage** oraz czy używany jest program z/OS Connect V1, czy z/OS Connect EE.

BytesMessage odebrane w programie z/OS Connect V1

Jeśli zostanie odebrana **BytesMessage**, można określić atrybuty **receiveConversion= "QMGR"** i **receiveCCSID= "37"** w definicji kolejki, która jest używana do odbierania komunikatu.

Przedstawiono to w poniższym przykładzie. W tym przypadku definicja `sampleQ2Receive` ma ustawione zarówno atrybuty **receiveConversion**, jak i **receiveCCSID**.

```
<jmsConnectionFactory
  id="sampleCF2"
  jndiName="jms/sampleCF2"
  connectionManagerRef="sampleCF2ConnectionManager">

  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21" />
</jmsConnectionFactory>

<connectionManager
  id="sampleCF2ConnectionManager"
  maxPoolSize="5" />

<jmsQueue id="sampleQ2Send"
  jndiName="jms/sampleQ2Send">

  <properties.wmqJms
    baseQueueName="SampleQ2Send"
    CCSID="37" />
</jmsQueue>

<jmsQueue id="sampleQ2Receive"
  jndiName="jms/sampleQ2Receive">

  <properties.wmqJms
    baseQueueName="SampleQ2Receive"
    receiveCCSID="37"
    receiveConversion="QMGR" />
</jmsQueue>

<zosConnectService
  id="samplezOSConnectService2"
  invokeURI="/samplezOSConnectService2"
  serviceName="samplezOSConnectService2_name"
  serviceRef="samplezOSConnectService2_MQ"
  dataXformRef="xformJSON2Byte" />

<mqzOSConnectService
  id="samplezOSConnectService2_MQ"
  connectionFactory="jms/sampleCF2"
  mqmdFormat="AFORMAT"
  destination="jms/sampleQ2Send"
  replyDestination="jms/sampleQ3Receive" />

<zosConnectDataXform id="xformJSON2Byte"
  bindFileLoc="/XFORM_ROOT/bindfiles" bindFileSuffix=".bnd"
  requestSchemaLoc="/XFORM_ROOT/json" requestSchemaSuffix=".json"
  responseSchemaLoc="/XFORM_ROOT/json"
  responseSchemaSuffix=".json" />
```

BytesMessage odebrane w programie z/OS Connect EE

Jeśli zostanie odebrana **BytesMessage**, można określić atrybuty **receiveConversion= "QMGR"** i **receiveCCSID= "37"** w definicji kolejki, która jest używana do odbierania komunikatu.

Przedstawiono to w poniższym przykładzie. W tym przypadku definicja `sampleQ2Receive` ma ustawione zarówno atrybuty **receiveConversion**, jak i **receiveCCSID**.

```
<jmsConnectionFactory
```

```

id="sampleCF2"
jndiName="jms/sampleCF2"
connectionManagerRef="sampleCF2ConnectionManager">

<properties.wmqJms
  transportType="BINDINGS"
  queueManager="MQ21"/>
</jmsConnectionFactory>

<connectionManager
  id="sampleCF2ConnectionManager"
  maxPoolSize="5"/>

<jmsQueue id="sampleQ2Send"
  jndiName="jms/sampleQ2Send">

  <properties.wmqJms
    baseQueueName="SampleQ2Send"
    CCSID="37"/>
</jmsQueue>

<jmsQueue id="sampleQ2Receive"
  jndiName="jms/sampleQ2Receive">

  <properties.wmqJms
    baseQueueName="SampleQ2Receive"
    receiveCCSID="37"
    receiveConversion="QMGR"/>
</jmsQueue>

<zosconnect_zosConnectService
  id="samplezOSConnectService2"
  invokeURI="/samplezOSConnectService2"
  serviceName="samplezOSConnectService2_name"
  serviceRef="samplezOSConnectService2_MQ"
  dataXformRef="xformJSON2Byte"/>

<mqzosconnect_mqzOSConnectService
  id="samplezOSConnectService2_MQ"
  connectionFactory="jms/sampleCF2"
  mqmdFormat="AFORMAT"
  destination="jms/sampleQ2Send"
  replyDestination="jms/sampleQ3Receive"/>

<zosconnect_zosConnectDataXform id="xformJSON2Byte"
  bindFileLoc="/XFORM_ROOT/bindfiles" bindFileSuffix=".bnd"
  requestSchemaLoc="/XFORM_ROOT/json" requestSchemaSuffix=".json"
  responseSchemaLoc="/XFORM_ROOT/json"
  responseSchemaSuffix=".json"/>

```

TextMessage odebrane w programie z/OS Connect V1

Jeśli serwer **TextMessage** ma zostać odebrany, należy przekształcić komunikat w identyfikator CCSID oczekiwany przez transformację danych, domyślnie jest to 37.

Jeśli używana jest niestandardowa transformacja danych, a transformacja oczekuje innego identyfikatora CCSID od 37, można określić atrybut CCSID receiveText w polu Element usługi mqzOSConnect , aby ustawić odpowiedni identyfikator CCSID.

Przedstawiono to w poniższym przykładzie. W tym przypadku przykładowa definicja zosConnectService3_MQ ma atrybut receiveTextCCSID w atrybucie mqzOSConnectService element ustawionym na 1208 (UTF-8).

```

<jmsConnectionFactory
  id="sampleCF3"
  jndiName="jms/sampleCF3"
  connectionManagerRef="sampleCF3ConnectionManager">

  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21"/>
</jmsConnectionFactory>

<connectionManager
  id="sampleCF3ConnectionManager"
  maxPoolSize="5"/>

```

```

<jmsQueue id="sampleQ3Send"
  jndiName="jms/sampleQ3Send">
  <properties.wmqJms
    baseQueueName="SampleQ3Send"
    CCSID="37"/>
</jmsQueue>

<jmsQueue id="sampleQ3Receive"
  jndiName="jms/sampleQ3Receive">
  <properties.wmqJms
    baseQueueName="SampleQ3Receive"/>
</jmsQueue>

<zosConnectService
  id="samplez0SConnectService3"
  invokeURI="/samplez0SConnectService3"
  serviceName="samplez0SConnectService3_name"
  serviceRef="samplez0SConnectService3_MQ"
  dataXformRef="customDataXForm"/>

<mqz0SConnectService
  id="samplez0SConnectService3_MQ"
  connectionFactory="jms/sampleCF3"
  mqmdFormat="AFORMAT"
  destination="jms/sampleQ3Send"
  replyDestination="jms/sampleQ3Receive"
  receiveTextCCSID="1208"/>

```

TextMessage odebrane w programie z/OS Connect EE

Jeśli serwer **TextMessage** ma zostać odebrany, należy przekształcić komunikat w identyfikator CCSID oczekiwany przez transformację danych, domyślnie jest to 37.

Jeśli używana jest niestandardowa transformacja danych, a transformacja oczekuje innego identyfikatora CCSID od 37, można określić atrybut CCSID receiveText w polu Element usługi mqzOSConnect, aby ustawić odpowiedni identyfikator CCSID.

Przedstawiono to w poniższym przykładzie. W tym przypadku przykładowa definicja zosConnectService3_MQ ma atrybut receiveTextCCSID w atrybucie mqzOSConnectService element ustawionym na 1208 (UTF-8).

```

<jmsConnectionFactory
  id="sampleCF3"
  jndiName="jms/sampleCF3"
  connectionManagerRef="sampleCF3ConnectionManager">
  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21"/>
</jmsConnectionFactory>

<connectionManager
  id="sampleCF3ConnectionManager"
  maxPoolSize="5"/>

<jmsQueue id="sampleQ3Send"
  jndiName="jms/sampleQ3Send">
  <properties.wmqJms
    baseQueueName="SampleQ3Send"
    CCSID="37"/>
</jmsQueue>

<jmsQueue id="sampleQ3Receive"
  jndiName="jms/sampleQ3Receive">
  <properties.wmqJms
    baseQueueName="SampleQ3Receive"/>
</jmsQueue>

<zosconnect_zosConnectService
  id="samplez0SConnectService3"

```

```

invokeURI="/samplezOSConnectService3"
serviceName="samplezOSConnectService3_name"
serviceRef="samplezOSConnectService3_MQ"
dataXformRef="customDataXForm"/>

```

```

<mqzosconnect_mqzOSConnectService
id="samplezOSConnectService3_MQ"
connectionFactory="jms/sampleCF3"
mqmdFormat="AFORMAT"
destination="jms/sampleQ3Send"
replyDestination="jms/sampleQ3Receive"
receiveTextCCSID="1208"/>

```

Praca z aplikacjami MQI

Produkt MQ Service Provider udostępnia interfejs RESTful do istniejących aplikacji, które korzystają z produktu IBM MQ. Aplikacje te mogą wchodzić w interakcje z produktem IBM MQ przy użyciu interfejsu IBM MQ classes for JMS lub interfejsu MQI (Message Queue Interface).

Domyślnie program IBM MQ classes for JMS wysyła komunikaty, które zawierają nagłówek MQRFH2. Jednak większość aplikacji MQI nie używa nagłówków MQRFH2.

Jeśli aplikacja IBM MQ classes for JMS współdzieli z aplikacją MQI, która nie została zaprojektowana do pracy z nagłówkami MQRFH2, konieczne jest skonfigurowanie, aby zapobiec wysłaniu komunikatów IBM MQ classes for JMS zawierających nagłówki MQRFH2. Więcej informacji na ten temat zawiera sekcja [Odwzorowywanie komunikatów JMS na komunikaty produktu IBM MQ](#).

Jeśli produkt MQ Service Provider ma wysyłać komunikaty do kolejek, które będą wykorzystywane przez aplikacje MQI, nie należy oczekiwać nagłówka MQRFH2, który będzie potrzebny do skonfigurowania produktu z/OS Connect, aby zapobiec wysłaniu nagłówka MQRFH2.

Konfigurację tę można osiągnąć, dodając atrybut **targetClient= "MQ"** do odpowiedniej kolejki dostawcy przesyłania komunikatów produktu IBM MQ w produkcie `server.xml`.

W poniższej przykładowej konfiguracji przedstawiono jednokierunkową usługę MQ Service Provider z produktem **id** o wartości `mqiService`. Z kolei ta konfiguracja została skonfigurowana pod kątem używania kolejki dostawcy przesyłania komunikatów produktu MQ z serwerem **id** o wartości `mqiQueue`.

Parametr `mqiQueue` został skonfigurowany za pomocą atrybutu **targetClient= "MQ"**. Oznacza to, że gdy aplikacja wysyła komunikat JMS do tej kolejki (na przykład, jeśli aplikacja wysyła żądanie HTTP POST do `mqiService`), komunikat nie będzie dodany do nagłówka MQRFH2.

z/OS Connect V1

```

<mqzOSConnectService
id="mqiService"
connectionFactory="jms/mqiCF"
destination="jms/mqiQueue"/>

<jmsQueue
id="mqiQueue"
jndiName="jms/mqiQueue">

  <properties.wmqJms
baseQueueName="MQIQueue"
targetClient = "MQ"/>
</jmsQueue>

```

z/OS Connect EE

```

<mqzosconnect_mqzOSConnectService
id="mqiService"
connectionFactory="jms/mqiCF"
destination="jms/mqiQueue"/>

<jmsQueue
id="mqiQueue"
jndiName="jms/mqiQueue">

  <properties.wmqJms
baseQueueName="MQIQueue"

```

```
targetClient = "MQ"/>  
</jmsQueue>
```


Uwagi

Niniejsza publikacja została opracowana z myślą o produktach i usługach oferowanych w Stanach Zjednoczonych.

IBM może nie oferować w innych krajach produktów, usług lub opcji omawianych w tej publikacji. Informacje o produktach i usługach dostępnych w danym kraju można uzyskać od lokalnego przedstawiciela IBM. Odwołanie do produktu, programu lub usługi IBM nie oznacza, że można użyć wyłącznie tego produktu, programu lub usługi IBM. Zamiast nich można zastosować ich odpowiednik funkcjonalny pod warunkiem, że nie narusza to praw własności intelektualnej firmy IBM. Jednakże cała odpowiedzialność za ocenę przydatności i sprawdzenie działania produktu, programu lub usługi pochodzących od producenta innego niż IBM spoczywa na użytkowniku.

IBM może posiadać patenty lub złożone wnioski patentowe na towary i usługi, o których mowa w niniejszej publikacji. Używanie tego dokumentu nie daje żadnych praw do tych patentów. Pisemne zapytania w sprawie licencji można przesyłać na adres:

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

Zapytania w sprawie licencji dotyczących informacji kodowanych przy użyciu dwubajtowych zestawów znaków (DBCS) należy kierować do lokalnych działów IBM Intellectual Property Department lub zgłaszać na piśmie pod adresem:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

Poniższy akapit nie obowiązuje w Wielkiej Brytanii, a także w innych krajach, w których jego treść pozostaje w sprzeczności z przepisami prawa miejscowego: INTERNATIONAL BUSINESS MACHINES CORPORATION DOSTARCZA TĘ PUBLIKACJĘ W STANIE, W JAKIM SIĘ ZNAJDUJE ("AS IS"), BEZ JAKICHKOLWIEK GWARANCJI (RĘKOJMIĘ RÓWNIEŻ WYŁĄCZA SIĘ), WYRAŻNYCH LUB DOMNIEMANYCH, A W SZCZEGÓLNOŚCI DOMNIEMANYCH GWARANCJI PRZYDATNOŚCI HANDLOWEJ, PRZYDATNOŚCI DO OKREŚLONEGO CELU ORAZ GWARANCJI, ŻE PUBLIKACJA TA NIE NARUSZA PRAW OSÓB TRZECICH. Ustawodawstwa niektórych krajów nie dopuszczają zastrzeżeń dotyczących gwarancji wyraźnych lub domniemanych w odniesieniu do pewnych transakcji; w takiej sytuacji powyższe zdanie nie ma zastosowania.

Informacje zawarte w niniejszej publikacji mogą zawierać nieścisłości techniczne lub błędy typograficzne. Informacje te są okresowo aktualizowane, a zmiany te zostaną uwzględnione w kolejnych wydaniach tej publikacji. IBM zastrzega sobie prawo do wprowadzania ulepszeń i/lub zmian w produktach i/lub programach opisanych w tej publikacji w dowolnym czasie, bez wcześniejszego powiadomienia.

Wszelkie wzmianki w tej publikacji na temat stron internetowych innych podmiotów zostały wprowadzone wyłącznie dla wygody użytkowników i w żadnym wypadku nie stanowią zachęty do ich odwiedzania. Materiały dostępne na tych stronach nie są częścią materiałów opracowanych dla tego produktu IBM, a użytkownik korzysta z nich na własną odpowiedzialność.

IBM ma prawo do używania i rozpowszechniania informacji przystanych przez użytkownika w dowolny sposób, jaki uzna za właściwy, bez żadnych zobowiązań wobec ich autora.

Licencjodawcy tego programu, którzy chcieliby uzyskać informacje na temat programu w celu: (i) wdrożenia wymiany informacji między niezależnie utworzonymi programami i innymi programami (łącznie

z tym opisywanym) oraz (ii) wspólnego wykorzystywania wymienianych informacji, powinni skontaktować się z:

IBM Corporation
Koordynator współdziałania z oprogramowaniem, Dział 49XA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Informacje takie mogą być udostępnione, o ile spełnione zostaną odpowiednie warunki, w tym, w niektórych przypadkach, zostanie uiszczona stosowna opłata.

Licencjonowany program opisany w niniejszej publikacji oraz wszystkie inne licencjonowane materiały dostępne dla tego programu są dostarczane przez IBM na warunkach określonych w Umowie IBM z Klientem, Międzynarodowej Umowie Licencyjnej IBM na Program lub w innych podobnych umowach zawartych między IBM i użytkownikami.

Wszelkie dane dotyczące wydajności zostały zebrane w kontrolowanym środowisku. W związku z tym rezultaty uzyskane w innych środowiskach operacyjnych mogą się znacząco różnić. Niektóre pomiary mogły być dokonywane na systemach będących w fazie rozwoju i nie ma gwarancji, że pomiary wykonane na ogólnie dostępnych systemach dadzą takie same wyniki. Niektóre z pomiarów mogły być estymowane przez ekstrapolację. Rzeczywiste wyniki mogą być inne. Użytkownicy powinni we własnym zakresie sprawdzić odpowiednie dane dla ich środowiska.

Informacje dotyczące produktów innych niż produkty IBM pochodzą od dostawców tych produktów, z opublikowanych przez nich zapowiedzi lub innych powszechnie dostępnych źródeł. Firma IBM nie testowała tych produktów i nie może potwierdzić dokładności pomiarów wydajności, kompatybilności ani żadnych innych danych związanych z tymi produktami. Pytania dotyczące możliwości produktów innych podmiotów należy kierować do dostawców tych produktów.

Wszelkie stwierdzenia dotyczące przyszłych kierunków rozwoju i zamierzeń IBM mogą zostać zmienione lub wycofane bez powiadomienia.

Publikacja ta zawiera przykładowe dane i raporty używane w codziennych operacjach działalności gospodarczej. W celu kompleksowego ich zilustrowania podane przykłady zawierają nazwiska osób prywatnych, nazwy przedsiębiorstw oraz nazwy produktów. Wszystkie te nazwy/nazwiska są fikcyjne i jakiegokolwiek podobieństwo do istniejących nazw/nazwisk i adresów jest całkowicie przypadkowe.

LICENCJA W ZAKRESIE PRAW AUTORSKICH:

Niniejsza publikacja zawiera przykładowe aplikacje w kodzie źródłowym, ilustrujące techniki programowania w różnych systemach operacyjnych. Użytkownik może kopiować, modyfikować i dystrybuować te programy przykładowe w dowolnej formie bez uiszczania opłat na rzecz IBM, w celu projektowania, używania, sprzedaży lub dystrybucji aplikacji zgodnych z aplikacyjnym interfejsem programistycznym dla tego systemu operacyjnego, dla którego napisane zostały programy przykładowe. Programy przykładowe nie zostały gruntownie przetestowane. IBM nie może zatem gwarantować ani sugerować niezawodności, użyteczności i funkcjonalności tych programów.

W przypadku przeglądania niniejszych informacji w formie elektronicznej, zdjęcia i kolorowe ilustracje mogą nie być wyświetlane.

Informacje dotyczące interfejsu programistycznego

Informacje dotyczące interfejsu programistycznego, o ile są udostępniane, mają być pomocne podczas tworzenia oprogramowania aplikacji do użytku z tym programem.

Ten podręcznik zawiera informacje na temat planowanych interfejsów programistycznych, które umożliwiają klientom pisanie programów w celu uzyskania dostępu do usług produktu WebSphere MQ.

Informacje te mogą również zawierać informacje na temat diagnostyki, modyfikacji i strojenia. Tego typu informacje są udostępniane jako pomoc przy debugowaniu aplikacji.

Ważne: Informacji na temat diagnostyki, modyfikacji i strojenia nie należy używać jako interfejsu programistycznego, ponieważ może on ulec zmianie.

Znaki towarowe

IBM, logo IBM, ibm.com, są znakami towarowymi IBM Corporation, zarejestrowanymi w wielu systemach prawnych na całym świecie. Aktualna lista znaków towarowych IBM jest dostępna w serwisie WWW, w sekcji "Copyright and trademark information" (Informacje o prawach autorskich i znakach towarowych), pod adresem www.ibm.com/legal/copytrade.shtml. Nazwy innych produktów lub usług mogą być znakami towarowymi IBM lub innych podmiotów.

Microsoft oraz Windows są znakami towarowymi Microsoft Corporation w Stanach Zjednoczonych i/lub w innych krajach.

UNIX jest zastrzeżonym znakiem towarowym The Open Group w Stanach Zjednoczonych i/lub w innych krajach.

Linux jest zastrzeżonym znakiem towarowym Linusa Torvaldsa w Stanach Zjednoczonych i/lub w innych krajach.

Ten produkt zawiera oprogramowanie opracowane przez Eclipse Project (<http://www.eclipse.org/>).

Java oraz wszystkie znaki towarowe i logo dotyczące języka Java są znakami towarowymi lub zastrzeżonymi znakami towarowymi Oracle i/lub przedsiębiorstw afiliowanych Oracle.



Numer pozycji:

(1P) P/N: