

9.2

Administrowanie produktem IBM MQ

IBM

Uwaga

Przed użyciem tych informacji oraz produktu, którego dotyczą, należy zapoznać się z informacjami w sekcji [“Uwagi” na stronie 561](#).

To wydanie dotyczy wersji 9 wydanie 2 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 2007, 2024.**

Spis treści

Administrowanie.....	7
Sposoby administrowania menedżerami kolejek produktu IBM MQ i powiązаныmi zasobami.....	8
Administrowanie programem IBM MQ za pomocą komend sterujących.....	11
Administrowanie za pomocą komend MQSC.....	12
Interaktywne uruchamianie komend MQSC.....	14
Uruchamianie komend MQSC z plików tekstowych.....	18
Źródła, z których można wydawać komendy MQSC w systemie z/OS.....	21
Automatyczna konfiguracja ze skryptu MQSC przy uruchamianiu.....	22
Wartości ogólne i znaki o specjalnych znaczeniach.....	23
Automatyzacja administrowania produktem IBM MQ przy użyciu komend PCF.....	24
Wprowadzenie do formatów komend programowalnych IBM MQ.....	25
Korzystanie z interfejsu MQAI w celu uproszczenia korzystania z systemów PCF.....	37
Administrowanie przy użyciu REST API.....	74
Pierwsze kroki w produkcie administrative REST API.....	75
Zdalne administrowanie przy użyciu REST API.....	80
REST API datowniki.....	85
REST API obsługa błędów.....	85
REST API Wykrywanie.....	88
REST API Obsługa języków narodowych.....	89
Wersja REST API.....	90
Administrowanie przy użyciu konsoli WWW.....	92
Pierwsze kroki z konsolą WWW.....	92
Szybkie oprowadzanie po nowej konsoli WWW.....	94
Przełączanie między typami konsoli.....	116
Administrowanie przy użyciu IBM MQ Explorer.....	117
Co można zrobić za pomocą IBM MQ Explorer.....	117
Konfigurowanie produktu IBM MQ Explorer.....	119
Korzystanie z aplikacji paska zadań IBM MQ (tylko w wersji Windows).....	125
Aplikacja monitora alertów IBM MQ (tylko Windows).....	126
Praca z lokalnymi obiektami IBM MQ.....	126
Praca z menedżerami kolejek.....	126
Zatrzymywanie kanałów MQI.....	136
Praca z kolejkami lokalnymi.....	137
Praca z kolejkami zdalnymi.....	146
Praca z kolejkami aliasowymi.....	149
Praca z kolejkami modelami.....	151
Praca z kolejkami niedostarczonych komunikatów.....	152
Praca z tematami administracyjnymi.....	171
Praca z subskrypcjami.....	175
Praca z usługami.....	178
Zarządzanie obiektami w celu wyzwolenia.....	186
Używanie programu narzędziowego dmpmqmsg między dwoma systemami.....	188
Praca ze zdalnymi obiektami IBM MQ.....	191
Konfigurowanie menedżerów kolejek na potrzeby zdalnego administrowania.....	192
Zarządzanie serwerem komend w celu zdalnego administrowania.....	196
Wydawanie komend MQSC w zdalnym menedżerze kolejek.....	197
Konwersja danych między kodowanymi zestawami znaków.....	199
administrowanieManaged File Transfer.....	204
Uruchamianie agenta MFT.....	204
Wyświetlanie listy agentów MFT.....	209
Zatrzymywanie agenta MFT.....	210
Rozpoczynanie nowego przesyłania plików.....	211

Tworzenie zaplanowanego przesyłania plików.....	214
Praca z oczekującymi transferami plików.....	215
Wyzwalanie przesyłania plików.....	216
Monitorowanie przesyłania plików, które są w toku.....	217
Wyświetlanie statusu przesyłania plików w dzienniku przesyłania.....	220
Monitorowanie zasobów produktu MFT.....	222
Praca z szablonami przesyłania plików.....	253
Przesyłanie danych z plików do wiadomości.....	256
Przesyłanie danych z komunikatów do plików.....	264
Most protokołu.....	270
Most Connect:Direct.....	292
Praca z MFT z IBM Integration Bus.....	307
Odtwarzanie i restartowanie produktu MFT.....	307
Ustawianie limitu czasu na odzyskiwanie wstrzymanych transferów.....	308
administrowanieMQ Telemetry.....	313
Konfigurowanie menedżera kolejek dla telemetrii w systemach Linux i AIX.....	314
Konfigurowanie menedżera kolejek dla telemetrii w systemie Windows.....	316
Konfigurowanie rozproszonego kolejkowania w celu wysyłania komunikatów do klientów MQTT.....	318
MQTT - identyfikacja, autoryzacja i uwierzytelnianie klienta.....	320
Uwierzytelnianie kanału telemetrycznego przy użyciu protokołu TLS.....	327
Prywatność publikacji na kanałach telemetrycznych.....	329
Konfiguracja TLS dla klientów MQTT Java i kanałów telemetrycznych.....	329
Konfiguracja kanału telemetrycznego JAAS.....	334
Administrowanie klientem AMQP.....	336
Wyświetlanie obiektów produktu IBM MQ w użyciu przez klienty AMQP.....	336
Identyfikacja, autoryzacja i uwierzytelnianie klienta AMQP.....	338
Prywatność publikacji w kanałach.....	340
Konfigurowanie klientów AMQP za pomocą protokołu TLS.....	341
Rozłączanie klientów AMQP z menedżera kolejek.....	341
Administrowanie rozgłaszanie.....	342
Pierwsze kroki w trybie rozsyłania grupowego.....	342
IBM MQ Topologia tematu rozsyłania grupowego.....	343
Kontrolowanie wielkości komunikatów rozsyłania grupowego.....	344
Włączanie konwersji danych na potrzeby przesyłania komunikatów w trybie Multicast.....	346
Monitorowanie aplikacji rozsyłania grupowego.....	347
Niezawodność komunikatów rozsyłania grupowego.....	348
Zaawansowane zadania rozsyłania grupowego.....	348
administrowanieIBM MQ for IBM i.....	351
Zarządzanie produktem IBM MQ for IBM i za pomocą komend CL.....	352
Alternatywne sposoby administrowania produktem IBM MQ for IBM i.....	366
Zarządzanie pracą dla produktu IBM i.....	371
Dostępność, tworzenie kopii zapasowych, odtwarzanie i restartowanie w systemie IBM i.....	378
WYCISZANIEIBM MQ for IBM i.....	423
administrowanieIBM MQ for z/OS.....	427
Wysyłanie komend do programu IBM MQ for z/OS.....	428
Programy narzędziowe serwera IBM MQ for z/OS.....	436
Operacyjny IBM MQ for z/OS.....	439
Pisanie programów do administrowania programem IBM MQ for z/OS.....	460
Zarządzanie zasobami IBM MQ w systemie z/OS.....	472
Odtwarzanie i restartowanie w systemie z/OS.....	513
IBM MQ i IMS.....	536
Operacyjne Advanced Message Security w systemie z/OS.....	549
administrowanieIBM MQ Internet Pass-Thru.....	550
Uruchamianie i zatrzymywanie serwera MQIPT.....	550
Administrowanie programem MQIPT przy użyciu wiersza komend.....	553
Tworzenie kopii zapasowych.....	559
Dostrajanie wydajności.....	560

Uwagi.....	561
Informacje dotyczące interfejsu programistycznego.....	562
Znaki towarowe.....	563

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.

O tym zadaniu

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.

Więcej informacji na ten temat zawiera [“Praca z lokalnymi obiektami IBM MQ” na stronie 126](#).

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.




Więcej informacji na ten temat zawiera [“Praca ze zdalnymi obiektami IBM MQ” na stronie 191](#).


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 interfejsy API administrowania.

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

- [“Komendy sterujące produktem IBM MQ” na stronie 8](#)
- [“Komendy produktu IBM MQ Script \(MQSC\)” na stronie 8](#)
- [“Programowalne formaty komend \(PCFs\)” na stronie 9](#)
- [administrative REST API](#)
-  [“Język CL \(IBM i Control Language\)” na stronie 9](#)

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

-   [“IBM MQ Explorer” na stronie 10](#)
- [“IBM MQ Console” na stronie 10](#)
-  [“Usługa klastrowa produktu Microsoft \(MSCS\)” na stronie 10](#)

 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 427](#).

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 24.

Pojęcia pokrewne

[IBM MQ Przegląd techniczny](#)

Zadania pokrewne

[Planowanie](#)

[Konfigurowanie](#)

Odsyłacze pokrewne

[Porównanie zestawów komend](#)

Sposoby administrowania menedżerami kolejek produktu IBM MQ i powiązаныmi zasobami

Istnieje kilka różnych opcji administrowania menedżerami kolejek produktu IBM MQ i powiązаныmi zasobami.

Komendy sterujące produktem IBM MQ



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

Systemy IBM MQ for AIX, Linux®, and Windows udostępniają *komendy sterujące*, które są dostępne 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 nasłuchiwania, 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 z/OS. Komendy MQSC są podsumowywane w sekcji [Porównywanie zestawów komend](#).



W systemie AIX, Linux, and Windows można używać komend MQSC jako 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 [“Uruchamianie komend MQSC z plików tekstowych”](#) na stronie 18.

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 odpowiedzi MQSC dla komend, które są wysyłane z pliku skryptowego, są zwracane w pliku buforowy-buforu.

Więcej informacji na temat używania komend MQSC zawiera sekcja [“Administrowanie za pomocą komend MQSC”](#) na stronie 12.

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 IBM MQ: obiektami informacji uwierzytelniających, kanałami, obiektami nastuchiwania kanałów, listami nazw, definicjami procesów, menedżerami kolejek, kolejkami, usługami i klasami 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 25.

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

administrative REST API

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:

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

Produktem 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 nasłuchiwania, 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 [“Administrowanie przy użyciu IBM MQ Explorer”](#) na stronie 117.

Produkt IBM MQ Explorer może być używany do administrowania zdalnymi menedżerami kolejek na innych platformach, w tym z/OS.

Produkt IBM MQ Explorer może być instalowany jako część instalacji produktu (patrz sekcja [Instalowanie i deinstalowanie produktu IBM MQ](#)) lub z autonomicznego programu IBM MQ Explorer do pobrania dostępnego w produkcie Fix Central (patrz sekcja [Instalowanie i deinstalowanie produktu IBM MQ Explorer jako autonomicznej aplikacji w systemach Linux i Windows](#)).

IBM MQ Console

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

Więcej informacji na ten temat zawiera [“Administrowanie przy użyciu konsoli WWW”](#) na stronie 92.

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 MQ Klastry

Są to grupy dwóch lub większej liczby menedżerów kolejek na jednym lub większej liczby komputerów, które zapewniają automatyczne połączenie wzajemne i umożliwiają 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 *przetączenie 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.

ALW Administrowanie programem IBM MQ za pomocą komend sterujących

Komendy sterujące umożliwiają wykonywanie wielu zadań administracyjnych serwera IBM MQ w systemie AIX, 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 AIX, 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, CRTmqm.
- Flagę można wprowadzić jako -u, -U, /u lub /U.
- SYSTEM.DEAD.LETTER.QUEUE i jupiter.queue.manager muszą być wprowadzane dokładnie tak, jak pokazano.

Używanie komend sterujących w systemach AIX and Linux

Linux

AIX

W systemach IBM MQ for AIX or Linux komendy sterujące są wprowadzane w oknie powłoki.

W środowiskach UNIX and Linux komendy sterujące, w tym samą nazwę komendy, opcje i wszystkie argumenty, są rozróżniane wielkości 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.

Odsyłacze pokrewne

[Skorowidz komend sterujących programu IBM MQ](#)

Administrowanie za pomocą komend MQSC

Za pomocą komend MQSC można zarządzać obiektami menedżera kolejek, w tym samym menedżerem kolejek, kolejkami, definicjami procesów, kanałami, kanałami połączeń klientów, obiektami nasłuchiwania, usługami, listami nazw, klastrami i obiektami informacji uwierzytelniających. Komendy MQSC są dostępne na wszystkich platformach.

O tym zadaniu

Sposób wydawania komend MQSC jest zależny od platformy:

- **ALW** W systemie AIX, Linux, and Windows komendy MQSC są emitowane do menedżera kolejek za pomocą komendy `runmqsc`. Komendę `runmqsc` można uruchomić na kilka sposobów:
 - Interaktywnie, wydając komendy z klawiatury. Patrz [“Interaktywna uruchamianie komend MQSC” na stronie 14](#).
 - Z pliku tekstowego ASCII. Patrz [“Uruchamianie komend MQSC z plików tekstowych” na stronie 18](#).
 - W zdalnym menedżerze kolejek. Patrz [“Praca ze zdalnymi obiektami IBM MQ” na stronie 191](#).
- **z/OS** W systemie z/OS komendy MQSC mogą być wydawane z wielu źródeł, w zależności od komendy. Więcej informacji na ten temat zawiera sekcja [“Źródła, z których można wydawać komendy MQSC w systemie z/OS” na stronie 21](#).

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

Procedura

- 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 odstępów 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 zawierające spacje, małe litery lub znaki specjalne muszą być ujęte w pojedynczy cudzysłów, chyba że spełniony jest jeden z poniższych warunków:
 - Znaki specjalne to jeden lub kilka z następujących znaków:
 - Kropka (.)
 - Prawy ukośnik (/)
 - Podkreślenie (_)
 - Znak procentu (%)
 -  Komenda jest wydawana z poziomu paneli sterujących i sterujących IBM MQ for z/OS .
 - Łańcuch jest wartością ogólną kończącą się gwiazdką. (w systemie IBM i należy je ująć w pojedyncze cudzysłowy)
 - Łańcuch jest pojedynczą gwiazdką, na przykład TRACE (*) (w przypadku IBM i należy je ująć w pojedyncze cudzysłowy)
 - Łańcuch jest specyfikacją zakresu zawierającą 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.

- 

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 jest jednym z następujących atrybutów, gdy dwa pojedyncze cudzysłowy bez spacji są interpretowane jako łańcuch o zerowej długości:

 - TOPICSTR
 - SUB
 - USERDATA
 - SELECTOR

- 

W systemie z/OS, jeśli spacja ma być ujęta w znaki pojedynczego cudzysłowu, należy wprowadzić ją jako taką ("). Łańcuch zawierający nie znaki (") jest taki sam, jak wprowadzanie ().
- Wszystkie odstępy końcowe w atrybutach łańcuchowych, które są oparte na typach MQCHARV, takich jak SELECTOR, dane użytkownika podrzędnego, są traktowane jako znaczące, co oznacza, że produkt 'abc ' nie jest równy 'abc '.
- Nawias otwierający, po którym następuje nawias zamykający, bez istotnych informacji pomiędzy, jest niepoprawny, z wyjątkiem przypadków, w których zaznaczono inaczej. Na przykład następujący łańcuch nie jest poprawny:

```
NAME ( )
```

- 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; DEFI 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.

- Komendy MQSC używają określonych znaków specjalnych do określenia określonych znaczeń. Więcej informacji na temat tych znaków specjalnych oraz sposobu ich używania zawiera sekcja [“Wartości ogólne i znaki o specjalnych znaczeniach”](#) na stronie 23.

Zadania pokrewne

Rozwiązywanie problemów za pomocą komend MQSC

Odsyłacze pokrewne


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

Interaktywne uruchamianie komend MQSC

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

Zanim rozpoczniesz

Użytkownik może ustawić zachętę, która będzie wyświetlana po uruchomieniu komendy `runmqsc`. Patrz [“Ustawianie wiersza komend MQSC”](#) na stronie 15.

 Gdy komendy MQSC są uruchamiane interaktywnie na platformach UNIX and Linux, wiersz komend `runmqsc` obsługuje przywracanie komend, zakończenie komend oraz klawisze komend Emacs. Patrz [“Komenda runmqsc w systemach AIX i Linux”](#) na stronie 17.

O tym zadaniu

Procedura

1. Aby używać komend MQSC interaktywnie, otwórz okno komend lub powłokę i wprowadź następującą komendę:

```
runmqsc QMgrName
```

Gdzie *QMgrName* określa nazwę menedżera kolejek, który ma być przetwarzany za pomocą komend MQSC. Aby przetworzyć komendy MQSC w domyślnym menedżerze kolejek, można pozostawić puste pole *QMgrName*.

2. W razie potrzeby wpisz dowolne komendy MQSC. Na przykład, aby utworzyć kolejkę lokalną o nazwie `ORANGE.LOCAL.QUEUE`, należy wprowadzić następującą komendę:

```
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 (;).

3. Zatrzymaj pracę z komendami MQSC, wprowadzając następującą komendę:

```
end
```

Alternatywnie można użyć znaku EOF dla danego systemu operacyjnego.

Wyniki

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

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

Następujący komunikat wskazuje, że wystąpił błąd składniowy:

```
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
```

Komunikaty te są wysyłane do standardowego urządzenia wyjściowego. Jeśli komenda nie została wprowadzona poprawnie, należy zapoznać się z informacjami referencyjnymi dla komendy, aby znaleźć poprawną składnię. Patrz sekcja [Komendy MQSC](#).

Zadania pokrewne

[“Uruchamianie komend MQSC z plików tekstowych” na stronie 18](#)

Interaktywne uruchamianie komend MQSC jest odpowiednie dla szybkich testów, ale w przypadku bardzo długich komend lub wielokrotnego używania określonej sekwencji komend można przekierować plik stdin z pliku tekstowego. Można również przekierować dane wyjściowe do pliku.

Odsyłacze pokrewne

[runmqsc](#)

Ustawianie wiersza komend MQSC

Wiersz komend MQSC można ustawić na wybraną zachętę za pomocą zmiennej środowiskowej **MQPROMPT**.



Procedura

- Ustaw zmienną środowiskową **MQPROMPT** na wybrane przez siebie pytanie.

Zachęta jest wstawiana zarówno wtedy, gdy konsola **runmqsc** jest uruchamiana interaktywnie, jak i wtedy, gdy dane wejściowe są przekierowywane do programu **runmqsc** z pliku lub ze standardowego urządzenia wejściowego (stdin).

W wierszu komend można umieścić zwykły tekst, a także można wstawiać zmienne środowiskowe, używając notacji +VARIABLE+ w taki sam sposób, jak w przypadku definicji obiektów usług systemu IBM MQ. Więcej informacji na ten temat zawiera sekcja [“Wymiennie wstawki w definicjach usług” na stronie 182](#).

Istnieje szereg innych dodatkowych wymiennych wkładek IBM MQ, opisanych w poniższej tabeli.

Wymienny wkład	Opis
MQ_HOST_NAME	Nazwa hosta systemu
MQ_FILE_SEP	Separator plików specyficzny dla platformy: –  Linux  AIX W systemach AIX and Linux plik MQ_FILE_SEP ma wartość /.

Wymienny wkład	Opis
	<ul style="list-style-type: none"> – Windows W systemach Windows położenie pliku MQ_FILE_SEP to \
MQ_PATH_SEP,	Separator ścieżki specyficzny dla platformy: <ul style="list-style-type: none"> – Linux AIX W systemach AIX and Linux zmienna MQ_PATH_SEP ma wartość :. – Windows W systemach Windows położenie pliku MQ_PATH_SEP to ;
MQ_DATE_TIME,	Data i godzina w systemie lokalnym w stałym formacie YYYY-MM-DD hh:mm:ss.SSS , na przykład: <pre>2020-12-25 17:41:37.408</pre>

Uwagi:

- Wartości wymiennych wstawień MQ odnoszą się do instalacji systemu IBM MQ i hosta, z którym powiązana jest komenda **runmqsc** .
- Długość parametru MQPROMPT jest ograniczona do 256 znaków podczas rozwijania operacji wstawiania. Rozwinięcie MQPROMPT na tej wartości spowoduje, że cały łańcuch MQPROMPT zostanie obcięty bez rozwinięć.

Na przykład, aby ustawić wiersz komend na MQSC, wprowadź jedną z następujących komend:

- **Windows**

```
set "MQPROMPT=MQSC"
```

- **Linux** **AIX**

```
export MQPROMPT="MQSC"
```

Przykład

Poniższy przykład przedstawia ustawianie zmiennej **MQPROMPT** w systemie AIX . W polu jest wyświetlana nazwa użytkownika pobrana z powiązanej systemowej zmiennej środowiskowej, nazwa menedżera kolejek i nazwa hosta IBM MQ pobrana z wymiennych wstawień produktu MQ :

```
sh> export MQPROMPT="+USER+ @ +QMNAME+ @ +MQ_HOST_NAME+> "
sh> runmqsc MY.QMGR
5724-H72 (C) Copyright IBM Corp. 1994, 2024.
Starting MQSC for queue manager MY.QMGR.

myuser @ MY.QMGR @ aix1> DISPLAY QMSTATUS
```

```
C:\ > set "MQPROMPT+=+USERNAME+ @ +QMNAME+ @ +MQ_HOST_NAME+> "
C:\ > runmqsc MY.QMGR
5724-H72 (C) Copyright IBM Corp. 1994, 2024.
Starting MQSC for queue manager MY.QMGR.

myuser @ MY.QMGR @ WIN1> DISPLAY QMSTATUS
```

W poniższym przykładzie do powyższych przykładów komendy MQPROMPT zostanie dodany znacznik czasu, który jest pobierany z zastępowanych wstawień MQ :

```
sh> export MQPROMPT="+MQ_DATE_TIME+ +USER+ @ +QMNAME+ @ +MQ_HOST_NAME+> "
sh> runmqsc MY.QMGR
5724-H72 (C) Copyright IBM Corp. 1994, 2024.
Starting MQSC for queue manager MY.QMGR.
```



```
2020-11-24 18:10:00.404 myuser @ MY.QMGR @ aix1> DISPLAY QMSTATUS
```

```
C:\ > set "MQPROMPT="+MQ_DATE_TIME+ +USERNAME+ @ +QMNAME+ @ +MQ_HOST_NAME+> "  
C:\ > runmqsc MY.QMGR  
5724-H72 (C) Copyright IBM Corp. 1994, 2024.  
Starting MQSC for queue manager MY.QMGR.
```

```
2020-11-24 18:10:01.007 myuser @ MY.QMGR @ WIN1> DISPLAY QMSTATUS
```

Linux

AIX


Komenda runmqsc w systemach AIX i Linux

Wiersz komend **runmqsc** w systemach AIX 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 Linux zainstaluj produkt `ncurses`.

Instalowanie produktu `ncurses` lub `curses` w systemie AIX

Uwaga: W poniższym przykładzie użyto instrukcji dla produktu Linux .

Uruchom następującą komendę, aby znaleźć istniejące pakiety produktu `ncurses` :

```
rpm -qa | grep -i ncurses
```

Wymagane pakiety `ncurses` są następujące:

```
ncurses-term-6.1-7.20180224.el8.noarch  
ncurses-6.1-7.20180224.el8.x86_64  
ncurses-base-6.1-7.20180224.el8.noarch  
ncurses-c++-libs-6.1-7.20180224.el8.x86_64  
ncurses-libs-6.1-7.20180224.el8.x86_64  
ncurses-compat-libs-6.1-7.20180224.el8.x86_64  
ncurses-devel-6.1-7.20180224.el8.x86_64
```

Wszystkie wymagane pakiety `ncurses` wymienione w poprzednim tekście można zainstalować, uruchamiając następującą komendę:

```
yum install ncurses*
```

Dostosowywanie przypisać 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 dla zmiennej środowiskowej **MQ_OVERRIDE_LIBEDIT_LOAD** wartość 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

Interaktywne uruchamianie komend MQSC jest odpowiednie dla szybkich testów, ale w przypadku bardzo długich komend lub wielokrotnego używania określonej sekwencji komend można przekierować plik stdin z pliku tekstowego. Można również przekierować dane wyjściowe do pliku.

O tym zadaniu

Dane wejściowe dla komendy **runmqsc** są pobierane ze *standardowego urządzenia wejściowego*, zwanego również stdin. stdin jest urządzeniem, z którego pobierane są dane wejściowe do systemu. Zwykle jest to klawiatura, ale można na przykład określić, że dane wejściowe mają pochodzić z portu szeregowego lub pliku dyskowego.

Dane wyjściowe komendy **runmqsc** są wysyłane do *standardowego urządzenia wyjściowego*, zwanego również stdout. stdout jest urządzeniem, do którego wysyłane są dane wyjściowe z systemu. Zazwyczaj jest to ekran, ale można przekierować dane wyjściowe do portu szeregowego lub pliku.

Komendy MQSC można zbudować w skrypcie, jeśli używane są:

- **z/OS** Zestawy danych inicjowania CSQINP1, CSQINP2i CSQINPX lub program narzędziowy CSQUTIL w systemie z/OS.
- **IBM i** Komenda **STRMQM** w systemie IBM i.
- **ALW** Komenda **runmqsc** w systemie AIX, Linux, and Windows.

Wiersz komend MQSC można ustawić na wybraną zachętę za pomocą zmiennej środowiskowej **MQPROMPT**. Więcej informacji na ten temat zawiera sekcja [“Ustawianie wiersza komend MQSC”](#) na stronie 15.

Procedura

1. Utwórz plik tekstowy zawierający komendy MQSC, które mają zostać uruchomione.
 - W celu zapewnienia przenośności między środowiskami IBM MQ należy ograniczyć długość wiersza w plikach komend MQSC do 72 znaków.
 - Każda komenda musi się zaczynać w nowym wierszu.
 - Wiersz rozpoczynający się gwiazdką (*) na pierwszej pozycji jest ignorowany. Służy do wstawiania komentarzy do pliku.
 - Puste wiersze są ignorowane.
 - Znak plus (+) oznacza, że komenda jest kontynuowana od pierwszego niepustego znaku w następnym wierszu. Jeśli do kontynuowania komendy zostanie użyty znak +, należy pamiętać o pozostawieniu co najmniej jednego odstępu przed następnym parametrem (z wyjątkiem z/OS, gdzie nie jest to konieczne). Komentarze lub puste wiersze są odrzucane, gdy komenda jest składana w pojedynczy łańcuch.

- Znak minus (-) oznacza, że komenda ma być kontynuowana od początku następnego wiersza. Komentarze lub puste wiersze są odrzucane, gdy komenda jest składana w pojedynczy łańcuch.
- Komendy MQSC zawarte w komendzie Escape PCF (Programmable Command Format) nie mogą być kontynuowane ze znakiem plus ani znakiem minus. Cała komenda musi być zawarta w pojedynczej komendzie Escape. Informacje na temat komend PCF zawiera sekcja [“Wprowadzenie do formatów komend programowalnych IBM MQ”](#) na stronie 25.
- W systemie Multiplatformsi w systemie z/OS dla komend wprowadzonych z programu narzędziowego zadania wsadowego CSQUTIL można użyć znaku średnika (;), aby zakończyć komendę, nawet jeśli na końcu poprzedniego wiersza wprowadzono znak plus (+).
- Wiersz nie może kończyć się znakiem sterującym klawiatury (na przykład tabulacją).
- Jeśli komenda **runmqsc** zostanie uruchomiona w trybie klienta przez przekierowanie pliku stdin z pliku tekstowego i zostanie podana opcja **-u** w celu podania informacji autoryzacyjnych, komenda **runmqsc** nie poprosi o podanie hasła, a zamiast tego hasło zostanie odczytane z pliku stdin. Należy upewnić się, że pierwszym wierszem danych udostępnionym przez stdin jest hasło. Można to zrobić za pomocą narzędzi wiersza komend, takich jak "echo" lub "cat", i przekazać hasło, po którym następuje skrypt MQSC, do komendy **runmqsc** stdin.
- **Windows** W systemie Windows, jeśli w skrypcie komend (na przykład jako część opisu obiektu) używane są pewne znaki specjalne, takie jak znak funta (£) i logiczne NOT (~), są one wyświetlane w inny sposób w danych wyjściowych komendy, takiej jak **DISPLAY QLOCAL**.
- Więcej informacji na temat składni komend MQSC zawiera sekcja [Komendy MQSC](#).
- Do utworzenia pliku tekstowego można użyć przykładowych plików komend MQSC:

amqscos0.tst

Definicje obiektów używanych przez programy przykładowe.

amqscic0.tst

Definicje kolejek dla transakcji CICS .

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

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

2. Sprawdź, czy składnia komendy jest poprawna w lokalnym menedżerze kolejek bez uruchamiania komend. Użyj opcji **-v** w komendzie **runmqsc** .

V 9.2.0

- W systemie IBM MQ 9.2.0 należy użyć opcji **-f** , aby określić nazwę pliku tekstu wejściowego. Na przykład:

```
runmqsc -f myprog.in -v QmgrName
```

- W przypadku wersji Long Term Support wcześniejszych niż IBM MQ 9.2.0i wersji Continuous Delivery wcześniejszych niż IBM MQ 9.1.4 należy użyć operatora **<**, aby skierować komendy MQSC z wejściowego pliku tekstowego do komendy. Na przykład:

```
runmqsc -v QmgrName < myprog.in
```

Zwrócony raport jest podobny do raportu przedstawionego na [Rysunek 2](#) na stronie 21.

Podczas weryfikowania komend nie można określić zdalnego menedżera kolejek. Oznacza to, że nie można podać opcji **-w** .

3. Jeśli składnia komendy jest poprawna, usuń opcję **-v** , a następnie ponownie uruchom komendę **runmqsc** .

- W katalogu IBM MQ 9.2.0 uruchom (na przykład) następującą komendę:

```
runmqsc -f myprog.in QmgrName
```

- W przypadku wersji Long Term Support wcześniejszych niż IBM MQ 9.2.0 i wersji Continuous Delivery wcześniejszych niż IBM MQ 9.1.4 należy użyć jednej z następujących komend:
 - Operator < kieruje dane wejściowe z pliku tekstowego. Na przykład następująca komenda uruchamia sekwencję komend zawartych w pliku tekstowym myprog.in:

```
runmqsc QMgrName < myprog.in
```

- Operator > kieruje dane wyjściowe do pliku tekstowego. Na przykład poniższa komenda uruchamia sekwencję komend zawartych w pliku tekstowym myprog.in i zapisuje je w pliku o nazwie results.out:

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

Rysunek 1 na stronie 20 przedstawia fragment pliku komend MQSC myprog.in, a Rysunek 2 na stronie 21 -odpowiedni fragment danych wyjściowych w pliku results.out.

Przykład

Komendy MQSC są zapisywane w postaci czytelnej dla człowieka, czyli w postaci tekstu ASCII. W poniższym przykładzie przedstawiono wyodrębnienie z pliku komend MQSC zawierającego komendę MQSC **DEFINE QLOCAL**.

```
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

Po zakończeniu działania komendy **runmqsc** zostanie zwrócony raport. Poniżej przedstawiono przykład wyodrębniania z raportu:

```

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

Zadania pokrewne

“Ustawianie wiersza komend MQSC” na stronie 15

Wiersz komend MQSC można ustawić na wybraną zachętę za pomocą zmiennej środowiskowej **MQPROMPT**.

“Interaktywne uruchamianie komend MQSC” na stronie 14

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

Odsyłacze pokrewne

[runmqsc](#)

z/OS Źródła, z których można wydawać komendy MQSC w systemie z/OS

Komendy MQSC mogą być wydawane z różnych źródeł, w zależności od komendy.

Komendy mogą być wydawane z następujących źródeł:

- Konsola produktu z/OS lub jej odpowiednik
- Wejściowe zestawy danych inicjowania CSQINP1, CSQINP2, CSQINPT i CSQINPX
- Program wsadowy CSQUTIL
- Odpowiednio autoryzowane aplikacje, wysyłając komendy jako komunikaty do systemu SYSTEM.COMMAND.INPUT, kolejka

z/OS Szczegółowe informacje na ten temat zawiera sekcja [“Wysyłanie komend do programu IBM MQ for z/OS”](#) na stronie 428,

Jednak nie wszystkie komendy mogą być wydawane ze wszystkich tych źródeł. Komendy mogą być klasyfikowane w zależności od tego, czy mogą być wydawane z:

1

CSQINP1

2

CSQINP2

C

Konsola z/OS

R

Serwer komend i kolejka komend, za pomocą CSQUTIL, CSQINPT, CSQINPX lub aplikacji

W opisach komend w sekcji [Komendy MQSC](#) te źródła są identyfikowane przy użyciu znaków 1, 2, C i R w każdym opisie komendy.

Automatyczna konfiguracja ze skryptu MQSC przy uruchamianiu

Za pomocą programu IBM MQ 9.2.0 można skonfigurować menedżera kolejek w taki sposób, aby automatycznie stosował treść skryptu MQSC lub zestawu skryptów MQSC na każdym uruchomieniu menedżera kolejek.

Za pomocą tej funkcji można korzystać z konfiguracji, która może być modyfikowana, a następnie automatycznie ponownie odtwarza się przy następnym restarcie menedżera kolejek. Na przykład, jeśli skrypt lub skrypty znajdują się na podłączonym dysku, istnieje możliwość scentralizowanej konfiguracji, w której najnowsza wersja zostanie zastosowana do każdego menedżera kolejek w czasie, gdy są one uruchamiane.

Konkretnym scenariuszem, w którym może to być przydatne, jest zapewnienie jednolitego klastra, który zawiera te same definicje we wszystkich menedżerach kolejek w klastrze, poprzez zastosowanie jednego zestawu konfiguracji, który wszyscy mają zastosowanie. Aby uzyskać przykład tego przykładu, należy zapoznać się z sekcji [Tworzenie nowego klastra jednolitego](#).

Zanim zaczniesz

Można użyć następujących elementów:

1. Pojedynczy skrypt i tworzenie pliku tekstowego za pomocą komend MQSC.
2. Zestaw skryptów MQSC:
 - Aby zidentyfikować katalog, w którym będą istnieć konfiguracje,
 - W tym katalogu należy utworzyć pliki, z których każdy ma rozszerzenie `.mqsc`, na przykład `queues.mqsc`.

Biorąc pod uwagę, że ten skrypt jest ponownie stosowany w każdym uruchomieniu menedżera kolejek, ważne jest, aby można było odtworzyć komendy. Na przykład komenda **DEFINE** musi zawierać łańcuch **REPLACE**. W przeciwnym razie komenda zostanie wyświetlona jako niepowodzenie uruchomienia drugiego menedżera kolejek, ponieważ obiekt już istnieje.

Należy zauważyć, że w skrypcie MQSC każdy wiersz poprzedzony przedrostkiem `*` jest traktowany jako komentarz.

Włączanie automatycznej konfiguracji skryptów MQSC

Nowy menedżer kolejek można skonfigurować, używając opcji **-ic** do komendy `crtmqm` i wskazując konkretny plik, lub katalog. Podana wartość jest przechowywana w pliku `qm.ini` w sekcji `AutoConfig`, jako atrybut **MQSCConfig**.

Istniejący menedżer kolejek można skonfigurować w taki sposób, aby włączyć automatyczną konfigurację MQSC, dodając atrybut sekcji `AutoConfig` **MQSCConfig**, wskazując poprawny plik lub katalog. Na przykład:

```
AutoConfig:
MQSCConfig=C:\mq_configuration\uniclus.mqsc
```

Jak działa automatyczna konfiguracja?

Podczas uruchamiania menedżera kolejek konfiguracja identyfikowana przez atrybut sekcji `AutoConfig` **MQSCConfig** jest przekazywana za pomocą sprawdzania poprawności `runmqsc` w celu zapewnienia poprawnej składni, a następnie jest przechowywana w drzewie danych menedżera kolejek w katalogu `autocfg` jako pojedynczy plik `cached.mqsc`.

Gdy przetwarzana jest wiele plików z katalogu, są one przetwarzane w kolejności alfabetycznej, a jeśli zawiera ona komendę MQSC zakończenia lub `quit`, reszta zawartości tego pliku jest pomijana.

Podczas pierwszego uruchomienia menedżera kolejek niezdolność do odczytu pliku lub katalogu lub pliku z niepoprawną składnią MQSC uniemożliwia uruchomienie menedżera kolejek z odpowiednim komunikatem o błędzie zarówno do konsoli, jak i do dziennika błędów menedżera kolejek.

Podczas kolejnych restartów, jeśli plik lub katalog wskazywane jest jako nieczytelny lub zawiera niepoprawną składnię MQSC, używany jest poprzednio buforowany plik, a komunikat zapisany w dzienniku błędów menedżera kolejek podświetla ten plik.

V 9.2.2 W momencie, gdy treść `cached.mqsc` jest stosowana do menedżera kolejek, gdy wszystkie komendy MQSC zostały zastosowane, menedżer kolejek jest włączony dla aplikacji w celu nawiązania połączenia. Dziennik **runmqsc** konfigurowanych konfiguracji jest przechowywany w katalogu błędów menedżera kolejek, jako plik o nazwie `autocfgmqsc.LOG`.

Ponadto wszystkie komendy MQSC, które nie zakończyły się pomyślnie, są rejestrowane w dzienniku błędów menedżera kolejek, co oznacza, że wykonanie komendy nie powiodło się.

Wartości ogólne i znaki o specjalnych znaczeniach

Niektóre znaki, na przykład, ukośnik odwrotny (\) i podwójny cudzysłów (") znaki mają specjalne znaczenie, jeśli są używane z komendami MQSC. Niektóre znaki specjalne mogą być używane z parametrami, które mogą mieć wartości ogólne, ale muszą być określone poprawnie.

Poprzedzaj ukośnik odwrotny (\) i podwójny cudzysłów (") znaki z \, to znaczy wpisz \\ lub \" , jeśli w tekście mają być używane \ lub " .

Wszędzie tam, gdzie parametr może mieć wartość ogólną, jest on wprowadzany do końca z gwiazdką (*), na przykład ABC*. Wartość ogólna oznacza wszystkie wartości rozpoczynające się od; wartość ABC* oznacza wszystkie wartości rozpoczynające się od ABC. Jeśli w wartości są używane znaki wymagające cudzysłówów, to gwiazdka musi być umieszczona wewnątrz cudzysłówów, a więc 'abc*'. Gwiazdka musi być ostatnim lub jedynym znakiem w wartości.

Znak zapytania (?) i dwukropek (:) nie są dozwolone w wartościach ogólnych.

Jeśli konieczne jest użycie dowolnego z tych znaków specjalnych w polu (na przykład jako część opisu), należy ująć cały łańcuch w znaki pojedynczego cudzysłowu.


<i>Tabela 1. Opisy znaków, które mają specjalne znaczenie</i>	
Znak	Opis
	Odstępy są używane jako separatory. Wiele odstępów jest równoznaczne z pojedynczym odstępem, z wyjątkiem łańcuchów ujętych w apostrofy ('). Wszystkie odstępy końcowe w tych atrybutach łańcuchowych, które są oparte na typach MQCHARV, są traktowane jako znaczące.
,	Przecinki są używane jako separatory. Wiele przecinków jest odpowiednikiem jednego przecinka, z wyjątkiem łańcuchów, które są ujęte w apostrofy (').
'	Apostrof wskazuje początek lub koniec łańcucha. IBM MQ pozostawia wszystkie znaki ujęte w znaki cudzysłowu dokładnie w takiej postaci, w której zostały wprowadzone. Znaki zawierające apostrofy nie są uwzględniane podczas obliczania długości łańcucha.
"	Pojedyncze znaki cudzysłowu wewnątrz łańcucha są traktowane przez program IBM MQ jako jeden znak podczas obliczania długości łańcucha, a łańcuch nie jest zakończony.
=	 W systemie z/OS znak równości wskazuje początek wartości parametru, który jest zakończony przecinkiem lub odstępem.
(Nawias otwierający wskazuje początek wartości parametru lub listy wartości.
)	Nawias zamykający wskazuje na koniec wartości parametru lub listy wartości.
:	Dwukropek oznacza zakres włączający. Na przykład: (1: 5) oznacza (1,2,3,4, 5). Notacja ta może być używana tylko w komendach TRACE .

Tabela 1. Opisy znaków, które mają specjalne znaczenie (kontynuacja)

Znak	Opis
*	Gwiazdka oznacza wszystko. Na przykład: DISPLAY TRACE (*) oznacza wyświetlanie wszystkich danych śledzenia, a DISPLAY QUEUE (PAY*) oznacza wyświetlenie wszystkich kolejek o nazwach rozpoczynających się od PAY.

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 nastuchiwania, 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 25.

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.

Programowalne formaty komend są obsługiwane przez:

-  IBM MQ for AIX
-  IBM MQ for IBM i
-  IBM MQ dla Linux
-  IBM MQ for Windows
-  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 IBM MQ : obiektami informacji

uwierzytelniających, kanałami, obiektami nastuchiwania kanałów, listami nazw, definicjami procesów, menedżerami kolejek, kolejkami, usługami i klasami 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 parametry, 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 26](#)
- [“Odpowiedzi PCF w produkcie IBM MQ” na stronie 29](#)
-  [“Rozszerzone odpowiedzi” na stronie 31](#)
- [Reguły dotyczące nazewnictwa obiektów produktu IBM MQ](#)
- [“Sprawdzanie uprawnień dla komend PCF w produkcie IBM MQ” na stronie 32](#)


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 wiele 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

Multi

Kodowanie

Jeśli wysyłany jest system IBM MQ for Multiplatforms , należy ustawić to pole na kodowanie używane dla danych komunikatu. Konwersja jest wykonywana w razie potrzeby.

Multi

CodedCharSetId

Jeśli wysyłasz do systemu IBM MQ for Multiplatforms , to pole należy ustawić na identyfikator kodowanego zestawu znaków używany dla danych komunikatu. Konwersja jest wykonywana w razie potrzeby.

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 wygenerowania przez menedżer kolejek 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 mqPutBag przekształca zawartość podanej 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 worek zawiera komunikat administracyjny, a wywołanie mqAddInquiry zostało użyte do wstawienia wartości do torby, wartość elementu danych MQIASY_COMMAND musi być komendą INQUIRE rozpoznawaną przez interfejs MQAI.

Pełny opis wywołania mqPutBag można znaleźć w sekcji [mqPutBag](#).

Odbieranie komunikatów PCF z określonej kolejki

Aby odebrać komunikat z określonej kolejki, wywołanie mqGetBag pobiera 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 mqGetBag. 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 mqGetBag znajduje się w sekcji [mqGetBag](#).

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 nagłówka PCF komunikatu komendy jest MQCFT_COMMAND, generowane są tylko odpowiedzi standardowe. Takie komendy są obsługiwane na wszystkich platformach z wyjątkiem produktu z/OS. Older applications do not support PCF on z/OS ; the IBM MQ Windows Explorer is one such application (however, the IBM WebSphere MQ 6.0 or later IBM MQ Explorer does support PCF on z/OS).

Jeśli typem nagłówka 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 deskrypcie 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) struktury 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 deskrypcji 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 `MQACF_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 wysłana jest komenda.

Sprawdzanie uprawnień dla komend PCF w produkcji IBM MQ

Po przetworzeniu komendy `PCF UserIdentifier` z deskryptora komunikatu w komunikacie komendy jest używany dla wymaganych sprawdzeń uprawnień do obiektu w produkcji 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 przetwarzana jest komenda. 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 2 na stronie 34](#).

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 nasłuchujący 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 2 na stronie 34](#).

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



<i>Tabela 2. Uprawnienia do obiektu</i>		
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 nastuchiwania 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 nastuchujący kanału	dsp	crt
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

<i>Tabela 2. Uprawnienia do obiektu (kontynuacja)</i>		
Komenda	Upewnienie do obiektu IBM MQ	Upewnienie klasy (dla typu obiektu)
Kopiowanie procesu (zastępowanie) <i>patrz uwaga 1</i>	<i>od: dsp do: chg</i>	crt
Kopiuuj 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
Utwórz usługę	<i>(systemowa kolejka domyślna) dsp</i>	crt
Utwórz usługę (zastępowanie) <i>patrz uwaga 1</i>	<i>(systemowa kolejka domyślna) dsp to: chg</i>	crt
Usuń informacje uwierzytelniające	dsp i dlt	nie dotyczy
Usuń rekord uprawnień	<i>(obiekt menedżera kolejek) chg patrz Uwaga 4</i>	<i>patrz Uwaga 4</i>
Usuń kanał	dsp i dlt	nie dotyczy
Usuń program nastuchujący kanału	dsp i dlt	nie dotyczy

<i>Tabela 2. Uprawnienia do obiektu (kontynuacja)</i>		
Komenda	Uprawnienie do obiektu IBM MQ	Uprawnienie klasy (dla typu obiektu)
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 nastuchujący 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
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	<i>(obiekt menedżera kolejek) chg patrz Uwaga 4</i>	<i>patrz Uwaga 4</i>
Uruchom kanał	ctrl	nie dotyczy
Zamknij kanał	ctrl	nie dotyczy
Zatrzymaj połączenie	(obiekt menedżera kolejek) chg	nie dotyczy
Uruchom proces nastuchujący	ctrl	nie dotyczy

Tabela 2. Uprawnienia do obiektu (kontynuacja)		
Komenda	Uprawnienie do obiektu IBM MQ	Uprawnienie klasy (dla typu obiektu)
Zatrzymaj proces nastuchujący	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](#).

Multi 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 IBM i Linux 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 dla 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

Następujące przykładowe programy demonstrują użycie interfejsu MQAI do wykonywania różnych zadań:

- `amqsaicq.c`: utwórz kolejkę lokalną.
- `amqsaiem.c`: umożliwia wyświetlanie zdarzeń na ekranie za pomocą prostego monitora zdarzeń.
- `amqsailq.c`: służy do drukowania listy wszystkich kolejek lokalnych i ich bieżących głębokości.
- `amqsaicl.c`: drukuje listę wszystkich kanałów i ich typów.

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 38.

Odsyłacze 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.

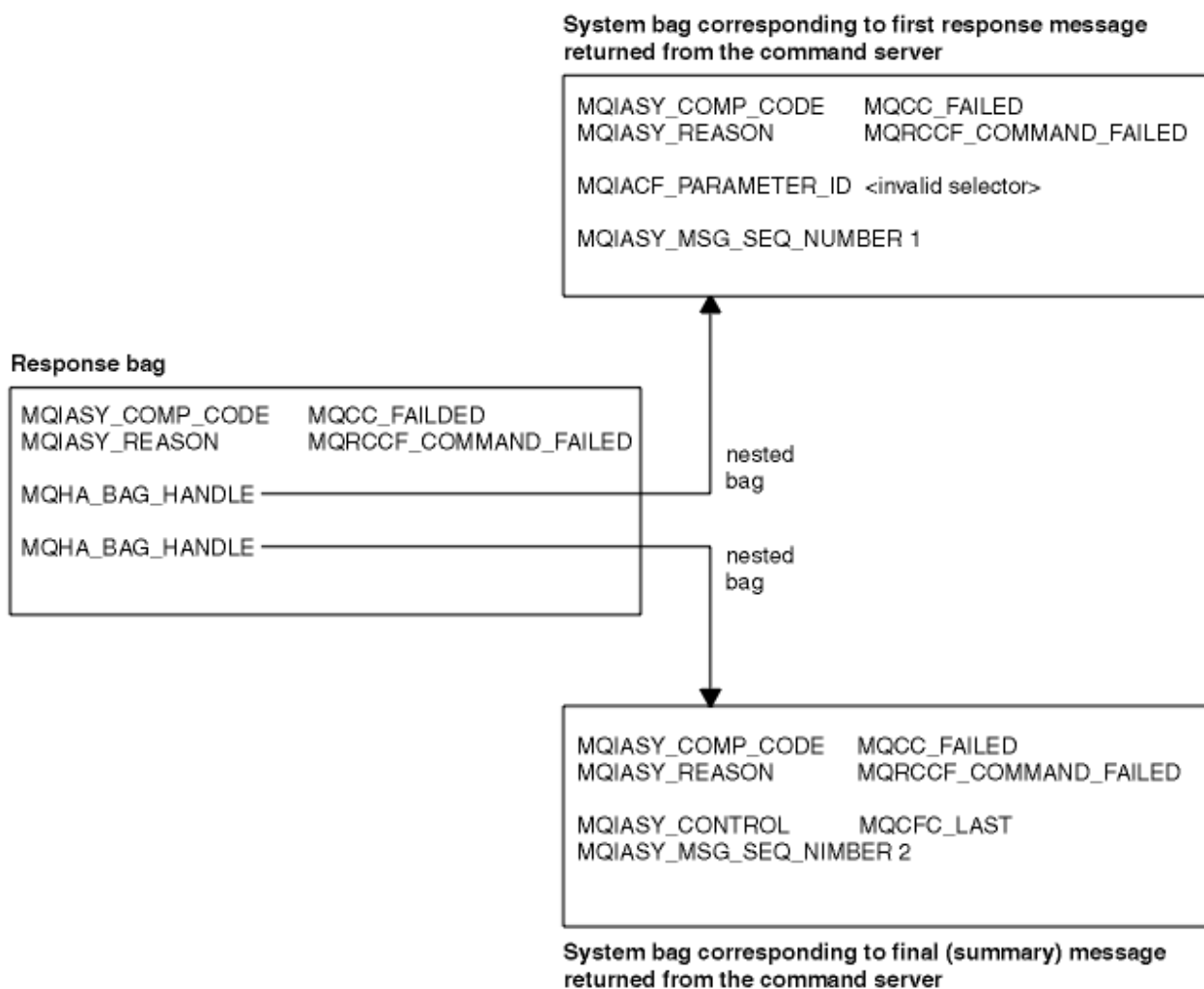
- Łańcuchy znaków w IBM MQ są puste dopełnione do stałej długości. Za pomocą języka C łańcuchy zakończone znakiem o kodzie zero mogą być zwykle dostarczane jako parametry wejściowe do interfejsów programistycznych produktu IBM MQ.
- Aby wyczyścić wartość atrybutu łańcuchowego, należy ustawić go jako pojedynczy pusty łańcuch, a nie pusty łańcuch.
- Należy rozważyć z wyprzedzeniem atrybuty, które mają zostać zmienione, a następnie zapytać o te atrybuty.
- Niektóre atrybuty nie mogą być zmieniane, na przykład nazwa kolejki lub typ kanału. Upewnij się, że użytkownik próbuje zmienić tylko te atrybuty, 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 73 .

Poniższy diagram przedstawia ten scenariusz:



Multi 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.

Konwersja danych

Łańcuchy zawarte w torbie danych MQAI mogą znajdować się w różnych kodowanych zestawach znaków, a te mogą być przekształcane za pomocą wywołania [mqSetInteger](#) .

Użycie deskryptora komunikatu

Interfejs MQAI generuje deskryptor komunikatu, który jest ustawiany na wartość początkową podczas tworzenia worka danych.

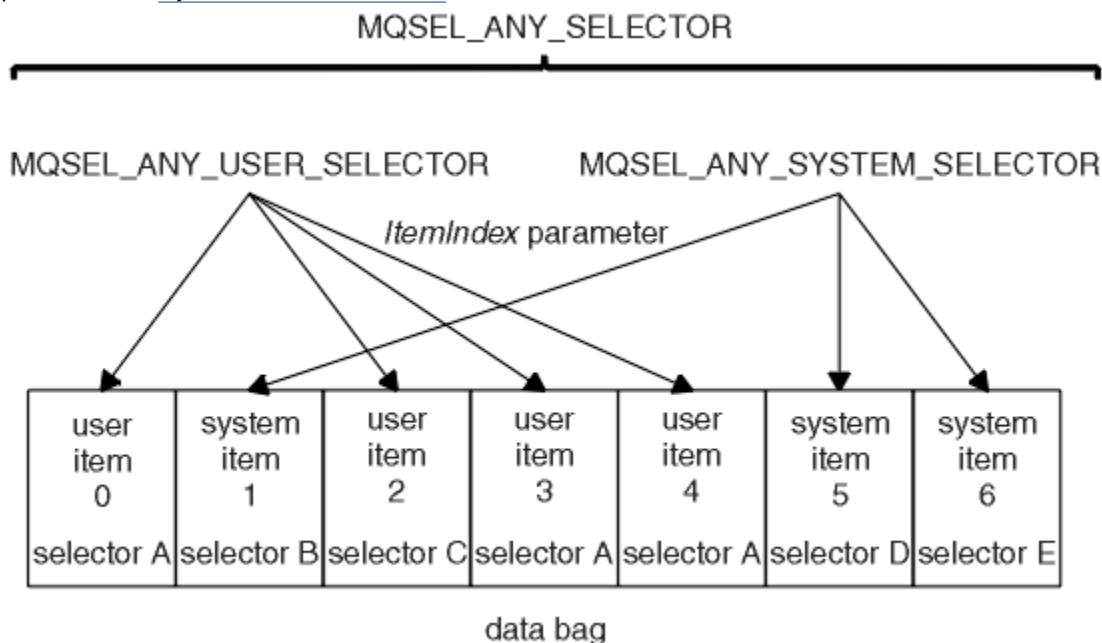
Multi Indeksowanie w interfejsie MQAI

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 3 na stronie 40](#).



Rysunek 3. Indeksowanie

W programie [Rysunek 3 na stronie 40](#) pozycja użytkownika 3 (selektor A) może być przywołana przez następujące pary indeksów:

- selektor A (ItemIndex 1)
- MQSEL_ANY_USER_SELECTOR (ItemIndex 2)
- MQSEL_ANY_SELECTOR (itemIndex 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” na stronie 69](#) i [“Usuwanie elementów danych” na stronie 72](#).

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

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, który jest dodawany do torby, jest uzyskiwany z parametru **Buffer** w wywołaniu `mqAddString` lub `mqSetString`. Identyfikator CCSID jest uzyskiwany z elementu systemu zawierającego selektor tabeli MQIASY_CODED_CHAR_SET_ID. Jest ona znana jako *identyfikator CCSID torby* i może zostać zmieniona za pomocą wywołania `mqSetInteger`.

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

Tabela 3 na stronie 41 przedstawia reguły stosowane podczas przekształcania worków danych w komunikaty i odwrotnie:

<i>Tabela 3. 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
mqBufferToBagmqBufferToBag	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 wysłanym 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

Tabela 3. Przetwarzanie CCSID (kontynuacja)			
Wywołanie MQAI	CCSID	Dane wejściowe do wywołania	Dane wyjściowe do wywołania
TorbamqGet	Identyfikatory CCSID łańcuchów w komunikacie	Wykorzystane	Niezmienione
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.

Pojęcia pokrewne

“Konwersja danych między kodowanymi zestawami znaków” na stronie 199

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.

“Plik ccsid_part2.tbl” na stronie 202

Plik ccsid_part2.tbl jest używany do dostarczania dodatkowych informacji o identyfikatorze CCSID. Plik ccsid_part2.tbl zastępuje plik ccsid.tbl, który był używany przed plikiem IBM MQ 9.0.

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:

Tabela 4. Typ komendy PCF	
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 4 na stronie 43.

Tabela 5. Parametry formatu i MsgType deskryptora MQMD		
Typ komendy PCF	Format	Typ komunikatu
MQCFT_COMMAND	ADMINISTRATOR MQFMT_ADMIN	MQMT_REQUEST
RAPORT MQCFT_REPORT	ADMINISTRATOR MQFMT_ADMIN	Raport_menedzera_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 5 na stronie 43 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 przedstawione w sekcji Tabela 6 na stronie 43.

Tabela 6. 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: Tabela 5 na stronie 43
<i>Expiry</i>	30 sekund (uwaga "1" na stronie 44)
<i>Feedback</i>	MQFB_NONE
<i>Encoding</i>	MQENC_NATIVE
<i>CodedCharSetId</i>	zależy od identyfikatora CCSID torby (uwaga "2" na stronie 44)
<i>Format</i>	Patrz: Tabela 5 na stronie 43
<i>Priority</i>	MQPRI_PRIORITY_AS_Q_DEF

Tabela 6. Wartości deskryptora komunikatu (kontynuacja)

Parametr	Wartość
Persistence	MQPER_NOT_PERSISTENT
MsgId	MQMI_NONE
CorrelId	MQCI_NONE
BackoutCount	0
ReplyToQ	patrz uwaga "3" na stronie 44
ReplyToQMgr	Wartość pusta

Uwagi:

1. Ta wartość może zostać przestonięta w wywołaniu mqExecute za pomocą parametru **OptionsBag**. Więcej informacji na ten temat zawiera sekcja mqExecute.
2. Patrz "Przetwarzanie konwersji danych w interfejsie MQAI" na stronie 41.
3. Nazwa kolejki odpowiedzi określonej przez użytkownika lub kolejki tymczasowej wygenerowanej przez produkt MQAI dla komunikatów typu MQMT_REQUEST. W przeciwnym razie puste.

Multi **Przykładowy program w języku C do tworzenia kolejki lokalnej (amqsaicq.c)**

Przykładowy program w języku C amqsaicq.c tworzy kolejkę lokalną przy użyciu 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, 2024.
/*
/*****
/*
/* 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 <cmqcfc.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 == MQRCC_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);
}

```

Multi Przykładowy program w języku C do wyświetlania zdarzeń za pomocą monitora zdarzeń (amqsaiem.c)

Przykładowy program w języku 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, 2024. 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, &itemCount, &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 */
            &itemType, /* 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;
}

```

Multi Przykładowy program w języku C do uzyskiwania informacji o obiektach kanału (amqsaicl.c)

Przykładowy program w języku C amqsaicl.c zapytuje 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, 2024. 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 <cmqfc.h> /* PCF */

```

```

#include <cmqbc.h> /* MQAI */
#include <cmqxc.h> /* MQCD */

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

/*****
/* DataTypes */
/*****
#if (MQAT_DEFAULT == MQAT_OS400)
typedef _RFILE OUTFILEHDL;
#else
typedef FILE OUTFILEHDL;
#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 */
/*****
#if (MQAT_DEFAULT == MQAT_OS400)
#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, &cAttrsBag,
                &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,
                ch1Name, &ch1NameLength, 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, &ch1Type,
                &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 /* Failed mqExecute */
{
    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);
}

```

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

Przykładowy program w języku C amqsailq.c pyta o bieżące zapewnienie 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, 2024.
/*
/*****/
/*
/* 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 <cmqcfc.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 depths. */
/* 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);
}

```

Multi Torby z danymi i MQAI

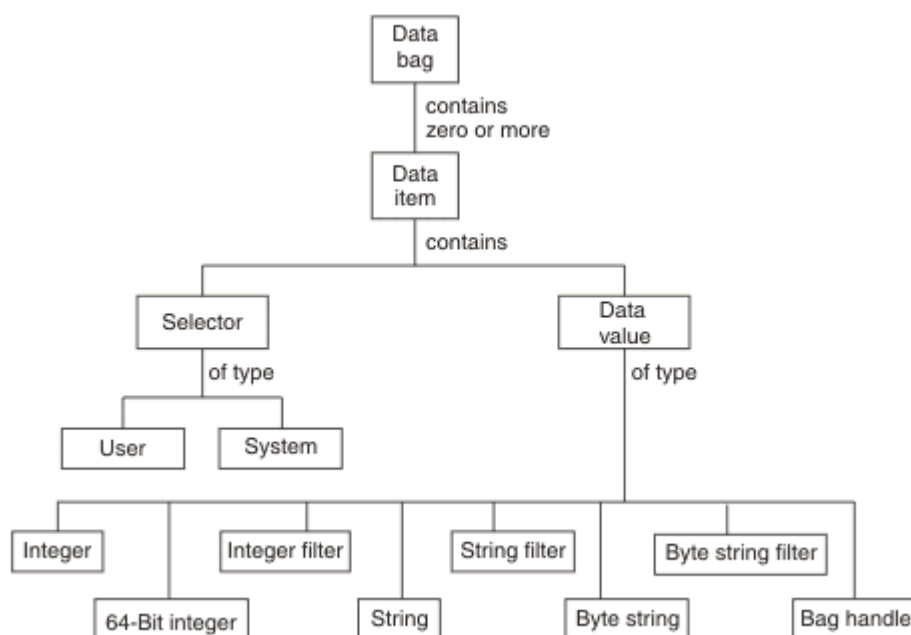
Worek danych jest środkiem do obsługi właściwości lub parametrów obiektów za pomocą interfejsu administracyjnego IBM MQ (MQAI).

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 66

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 40.

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



Rysunek 4. 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 65.

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 65.

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 65.
- Istnieje możliwość wysyłania danych między aplikacjami przy użyciu worków danych [“Umieszczanie i odbieranie worków danych przy użyciu interfejsu MQAI”](#) na stronie 66.
- Elementy danych można dodawać do toreb danych [“Dodawanie elementów danych do worków za pomocą interfejsu MQAI”](#) na stronie 67.
- Komendę uzyskiwania informacji można dodać w worku danych [“Dodawanie komendy uzyskiwania informacji do torby”](#) na stronie 68.
- Istnieje możliwość uzyskania informacji o danych w obrębie toreb danych [“Sprawdzanie w obrębie toreb danych”](#) na stronie 69.
- Istnieje możliwość zliczania elementów danych w obrębie worka danych [“Zliczanie elementów danych”](#) na stronie 72.
- Istnieje możliwość zmiany informacji w obszarze danych [“Zmiana informacji w torbie”](#) na stronie 69.
- Użytkownik może wyczyścić worek danych [“Kasowanie torby przy użyciu wywołania mqClearBag”](#) na stronie 71.
- Można obciąć worek danych [“Obcinanie torby przy użyciu wywołania mqTruncateBag”](#) na stronie 71.
- Istnieje możliwość przekształcenia worków i buforów [“Przekształcanie worków i buforów”](#) na stronie 71.

Multi

Tworzenie i usuwanie worków danych

Tworzenie worków danych

Aby korzystać z interfejsu MQAI, należy najpierw utworzyć worek danych przy użyciu wywołania `mqCreateBag`. 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 66.
- 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 z komendy `mqCreateBag` i musi być podany we wszystkich innych wywołaniach, w których używany jest worek danych.

Pełny opis wywołania `mqCreateBag` znajduje się w sekcji [mqCreateBag](#).

Usuwanie worków danych

Wszystkie elementy, które są tworzone przez użytkownika, również muszą zostać usunięte przy użyciu wywołania `mqDeleteBag`. 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 73. Kod użytkownika nie może usunąć torby systemowej.

Pełny opis wywołania `mqDeleteBag` znajduje się w sekcji [mqDeleteBag](#).

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 28

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 67.

Typy elementów danych

Po utworzeniu worka danych można je zapęścić 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 72.
- [“Usuwanie elementów danych”](#) na stronie 72.
- [“Dodawanie elementów danych do worków za pomocą interfejsu MQAI”](#) na stronie 67.
- [“Filtrowanie i odpytywanie elementów danych”](#) na stronie 68.

Elementy systemowe i MQAI

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

- Generowanie nagłówków 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 69), ale użytkownik nie może zmienić opcji tworzenia worka. Nie można usuwać elementów systemowych. (Patrz [“Usuwanie elementów danych”](#) na stronie 72).

Dodawanie elementów danych do worków za pomocą interfejsu MQAI

Gdy worek danych jest tworzony przy użyciu interfejsu administracyjnego IBM MQ (MQAI), można je zapęścić 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 66.

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 5 na stronie 68](#). Elementy są identyfikowane przez selektor. Zwykle jeden selektor identyfikuje tylko jeden element, ale nie zawsze jest to przypadek. Jeśli element danych o określonym selektorze jest już obecny w torbie, dodatkowa instancja tego selektora jest dodawana do końca torby.



Rysunek 5. 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 `mqAddByteStringFilter` zgodnie z opisem w sekcji [mqAddByteStringFilter](#) .

Więcej informacji na temat dodawania elementów danych do torby zawiera sekcja [“Elementy systemowe i MQAI” na stronie 67](#).

Multi Dodawanie komendy uzyskiwania informacji do torby

Wywołanie zapytania `mqAdd` służy 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 `mqAdd` można znaleźć w sekcji [ZapytaniemqAdd](#).

Multi 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 kolejki), 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 25. 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, ...)
```

Multi

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](#).

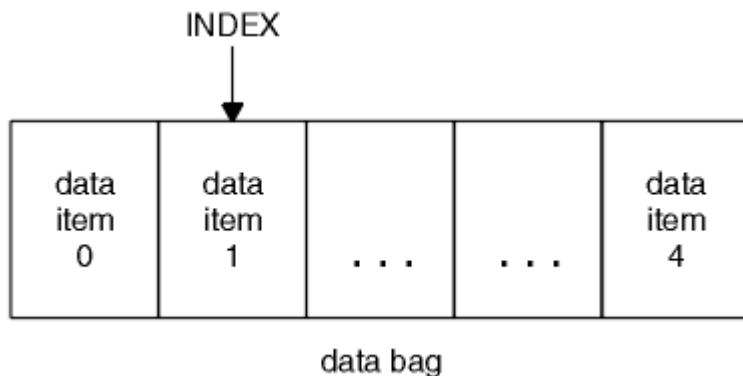
Można również zapytać o typ (liczba całkowita, 64-bitowa liczba całkowita, filtr liczb całkowitych, łańcuch znaków, filtr łańcuchowy, łańcuch bajtowy, filtr łańcucha bajtowego lub uchwyt worka) konkretnego elementu przy użyciu wywołania `mqInquireItemInfo`. Patrz [mqInquireItemInfo](#).

Multi

Zmiana informacji w torbie

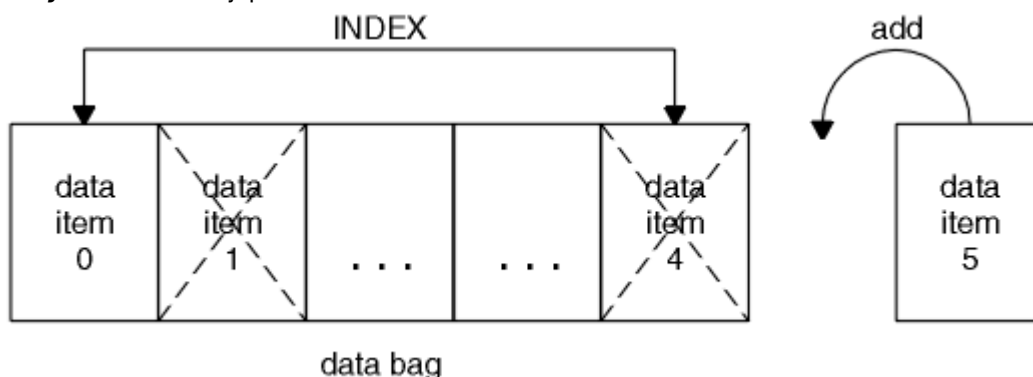
Interfejs MQAI umożliwia zmianę informacji w torbie przy użyciu wywołań `mqSet` *. Dostępne są następujące możliwości:

1. Modyfikowanie elementów danych w torbie. Indeks pozwala na zastąpienie pojedynczej instancji parametru identyfikując wystąpienie elementu do zmodyfikowania (patrz Rysunek 6 na stronie 70).



Rysunek 6. Modyfikowanie pojedynczego elementu danych

2. Usuń wszystkie istniejące wystąpienia określonego selektora i dodaj nowe wystąpienie do końca torby. (Patrz sekcja Rysunek 7 na stronie 70). Specjalna wartość indeksu umożliwia zastępowanie **wszystkich** instancji parametru.



Rysunek 7. Modyfikowanie wszystkich elementów danych

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

Wywołanie `mqSetInteger` umożliwia modyfikowanie elementów całkowitoliczbowych w ramach torby. Wywołanie `mqSetInteger64` pozwala na zmodyfikowanie 64-bitowych elementów całkowitych. Wywołanie `mqSetIntegerFilter` umożliwia modyfikowanie elementów filtru liczby całkowitej. Wywołanie `mqSetString` umożliwia modyfikowanie elementów łańcucha znaków. Wywołanie `mqSetStringFilter` umożliwia modyfikowanie elementów filtru łańcuchów. Wywołanie `mqSetByteString` umożliwia modyfikowanie elementów łańcucha bajtowego. Wywołanie `mqSetByteStringFilter` umożliwia modyfikowanie elementów filtru 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:

- [mqSetInteger](#)
- [mqSetInteger64](#)
- [mqSetIntegerFilter](#)
- [mqSetString](#)
- [mqSetStringFilter](#)
- [mqSetByteString](#)
- [mqSetByteStringFilter](#)

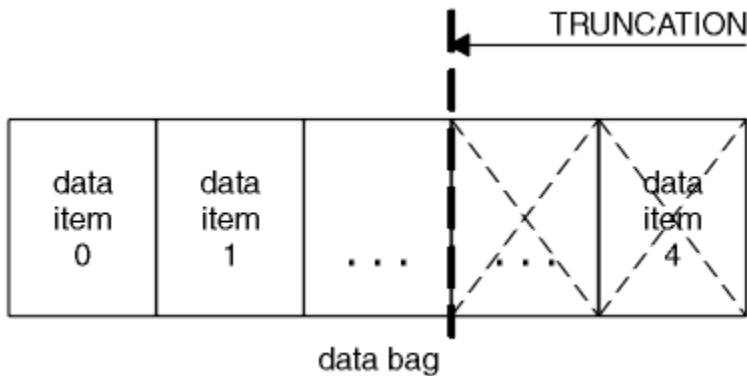
Multi Kasowanie torby przy użyciu wywołania `mqClearBag`

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

Pełny opis wywołania `mqClearBag` znajduje się w sekcji [mqClearBag](#).

Multi Obcinanie torby przy użyciu wywołania `mqTruncateBag`

Wywołanie `mqTruncateBag` redukuje liczbę elementów użytkownika w worku użytkownika, usuwając elementy z końca torby, zaczynając od ostatnio dodanego elementu. Na przykład można użyć tego samego nagłówka w celu wygenerowania więcej niż jednego komunikatu.

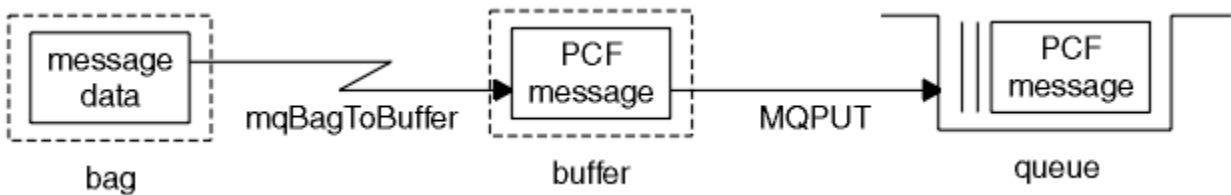


Rysunek 8. Obcinanie torby

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

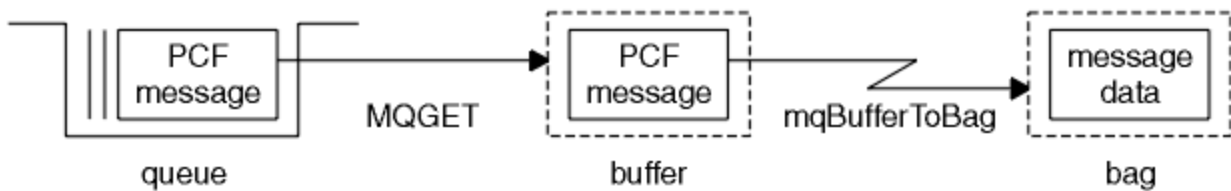
Multi Przekształcanie worków i buforów

Aby wysłać dane między aplikacjami, po pierwsze dane komunikatu są umieszczane w torbie. Następnie dane znajdujące się w torbie są przekształcane w komunikat PCF przy użyciu wywołania `mqBagToBuffer`. Komunikat PCF jest wysyłany do wymaganej kolejki przy użyciu wywołania `MQPUT`. Pokazano to na rysunku Rysunek 9 na stronie 71. Pełny opis wywołania `mqBagToBuffer` znajduje się w sekcji [mqBagToBuffer](#).



Rysunek 9. 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 przy użyciu wywołania `mqBufferToBag`, udostępniając bufor, który zawiera poprawny komunikat PCF. Pokazano to na rysunku Rysunek 10 na stronie 71. Pełny opis wywołania `mqBufferToBag` znajduje się w sekcji [mqBufferToBag](#).



Rysunek 10. Przekształcanie komunikatów PCF w formularz torby

Multi Zliczanie elementów danych

Wywołanie `mqCountItems` zlicza liczbę elementów użytkownika, elementów systemowych lub obu tych elementów, które są zapisane w rekordzie 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` znajduje się w sekcji [mqCountItems](#).

Multi Usuwanie elementów danych

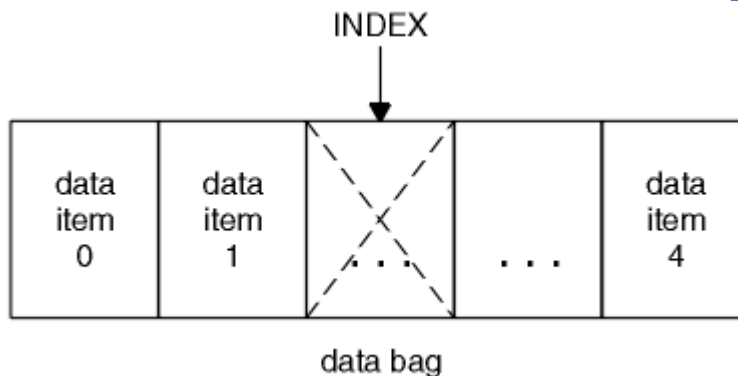
Elementy z worków można usuwać na wiele sposobów. Dostępne są następujące możliwości:

- Usuń jeden lub więcej elementów użytkownika z torby. Szczegółowe informacje na ten temat zawiera sekcja [“Usuwanie elementów danych z torby przy użyciu wywołania `mqDeleteItem`”](#) na stronie 72.
- Usuń wszystkie elementy użytkownika z torby, czyli skasuj torbę. Szczegółowe informacje na ten temat zawiera sekcja [“Kasowanie torby przy użyciu wywołania `mqClearBag`”](#) na stronie 71.
- Usuń elementy użytkownika z końca torby, to znaczy obetnij torbę. Szczegółowe informacje na ten temat zawiera sekcja [“Obcinanie torby przy użyciu wywołania `mqTruncateBag`”](#) na stronie 71.

Multi Usuwanie elementów danych z torby przy użyciu wywołania `mqDeleteItem`

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

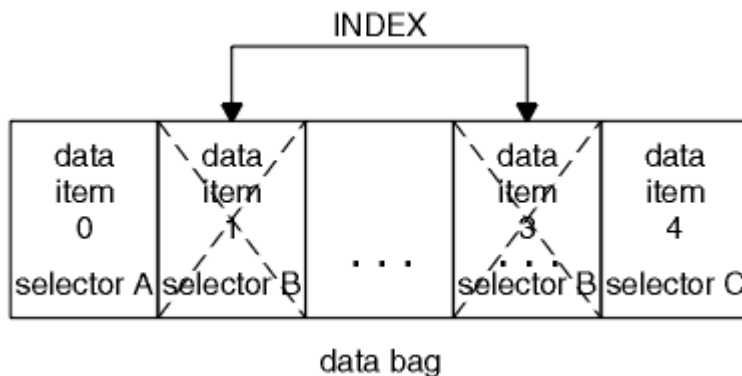
1. Pojedyncze wystąpienie określonego selektora. (Patrz sekcja [Rysunek 11](#) na stronie 72).



Rysunek 11. Usuwanie pojedynczego elementu danych

lub wersji

2. Wszystkie wystąpienia określonego selektora. (Patrz sekcja [Rysunek 12](#) na stronie 73).



Rysunek 12. Usuwanie wszystkich elementów danych

Uwaga: Indeks zachowuje kolejność wstawiania w worku, ale może wpływać na indeksy innych elementów danych. Na przykład wywołanie `mqDeleteItem` nie 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 `mqDeleteItem` można znaleźć w sekcji [mqDeleteItem](#).

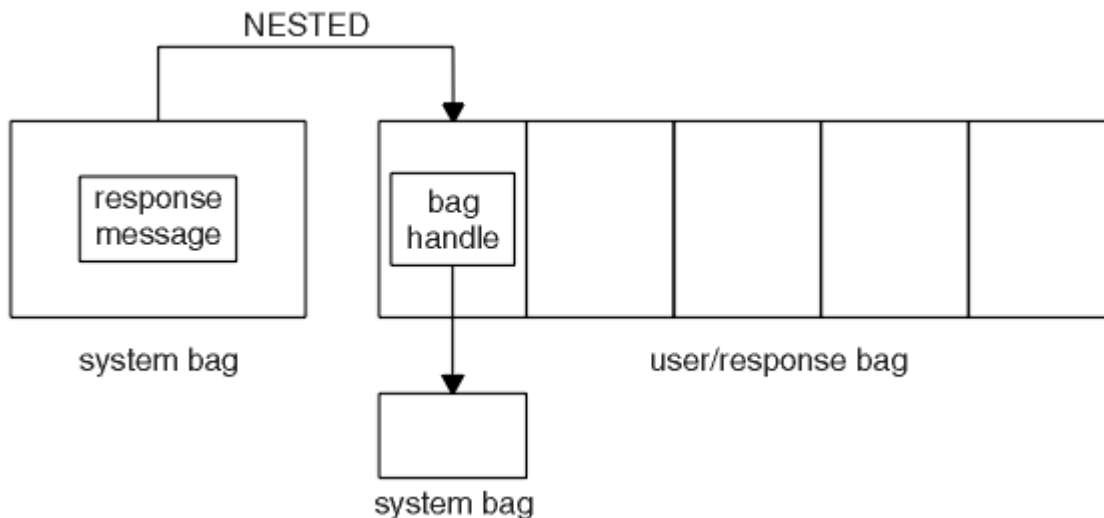
Multi Wysyłanie komend administracyjnych do serwera komend qm przy użyciu wywołania `mqExecute`

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 13 na stronie 74](#).



Rysunek 13. Zagnieżdzenie

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](#).

Administrowanie przy użyciu REST API

Za pomocą administracyjnej REST API można administrować obiektami produktu IBM MQ, takimi jak menedżery kolejek i kolejki, a także agenty i transfery produktu Managed File Transfer. Informacje są wysyłane do administracyjnej REST API i odbierane z niej 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 rozpoczniesz

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

Procedura

- [“Pierwsze kroki w produkcie administrative REST API” na stronie 75](#)
- [“Korzystanie z administrative REST API” na stronie 79](#)
- [“Zdalne administrowanie przy użyciu REST API” na stronie 80](#)
- [“REST API datowniki” na stronie 85](#)
- [“REST API obsługa błędów” na stronie 85](#)
- [“REST API Wykrywanie” na stronie 88](#)
- [“REST API Obsługa języków narodowych” na stronie 89](#)


Pierwsze kroki w produkcie administrative REST API


Szybkie rozpoczęcie pracy z serwerem administrative REST API i wypróbowanie kilku przykładowych żądań przy użyciu komendy cURL w celu utworzenia, zaktualizowania, wyświetlenia i usunięcia kolejki.

Zanim rozpoczniesz

Aby rozpocząć korzystanie z produktu administrative REST API, przykłady w tej czynności mają następujące wymagania:

- W przykładach cURL używa się do tworzenia żądań REST w celu wyświetlania informacji o menedżerach kolejek w systemie oraz do tworzenia kolejki, aktualizowania, wyświetlania i usuwania kolejek. Dlatego aby wykonać to zadanie, należy zainstalować w systemie cURL .
- Aby wykonać tę czynność, użytkownik musi mieć pewne uprawnienia pozwalające na użycie komendy **dspmweb**:

–  W systemie z/OS użytkownik musi mieć uprawnienia do uruchamiania komendy **dspmweb** i dostęp z uprawnieniami do zapisu do pliku `mqwebuser.xml`.


–  W przypadku wszystkich innych systemów operacyjnych użytkownik musi być [użytkownikiem uprzywilejowanym](#).

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

Procedura

1. Jeśli serwer mqweb nie jest jeszcze skonfigurowany do użycia przez produkt administrative REST API, produkt administrative REST API for MFT, produkt messaging REST API lub produkt IBM MQ Console, skonfiguruj serwer mqweb.

Więcej informacji na temat tworzenia podstawowej konfiguracji serwera mqweb z podstawowym rejestrem zawiera sekcja [Podstawowa konfiguracja serwera mqweb](#).

2.  W systemie z/OS ustaw zmienną środowiskową `WLP_USER_DIR` tak, aby można było używać komendy **dspmweb** . Należy ustawić zmienną tak, aby wskazywała konfigurację serwera mqweb, wprowadzając następującą komendę:

```
export WLP_USER_DIR=WLP_user_directory
```

, gdzie `WLP_user_directory` jest nazwą katalogu, który jest przekazywany do `crtmqweb`. Na przykład:

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

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

3. Określ adres URL REST API , wprowadzając następującą komendę:

```
dspmqweb status
```

W przykładach przedstawionych w poniższych krokach przyjęto, że adres URL REST API jest domyślnym adresem URL `https://localhost:9443/ibmmq/rest/v1/`. Jeśli używany jest adres URL inny niż domyślny, należy podać go w poniższych krokach.

4. Wypróbuj żądanie GET dla zasobu `qmgr` , korzystając z podstawowego uwierzytelniania przy użyciu użytkownika `mqadmin` :

```
curl -k https://localhost:9443/ibmmq/rest/v2/admin/qmgr -X GET -u mqadmin:mqadmin
```

5. Tworzenie, wyświetlanie, modyfikowanie i usuwanie kolejki przy użyciu zasobu `mqsc` :

W tym przykładzie używany jest menedżer kolejek `QM1`. Utwórz menedżer kolejek o tej samej nazwie lub zastąp istniejący menedżer kolejek w systemie.

- a) Wykonaj żądanie POST dla zasobu `mqsc` , aby utworzyć kolejkę lokalną:

W treści żądania nazwa nowej kolejki jest ustawiona na `Q1`. Używane jest uwierzytelnianie podstawowe, a nagłówek HTTP produktu `ibm-mq-rest-csrf-token` z dowolną wartością jest ustawiany w żądaniu REST cURL . Ten dodatkowy nagłówek jest wymagany dla żądań POST, PATCH i DELETE:

```
curl -k https://localhost:9443/ibmmq/rest/v2/admin/action/qmgr/QM1/mqsc -X POST -u mqadmin:mqadmin -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json" --data '{"type": "runCommandJSON", "command": "define", "qualifier": "qlocal", "name": "Q1"}'
```

- b) Wykonaj żądanie POST dla zasobu `mqsc` , aby wyświetlić kolejkę lokalną utworzoną w kroku [“5.a”](#) na stronie 76:

```
curl -k https://localhost:9443/ibmmq/rest/v2/admin/action/qmgr/QM1/mqsc -X POST -u mqadmin:mqadmin -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json" --data '{"type": "runCommandJSON", "command": "display", "qualifier": "qlocal", "name": "Q1"}'
```

- c) Wykonaj żądanie POST dla zasobu `mqsc` , aby zasób mógł zaktualizować opis kolejki:

```
curl -k https://localhost:9443/ibmmq/rest/v2/admin/action/qmgr/QM1/mqsc -X POST -u mqadmin:mqadmin -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json" --data '{"type": "runCommandJSON", "command": "alter", "qualifier": "qlocal", "name": "Q1", "parameters": {"descr": "new description"}'}
```

- d) Utwórz żądanie POST dla zasobu `mqsc` , aby wyświetlić nowy opis kolejki. Określ atrybut **responseParameters** w treści żądania, tak aby odpowiedź obejmował pole opisu:

```
curl -k https://localhost:9443/ibmmq/rest/v2/admin/action/qmgr/QM1/mqsc -X POST -u mqadmin:mqadmin -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json" --data '{"type": "runCommandJSON", "command": "display", "qualifier": "qlocal", "name": "Q1", "responseParameters": [{"descr"}]}'
```

- e) Wykonaj żądanie POST dla zasobu `mqsc` , aby usunąć kolejkę:

```
curl -k https://localhost:9443/ibmmq/rest/v2/admin/action/qmgr/QM1/mqsc -X POST -u mqadmin:mqadmin -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json" --data '{"type": "runCommandJSON", "command": "delete", "qualifier": "qlocal", "name": "Q1"}'
```

- f) Wykonaj żądanie POST w zasobie `mqsc` , aby udowodnić, że kolejka została usunięta:

```
curl -k https://localhost:9443/ibmmq/rest/v2/admin/action/qmgr/QM1/mqsc -X POST -u mqadmin:mqadmin -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json" --data '{"type": "runCommandJSON", "command": "display", "qualifier": "qlocal", "name": "Q1"}'
```



Co dalej

- W przykładach do zabezpieczenia żądania użyto uwierzytelniania podstawowego. Zamiast tego można użyć uwierzytelniania opartego na znacznikach lub uwierzytelniania opartego na kliencie. Więcej informacji na ten temat zawiera sekcja [Korzystanie z uwierzytelniania przy użyciu certyfikatu klienta za pomocą interfejsu REST API i konsoli IBM MQ Console](#) oraz sekcja [Korzystanie z uwierzytelniania opartego na znacznikach za pomocą interfejsu REST API](#).
- Więcej informacji na temat korzystania z produktu administrative REST API i konstruowania adresów URL z parametrami zapytania: [“Korzystanie z administrative REST API” na stronie 79](#).
- Przejrzyj informacje uzupełniające dotyczące dostępnych zasobów produktu administrative REST API i wszystkich dostępnych opcjonalnych parametrów zapytania: [Skorowidz produktu administrative REST API](#).
- Sekcja zawiera informacje na temat sposobu korzystania z produktu administrative REST API do administrowania obiektami IBM MQ w systemach zdalnych: [“Zdalne administrowanie przy użyciu REST API” na stronie 80](#).
- Learn how to use the administrative REST API with MFT: [“Pierwsze kroki z REST API dla MFT” na stronie 77](#).
- Wykryj messaging REST API, interfejs RESTful dla przesyłania komunikatów produktu IBM MQ : [Przesyłanie komunikatów za pomocą REST API](#).
- Wykryj IBM MQ Console, oparty na przeglądarce interfejs GUI: [“Administrowanie przy użyciu konsoli WWW” na stronie 92](#).


Pierwsze kroki z REST API dla MFT

Szybkie rozpoczęcie pracy z programem administrative REST API for Managed File Transfer i wypróbowanie kilku przykładowych żądań w celu wyświetlenia statusu agenta MFT oraz wyświetlenia listy transferów.

Zanim rozpoczniesz

- W przykładach cURL jest używany do wysyłania żądań REST w celu wyświetlenia listy transferów i wyświetlenia statusu agenta MFT . Dlatego aby wykonać to zadanie, należy zainstalować w systemie cURL .
- Aby wykonać tę czynność, użytkownik musi mieć pewne uprawnienia pozwalające na użycie komendy **dspmweb**:
 -  W systemie z/OS użytkownik musi mieć uprawnienia do uruchamiania komendy **dspmweb** i dostęp z uprawnieniami do zapisu do pliku mqwebuser.xml.
 -  W przypadku wszystkich innych systemów operacyjnych użytkownik musi być [użytkownikiem uprzywilejowanym](#).

Procedura

1. Upewnij się, że serwer mqweb jest skonfigurowany dla produktu administrative REST API dla produktu MFT:
 - Jeśli serwer mqweb nie jest jeszcze skonfigurowany do użycia przez produkt administrative REST API, produkt administrative REST API for MFT, produkt messaging REST API lub produkt IBM MQ Console, skonfiguruj serwer mqweb. Więcej informacji na temat tworzenia podstawowej konfiguracji serwera mqweb z podstawowym rejestrem zawiera sekcja [Podstawowa konfiguracja serwera mqweb](#).
 - Jeśli serwer mqweb jest skonfigurowany, należy upewnić się, że krok 8 [Konfiguracja podstawowa dla serwera mqweb](#) został zakończony, aby umożliwić administrative REST API dla produktu MFT.
2. 

W systemie z/OSustaw zmienną środowiskową `WLP_USER_DIR` tak, aby można było używać komendy **dspmweb**. Należy ustawić zmienną tak, aby wskazywała konfigurację serwera mqweb, wprowadzając następującą komendę:

```
export WLP_USER_DIR=WLP_user_directory
```

, gdzie `WLP_user_directory` jest nazwą katalogu, który jest przekazywany do `crtmqweb`. Na przykład:

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

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

3. Określ adres URL REST API , wprowadzając następującą komendę:

```
dspmweb status
```

W przykładach przedstawionych w poniższych krokach przyjęto, że adres URL REST API jest domyślnym adresem URL `https://localhost:9443/ibmmq/rest/v1/`. Jeśli używany jest adres URL inny niż domyślny, należy podać go w poniższych krokach.

4. Wykonaj żądanie GET w zasobie `agent` , aby zwrócić podstawowe informacje o wszystkich agentach, w tym nazwę, typ i stan:

```
curl -k https://localhost:9443/ibmmq/rest/v2/admin/mft/agent/ -X GET -u mftadmin:mftadmin
```

5. Utwórz kilka operacji przesyłania, które mają być wyświetlane za pomocą komendy **fteCreateTransfer** .

Serwer mqweb buforuje informacje o transferach i zwraca te informacje, gdy żądanie jest wykonywane. Ta pamięć podręczna jest resetowana po zrestartowaniu serwera mqweb. Można sprawdzić, czy serwer został zrestartowany, wyświetlając pliki `console.log` i `messages.log` , czy z/OS, przeglądając dane wyjściowe z uruchomionego zadania.

6. Wykonaj żądanie GET w zasobie `transfer` , aby zwrócić szczegółowe informacje o maksymalnie czterech transferach, które zostały wykonane od momentu uruchomienia serwera mqweb:

```
curl -k https://localhost:9443/ibmmq/rest/v2/admin/mft/transfer?limit=4 -X GET -u mftadmin:mftadmin
```

Co dalej

- W przykładach do zabezpieczenia żądania użyto uwierzytelniania podstawowego. Zamiast tego można użyć uwierzytelniania opartego na znacznikach lub uwierzytelniania opartego na kliencie. Więcej informacji na ten temat zawiera sekcja [Korzystanie z uwierzytelniania opartego na znacznikach za pomocą interfejsu REST API](#) oraz sekcja [Korzystanie z uwierzytelniania przy użyciu certyfikatu klienta za pomocą interfejsu REST API i konsoli IBM MQ Console](#).
- Więcej informacji na temat korzystania z produktu administrative REST API i konstruowania adresów URL z parametrami zapytania: [“Korzystanie z administrative REST API” na stronie 79](#).
- Przejrzyj informacje uzupełniające dla dostępnych zasobów administrative REST API dla zasobów MFT i wszystkich dostępnych opcjonalnych parametrów zapytania: [Skorowidz produktu administrative REST API](#).
- Wykryj messaging REST API, interfejs RESTful dla przesyłania komunikatów produktu IBM MQ : [Przesyłanie komunikatów za pomocą REST API](#).
- Wykryj IBM MQ Console, oparty na przeglądarce interfejs GUI: [“Administrowanie przy użyciu konsoli WWW” na stronie 92](#).

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:

- Aby można było korzystać z produktu administrative REST API, należy uwierzytelnić się na serwerze mqweb. Uwierzytelnianie można uwierzytelnić za pomocą podstawowego uwierzytelniania HTTP, uwierzytelniania za pomocą 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 79 i [“1.b”](#) na stronie 79.

Procedura

1. Utwórz adres URL:

- a) Określ adres URL przedrostka, wprowadzając następującą komendę:

```
dspmweb status
```

Adres URL, który ma być używany, zawiera frazę /ibmmq/rest/.

- b) Dodaj zasób do ścieżki adresu URL.

Dostępne są następujące zasoby produktu IBM MQ :

- [/admin/instalacja](#)
- [/admin/qmgr](#)
- [/admin/queue](#)
- [/admin/subskrypcja](#)
- [/admin/channel](#)
- [/action/qmgr/{qmgrname}/mqsc](#)

Dostępne są następujące zasoby produktu Managed File Transfer :

- [/admin/agent](#)
- [/admin/transfer](#)
- [/admin/monitor](#)

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

```
https://localhost:9443/ibmmq/rest/v2/admin/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:

```
https://localhost:9443/ibmmq/rest/v2/admin/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:

```
https://localhost:9443/ibmmq/rest/v2/admin/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 JSON i 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.

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 administracyjną 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 zdalnym 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 75](#).
- 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 administrować, musi mieć wartość IBM MQ 8.0 lub późniejszą.
- 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`
 - `/channel`
 - `/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 za pomocą komendy **setmqweb** w celu ustawienia atrybutu `mqRestGatewayQmgr` na nazwę menedżera kolejek gatewaya lub wysyłając nazwę 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 [“Konfigurowanie menedżerów kolejek na potrzeby zdalnego administrowania” na stronie 192](#).
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. Istnieje możliwość skonfigurowania domyślnego menedżera kolejek bramy, określenia menedżera kolejek bramy w nagłówku HTTP lub użycia kombinacji obu podejść:
- 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.

Jeśli masz trudności z używaniem administracyjnego REST API do administrowania zdalnymi menedżerami kolejek:

- Sprawdź, czy zdalny menedżer kolejek jest uruchomiony.
- Sprawdź, czy serwer komend jest uruchomiony w systemie zdalnym.
- Sprawdź, czy interwał odłączania kanału nie utracił ważności. Na przykład, jeśli kanał został uruchomiony, ale później został zamknięty po pewnym czasie. Jest to szczególnie ważne w przypadku ręcznego uruchomienia kanałów.

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 nastuchiwanie, kanały i kolejki są skonfigurowane i uruchomione:

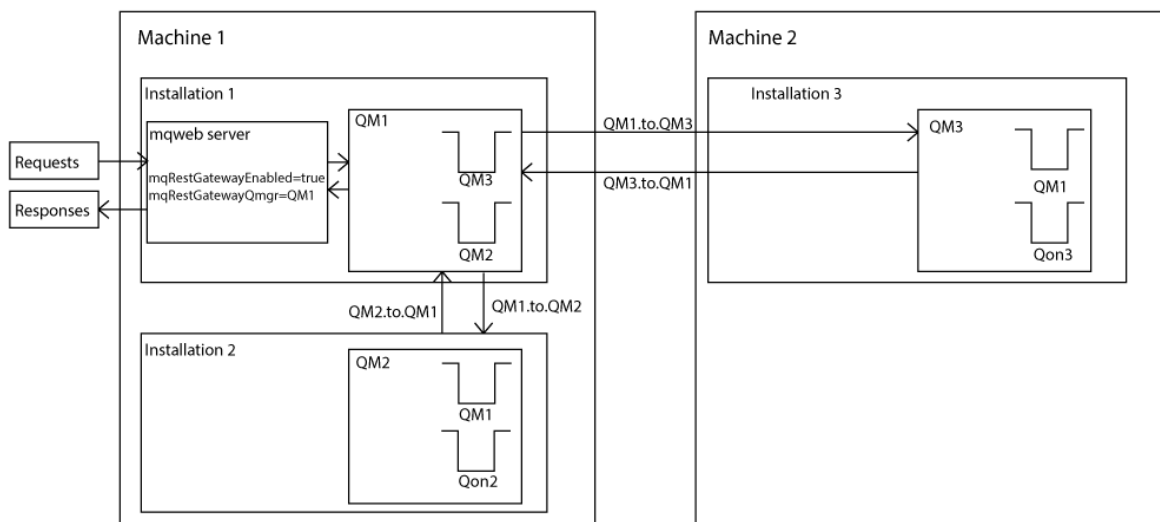
- W menedżerze kolejek QM1, w programie Installation 1, w systemie Machine 1:
 - Kanał nadawczy QM1.to.QM2
 - Kanał odbiorczy QM2.to.QM1
 - Kanał nadawczy QM1.to.QM3
 - Kanał odbiorczy QM3.to.QM1
 - Kolejka transmisji QM2
 - Kolejka transmisji QM3
 - Nastuchiwanie skonfigurowane na porcie 1414
- W menedżerze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

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 14. 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"
  }
}
```

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).

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, który wskazuje, że komunikat pochodzi z interfejsu API REST produktu IBM MQ .

nnnn

Unikalny numer identyfikujący komunikat.

X

Pojedyncza litera oznaczająca 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 kolejek. 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.

completionCode

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.

 W produkcie IBM MQ 9.1.2 to pole nie jest już zwracane.

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 współużytkowania 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"
}]}
```

REST API Wykrywanie

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

Ważne: Funkcja wykrywania interfejsu API została ustabilizowana i nadal można korzystać z tej funkcji. Produkt IBM MQ nie obsługuje obecnie korzystania z opcji mpOpenAPI .

Należy włączyć zabezpieczenia serwera mqweb, aby wyświetlić dokumentację Swagger za pomocą wykrywania interfejsu API. Więcej informacji na temat kroków, które są wymagane do włączenia zabezpieczeń, zawiera sekcja [Zabezpieczenia produktu IBM MQ Console i REST API](#).

Procedura

1. Znajdź plik mqwebuser.xml w jednym z następujących katalogów:

- ▶ **ALW** MQ_DATA_PATH/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** 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> :
- 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. Treścią tego nagłówka może być dowolny łańcuch, w tym pusty.

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 . Aby wyświetlić adres URL, należy użyć komendy **dspmqweb status** .

REST API Obsługa języków narodowych

Produkt REST API obsługuje, z pewnymi kwalifikacjami, możliwość określania języków narodowych jako części żądania HTTP.

Tło

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 89 wyjaśnia, jakie dane są tłumaczone.

Wskazywanie 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 obsługiwane 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 nieobskowanego 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.

Wersja REST API

Numer wersji produktu REST API jest częścią podstawowego adresu URL dla żądań REST. Na przykład: <https://localhost:9443/ibmmq/rest/v2/admin/installation>. Numer wersji jest używany do izolowania klientów od zmian w REST API, które mogą zostać wprowadzone w przyszłych wersjach.

V 9.2.0 W produkcie IBM MQ 9.2.0 wprowadzono wersję 2 produktu REST API. This version increase applies to the administrative REST API, messaging REST API, and MFT REST API. Ta zmiana

wersji powoduje zmianę adresu URL zasobu, który jest używany dla REST API. Przedrostkiem adresu URL dla adresów URL zasobów w wersji 2 jest następujący adres URL:

```
https://host:port/ibmmq/rest/v2/
```

Niektóre zmiany wprowadzone w REST API mogą zmienić istniejącą funkcję REST API tak, aby klienci korzystali z REST API, które mogły być aktualizowane. Aby zapobiec aktualizowaniu takich zmian przed wymuszaniem aktualizacji klientów, zwiększa się numer wersji produktu REST API, a istniejąca funkcja jest stabilna w stosunku do poprzedniej liczby. Nowa funkcja, która może zmienić istniejącą funkcję, jest dodawana do REST API w nowym numerze wersji. W związku z tym klienci mogą nadal używać produktu REST API w poprzedniej wersji bez aktualizacji.

Zmiany w produkcie REST API, które mogą spowodować konieczność zaktualizowania klienta, obejmują następujące zmiany:

- Usunięcie obsługi istniejącego atrybutu w formacie JSON, który jest wysyłany do REST API lub zwrócony z niego.
- Usunięcie adresu URL, komendy HTTP lub nagłówka. Na przykład, jeśli nazwa adresu URL lub nagłówek zostanie zmieniona, lub jeśli użyto innego komendy.
- Dodanie nowego obowiązkowego atrybutu JSON do danych, które są wysyłane do istniejącego adresu URL.
- Dodanie nowego obowiązkowego nagłówka HTTP do danych, które są wysyłane do istniejącego adresu URL.
- Dodanie nowego obowiązkowego parametru zapytania do istniejącego adresu URL.

Gdy ten typ zmiany jest wprowadzany do funkcji REST API, która istniała w wersji Long Term Support (LTS), numer wersji REST API jest zwiększany dla pierwszej z tych zmian. Wszystkie kolejne zmiany wprowadzone w wersji Continuous Delivery (CD), które mogą wymagać zmian w klientach, które korzystają z REST API, używają nowego numeru wersji.

Ten numer wersji pozostaje taki sam w kolejnych wersjach produktu CD do czasu wydania kolejnej wersji produktu LTS. Oznacza to, że numer wersji zwiększa się co najwyżej raz między kolejnymi wersjami produktu LTS.

Gdy numer wersji zostanie zwiększony, istniejąca funkcja REST API jest ustabilizowana pod starym numerem wersji. Oznacza to, że istniejąca funkcja REST API, która była dostępna w wersji LTS, pozostaje dostępna pod starym numerem wersji, ale nie wprowadzono żadnych dalszych zmian w tej wersji. Każda nowa funkcja, która jest dodawana do REST API, jest dodawana do nowej wersji produktu REST API. Jednak wszelkie dodatki wprowadzone do REST API w wersji CD przed zwiększeniem wersji nie są gwarantowane do uwzględnienia w starszej wersji produktu REST API.

Istniejące klienci mogą nadal używać produktu REST API pod starym numerem wersji bez konieczności wprowadzania żadnych zmian. Starsze wersje produktu REST API mogą być nieaktualne, a następnie usunięte.

Niektóre zmiany nie wymagają zmian w klientach, które korzystają z REST API. Zmiany te nie powodują zwiększenia liczby wersji. Dlatego należy upewnić się, że żaden klient korzystający z produktu REST API nie musi być aktualizowany po wprowadzeniu tych typów zmian. Zmiany wprowadzone w produkcie REST API mogą obejmować następujące zmiany:

- Dodanie nowego atrybutu JSON do istniejących danych, które są zwracane z REST API.
- Dodanie nowego adresu URL.
- Dodanie nowego komendy HTTP do istniejącego adresu URL.
- Dodanie nowego kodu statusu do istniejącego adresu URL.
- Dodanie nowych opcjonalnych atrybutów JSON do danych, które są wysyłane do istniejącego adresu URL.
- Dodanie nowych parametrów zapytania dla istniejącego adresu URL.
- Dodanie nowych nagłówków do danych, które są wysyłane do istniejącego adresu URL.

- Powrót nowych nagłówków z REST API.

Zmiany w nowej funkcji API REST produktu Continuous Delivery

W przypadku nowej funkcji REST API, która jest dodawana w wersji CD, wszystkie zmiany wprowadzone w tej nowej funkcji, które mogą wymagać wprowadzenia zmian w klientach REST API, nie będą zwiększać numeru wersji. Oznacza to, że nowa funkcja może zostać zmieniona przed następną wersją produktu LTS bez zwiększania numeru wersji. Jeśli funkcja jest dołączona do wersji LTS, wszystkie kolejne zmiany, które mogą wymagać zmian w klientach REST API, zwiększają numer wersji.

Przykład

1. W wersji LTS wydania X REST API jest w wersji 1.
2. W produkcie CD, wersja X.0.1, dodano obsługę nowego adresu URL. Zmiana ta nie wymaga zmian w klientach, które korzystają z REST API. Oznacza to, że REST API pozostaje w wersji 1.
3. W wersji CD X.0.2 dodano obsługę nowego adresu URL. Ta zmiana nie wymaga zmian w klientach, które korzystają z interfejsu REST API. Oznacza to, że REST API pozostaje w wersji 1.
4. W wersji LTS (Y), REST API jest w wersji 1.
5. W wersji CD, wersja Y.0.1, nazwa istniejącego adresu URL jest zmieniana. Ta zmiana może wymagać wprowadzenia zmian w klientach, które korzystają z REST API. Oznacza to, że nowa wersja produktu REST API jest tworzona w wersji 2. Nazwa adresu URL o zmienionej nazwie jest zawarta w wersji 2 produktu REST API wraz z całą istniejącą funkcją. Każda nowa funkcja, która jest dodawana do REST API, jest dodawana do wersji 2. Wersja 1 pozostaje stabilna na poziomie w wersji LTS Y.
6. W produkcie CD o wersji Y.0.2 nazwa innego istniejącego adresu URL jest zmieniana. Ponieważ wersja jest już zwiększana w produkcie CD, wersja Y, REST API pozostaje w wersji 2. Wersja 1 pozostaje stabilna na poziomie w wersji LTS Y.
7. W wersji LTS wydanie Z, REST API pozostaje w wersji 2. Wersja 1 pozostaje stabilna na poziomie w wersji LTS Y.

Administrowanie przy użyciu konsoli WWW

Korzystając z konsoli WWW, można wykonywać podstawowe czynności administracyjne.

V 9.2.0 W produkcie IBM MQ 9.2.0 dostępna jest nowa konsola WWW, znana pod nazwą New Web Console, patrz [“Szybkie oprowadzanie po nowej konsoli WWW”](#) na stronie 94.

Można, jeśli chcesz, kontynuować korzystanie z Dashboard Web Console na wielu platformach, patrz [“Przełączanie między typami konsoli”](#) na stronie 116.

Uwaga: Podczas korzystania z konsoli WWW nie należy wyłączać serwera komend na żadnym z menedżerów kolejek. Jeśli serwer komend jest wyłączony dla menedżera kolejek, konsola WWW nie reaguje z długimi opóźnieniami w przetwarzaniu komend. Wszystkie komendy, które są wysyłane do menedżera kolejek, dla którego serwer komend jest wyłączony.

Zadania pokrewne

V 9.2.0 [Śledzenie nowej konsoli WWW](#)
[Śledzenie konsoli WWW panelu kontrolnego](#)

V 9.2.0 Pierwsze kroki z konsolą WWW

Szybkie rozpoczęcie pracy z konsolą WWW.

Zanim rozpocznie

Aby wykonać tę czynność, użytkownik musi mieć pewne uprawnienia pozwalające na użycie komendy [**dspmweb**](#):

- ▶ **z/OS** W systemie z/OS użytkownik musi mieć uprawnienia do uruchamiania komendy **dspmqweb** i dostęp z uprawnieniami do zapisu do pliku mqwebuser.xml.
- ▶ **Multi** W przypadku wszystkich innych systemów operacyjnych użytkownik musi być użytkownikiem uprzywilejowanym.

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

Procedura

1. Jeśli serwer mqweb nie jest jeszcze skonfigurowany do użycia przez konsolę WWW, skonfiguruj serwer mqweb.

Więcej informacji na temat tworzenia podstawowej konfiguracji serwera mqweb z podstawowym rejestrem zawiera sekcja [Podstawowa konfiguracja serwera mqweb](#).

2. ▶ **z/OS**

W systemie z/OS ustaw zmienną środowiskową WLP_USER_DIR tak, aby można było używać komendy **dspmqweb**. Należy ustawić zmienną tak, aby wskazywała konfigurację serwera mqweb, wprowadzając następującą komendę:

```
export WLP_USER_DIR=WLP_user_directory
```

, gdzie *WLP_user_directory* jest nazwą katalogu, który jest przekazywany do crtmqweb. Na przykład:

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

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

3. Określ identyfikator URI dla konsoli WWW, wprowadzając następującą komendę:

```
dspmqweb status
```

Komenda generuje dane wyjściowe podobne do poniższych:

```
MQWB1124I: Server 'mqweb' is running.
URLS:
https://localhost:9443/ibmmq/rest/v1/
https://localhost:9443/ibmmq/console/
```

Identyfikator URI dla konsoli WWW kończy się przyrostkiem *console/*.

4. Połącz się z konsolą WWW, wprowadzając adres URL z programu ["3"](#) na stronie [93](#) w przeglądarce.

Przeglądarka może utworzyć wyjątek zabezpieczeń, ponieważ certyfikat domyślny dostarczony z serwerem mqweb nie jest zaufanym certyfikatem. Wybierz tę opcję, aby przejść do konsoli WWW.

5. Zaloguj się do konsoli WWW. Użyj nazwy użytkownika mqadmini hasła mqadmin.

Co dalej

Domyślnie konsola WWW używa uwierzytelniania opartego na tokenach do uwierzytelniania użytkowników. Można również użyć uwierzytelniania za pomocą certyfikatu klienta. Więcej informacji na ten temat zawiera sekcja [Korzystanie z uwierzytelniania przy użyciu certyfikatu klienta przy użyciu konsoli REST API i konsoli WWW](#).

▶ **z/OS** 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.

Pojęcia pokrewne

“Administrowanie przy użyciu konsoli WWW” na stronie 92

Korzystając z konsoli WWW, można wykonywać podstawowe czynności administracyjne.

Zadania pokrewne

“Praca z menedżerami kolejek lokalnych” na stronie 98

Lokalne menedżery kolejek są tworzone, konfigurowane i sterujące z poziomu najwyższego poziomu

widoku Zarządzanie  **Manage**

Multi Ograniczenia w systemie IBM MQ for Multiplatforms

W przypadku korzystania z produktu IBM MQ Console do zarządzania menedżerami kolejek w systemie IBM MQ for Multiplatforms obowiązują następujące ograniczenia.

- Produktu IBM MQ Console nie można używać do pracy z kanałami AMQP.
- Produktu IBM MQ Console nie można używać do pracy z kanałami MQTT.

Pojęcia pokrewne

“Administrowanie przy użyciu konsoli WWW” na stronie 92

Korzystając z konsoli WWW, można wykonywać podstawowe czynności administracyjne.

Zadania pokrewne

“Praca z menedżerami kolejek lokalnych” na stronie 98

Lokalne menedżery kolejek są tworzone, konfigurowane i sterujące z poziomu najwyższego poziomu

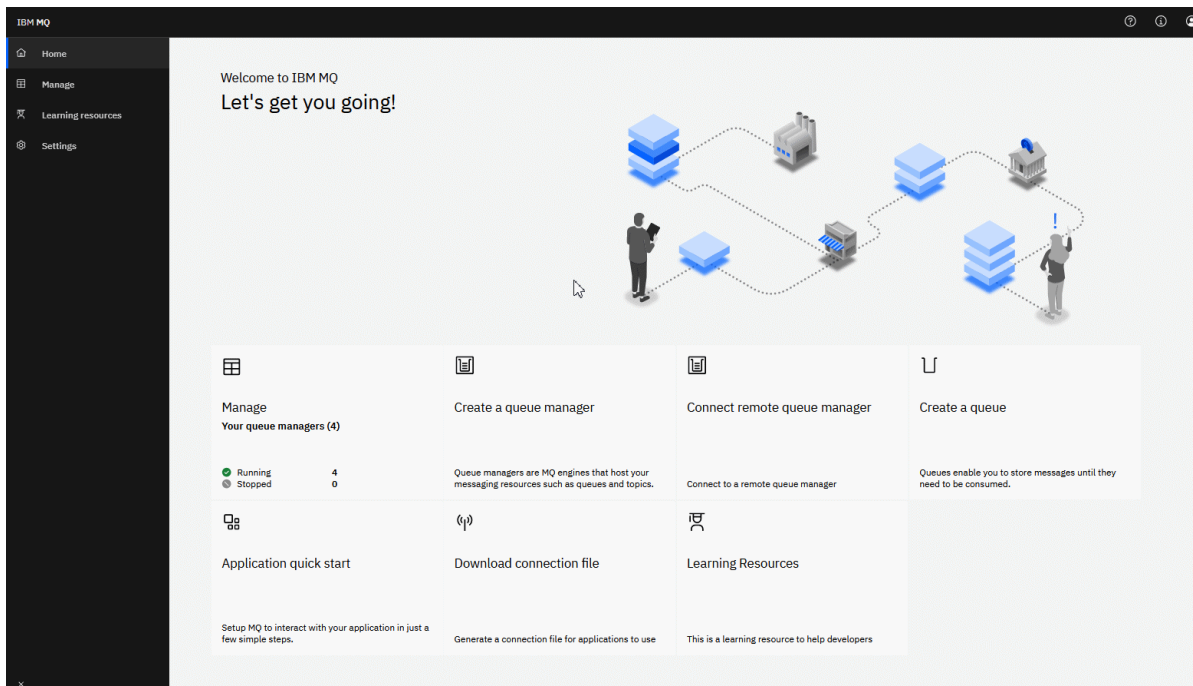
widoku Zarządzanie  **Manage**

V 9.2.0 Szybkie oprowadzanie po nowej konsoli WWW

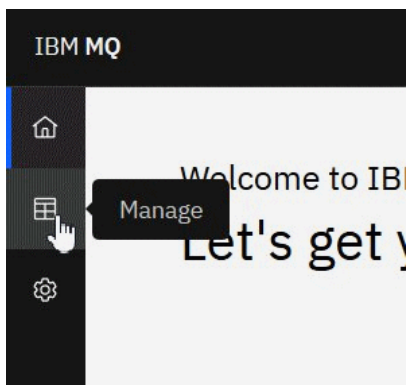
Szybka prezentacja umożliwia rozpoczęcie pracy z New Web Console. Poniżej znajdują się bardziej szczegółowe tematy dotyczące jego użycia.

New Web Console jest preferowanym interfejsem WWW, ale jeśli ma zostać przeprowadzony przy użyciu istniejącej konsoli (Dashboard Web Console), można przełączyć się z powrotem na tę konsolę, patrz sekcja “Przełączanie między typami konsoli” na stronie 116.

Po pierwszym zalogowaniu się do New Web Console zostanie wyświetlona strona docelowa. W tym miejscu można wybrać zarządzanie istniejącymi menedżerami kolejek, tworzenie menedżera kolejek lub kolejki, nawigowanie do niektórych tematów związanych z edukacją lub otwieranie informacji o produkcie IBM MQ w produkcie IBM Documentation. Można również uruchomić szybki start aplikacji, który poprowadzi użytkownika przez proces szybkiego i łatwego konfigurowania przesyłania komunikatów między nowymi lub istniejącymi menedżerami kolejek i aplikacjami.



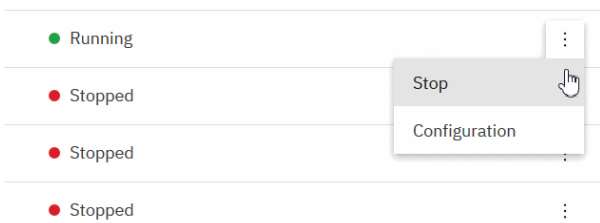
Alternatywnie można po prostu kliknąć ikonę Zarządzaj, aby rozpocząć zarządzanie obiektami IBM MQ .



W widoku zarządzania początkowo wyświetlane są menedżery kolejek i ich bieżący stan. Można również tworzyć nowe menedżery kolejek oraz łączyć zdalne menedżery kolejek.

Queue manager name ↑	Version	Status
MpgQM2	9.1.5.0	Running
MyPortQmgr	9.1.5.0	Stopped
QM1	9.1.5.0	Stopped
qmanotherone	9.1.5.0	Stopped
qmq	9.1.5.0	Stopped
qnd	9.1.5.0	Stopped
qne	9.1.5.0	Stopped
RPilotFinalTest	9.1.5.0	Running
sdwe	9.1.5.0	Stopped
Steves_qm	9.1.5.0	Stopped

Każdy menedżer kolejek ma menu umożliwiające zatrzymanie lub skonfigurowanie działającego menedżera kolejek albo uruchomienie lub usunięcie zatrzymanego menedżera kolejek.



Rekordy uprawnień, obiekty informacji uwierzytelniających oraz rekordy uwierzytelniania kanału dla menedżera kolejek można znaleźć na karcie **Zabezpieczenia** na stronie **Konfiguracja** menedżera kolejek, w której można tworzyć i dodawać nowe.


Kliknij nazwę działającego menedżera kolejek, aby otworzyć jego panel kontrolny.

The screenshot shows the IBM MQ Queue Manager interface for 'MpgQM'. It features a top navigation bar with 'Queues', 'Topics', 'Subscriptions', and 'Communication' tabs. Below the tabs are three summary cards: '1 Full capacity' (1 user-defined, 0 system), '2 Greater than 50% capacity' (1 user-defined, 1 system), and '61 Less than 1% capacity' (15 user-defined, 46 system). A table below lists queues with columns for Name, Type, Depth %, and Maximum depth. A 'Create +' button is visible in the top right of the table area.


Name ↑	Type	Depth %	Maximum depth
AMQ.MQEXPLORER.5E68C78B20805304	Local	0%	0/5000
Queueue1	Local	0%	0/1
Queueue2	Local	0%	0/5000
Queueue3	Local	0%	0/5000
Queueue4	Local	0%	0/5000

Z poziomu panelu kontrolnego menedżera kolejek można wykonać następujące działania:


Na karcie **Kolejki** :

- Tworzenie nowych kolejek
- Skonfiguruj istniejące kolejki 
- Kliknij nazwę kolejki, aby wyświetlić istniejące komunikaty i utworzyć nowe.



Na karcie **Tematy** :


- Tworzenie nowych tematów
- Skonfiguruj istniejące tematy 
- Kliknij nazwę tematu, aby wyświetlić zgodne subskrypcje

Na karcie **Subskrypcje** :

- Utwórz nowe subskrypcje zarządzane lub niezarządzane
- Skonfiguruj istniejące subskrypcje 

Na karcie **Komunikacja** :

- Obiekty nasłuchiwania:
 - Uruchamianie, zatrzymywanie i konfigurowanie programów nasłuchujących 
 - Tworzenie nowych obiektów nasłuchiwania
- Kanały menedżera kolejek:
 - Uruchamianie, zatrzymywanie, ping i konfigurowanie kanałów 
 - Tworzenie nowych kanałów
 - Resetuj kanały (z pozycji menu **Zaawansowane**)
 - Rozstrzyganie wątpliwych komunikatów w kanałach (z pozycji menu **Zaawansowane**)


- Kanały aplikacji:
 - Uruchamianie, zatrzymywanie, ping i konfigurowanie kanałów 
 - Tworzenie nowych kanałów
 - Resetuj kanały (z pozycji menu **Zaawansowane**)
 - Rozstrzygnięcie wątpliwych komunikatów w kanałach (z pozycji menu **Zaawansowane**)


Praca z menedżerami kolejek lokalnych

Lokalne menedżery kolejek są tworzone, konfigurowane i sterujące z poziomu najwyższego poziomu

widoku Zarządzanie  **Manage**

O tym zadaniu

 **Multi** W widoku Zarządzanie wyświetlane są lokalne menedżery kolejek, 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/OS** W systemie z/OS w widoku Zarządzanie wyświetlane są menedżery kolejek, które znajdują się w tej samej wersji, co IBM MQ Console, i są zdefiniowane w systemie, w którym działa produkt MQ Console. Menedżery kolejek w innej wersji niż MQ Console nie są wyświetlane na liście.

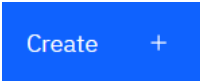


Z listy można wybrać pojedyncze menedżery kolejek, z którymi ma pracować praca.




Uwaga: Produkt IBM MQ Console nie obsługuje replikowanych menedżerów kolejek danych (RDQMs).

Procedura

- Aby utworzyć nowy menedżer kolejek lokalnych:

 **Create** +

- Kliknij przycisk tworzenia  w widoku listy menedżera kolejek.
 - Wprowadź nazwę nowego menedżera kolejek. Nazwa może zawierać maksymalnie 48 znaków. Poprawnymi znakami są litery i cyfry oraz znaki ":", "/", "_i "%".
 - Opcjonalne: Wprowadź dostępny port TCP/IP dla menedżera kolejek, na którym ma być nasłuchiwać. Numer portu nie może być większy niż 65535.
 - Kliknij opcję **Create**. Nowy menedżer kolejek zostanie utworzony i uruchomiony.
- Aby uruchomić lokalny menedżer kolejek:
 - Znajdź menedżer kolejek, który ma zostać uruchomiony na liście.
 - Wybierz opcję **Start** z menu .
 - Aby zatrzymać lokalny menedżer kolejek:
 - Wybierz z listy menedżer kolejek, który ma zostać zatrzymany, w widżecie menedżera kolejek lokalnych.
 - Wybierz opcję **Zatrzymaj** z menu .
 - Aby usunąć lokalny menedżer kolejek:
 - Jeśli menedżer kolejek jest uruchomiony, zatrzymaj go.

- b) Wybierz opcję **Konfiguruj** z menu , a następnie wybierz opcję **Usuń menedżer kolejek**.
- c) Potwierdź, że chcesz usunąć menedżera kolejek, wprowadzając jego nazwę w oknie potwierdzenia. Menedżer kolejek i wszystkie powiązane obiekty są usuwane.
- Aby wyświetlić i edytować właściwości lokalnego menedżera kolejek:
 - a) Upewnij się, że menedżer kolejek jest uruchomiony, i znajdź go na liście menedżerów kolejek.
 - b) Wybierz opcję **Konfiguruj** z menu .
 - c) Upewnij się, że zaznaczona jest karta **Właściwości**. 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.
- Aby pracować z ustawieniami zabezpieczeń dla lokalnego menedżera kolejek:
 - a) Upewnij się, że menedżer kolejek jest uruchomiony, a następnie wybierz go na liście menedżerów kolejek.
 - b) Wybierz opcję **Konfiguruj** z menu .
 - c) Upewnij się, że wybrana jest karta **Zabezpieczenia**.
 - d) Istnieje możliwość pracy z obiektami uwierzytelniania, rekordami autoryzacji lub obiektami uwierzytelniania kanału. Aby uzyskać więcej informacji, odwiedź następujące tematy:
 - “Praca z obiektami informacji uwierzytelniających” na stronie 99
 - “Praca z rekordami uprawnień menedżera kolejek” na stronie 101
 - “Praca z rekordami uwierzytelniania kanału” na stronie 102

V9.2.0 Praca z obiektami informacji uwierzytelniających


Za pomocą konsoli można dodawać i usuwać obiekty informacji uwierzytelniających w menedżerze kolejek. Możliwe jest również wyświetlanie i ustawianie właściwości oraz zarządzanie rekordami uprawnień dla obiektów.

O tym zadaniu

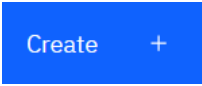





W widoku Informacje o uwierzytelnianiu wyświetlane są informacje o uwierzytelnianiu, które istnieją dla konkretnego menedżera kolejek. Z listy można wybrać pojedyncze informacje uwierzytelniania, z którymi ma pracować praca.


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.

Procedura

- Aby wyświetlić informacje uwierzytelniające dla menedżera kolejek:
 - a) Upewnij się, że menedżer kolejek jest uruchomiony, a następnie wybierz go na liście menedżerów kolejek.
 - b) Wybierz opcję **Konfiguruj** z menu .
 - c) Upewnij się, że wybrana jest karta **Zabezpieczenia**.
 - d) Wybierz opcję **Informacje uwierzytelniające** z panelu nawigacyjnego.
- Aby dodać obiekt informacji uwierzytelniającej:



- a) Kliknij przycisk tworzenia  w widoku listy informacji uwierzytelniających.
 - b) Określ nazwę obiektu informacji uwierzytelniającej. Poprawnymi znakami są litery i cyfry oraz znaki ".", "/", "_" i "%".
 - 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. Opcjonalnie można określić nazwę użytkownika i hasło dla użytkownika uzyskanego w celu uzyskania dostępu do serwera LDAP.
 - 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, wówczas można pominąć numer portu. Adresy URL HTTP są zdefiniowane w dokumencie RFC 1738.
 - W przypadku produktu **IDPW OS** nie ma żadnych dodatkowych wymagań, chociaż można opcjonalnie określić dalsze opcje dla tego typu uwierzytelniania.
 - 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. Opcjonalnie można określić dalsze opcje dla tego typu uwierzytelniania.
 - e) Kliknij przycisk **Add** (Dodaj).
- Aby usunąć obiekt informacji uwierzytelniającej:
 - a) Wybierz ikonę spanner  dla obiektu informacji uwierzytelniającej, który ma zostać usunięty z listy.
 - b) W widoku właściwości obiektu kliknij opcję **Usuń obiekt informacji uwierzytelniających**.
 - c) Potwierdź, że chcesz usunąć obiekt informacji uwierzytelniającej, klikając przycisk **Usuń**. Obiekt został usunięty.
 - Aby wyświetlić i edytować właściwości obiektu informacji uwierzytelniającej:
 - a) Wybierz ikonę spanner  dla obiektu informacji uwierzytelniającej, który ma zostać wyświetlony z listy.
 - b) Aby edytować wyświetlone właściwości, kliknij przycisk Edytuj .
 - c) Edytuj wybrane właściwości. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend.
 - d) Kliknij przycisk **Zapisz**, aby zapisać zmiany.
 - Aby wyświetlić i edytować rekordy uprawnień dla obiektu informacji uwierzytelniającej:
 - a) Wybierz ikonę spanner  dla obiektu informacji uwierzytelniającej, dla którego ma zostać wyświetlony rekord uprawnień z listy.
 - b) Wybierz kartę **Zabezpieczenia**.
 - c) Aby edytować lub usunąć istniejący rekord uprawnień, wybierz z menu opcję **Edytuj** lub **Usuń** .

d) Aby dodać nowy rekord uprawnień, kliknij przycisk **Dodaj** , podaj szczegóły nowego rekordu uprawnień i kliknij przycisk **Utwórz**.





Praca z rekordami uprawnień menedżera kolejek


Użytkownik może sterować dostępem użytkowników i grup do menedżerów kolejek, określając rekord uprawnień dla tego użytkownika lub grupy.

O tym zadaniu

Użytkownik może precyzyjnie dostroić dostęp, jaki użytkownik lub grupa użytkowników przesyłania komunikatów ma do konkretnego menedżera kolejek przy użyciu rekordów uprawnień. Istnieją dwa typy rekordów uprawnień: rekordy uprawnień, które sterują uprawnieniami ogólnymi i tworzą rekordy uprawnień kontrolujący, którzy użytkownicy i grupy mogą tworzyć obiekty dla menedżera kolejek.

Procedura

- Aby wyświetlić rekordy uprawnień dla menedżera kolejek:
 - a) Upewnij się, że menedżer kolejek jest uruchomiony, a następnie wybierz go na liście menedżerów kolejek.
 - b) Wybierz opcję **Konfiguruj** z menu .
 - c) Upewnij się, że wybrana jest karta **Zabezpieczenia**.
 - d) Wybierz opcję **Rekordy uprawnień** z panelu nawigacyjnego. W widoku wyświetlane są rekordy uprawnień w dwóch panelach, co umożliwia pracę z rekordami uprawnień ogólnych oraz z rekordami uprawnień do tworzenia.
- Aby dodać rekord uprawnień ogólnych:
 - a) Kliknij przycisk dodawania  w widoku listy rekordów uprawnień.
 - b) Wybierz, czy dodajesz rekord uprawnień dla użytkownika, czy grupy.
 - c) Podaj nazwę użytkownika lub grupy, dla której dodawany jest rekord uprawnień (rekord uprawnień przyjmuje to jako nazwę)
 - d) Wybierz uprawnienia, które chcesz nadać (patrz -- więcej informacji na temat uprawnień)
 - e) Kliknij opcję **Create**.
- Aby dodać rekord tworzenia uprawnień:
 - a) Kliknij przycisk dodawania  w widoku listy rekordów uprawnień do tworzenia.
 - b) Wybierz, czy dodajesz rekord uprawnień dla użytkownika, czy grupy.
 - c) Podaj nazwę użytkownika lub grupy, dla której dodawany jest rekord uprawnień (rekord uprawnień przyjmuje to jako nazwę)
 - d) Wybierz typy obiektów, które mają zostać utworzone przez użytkownika.
 - e) Kliknij opcję **Create**.
- Aby usunąć rekord uprawnień:
 - a) Otwórz menu  dla rekordu uprawnień, który ma zostać usunięty, a następnie wybierz opcję **Usuń**.
 - b) Potwierdź, że chcesz usunąć obiekt informacji uwierzytelniającej, klikając przycisk **Usuń**. Obiekt został usunięty.
- Aby wyświetlić i edytować właściwości rekordu uprawnień:

- a) Otwórz menu  dla rekordu uprawnień, który ma zostać usunięty, a następnie wybierz opcję **Edytuj**.
- b) Zmień ustawienia zgodnie z wymaganiami, a następnie kliknij przycisk **Zapisz**, aby zapisać zmiany.

Praca z rekordami uwierzytelniania kanału


Za pomocą programu IBM MQ Console można dodawać i usuwać rekordy uwierzytelniania kanału w menedżerze kolejek. Istnieje również możliwość wyświetlenia i ustawienia właściwości dla rekordów uwierzytelniania kanału.

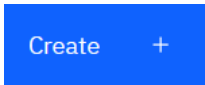
O tym zadaniu

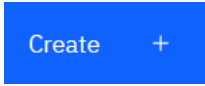
Aby wykonać bardziej precyzyjną kontrolę nad dostępem, który jest nadawany do łączenia systemów na poziomie kanału, można użyć rekordów uwierzytelniania kanału.


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

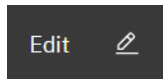
- Aby wyświetlić informacje o uwierzytelnianiu kanału dla menedżera kolejek:
 - a) Upewnij się, że menedżer kolejek jest uruchomiony, a następnie wybierz go na liście menedżerów kolejek.
 - b) Wybierz opcję **Konfiguruj** z menu .
 - c) Upewnij się, że wybrana jest karta **Zabezpieczenia**.
 - d) Na panelu nawigacyjnym wybierz opcję **Uwierzytelnianie kanału**.
- Aby dodać rekord uwierzytelniania kanału:

- a) Kliknij przycisk tworzenia  w widoku listy informacji o uwierzytelnianiu kanału.
- b) Wybierz typ reguły, który ma być używany. Wybierz jedną z opcji: **Zezwalaj**, **Blokuj** lub **Warn**.
- c) Wybierz typ tożsamości, dla której konfigurowana jest reguła uwierzytelniania kanału. Dostępne są różne typy tożsamości, w zależności od wybranego typu reguły.
- d) Podaj wymagane informacje dla tożsamości, która jest określana. Domyślnie wyświetlane są minimalne zalecane właściwości, dla których można podać wartości. Wszystkie dostępne właściwości można wyświetlić, wybierając opcję **Pokaż wszystkie dostępne opcje**.

- e) Kliknij przycisk tworzenia , aby utworzyć rekord uwierzytelniania kanału. Więcej informacji na temat dostępnych ustawień dla rekordów uwierzytelniania kanału znajduje się w sekcji [Rekordy uwierzytelniania kanału i SET CHLAUTH](#).

- Aby usunąć rekord uwierzytelniania kanału:
 - a) Kliknij ikonę spanner  znajdującą się obok rekordu uwierzytelniania kanału, który ma zostać usunięty.
 - b) W widoku Edycja uwierzytelniania kanału kliknij opcję **Usuń obiekt uwierzytelniania kanału**.
 - 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) Kliknij ikonę spanner  znajdującą się obok rekordu uwierzytelniania kanału, który ma być edytowany lub wyświetlany. Zostaną wyświetlone właściwości.



- b) Kliknij przycisk Edytuj
- c) Edytuj wybrane właściwości. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend.
- d) Kliknij przycisk **Zapisz**, aby zapisać zmiany.

V 9.2.3 Dodawanie zdalnego menedżera kolejek

Za pomocą konsoli IBM MQ Console można administrować menedżerem kolejek działającym w systemie zdalnym.

Zanim rozpoczniesz

Należy przygotować menedżer kolejek w systemie zdalnym, aby mógł być administrowany zdalnie, patrz krok "1" na stronie 105 produktu "[Łączenie zdalnego menedżera kolejek z serwerem MQ Console za pomocą wiersza komend](#)" na stronie 104.

Należy również skonfigurować plik konfiguracyjny, który steruje sposobem korzystania z połączeń zdalnych z serwera IBM MQ Console. Plik konfiguracyjny należy utworzyć za pomocą komendy **setmqweb** z parametrem **remote** (patrz sekcja [Konfigurowanie zachowania połączenia zdalnego menedżera kolejek i setmqweb](#)). Pliku konfiguracyjnego nie można edytować bezpośrednio.

O tym zadaniu

Aby określić szczegóły połączenia zdalnego, należy użyć tabeli definicji połączenia klienta (CCDT) w formacie JSON. Tabelę CCDT JSON można utworzyć za pomocą edytora tekstu (patrz krok "2" na stronie 106 produktu "[Łączenie zdalnego menedżera kolejek z serwerem MQ Console za pomocą wiersza komend](#)" na stronie 104) lub można go utworzyć za pomocą programu IBM MQ Console.

Można również utworzyć tabelę definicji kanału klienta z poziomu produktu IBM MQ Console, określając szczegóły połączenia bezpośrednio podczas dodawania zdalnego menedżera kolejek.

Istnieje również możliwość połączenia zdalnego menedżera kolejek z serwerem IBM MQ Console za pomocą wiersza komend dla wszystkich wymaganych zadań (oprócz przygotowania zdalnego menedżera kolejek i utworzenia tabeli definicji kanału klienta). Patrz "[Łączenie zdalnego menedżera kolejek z serwerem MQ Console za pomocą wiersza komend](#)" na stronie 104.

Procedura

- Aby dodać zdalny menedżer kolejek, należy określić istniejącą tabelę CCDT:
 - a) Na stronie głównej kliknij opcję **Połącz zdalny menedżer kolejek**.
 - b) Określ nazwę menedżera kolejek zdalnych.
 - c) Opcjonalnie podaj unikalną nazwę dla menedżera kolejek. Jeśli nie zostanie podana unikalna nazwa, rzeczywista nazwa będzie używana z przedrostkiem "remote-".
 - d) Upewnij się, że wybrana jest opcja **Connect using a JSON CCDT**.
 - e) Kliknij przycisk **Przeglądaj** i wybierz plik zawierający tabelę CCDT JSON, która ma być używana.
 - f) Kliknij przycisk **Dalej**, aby przejść do strony użytkownika i opcjonalnie podaj nazwę użytkownika i hasło, aby połączyć się ze zdalnym menedżerem kolejek. Jeśli te informacje nie zostaną określone, informacje uwierzytelniania będą pobierane z pliku konfiguracyjnego połączenia zdalnego.
 - g) Kliknij przycisk **Dalej**, aby przejść do strony Certyfikat. Jeśli w tabeli CCDT określone są informacje "transmissionSecurity", to jest to informacja używana. Opcjonalnie można wkleić certyfikat (jako klucz publiczny zakodowany przy użyciu base64), a ten certyfikat zostanie dodany do globalnego magazynu zaufanych certyfikatów.

Certyfikat jest tymczasowo przechowywany w produkcie `WLP_USER_DIR/generated.crts/uniqueName-qmgrName.crt`, zanim zostanie dodany do magazynu zaufanych certyfikatów. Po pomyślnym dodaniu połączenia certyfikat zostanie usunięty z tej lokalizacji.

- h) Kliknij przycisk **Dalej**, aby wyświetlić stronę podsumowania. Za pomocą przycisku **Wstecz** można ponownie odwiedzać poprzednie strony i wprowadzać poprawki. Jeśli informacje są szczęśliwe, kliknij opcję **Połącz**, aby połączyć się ze zdalnym menedżerem kolejek.
- Aby dodać zdalny menedżer kolejek i podać informacje o połączeniu ręcznie:
 - a) Na stronie głównej kliknij opcję **Połącz zdalny menedżer kolejek**.
 - b) Określ nazwę menedżera kolejek zdalnych.
 - c) Opcjonalnie podaj unikalną nazwę dla menedżera kolejek. Jeśli nie zostanie podana unikalna nazwa, rzeczywista nazwa będzie używana z przedrostkiem "remote-".
 - d) Wybierz opcję **Ręczne wprowadzanie**.
 - e) Wprowadź nazwę kanału połączenia klienckiego, który będzie używany przez połączenie.
 - f) Podaj nazwę hosta, na którym jest uruchomiony zdalny menedżer kolejek. Jeśli zostaną wykryte zdalne instalacje produktu MQ, zostaną wyświetlone nazwy hostów i można wybrać host zdalnego menedżera kolejek, z którym ma zostać nawiązane połączenie. W niektórych konfiguracjach sieci nie jest możliwe wykrywanie zdalnych instancji MQ. W takim przypadku należy ręcznie dodać nazwę hosta i port.
 - g) Kliknij przycisk **Dalej**, aby przejść do strony użytkownika i opcjonalnie podaj nazwę użytkownika i hasło, aby połączyć się ze zdalnym menedżerem kolejek. Jeśli te informacje nie zostaną określone, informacje uwierzytelniania będą pobierane z pliku konfiguracyjnego połączenia zdalnego.
 - h) Kliknij przycisk **Dalej**, aby przejść do strony Certyfikat. Z listy rozwijanej można wybrać protokół SSL CipherSpec. Opcjonalnie można wkleić certyfikat (jako klucz publiczny zakodowany przy użyciu base64), a ten certyfikat zostanie dodany do globalnego magazynu zaufanych certyfikatów.

Certyfikat jest tymczasowo przechowywany w produkcie `WLP_USER_DIR/generated.crts/uniqueName-qmgrName.crt`, zanim zostanie dodany do magazynu zaufanych certyfikatów. Po pomyślnym dodaniu połączenia certyfikat zostanie usunięty z tej lokalizacji.
 - i) Kliknij przycisk **Dalej**, aby wyświetlić stronę podsumowania. Za pomocą przycisku **Wstecz** można ponownie odwiedzać poprzednie strony i wprowadzać poprawki. Jeśli informacje są szczęśliwe, kliknij opcję **Połącz**, aby połączyć się ze zdalnym menedżerem kolejek.

Podane informacje o połączeniu są zapisywane w pliku CCDT w katalogu WWW użytkownika. Ścieżka to `WLP_USER_DIR/generated.ccdt/ccdt-uniqueName`.

Wyniki

Zdalny menedżer kolejek jest wyświetlany na liście menedżerów kolejek zdalnych w IBM MQ Console. Jeśli nawiązanie połączenia powiedzie się, można administrować obiektami zdalnego menedżera kolejek w taki sam sposób, jak praca z obiektami lokalnego menedżera kolejek.

V9.2.3 Łączenie zdalnego menedżera kolejek z serwerem MQ Console za pomocą wiersza komend

Zdalne menedżery kolejek można połączyć z MQ Console za pomocą komendy `setmqweb remote` w wierszu komend. Zdalny menedżer kolejek może być menedżerem kolejek, który jest uruchomiony w innej instalacji w tym samym systemie, co MQ Console, lub menedżer kolejek działający w innym systemie.

Zanim rozpoczniesz

- Upewnij się, że serwer mqweb jest skonfigurowany w taki sposób, aby zezwalał na połączenia zdalnego menedżera kolejek z serwerem MQ Console. Więcej informacji na ten temat zawiera sekcja [Konfigurowanie zachowania połączeń zdalnego menedżera kolejek](#).

Procedura

1. Skonfiguruj lokalny menedżer kolejek QML, aby akceptować połączenia zdalne:

- a) Utwórz kanał połączenia z serwerem, aby umożliwić zdalne administrowanie menedżerem kolejek.

Za pomocą konsoli MQ Console można utworzyć kanał połączenia z serwerem lub za pomocą komendy MQSC **DEFINE CHANNEL** w wierszu komend.

Na przykład, aby utworzyć kanał połączenia z serwerem QM1 . SVRCONN dla menedżera kolejek zdalnych QM1, należy wprowadzić następujące komendy:

```
runmqsc QM1
DEFINE CHANNEL(QM1.SVRCONN) CHLTYPE(SVRCONN) TRPTYPE(TCP)
```

Więcej informacji na temat opcji **DEFINE CHANNEL** i dostępnych opcji zawiera sekcja [DEFINE CHANNEL\(DEFINIOWANIE KANAŁU\)](#).

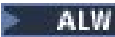
- b) Upewnij się, że kanał połączenia z serwerem zezwala na dostęp do kanału przez odpowiedni identyfikator użytkownika. Ten identyfikator użytkownika musi być identyfikatorem użytkownika, który jest uruchamiany przez serwer mqweb w systemie, w którym MQ Console nawiązuje połączenie zdalne.

Za pomocą konsoli MQ Console można utworzyć odpowiedni rekord uprawnień lub za pomocą komendy MQSC **SET CHLAUTH** w wierszu komend.

Na przykład, aby autoryzować użytkownika, `exampleUser`, aby uzyskać dostęp do QM1 . SVRCONN dla zdalnego menedżera kolejek QM1, należy wprowadzić następującą komendę:

```
SET CHLAUTH(QM1.SVRCONN) TYPE(ADDRESSMAP) ADDRESS('*') MCAUSER('exampleUser')
```

W tym przykładzie parametr **address** jest ustawiony w taki sposób, że produkt `exampleUser` może łączyć się z dowolnym adresem IP. Zamiast tego można ustawić parametr **address** na określony adres IP. Na przykład można ograniczyć dostęp tylko do adresu IP, z którego zdalnie MQ Console łączy się z menedżerem kolejek. Więcej informacji na temat opcji dostępnych dla tej komendy można znaleźć w sekcji [SET CHLAUTH](#).

- c) 

Utwórz program nasłuchujący, aby akceptować przychodzące połączenia sieciowe:


Za pomocą MQ Console można utworzyć program nasłuchujący lub za pomocą komendy MQSC **DEFINE LISTENER** w wierszu komend.

Na przykład, aby utworzyć program nasłuchujący REMOTE . LISTENER na porcie 1414 dla zdalnego menedżera kolejek QM1, należy wprowadzić następującą komendę:


```
DEFINE LISTENER(REMOTE.LISTENER) TRPTYPE(TCP) PORT(1414)
```

- d) Upewnij się, że nasłuchiwanie jest uruchomione:

Za pomocą konsoli MQ Console można uruchomić program nasłuchujący. Można też użyć komendy MQSC **START LISTENER** w wierszu komend.

 Na przykład w systemie AIX, Linux, and Windows , aby uruchomić program nasłuchujący REMOTE . LISTENER dla menedżera kolejek QM1, należy wprowadzić następującą komendę:

```
START LISTENER(REMOTE.LISTENER)
```

 Na przykład w systemie z/OS, aby uruchomić program nasłuchujący, wprowadź następującą komendę:

```
START LISTENER TRPTYPE(TCP) PORT(1414)
```

Należy pamiętać, że przestrzeń adresowa inicjatora kanału musi zostać uruchomiona przed uruchomieniem programu nasłuchującego w systemie z/OS.

2. Utwórz plik CCDT JSON zawierający informacje o połączeniu zdalnego menedżera kolejek:

- Wygeneruj plik CCDT z definicji lokalnego menedżera kolejek za pomocą MQ Console powiązanego z tą samą instalacją, co menedżer kolejek, z którym ma zostać nawiązane połączenie zdalne.

Na panelu **Home** (Strona główna) kliknij kafel **Download connection file** (Pobierz plik połączenia).

- Utwórz plik CCDT w formacie JSON, który definiuje połączenie. Więcej informacji na temat tworzenia tabeli definicji kanału klienta w formacie JSON zawiera sekcja [Konfigurowanie formatu CCDT w formacie JSON](#).

Plik CCDT musi zawierać informacje o `name`, `clientConnection` i `type`. Oznacza to, że plik CCDT musi zawierać dane połączenia dla zdalnego menedżera kolejek (QM1) i musi być zapisany na hoście lokalnego menedżera kolejek (QML), na tym samym hoście, na którym znajduje się MQ Console. W podsumowaniu jest to dla lokalnego menedżera kolejek QML łączenie się ze zdalnym menedżerem kolejek QM1.

Opcjonalnie można dołączyć dodatkowe informacje, takie jak informacje o produkcie `transmissionSecurity`. Więcej informacji na temat wszystkich definicji atrybutów kanału CCDT znajduje się w sekcji [Pełna lista definicji atrybutów kanału CCDT](#).

W poniższym przykładzie przedstawiono podstawowy plik CCDT JSON dla połączenia zdalnego menedżera kolejek. Ustawia nazwę kanału na tę samą nazwę, co przykładowy kanał połączenia z serwerem, utworzony w kroku "1" na stronie 105, oraz port połączenia z taką samą wartością, jak port używany przez program nasłuchujący. Host połączenia jest ustawiony na nazwę hosta systemu, na którym uruchomiony jest przykładowy menedżer kolejek zdalnych (QM1):

```
{
  "channel": [
    {
      "name": "QM1.SVRCONN",
      "clientConnection": {
        "connection": [
          {
            "host": "example.com",
            "port": 1414
          }
        ],
        "queueManager": "QM1"
      },
      "transmissionSecurity": {
        "cipherSpecification": "",
        "certificateLabel": "",
        "certificatePeerName": ""
      },
      "type": "clientConnection"
    }
  ]
}
```

- ## 3. Komenda **setmqweb remote** służy do dodawania informacji o zdalnym menedżerze kolejek do konfiguracji produktu MQ Console. Należy użyć komendy **setmqweb**, która jest powiązana z instalacją, z której mają być wyświetlane informacje o zdalnym menedżerze kolejek.

Aby dodać zdalny menedżer kolejek do serwera MQ Console, należy podać nazwę menedżera kolejek, unikalną nazwę menedżera kolejek (aby odróżnić inne zdalne menedżery kolejek, które mogą mieć tę samą nazwę menedżera kolejek), a także adres URL CCDT dla menedżera kolejek. Istnieje kilka dodatkowych opcji, które można określić, takich jak nazwa użytkownika i hasło używane w połączeniu zdalnego menedżera kolejek lub szczegóły magazynu zaufanych certyfikatów i magazynu kluczy. Pełną listę parametrów, niż można podać przy użyciu komendy **setmqweb remote**, można znaleźć w sekcji [setmqweb](#).

Na przykład, aby dodać przykładowy zdalny menedżer kolejek QM1, korzystając z przykładowego pliku CCDT i określając, że nazwą użytkownika, która ma być używana dla połączenia, jest `exampleUser`, należy wprowadzić następującą komendę:

```
setmqweb remote add -uniqueName "MACHINEAQM1" -qmgrName "QM1" -ccdtURL  
"c:\myccdts\ccdt.json" -username "exampleUser" -password "password"
```

Wyniki

Zdalny menedżer kolejek jest wyświetlany na liście menedżerów kolejek zdalnych w IBM MQ Console. Jeśli nawiązanie połączenia powiedzie się, można administrować obiektami zdalnego menedżera kolejek w taki sam sposób, jak praca z obiektami lokalnego menedżera kolejek.

V 9.2.0 Praca z obiektami IBM MQ

Każdy menedżer kolejek produktu IBM MQ ma kilka różnych typów obiektów powiązanych z tym menedżerem kolejek.

O tym zadaniu

Za pomocą konsoli można pracować z następującymi typami obiektów IBM MQ :

- Kolejki
- Tematy
- Subskrypcje
- Obiekty komunikacyjne:
 - Procesy nasłuchujące
 - Kanały menedżera kolejek
 - Kanały aplikacji

Procedura

Aby pracować z obiektem IBM MQ :

1. W widoku listy menedżerów kolejek kliknij menedżer kolejek, do którego należą obiekty, z którymi chcesz pracować.
2. Kliknij kartę Kolejki, Tematy, Subskrypcje lub Komunikacja, aby wybrać typ obiektu, z którym chcesz pracować.
3. Szczegółowe instrukcje dotyczące pracy z tym typem obiektu można znaleźć w jednym z poniższych tematów.

V 9.2.0 Praca z kolejkami

Na karcie **Kolejki** można wyświetlić kolejki, które istnieją dla konkretnego menedżera kolejek. Można dodawać i usuwać kolejki, dodawać i usuwać komunikaty w kolejce, przeglądać komunikaty, wyświetlać i ustawiać właściwości kolejki, a także zarządzać rekordami uprawnień kolejki.


O tym zadaniu

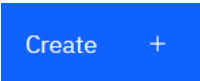
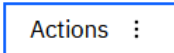




Widok kolejek zawiera listę kolejek, które istnieją dla konkretnego menedżera kolejek. Dostęp do listy kolejek można uzyskać, klikając menedżer kolejek i wybierając kartę **Kolejki** . Z listy można wybrać poszczególne kolejki, z którymi ma pracować praca.

z/OS Nie można wyświetlać ani edytować rekordów uprawnień dla kolejek w systemie z/OS.

Procedura

- Aby dodać kolejkę:

- a) Na karcie **Kolejki** kliknij przycisk tworzenia .
- b) Wybierz typ kolejki, która ma zostać utworzona:
- Lokalne komunikaty przechowywania w kolejce w menedżerze kolejek, do którego należy.
 - Kolejka aliasowa-wskaźnik do innej kolejki w tym samym menedżerze kolejek.
 - Kolejka zdalna-wskaźnik do innej kolejki w innym menedżerze kolejek.
 - Kolejka modelowa-szablon dla kolejki używanej podczas tworzenia dynamicznego menedżera kolejek.
- c) Podaj wymagane informacje na temat typu tworzonej kolejki. Domyślnie wyświetlane są minimalne zalecane właściwości, dla których można podać wartości. Wszystkie dostępne właściwości można wyświetlić, wybierając opcję **Pokaż wszystkie dostępne opcje**.
- d) Kliknij opcję **Create**. Zostanie utworzona nowa kolejka.
- Aby umieścić komunikaty w kolejce:
 - a) Kliknij kolejkę, do której mają zostać dodane komunikaty na liście w widoku listy kolejek. Nie można wybrać kolejki modelowej.

- b) Kliknij przycisk  **Utwórz**.
- c) Wprowadź komunikat, który ma zostać umieszczony w kolejce.
- d) Kliknij opcję **Create**.
- Aby wyczyścić komunikaty z kolejki:
 - a) Kliknij kolejkę lokalną, z której mają być usunięte komunikaty z listy kolejek.
 - b) Kliknij przycisk Działania  i wybierz opcję **Wyczyść komunikaty**.
 - c) Potwierdź, że chcesz wyczyścić kolejkę, klikając opcję **Wyczyść komunikaty**.
 - Aby przeglądać komunikaty w kolejce, należy kliknąć kolejkę w widoku listy kolejek. Zostanie wyświetlona lista komunikatów znajdujących się w tej kolejce.
 - Aby usunąć kolejkę:
 - a) Kliknij ikonę spanner  znajdującą się obok kolejki, która ma zostać usunięta.
 - b) W widoku Edytuj kolejkę kliknij opcję **Usuń kolejkę**.
 - c) Potwierdź, że chcesz usunąć kolejkę, klikając przycisk **Usuń**. Kolejka została usunięta.
 - Aby wyświetlić i edytować właściwości kolejki:
 - a) Kliknij ikonę spanner  znajdującą się obok kolejki, która ma być edytowana.
 - b) Kliknij przycisk  **Edytuj**.
 - c) Edytuj wybrane właściwości. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend. Więcej informacji na temat właściwości zawiera sekcja Właściwości kolejki w dokumentacji programu MQ Explorer .
 - d) Kliknij przycisk **Zapisz**, aby zapisać zmiany.
 - Aby wyświetlić i edytować rekordy uprawnień dla kolejki:
 - a) Kliknij ikonę spanner  znajdującą się obok kolejki, która ma być edytowana, dla której chcesz edytować rekordy uprawnień.
 - b) Kliknij kartę **Zabezpieczenia**.

- c) Praca z rekordami uprawnień w sposób opisany dla rekordów uprawnień menedżera kolejek. Patrz sekcja “Praca z rekordami uprawnień menedżera kolejek” na stronie 101.

V9.2.0 Praca z wątkami

Za pomocą programu IBM MQ Console można dodawać i usuwać tematy oraz wyświetlać i ustawiać właściwości tematu.

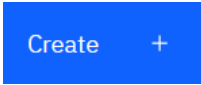
O tym zadaniu

W widoku tematów znajduje się lista tematów, które istnieją dla konkretnego menedżera kolejek. Dostęp do listy tematów można uzyskać, klikając menedżer kolejek i wybierając kartę **Tematy**. 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:

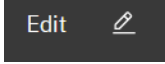
- a) Na karcie **Tematy** kliknij przycisk tworzenia .
- b) Podaj wymagane informacje dla tworzonego tematu. Domyślnie wyświetlane są minimalne zalecane właściwości, dla których można podać wartości. Wszystkie dostępne właściwości można wyświetlić, wybierając opcję **Pokaż wszystkie dostępne opcje**.
- c) Kliknij opcję **Create**. Zostanie utworzony nowy temat.

- Aby usunąć temat:

- a) Kliknij ikonę spanner  znajdującą się obok tematu, który ma zostać usunięty.
- b) W widoku Edytuj kolejkę kliknij opcję **Usuń temat**.
- c) Potwierdź, że chcesz usunąć temat, klikając przycisk **Usuń**. Temat zostanie usunięty.

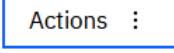
- Aby wyświetlić i edytować właściwości tematu, wykonaj następujące czynności:

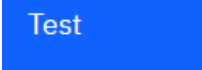
- a) Kliknij ikonę spanner  znajdującą się obok tematu, który ma być edytowany.

- b) Kliknij przycisk Edytuj .
- c) Edytuj wybrane właściwości. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend. Więcej informacji na temat właściwości zawiera sekcja Właściwości tematu w dokumentacji programu MQ Explorer.
- d) Kliknij przycisk **Zapisz**, aby zapisać zmiany.

- Aby opublikować komunikat w temacie, należy mieć co najmniej jedną zgodną subskrypcję. Subskrypcję testową można utworzyć, jeśli jest to wymagane:

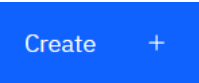
- a) Kliknij temat, który ma zostać opublikowany na liście tematów.
- b) Następne działanie zależy od tego, czy w tym temacie są zgodne subskrypcje:
Jeśli nie ma zgodnych subskrypcji:

- a. Kliknij przycisk Działania  i wybierz opcję **Testuj temat**.

- b. Kliknij przycisk Test . Komunikat testowy jest zapisywany w subskrypcji testowej.

Jeśli istnieją zgodne subskrypcje tematu, wykonaj następujące czynności:

a. Kliknij nazwę subskrypcji.


A blue rectangular button with the text "Create" and a plus sign "+" to its right.

b. Kliknij przycisk Utwórz

c. Wprowadź komunikat, który ma zostać opublikowany.

A blue rectangular button with the text "Put" in white.

d. Kliknij przycisk Wstaw. Komunikat jest zapisywany we wszystkich zgodnych subskrypcjach.

- Aby zasubskrybować temat, patrz [“Praca z subskrypcjami”](#) na stronie 110:
- Aby wyświetlić i edytować rekordy uprawnień dla tematu, wykonaj następujące czynności:
 - a) Kliknij ikonę spanner  znajdującą się obok tematu, dla którego mają być edytowane rekordy uprawnień.
 - b) Kliknij kartę **Zabezpieczenia**.
 - c) Praca z rekordami uprawnień opisanymi dla rekordów uprawnień menedżera kolejek znajduje się w sekcji [“Praca z rekordami uprawnień menedżera kolejek”](#) na stronie 101.

V 9.2.0

Praca z subskrypcjami

Za pomocą programu IBM MQ Console można dodawać i usuwać subskrypcje oraz wyświetlać i ustawiać właściwości subskrypcji.

O tym zadaniu

W widoku subskrypcji znajduje się lista subskrypcji, które istnieją dla konkretnego menedżera kolejek. Dostęp do listy subskrypcji można uzyskać, klikając menedżer kolejek i wybierając kartę **Subskrypcje**. Z listy można wybrać pojedyncze subskrypcje, z którymi ma pracować praca.

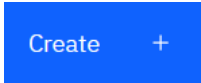
Więcej informacji na temat subskrypcji zawiera sekcja [Subskrybenci i subskrypcje](#) oraz sekcja [DEFINIOWANIE SUBSKRYPCJI PODRZĘDNEJ](#).

z/OS

Nie można wyświetlać ani edytować rekordów uprawnień dla subskrypcji w systemie z/OS.

Procedura

- Aby dodać subskrypcję:

A blue rectangular button with the text "Create" and a plus sign "+" to its right.


a) Na karcie **Subskrypcje** kliknij przycisk tworzenia

b) Wybierz, czy ma zostać utworzona subskrypcja zarządzana, czy niezarządzana.

c) Podaj wymagane informacje dla tworzonej subskrypcji. Domyślnie wyświetlane są minimalne zalecane właściwości, dla których można podać wartości. Wszystkie dostępne właściwości można wyświetlić, wybierając opcję **Pokaż wszystkie dostępne opcje**.

d) Kliknij opcję **Create**. Zostanie utworzona nowa subskrypcja.


- Aby usunąć subskrypcję:

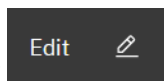
a) Kliknij ikonę spanner  znajdującą się obok subskrypcji, która ma zostać usunięta.

b) W widoku Edytuj kolejkę kliknij opcję **Usuń subskrypcję**.

c) Potwierdź, że chcesz usunąć subskrypcję, klikając przycisk **Usuń**. Subskrypcja została usunięta.

- Aby wyświetlić i edytować właściwości subskrypcji:

a) Kliknij ikonę spanner  znajdującą się obok subskrypcji, która ma być edytowana.



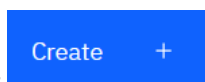
b) Kliknij przycisk Edytuj

c) Edytuj wybrane właściwości. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend.

d) Kliknij przycisk **Zapisz**, aby zapisać zmiany.

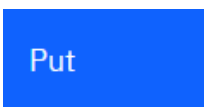
• Aby opublikować komunikat w temacie, w którym subskrypcja została zasubskrybowana:

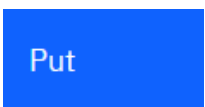
a) Kliknij subskrypcję, której temat ma zostać opublikowany na liście subskrypcji.



b) Kliknij przycisk Utwórz

c) Wprowadź komunikat, który ma zostać opublikowany.



d) Kliknij przycisk Wstaw . Komunikat jest zapisywany we wszystkich subskrypcjach, które są zgodne z tematem, do którego opublikowano.

Praca ze nasłuchiwaniami

Za pomocą IBM MQ Console można dodawać i usuwać obiekty nasłuchiwania, uruchamiać i zatrzymują obiekty nasłuchiwania, wyświetlać i ustawiać właściwości nasłuchiwania oraz zarządzać rekordami uprawnień dla obiektu nasłuchiwania.

O tym zadaniu

W widoku nasłuchiwania wyświetlane są obiekty nasłuchiwania, które istnieją dla konkretnego menedżera kolejek. Można wybrać poszczególne programy nasłuchujące do pracy.

Procedura

• Aby utworzyć obiekt nasłuchiwania:

a) Na karcie **Komunikacja** upewnij się, że widok nasłuchiwania jest wyświetlany, a następnie kliknij



b) Podaj wymagane informacje dla tworzonego obiektu nasłuchiwania. Domyślnie wyświetlane są minimalne zalecane właściwości, dla których można podać wartości. Wszystkie dostępne właściwości można wyświetlić, wybierając opcję **Pokaż wszystkie dostępne opcje**.

c) Kliknij opcję **Create**. Zostanie utworzony nowy obiekt nasłuchiwania.

• Aby uruchomić program nasłuchujący:

a) Znajdź nasłuchiwanie, które ma zostać uruchomione na liście.

b) Wybierz opcję **Start** z menu .



• Aby zatrzymać proces nasłuchujący:

a) Znajdź nasłuchiwanie, które ma zostać uruchomione na liście.

b) Wybierz opcję **Zatrzymaj** z menu .

• Aby wyświetlić i edytować właściwości obiektu nasłuchiwania:

a) Znajdź obiekt nasłuchiwania na liście.


- b) Wybierz opcję **Konfiguruj** z menu  .
 - c) Upewnij się, że zaznaczona jest karta **Właściwości** . Aby edytować właściwości, kliknij przycisk  .
Edytuj
 - d) Edytuj wybrane właściwości. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend. Więcej informacji na temat właściwości zawiera sekcja [Właściwości obiektu nasłuchiwania](#) w dokumentacji programu MQ Explorer.
 - e) Kliknij przycisk **Zapisz**, aby zapisać zmiany.
- Aby wyświetlić i edytować rekordy uprawnień dla obiektu nasłuchiwania:
 - a) Znajdź obiekt nasłuchiwania na liście.
 - b) Wybierz opcję **Konfiguruj** z menu  .
 - c) Kliknij kartę **Zabezpieczenia**.
 - d) Praca z rekordami uprawnień opisanymi dla rekordów uprawnień menedżera kolejek znajduje się w sekcji [“Praca z rekordami uprawnień menedżera kolejek”](#) na stronie 101.
 - Aby usunąć obiekt nasłuchiwania:
 - a) Znajdź obiekt nasłuchiwania na liście.
 - b) Wybierz opcję **Konfiguruj** z menu  .
 - c) Kliknij opcję **Usuń program nasłuchujący**.

Praca z kanałami menedżera kolejek

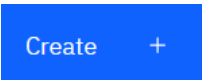

Za pomocą programu IBM MQ Console można korzystać z kanałów menedżera kolejek: można dodawać i usuwać kanały menedżera kolejek, uruchamiać i zatrzymują kanały, resetować i rozwiązywać kanały, a także kanały ping. Użytkownik może również wyświetlać i ustawiać właściwości kanału menedżera kolejek oraz zarządzać rekordami uprawnień dla kanału.




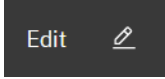




O tym zadaniu


Kanał menedżera kolejek jest logicznym łączem komunikacyjnym służącym do przesyłania komunikatów między menedżerami kolejek w sieci. Widok kanału menedżera kolejek zawiera panel, który umożliwia szybkie wyświetlanie liczby uruchomionych kanałów, liczbę powtórzeń i liczbę zatrzymanych kanałów.

 Nie można wyświetlać ani edytować rekordów uprawnień dla kanału w systemie z/OS.

Procedura

- Aby dodać kanał menedżera kolejek:
 - a) Na karcie **Komunikacja** upewnij się, że widok kanałów menedżera kolejek jest wyświetlany, a następnie kliknij przycisk tworzenia  .
 - b) Wybierz typ kanału menedżera kolejek, który ma zostać utworzony, a następnie kliknij przycisk  .
Dalej
 - c) Podaj wymagane informacje dla tworzonego kanału. Domyślnie wyświetlane są minimalne zalecane właściwości, dla których można podać wartości. Wszystkie dostępne właściwości można wyświetlić, wybierając opcję **Pokaż wszystkie dostępne opcje**.

- d) Kliknij opcję **Create**. Nowy kanał zostanie utworzony przy użyciu statusu **nieaktywna**.
- Aby uruchomić kanał menedżera kolejek:
 - a) Znajdź kanał, który ma zostać uruchomiony na liście.
 - b) Wybierz opcję **Start** z menu .
- Aby zatrzymać kanał menedżera kolejek:
 - a) Znajdź kanał, który ma zostać zatrzymany na liście.
 - b) Wybierz opcję **Zatrzymaj** z menu .
- Aby wyświetlić właściwości kanału menedżera kolejek:
 - a) Znajdź kanał na liście.
 - b) Wybierz opcję **Konfiguruj** z menu .
 - c) Upewnij się, że zaznaczona jest karta **Właściwości**. Aby edytować właściwości, kliknij przycisk  Edytuj
 - d) Edytuj wybrane właściwości. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend. Więcej informacji na temat właściwości można znaleźć w sekcji [Właściwości kanału](#) w dokumentacji programu MQ Explorer.
 - e) Kliknij przycisk **Zapisz**, aby zapisać zmiany.
- Aby zresetować kanał menedżera kolejek:
 - a) Znajdź kanał na liście.
 - b) Wybierz opcję **Zaawansowane** z menu .
 - c) W sekcji **Resetuj** 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 rozwiązać kanał wysyłający lub kanał serwera:
 - a) Znajdź kanał na liście.
 - b) Wybierz opcję **Zaawansowane** z menu .
 - c) W sekcji **Rozstrzygnij** wybierz, czy chcesz zatwierdzić, czy wycofać bieżące zadanie wsadowe komunikatów, klikając opcję **Odtwórz komunikaty do kolejki transmisji** lub **Odrzuć komunikaty**.
- Aby wykonać komendę ping dla kanału menedżera kolejek:
 - a) Znajdź kanał na liście.
 - b) Wybierz opcję **Ping** z menu .
- Aby wyświetlić i edytować rekordy uprawnień dla kanału menedżera kolejek:
 - a) Znajdź kanał na liście.
 - b) Wybierz opcję **Konfiguruj** z menu .
 - c) Kliknij kartę **Zabezpieczenia**.


- d) Praca z rekordami uprawnień opisanymi dla rekordów uprawnień menedżera kolejek znajduje się w sekcji “Praca z rekordami uprawnień menedżera kolejek” na stronie 101.
- Aby usunąć kanał menedżera kolejek:
 - a) Znajdź kanał na liście.
 - b) Wybierz opcję **Konfiguruj** z menu  .
 - c) Kliknij opcję **Usuń kanał**.

Praca z kanałami aplikacji

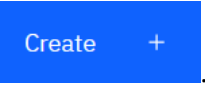




Za pomocą programu IBM MQ Console można pracować z kanałami aplikacji: można dodawać i usuwać kanały, uruchamiać i zatrzymują kanały, resetować i rozwiązywać kanały, a także kanały ping. Użytkownik może również wyświetlać i ustawiać właściwości kanału aplikacji oraz zarządzać rekordami uprawnień dla kanału.

O tym zadaniu

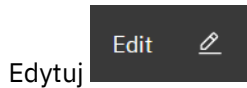
Kanał aplikacji jest logicznym łączem komunikacyjnym używanym przez aplikacje do nawiązywania połączenia z menedżerem kolejek w sieci. Widok kanału aplikacji zawiera panel, który umożliwi szybkie wyświetlanie liczby uruchomionych kanałów, liczbę powtórzenia i liczbę zatrzymanych kanałów.

 Nie można wyświetlać ani edytować rekordów uprawnień dla kanału w systemie z/OS.






Procedura

- Aby dodać kanał aplikacji:
 - a) Na karcie **Komunikacja** upewnij się, że widok kanałów aplikacji jest wyświetlany, a następnie kliknij przycisk tworzenia  .
 - b) Kliknij przycisk Dalej  .
 - c) Podaj wymagane informacje dla tworzonego kanału. Domyślnie wyświetlane są minimalne zalecane właściwości, dla których można podać wartości. Wszystkie dostępne właściwości można wyświetlić, wybierając opcję **Pokaż wszystkie dostępne opcje**.
 - d) Kliknij opcję **Create**. Nowy kanał zostanie utworzony przy użyciu statusu **nieaktywna** .
- Aby uruchomić kanał aplikacji:
 - a) Znajdź kanał, który ma zostać uruchomiony na liście.
 - b) Wybierz opcję **Start** z menu  .
- Aby zatrzymać kanał aplikacji:
 - a) Znajdź kanał, który ma zostać zatrzymany na liście.
 - b) Wybierz opcję **Zatrzymaj** z menu  .
- Aby wyświetlić właściwości kanału aplikacji:
 - a) Znajdź kanał na liście.
 - b) Wybierz opcję **Konfiguruj** z menu  .

c) Upewnij się, że zaznaczona jest karta **Właściwości**. Aby edytować właściwości, kliknij przycisk




Edytuj

- d) Edytuj wybrane właściwości. Jeśli pole tekstowe właściwości jest wyłączone, właściwość ta jest tylko do odczytu lub może być edytowana tylko z poziomu wiersza komend. Więcej informacji na temat właściwości można znaleźć w sekcji [Właściwości kanału](#) w dokumentacji programu MQ Explorer.
- e) Kliknij przycisk **Zapisz**, aby zapisać zmiany.
- Aby zresetować kanał aplikacji, wykonaj następujące czynności:
 - a) Znajdź kanał na liście.
 - b) Wybierz opcję **Zaawansowane** z menu .
 - c) W sekcji **Resetuj** 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 rozwiązać kanał wysyłający lub kanał serwera:
 - a) Znajdź kanał na liście.
 - b) Wybierz opcję **Zaawansowane** z menu .
 - c) W sekcji **Rozstrzygnij** wybierz, czy chcesz zatwierdzić, czy wycofać bieżące zadanie wsadowe komunikatów, klikając opcję **Odtwórz komunikaty do kolejki transmisji** lub **Odrzuć komunikaty**.
 - Aby wykonać komendę ping dla kanału:
 - a) Znajdź kanał na liście.
 - b) Wybierz opcję **Ping** z menu .
 - Aby wyświetlić i edytować rekordy uprawnień dla kanału aplikacji, wykonaj następujące czynności:
 - a) Znajdź kanał na liście.
 - b) Wybierz opcję **Konfiguruj** z menu .
 - c) Kliknij kartę **Zabezpieczenia**.
 - d) Praca z rekordami uprawnień opisanymi dla rekordów uprawnień menedżera kolejek znajduje się w sekcji ["Praca z rekordami uprawnień menedżera kolejek"](#) na stronie 101.
 - Aby usunąć kanał aplikacji:
 - a) Znajdź kanał na liście.
 - b) Wybierz opcję **Konfiguruj** z menu .
 - c) Kliknij opcję **Usuń kanał**.

V 9.2.0 Ustawienia konsoli WWW

Istnieje możliwość określenia niektórych ustawień ogólnych dla nowej konsoli WWW.

Kliknij ikonę ustawień  **Settings**, aby przetączyć się do widoku ustawień konsoli WWW.

Za pomocą ustawień można sterować następującymi opcjami:

- Automatyczne odświeżanie menedżerów kolejek co dziesięć sekund. Ta funkcja może być włączona lub wyłączona.
- Określa, czy wyświetlane są obiekty systemowe. Można określić tę wartość dla wszystkich typów obiektów lub wybrać poszczególne typy obiektów.
- Określa, czy informacje śledzenia są gromadzone, czy nie.

Multi

z/OS

V 9.2.0

Przełączanie między typami konsoli



Użytkownik może przełączać się między New Web Console (domyślną konsolą WWW dla produktu IBM MQ 9.2) a Dashboard Web Console (która była konsolą WWW dla wcześniejszych wersji produktu IBM MQ).

O tym zadaniu

Przełączanie między konsolami jest możliwe za pomocą komendy **setmqweb**.

Jeśli użytkownik przełączy się ponownie za pomocą konsoli Dashboard Web Console, należy zapoznać się z informacjami w sekcji [Konsola WWW panelu kontrolnego](#) w dokumentacji produktu IBM MQ 9.1 . dla instrukcji użycia.

Aby wykonać tę czynność, użytkownik musi mieć pewne uprawnienia pozwalające na użycie komend **dspmqweb** i **setmqweb**:

-  W systemie z/OS musisz mieć uprawnienie do uruchamiania komend **dspmqweb** i **setmqweb** oraz uprawnienie do zapisu w pliku `mqwebuser.xml`.
-  W przypadku wszystkich innych systemów operacyjnych użytkownik musi być [użytkownikiem uprzywilejowanym](#).



Ostrzeżenie:

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`. Na przykład:

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

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

Procedura

- Aby przełączyć się na Dashboard Web Console, należy użyć następujących komend:
 - Ustaw właściwość `mqConsoleEnableUnsafeInline` na wartość `true`, wprowadzając następującą komendę:

```
setmqweb properties -k mqConsoleEnableUnsafeInline -v true
```

Należy zwrócić uwagę, że ustawienie tej właściwości rozluźnia zestaw reguł CSP w celu włączenia funkcji `unsafe-inline`, co może spowodować, że konfiguracja jest mniej bezpieczna, ale jest wymagana przez Dashboard Web Console.

- Przełącz serwer mqweb, aby używany był Dashboard Web Console, wprowadzając następującą komendę:

```
setmqweb properties -k mqConsoleEarName -v com.ibm.mq.console
```

- Aby powrócić do ustawień domyślnych dla serwera WWW mq (w tym New Web Console), użyj następującej komendy:

```
setmqweb properties -r
```

- Aby powrócić do New Web Console podczas zachowywania innych dostosowań wprowadzonych na serwerze mqweb, należy użyć następujących komend:

- Ustaw właściwość Inline mqConsoleEnableUnsafeInline na wartość false, aby przywrócić konfigurację zabezpieczeń, wprowadzając następującą komendę:

```
setmqweb properties -k mqConsoleEnableUnsafeInline -v false
```

- Przełącz serwer mqweb, aby używany był New Web Console , wprowadzając następującą komendę:

```
setmqweb properties -k mqConsoleEarName -v com.ibm.mq.webconsole
```

Odsyłacze pokrewne

[setmqweb](#)

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 dla systemu Linux x86-64 udostępniają interfejs administracyjny o nazwie IBM MQ Explorer , który umożliwia wykonywanie zadań administracyjnych jako alternatywa dla używania komend sterujących lub komend 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 119.

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.

Produkt IBM MQ Explorer można zainstalować w ramach instalacji produktu (patrz sekcja [Instalowanie i deinstalowanie produktu IBM MQ](#)) lub z poziomu autonomicznego produktu IBM MQ Explorer dostępnego w produkcie Fix Central (patrz sekcja [Instalowanie i deinstalowanie produktu IBM MQ Explorer jako autonomiczna aplikacja w systemach Linux i Windows](#)).

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).
- Definiowanie, wyświetlanie i modyfikowanie definicji obiektów produktu IBM MQ , takich jak kolejki i kanały.
- Przeglądanie komunikatów w kolejce.
- Uruchamianie i zatrzymywanie kanału.
- Wyświetl informacje o statusie dotyczące kanału, obiektu nasłuchiwanego, kolejki lub obiektów usług.
- Wyświetlanie menedżerów kolejek w klastrze.
- Sprawdź, czy aplikacje, użytkownicy lub kanały mają otwartą określoną kolejkę.
- Utwórz nowy klaster menedżera kolejek przy użyciu kreatora tworzenia nowego klastra.
- Dodaj menedżera kolejek do klastra przy użyciu kreatora dodawania menedżera kolejek do klastra.
- Zarządzaj obiektem informacji uwierzytelniającej, który jest używany z zabezpieczeniami kanału TLS (Transport Layer Security).
- Tworzenie i usuwanie inicjatorów kanałów, monitorów wyzwalaczy i programów nasłuchujących.
- Uruchom lub zatrzymaj serwery komend, inicjatory kanału, monitory wyzwalacza i programy nasłuchujące.
- Ustaw konkretne usługi na uruchom automatycznie, gdy uruchamiany jest menedżer kolejek.
- Zmodyfikuj właściwości menedżerów kolejek.
- Zmień lokalny domyślny menedżer kolejek.
- Wywołaj interfejs GUI programu **strmqikm** (ikeyman) , aby 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 i IBM MQ obiekty 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 nasłuchiwanego 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.

Klustry

Klustry 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 119](#)
- [“zabezpieczenia dla IBM MQ Explorer” na stronie 120](#)
- [“Wyświetlanie i ukrywanie menedżerów kolejek i klastrów w produkcie IBM MQ Explorer” na stronie 123](#)
- [“Przypisanie do klastra i IBM MQ Explorer” na stronie 124](#)
- [“Konwersja danych dla IBM MQ Explorer” na stronie 125](#)

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 AIX 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 IBM MQ zawiera sekcja [Udzielanie dostępu do obiektu IBM MQ w systemach AIX, Linux, and 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 za pomocą 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 121 i [“Zadania w systemie, w którym znajduje się IBM MQ Explorer”](#) na stronie 122.

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 for AIX or Linux systems.
-  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). GUI lub za pomocą opcji -stash komendy **runmqckm** lub komendy **runmqakm** .
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 AIX, 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 MQExplorer lub komendy **strmqcfg** .
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 122, 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 121 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 deskrytorze 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.

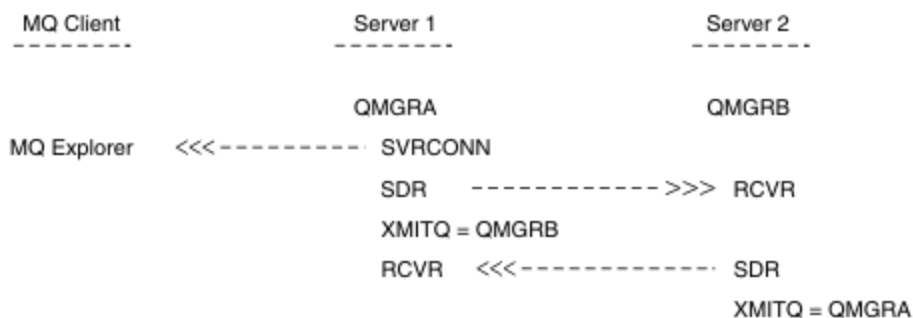
Jeśli połączenie jest następnie używane do łączenia się z docelowym menedżerem kolejek za pośrednictwem pośredniego menedżera kolejek, identyfikator userId jest ponownie generowany w parametrze *UserIdentifier* deskryptora komunikatu (MQMD). Aby program nasłuchujący MCA w docelowym menedżerze kolejek 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 [“Konfigurowanie menedżerów kolejek na potrzeby zdalnego administrowania” na stronie 192](#).

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, klaster 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 **Przeglądaj** 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](#).

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.

Aplikacja monitora alertów IBM MQ (tylko Windows)

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

Praca z 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 117.

Procedura

- Informacje zawarte w poniższych tematach ułatwiają administrowanie lokalnymi obiektami produktu IBM MQ.
 - [Programy aplikacji korzystające z interfejsu MQI](#)
 - [“Administrowanie za pomocą komend MQSC”](#) na stronie 12
 - [“Wyświetlanie i modyfikowanie atrybutów menedżera kolejek”](#) na stronie 134
 - [“Praca z kolejkami lokalnymi”](#) na stronie 137
 - [“Praca z kolejkami aliasowymi”](#) na stronie 149
 - [“Praca z kolejkami modelami”](#) na stronie 151
 - [“Praca z usługami”](#) na stronie 178
 - [“Zarządzanie obiektami w celu wyzwolenia”](#) na stronie 186

Praca z menedżerami kolejek

Do uruchamiania i zatrzymywania menedżera kolejek można używać komend sterujących. Aby wyświetlić lub zmienić atrybuty menedżera kolejek, można użyć komend MQSC.

Zadania pokrewne

[Tworzenie menedżerów kolejek na wielu platformach](#)

Uruchamianie menedżera kolejek

Po utworzeniu menedżera kolejek należy go uruchomić, aby umożliwić mu przetwarzanie komend lub wywołań MQI.

O tym zadaniu

Menedżer kolejek można uruchomić za pomocą komendy **strmqm**. Opis komendy **strmqm** i jej opcji można znaleźć w sekcji [strmqm](#).

Windows **Linux** Alternatywnie w systemach Windows i Linux (platformy x86 i x86-64) można uruchomić menedżer kolejek przy użyciu konsoli IBM MQ Explorer.

Windows W systemie Windows menedżer kolejek może być uruchamiany automatycznie podczas uruchamiania systemu za pomocą konsoli IBM MQ Explorer. Więcej informacji na ten temat zawiera sekcja [“Administrowanie przy użyciu IBM MQ Explorer” na stronie 117](#).

Procedura

- Aby uruchomić menedżer kolejek za pomocą komendy **strmqm**, należy wprowadzić komendę, po której następuje nazwa menedżera kolejek, który ma zostać uruchomiony.

Na przykład, aby uruchomić menedżer kolejek o nazwie QMB, należy wprowadzić następującą komendę:

```
strmqm QMB
```

Uwaga: Z instalacji powiązanej z menedżerem kolejek, z którym pracuje użytkownik, należy użyć komendy **strmqm**. Za pomocą komendy `dspmqr -o installation` można dowiedzieć się, która instalacja menedżera kolejek jest powiązana.

Komenda **strmqm** nie zwraca elementu sterującego, dopóki menedżer kolejek nie zostanie uruchomiony i będzie gotowy do akceptowania żądań połączenia.

- Windows** **Linux**
Aby uruchomić menedżer kolejek przy użyciu konsoli IBM MQ Explorer, wykonaj następujące kroki:
 - Otwórz produkt IBM MQ Explorer.
 - W widoku Navigator wybierz menedżer kolejek.
 - Kliknij polecenie **Uruchom**.

Wyniki

Zostanie uruchomiony menedżer kolejek.

Jeśli uruchomienie menedżera kolejek trwa dłużej niż kilka sekund, program IBM MQ wysyła komunikaty informacyjne w sposób sporny, w którym należy określić postęp uruchamiania.

Multi Zatrzymywanie menedżera kolejek

Aby zatrzymać menedżer kolejek, można użyć komendy **endmqm**. Ta komenda udostępnia cztery sposoby zatrzymania menedżera kolejek: kontrolowane lub wyciszone, zamknięcie, natychmiastowe zamknięcie systemu, zamknięcie z góry i zamknięcie oczekiwania. Alternatywnie w systemach Windows i Linux można zatrzymać menedżer kolejek za pomocą konsoli IBM MQ Explorer.

O tym zadaniu

Istnieją cztery sposoby zatrzymania menedżera kolejek z jedną instancją za pomocą komendy **endmqm**:

Kontrolowane (wygaszone) zamknięcie systemu

Domyślnie komenda **endmqm** wykonuje wygaszane zamknięcie określonego menedżera kolejek.

Wygaszone zamknięcie czeka do czasu, aż wszystkie połączone aplikacje zostaną odłączone, więc może zająć trochę czasu.

natychmiastowe zatrzymanie

W przypadku natychmiastowego zamknięcia wszystkie bieżące wywołania MQI są dozwolone do zakończenia, ale żadne nowe wywołania nie powiodą się. Ten typ zamknięcia systemu nie czeka na rozłączenie aplikacji z menedżerem kolejek.

zamknięcie systemu z wywłaszczaniem

Menedżer kolejek zostanie natychmiast zatrzymany. Tego typu wyłączenia należy używać tylko w wyjątkowych okolicznościach, na przykład gdy menedżer kolejek nie jest zatrzymany w wyniku normalnej komendy **endmqm**.

Oczekiwanie na zamknięcie

Ten typ zamknięcia systemu jest równoważny z kontrolowanym zamknięciem, z wyjątkiem tego, że sterowanie jest zwracane do użytkownika dopiero po zatrzymaniu menedżera kolejek.

Komenda **endmqm** zatrzymuje wszystkie instancje menedżera kolejek z wieloma instancjami w taki sam sposób, w jaki zatrzymuje pojedynczy menedżer kolejek instancji. **endmqm** można wprowadzić w aktywnej instancji lub w jednej z instancji rezerwowych menedżera kolejek z wieloma instancjami. Należy jednak wydać komendę **endmqm** w aktywnej instancji, aby zakończyć działanie menedżera kolejek.

V 9.2.0 W programie IBM MQ 9.1.4 dostępna jest opcja zakończenia menedżera kolejek w czasie określonym przez określoną liczbę sekund. Więcej informacji na ten temat zawiera sekcja [“Kończenie menedżera kolejek w czasie docelowym”](#) na stronie 130.

Szczegółowy opis komendy **endmqm** i jej opcji można znaleźć w sekcji [endmqm](#).

Wskazówka: Problemy z zamknięciem menedżera kolejek są często spowodowane przez aplikacje. Na przykład, gdy aplikacje:

- Nie sprawdzają poprawnie kodów powrotu MQI
- Nie żądają powiadomienia o wygaszeniu
- Zakończ bez rozłączania z menedżerem kolejek (przez wywołanie wywołania MQDISC)

Jeśli wystąpi problem podczas próby zatrzymania menedżera kolejek, można przerwać wykonywanie komendy **endmqm** za pomocą kombinacji klawiszy Ctrl-C. Następnie można wprowadzić inną komendę **endmqm**, ale tym razem z parametrem określającym, jakiego typu jest wymagane wyłączenie.

Windows **Linux** Alternatywą dla użycia komendy **endmqm**, w systemach Windows i Linux, można zatrzymać menedżer kolejek przy użyciu konsoli IBM MQ Explorer w celu przeprowadzenia kontroli lub natychmiastowego zamknięcia systemu.

Procedura

- Aby zatrzymać menedżer kolejek za pomocą komendy **endmqm**, należy wprowadzić komendę, po której następuje odpowiedni parametr, jeśli jest to wymagane, oraz nazwę menedżera kolejek, który ma zostać zatrzymany.

Uwaga: Z instalacji powiązanej z menedżerem kolejek, z którym pracuje użytkownik, należy użyć komendy **endmqm**. Aby dowiedzieć się, z której instalacji jest powiązany menedżer kolejek, należy użyć komendy **dspmq**:

```
dspmq -o installation
```

- Aby przeprowadzić kontrolowane (wygaszane) zamknięcie systemu, należy wprowadzić komendę **endmqm**, jak pokazano w poniższym przykładzie, który zatrzymuje menedżer kolejek o nazwie QMB:

```
endmqm QMB
```

Alternatywnie, wprowadzenie komendy **endmqm** z parametrem **-c**, jak pokazano w poniższym przykładzie, jest równoznaczne z komendą **endmqm QMB**.

```
endmqm -c QMB
```


W obu przypadkach sterowanie jest zwracane do użytkownika natychmiast, a użytkownik nie jest powiadamiany, gdy menedżer kolejek został zatrzymany. Jeśli komenda ma czekać, aż wszystkie aplikacje zostaną zatrzymane, a menedżer kolejek zakończył działanie przed zwróceniem kontroli, należy użyć parametru **-w**, tak jak pokazano w poniższym przykładzie.

```
endmqm -w QMB
```

- Aby przeprowadzić natychmiastowe zamknięcie systemu, należy wprowadzić komendę **endmqm** z parametrem **-i**, jak pokazano w poniższym przykładzie:


```
endmqm -i QMB
```

- Aby przeprowadzić wstępne zamknięcie systemu, należy wprowadzić komendę **endmqm** z parametrem **-p**, jak to pokazano w poniższym przykładzie:

```
endmqm -p QMB
```



Ostrzeżenie: Wyłączenie z góry może mieć nieprzewidywalne konsekwencje dla połączonych aplikacji. Nie należy używać tej opcji, chyba że wszystkie inne próby zatrzymania menedżera kolejek przy użyciu normalnej komendy **endmqm** nie powiodły się.

 Jeśli wyłączenie z góry nie działa, należy zamiast tego spróbować “Ręczne zatrzymywanie menedżera kolejek” na stronie 130.

- Aby zażądać automatycznego ponownego nawiązania połączenia z klientem, wprowadź komendę **endmqm** z parametrem **-r**. Ten parametr ma wpływ na ponowne nawiązanie połączenia klientów z innymi menedżerami kolejek w ich grupie menedżerów kolejek.

Uwaga: Zakończenie menedżera kolejek przy użyciu domyślnej komendy **endmqm** nie powoduje wyzwolenia automatycznego ponownego połączenia z klientem.

- Aby przestać do instancji rezerwowej menedżera kolejek z wieloma instancjami po zamknięciu aktywnej instancji, należy wprowadzić komendę **endmqm** z parametrem **-s** w aktywnej instancji menedżera kolejek z wieloma instancjami.
- Aby zakończyć instancję rezerwową menedżera kolejek z wieloma instancjami i pozostawić działającą instancję aktywną, należy wprowadzić komendę **endmqm** z parametrem **-x** w instancji rezerwowej menedżera kolejek z wieloma instancjami.

W systemach Windows i Linux, aby zatrzymać menedżer kolejek przy użyciu produktu IBM MQ Explorer, wykonaj następujące kroki:

- Otwórz produkt IBM MQ Explorer.
- Wybierz menedżera kolejek z widoku Navigator.
- Kliknij polecenie **Zatrzymaj**.
Zostanie wyświetlony panel **Menedżer kolejek końcowych**.
- Wybierz opcję **Kontrolowane** lub **Natychmiast**.
- Kliknij przycisk **OK**.
Menedżer kolejek zostanie zatrzymany.

Zadania pokrewne

Stosowanie aktualizacji poziomu konserwacyjnego do menedżerów kolejek z wieloma instancjami w systemie AIX

Stosowanie aktualizacji poziomu konserwacyjnego do menedżerów kolejek z wieloma instancjami w systemie Linux

Stosowanie aktualizacji poziomu konserwacyjnego do menedżerów kolejek z wieloma instancjami w systemie Windows

Kończenie menedżera kolejek w czasie docelowym

Menedżer kolejek można zakończyć w czasie określonym przez określoną liczbę sekund, z lub bez konieczności przerywania ważnych zadań konserwacji menedżera kolejek.

Istnieją dwa sposoby określania czasu docelowego, gdy używana jest komenda **endmqm**. Opcja **-t** umożliwia wykonanie niezbędnych zadań konserwacji menedżera kolejek, co może wydłużyć fazę zakończenia menedżera kolejek. Opcja **-tp** przerywa niezbędne zadania konserwacji menedżera kolejek, jeśli jest to konieczne do wykonania określonego czasu docelowego.

Jeśli zostanie określony czas docelowy, typ zamknięcia systemu **-w**, **-i** lub **-p** wskazuje typ zamknięcia systemu.

Uwaga: Zamknięcie systemu **immediate** jest nadal uporządkowane, różniące się od zamknięcia systemu **controlled** przede wszystkim w taki sposób, że wszystkie działające aplikacje są wygaszane. Zamknięcie systemu **immediate** nadal wykonuje czynności porządkowe, takie jak upakowanie kolejki i utrwalanie komunikatów NPMCLASS (HIGH), które mogą być czasochłonne, podczas gdy zamknięcie systemu kończy działanie tych dodatkowych działań, gdy będą one zakłócać czas realizacji docelowego.

Menedżer kolejek eskaluje typ zamknięcia, jeśli jest to konieczne, w celu spełnienia docelowego czasu. Na przykład:

- 10 sekundy-sekunda **-t**, rozpoczynająca się od **-w**, może być wygaszona siedem sekund, natychmiastowe zamknięcie menedżera kolejek, w tym dwie sekundy, natychmiastowe zamknięcie systemu, a następnie natychmiastowe zamknięcie systemu bez dalszej konserwacji:

```
endmqm -w -t 10 queue_manager
```

- 10 sekunda **-tp** może być wygaszona przez siedem sekund, natychmiastowe zamknięcie menedżera kolejek, w tym dwie sekundy, natychmiastowe zamknięcie systemu bez dalszej konserwacji, a następnie uruchomienie zakończenia IBM MQ procesów:

```
endmqm -c -tp 10 queue_manager
```

- Dwa drugie miejsce docelowe **-tp** na serwerze **-i** może być jednym drugim natychmiastowym zamknięciem menedżera kolejek, w tym porządkiem porządkowania, jednym drugim natychmiastowym zamknięciem bez dalszej konserwacji, a następnie uruchamiają zakończenie procesów IBM MQ :

```
endmqm -i -tp 2 queue_manager
```

- Drugim celem w wersji **-w** może być 0.1 sekund na **wait**, na przykład, na tyle długo, aby wysłać kody powrotu IBM MQ do połączonych aplikacji, 0.9 sekund natychmiastowe zamknięcie menedżera kolejek, w tym porządkowanie, a następnie natychmiastowe zamknięcie bez dalszej konserwacji, a następnie rozpocząć zakończenie procesów IBM MQ .

Odsyłacze pokrewne

[endmqm \(zakończenie menedżera kolejek\)](#)

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” na stronie 127. 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

- 

Informacje na temat zatrzymywania menedżera kolejek w systemie Windows zawiera sekcja [“Ręczne zatrzymywanie menedżera kolejek w systemie Windows”](#) na stronie 131.



Aby zatrzymać menedżera kolejek w systemie AIX lub Linux, należy zapoznać się z [“Ręczne zatrzymywanie menedżera kolejek w systemie AIX and Linux”](#) na stronie 132.

Zadania pokrewne

[Tworzenie i zarządzanie menedżerami kolejek na wielu platformach](#)

Odsyłacze pokrewne

[endmqm](#)

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* (`proceexp.exe`), które jest dostępne do pobrania z serwisu WWW produktu Microsoft pod adresem <http://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):

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
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)

Tabela 7. Procesy Windows , które mają zostać zatrzymane, jeśli są uruchomione (kontynuacja)	
Nazwa procesu	Opis
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

- Zatrzymaj usługę IBM MQ z poziomu opcji **Narzędzia administracyjne > Usługi** na panelu sterowania programu Windows .
- Jeśli próbowano wszystkich metod, a menedżer kolejek nie został zatrzymany, zrestartuj system.

Linux AIX **Ręczne zatrzymywanie menedżera kolejek w systemie AIX and**

Linux

Jeśli nie można zatrzymać menedżera kolejek w systemie AIX lub Linux za pomocą komendy **endmqm** , można spróbować ręcznie zatrzymać menedżer kolejek, kończąc wszystkie działające procesy i zatrzymując usługę IBM MQ .

O tym zadaniu

Aby zatrzymać menedżer kolejek w systemie AIX lub 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

- 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
```

- 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 za pomocą identyfikatorów PID, które mają być zabite, jedno po jednym, za każdym razem wydając tę komendę.

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 8. Procesy AIX and 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
runmqlsr	Proces nastuchiwania kanału

Uwaga: Komendy **kill -9** można użyć do zakończenia procesów, które nie zostały zatrzymane.

Multi Restartowanie menedżera kolejek

Za pomocą komendy **strmqm** w celu zrestartowania menedżera kolejek lub w systemach Windows i Linux x86-64 można zrestartować menedżer kolejek z programu IBM MQ Explorer.

O tym zadaniu

Menedżer kolejek można zrestartować za pomocą komendy **strmqm**. Opis komendy **strmqm** i jej opcji można znaleźć w sekcji [strmqm](#).



Windows **Linux** W systemach Windows i Linux x86-64 można zrestartować menedżer kolejek, korzystając z IBM MQ Explorer w ten sam sposób, co w przypadku uruchamiania menedżera kolejek.

Procedura

- Aby zrestartować menedżer kolejek za pomocą komendy **strmqm**, należy wprowadzić komendę, po której następuje nazwa menedżera kolejek, który ma zostać zrestartowany.

Na przykład, aby uruchomić menedżer kolejek o nazwie `strmqm saturn.queue.manager`, należy wprowadzić następującą komendę:

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

-   Aby uruchomić menedżer kolejek przy użyciu konsoli IBM MQ Explorer, wykonaj następujące kroki:
 - a) Otwórz produkt IBM MQ Explorer.
 - b) W widoku Navigator wybierz menedżer kolejek.
 - c) Kliknij polecenie **Uruchom**.

Wyniki

Zostanie zrestartowany menedżer kolejek.

Jeśli restart menedżera kolejek trwa dłużej niż kilka sekund, program IBM MQ wysyła komunikaty informacyjne w sposób sporny, w którym szczegółowo określa się postęp uruchamiania.

Wyświetlanie i modyfikowanie atrybutów menedżera kolejek

Aby wyświetlić lub zmienić atrybuty menedżera kolejek, można użyć komend MQSC.

O tym zadaniu

Komenda **DISPLAY QMGR** służy do wyświetlania parametrów menedżera kolejek dla menedżera kolejek i komendy **ALTER QMGR** w celu zmiany parametrów menedżera kolejek dla lokalnego menedżera kolejek.

Procedura

- Aby wyświetlić atrybuty menedżera kolejek określonego w komendzie `runmqsc`, należy użyć komendy MQSC **DISPLAY QMGR**:

```
DISPLAY QMGR
```

W poniższym przykładzie przedstawiono typowe dane wyjściowe tej komendy:

```
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)
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)
```

PSSYNCP (IFPER)	QMID(QM1_2011-05-27_16.14.01)
PSMODE (ENABLED)	REMOETE (DISABLED)
REPOS ()	REPOSNL ()
ROUTEREC (MSG)	SCHINIT (QMGR)
SCMDSERV (QMGR)	SSLCLNL ()
SSLCRYP ()	SSLEV (DISABLED)
SSLFIPS (NO)	SSLKEYR (C:\Program Files\IBM\WebSphere
MQ\Data\qmgrs\QM1\ssl\key)	
SSLKEYC (0)	STATACLS (QMGR)
STATCHL (OFF)	STATINT (1800)
STATMQI (OFF)	STATQ (OFF)
STRSTPEV (ENABLED)	SYNCP
TREELIFE (1800)	TRIGINT (999999999)

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 today SYSTEM.*. Nawiasy są wymagane.

- 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.

Parametry, które nie zostały podane w komendzie **ALTER QMGR**, powodują, że istniejące wartości tych parametrów pozostaną niezmienione.

Zadania pokrewne

[Tworzenie menedżerów kolejek na wielu platformach](#)

Odsyłacze pokrewne

[Atrybuty dla menedżera kolejek](#)

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

[WYŚWIETL QMGR](#)

[ALTER QMGR](#)

Multi Usuwanie menedżera kolejek

Menedżer kolejek można usunąć za pomocą komendy **dltmqm**. Alternatywnie w systemach Windows i Linux można użyć programu IBM MQ Explorer do usunięcia menedżera kolejek.

Zanim rozpocznesz



Ostrzeżenie:

- Usunięcie menedżera kolejek jest krokiem drastycznym, ponieważ wszystkie zasoby powiązane z menedżerem kolejek, w tym wszystkie kolejki i ich komunikaty oraz wszystkie definicje obiektów, są usuwane. Jeśli używana jest komenda **dltmqm**, nie zostanie wyświetlona zachęta, która umożliwia zmianę zdania. Po naciśnięciu klawisza Enter wszystkie powiązane zasoby zostaną utracone.

- Windows W systemie Windows usunięcie menedżera kolejek powoduje również usunięcie menedżera kolejek z automatycznej listy startowej (opisanej w sekcji [“Uruchamianie menedżera kolejek”](#) na stronie 126). Po zakończeniu działania komendy wyświetlany jest komunikat IBM MQ queue manager ending ; użytkownik nie jest poinformowany o tym, że menedżer kolejek został usunięty.
- Usunięcie menedżera kolejek klastra nie powoduje usunięcia go z klastra. Więcej informacji na ten temat można znaleźć w uwagach dotyczących użycia w katalogu [dltmqm](#).

O tym zadaniu

Menedżer kolejek można usunąć za pomocą komendy **dltmqm** . Opis komendy **dltmqm** i jej opcji można znaleźć w sekcji [dltmqm](#). Upewnij się, że tylko zaufani administratorzy mają uprawnienia do używania tej komendy. (Informacje na temat bezpieczeństwa zawiera sekcja [Konfigurowanie zabezpieczeń w systemie AIX, Linux, and Windows](#)).

Windows
Linux
 Alternatywnie w systemach Windows i Linux (platformy x86 i x86-64) można usunąć menedżer kolejek za pomocą konsoli IBM MQ Explorer.

Procedura

- Aby usunąć menedżera kolejek za pomocą komendy **dltmqm** , wykonaj następujące kroki:
 - Zatrzymaj menedżer kolejek.
 - Wydadaj następującą komendę:

```
dltmqm QMB
```

Uwaga: Z instalacji powiązanej z menedżerem kolejek, z którym pracuje użytkownik, należy użyć komendy **dltmqm** . Za pomocą komendy `dspmqr -o installation` można dowiedzieć się, która instalacja menedżera kolejek jest powiązana.

- Windows
Linux
 Aby usunąć menedżera kolejek za pomocą konsoli IBM MQ Explorer, wykonaj następujące kroki:
 - Otwórz produkt IBM MQ Explorer.
 - W widoku Navigator wybierz menedżer kolejek.
 - Jeśli menedżer kolejek nie jest zatrzymany, zatrzymaj go.
Aby zatrzymać menedżer kolejek, kliknij go prawym przyciskiem myszy, a następnie kliknij opcję **Zatrzymaj**.
 - Usuń menedżer kolejek.
Aby usunąć menedżera kolejek, kliknij go prawym przyciskiem myszy, a następnie kliknij opcję **Usuń**.

Wyniki

Menedżer kolejek jest usuwany.

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:

Wyciszanie

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).

Pojęcia pokrewne

[Kanały](#)

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](#).

Odsyłacze pokrewne

[Ograniczenia nazewnictwa dla kolejek](#)

[Ograniczenia nazewnictwa dla innych obiektów](#)

Definiowanie kolejki lokalnej za pomocą komendy DEFINE QLOCAL

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.

O tym zadaniu

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 jest ona tworzona w instalacji systemu.

Procedura

- Aby utworzyć kolejkę lokalną, należy wprowadzić komendę **DEFINE QLOCAL** w sposób przedstawiony w poniższym przykładzie.

W tym przykładzie komenda **DEFINE QLOCAL** definiuje kolejkę o nazwie ORANGE.LOCAL.QUEUE o następujących parametrach:

- Jest on włączony dla operacji pobierania, jest 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 wyzwalacza.
- 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)
```

Uwagi:

1. Z wyjątkiem wartości dla opisu, wszystkie wartości atrybutów wyświetlane w przykładzie są wartościami domyślnymi. Przykłady te zostały uwzględnione w celach ilustracyjnych. 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 za pomocą komendy DISPLAY QUEUE”](#) na stronie 138.
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 za pomocą komendy ALTER QLOCAL lub DEFINE QLOCAL”](#) na stronie 140.

Odsyłacze pokrewne

[DEFINE QLOCAL](#)

Wyświetlanie domyślnych atrybutów obiektów za pomocą komendy DISPLAY QUEUE

Komendy **DISPLAY QUEUE** można użyć do wyświetlenia atrybutów, które zostały pobrane z obiektu domyślnego podczas definiowania obiektu IBM MQ .

O tym zadaniu

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. Można użyć komendy **DISPLAY QUEUE** , aby sprawdzić, jakie są te atrybuty.

Procedura

- Aby wyświetlić domyślne atrybuty obiektu dla kolejki lokalnej, należy użyć następującej komendy:

```
DISPLAY QUEUE (SYSTEM.DEFAULT.LOCAL.QUEUE)
```

Składnia komendy **DISPLAY** 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.

Odsyłacze pokrewne

[WYŚWIETL KOLEJKĘ](#)

[Kolejki DEFINE](#)

Kopiowanie definicji kolejki lokalnej z DEFINE QLOCAL

Definicję kolejki można skopiować za pomocą atrybutu **LIKE** w komendzie **DEFINE QLOCAL**.

O tym zadaniu

Za pomocą komendy **DEFINE** z atrybutem **LIKE** można utworzyć kolejkę o takich samych atrybutach, jak określona kolejka, a nie domyślną kolejką lokalną systemu. Można również użyć tego formularza komendy **DEFINE**, aby skopiować definicję kolejki, ale zastąpić jedną lub więcej zmian w atrybutach oryginału.

Uwagi:

1. Jeśli w komendzie **DEFINE** używany jest atrybut **LIKE**, kopiowane są tylko atrybuty kolejki. Nie kopiujesz komunikatów w kolejce.
2. Jeśli definiujesz kolejkę lokalną, bez określania **LIKE**, jest ona taka sama, jak:

```
DEFINE LIKE(SYSTEM.DEFAULT.LOCAL.QUEUE)
```

Procedura

- Aby utworzyć kolejkę o tych samych atrybutach, co podana kolejka, a nie jako domyślna kolejka lokalna systemu, należy wprowadzić komendę **DEFINE**, jak pokazano w poniższym przykładzie.

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.

W tym przykładzie tworzona jest kolejka o tych samych atrybutach, co kolejka ORANGE.LOCAL.QUEUE, a nie domyślna systemowa kolejka lokalna:

```
DEFINE QLOCAL (MAGENTA.QUEUE) +
LIKE (ORANGE.LOCAL.QUEUE)
```

- Aby skopiować definicję kolejki, ale zastąpić jedną lub większą liczbę zmian w atrybutach oryginału, wprowadź komendę **DEFINE**, jak pokazano w poniższym przykładzie.

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:

```
DEFINE QLOCAL (THIRD.QUEUE) +
LIKE (ORANGE.LOCAL.QUEUE) +
MAXMSGL(1024);
```

Odsyłacze pokrewne

[Kolejki DEFINE](#)

Zmiana atrybutów kolejki lokalnej za pomocą komendy ALTER QLOCAL lub DEFINE QLOCAL

Atrybuty kolejki można zmieniać na dwa sposoby, za pomocą komendy **ALTER QLOCAL** lub komendy **DEFINE QLOCAL** z atrybutem **REPLACE**.

O tym zadaniu

Istnieje możliwość użycia atrybutu **REPLACE** komendy **ALTER** i **DEFINE** w celu zastąpienia istniejącej definicji określoną nową definicją. Różnica między używaniem **ALTER** i **DEFINE** polega na tym, że **ALTER** z **REPLACE** nie zmienia nieokreślonych parametrów, ale **DEFINE** z **REPLACE** ustawia wszystkie parametry.

Procedura

- Aby zmienić atrybuty kolejki, należy użyć komendy **ALTER** lub komendy **DEFINE**, tak jak przedstawiono to w poniższych przykładach.

W tych przykładach maksymalna długość komunikatu w kolejce ORANGE.LOCAL.QUEUE jest zmniejszona 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. Na przykład, jeśli kolejka została wcześniej zablokowana, to jest ona zmieniana tak, aby włączyć, ponieważ włączona jest opcja domyślna, zgodnie z ustawieniem 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.

Odsyłacze pokrewne

[Kolejki ALTER](#)

[ALTER QLOCAL](#)

[Kolejki DEFINE](#)

[DEFINE QLOCAL](#)

Kasowanie kolejki lokalnej z CLEAR QLOCAL

Aby wyczyścić kolejkę lokalną, można użyć komendy **CLEAR QLOCAL**.

Zanim rozpoczniesz

Nie można wyczyścić kolejki, jeśli:

- Istnieją niezatwierdzone komunikaty, które zostały umieszczone w kolejce w punkcie synchronizacji.
- Aplikacja aktualnie ma otwartą kolejkę.

O tym zadaniu

Aby skasować kolejkę lokalną za pomocą komendy **CLEAR QLOCAL** , nazwa kolejki musi być zdefiniowana w lokalnym menedżerze kolejek.

Uwaga: Nie ma podpowiedzi, która pozwala Ci zmienić zdanie; po naciśnięciu klawisza Enter wiadomości zostaną utracone.

Procedura

Aby wyczyścić komunikaty z kolejki lokalnej, należy użyć **CLEAR QLOCAL** , jak pokazano w poniższym przykładzie.

W tym przykładzie wszystkie komunikaty są usuwane z kolejki lokalnej o nazwie MAGENTA.QUEUE:

```
CLEAR QLOCAL (MAGENTA.QUEUE)
```

Odsyłacze pokrewne

[CLEAR QLOCAL](#)

Usuwanie kolejki lokalnej z DELETE QLOCAL

Aby usunąć kolejkę lokalną, można użyć komendy MQSC **DELETE QLOCAL** .

O tym zadaniu

Kolejka nie może zostać usunięta, jeśli zawiera niezatwierdzone komunikaty.

Jeśli kolejka ma jeden lub więcej zatwierdzonych komunikatów i nie ma żadnych niezatwierdzonych komunikatów, może zostać usunięta tylko wtedy, gdy zostanie podana opcja **PURGE** . Następnie usunięcie zostanie usunięte, nawet jeśli w podanej kolejce znajdują się zatwierdzone komunikaty, a komunikaty te są również usuwane.

Określenie **NOPURGE** zamiast **PURGE** zapewnia, że kolejka nie zostanie usunięta, jeśli zawiera jakiegokolwiek zatwierdzone komunikaty.

Procedura

- Aby usunąć kolejkę lokalną, należy użyć komendy **DELETE QLOCAL** , jak pokazano w poniższym przykładzie.

Ten przykład usuwa kolejkę PINK.QUEUE , jeśli w kolejce nie ma zatwierdzonych komunikatów:

```
DELETE QLOCAL (PINK.QUEUE) NOPURGE
```

Ten przykład usuwa kolejkę PINK.QUEUE nawet wtedy, gdy w kolejce znajdują się zatwierdzone komunikaty:

```
DELETE QLOCAL (PINK.QUEUE) PURGE
```

Odsyłacze pokrewne

[USUŃ QLOCAL](#)

Przeglądanie kolejek za pomocą przykładowego programu

Produkt IBM MQ udostępnia przykładową przeglądarkę kolejek, której można używać do przeglądania treści komunikatów w kolejce.


```
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
```

Odsyłacze pokrewne

[Przykładowy program przeglądarki](#)

Włączanie dużych kolejek

IBM MQ obsługuje kolejki większe niż 2 TB.

Windows

W systemach Windows obsługa dużych plików jest dostępna bez dodatkowego włączenia.

Linux

AIX

W systemach AIX and Linux konieczne jest jawne włączenie obsługi dużych plików, zanim możliwe będzie utworzenie zbiorów kolejek o wielu gigabajtach lub terabajtach. 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 o wielu gigabajtach lub terabajtach. 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 [MQ Dokumenty dotyczące wydajności](#) dla raportów dotyczących wydajności dla poszczególnych platform.

V 9.2.0

Z poziomu produktu IBM MQ 9.1.5 można kontrolować wielkość plików kolejek przy użyciu nowego atrybutu w kolejkach lokalnych i modelowych. Więcej informacji zawiera temat [“Modyfikowanie plików kolejek produktu IBM MQ”](#) na stronie 143.

Multi

V 9.2.0

Modyfikowanie plików kolejek produktu IBM MQ

Z poziomu produktu IBM MQ 9.2.0 można sterować wielkością plików kolejek za pomocą atrybutu w kolejkach lokalnych i modelowych. Można wyświetlić bieżącą wielkość pliku kolejki oraz maksymalną wielkość, do której obecnie jest w stanie rozwinąć się (na podstawie wielkości bloku aktualnie używanego w tym pliku), używając dwóch atrybutów statusu kolejki.

Atrybut używany do modyfikowania plików kolejki

Atrybut w kolejkach lokalnych i modelowych jest następujący:

MAXFSIZE

Określa maksymalną wielkość pliku kolejki używanego przez kolejkę (w megabajtach).

Więcej informacji na ten temat można znaleźć w sekcji [MAXFSIZE](#) i [“Zmiana wielkości pliku kolejki produktu IBM MQ”](#) na stronie 144 .

Atrybut PCF dla tego atrybutu ma wartość MQIA_MAX_Q_FILE_SIZE; patrz [Zmiana, kopiowanie i tworzenie kolejki](#).

Istnieją dwa atrybuty statusu kolejki:

CURFSIZE

Wyświetla bieżącą wielkość pliku kolejki (w megabajtach) zaokrągloną w górę do najbliższego megabajta.

Więcej informacji na ten temat zawiera sekcja [CURFSIZE](#).

Atrybut PCF dla tego atrybutu ma wartość MQIA_CUR_Q_FILE_SIZE.

CURMAXFS

Wskazuje bieżącą maksymalną wielkość, do której może rosnąć plik kolejki, zaokrągloną w górę do najbliższego megabajta, z uwagi na bieżącą wielkość bloku w użyciu w kolejce.

Więcej informacji na ten temat zawiera sekcja [CURMAXFS](#).

Atrybut PCF dla tego atrybutu ma wartość MQIA_CUR_MAX_FILE_SIZE.

Więcej informacji na temat tych dwóch atrybutów PCF można znaleźć w sekcji [Inquire Queue \(Kolejka zapytań\)](#) i [Inquire Queue \(response\)](#) (zapytanie o kolejkę).

Atrybuty te można ustawiać i wyświetlać za pomocą komend MQSC, IBM MQ Exploreri administrative REST API.

Uwaga: Wartość parametru MAXFSIZE można wyświetlić tylko w IBM MQ Console. Nie można skonfigurować wartości.

Wielkość bloku i granulacja

Pliki kolejek są podzielone na segmenty nazywane blokami. Aby zwiększyć maksymalną wielkość pliku kolejki, może być konieczna zmiana wielkości bloku lub granulacji kolejki przez menedżer kolejek.

Jeśli nowo zdefiniowana kolejka zostanie utworzona z dużą wartością MAXFSIZE, kolejka zostanie utworzona z odpowiednią wielkością bloku. Jeśli jednak istniejąca kolejka ma zwiększoną wartość MAXFSIZE, na przykład za pomocą komendy ALTER QLOCAL, może być konieczne opróżnienie kolejki w celu ponownego skonfigurowania kolejki przez menedżer kolejek.

Więcej informacji zawiera sekcja [“Obliczanie ilości danych, jaką może przechowywać plik kolejki produktu IBM MQ”](#) na stronie 145.



Ostrzeżenie: Niektóre systemy plików i systemy operacyjne mają ograniczenia dotyczące wielkości całego systemu plików lub wielkości pojedynczego pliku. Należy sprawdzić limity w systemach, w których używane są przedsiębiorstwa.

Odsyłacze pokrewne

[INSTRUKCJE ALTER QUEUES](#)

[WYŚWIETL KOLEJKĘ](#)

[WYŚWIETL STATUS QSTATUS](#)

Zmiana wielkości pliku kolejki produktu IBM MQ

Można zwiększyć lub zmniejszyć maksymalną wielkość pliku kolejki.

Zanim rozpoczniesz

Przed ustawieniem nowej wielkości dla pliku kolejki należy użyć komendy [DISPLAY QLOCAL](#), aby wyświetlić wielkość pliku kolejki, który ma zostać zmieniony. Na przykład wprowadź następującą komendę:

```
DISPLAY QLOCAL(SYSTEM.DEFAULT.LOCAL.QUEUE) MAXFSIZE
```

Otrzymujesz następujące dane wyjściowe:

```
AMQ8409I: Display queue details
QUEUE(SYSTEM.DEFAULT.LOCAL.QUEUE)      TYPE(QLOCAL)
MAXFSIZE(DEFAULT)
```


co wskazuje, że maksymalna wielkość pliku kolejki jest wartością domyślną wynoszącą 2 088,960 MB.

O tym zadaniu

Poniższe procedury pokazują, w jaki sposób:

- Zmniejsz maksymalną wielkość, do której może rosnać plik kolejki.
- Zwiększ maksymalną wielkość, do której może rosnać plik kolejki.



Ostrzeżenie: Należy zachować ostrożność przy zwiększaniu wielkości plików kolejek bez uwzględniania sposobu zapisywania aplikacji i ewentualnego wpływu na wydajność. Dostęp do komunikatów losowo w bardzo dużym pliku kolejki może być bardzo powolny.

Jeśli użytkownik rozważa zwiększenie maksymalnej wielkości pliku kolejki poza domyślną wartością, należy zachować ostrożność podczas używania selektorów komunikatów, takich jak identyfikatory korelacji i łańcuchy selektora IBM MQ classes for JMS . Większe pliki kolejki są lepiej dostosowane do pierwszego-w pierwszym-wyjściu dostępu do kolejki.

Bardzo duże ilości danych w poszczególnych plikach kolejek powinny być wykonywane tylko w menedżerach kolejek skonfigurowanych na potrzeby rejestrowania cyklicznego lub w przypadku, gdy w danej kolejce nie włączono obrazowania nośników.

Nie należy ograniczać wielkości kolejek SYSTEM, ponieważ może to mieć wpływ na działanie menedżera kolejek.

Procedura

1. Zmniejsz maksymalną wielkość pliku kolejki

- a) Wydad następującą komendę, aby utworzyć plik lokalny o nazwie SMALLQUEUEo wielkości 500 gigabajtów:

```
DEFINE QLOCAL(SMALLQUEUE) MAXFSIZE(512000)
2 : DEFINE QLOCAL(SMALLQUEUE) MAXFSIZE(512000)
AMQ8006I: IBM MQ queue created
```

i zostanie wyświetlony komunikat: AMQ8006I:

Uwaga: W przypadku skonfigurowania kolejki o wartości mniejszej niż ilość danych już znajdujących się w pliku, nowe komunikaty nie mogą być umieszczane w kolejce.

Jeśli aplikacja podejmie próbę umieszczenia komunikatu w pliku kolejki, w którym nie ma wystarczającej ilości miejsca, aplikacja otrzyma kod powrotu MQRC_Q_SPACE_NOT_AVAILABLE. Gdy wystarczająca liczba komunikatów jest niszcących odczytywanych z kolejki, aplikacje mogą rozpocząć umieszczanie nowych komunikatów w kolejce.

2. Zwiększ maksymalną wielkość pliku kolejki.

- a) Wydad następującą komendę, aby utworzyć plik lokalny o nazwie LARGEQUEUE, o wielkości 5 terabajtów:

```
DEFINE QLOCAL(LARGEQUEUE) MAXFSIZE(5242880)
3 : DEFINE QLOCAL(LARGEQUEUE) MAXFSIZE(5242880)
AMQ8006I: IBM MQ queue created
```

Multi V 9.2.0 **Obliczanie ilości danych, jaką może przechowywać plik kolejki produktu IBM MQ**

Ilość danych, które mogą być przechowywane w kolejce, jest ograniczona przez wielkość poszczególnych bloków, do których jest podzielona kolejka.

Wielkość bloku i granulacja

Domyślna wielkość bloku to 512 bajtów. Aby można było obsługiwać pliki kolejki większe niż dwa terabajty, menedżer kolejek musi zwiększyć wielkość bloku.

Wielkość bloku jest obliczana automatycznie podczas konfigurowania parametru `MAXFSIZE` dla kolejki, ale poprawiona wielkość bloku nie może zostać zastosowana do kolejki, jeśli kolejka ma już komunikaty w tej kolejce. Gdy kolejka jest pusta, menedżer kolejek automatycznie modyfikuje wielkość bloku w celu obsługi skonfigurowanej wartości `MAXFSIZE`.

Komenda `DISPLAY QSTATUS` zawiera nowy atrybut `CURMAXFS`, który pozwala na potwierdzenie, że kolejka została zmodyfikowana w celu użycia nowej wielkości bloku.

W poniższym przykładzie wartość `CURMAXFS` o wartości 4177920 potwierdza, że plik kolejki jest obecnie w stanie wyrosnąć na około cztery terabajty wielkości. Jeśli wartość parametru `MAXFSIZE` skonfigurowana w kolejce jest większa niż wartość `CURMAXFS`, to menedżer kolejek nadal oczekuje na opróżnienie kolejki, zanim ponownie skonfiguruje wielkość bloku w pliku kolejki.

```
DISPLAY QSTATUS(LARGEQUEUE) CURMAXFS
  2 : DISPLAY QSTATUS(LARGEQUEUE) CURMAXFS
AMQ8450I: Display queue status details
        QUEUE(LARGEQUEUE)                TYPE(Queue)
        CURMAXFS(4177920)                CURDEPTH(100000)
```

Sprawdzanie wielkości pliku kolejki

Bieżącą wielkość pliku kolejki na dysku można wyświetlić w megabajtach, używając atrybutu `CURFSIZE` w komendzie `DISPLAY QSTATUS`. Może to być przydatne na platformach, takich jak IBM MQ Appliance, gdzie nie ma możliwości bezpośredniego dostępu do systemu plików.

```
DISPLAY QSTATUS(SMALLQUEUE) CURFSIZE
  1 : DISPLAY QSTATUS(SMALLQUEUE) CURFSIZE
AMQ8450I: Display queue status details
        QUEUE(SMALLQUEUE)                TYPE(Queue)
        CURDEPTH(4024)                   CURFSIZE(10)
```

Uwaga: Gdy kolejka ma usunięte komunikaty, atrybut `CURFSIZE` może nie zmniejszać się od razu.

Zwykle nieużywane miejsce w pliku kolejki jest zwalniane tylko w następujących okolicznościach:

- Gdy żadne aplikacje nie mają otwartych kolejek
- Po 1000 operacji zapisu w dzienniku menedżera kolejek lub
- Przy zamykaniu menedżera kolejek

Odsyłacze pokrewne

[INSTRUKCJE ALTER QUEUES](#)

[WYŚWIETL STATUS QSTATUS](#)

Praca z kolejkami zdalnymi

Kolejka zdalna jest lokalną definicją kolejki zdalnej. Oznacza to, że definicja w lokalnym menedżerze kolejek 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.

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 saturn.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, której 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.

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 ządania* .

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 ządania 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](#).

Praca z kolejkami aliasowymi

Kolejkę aliasową można zdefiniować tak, aby odwoływał się pośrednio do innej kolejki lub tematu.



Ostrzeżenie: Listy dystrybucyjne nie obsługują korzystania z kolejek aliasowych, które wskazują na obiekty tematów. Jeśli kolejka aliasowa wskazuje na obiekt tematu na liście dystrybucyjnej, program IBM MQ zwraca MQRC_ALIAS_BASE_Q_TYPE_ERROR.

Kolejka, do której odwołuje się kolejka aliasowa, może być jedną z następujących kolejek:

- Kolejka lokalna (patrz sekcja [“Definiowanie kolejki lokalnej za pomocą komendy DEFINE QLOCAL” na stronie 137](#)).
- Lokalna definicja kolejki zdalnej (patrz sekcja [“Praca z kolejkami zdalnymi” na stronie 146](#)).
- Temat.

Kolejka aliasowa nie jest kolejką rzeczywistą, ale definicją, która jest tłumaczona na kolejkę rzeczywistą (lub docelową) w czasie wykonywania. Definicja kolejki aliasowej określa kolejkę docelową. Gdy aplikacja wysyła wywołanie MQOPEN do kolejki aliasowej, menedżer kolejek tłumaczy alias na nazwę kolejki docelowej.

Kolejka aliasowa nie może zostać przetłumaczona na inną lokalnie zdefiniowaną kolejkę aliasową. Kolejka aliasowa może jednak zostać przetłumaczona na kolejki aliasowe zdefiniowane w innym miejscu w klastrach, do których należy menedżer kolejek lokalnych. Więcej informacji na ten temat zawiera sekcja [Rozstrzygnięcie nazw](#).

Kolejki aliasowe są przydatne w następujących sytuacjach:

- Nadawanie różnym aplikacjom różnych poziomów uprawnień dostępu do kolejki docelowej.
- Zezwalanie różnym aplikacjom na pracę z tą samą kolejką na różne sposoby. (Być może chcesz przypisać różne domyślne priorytety lub różne domyślne wartości trwałości).
- Upraszcza konserwację, migrację i równoważenie obciążenia. (Być może chcesz zmienić nazwę kolejki docelowej bez konieczności zmiany aplikacji, która nadal używa aliasu).

Na przykład założmy, że została opracowana aplikacja do umieszczania komunikatów w kolejce o nazwie MY.ALIAS.QUEUE. Określa nazwę tej kolejki podczas wykonywania żądania MQOPEN i pośrednio, jeśli umieszcza komunikat w tej kolejce. Aplikacja nie wie, że kolejka jest kolejką aliasową. Dla każdego wywołania MQI używającego tego aliasu menedżer kolejek tłumaczy rzeczywistą nazwę kolejki, która może być kolejką lokalną lub zdalną zdefiniowaną w tym menedżerze kolejek.

Zmieniając wartość atrybutu TARGET, 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 parametr 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.

W przypadku zmiany definicji aliasu można 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 pojedyncza kolejka (kolejka docelowa) miała różne atrybuty dla różnych aplikacji. W tym celu należy zdefiniować dwa aliasy, po jednym dla każdej aplikacji. Założmy, że istnieją dwie aplikacje:

- Aplikacja ALPHA może umieszczać komunikaty w kolorze YELLOW.QUEUE, ale nie jest dozwolone pobieranie z niego komunikatów.
- Aplikacja BETA może pobrać komunikaty z YELLOW.QUEUE, ale nie jest dozwolone umieszczanie w nim komunikatów.

Poniższa komenda definiuje alias, który jest włączony i wyłączony dla aplikacji ALPHA:

```
DEFINE QALIAS (ALPHAS.ALIAS.QUEUE) +  
TARGET (YELLOW.QUEUE) +  
PUT (ENABLED) +  
GET (DISABLED)
```

Poniższa komenda definiuje alias, który jest wyłączany i włączany dla BETA aplikacji:

```
DEFINE QALIAS (BETAS.ALIAS.QUEUE) +  
TARGET (YELLOW.QUEUE) +  
PUT (DISABLED) +  
GET (ENABLED)
```

ALPHA używa nazwy kolejki ALPHAS.ALIAS.QUEUE w wywołaniach MQI; BETA używa nazwy kolejki BETAS.ALIAS.QUEUE. Obie mają dostęp do tej samej kolejki, ale na różne sposoby.

Podczas definiowania aliasów kolejek można używać atrybutów LIKE i REPLACE w taki sam sposób, jak w przypadku kolejek lokalnych.

Używanie innych komend z kolejkami aliasowymi

Aby wyświetlić lub zmienić atrybuty kolejki aliasowej lub usunąć obiekt kolejki aliasowej, można użyć odpowiednich komend MQSC. Na przykład:

Użyj komendy **DISPLAY QALIAS**, aby wyświetlić atrybuty kolejki aliasowej:

```
DISPLAY QALIAS (ALPHAS.ALIAS.QUEUE)
```

Użyj komendy **ALTER QALIAS**, aby zmienić nazwę kolejki podstawowej, na którą jest tłumaczony alias, gdzie opcja **force** wymusza zmianę, nawet jeśli kolejka jest otwarta:

```
ALTER QALIAS (ALPHAS.ALIAS.QUEUE) TARGET(ORANGE.LOCAL.QUEUE) FORCE
```

Użyj komendy **DELETE QALIAS**, aby usunąć ten alias kolejki:

```
DELETE QALIAS (ALPHAS.ALIAS.QUEUE)
```

Nie można usunąć kolejki aliasowej, jeśli aplikacja ma obecnie otwartą kolejkę.

Odsyłacze pokrewne

[ZMIENŃ QALIAS](#)

[ZDEFINIUJ QALIAS](#)

[USUŃ QALIAS](#)

[Lista dystrybucyjna](#)

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 `SYSYSTEM.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)
```

Odsyłacze pokrewne

[ALTER QMODEL](#)

[DEFINE QMODEL](#)

[USUŃ QMODEL](#)

[WYŚWIETL KOLEJKĘ](#)

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 **USEDLQ** 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 **USEDLQ** 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 [“Przetwarzanie komunikatów w kolejce niedostarczonych komunikatów programu IBM MQ” na stronie 152.](#)

Pojęcia pokrewne

[Kolejki niedostarczane](#)

Zadania pokrewne

[Rozwiązywanie problemów z niedostarczonym](#)

Odsyłacze pokrewne

[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 (**MCA**s) 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.

Pojęcia pokrewne

[Kolejki niedostarczane](#)

Zadania pokrewne

[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 152.

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/QTXTSRC)
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 umieścić komunikaty w kolejkach z uprawnieniami identyfikatora użytkownika w kontekście komunikatu.

Pojęcia pokrewne

Kolejki niedostarczane

Zadania pokrewne

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* | ' ' (wartość domyślna))

Nazwa kolejki DLQ, która ma być przetwarzana:

1. Każda wartość UDLMSGQ (lub *DFT) podana jako parametr komendy **STRMQMDLQ** przesłania dowolną wartość INPUTQ w tabeli reguł.
2. Jeśli jako parametr komendy **STRMQMDLQ** zostanie podana pusta wartość UDLMSGQ, zostanie użyta wartość INPUTQ w tabeli reguł.
3. Jeśli jako parametr komendy **STRMQMDLQ** zostanie podana pusta wartość UDLMSGQ, a w tabeli reguł zostanie podana pusta wartość INPUTQ, zostanie użyta systemowa domyślna kolejka niedostarczonych komunikatów.

INPUTQM (*QueueManagerNazwa* | ' ' (wartość domyślna))

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 (domyślny obszar roboczy))

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* (domyślnie) | *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 (*ApplIdentityData* | * (wartość domyślna))

Wartość *ApplIdentityData* komunikatu w DLQ, określona w deskrytorze komunikatu, MQMD.

APPLNAME (*PutApplNazwa* | * (wartość domyślna))

Nazwa aplikacji, która wywołała wywołanie MQPUT lub MQPUT1 , zgodnie z wartością określoną w polu *PutApplNazwa* deskrytora komunikatu, MQMD, komunikatu w DLQ.

APPLTYPE (*PutApplTyp* | * (wartość domyślna))

Wartość *PutApplType* określona w deskrytorze komunikatu, MQMD, komunikatu w DLQ.

DESTQ (*QueueName* | * (wartość domyślna))

Nazwa kolejki komunikatów, dla której jest przeznaczony komunikat.

DESTQM (*QueueManagerNazwa* | * (wartość domyślna))

Nazwa menedżera kolejek dla kolejki komunikatów, dla której jest przeznaczony komunikat.

FEEDBACK (*Opinia* | * (wartość domyślna))

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* | * (wartość domyślna))

Nazwa, która jest używana przez nadawcę wiadomości do opisu formatu danych komunikatu.

MSGTYPE (*MsgType* | * (wartość domyślna))

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 domyślna))

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* | * (wartość domyślna))

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* | * (wartość domyślna))

Nazwa kolejki odpowiedzi podana w deskrytorze komunikatu, MQMD, komunikatu w DLQ.

REPLYQM (*QueueManagerNazwa* | * (wartość domyślna))

Nazwa menedżera kolejek dla kolejki odpowiedzi określonej w słowie kluczowym REPLYQ.

USERID (*UserIdentifier* | * (wartość domyślna))

Identyfikator użytkownika, który zainicjował komunikat w DLQ, zgodnie z określonym w deskrytorze 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.

ODRZUC

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 | ' ' (wartość domyślna))

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 (YES (domyślnie) |NO)

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 (wartość domyślna) | 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 (domyślnie))

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. Znaku zapytania (?) można użyć w miejsce dowolnego pojedynczego znaku, z wyjątkiem odstępu 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 utrzymuje rekord tych komunikatów 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 REALLY . 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](#).

Pojęcia pokrewne

[Kolejki niedostarczane](#)

Zadania pokrewne

[Rozwiązywanie problemów z niedostarczonym](#)

Przykładowa procedura obsługi DLQ `amqsd1q`

Oprócz programu obsługi niedostarczonych komunikatów wywoływanego za pomocą komendy `runmqdlq`, IBM MQ udostępnia źródło przykładowej procedury obsługi DLQ `amqsd1q` z funkcją podobną do funkcji udostępnianej przez `runmqdlq`.

Program **amqsd1q** można dostosować, aby udostępnić program obsługi DLQ spełniający wymagania użytkownika. Na przykład można zdecydować, że program obsługi DLQ może przetwarzać komunikaty bez nagłówek niewystających wiadomości. (Zarówno domyślna procedura obsługi DLQ, jak i przykład, **amqsd1q**, przetwarzają tylko te komunikaty w kolejce DLQ, które rozpoczynają się od nagłówka niedostarczonego komunikatu MQDLH. Komunikaty, które nie rozpoczynają się od MQDLH, są identyfikowane jako błędne i pozostają w DLQ w nieskończoność).

`MQ_INSTALLATION_PATH` reprezentuje katalog wysokiego poziomu, w którym jest zainstalowany produkt IBM MQ.

W systemie IBM MQ for Windows źródło pliku `amqsd1q` jest dostarczane w katalogu:

```
MQ_INSTALLATION_PATH\tools\c\samples\d1q
```

i 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
```

i skompilowana wersja jest dostarczana w katalogu:

```
MQ_INSTALLATION_PATH/samp/bin
```

V9.2.3 Dołączona jest zbudowana wersja programu przykładowego o nazwie **amqsd1qc**. Tej opcji można użyć do nawiązania połączenia ze zdalnym menedżerem kolejek w trybie klienta. Aby skorzystać z opcji **amqsd1qc**, należy ustawić jedną ze zmiennych środowiskowych **MQSERVER**, **MQCHLLIB** lub **MQCHLTAB** w celu określenia sposobu nawiązywania połączenia z menedżerem kolejek. Na przykład:

```
export MQSERVER="SYSTEM.DEF.SVRCONN/TCP/myappliance.co.uk(1414)"
```

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.

Pojęcia pokrewne

Kolejki niedostarczane

Zadania pokrewne

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| ' ' (wartość domyślna))

Nazwa kolejki DLQ, która ma być przetwarzana:

1. Każda wartość INPUTQ podana jako parametr komendy `runmqdlq` nadpisuje dowolną wartość INPUTQ w tabeli reguł.
2. Jeśli wartość INPUTQ nie zostanie określona jako parametr dla komendy `runmqdlq`, 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| ' ' (wartość domyślna))

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 (domyślny obszar roboczy))

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 (domyślnie) |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.

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, 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 domyślna))

Wartość *ApplIdentityData* podana w deskrytorze komunikatu, MQMD, komunikatu w DLQ.

APPLNAME (*PutApplNazwa* | * (wartość domyślna))

Nazwa aplikacji, która wywołała wywołanie MQPUT lub MQPUT1, zgodnie z wartością określoną w polu *PutApplName* deskryptora komunikatu MQMD komunikatu w DLQ.

APPLTYPE (*PutApplTyp* | * (wartość domyślna))

Wartość *PutApplType* określona w deskrytorze komunikatu MQMD komunikatu w DLQ.

DESTQ (*QueueName* | * (wartość domyślna))

Nazwa kolejki komunikatów, dla której jest przeznaczony komunikat.

DESTQM (*QueueManagerNazwa* | * (wartość domyślna))

Nazwa menedżera kolejek dla kolejki komunikatów, dla której jest przeznaczony komunikat.

FEEDBACK (*Opinia* | * (wartość domyślna))

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* | * (wartość domyślna))

Nazwa, która jest używana przez nadawcę wiadomości do opisanie formatu danych komunikatu.

MSGTYPE (*MsgType* | * (wartość domyślna))

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 domyślna))

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* | * (wartość domyślna))

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* | * (wartość domyślna))

Nazwa kolejki odpowiedzi określonej w deskrytorze komunikatu, MQMD, komunikatu w DLQ.

REPLYQM (*QueueManagerNazwa* | * (wartość domyślna))

Nazwa menedżera kolejek dla kolejki odpowiedzi, zgodnie z określonym w deskrytorze komunikatu, MQMD komunikatu w DLQ.

USERID (*UserIdentifier* | * (wartość domyślna))

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 | ' ' (wartość domyślna))

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 (YES (domyślnie) | NO)

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 (wartość domyślna) | 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 (domyślnie))

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 wzorzec 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 wchodzi 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.

Pojęcia pokrewne

Kolejki niedostarczane

Zadania pokrewne

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 opróżnia 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)
```

Pojęcia pokrewne

[Kolejki niedostarczane](#)

Zadania pokrewne

[Rozwiązywanie problemów z niedostarczonym](#)

Odsyłacze pokrewne

[runmqdlq \(uruchomienie procedury obsługi kolejki niedostarczonych komunikatów\)](#)

Praca z tematami administracyjnymi

Aby zarządzać tematami administracyjnymi, należy użyć komend MQSC.

Szczegółowe informacje na temat tych komend zawiera sekcja [Komendy MQSC](#).

Pojęcia pokrewne

Administracyjne obiekty tematów

“Definiowanie tematu administracyjnego” na stronie 172

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 172

Aby wyświetlić obiekt tematu administracyjnego, należy użyć komendy MQSC **DISPLAY TOPIC** .

“Zmiana atrybutów tematu administracyjnego” na stronie 173

Atrybuty tematu można zmieniać na dwa sposoby, korzystając z komendy **ALTER TOPIC** lub **DEFINE TOPIC** z atrybutem **REPLACE** .

“Kopiowanie definicji tematu administracyjnego” na stronie 174

Definicję tematu można skopiować za pomocą atrybutu LIKE w komendzie **DEFINE** .

“Usuwanie definicji tematu administracyjnego” na stronie 174

Aby usunąć temat administracyjny, można użyć komendy MQSC **DELETE TOPIC** .

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) +  
NPMGDLV (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 172.
- 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 173.

Odsyłacze pokrewne

ZDEFINIUIJ 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) DEFPRTY(ASPARENT)  
NPMSGDLV(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)
```

Odsyłacze 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.

Odsyłacze 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 kopiowania 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)
```

Odsyłacze 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](#).

Odsyłacze pokrewne

[Usuń temat](#)

Praca z subskrypcjami

Do zarządzania subskrypcjami można używać komend MQSC.

Subskrypcje mogą być jednym z trzech typów zdefiniowanych w atrybucie **SUBTYPE** :

admin

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 175](#)

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 176](#)

Za pomocą komendy **DISPLAY SUB** można wyświetlić skonfigurowane atrybuty dowolnej subskrypcji znanej menedżerowi kolejek.

[“Zmiana atrybutów subskrypcji lokalnej” na stronie 177](#)

Atrybuty subskrypcji można zmienić na dwa sposoby, korzystając z komendy **ALTER SUB** lub **DEFINE SUB** z atrybutem **REPLACE** .

[“Kopiowanie definicji subskrypcji lokalnej” na stronie 177](#)

Definicję subskrypcji można skopiować za pomocą atrybutu **LIKE** w komendzie **DEFINE** .

[“Usuwanie subskrypcji lokalnej” na stronie 178](#)

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 176.
- 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 177.
- Jeśli kolejka SUBQ nie istnieje, wykonanie tej komendy nie powiedzie się.

Odsyłacze 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)
DURABLE(YES)
TOPICSTR(ORANGE)
```

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)
DURABLE(YES)
TOPICSTR(ORANGE)
```

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)
```

Odsyłacze 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.

Odsyłacze 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 THIRD.SUBi określić, że identyfikatorem correlID dostarczoną publikacji jest THIRD, a nie publikatorów correlID. Na przykład:

```
DEFINE SUB(THIRD.SUB) +  
  LIKE(BLUE) +  
  DESTCURL(ORANGE)
```

Odsyłacze 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)
```

Odsyłacze 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 176.
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 za pomocą przykładowego programu”](#) na stronie 141.

Odsyłacze 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.

Odsyłacze 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 zatrzymywany.

TYLKO startonly

Usługa ma zostać uruchomiona w tym samym czasie co menedżer kolejek, ale nie jest wymagana do zatrzymania, gdy menedżer kolejek jest zatrzymywany.

Pojęcia pokrewne

“Zarządzanie usługami” na stronie 180

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**.

Odsyłacze 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 181](#)

Po uruchomieniu usługi środowisko, w którym uruchamiany jest proces usługi, jest dziedziczone ze środowiska menedżera kolejek. Istnieje możliwość zdefiniowania dodatkowych zmiennych środowiskowych, które mają zostać ustawione w środowisku procesu usługi, przez dodanie zmiennych, które mają zostać zdefiniowane, do jednego z plików przestaniania środowiska `service.env`.

[Zatrzymaj usługę](#)

[Uruchom usługę](#)

Dodatkowe zmienne środowiskowe



Po uruchomieniu usługi środowisko, w którym uruchamiany jest proces usługi, jest dziedziczone ze środowiska menedżera kolejek. Istnieje możliwość zdefiniowania dodatkowych zmiennych środowiskowych, które mają zostać ustawione w środowisku procesu usługi, przez dodanie zmiennych, które mają zostać zdefiniowane, do jednego z plików przestaniania środowiska `service.env`.

Pliki, do których można dodawać zmienne środowiskowe

Istnieją dwa możliwe pliki, do których można dodać zmienne środowiskowe:



Plik `service.env` o zasięgu komputera

Ten plik znajduje się w katalogu:

-  `/var/mqm` w systemach AIX 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 przestaniania środowiska dla menedżera kolejek o nazwie `QMNAME` jest następujące:

- –  W systemach AIX and Linux : `/var/mqm/qmgrs/QMNAME/service.env`
-  W systemach Windows :
`C:\ProgramData\IBM\MQ\qmgrs\QMNAME\service.env`

Oba pliki są przetwarzane, jeśli są dostępne, z definicjami w pliku zasięgu menedżera kolejek, które mają pierwszeństwo przed definicjami w pliku zasięgu komputera.

Zmienne środowiskowe, które można określić w pliku `service.env`.

W pliku `service.env` można podać dowolną zmienną środowiskową. Jeśli na przykład usługa systemu IBM MQ uruchamia kilka komend, przydatne może być ustawienie zmiennej użytkownika `PATH` w pliku `service.env`. Wartości ustawione dla zmiennej nie mogą być zmiennymi środowiskowymi; na przykład `CLASSPATH= %CLASSPATH%` jest niepoprawna. Podobnie w przypadku ścieżki Linux `PATH=$PATH :/opt/mqm/bin` dałoby nieoczekiwane wyniki.

Zmienna `CLASSPATH` musi być zapisana wielkimi literami, a instrukcja ścieżki klasy może zawierać tylko literały. Niektóre usługi (na przykład Telemetry) ustawiają własną ścieżkę klasy. Zmienna `CLASSPATH` zdefiniowana w zmiennej `service.env` zostanie do niej dodana.

Format zmiennych zdefiniowanych w pliku `service.env` jest listą par nazw i wartości zmiennych. Każda zmienna musi być zdefiniowana w nowym wierszu, a każda zmienna jest przyjmowana tak, jak jest jawnie zdefiniowana, łącznie z białymi spacjami.

Przykład zastosowania właściwości `service.env`

```
#####  
##  
## <N_OCO_COPYRIGHT> ##  
## Licensed Materials - Property of IBM ##  
## ##  
## 63H9336 ##  
## (C) Copyright IBM Corporation 2005, 2024. ##  
## ##  
## <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 182](#)

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 :

- Linux AIX W systemach AIX and Linux położeniem katalogu danych produktu IBM MQ jest `/var/mqm/`
- Windows 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**, **STDOOUT** lub **STDERR** . Przykłady tego programu znajdują się w sekcji [“Przykłady korzystania z obiektów usług”](#) na stronie 183.

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/runmqtm') +
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 182 .

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 inicjujący 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 odbiera zmiany wprowadzone w produkcie [“4” na stronie 183](#).

Uwaga: Komenda MQSC **STOP SERVICE** może być używana tylko wtedy, gdy w definicji usługi jest określony argument **STOPCMD** .

Odsyłacze 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 AIX lub Linux do zapisu w dzienniku systemowym.
+QMNAME+ jest znacznikiem reprezentującym nazwę menedżera kolejek.

Odsyłacze 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.

Odsyłacze 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.

Odsyłacze 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.

Odsyłacze 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 inicjujący 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.

Używanie programu narzędziowego dmpmqmsg między dwoma systemami

Program narzędziowy **dmpmqmsg** (uprzednio zwany *qload*) umożliwia kopiowanie lub przenoszenie zawartości kolejki lub jej komunikatów do pliku.

Przegląd

Plik utworzony za pomocą opcji **dmpmqmsg** może zostać zapisany w razie potrzeby i użyty w późniejszym czasie w celu ponownego załadowania komunikatów do kolejki.

Ważne:

1. Plik ma konkretny format zrozumiały dla programu narzędziowego. Jednak plik jest czytelny dla człowieka, dzięki czemu można go zaktualizować w edytorze przed przetądowaniem. Jeśli plik jest edytowany, nie można zmieniać jego formatu.
2. W systemie IBM MQ 9.1 program narzędziowy **dmpmqmsg** jest dostarczany z zestawem plików środowiska wykonawczego dla systemu AIX, Linux, and Windows, dlatego jest on dostępny zarówno na serwerze, jak i na kliencie z systemem IBM MQ. W wersjach wcześniejszych niż IBM MQ 9.1 program narzędziowy był dostarczany tylko z pakietem serwera.

Możliwe są następujące zastosowania:

- Zapisywanie komunikatów znajdujących się w kolejce do pliku. Możliwe, że na potrzeby archiwizacji, a następnie przetądowania z powrotem do kolejki.
- Przetądowywanie kolejki z komunikatami, które zostały wcześniej zapisane w pliku.
- Usuwanie starych komunikatów z kolejki.
- Powtarzanie komunikatów testowych z przechowywanego miejsca, nawet zachowywanie poprawnego czasu między komunikatami, jeśli jest to wymagane.



Ostrzeżenie: Pakiet SupportPac MO03 użył parametru **-1** do określenia powiązania lokalnego lub klienta. **-1** został zastąpiony przez parametr **-c**.

-P jest teraz używany dla informacji o stronie kodowej zamiast **-c**.

Więcej informacji na temat komendy i dostępnych parametrów zawiera sekcja [dmpmqmsg](#).

Przykład użycia programu narzędziowego dmpmqmsg w systemie Linux na komputerze z systemem Windows

Na komputerze z systemem Linux znajduje się menedżer kolejek z komunikatami w kolejce (*Q1*), który ma zostać przeniesiony do innej kolejki (*Q2*) w tym samym menedżerze kolejek. Program narzędziowy **dmpmqmsg** ma zostać zainicjowany na komputerze z systemem Windows.

Kolejka (*Q1*) zawiera cztery komunikaty, które zostały dodane za pomocą przykładowej aplikacji **amqspu**t (menedżer kolejek lokalnych) lub **amqspu**tc (menedżer kolejek zdalnych).

Na komputerze z systemem 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 tak, aby wskazywała na menedżer kolejek w programie Linux. Na przykład:

```
set MQSERVER=SYSTEM.DEF.SVRCONN/TCP/veracruz.x.com(1414)
```

gdzie *veracruz* jest nazwą komputera.

Uruchom program narzędziowy **dmpmqmsg**, aby odczytać dane z kolejki *Q1* i zapisać dane wyjściowe w pliku `c:\temp\mqqlload.txt`.

Nawiąż połączenie jako klient zdalny z menedżerem kolejek *QM_VER*, działającym na hoście Linux i porcie określonym przez parametr *MQSERVER*. Połączenie jest nawiązywane jako klient zdalny przy użyciu atrybutu: `-c`.

```
dmpmqmsg -m QM_VER -i Q1 -f c:\temp\mqqlload.txt -c
Read      - Files:    0  Messages:    4  Bytes:          22
Written   - Files:    1  Messages:    4  Bytes:          22
```

Plik wyjściowy `c:\temp\mqqlload.txt` zawiera tekst w formacie zrozumiałym dla programu narzędziowego **dmpmqmsg**.

Na komputerze z systemem Windows wprowadź komendę **dmpmqmsg** (używając opcji `-o` zamiast opcji `-i`), aby załadować kolejkę (*Q2*) na komputerze z systemem Linux z pliku na komputerze z systemem Windows :

```
dmpmqmsg -m QM_VER -o Q2 -f c:\temp\mqqlload.txt -c
Read      - Files:    1  Messages:    4  Bytes:          22
Written   - Files:    0  Messages:    4  Bytes:          22
```

Na komputerze z systemem Linux należy zauważyć, że w kolejce znajdują się teraz cztery wiadomości, 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 systemem 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 zawiera sekcja [dmpmqmsg](#).

Pojęcia pokrewne

“Przykłady użycia programu narzędziowego **dmpmqmsg**” na stronie 189

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 190, 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 190 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.

Kopiuje 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
```

Przeniesienie komunikaty z jednej kolejki do innej

Odmiana w systemie “Ładowanie kolejki z pliku” na stronie 190. 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 190, 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](#).

Praca ze zdalnymi obiektami IBM MQ

Obiekty IBM MQ można administrować w zdalnych menedżerach kolejek za pomocą komend MQSC, komend PCF lub administracyjnej REST API. Zanim będzie można użyć dowolnej z tych metod, należy zdefiniować kolejki transmisji i kanały między lokalnym menedżerem kolejek a zdalnym menedżerem kolejek, aby komendy mogły być wysyłane do zdalnego menedżera kolejek, a odpowiedzi otrzymane przez lokalny menedżer kolejek. Alternatywnie można skonfigurować klaster menedżera kolejek, a następnie użyć tych samych metod zdalnego administrowania.

O tym zadaniu

Aby przygotować menedżery kolejek do zdalnego administrowania, należy skonfigurować następujące obiekty w lokalnym menedżerze kolejek:

- Obiekt nasłuchiwania.
- Kolejka transmisji, która ma nazwę zdalnego menedżera kolejek.
- Kanał nadawczy, który zawiera szczegóły połączenia ze zdalnym menedżerem kolejek.
- Kanał odbiorczy, który ma taką samą nazwę jak kanał nadawczy w zdalnym menedżerze kolejek.

Należy również skonfigurować następujące obiekty w zdalnym menedżerze kolejek:

- Obiekt nastuchiwania.
- Kolejka transmisji, która ma nazwę lokalnego menedżera kolejek.
- Kanał nadawczy, który zawiera szczegóły połączenia z lokalnym menedżerem kolejek.
- Kanał odbiorczy, który ma taką samą nazwę jak kanał nadawczy w lokalnym menedżerze kolejek.

Więcej informacji na temat konfigurowania tych obiektów zawiera sekcja [“Konfigurowanie menedżerów kolejek na potrzeby zdalnego administrowania”](#) na stronie 192.

Alternatywnie można skonfigurować klastr menedżera kolejek. *Klastr* 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 klastr 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 błędu w definicjach.

Aby skonfigurować klastr, 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.

Więcej informacji na temat konfigurowania klastra zawiera sekcja [Konfigurowanie klastra menedżera kolejek](#).

Procedura

- Informacje na temat administrowania zdalnymi obiektami produktu IBM MQ można znaleźć w następujących podtematach:
 - [“Konfigurowanie menedżerów kolejek na potrzeby zdalnego administrowania”](#) na stronie 192
 - [“Zarządzanie serwerem komend w celu zdalnego administrowania”](#) na stronie 196
 - [“Wydawanie komend MQSC w zdalnym menedżerze kolejek”](#) na stronie 197
 - [“Konwersja danych między kodowanymi zestawami znaków”](#) na stronie 199

Konfigurowanie menedżerów kolejek na potrzeby zdalnego administrowania

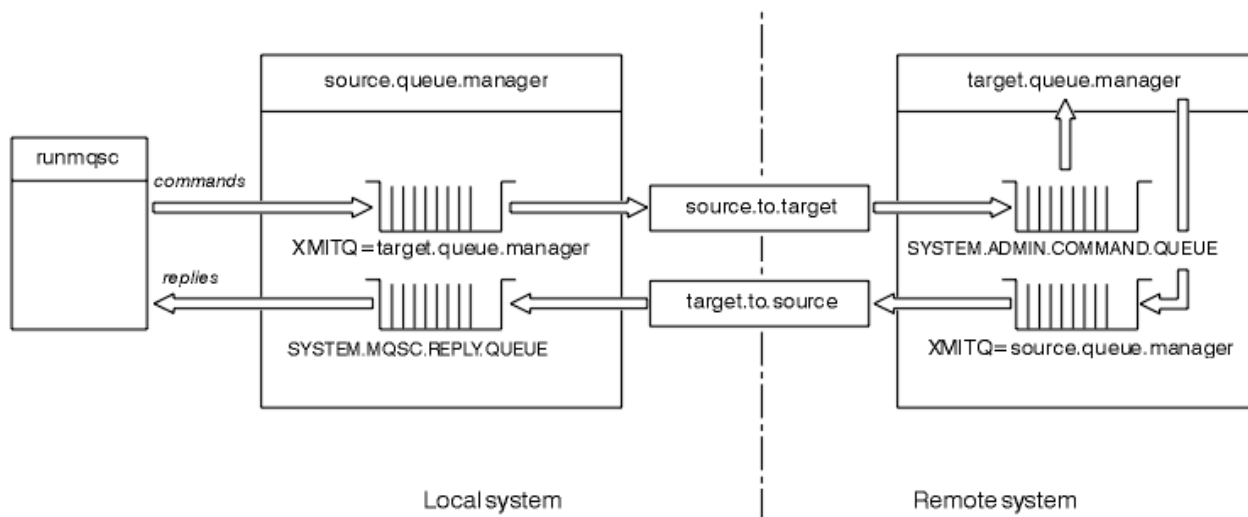
Zdalny menedżer kolejek można administrować z lokalnego menedżera kolejek za pomocą komend administrative REST API, MQSC lub PCF. Zdalny menedżer kolejek może znajdować się w tym samym systemie, w innej instalacji lub w innym systemie z tym samym środowiskiem lub z innym środowiskiem IBM MQ. Zanim możliwe będzie zdalne administrowanie menedżerem kolejek z lokalnego menedżera kolejek, należy utworzyć kanał nadawczy i odbiorczy, program nastuchujący oraz kolejkę transmisji dla każdego menedżera kolejek. Te kanały i kolejki umożliwiają wysyłanie komend do zdalnego menedżera kolejek i odpowiedzi, które mają być odebrane w lokalnym menedżerze kolejek. Procedura tworzenia tych kolejek i kanałów jest taka sama, niezależnie od tego, czy mają być używane komendy administrative REST API, MQSC, czy PCF.

Zanim rozpocznie

- W poniższej procedurze użyto przykładowych menedżerów kolejek `source.queue.manager1` i `target.queue.manager1`. Należy utworzyć i uruchomić te menedżery kolejek w systemie, aby wykonać te kroki, lub zastąpić własne nazwy menedżerów kolejek w odpowiednich krokach.
- W poniższej procedurze używany jest protokół TCP/IP jako typ transportu. Aby wykonać to zadanie, należy znać adres IP obu systemów.

- Poniższa procedura tworzy obiekty nasłuchiwania, które korzystają z portów sieciowych 1818 w systemie lokalnym i 1819 w systemie zdalnym. Można użyć innych portów, ale należy zastąpić wartości portów w odpowiednich krokach.
- Komendy należy uruchamiać lokalnie lub za pomocą narzędzia sieciowego, takiego jak Telnet.

O tym zadaniu



Rysunek 15. Konfigurowanie kanałów i kolejek do zdalnego administrowania

W programie [Rysunek 15 na stronie 193](#) wyświetlana jest konfiguracja menedżerów kolejek, kolejek i kanałów, które są potrzebne do zdalnego administrowania:

- Obiekt `source.queue.manager` jest źródłowym menedżerem kolejek, z którego można wydawać komendy administrative REST API, MQSC lub PCF, a także do których zwracane są wyniki tych komend.
- Obiekt `target.queue.manager` to nazwa docelowego menedżera kolejek, który przetwarza komendy i generuje dowolne komunikaty operatora.
- Komendy są umieszczane w kolejce transmisji, która ma taką samą nazwę jak zdalny menedżer kolejek. W tym przypadku `target.queue.manager`. Kolejka transmisji jest wyspecjalizowaną kolejką lokalną, która tymczasowo przechowuje komunikaty, zanim agent MCA pobiera je i wysyła do zdalnego menedżera kolejek.
- Komendy są wysyłane przez kanał produktu `source.to.target` do serwera `SYSTEM.ADMIN.COMMAND.QUEUE` w zdalnym menedżerze kolejek. Każdy koniec kanału ma osobną definicję. Jeden koniec to nadawca, a drugi-odbiorca. Te dwie definicje muszą mieć taką samą nazwę i razem tworzą jeden kanał komunikatów.
- Dane wyjściowe komendy są umieszczane w zdalnej kolejce transmisji, która ma taką samą nazwę, jak lokalny menedżer kolejek, z którego została wysłana komenda. W tym przypadku `source.queue.manager`.
- Dane wyjściowe są wysyłane przez kanał `target.to.source` do odpowiedniej kolejki odpowiedzi, gdzie jest ona pobierana i wyprowadzana przez oryginalną komendę.

Procedura

1. W menedżerze kolejek systemu zdalnego upewnij się, że kolejka komend `SYSTEM.ADMIN.COMMAND.QUEUE` jest obecna. Ta kolejka jest tworzona domyślnie, gdy tworzony jest menedżer kolejek.
2. W systemie zdalnym sprawdź, czy serwer komend jest uruchomiony w menedżerze kolejek. Jeśli serwer komend nie jest uruchomiony, zdalne administrowanie nie jest możliwe:

- a) Uruchom program **runmqsc** dla menedżera kolejek. Na przykład w przypadku menedżera kolejek `target.queue.manager` należy wprowadzić następującą komendę:

```
runmqsc target.queue.manager
```

- b) Wyświetl status serwera komend, wprowadzając następującą komendę:

```
DISPLAY QMSTATUS CMDSERV
```

- c) Wyjdź z programu **runmqsc**, wprowadzając następującą komendę:

```
end
```

- d) Jeśli serwer komend nie jest uruchomiony, uruchom go. Na przykład w przypadku menedżera kolejek `target.queue.manager` należy wprowadzić następującą komendę:

```
strmqcsv target.queue.manager
```

3. Zdefiniuj kanały, nastuchiwanie i kolejkę transmisji w lokalnym menedżerze kolejek:

- a) Uruchom program **runmqsc** dla menedżera kolejek. Na przykład w przypadku menedżera kolejek `source.queue.manager` należy wprowadzić następującą komendę:

```
runmqsc source.queue.manager
```

- b) Zdefiniuj kanał nadawczy. Ten kanał nadawczy musi mieć taką samą nazwę, jak kanał odbiorczy w zdalnym menedżerze kolejek. Na przykład wprowadź następującą komendę MQSC, zastępując wartość **CONNNAME** adresem IP dla zdalnego menedżera kolejek i numerem portu programu nastuchującego:

```
DEFINE CHANNEL ('source.to.target') +  
CHLTYPE(SDR) +  
CONNNAME (localhost:1819) +  
XMITQ ('target.queue.manager') +  
TRPTYPE(TCP)
```

- c) Zdefiniuj kanał odbiorczy. Ten kanał odbiorczy musi mieć taką samą nazwę, jak kanał nadawczy w zdalnym menedżerze kolejek. Na przykład:

```
DEFINE CHANNEL ('target.to.source') +  
CHLTYPE(RCVR) +  
TRPTYPE(TCP)
```

- d) Zdefiniuj program nastuchujący w lokalnym menedżerze kolejek. Na przykład:

```
DEFINE LISTENER ('source.queue.manager') +  
TRPTYPE (TCP) +  
PORT (1818)
```

- e) Zdefiniuj kolejkę transmisji w lokalnym menedżerze kolejek. Ta kolejka transmisji musi mieć taką samą nazwę, jak zdalny menedżer kolejek. Na przykład:

```
DEFINE QLOCAL ('target.queue.manager') +  
USAGE (XMITQ)
```

- f) Uruchom program nastuchujący. Na przykład:

```
START LISTENER ('source.queue.manager')
```

- g) Wyjdź z programu **runmqsc**, wprowadzając następującą komendę:

```
end
```

4. Zdefiniuj kanały, nastuchiwanie i kolejkę transmisji w zdalnym menedżerze kolejek:

- a) Uruchom program **runmqsc** dla menedżera kolejek. Na przykład w przypadku menedżera kolejek `target.queue.manager` należy wprowadzić następującą komendę:

```
runmqsc target.queue.manager
```

- b) Zdefiniuj kanał nadawczy. Ten kanał nadawczy musi mieć taką samą nazwę, jak kanał odbiorczy w lokalnym menedżerze kolejek. Na przykład wprowadź następującą komendę MQSC, zastępując wartość **CONNNAME** adresem IP lokalnego menedżera kolejek i numerem portu dla programu nastuchującego:

```
DEFINE CHANNEL ('target.to.source') +  
CHLTYPE(SDR) +  
CONNNAME (localhost:1818) +  
XMITQ ('source.queue.manager') +  
TRPTYPE(TCP)
```

- c) Zdefiniuj kanał odbiorczy. Ten kanał odbiorczy musi mieć taką samą nazwę, jak kanał nadawczy w lokalnym menedżerze kolejek. Na przykład należy wprowadzić następującą komendę:

```
DEFINE CHANNEL ('source.to.target') +  
CHLTYPE(RCVR) +  
TRPTYPE(TCP)
```

- d) Zdefiniuj obiekt nastuchiwania. Na przykład:

```
DEFINE LISTENER ('target.queue.manager') +  
TRPTYPE (TCP) +  
PORT (1819)
```

- e) Zdefiniuj kolejkę transmisji. Ta kolejka transmisji musi mieć taką samą nazwę, jak lokalny menedżer kolejek. Na przykład:

```
DEFINE QLOCAL ('source.queue.manager') +  
USAGE (XMITQ)
```

- f) Uruchom program nastuchujący. Na przykład:

```
START LISTENER ('target.queue.manager')
```

- g) Wyjdź z programu **runmqsc**, wprowadzając następującą komendę:

```
end
```

5. Uruchom kanał nadawczy w systemie lokalnym:

- a) Uruchom program **runmqsc** dla menedżera kolejek. Na przykład w przypadku menedżera kolejek `source.queue.manager` należy wprowadzić następującą komendę:

```
runmqsc source.queue.manager
```

- b) Uruchom kanał nadawczy. Na przykład:

```
START CHANNEL ('source.to.target')
```

- c) Wyjdź z programu **runmqsc**, wprowadzając następującą komendę:

```
end
```

6. Uruchom kanał nadawczy w systemie zdalnym:

- a) Uruchom komendę `runmqsc` dla menedżera kolejek. Na przykład w przypadku menedżera kolejek `target.queue.manager` należy wprowadzić następującą komendę:

```
runmqsc target.queue.manager
```

- b) Uruchom kanał nadawczy. Na przykład:

```
START CHANNEL ('target.to.source')
```

- c) Wyjdź z programu **runmqsc**, wprowadzając następującą komendę:

```
end
```

7. Przetestuj, czy konfiguracja została zakończona pomyślnie, wysyłając komendę MQSC z systemu lokalnego do zdalnego menedżera kolejek:

- a) Uruchom program **runmqsc** dla zdalnego menedżera kolejek z systemu lokalnego. Na przykład:

```
runmqsc -w 30 -m source.queue.manager target.queue.manager
```

- b) Wyświetl kolejki w zdalnym menedżerze kolejek, wprowadzając następującą komendę:

```
DISPLAY QUEUE (*)
```

Po pomyślnym zakończeniu wyświetlania zostanie wyświetlona lista kolejek ze zdalnego menedżera kolejek.

- c) Jeśli te kroki nie działają, sprawdź, czy kanały w obu systemach są w stanie działania. Jeśli kanały nie są uruchomione i nie uruchamiają się, sprawdź, czy kanały i kolejki transmisji są skonfigurowane poprawnie i czy serwer komend jest uruchomiony. Na przykład sprawdź, czy dla kanałów nadawczych określono poprawną wartość `CONNNAME`, a także czy kolejki transmisji mają poprawne nazwy. Ponadto należy sprawdzić dzienniki menedżera kolejek pod kątem wyjątków zabezpieczeń, które mogą pomóc w rozwiązaniu problemu.

Wyniki

Menedżery kolejek są skonfigurowane do zdalnego administrowania zdalnym menedżerem kolejek z systemu lokalnego.

Co dalej

- Informacje na temat zdalnego administrowania za pomocą komend MQSC: [“Wydawanie komend MQSC w zdalnym menedżerze kolejek”](#) na stronie 197
- Dowiedz się więcej na temat pisania programów administracyjnych za pomocą komend PCF: [“Korzystanie z formatów komend programowalnych IBM MQ”](#) na stronie 26.
- Więcej informacji na temat używania konsoli administrative REST API do zdalnego administrowania: [“Zdalne administrowanie przy użyciu REST API”](#) na stronie 80.

Zarządzanie serwerem komend w celu zdalnego administrowania

Z każdym menedżerem kolejek powiązany jest 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. Użytkownik może uruchomić, zatrzymać i wyświetlić status serwera komend. Serwer komend jest obowiązkowy dla wszystkich funkcji administracyjnych związanych z komendami PCF, MQAI, a także dla zdalnego administrowania.

Zanim rozpoczniesz

W zależności od wartości atrybutu menedżera kolejek **SCMDSERV**, serwer komend jest uruchamiany automatycznie podczas uruchamiania menedżera kolejek lub musi być uruchamiany ręcznie. Jeśli serwer komend jest uruchamiany automatycznie, nie można użyć komend `strmqcsv` ani `endmqcsv`, aby uruchomić i zatrzymać serwer komend. Wartość atrybutu **SCMDSERV** można zmienić za pomocą komendy `MQSC ALTER QMGR`. Domyślnie serwer komend jest uruchamiany automatycznie.

Zatrzymanie menedżera kolejek powoduje również zakończenie powiązanego z nim serwera komend.

Procedura

- Wyświetl status serwera komend:
 - Uruchom program **runmqsc** dla odpowiedniego menedżera kolejek, wprowadzając następującą komendę:

```
runmqsc target.queue.manager
```

gdzie `target.queue.manager` jest menedżerem kolejek, dla którego wyświetlany jest serwer komend.

- Wyświetl status serwera komend, wprowadzając następującą komendę MQSC:

```
DISPLAY QMSTATUS CMDSERV
```

- Wyjdź z programu **runmqsc**, wprowadzając następującą komendę:

```
end
```

- Jeśli serwer komend nie jest ustawiony do uruchamiania automatycznie, uruchom serwer komend, wprowadzając następującą komendę:

```
strmqcsv target.queue.manager
```

gdzie `target.queue.manager` jest menedżerem kolejek, dla którego uruchamiany jest serwer komend.

- Jeśli serwer komend nie jest ustawiony do automatycznego uruchamiania, zatrzymaj serwer komend, wprowadzając następującą komendę:

```
endmqcsv target.queue.manager
```

gdzie `target.queue.manager` jest menedżerem kolejek, dla którego zatrzymano serwer komend.

Domyślnie serwer komend zatrzymuje się w kontrolowany sposób. Serwer komend można zatrzymać natychmiast, dodając flagę `-i` do komendy.

Wydawanie komend MQSC w zdalnym menedżerze kolejek

Po skonfigurowaniu menedżerów kolejek w celu zdalnego administrowania można użyć konkretnego formularza komendy **runmqsc** w systemie lokalnym, aby uruchomić komendy MQSC w zdalnym menedżerze kolejek. Każda komenda jest wysyłana jako komenda Escape PCF do kolejki komend (`SYSTEM.ADMIN.COMMAND.QUEUE`, menedżera kolejek zdalnych. Odpowiedzi są odbierane w systemie `SYSTEM.MQSC.REPLY.QUEUE` .

Zanim rozpoczniesz

Aby zdalnie administrować menedżerem kolejek za pomocą komend MQSC, należy wykonać kroki opisane w sekcji [“Konfigurowanie menedżerów kolejek na potrzeby zdalnego administrowania”](#) na stronie 192 , aby skonfigurować kanały, kolejki transmisji, obiekty nasłuchiwania i serwer komend.

Procedura

1. Upewnij się, że serwer komend jest uruchomiony w zdalnym menedżerze kolejek.

Więcej informacji na temat uruchamiania serwera komend w menedżerze kolejek zawiera sekcja [“Zarządzanie serwerem komend w celu zdalnego administrowania”](#) na stronie 196.

2. W źródłowym menedżerze kolejek można następnie uruchomić komendy MQSC na jeden z dwóch sposobów:

- Interaktywnie, uruchamiając program **runmqsc** z następującymi komendami:

- **z/OS** Jeśli zdalny menedżer kolejek znajduje się w systemie z/OS, wprowadź następującą komendę:

```
runmqsc -w 30 -x -m source.queue.manager target.queue.manager
```

- **Multi** Jeśli zdalny menedżer kolejek nie znajduje się w systemie z/OS, wprowadź następującą komendę:

```
runmqsc -w 30 -m source.queue.manager target.queue.manager
```

- Z pliku komend:

- a. Umieść komendy MQSC, które mają być uruchamiane w systemie zdalnym w pliku tekstowym, po jednej komendzie w wierszu.
- b. Sprawdź, czy komendy MQSC są dostępne w lokalnym menedżerze kolejek, używając opcji **-v** w komendzie **runmqsc**. Opcja **-v** sprawdza, czy komendy są poprawne, ale nie uruchamia ich. Należy pamiętać, że niektóre komendy mogą się nie powieść, jeśli mają one zastosowanie do zdalnego menedżera kolejek, ale nie mają zastosowania do lokalnego menedżera kolejek:

```
runmqsc -v source.queue.manager < myCmdFile.in > results.out
```

`myCmdFile.in` zawiera komendy MQSC używane do sprawdzania, a plik `results.out` zawiera wyniki weryfikacji dla komend.

- c. Uruchom plik komend w zdalnym menedżerze kolejek, wprowadzając jedną z następujących komend:

- **z/OS** Jeśli zdalny menedżer kolejek znajduje się w systemie z/OS, wprowadź następującą komendę:

```
runmqsc -w 30 -x -m source.queue.manager target.queue.manager < myCmdFile.in >  
results.out
```

- **Multi** Jeśli zdalny menedżer kolejek nie znajduje się w systemie z/OS, wprowadź następującą komendę:

```
runmqsc -w 30 -m source.queue.manager target.queue.manager < myCmdFile.in >  
results.out
```

Używane parametry to następujące parametry:

-w sekundy

Określa, że komendy MQSC są uruchamiane w trybie pośrednim, w którym komendy są umieszczane w kolejce wejściowej serwera komend i wykonywane w taki sposób, aby były wykonywane w kolejności.

Zmienna *sekundy* określa czas oczekiwania (w sekundach) na odpowiedź ze zdalnego menedżera kolejek. Wszystkie odpowiedzi odebrane po tym czasie są odrzucane, ale komendy MQSC nadal

działają w zdalnym menedżerze kolejek. Po uruchomieniu komendy w lokalnym menedżerze kolejek generowany jest 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.

-x

Określa, że zdalny menedżer kolejek jest menedżerem kolejek produktu z/OS .

-m **localQMgrNazwa**

Określa nazwę lokalnego menedżera kolejek, który ma być używany do wprowadzania komend do menedżera kolejek zdalnych.

Co dalej

Jeśli masz problemy z uruchamianiem zdalnie komendy MQSC:

- Sprawdź, czy zdalny menedżer kolejek jest uruchomiony.
- Sprawdź, czy serwer komend jest uruchomiony w systemie zdalnym.
- Sprawdź, czy interwał odłączania kanału nie utracił ważności. Na przykład, jeśli kanał został uruchomiony, ale później został zamknięty po pewnym czasie. Jest to szczególnie ważne w przypadku ręcznego uruchomienia kanałów.
- Upewnij się, że żądania wysyłane z lokalnego menedżera kolejek mają sens w stosunku do docelowego menedżera kolejek. Na przykład żądania, które zawierają parametry, które nie są obsługiwane w zdalnym menedżerze kolejek.
- Więcej informacji na ten temat zawiera sekcja [Rozwiązywanie problemów z komendami MQSC](#).


Konwersja danych między kodowanymi zestawami znaków

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.

Informacje na temat konwersji znaków w formacie EBCDIC newline (NL) na ASCII zawiera sekcja [Wszystkie menedżery kolejek](#).

Obsługiwane konwersje są zdefiniowane w sekcji [Przetwarzanie konwersji danych](#).

 **V 9.2.0** Począwszy od wersji IBM MQ 9.2.0, konwersja między identyfikatorami CCSID 37 i 500 jest obsługiwana w systemach IBM MQ Appliance, Windows, Linux i macOS.

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 za pomocą skryptu cyrylicowego) 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.

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 203](#), 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 201.

Rozszerzona obsługa konwersji danych Unicode

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 podzbiórce o stałej szerokości UTF-16.

W produkcie IBM MQ 9.0 produkt 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 obsługiwane także w przypadkach, gdy wstępnie złożony znak w jednym identyfikatorze CCSID jest odwzorowany 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 również obsługę konwersji do i z kodu Unicode i sześciu nowych identyfikatorów CCSID (od 1374 do 1379).



Plik `ccsid_part2.tbl`


Z programu IBM MQ 9.0 udostępniany jest dodatkowy plik `ccsid_part2.tbl`.


Plik `ccsid_part2.tbl` ma pierwszeństwo przed plikiem `ccsid.tbl` oraz:

- Umożliwia dodawanie i modyfikowanie pozycji CCSID
- Określa domyślną konwersję danych
- Określa dane dla różnych poziomów komend

`ccsid_part2.tbl` ma zastosowanie tylko do następujących platform:

-  Wszystkie wersje systemu Linux
-  Windows

 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.


 W systemie IBM MQ for Linux produkt `ccsid_part2.tbl` znajduje się w katalogu `MQDataRoot/conv/table`, a 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 [“Plik `ccsid_part2.tbl`”](#) na stronie 202.

Plik `ccsid.tbl`



Na platformach innych niż te, do których ma zastosowanie produkt `ccsid_part2.tbl`, plik `ccsid.tbl` jest używany do następujących celów:

-  W systemie operacyjnym AIX obsługiwane zestawy kodów są przechowywane wewnętrznie przez system operacyjny.
- Określa on 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

Metoda domyślnej konwersji danych została zmieniona z IBM MQ 9.0 na następujące platformy:

-  Wszystkie wersje systemu Linux
-  Windows

Więcej informacji na ten temat zawiera `ccsid_part2.tbl` sekcja [“Określanie domyślnej konwersji danych”](#) na stronie 203.

Jeśli kanały są skonfigurowane między dwoma komputerami, w których konwersja danych nie jest normalnie obsługiwana, należy włączyć domyślną konwersję danych dla kanałów do pracy.

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

Plik `ccsid_part2.tbl` jest używany do dostarczania dodatkowych informacji o identyfikatorze CCSID. Plik `ccsid_part2.tbl` zastępuje plik `ccsid.tbl`, który był używany przed plikiem IBM MQ 9.0.

Uwaga: Plik `ccsid.tbl`, który był używany przed plikiem IBM MQ 9.0 do dostarczenia dodatkowych informacji o identyfikatorze CCSID, jest nadal analizowany przez plik IBM MQ i nie powinien być usuwany. Jednak wpisy w pliku `ccsid_part2.tbl` mają pierwszeństwo przed innymi wpisami w pliku `ccsid.tbl`.



Należy użyć parametru `ccsid_part2.tbl`, a nie `ccsid.tbl`, ponieważ `ccsid_part2.tbl`:

- Zawiera obsługę wartości kodowania Unicode. Począwszy od wersji IBM MQ 9.0, produkt obsługuje wszystkie znaki Unicode zdefiniowane w standardzie Unicode 8.0 podczas konwersji danych, w tym pełną obsługę kodowania UTF-16. Więcej informacji na ten temat zawiera sekcja [“Konwersja danych między kodowanymi zestawami znaków”](#) na stronie 199.
- Umożliwia określenie wersji pozycji CCSID, tak aby pozycje te miały zastosowanie tylko do wybranych poziomów komend.



Pliku `ccsid_part2.tbl` można użyć do:

- Dodaj lub zmodyfikuj pozycje CCSID
- Określa domyślną konwersję danych
- Określa dane dla różnych poziomów komend

Plik `ccsid_part2.tbl` dotyczy tylko następujących platform:

-  Wszystkie wersje systemu Linux
-  Windows

Położenie pliku `ccsid_part2.tbl` zależy od używanej platformy:

-  Katalog `MQDataRoot/conv/table` we wszystkich wersjach systemu Linux.
-  Katalog `MQDataRoot\conv\table` w systemie Windows.

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>
```

Przykładowy wpis dla identyfikatora CCSID 1200 (UTF-16):

```
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 = Euc

7 = GB18030

8 = UTF-16

9 = UTF-32

Podczas edycji pliku:

- Komentarz można określić za pomocą symbolu # na początku wiersza. Zapobiega to próbie przeanalizowania wiersza przez program IBM MQ .
- Nie można podać komentarzy wstawianych.
- Należy upewnić się, że nie są tworzone puste wiersze.
- Nie można dodawać nowych wpisów na końcu pliku.

Nowe pozycje CCSID powinny zostać dodane przed informacjami o tabeli ACRI.

Określanie domyślnej konwersji danych

Jeśli konwersja między dwoma identyfikatorami CCSID nie jest obsługiwana, można zdefiniować domyślne identyfikatory CCSID konwersji, które są używane do konwersji między identyfikatorami ASCII lub podobnymi i identyfikatorami CCSID EBCDIC.

Jeśli ta funkcja zostanie włączona, do transmisji i nagłówków komunikatów będzie używana konwersja domyślna, a także podczas konwersji danych użytkownika.

Konwersje domyślne można włączyć, tworząc dwa wiersze podobne do następujących:

```
default      0      500      1      1      0
default      0      850      1      2      0
```

Pierwszy wiersz ustawia wartość domyślną dla identyfikatorów CCSID EBCDIC na 500, a drugi wiersz ustawia wartość domyślną dla identyfikatorów CCSID ASCII i podobnych identyfikatorów CCSID na 850.

Określanie danych dla różnych poziomów komend

Aby określić pozycje CCSID dla różnych poziomów komend systemu IBM MQ , należy użyć symbolu dwukropka, po którym następuje poziom komendy (lub poziomy komend) systemu IBM MQ , do którego ma być zastosowana następna sekcja.

Liczba ta reprezentuje minimalny poziom komend, na którym musi działać menedżer kolejek lub klient. Na przykład, jeśli bieżący menedżer kolejek jest na poziomie komend 900 i napotka opcję poziomu komend 800 lub 900, zostaną odczytane identyfikatory CCSID.

Jednak menedżer kolejek na poziomie 800 ignoruje wszystkie identyfikatory CCSID w sekcji 900.

Określony 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 komend na wszystkie poziomy komend, należy podać liczbę zero.

Podczas pierwszego analizowania pliku `ccsid_part2.tbl` program IBM MQ traktuje wszystkie napotkane identyfikatory CCSID jako poprawne dla wszystkich poziomów komend systemu IBM MQ.

Kontrola wersji jest uruchamiana tylko wtedy, gdy program IBM MQ napotka pierwszą flagę poziomu 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
```

administrowanie Managed File Transfer

Za pomocą komend Managed File Transfer można administrować programem Managed File Transfer. Do niektórych zadań administracyjnych można również użyć IBM MQ Explorer .

Rozpocznij przesłanie, umieszczając komunikat w kolejce komend agenta

Przesyłanie plików można również uruchomić, umieszczając komunikat przesłania plików w kolejce komend agenta źródłowego. Przykładowa nazwa kolejki komend to `SYSTEM.FTE.COMMAND.AGENT01`. Należy upewnić się, że komunikat dociera do kolejki komend odpowiedniego agenta źródłowego. Jeśli komunikat zostanie odebrany przez agenta, który nie jest zgodny z informacjami źródłowymi w kodzie XML, komunikat zostanie odrzucony.


Kod XML żądania przesłania musi być zgodny ze schematem produktu `FileTransfer.xsd` i musi być używany jako element główny elementu `<request>`. Informacje na temat struktury i treści komunikatu żądania przesłania znajdują się w sekcji [Format komunikatu żądania przesłania plików](#) . Sposób umieszczenia komunikatu żądania transferu w kolejce komend agenta jest specyficzny dla zadania. Na przykład można użyć funkcji API IBM MQ Java w celu programowego umieszczenia komunikatu w kolejce.


Uruchamianie agenta MFT

Zanim będzie można użyć agenta Managed File Transfer do przesłania plików, należy najpierw uruchomić agenta.

O tym zadaniu



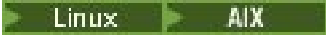

Program Managed File Transfer Agent można uruchomić z poziomu wiersza komend. W takim przypadku proces agenta zostanie zatrzymany po wylogowaniu się z systemu.

 W systemie AIX, Linux, and Windows można skonfigurować agenta w taki sposób, aby był on kontynuowany po wylogowaniu się z systemu i w dalszym ciągu otrzymując przesłanie plików.

 W systemie z/OS można skonfigurować agenta, tak aby uruchamiał jako zadanie uruchomione z JCL bez konieczności wykonywania sesji interaktywnej.

Należy zauważyć, że jeśli agent napotka błąd nienaprawialny podczas jego działania, zostanie wygenerowany pierwszy przechwytywanie danych o błędzie (FDC), a agent zostanie zatrzymany.

Procedura

- Aby uruchomić agenta z poziomu wiersza komend, należy użyć komendy **fteStartAgent** . Więcej informacji na ten temat zawiera sekcja [fteStartAgent](#) .
-  Aby skonfigurować agenta w taki sposób, aby był on kontynuowany po wylogowaniu się z systemu:
 -  W systemie Windows skonfiguruj agenta, tak aby był uruchamiany jako usługa Windows . Więcej informacji na ten temat zawiera sekcja [“Uruchamianie agenta MFT jako usługi Windows”](#) na stronie 205.
 -  W systemie AIX and Linux skonfiguruj agenta w taki sposób, aby uruchamiał się automatycznie podczas restartu za pomocą pliku skryptowego. Więcej informacji na ten temat zawiera sekcja [“Uruchamianie agenta MFT podczas uruchamiania systemu AIX and Linux”](#) na stronie 207.
-  W systemie z/OS należy skonfigurować agenta, aby uruchamiał jako zadanie uruchomione z JCL bez konieczności wykonywania sesji interaktywnej. Więcej informacji na ten temat zawiera [“Uruchamianie agenta MFT w systemie z/OS”](#) na stronie 208.

Uruchamianie agenta MFT jako usługi Windows

Agenta można uruchomić jako usługę Windows , dzięki czemu po wylogowaniu się z programu Windows agent będzie nadal działał i będzie mógł odbierać przesyłanie plików.

O tym zadaniu

W systemie Windows po uruchomieniu agenta z poziomu wiersza komend proces agenta jest uruchamiany z użyciem nazwy użytkownika używanej do logowania się do produktu Windows. Po wylogowaniu się z systemu proces agenta zostanie zatrzymany. Aby zapobiec zatrzymaniu agenta, można skonfigurować agenta w taki sposób, aby był uruchamiany jako usługa Windows . Uruchamianie jako usługa Windows umożliwia również automatyczne uruchamianie agentów po uruchomieniu lub zrestartowaniu środowiska produktu Windows .

Wykonaj następujące kroki, aby uruchomić agenta, który działa jako usługa Windows . Aby uruchomić agenta jako usługę Windows , należy uruchomić produkt Managed File Transfer w jednej z obsługiwanych wersji produktu Windows . Lista obsługiwanych środowisk znajduje się w sekcji [Wymagania systemowe produktu IBM MQ](#) .

Dokładne kroki zależą od tego, czy został już utworzony agent, czy też jest tworzony agent. Obie opcje są opisane w poniższych krokach.

Procedura

1. If you are creating a Managed File Transfer agent, use the **fteCreateAgent**, **fteCreateCDAgent**, or **fteCreateBridgeAgent** command. Podaj parametr **-s** , aby uruchomić agenta jako usługę Windows . W poniższym przykładzie agent AGENT1 jest tworzony, który ma menedżera kolejek agenta QMGR1. Usługa Windows jest uruchamiana przy użyciu nazwy użytkownika fteuser, która ma przypisane hasło ftepassword.

```
fteCreateAgent -agentName AGENT1 -agentQMGr QMGR1 -s -su fteuser -sp ftepassword
```

Po parametrze **-s** można opcjonalnie podać nazwę usługi. Jeśli nazwa nie zostanie podana, usługa ma nazwę `mqmftAgentAGENTQMGR`, gdzie *AGENT* jest nazwą agenta, a *QMGR* jest nazwą menedżera kolejek agenta. W tym przykładzie domyślną nazwą usługi jest `mqmftAgentAGENT1QMGR1`.

Uwaga: Konto użytkownika Windows określone za pomocą parametru **-su** musi mieć uprawnienia **Log on as a service**. Informacje na temat konfigurowania tego produktu zawiera sekcja [Wskazówki dotyczące uruchamiania agenta lub programu rejestrującego produktu MFT jako usługi produktu Windows](#).

For more information, see [Agent fteCreate, fteCreateCDAgent: tworzenie agenta mostu Connect:Direct, or fteCreateBridgeAgent](#) (tworzenie i konfigurowanie agenta mostu protokołu MFT).

2. If you followed the previous step to create an agent, run the MQSC commands that are generated by the **fteCreateAgent**, **fteCreateCDAgent**, or **fteCreateBridgeAgent** command. Te komendy tworzą kolejki produktu IBM MQ, które są wymagane przez agenta.

Na przykład dla agenta o nazwie *AGENT1*, menedżera kolejek agenta o nazwie *QMGR1* i menedżera kolejek koordynacji o nazwie *COORDQMGR1*, należy uruchomić następującą komendę:

```
runmqsc QMGR1 MQ_DATA_PATH\mqft\config\COORDQMGR1\agents\AGENT1\AGENT1_create.mqsc
```

3. Jeśli nie wykonałeś poprzednich kroków, aby utworzyć agenta, a zamiast tego chcesz skonfigurować istniejącego agenta do uruchomienia jako usługę Windows, najpierw zatrzymaj agenta, jeśli jest on uruchomiony, a następnie zmodyfikuj jego konfigurację.

a) W poniższym przykładzie użyto agenta o nazwie *AGENT1*. Uruchom następującą komendę:

```
fteStopAgent AGENT1
```

b) Użyj komendy **fteModifyAgent**, aby skonfigurować agenta do uruchamiania jako usługę Windows:

```
fteModifyAgent -agentName AGENT1 -s -su fteuser -sp ftepassword
```

Więcej informacji na ten temat zawiera sekcja [fteModifyAgent: uruchom agenta MFT jako usługę Windows](#).

4. Uruchom agenta za pomocą komendy **fteStartAgent**. Alternatywnie można użyć narzędzia Windows Services, które jest dostępne w narzędziach administracyjnych w Panelu sterowania, wybranym w menu Start pulpitu Windows, w celu uruchomienia usługi.

```
fteStartAgent AGENT1
```

Usługa jest nadal uruchamiana, nawet jeśli użytkownik wyloguje się z produktu Windows. Aby upewnić się, że usługa zostanie zrestartowana po zrestartowaniu produktu Windows po zamknięciu, pole **Typ uruchamiania** w narzędziu Windows Services jest domyślnie ustawione na wartość **Automatyczny**. Zmień tę wartość na **Ręcznie**, jeśli nie chcesz, aby usługa była restartowana po zrestartowaniu produktu Windows.

5. Opcjonalne: Aby zatrzymać agenta, użyj komendy `fteStopAgent` lub użyj narzędzia Windows Services. Na przykład z poziomu wiersza komend uruchom następującą komendę:

```
fteStopAgent AGENT1
```

- Gdy komenda **fteStopAgent** jest uruchamiana jako usługa, komenda zawsze uruchamiana jest za pomocą parametru **-i**, niezależnie od tego, czy został określony ten parametr. Parametr **-i** natychmiast zatrzymuje agenta bez wykonywania operacji przesyłania, które są w toku. Jest to spowodowane ograniczeniem usługi Windows.

Co dalej

Jeśli występują problemy z uruchomieniem usługi produktu Windows , należy zapoznać się z sekcji [Wskazówki dotyczące uruchamiania agenta lub programu rejestrującego produktu MFT w postaci usługi produktu Windows](#). W tym temacie opisano także położenie plików dziennika usługi produktu Windows .

Linux

AIX

Uruchamianie agenta MFT podczas uruchamiania systemu AIX and Linux

Managed File Transfer Agent można skonfigurować w taki sposób, aby uruchamiał się podczas uruchamiania systemu w systemie AIX and Linux. Po wylogowaniu agent będzie kontynuował działanie i będzie mógł odbierać przesyłanie plików.

Po utworzeniu i skonfigurowaniu agenta za pomocą jednej z następujących komend produktu Managed File Transfer : **fteCreateAgent**, **fteCreateCDAgent**, lub **fteCreateBridgeAgent**, można ją skonfigurować w taki sposób, aby uruchamiał się automatycznie podczas restartu na komputerach z produktem AIX and Linux przy użyciu pliku skryptowego, który po prostu wykonuje następującą komendę:

```
su -l mqmft_user -c mq_install_root/bin/fteStartAgent agent_name
```

Gdzie *mq_install_root* jest katalogiem głównym wymaganej instalacji produktu Managed File Transfer , wartością domyślną jest: */opt/mqm* , a *nazwa_agenta* jest nazwą serwera Managed File Transfer Agent , który ma zostać uruchomiony. Korzystanie z tego pliku skryptowego zależy od konkretnego systemu operacyjnego. Na przykład w sekcji Linux dostępne są dodatkowe opcje.

Linux

Linux

W przypadku systemów Linux istnieje wiele sposobów uruchamiania aplikacji podczas procesu uruchamiania systemu. Ogólnie, należy rozważyć następujące kroki:

1. Utwórz plik o nazwie */etc/rc.mqmft* z treścią:

```
#!/bin/sh
su -l mqmft_user -c mq_install_root/bin/fteStartAgent agent_name"
```

Gdzie *mqmft_user* jest identyfikatorem użytkownika, który ma być uruchamiany przez proces agenta. Ten identyfikator użytkownika musi należeć do grupy *mqm*.

2. Ustaw plik wykonywalny, na przykład:

```
chmod 755 /etc/rc.mqmft
```

3. Następnie dodaj następujący wiersz do programu */etc/inittab*:

```
mqmft:5:boot:/etc/rc.mqmft
```

Innymi sposobami uruchamiania agenta podczas startu systemu Linux są dodawanie linii skryptów do pliku */etc/rc.d/rc.local* lub Linux SuSe, dodawanie linii skryptów do pliku */etc/init.d/boot.local* . Należy wybrać metodę, która działa najlepiej dla danego środowiska. Poniżej znajduje się więcej informacji na temat innych sposobów uruchamiania agenta podczas uruchamiania na konkretnych dystrybucjach Linux , które są obsługiwane:

SLES 10 i 11

W przypadku systemów SUSE Linux Enterprise Server (SLES) 10 i 11 wykonaj następujące kroki:

1. Jako systemowy identyfikator użytkownika root utwórz własny plik */etc/init.d/rc.local* .

2. Dodaj następujące wiersze do pliku `rc.rclocal` :

```
#!/bin/sh
### BEGIN INIT INFO
# Provides: rc.rclocal
# Required-Start: $network $syslog
# Required-Stop: $network $syslog
# Default-Stop: 0 1 2 6
# Description: MQMFT agent startup
### END INIT INFO
su -l mqmft_user"-c mq_install_root/bin/fteStartAgent agent_name"
```

3. Wykonaj następujące komendy:

```
chmod 755 rc.rclocal

chkconfig --add rc.rclocal
```

Uruchamianie agentów Managed File Transfer w systemie Linux z produktem systemd

Linux

Wykonaj następującą procedurę:

1. Utwórz plik w folderze systemowym `/etc/systemd/` i nazwij go, na przykład `<agentname>.service`. Dodaj następującą treść, gdzie `<agentname>` to `MFT_AGT_LNX_0`.

```
# vi /etc/systemd/system/MFT_AGT_LNX_0.service
[Unit]
Description=IBM MQ MFT MFT_AGT_LNX_0
[Service]
ExecStart=/opt/mqm/bin/fteStartAgent MFT_AGT_LNX_0
ExecStop=/opt/mqm/bin/fteStopAgent MFT_AGT_LNX_0
Type=forking
User=mqm
Group=mqm
KillMode=none
```

2. Aby włączyć usługę, uruchom następujące komendy:

```
# systemctl enable MFT_AGT_LNX_0
# systemctl daemon-reload
```

3. Aby uruchomić agenta i sprawdzić jego status, uruchom następujące komendy:

```
# systemctl start MFT_AGT_LNX_0
# systemctl status MFT_AGT_LNX_0
```

z/OS

Uruchamianie agenta MFT w systemie z/OS

W systemie z/OS, oprócz uruchamiania komendy **fteStartAgent** z sesji z/OS UNIX System Services , można uruchomić agenta jako uruchomione zadanie z JCL bez potrzeby sesji interaktywnej.

Uruchomione zadanie jest używane, ponieważ działa z określonym identyfikatorem użytkownika i nie ma na niego wpływu wylogowanie się użytkowników.

Uwaga: Uruchomione zadania są zwykle uruchamiane przez użytkownika administracyjnego, który może nie mieć uprawnień do logowania się i dlatego nie jest możliwe zalogowanie się do systemu z/OS jako użytkownik, który uruchomił agenta. Komendy **fteStartAgent**, **fteStopAgent**, **fteSetAgentTraceLevel** i **fteShowAgentDetails** z określonym parametrem **-d** nie mogą być wydawane dla tego agenta.

W systemach IBM MQ 9.0.2 i IBM MQ 9.0.0 Fix Pack 1 można użyć właściwości agenta **adminGroup** z agentami Managed File Transfer w systemie z/OS. Można zdefiniować grupę menedżera zabezpieczeń, na przykład MFTADMIN, a następnie dodać do tej grupy identyfikator użytkownika i identyfikator TSO administratora uruchomionego zadania. Zmodyfikuj plik właściwości agenta i ustaw właściwość **adminGroup** na nazwę tej grupy menedżera zabezpieczeń.

```
adminGroup=MFTADMIN
```

Członkowie tej grupy mogą następnie wydawać komendy **fteStartAgent**, **fteStopAgent** i **fteSetAgentTraceLevel** oraz komendę **fteShowAgentDetails** z określonym parametrem **-d** dla agenta działającego jako uruchomione zadanie.

Więcej informacji na ten temat zawiera opis właściwości **adminGroup** w sekcji [Plik MFT agent.properties](#).

Jako aplikacja Java agent jest aplikacją z/OS UNIX System Services, którą można uruchomić z JCL za pomocą elementu BFGAGSTP z wygenerowanego zestawu danych biblioteki PDSE komendy Managed File Transfer dla agenta. Więcej informacji na temat tworzenia zestawu danych biblioteki PDSE komendy MFT i dostosowywania go dla wymaganego agenta zawiera sekcja [Tworzenie zestawu danych komendy agenta lub programu rejestrującego MFT](#).

Umożliwianie agentom nawiązywania połączeń ze zdalnymi menedżerami kolejek systemu z/OS

LTS

Agent MFT w systemie z/OS może połączyć się z menedżerem kolejek w systemie z/OS za pomocą połączenia klienckiego w następujących scenariuszach:

- Agent MFT ma numer IBM MQ 9.2.0 Long Term Support (obsługa długoterminowa) z zastosowanym raportem APAR PH56722 i został powiązany z identyfikatorem produktu (PID) IBM MQ Advanced for z/OS Value Unit Edition lub IBM MQ Advanced for z/OS
- Agent MFT jest w IBM MQ 9.2.0 i został powiązany z identyfikatorem PID IBM MQ Advanced for z/OS Value Unit Edition

Szczegółowe informacje na temat produktów IBM MQ, powiązanych z nimi wartości PID i klasyfikacji eksportu zawiera sekcja [Identyfikatory produktów IBM MQ i informacje o eksporcie](#).

Informacje na temat ustawiania identyfikatora PID powiązanego z instalacją MFT zawiera sekcja [fteSetProductID](#).

Identyfikator PID, pod którym działa agent, jest wyświetlany w dzienniku podczas uruchamiania agenta.

Uwaga: Agent MFT w systemie z/OS działający z dowolnym innym identyfikatorem PID może nawiązać połączenie z lokalnym menedżerem kolejek tylko przy użyciu połączenia w trybie powiązań.

Jeśli agent podejmie próbę nawiązania połączenia z menedżerem kolejek, który nie jest uruchomiony w systemie z/OS, zostanie wygenerowany komunikat BFGQM1044E, a uruchamianie agenta zostanie zakończone.

Odsyłacze pokrewne

[“Zatrzymywanie agenta MFT w systemie z/OS” na stronie 210](#)

Jeśli program Managed File Transfer Agent jest uruchamiany w systemie z/OS jako uruchomione zadanie z JCL, agent akceptuje komendy operatora z/OS **MODIFY** i **STOP**, oprócz komendy **fteStopAgent**.

[Plik MFT agent.properties](#)

Wyświetlanie listy agentów MFT

Listę agentów Managed File Transfer zarejestrowanych w określonym menedżerze kolejek można wyświetlić za pomocą wiersza komend lub IBM MQ Explorer.

O tym zadaniu

Aby wyświetlić listę agentów za pomocą wiersza komend, patrz [Komenda Agenty produktu fteList](#).

Aby wyświetlić listę agentów za pomocą konsoli IBM MQ Explorer, w widoku Navigator kliknij opcję **Agenty** w sekcji Nazwa menedżera kolejek koordynacji.

Jeśli agent nie jest wymieniony w komendzie **fteListAgents** lub nie jest wyświetlany w IBM MQ Explorer, należy użyć schematu blokowego diagnozy w następującym temacie, aby znaleźć i rozwiązać problem: [Co zrobić, jeśli agent MFT nie jest wymieniony w komendzie fteListAgents](#).

Zatrzymywanie agenta MFT

Agenta Managed File Transfer można zatrzymać z poziomu wiersza komend. Zatrzymując agenta, można wyciszać agenta i umożliwić agentowi zakończenie bieżącego przesyłania plików przed zatrzymaniem. Można również określić parametr **-i** w wierszu komend, aby natychmiast zatrzymać agenta. Gdy agent został zatrzymany, nie można używać tego agenta do przesyłania plików do czasu jego zrestartowania.

Zanim rozpocznie

Aby sprawdzić nazwy agentów powiązanych z menedżerem kolejek, można wyświetlić listę agentów za pomocą IBM MQ Explorer lub wiersza komend. Patrz sekcja [fteListAgents](#).

O tym zadaniu

Informacje na temat zatrzymywania agenta z poziomu wiersza komend zawiera sekcja [fteStopAgent](#).

Jeśli agent został skonfigurowany do uruchamiania jako usługa Windows, uruchomienie komendy **fteStopAgent** spowoduje również zatrzymanie usługi Windows. Alternatywnie można zatrzymać agenta, zatrzymując usługę za pomocą narzędzia Windows Services. Aby uzyskać więcej informacji, patrz ["Uruchamianie agenta MFT jako usługi Windows" na stronie 205](#).

Zatrzymywanie agenta MFT w systemie z/OS

Jeśli program Managed File Transfer Agent jest uruchamiany w systemie z/OS jako uruchomione zadanie z JCL, agent akceptuje komendy operatora z/OS **MODIFY** i **STOP**, oprócz komendy **fteStopAgent**.

Uruchomione zadanie jest używane, ponieważ jest uruchamiane z określonym identyfikatorem użytkownika i nie ma wpływu na wylogowanie użytkowników.

Uwaga: Uruchomione zadania są zwykle uruchamiane przez użytkownika administracyjnego, który może nie mieć uprawnień do logowania, a więc nie jest możliwe zalogowanie się do systemu z/OS jako użytkownik, w którym działa agent. Komendy **fteStartAgent**, **fteStopAgent**, **fteSetAgentTraceLevel** i komendy **fteShowAgentDetails** z podanym parametrem **-d** nie mogą być wydawane dla tego agenta.

W przypadku produktów IBM MQ 9.0.2 i IBM MQ 9.0.0 Fix Pack 1 można użyć właściwości **adminGroup** agenta z agentami Managed File Transfer w systemie z/OS. Można zdefiniować grupę menedżerów zabezpieczeń, na przykład MFTADMIN, a następnie dodać do tej grupy identyfikator użytkownika uruchomionego zadania i identyfikatory TSO TSO. Zmodyfikuj plik właściwości agenta i ustaw właściwość **adminGroup** tak, aby była ona nazwą tej grupy menedżerów zabezpieczeń.

```
adminGroup=MFTADMIN
```

Członkowie tej grupy mogą następnie wprowadzić komendy **fteStartAgent**, **fteStopAgent** i **fteSetAgentTraceLevel** oraz komendę **fteShowAgentDetails** z parametrem **-d** określonym dla agenta, który jest uruchomiony jako zadanie uruchomione.

Więcej informacji na ten temat zawiera opis właściwości **adminGroup** w pliku [Plik MFT agent.properties](#).

Zamknięcie kontrolowanego agenta za pomocą komendy z/OS MODIFY (F)

Komenda **MODIFY** umożliwia zatrzymanie agenta w kontrolowany sposób jako alternatywa dla komendy **fteStopAgent**. Agent wykonuje wszystkie transfery, które są obecnie w toku, ale agent nie uruchamia żadnych nowych operacji przesyłania.

Na przykład:

```
F job_name,APPL=STOP
```

gdzie *nazwa_zadania* to zadanie, w którym działa proces agenta.

Natychmiastowe zamknięcie agenta za pomocą komendy z/OS STOP (P)

Komenda **STOP** jest równoznaczna z natychmiastowym zatrzymaniem, używając komendy **fteStopAgent** z parametrem **-i**. Agent jest zatrzymany natychmiast nawet wtedy, gdy agent obecnie przesyła plik.

Na przykład:

```
P job_name
```

gdzie *nazwa_zadania* to zadanie, w którym działa proces agenta.

Odsyłacze pokrewne

“Uruchamianie agenta MFT w systemie z/OS” na stronie 208

W systemie z/OS, oprócz uruchamiania komendy **fteStartAgent** z sesji z/OS UNIX System Services, można uruchomić agenta jako uruchomione zadanie z JCL bez potrzeby sesji interaktywnej.

[Plik MFT agent.properties](#)

Rozpoczynanie nowego przesyłania plików

Istnieje możliwość uruchomienia nowego przesyłania plików z poziomu IBM MQ Explorer lub z wiersza komend. Można również wybrać przesyłanie pojedynczego pliku lub wielu plików w grupie.

O tym zadaniu

Aby uruchomić nowe przesyłanie plików z poziomu wiersza komend, należy zapoznać się z [komendą przesyłania komendy fteCreate](#).

Aby uruchomić nowy transfer plików za pomocą kreatora **Tworzenie nowego zarządzanego przesyłania plików** w programie IBM MQ Explorer, wykonaj następujące kroki:

Procedura

1. W widoku Navigator kliknij opcję **Managed File Transfer**(Przesyłanie plików zarządzanych). **Centrala przesyłania plików zarządzanych** jest wyświetlana w widoku zawartości.
2. Wszystkie menedżery kolejek koordynacji są wyświetlane w widoku Navigator. Rozwiń nazwę menedżera kolejek koordynacji, dla którego zarejestrowano agenta, dla którego ma być przesyłane przesyłanie. Jeśli aktualnie nawiązano połączenie z menedżerem kolejek koordynacji innym niż ten, który ma być używany do przesyłania, należy kliknąć prawym przyciskiem myszy nazwę menedżera kolejek koordynacji w widoku Navigator i kliknąć opcję **Rozłącz**. Następnie kliknij prawym przyciskiem myszy nazwę menedżera kolejek koordynacji, który ma być używany, i kliknij opcję **Połącz**.
3. Uruchom kreator **Tworzenie nowego zarządzanego przesyłania plików** przy użyciu jednej z następujących metod:
 - a) Kliknij prawym przyciskiem myszy nazwę dowolnego z następujących węzłów w widoku Navigator: odpowiedni menedżer kolejek koordynacji, **Szablony przesyłania**, **Dziennik przesyłania** lub **Oczekujące operacje przesyłania**. Następnie kliknij przycisk **Nowe przesyłanie**, aby uruchomić kreator.

- b) Kliknij opcję **Plik > Nowy > Inne > Kreatory przesyłania plików zarządzanych > Kreator nowego przesyłania**.
4. Postępuj zgodnie z instrukcjami wyświetlanym na panelach kreatora. Dla każdego panelu dostępna jest również pomoc kontekstowa. Aby uzyskać dostęp do pomocy kontekstowej w systemie Windows, naciśnij klawisz F1. W systemie Linux naciśnij klawisz Ctrl+F1 lub Shift+F1.

Korzystanie z plików definicji przesyłania

Istnieje możliwość określenia pliku definicji przesyłania, który może być użyty do utworzenia przesyłania plików. Plik definicji przesyłania to plik XML, który definiuje niektóre lub wszystkie informacje wymagane do utworzenia przesyłania.

Pliki definicji przesyłania są przydatne, gdy użytkownik chce określić wiele plików źródłowych i wiele plików docelowych w jednej operacji przesyłania. Za pomocą pliku definicji przesyłania można przestać złożone przesyłanie plików. Istnieje możliwość ponownego użycia i współużytkowania pliku definicji przesyłania.

Można użyć dwóch formatów dla pliku definicji przesyłania, a oba te formaty różnią się nieznacznie, zarówno w przypadku schematu produktu FileTransfer.xsd. Schemat ten można znaleźć w katalogu samples\schema w instalacji produktu Managed File Transfer.

Obsługiwane są następujące dwa formaty plików definicji przesyłania:

- Definicja plików źródłowych i docelowych dla przesyłania. W tej definicji jako element główny używany jest element **transferSpecifications**.
- Definicja całego transferu, w tym pliki źródłowe i docelowe oraz agenty źródłowe i docelowe. W tej definicji jako element główny używany jest element **request**.
 - Pliki o tym formacie mogą być generowane z poziomu komendy **fteCreateTransfer** przy użyciu parametru **-gt**.

W poniższym przykładzie przedstawiono format pliku definicji przesyłania, który określa tylko pliki źródłowe i docelowe dla operacji przesyłania:

```
<?xml version="1.0" encoding="UTF-8"?>
<transferSpecifications xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <item checksumMethod="MD5" mode="text">
    <source recursive="false" disposition="leave">
      <file>textTransferTest.txt</file>
    </source>
    <destination type="directory" exist="overwrite">
      <file>c:\targetfiles</file>
    </destination>
  </item>
</transferSpecifications>
```

Aby przestać ten format pliku definicji przesyłania, należy określić agenty źródłowe i docelowe w wierszu komend:

```
fteCreateTransfer -sa AGENT1 -sm agent1qm -da AGENT2 -dm agent2qm -td
c:\definitions\example1.xml
```

Poniższy przykład jest formatem pliku definicji przesyłania, który określa wszystkie informacje wymagane do przesyłania:

```
<?xml version="1.0" encoding="UTF-8"?>
<request version="3.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>example.com.</hostName>
      <userID>fteuser</userID>
    </originator>
    <sourceAgent agent="AGENT1" QMgr="agent1qm"/>
    <destinationAgent agent="AGENT2" QMgr="agent2qm"/>
  </transferSet>
```

```

<item mode="binary" checksumMethod="MD5">
  <source recursive="false" disposition="leave">
    <file>c:\sourcefiles\*.jpg</file>
  </source>
  <destination type="directory" exist="error">
    <file>/targetfiles/images</file>
  </destination>
</item>
</transferSet>
</managedTransfer>
</request>

```

Plik można wygenerować w tym formacie, korzystając z parametru **-gt** w komendzie **fteCreateTransfer**. Podczas przesyłania pliku definicji przesyłania za pomocą tego formatu nie ma potrzeby określania żadnych innych elementów w wierszu komend:

```
fteCreateTransfer -td c:\definitions\example2.xml
```

Informacje o agencji źródłowym i agencji docelowym można przesłonić w wierszu komend, przechodząc do normalnych parametrów oprócz pliku definicji przesyłania. Na przykład:

```
fteCreateTransfer -da AGENT9 -dm agent9qm -td c:\definitions\example2.xml
```

W tym przykładzie użyto opcji wiersza komend w celu przesłonięcia agenta docelowego zdefiniowanego w pliku definicji przesyłania przy użyciu produktu **AGENT9** i docelowego menedżera kolejek zdefiniowanego w pliku definicji przesyłania jako **agent9qm**.

Oba opisane formaty mogą zawierać jeden lub więcej elementów < item>. Więcej informacji na temat elementu < item> można znaleźć w sekcji Format komunikatu żądania przesyłania plików. Każdy z tych elementów przesyłania definiuje źródłową i docelową parę plików z dodatkowymi atrybutami w celu kontrolowania zachowania przesyłania. Na przykład można określić następujące zachowanie:

- Informacja o tym, czy operacja przesyłania używa sumy kontrolnej
- Określa, czy przesyłanie jest tekstem, czy binarnie
- Informacja o tym, czy usunąć plik źródłowy po zakończeniu przesyłania
- Czy zastąpić plik docelowy, jeśli plik istnieje

Zaletą korzystania z plików definicji przesyłania jest to, że można określić dodatkowe opcje, które nie są dostępne w wierszu komend. Na przykład podczas wykonywania operacji przesyłania komunikatów do plików można określić atrybut groupId przy użyciu pliku definicji przesyłania. Ten atrybut określa identyfikator grupy IBM MQ komunikatów, które są odczytywane z kolejki. Inną zaletą plików definicji przesyłania jest to, że można określić różne opcje dla każdej pary plików. Na przykład można określić, czy używana jest suma kontrolna, czy też plik jest przesyłany w trybie tekstowym, czy binarnym, na podstawie pliku według plików. Jeśli używany jest wiersz komend, te same opcje mają zastosowanie do każdego pliku w operacji przesyłania.

Na przykład:

```

<item checksumMethod="none" mode="binary">
  <source disposition="leave">
    <file>c:\sourcefiles\source1.doc</file>
  </source>
  <destination type="file" exist="error">
    <file>c:\destinationfiles\destination1.doc</file>
  </destination>
</item>

<item checksumMethod="MD5" mode="text">
  <source disposition="delete">
    <file>c:\sourcefiles\source2.txt</file>
  </source>
  <destination type="file" exist="overwrite">
    <file encoding="UTF8" EOL="CRLF">c:\destinationfiles\destination2.txt</file>
  </destination>
</item>

```

```
<item checksumMethod="none" mode="text">
  <source recursive="false" disposition="leave">
    <file>c:\originfiles\source3.txt</file>
  </source>
  <destination type="file" exist="overwrite">
    <file>c:\targetfiles\destination3.txt</file>
  </destination>
</item>
```

z/OS Elementów można używać do przesyłania plików z systemu rozproszonego do systemu z/OS :

```
z/OS
<item checksumMethod="none" mode="text">
  <source recursive="false" disposition="leave">
    <file>textTransferTest.txt</file>
  </source>
  <destination type="dataset" exist="overwrite">
    <file encoding="IBM-1047">//TEXT.TRANS.TEST</file>
  </destination>
</item>
```

z/OS W tym przykładzie plik `textTransferTest.txt` jest transferowany z agenta źródłowego do zestawu danych `//TEXT.TRANS.TEST` w agencie docelowym w trybie tekstowym. Ten transfer przekształca dane źródłowe z domyślnego kodowania agenta źródłowego (nie określono atrybutu kodowania źródła) na stronę kodową: `IBM-1047`.

Tworzenie zaplanowanego przesyłania plików

Istnieje możliwość zaplanowania nowego przesyłania plików z poziomu IBM MQ Explorer lub z wiersza komend. Zaplanowane przesyłanie może zawierać pojedyncze pliki lub wiele plików w grupie. Można wykonać zaplanowaną operację przesyłania plików raz lub powtórzyć operację przesyłania wiele razy.

O tym zadaniu

Harmonogram przesyłania plików można skonfigurować w taki sposób, aby wystąpił jeden raz lub w następujących odstępach czasu:

- Co minutę
- Co godzinę
- Dzielne
- Tygodniowy
- Co miesiąc
- Raz w roku

Następnie można określić, jakie wystąpienia mają zostać zatrzymane w następujących punktach:

- O określonej godzinie i godzinie
- Po określonej liczbie wystąpień

Alternatywnie można określić, że wystąpienia będą kontynuowane na zawsze.

Jeśli zaplanowane przesyłanie jest uruchamiane codziennie o tej samej godzinie, należy użyć atrybutu **`adjustScheduleTimeForDaylightSaving`** w pliku właściwości agenta, aby dostosować czas, jaki ma miejsce w harmonogramie, gdy zegary zmieniają się. Więcej informacji na ten temat zawiera sekcja [Plik MFT agent.properties](#).

Aby utworzyć nowy zaplanowany transfer plików przy użyciu wiersza komend, należy użyć parametrów planowania (**`-tb`**, **`-ss`**, **`-oi`**, **`-of`**, **`-oci`** **`-es`**) dla komendy [fteCreateTransfer](#).

Aby utworzyć nowy zaplanowany transfer plików za pomocą kreatora **Tworzenie nowego zarządzanego przesyłania plików** w programie IBM MQ Explorer, wykonaj następujące kroki:

Procedura

1. W widoku Navigator kliknij opcję **Managed File Transfer**(Przesyłanie plików zarządzanych). **Centrala przesyłania plików zarządzanych** jest wyświetlana w widoku zawartości.
2. Wszystkie menedżery kolejek koordynacji są wyświetlane w widoku Navigator . Rozwiń nazwę menedżera kolejek koordynacji, dla którego zarejestrowano agenta, dla którego ma być przesyłane przesyłanie. Jeśli aktualnie nawiązano połączenie z menedżerem kolejek koordynacji innym niż ten, który ma być używany do przesyłania, należy kliknąć prawym przyciskiem myszy nazwę menedżera kolejek koordynacji w widoku Navigator i kliknąć opcję **Rozłącz**. Następnie kliknij prawym przyciskiem myszy nazwę menedżera kolejek koordynacji, który ma być używany, i kliknij opcję **Połącz**.
3. Uruchom kreator **Tworzenie nowego zarządzanego przesyłania plików** przy użyciu jednej z następujących metod:
 - a) Kliknij prawym przyciskiem myszy nazwę dowolnego z następujących węzłów w widoku Navigator : odpowiedni menedżer kolejek koordynacji, **Szablony przesyłania**, **Dziennik przesyłań** lub **Oczekujące operacje przesyłania**. Następnie kliknij przycisk **Nowe przesyłanie** , aby uruchomić kreator.
 - b) Kliknij opcję **Plik > Nowy > Inne > Kreatory przesyłania plików zarządzanych > Kreator nowego przesyłania** .
4. Postępuj zgodnie z instrukcjami wyświetlanym na panelach kreatora. Upewnij się, że zaznaczone jest pole wyboru **Włącz przesyłanie harmonogramu** i wprowadź szczegóły harmonogramu na karcie **Harmonogram** . Zaplanowane operacje przesyłania plików rozpoczynają się w ciągu minuty od godziny rozpoczęcia harmonogramu, jeśli nie ma żadnych problemów, które mogą mieć wpływ na transfer. Na przykład mogą wystąpić problemy z siecią lub agentem, które uniemożliwiają uruchomienie zaplanowanego przesyłania. Dla każdego panelu dostępna jest pomoc kontekstowa. Aby uzyskać dostęp do pomocy kontekstowej w systemie Windows, naciśnij klawisz F1. W systemie Linux naciśnij klawisz Ctrl+F1 lub Shift+F1.

Wyniki

Więcej informacji na temat komunikatów związanych z planowanymi transferami plików zawiera sekcja [Formaty komunikatów dziennika zaplanowanego przesyłania plików](#).

Praca z oczekującymi transferami plików

Istnieje możliwość wyświetlenia zaplanowanych operacji przesyłania plików, które oczekują na IBM MQ Explorer. W oknie **Oczekujące operacje przesyłania** wyświetlane są wszystkie oczekujące operacje przesyłania zarejestrowane w menedżerze kolejek koordynacji, z którym użytkownik jest aktualnie połączony.

O tym zadaniu


Aby wyświetlić status zaplanowanego przesyłania plików, które nie zostało jeszcze uruchomione, należy wykonać następujące czynności:

Procedura

1. Rozwiń pozycję **Managed File Transfer** (Managed File Transfer) w widoku Navigator . **Centrala przesyłania plików zarządzanych** jest wyświetlana w widoku zawartości.
2. Wszystkie menedżery kolejek koordynacji są wyświetlane w widoku Navigator . Rozwiń nazwę menedżera kolejek koordynacji, który został użyty do zaplanowanego przesyłania. Aby zmienić menedżer kolejek koordynacji, z którym nawiązano połączenie, kliknij prawym przyciskiem myszy nazwę menedżera kolejek koordynacji, który ma być używany w widoku Navigator , a następnie kliknij opcję **Połącz**.
3. Kliknij opcję **Oczekujące transfery**. W widoku Zawartość zostanie wyświetlone okno **Oczekujące operacje przesyłania** .

4. W oknie **Oczekujące operacje przesyłania** wyświetlane są następujące szczegóły dotyczące zaplanowanych operacji przesyłania plików:
- Nazwa** Numer zaplanowanego przesyłania plików. Ten numer jest przypisywany automatycznie.
 - Źródło** Nazwa agenta źródłowego.
 - Plik źródłowy** Nazwa pliku, który ma zostać przesyłany w jego systemie hosta.
 - Cel** Nazwa agenta docelowego.
 - Plik docelowy** Nazwa pliku po przestaniu go do systemu docelowego.
 - Zaplanowane uruchomienie (wybrana strefa czasowa)** Godzina i godzina planowanego rozpoczęcia przesyłania plików w wybranej strefie czasowej administratora. Aby zmienić wyświetlaną strefę czasową, należy kliknąć opcję **Okna > Preferencje > IBM MQ Explorer > Przesyłanie plików zarządzanych** i wybrać alternatywną strefę czasową z listy **Strefa czasowa:** . Kliknij przycisk **OK**.
 - Powtarzaj co** Jeśli wybrano opcję powtarzania zaplanowanego przesyłania, podany przedział czasu, który ma zostać powtórzony, wyrażony jako liczba.
 - Typ powtarzania** Jeśli wybrano opcję powtarzania zaplanowanego przesyłania, typ okresu powtarzania określony dla operacji przesyłania plików. Typ może mieć jedną z następujących wartości: minutes(minuty), hours(godziny), days(dni), tygodni(tygodnie), months(miesiące) lub years(lata).
 - Powtarzaj do** Jeśli wybrano opcję powtarzania zaplanowanego przesyłania, szczegóły dotyczące tego, kiedy powtarzający się transfer pliku mają zostać zatrzymane. Na przykład określona data i godzina, lub po określonej liczbie wystąpień.

Wyniki

Aby odświeżyć elementy wyświetlane w oknie **Oczekujące operacje przesyłania** , należy kliknąć przycisk **Odśwież**  na pasku narzędzi widoku zawartości.

Aby anulować oczekujący transfer pliku, kliknij prawym przyciskiem myszy konkretny transfer i kliknij przycisk **Anuluj**. Anulowanie przesyłania powoduje całkowite usunięcie żądania przesyłania pliku.

Wyzwalanie przesyłania plików

Istnieje możliwość ustawienia określonych warunków wyzwalacza dla operacji przesyłania plików, które muszą być spełnione, zanim będzie możliwe przestanie tego transferu. Jeśli warunki wyzwalania nie są spełnione, przesyłanie plików nie odbywa się, a komunikat dziennika jest opcjonalnie wysyłany w celu zarejestrowania faktu, że przesyłanie nie miało miejsca. Żądanie przesyłania pliku jest następnie odrzucane. Na przykład można skonfigurować przesyłanie plików, które ma miejsce tylko wtedy, gdy nazwany plik w systemie, w którym znajduje się agent źródłowy, ma określoną wielkość lub jeśli dany plik o określonej nazwie istnieje w systemie, w którym znajduje się agent źródłowy. Operację przesyłania wyzwalanego pliku można skonfigurować z poziomu IBM MQ Explorer lub z wiersza komend.

O tym zadaniu

Istnieje możliwość ciągłego monitorowania zasobu, aby warunek wyzwalacza był spełniony. Więcej informacji na temat monitorowania zasobów zawiera sekcja [“Monitorowanie zasobów produktu MFT” na stronie 222](#).

Istnieją trzy różne warunki wyzwalania, które można ustawić. Warunki są następujące:

- Jeśli dany plik istnieje w tym samym systemie, co agent źródłowy
- Jeśli dany plik nie istnieje w tym samym systemie, co agent źródłowy
- Jeśli określony plik ma określoną wielkość, w systemie, w którym znajduje się agent źródłowy (wielkość może być wyrażona w bajtach, KB, MB lub GB). Te jednostki miary używają konwencji 2^{10} , na przykład 1 KB jest równe 1024 bajtom, a 1 MB jest równe 1024 kB.

Typy wyzwalania na powyższej liście mogą być łączone na dwa sposoby:

- W przypadku jednego warunku można określić więcej niż jeden plik w systemie, w którym znajduje się agent źródłowy. Spowoduje to wyzwolenie operacji przesyłania, jeśli dowolny z określonych plików spełnia warunek (operator boolowski OR).
- Można określić wiele warunków. Spowoduje to wyzwolenie operacji przesyłania tylko wtedy, gdy spełnione są wszystkie warunki (operator boolowski AND).

Można również połączyć wyzwolony transfer z zaplanowanym transferem. Więcej informacji na ten temat zawiera sekcja [Tworzenie zaplanowanego przesyłania plików](#). W tym przypadku warunki wyzwacza są wartościowane w czasie, w którym harmonogram ma być uruchamiany lub dla powtarzającego się harmonogramu za każdym razem, gdy harmonogram ma być uruchamiany.

Wyzwalane transfery nie są obsługiwane przez agenty mostu protokołu.

Aby utworzyć wyzwalone przesyłanie plików za pomocą wiersza komend, należy użyć parametru **-tr** w komendzie [fteCreateTransfer](#).

Aby utworzyć zaplanowany transfer plików za pomocą kreatora **Utwórz nowy zarządzany transfer plików** w programie IBM MQ Explorer, wykonaj następujące kroki:

Procedura

1. W widoku Navigator kliknij opcję **Managed File Transfer**(Przesyłanie plików zarządzanych). **Centrala przesyłania plików zarządzanych** jest wyświetlana w widoku zawartości.
2. Wszystkie menedżery kolejek koordynacji są wyświetlane w widoku Navigator. Rozwiń nazwę menedżera kolejek koordynacji, który został użyty do zaplanowanego przesyłania. Aby zmienić menedżer kolejek koordynacji, z którym nawiązano połączenie, kliknij prawym przyciskiem myszy nazwę menedżera kolejek koordynacji, który ma być używany w widoku Navigator, a następnie kliknij opcję **Połącz**.
3. Uruchom kreator **Tworzenie nowego zarządzanego przesyłania plików** przy użyciu jednej z następujących metod:
 - a) Kliknij prawym przyciskiem myszy nazwę dowolnego z następujących węzłów w widoku Navigator: odpowiedni menedżer kolejek koordynacji, **Szablony przesyłania**, **Dziennik przesyłań** lub **Oczekujące operacje przesyłania**. Następnie kliknij przycisk **Nowe przesyłanie**, aby otworzyć kreator.
 - b) Kliknij opcję **Plik > Nowy > Inne > Kreatory przesyłania plików zarządzanych > Kreator nowego przesyłania**.
4. Postępuj zgodnie z instrukcjami wyświetlanymi na panelach kreatora. Upewnij się, że zaznaczone jest pole wyboru **Włącz wyzwalone przesyłanie** na karcie **Wyzwalacze**, a następnie wypełnij pola na tej karcie, aby skonfigurować wyzwalenie. Dla każdego panelu dostępna jest pomoc kontekstowa. Aby uzyskać dostęp do pomocy kontekstowej w systemie Windows, naciśnij klawisz F1. W systemie Linux naciśnij kombinację klawiszy **Ctrl+F1** lub **Shift+F1**.


Monitorowanie przesyłania plików, które są w toku

Można monitorować przesyłanie plików, które jest w toku, korzystając z karty **Postęp przesyłania plików zarządzanych-bieżący postęp przesyłania** w produkcie IBM MQ Explorer. Ten transfer plików może być uruchamiany z poziomu IBM MQ Explorer lub z wiersza komend. Na karcie wyświetlane są również informacje o postępie zaplanowanych operacji przesyłania w momencie rozpoczęcia zaplanowanych operacji przesyłania.

O tym zadaniu

Jeśli produkt IBM MQ Explorer ma być używany do monitorowania transferów powiązanych z menedżerem kolejek koordynacji w systemie zdalnym, należy postępować zgodnie z instrukcjami w temacie ["Konfigurowanie produktu IBM MQ Explorer do monitorowania menedżera kolejek koordynacji zdalnej"](#) na stronie 219.

Poprzednie informacje o przesyłaniu plików nie są zachowywane po zatrzymaniu i zrestartowaniu IBM MQ Explorer. Po restarcie informacje o przeszłych transferach są usuwane z karty **Postęp bieżącego**


przesyłania . Zakończone transfery można usunąć za pomocą opcji **Usuń zakończone transfery**  w dowolnym momencie, gdy program IBM MQ Explorer jest otwarty.


Procedura

Po rozpoczęciu przesyłania nowego pliku za pomocą programu IBM MQ Explorer lub wiersza komend można monitorować postęp operacji przesyłania na karcie **Postęp bieżącego przesyłania** . Dla każdego przesyłania wyświetlane są następujące informacje:

- a) **Źródło**. Nazwa agenta używanego do przesyłania pliku z systemu źródłowego.
- b) **Miejsce docelowe**. Nazwa agenta używanego do odbierania pliku w systemie docelowym.
- c) **Bieżący plik**. Nazwa aktualnie przesyłanego pliku. Część pojedynczego pliku, który został już przesłany, jest wyświetlana w B, KiB, MiB. GiB lub TiB wraz z sumą wielkości pliku w nawiasach. Jednostka miary wyświetlana jest zależna od wielkości pliku.
B to liczba bajtów na sekundę. KiB/s to kibibajtów na sekundę, gdzie 1 kibibajt jest równy 1024 bajtom. MiB/s to mebibajtów na sekundę, gdzie 1 mebibajt jest równy 1 048 576 bajtów. GiB/s to gibibajtów na sekundę, gdzie 1 gibibajt jest równy 1 073 741 824 bajty. TiB/s to tebibajtów na sekundę, gdzie 1 tebibajt jest równy 1 099 511 627 776 bajtów.
- d) **Numer pliku**. Jeśli przesyłany jest więcej niż jeden plik, ta liczba reprezentuje, jak daleko jest łączna grupa plików, jaką jest transfer.
- e) **Postęp**. Pasek postępu pokazuje, jak ukończy bieżący transfer pliku jako wartość procentową.
- f) **Stawka**. Szybkość przesyłania pliku w ciągu KiB/s (kibibajtów na sekundę, gdzie 1 kibibajt jest równy 1024 bajtom).
- g) **Uruchomiona (wybrana strefa czasowa)**. Czas uruchomienia przesyłania pliku, który jest prezentowany w wybranej strefie czasowej administratora. Aby zmienić wyświetlaną strefę czasową, należy kliknąć opcję **Okna > Preferencje > IBM MQ Explorer > Przesyłanie plików zarządzanych** i wybrać alternatywną strefę czasową z listy **Strefa czasowa:** . Kliknij przycisk **OK**.
Jeśli podczas przesyłania pliku operacja przesyłania jest wprowadzana do stanu odtwarzania, uruchamiane są aktualizacje czasu, które będą odzwierciedlały czas wznowienia przesyłania plików.

Wyniki

Ta karta regularnie odświeża informacje automatycznie, ale aby wymusić odświeżenie widoku wyświetlanego na karcie **Postęp bieżącego przesyłania** , należy kliknąć opcję **Odśwież**  na pasku narzędzi widoku zawartości.

Aby usunąć przesyłanie plików z karty **Postęp bieżącego przesyłania** , kliknij opcję **Usuń zakończone transfery**  na pasku narzędzi widoku zawartości. Kliknięcie tego przycisku powoduje usunięcie szczegółów przesyłania plików tylko z karty; nie zatrzymuje ani nie anuluje bieżącego lub zaplanowanego przesyłania.

Aby powrócić do karty **Bieżący postęp przesyłania** po jej zamknięciu, można wyświetlić kartę, klikając opcję **Okna > Pokaż widok > Inne > Inne > Postęp przesyłania plików zarządzanych-bieżący postęp przesyłania**. Kliknij przycisk **OK**.

Co dalej

Ponadto możliwe jest tworzenie aplikacji na potrzeby monitorowania niestandardowego przesyłania plików. Można to osiągnąć, tworząc subskrypcję dla odpowiedniego tematu administracyjnego produktu Managed File Transfer (programowo lub administracyjnie), a aplikacja monitorującego może następnie odbierać publikacje dotyczące działania przesyłania plików produktu Managed File Transfer na ten temat. Więcej informacji na temat tematu subskrypcji i formatu komunikatu publikacji można znaleźć w sekcji [Przykłady komunikatów dotyczących postępu przesyłania plików](#).

Zadania pokrewne

[“Konfigurowanie produktu IBM MQ Explorer do monitorowania menedżera kolejek koordynacji zdalnej” na stronie 219](#)

Program IBM MQ Explorer służy do monitorowania transferów plików powiązanych z menedżerem kolejek koordynacji działającym w systemie zdalnym. W produkcie IBM WebSphere MQ 7.5 lub nowszym wymagany jest system, który może uruchomić serwer IBM MQ Explorer. Aby można było nawiązać połączenie ze zdalnym menedżerem kolejek koordynacji, musi być zainstalowany komponent IBM MQ Explorer.

[“Wyświetlanie statusu przesyłania plików w dzienniku przesyłania” na stronie 220](#)

Szczegóły przesyłania plików można wyświetlić za pomocą opcji **Dziennik przesyłania** w produkcie IBM MQ Explorer. Mogą to być transfery uruchomione z wiersza komend lub z IBM MQ Explorer. Można również dostosować to, co jest wyświetlane w oknie **Dziennik przesyłania**.

Konfigurowanie produktu IBM MQ Explorer do monitorowania menedżera kolejek koordynacji zdalnej

Program IBM MQ Explorer służy do monitorowania transferów plików powiązanych z menedżerem kolejek koordynacji działającym w systemie zdalnym. W produkcie IBM WebSphere MQ 7.5 lub nowszym wymagany jest system, który może uruchomić serwer IBM MQ Explorer. Aby można było nawiązać połączenie ze zdalnym menedżerem kolejek koordynacji, musi być zainstalowany komponent IBM MQ Explorer.

O tym zadaniu

Założenia: uprawnienie do nawiązywania połączenia ze zdalnym menedżerem kolejek koordynacji przez skonfigurowanie menedżera kolejek w taki sposób, aby zezwalał na połączenia zdalne.

Więcej informacji na temat sposobu konfigurowania zawiera sekcja [Nawiązywanie połączenia z menedżerem kolejek w trybie klienta z uwierzytelnianiem kanału i Zarządzanie uprawnieniami do zasobów specyficznych dla produktu MFT](#).

Aby monitorować menedżery kolejek i przesyłanie plików między agentami w systemie, w którym nie jest uruchomiony produkt Windows lub Linux, należy skonfigurować program IBM MQ Explorer do połączenia z systemem zdalnym za pomocą następujących kroków:

Procedura

1. Uruchom lokalny IBM MQ Explorer.
2. Po załadowaniu produktu IBM MQ Explorer kliknij prawym przyciskiem myszy folder **Managed File Transfer** (Managed File Transfer) i wybierz opcję **Nowa konfiguracja** (New configuration).
3. Następnie należy przejść przez kreator, wybrać menedżer kolejek koordynacji i komend, a następnie zdefiniować nazwę dla konfiguracji.
4. Kliknij przycisk **Zakończ**, aby zakończyć definiowanie.
5. Po zakończeniu definiowania kliknij prawym przyciskiem myszy definicję, a następnie wybierz opcję **Połącz**.

Wyniki

Teraz należy uruchomić produkt IBM MQ Explorer i użyć go do monitorowania aktywności przesyłania dla sieci Managed File Transfer powiązanej z menedżerem kolejek koordynacji.

Zadania pokrewne

[“Monitorowanie przesyłania plików, które są w toku” na stronie 217](#)

Można monitorować przesyłanie plików, które jest w toku, korzystając z karty **Postęp przesyłania plików zarządzanych-bieżący postęp przesyłania** w produkcie IBM MQ Explorer. Ten transfer plików może być uruchamiany z poziomu IBM MQ Explorer lub z wiersza komend. Na karcie wyświetlane są również

informacje o postępie zaplanowanych operacji przesyłania w momencie rozpoczęcia zaplanowanych operacji przesyłania.


“Wyświetlanie statusu przesyłania plików w dzienniku przesyłania” na stronie 220

Szczegóły przesyłania plików można wyświetlić za pomocą opcji **Dziennik przesyłania** w produkcie IBM MQ Explorer. Mogą to być transfery uruchomione z wiersza komend lub z IBM MQ Explorer. Można również dostosować to, co jest wyświetlane w oknie **Dziennik przesyłania**.

Wyświetlanie statusu przesyłania plików w dzienniku przesyłania

Szczegóły przesyłania plików można wyświetlić za pomocą opcji **Dziennik przesyłania** w produkcie IBM MQ Explorer. Mogą to być transfery uruchomione z wiersza komend lub z IBM MQ Explorer. Można również dostosować to, co jest wyświetlane w oknie **Dziennik przesyłania**.



Procedura

1. Rozwiń pozycję **Managed File Transfer** (Managed File Transfer) w widoku Navigator , a następnie rozwiń nazwę menedżera kolejek koordynacji, dla którego ma zostać wyświetlony dziennik przesyłania.
2. Kliknij opcję **Prześlij dziennik** w widoku Navigator . **Dziennik przesyłania** jest wyświetlany w widoku Zawartość.
3. W oknie **Dziennik przesyłania** wyświetlane są następujące informacje na temat przesyłania plików:
 - a) **Źródło** Nazwa agenta w systemie, w którym znajduje się plik źródłowy.
 - b) **Miejsce docelowe** Nazwa agenta w systemie, do którego ma zostać przesunięty plik.
 - c) **Stan zakończenia** Status przesyłania plików. Stan może mieć jedną z następujących wartości: Uruchomiony, W toku, Powodzenie, Częściowo powodzenie, Anulowane lub Niepowodzenie.
 - d) **Właściciel** Identyfikator użytkownika na hoście, który złożył żądanie przesyłania.
 - e) **Uruchomiona (wybrana strefa czasowa)** Data i godzina przyjęcia żądania przesyłania plików przez agenta Managed File Transfer , które zostało przedstawione w wybranej strefie czasowej administratora. Aby zmienić wyświetlaną strefę czasową, należy kliknąć opcję **Okna > Preferencje > IBM MQ Explorer > Managed File Transfer** , a następnie wybrać alternatywną strefę czasową z listy **Strefa czasowa:** . Kliknij przycisk **OK**.
 - f) **Stan zarejestrowany (wybrana strefa czasowa)** (ta kolumna nie jest wyświetlana domyślnie). Aby wyświetlić kolumnę, należy użyć okna **Skonfiguruj kolumny dziennika przesyłania**  . Data i godzina zarejestrowania stanu zakończenia, w strefie czasowej wybranej przez administratora.
 - g) **Nazwa zadania** Identyfikator określony przez użytkownika przy użyciu parametru **-jn** produktu **fteCreateTransfer** lub w skrypcie Ant .
 - h) **Identyfikator przesyłania** Unikalny identyfikator przesyłania plików.
 - i) **Connect: Direct** Szczegóły dotyczące **Numer procesu, Nazwa procesu, Węzeł podstawowy, Węzeł dodatkowy, Typ źródła i Typ miejsca docelowego** są wyświetlane.

Wyniki

Uwaga: Wewnętrzny format dziennika przesyłania został zmieniony w IBM MQ 8.0.0 Fix Pack 1 dla raportu APAR IC99545. W rezultacie, jeśli produkt IBM MQ Explorer zostanie zaktualizowany do wersji V8.0.0.1 lub nowszej, a następnie zostanie odtworzony do wersji V8.0.0.0, nie zostanie wyświetlony żaden kod XML kontroli dla operacji przesyłania, które miały miejsce, gdy produkt IBM MQ Explorer był w wersji V8.0.0.1. Panel XML w oknie **Właściwości** dla tych transferów będzie zawierał puste pole tekstowe.

Aby wyświetlić dalsze szczegóły na temat zakończonego transferu, rozwiń pozycję, którą interesują Cię, klikając znak plus (+). Następnie można zobaczyć wszystkie nazwy plików źródłowych i docelowych zawarte w tym przesyłaniu. Jeśli jednak przesyłanie jest obecnie w toku i składa się z wielu plików, można wyświetlić tylko te pliki, które zostały już przesłane do tej pory.

Aby odświeżyć elementy wyświetlane w oknie **Dziennik przesyłania**, należy kliknąć przycisk **Odśwież**  na pasku narzędzi widoku zawartości. Informacje o przesyłaniu plików znajdujące się w dzienniku przesyłania pozostają w dzienniku po zatrzymaniu i zrestartowaniu IBM MQ Explorer. Jeśli chcesz usunąć wszystkie zakończone operacje przesyłania plików z dziennika, kliknij opcję **Usuń zakończone transfery**  na pasku narzędzi widoku zawartości.

Aby usunąć pojedyncze zakończone przesyłanie plików z dziennika, kliknij je prawym przyciskiem myszy, a następnie kliknij opcję **Usuń**. Jeśli operacja przesyłania zostanie usunięta, nie zostanie ona zatrzymana lub anulowana, która jest w toku lub została zaplanowana; usuwane są tylko zapisane dane historyczne.

Aby skopiować unikalny identyfikator przesyłania do schowka, kliknij ten transfer prawym przyciskiem myszy, a następnie kliknij opcję **Kopiuj identyfikator**.

Metadane i pełny kod XML kontroli dla operacji przesyłania są dostępne w menu podręcznym, w ramach działania **Właściwości**.

Zadania pokrewne

[“Monitorowanie przesyłania plików, które są w toku” na stronie 217](#)

Można monitorować przesyłanie plików, które jest w toku, korzystając z karty **Postęp przesyłania plików zarządzanych-bieżący postęp przesyłania** w produkcie IBM MQ Explorer. Ten transfer plików może być uruchamiany z poziomu IBM MQ Explorer lub z wiersza komend. Na karcie wyświetlane są również informacje o postępie zaplanowanych operacji przesyłania w momencie rozpoczęcia zaplanowanych operacji przesyłania.

[“Konfigurowanie dziennika przesyłania” na stronie 221](#)

Użytkownik może skonfigurować informacje wyświetlane oraz sposób wyświetlania informacji w oknie **Dziennik przesyłania** w produkcie IBM MQ Explorer.

[“Ustawianie limitu czasu na odzyskiwanie wstrzymanych transferów” na stronie 308](#)

Można ustawić limit czasu odzyskiwania transferu dla wstrzymanych transferów plików, który ma zastosowanie do wszystkich transferów dla agenta źródłowego. Można również ustawić limit czasu odzyskiwania transferu dla pojedynczego przesyłania. Jeśli zostanie ustawiony określony czas (w sekundach), podczas którego agent źródłowy próbuje odzyskać wstrzymane przesyłanie plików, a operacja przesyłania nie powiedzie się, gdy agent osiągnie limit czasu, przesyłanie nie powiedzie się.


Konfigurowanie dziennika przesyłania

Użytkownik może skonfigurować informacje wyświetlane oraz sposób wyświetlania informacji w oknie **Dziennik przesyłania** w produkcie IBM MQ Explorer.


O tym zadaniu

Aby zmienić kolejność kolumn w polu **Dziennik przesyłania**, należy kliknąć tytuł kolumny, która ma zostać przeniesiona, a następnie przeciągnąć kolumnę do nowej pozycji. Nowa kolejność kolumn jest zachowywana tylko do czasu następnego zatrzymania i zrestartowania serwera IBM MQ Explorer.

Aby filtrować wpisy w polu **Dziennik przesyłania**, wprowadź łańcuch w polu **Filtruj wyświetlone pozycje dziennika**. Aby odtworzyć wszystkie pozycje w dzienniku, usuń łańcuch wprowadzony z pola. W tym polu można użyć dowolnego poprawnego wyrażenia regularnego Java. Więcej informacji na ten temat zawiera sekcja [Wyrażenia regularne używane przez MFT](#).

Aby dostosować, które kolumny są wyświetlane w dzienniku przesyłania, należy użyć opcji **Skonfiguruj kolumny dziennika przesyłania** . Wykonaj następujące czynności, aby rozpocząć i użyć okna **Konfigurowanie kolumn dziennika przesyłania**.

Procedura

1. Upewnij się, że w widoku Zawartość jest otwarty **Dziennik przesyłania**. Kliknij opcję **Konfiguruj kolumny dziennika przesyłania**  na pasku narzędzi widoku zawartości. Zostanie otwarte okno **Konfigurowanie kolumn dziennika przesyłania**.

2. Aby dostosować widok **Dziennik przesyłania**, zaznacz lub usuń zaznaczenie poszczególnych pól wyboru dla kolumn, które mają być wyświetlane lub ukryte. Można kliknąć opcję **Wybierz wszystko**, a następnie przycisk **OK**, aby zaznaczyć wszystkie pola wyboru, lub opcję **Anuluj wybór wszystkiego**, a następnie przycisk **OK**, aby wyczyścić wszystkie pola wyboru.

Zadania pokrewne

“Monitorowanie przesyłania plików, które są w toku” na stronie 217

Można monitorować przesyłanie plików, które jest w toku, korzystając z karty **Postęp przesyłania plików zarządzanych-bieżący postęp przesyłania** w produkcie IBM MQ Explorer. Ten transfer plików może być uruchamiany z poziomu IBM MQ Explorer lub z wiersza komend. Na karcie wyświetlane są również informacje o postępie zaplanowanych operacji przesyłania w momencie rozpoczęcia zaplanowanych operacji przesyłania.

“Wyświetlanie statusu przesyłania plików w dzienniku przesyłania” na stronie 220

Szczegóły przesyłania plików można wyświetlić za pomocą opcji **Dziennik przesyłania** w produkcie IBM MQ Explorer. Mogą to być transfery uruchomione z wiersza komend lub z IBM MQ Explorer. Można również dostosować to, co jest wyświetlane w oknie **Dziennik przesyłania**.

Monitorowanie zasobów produktu MFT

Zasoby produktu Managed File Transfer można monitorować, na przykład w kolejce lub w katalogu. Gdy warunek dotyczący tego zasobu jest spełniony, monitor zasobów uruchamia zadanie, takie jak przesyłanie plików. Monitor zasobów można utworzyć za pomocą komendy **fteCreateMonitor** lub widoku **Monitory** w module dodatkowym Managed File Transfer dla produktu IBM MQ Explorer.

O tym zadaniu

Monitorowanie zasobów produktu Managed File Transfer używa następującej terminologii:

Monitor zasobów

Monitor zasobów jest przetwarzanym procesem, który odpytuje zasób (taki jak katalog lub kolejka) w predefiniowanym odstępie czasu, aby sprawdzić, czy treść zasobu została zmieniona. Jeśli mają one, zawartość jest porównywana z zestawem warunków dla tego monitora. Jeśli istnieje zgodność, zadanie dla tego monitora zostanie uruchomione.

Zasób

Zasób systemowy, który monitor zasobów bada w każdym odstępie czasu odpytywania, który ma być porównywany z warunkami wyzwacza. Monitorowane zasoby mogą być kolejkami, katalogami lub zagnieżdżonymi strukturami katalogów.

Warunek warunku i wyzwacza

Warunek jest wyrażeniem, które jest wartościowane (zwykle jest to niezgodne z treścią monitorowanego zasobu). Jeśli wynikiem wartościowania wyrażenia jest wartość true, warunek przyczynia się do ogólnego warunku wyzwacza.

Warunek wyzwacza jest ogólnym warunkiem, który jest spełniony, gdy spełnione są wszystkie warunki. Gdy warunek wyzwacza jest spełniony, zadanie może być kontynuowane.

Zadanie

Zadanie to operacja, która jest uruchamiana, gdy spełniony jest warunek wyzwacza lub zestaw warunków. Obsługiwane zadania to przesyłanie plików i wywołanie komendy.

Plik wyzwacza

Plik wyzwacza to plik, który jest umieszczany w katalogu monitorowanym w celu wskazania, że może się rozpocząć zadanie (zwykle jest to przeniesienie). Na przykład może to oznaczać, że wszystkie pliki do przetworzenia dotarły do znanej lokalizacji i mogą zostać przesłane lub w inny sposób działają na nie. Nazwa pliku wyzwacza może być użyta do określenia plików, które mają zostać przesłane przy użyciu podstawiania zmiennych. Więcej informacji na ten temat zawiera [“Dostosowywanie zadań monitora zasobów produktu MFT przy użyciu podstawiania zmiennych”](#) na stronie 234.

Plik wyzwacza jest również znany jako gotowy plik lub plik go. Jednak w tej dokumentacji zwykle określa się go jako plik wyzwacza.

Monitorowanie zasobów nie jest obsługiwane przez agenty mostu protokołu ani agenty mostu Connect:Direct.

Pojęcia dotyczące monitorowania zasobów produktu MFT

Przegląd kluczowych pojęć związanych z funkcją monitorowania zasobów produktu Managed File Transfer .

Monitory zasobów

Monitor zasobów tworzy się za pomocą komendy **fteCreateMonitor** , która tworzy i uruchamia nowy monitor zasobów z poziomu wiersza komend. Monitor zasobów jest powiązany z agentem Managed File Transfer i jest aktywny tylko wtedy, gdy ten agent jest uruchomiony. Gdy agent monitorowania zostanie zatrzymany, monitor zasobów będzie monitorowany. Jeśli agent jest już uruchomiony po utworzeniu monitora zasobów, monitor zasobów zostanie uruchomiony natychmiast. Agent monitorowania musi być również agentem źródłowym czynności, która jest inicjowana przez monitor zasobów.

Nazwy monitorów zasobów muszą być unikalne w ramach swojego agenta. Nazwa monitora zasobów musi mieć co najmniej jeden znak długości i nie może zawierać znaków gwiazdki (*), procentu (%) ani znaku zapytania (?). Przypadek, w którym podana jest nazwa monitora zasobów, jest ignorowany, a nazwa monitora zasobów jest przekształcana na wielkie litery. Jeśli zostanie podjęta próba utworzenia monitora zasobów o nazwie, która już istnieje, żądanie zostanie zignorowane, a próba zostanie zaprotokołowana w temacie dziennika monitora zasobów.

Uwaga: Nie można utworzyć monitora zasobów z definicją zadania, która zawiera zaplanowane operacje przesyłania.

V 9.2.2 W przypadku systemu Long Term Supporti dla Continuous Delivery przed IBM MQ 9.2.2 jedynym sposobem zatrzymania monitora zasobów jest zatrzymanie agenta, który uruchomił operację monitorowania. Aby zrestartować monitor zasobów, należy ponownie uruchomić agenta. Produkt IBM MQ 9.2.2 umożliwia uruchamianie i zatrzymywanie monitorów zasobów bez konieczności zatrzymywania lub restartowania agenta. Aby uzyskać więcej informacji, zapoznaj się z sekcją: [“Uruchamianie i zatrzymywanie monitorów zasobów” na stronie 226.](#)

Nie ma ograniczeń dotyczących liczby monitorów zasobów, które można utworzyć na agencie, a wszystkie są uruchamiane z tym samym priorytetem. Należy wziąć pod uwagę wpływ nakładających się zasobów monitorowanych, powodujących konflikty warunków wyzwalaczy i częstotliwości odpytywania zasobów.

Nakładające się monitory zasobów mogą powodować:

- Możliwy rywalizacja o źródłową lokalizację/elementy.
- Możliwe zduplikowane żądania przesyłania dla tych samych elementów źródłowych.
- Nieoczekiwane błędy lub niepowodzenia związane z przesyłaniem z powodu konfliktów elementów źródłowych.

Jeśli wiele monitorów skanuje to samo położenie i może wyzwać te same elementy, można potencjalnie zakończyć się problemem dwóch różnych monitorów, które wysyłają zarządzane żądania przesyłania dla tego samego elementu.

Monitory zasobów przyjrzą się zawartości zasobów po każdym okresie odpytywania. Zawartość zasobu jest porównywana z warunkami wyzwalania, a jeśli warunki te są spełnione, wywołane jest zadanie powiązane z monitorem zasobów.

Zadanie jest uruchamiane asynchronicznie. Jeśli wystąpi zgodność warunku i zadanie zostanie uruchomione, monitor zasobów będzie kontynuował odpytywanie o dalsze zmiany w treści zasobu. Na przykład, jeśli dopasowanie wystąpiło, ponieważ plik o nazwie `reports.go` dotarł do katalogu monitorowanego, zadanie zostanie uruchomione jeden raz. W następnym okresie odpytywania, nawet jeśli plik nadal istnieje, zadanie nie zostanie uruchomione ponownie. Jeśli jednak plik zostanie usunięty, a następnie ponownie umieszczony w katalogu lub plik zostanie zaktualizowany (taki, że atrybut daty ostatniej modyfikacji zostanie zmieniony), to następne sprawdzenie warunku wyzwalacza spowoduje ponowne wywołanie zadania.

Przed IBM MQ 9.1.5, jeśli monitor zasobów wykonuje odpytywanie, które trwa dłużej niż przedział czasu odpytywania, oznacza to, że następne odpytywanie rozpoczyna się od momentu zakończenia bieżącego, bez przerwy między, co może mieć wpływ na to, jak szybko monitory zasobów mogą wprowadzać pracę do agenta. Może to spowodować problemy z wydajnością, jeśli elementy, które znajdują się podczas pierwszego odpytywania, są nadal dostępne w momencie, gdy ma miejsce druga.

V 9.2.0 Z poziomu produktu IBM MQ 9.1.5 monitor zasobów korzysta z usługi ScheduledExecutori inicjuje następne odpytywanie dopiero po zakończeniu poprzedniego odpytywania i skonfigurowanym czasie odstępu czasu odpytywania. Oznacza to, że między przedziałami odpytywania zawsze będzie taka luka, a nie inna odpytywanie rozpoczynanie od razu po poprzednim odpytywaniu, jeśli czas odpytywania był dłuższy niż odstęp czasu odpytywania.

V 9.2.0 Jeśli z programu IBM MQ 9.1.3 nie powiodło się przesłanie pliku, można skasować historię monitora zasobów, co pozwala na przesłanie innego żądania transferu bez konieczności usunięcia pliku i umieszczenia go w katalogu ponownie lub zaktualizowanie pliku w celu zmiany jego atrybutu daty ostatniej modyfikacji. Czyszczenie historii jest przydatne, na przykład w sytuacjach, w których istnieje potrzeba przekazania pliku, ale modyfikacja pliku nie jest możliwa. Więcej informacji na ten temat zawiera sekcja [“Czyszczenie historii monitora zasobów”](#) na stronie 251.

Zasoby

Monitory zasobów w produkcie Managed File Transfer mogą odpytywać zawartość następujących dwóch typów zasobów:

Katalogi lub zagnieżdżone struktury katalogów

Typowym scenariuszem jest monitorowanie katalogu w celu uzyskania pliku wyzwalacza. Aplikacja zewnętrzna może przetwarzać wiele plików i umieszczać je w znanym katalogu źródłowym. Gdy aplikacja zakończy przetwarzanie, oznacza to, że pliki są gotowe do przesłania lub w inny sposób zostały wykonane, umieszczając plik wyzwalacza w monitorowanym położeniu. Plik wyzwalacza może zostać wykryty przez monitor zasobów Managed File Transfer i zainicjowane przesłanie tych plików z katalogu źródłowego do innego Managed File Transfer Agent .

Domyślnie określony katalog jest monitorowany. Aby sprawdzić również, podkatalogi ustawiają poziom rekurencji w komendzie **fteCreateTransfer** .

Poniżej przedstawiono dwa przykłady monitorowania katalogu:

- Monitoruj plik wyzwalacza (na przykład `trigger.file`), a następnie przesyłaj znak wieloznaczny (na przykład `*.zip`).
- Monitoruj `*.zip` , a następnie przesyłaj `${FilePath}` (na przykład plik, który wyzwolił transfer). Więcej informacji na temat podstawiania zmiennych zawiera sekcja [“Dostosowywanie zadań monitora zasobów produktu MFT przy użyciu podstawiania zmiennych”](#) na stronie 234.

Uwaga: Nie należy tworzyć monitora, który monitoruje `*.zip` , a następnie przesyła `*.zip` . Monitor próbuje uruchomić przesłanie pliku `*.zip` dla każdego pliku `.zip` w systemie. Oznacza to, że monitor generuje * liczbę transferów dla `*.zip` .

Przykład tworzenia monitora zasobów w celu monitorowania katalogu można znaleźć w sekcji [“Monitorowanie katalogu i używanie podstawiania zmiennych”](#) na stronie 232.

Kolejki produktu IBM MQ

Przykładem monitorowania kolejki jest generowanie komunikatów przez aplikację zewnętrzną i umieszczanie ich w znanej kolejce o tym samym identyfikatorze grupy. Gdy aplikacja zakończy wprowadzanie komunikatów do kolejki, oznacza to, że grupa została zakończona. Pełna grupa komunikatów może zostać wykryta przez monitor zasobów Managed File Transfer i zainicjowana jest operacja przesłania grupy komunikatów z kolejki źródłowej do pliku. Przykład tworzenia monitora zasobów w celu monitorowania kolejki można znaleźć w sekcji [“Przykład: konfigurowanie zasobu MFT”](#) na stronie 233.

Uwaga: Dla każdej kolejki można określić tylko jeden monitor. Jeśli użytkownik określi więcej niż jeden monitor w celu odpytywania kolejki produktu IBM MQ , wystąpi nieprzewidywalne zachowanie.

Monitorowanie zestawów danych nie jest obsługiwane.

Warunki i warunki wyzwalania

Warunek jest spełniony, jeśli zasób zawiera wartość, która jest zgodna z innym łańcuchem lub wzorcem. Warunki mogą mieć jedną z następujących wartości:

- Zgodność z nazwą pliku (wzorzec)
- Brak zgodności z nazwą pliku (wzorzec)
- Wielkość pliku
- Zgodność, jeśli wielkość pliku pozostaje taka sama dla liczby odpytywań

Zgodność nazwy pliku może być wyrażona jako:

- Dokładna zgodność łańcucha
- Prosty znak wieloznaczny zgodny z opisem w sekcji [Korzystanie ze znaków wieloznacznych w produkcie MFT](#)
- Zgodność wyrażenia regularnego

Nazwy plików mogą być również wykluczane z dopasowywania nazw plików przy użyciu znaku wieloznacznego lub wyrażenia regularnego Java, które identyfikuje nazwy plików, które nigdy nie są zgodne.

Po wykryciu zgodnego pliku jego ostatni zmodyfikowany znacznik czasu jest zachowywany. Jeśli kolejne ankiety wykryją, że plik został zmieniony, warunek wyzwalacza ponownie zostanie spełniony, a zadanie zostanie uruchomione. Jeśli warunek ma być wykryty, gdy plik nie istnieje, jeśli żaden plik w monitorowanym katalogu nie jest zgodny z wzorcem nazwy pliku, zadanie zostanie uruchomione. Jeśli plik zostanie następnie dodany do katalogu, który jest zgodny z wzorcem nazwy pliku, zadanie zostanie uruchomione tylko wtedy, gdy plik zostanie usunięty.

Zadania

Produkt Managed File Transfer obsługuje następujące dwa typy zadań, które można skonfigurować do uruchamiania przez monitory zasobów:

Zadania przesyłania plików

Zadania przesyłania plików są definiowane w taki sam sposób, jak inne przesyłanie plików. Przydatnym sposobem generowania kodu XML zadania wymaganego przez monitor jest uruchomienie komendy `fteCreateTransfer` z parametrem `-gt`. Ta komenda generuje definicję zadania jako dokument XML, w tym specyfikację przesyłania. Następnie należy przekazać nazwę dokumentu XML zadania jako wartość parametru `-mt` w komendzie `fteCreateMonitor`. Po uruchomieniu programu `fteCreateMonitor` odczytuje on dokument XML zadania. Po uruchomieniu programu `fteCreateMonitor` wszystkie zmiany dokonane w pliku XML zadania nie są używane przez monitor.

W przypadku korzystania z zadania przesyłania plików można wybrać, ile warunków wyzwalacza jest wsadowych dla zadania. Domyślnie dla jednego warunku wyzwalacza można uruchomić jedno zadanie. Można uruchomić komendę `fteCreateMonitor` z opcją `-bs`, aby wybrać liczbę warunków wyzwalacza, które są połączone razem w jedno zadanie.

Zadania komend

Zadania komend mogą uruchamiać skrypty programu Ant, wywoływać programy wykonywalne lub uruchamiać zadania JCL. Więcej informacji na ten temat zawiera sekcja ["Konfigurowanie zadań monitora programu MFT w celu uruchamiania komend i skryptów"](#) na stronie 227.

Pliki wyzwalaczy

Za pomocą zawartości pliku wyzwalacza w monitorze zasobów można zdefiniować zestaw plików do przestania w pojedynczym żądaniu transferu. Za każdym razem, gdy wykryty zostanie zgodny plik wyzwalacza, jego zawartość jest analizowana dla ścieżek do plików źródłowych i opcjonalnie dla docelowych ścieżek plików. Te ścieżki do plików są następnie używane do definiowania elementów plików

w określonym pliku XML przesyłania zadań, który jest wprowadzany jako pojedyncze żądanie przesyłania do agenta. Definicja monitora zasobów określa, czy treść wyzwalacza jest włączona.

Format każdego pliku wyzwalacza jest pojedynczą ścieżką do pliku, który ma zostać przesunięty w każdej linii tekstu. Domyślnym formatem wiersza jest pojedyncza ścieżka do pliku źródłowego lub ścieżka do pliku źródłowego i docelowego oddzielona przecinkiem.

Więcej informacji i przykładów można znaleźć w sekcji [“Korzystanie z pliku wyzwalacza”](#) na stronie 242.

Uruchamianie i zatrzymywanie monitorów zasobów

W przypadku systemu Long Term Supporti dla Continuous Delivery przed IBM MQ 9.2.2 jedynym sposobem zatrzymania monitora zasobów jest zatrzymanie agenta, który uruchomił operację monitorowania. Aby zrestartować monitor zasobów, należy ponownie uruchomić agenta. Więcej informacji na ten temat zawierają sekcje [“Uruchamianie agenta MFT”](#) na stronie 204 i [“Zatrzymywanie agenta MFT”](#) na stronie 210.

V 9.2.2 Program IBM MQ 9.2.2 umożliwia uruchamianie i zatrzymywanie monitorów zasobów bez konieczności zatrzymywania lub restartowania agenta za pomocą komend **fteStartMonitor** i **fteStopMonitor**. Jest to przydatne, na przykład, w następujących sytuacjach:

- Jeśli agent ma wiele monitorów zasobów, a tylko niektóre z nich napotkały błędy, ale pozostałe monitory zasobów nadal działają poprawnie, należy po prostu zrestartować monitory zasobów, które nie powiodły się.
- Aby zatrzymać monitor zasobów w celu wykonania niektórych prac konserwacyjnych lub jeśli monitor zasobów nie jest wymagany przez określony czas, a użytkownik nie chce, aby działał niepotrzebnie, zużywa w ten sposób cenne zasoby systemowe.

V 9.2.2 Więcej informacji na ten temat zawiera sekcja [Uruchamianie monitora zasobów produktu MFT](#) i [Zatrzymywanie monitora zasobów produktu MFT](#).

V 9.2.2

<i>Tabela 9. Zachowanie monitora zasobów w zależności od tego, która komenda jest uruchomiona</i>	
Komenda	Zachowanie monitora zasobów
V 9.2.2 fteStartMonitor	Jeśli agent jest uruchomiony, monitor zasobów jest uruchamiany, jeśli jest aktualnie zatrzymany.
V 9.2.2 fteStopMonitor	Jeśli agent jest uruchomiony, monitor zasobów jest zatrzymany, jeśli jest aktualnie uruchomiony.
fteStartAgent	Monitor zasobów jest uruchamiany jako część agenta uruchamianego, niezależnie od wcześniejszych wywołań programu fteStopMonitor .
fteStopAgent	Wszystkie działające monitory zasobów są zatrzymane.

Tworzenie i odtwarzanie kopii zapasowych monitorów zasobów

Użytkownik może utworzyć kopię zapasową monitorów zasobów, które zostały już zdefiniowane, aby można było ich ponownie użyć w przyszłości. Dostępne są różne opcje, których można użyć w następujący sposób:

- Komenda **fteCreateMonitor** z parametrem **-ox** służy do eksportowania konfiguracji monitora zasobów do pliku XML, a przy użyciu parametru **-ix** do odtworzenia monitora zasobów przez zaimportowanie konfiguracji monitora zasobów z pliku XML.

- Aby wyeksportować definicję pojedynczego monitora zasobów do pliku XML, należy użyć komendy **fteListMonitors z -ox**.
- Użyj komendy **fteListMonitors z -od**, aby wyeksportować wiele definicji monitora zasobów do określonego katalogu. Każda definicja monitora zasobów jest zapisywana w oddzielnym pliku XML. Można również użyć opcji **-od**, aby wyeksportować pojedynczą definicję monitora zasobów do określonego katalogu.

Więcej informacji na ten temat zawiera [“Tworzenie i odtwarzanie kopii zapasowych monitorów zasobów MFT” na stronie 249](#).

Rejestrowanie monitora zasobów

W produkcie IBM MQ 9.1.0 produkt Managed File Transfer obejmuje rejestrowanie monitora zasobów. Więcej informacji na ten temat zawiera sekcja [“Rejestrowanie monitorów zasobów MFT” na stronie 245](#).

Pojęcia pokrewne

[“Dostosowywanie zadań monitora zasobów produktu MFT przy użyciu podstawiania zmiennych” na stronie 234](#)

Gdy warunki wyzwalacza aktywnego monitora zasobów są spełnione, wywoływane jest zdefiniowane zadanie. Oprócz wywołania zadania przesyłania lub komendy z tym samym agentem docelowym lub tą samą nazwą pliku docelowego za każdym razem, można także zmodyfikować definicję zadania w czasie wykonywania. W tym celu należy wstawiać nazwy zmiennych do pliku XML definicji zadania. Gdy monitor stwierdzi, że warunki wyzwalacza są spełnione i że definicja zadania zawiera nazwy zmiennych, zastępuje nazwy zmiennych wartościami zmiennych, a następnie wywołuje zadanie.

Zadania pokrewne

[“Konfigurowanie zadań monitora programu MFT w celu uruchamiania komend i skryptów” na stronie 227](#)
 Monitory zasobów nie ograniczają się do wykonywania operacji przesyłania plików jako powiązane z nimi zadanie. Można również skonfigurować monitor tak, aby wywołał inne komendy z agenta monitorowania, w tym programy wykonywalne, skrypty Ant lub zadania JCL. Aby wywołać komendy, należy zmodyfikować plik XML definicji zadania monitorowania w taki sposób, aby zawierał jeden lub więcej elementów komendy z odpowiednimi parametrami wywołania komendy, takimi jak argumenty i właściwości.

[“Przykład: konfigurowanie zasobu MFT” na stronie 233](#)

Kolejkę IBM MQ można określić jako zasób, który ma być monitorowany przez monitor zasobów za pomocą parametru **-mq** za pomocą komendy **fteCreateMonitor**.

[“Monitorowanie kolejki i używanie podstawiania zmiennych” na stronie 239](#)

Za pomocą komendy **fteCreateMonitor** można monitorować kolejkę i przysyłać komunikaty z monitorowanej kolejki do pliku. Wartość dowolnej właściwości komunikatu IBM MQ w pierwszym komunikacie, która ma zostać odczyta z monitorowanej kolejki, może zostać podstawiona w definicji XML zadania i użyta do zdefiniowania zachowania przesyłania.

Odsyłacze pokrewne

fteCreateMonitor: tworzenie monitora zasobów MFT

Monitory fteListMonitors: lista monitorów zasobów MFT

fteDeleteMonitor: usuwanie monitora zasobów produktu MFT

Konfigurowanie zadań monitora programu MFT w celu uruchamiania komend i skryptów

Monitory zasobów nie ograniczają się do wykonywania operacji przesyłania plików jako powiązane z nimi zadanie. Można również skonfigurować monitor tak, aby wywołał inne komendy z agenta monitorowania, w tym programy wykonywalne, skrypty Ant lub zadania JCL. Aby wywołać komendy, należy zmodyfikować plik XML definicji zadania monitorowania w taki sposób, aby zawierał jeden lub więcej elementów komendy z odpowiednimi parametrami wywołania komendy, takimi jak argumenty i właściwości.

O tym zadaniu

Ścieżka do pliku wykonywalnego programu, skryptu Ant lub zadania JCL, które ma wywoływać agent monitorowania, musi być dołączona do `commandPath` agenta monitorowania. Informacje na temat właściwości ścieżki komendy można znaleźć w sekcji [commandPath MFT property](#).

Dokument XML definicji czynności można utworzyć w jeden z następujących sposobów:

- Utwórz dokument XML definicji czynności ręcznie zgodnie ze schematem produktu `FileTransfer.xsd`.
- Użyj wygenerowanego dokumentu XML jako podstawy dla definicji zadania.

Niezależnie od tego, czy zadanie przesyłania ma być wykonane, czy też zadanie komendy, definicja zadania musi rozpoczynać się od elementu głównego `<request>`. Element potomny produktu `<request>` musi mieć wartość `<managedTransfer>` lub `<managedCall>`. Zwykle należy wybrać opcję `<managedCall>`, gdy istnieje jedna komenda lub skrypt do uruchomienia, oraz `<managedTransfer>`, jeśli zadanie ma obejmować przesyłanie plików i opcjonalnie do czterech wywołań komend.

Procedura

- Aby ręcznie utworzyć dokument XML definicji czynności zgodnie ze schematem produktu `FileTransfer.xsd`, należy zapoznać się z dokumentem [“Ręczne tworzenie definicji XML definicji czynności zgodnie ze schematem”](#) na stronie 228.
- Aby utworzyć definicję zadania, modyfikując wygenerowany dokument, dokonaj edycji dokumentu XML wygenerowanego przez parametr `fteCreateTransfer -gt`. Więcej informacji na ten temat zawiera sekcja [“Tworzenie dokumentu definicji czynności przez modyfikację wygenerowanego dokumentu”](#) na stronie 230.

Ręczne tworzenie definicji XML definicji czynności zgodnie ze schematem

Plik XML definicji czynności można utworzyć ręcznie zgodnie ze schematem `FileTransfer.xsd`.

O tym zadaniu

Schemat `FileTransfer.xsd` można znaleźć w `MQ_INSTALLATION_PATH/mqft/samples/schema`. Więcej informacji na temat tego schematu znajduje się w sekcji [Format komunikatu żądania przesyłania plików](#).

Przykład

W poniższym przykładzie przedstawiono przykładowy dokument XML definicji zadania zapisany jako `cleanuptask.xml`, który używa elementu `<managedCall>` do wywołania skryptu Ant o nazwie `RunCleanup.xml`. Skrypt `RunCleanup.xml` Ant musi znajdować się w katalogu `commandPath` agenta monitorowania.

```
<?xml version="1.0" encoding="UTF-8"?>
<request version="4.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedCall>
    <originator>
      <hostName>hostName</hostName>
      <userID>userID</userID>
      <mqmdUserID>mqmdUserID</mqmdUserID>
    </originator>
    <agent QMgr="QM1" agent="AGENT1"/>
    <reply QMGR="QM1">reply</reply>
    <transferSet priority="1">
      <metaDataSet>
        <metaData key="name1">value1</metaData>
      </metaDataSet>
      <call>
        <command name="RunCleanup.xml" type="antscript" retryCount="2"
          retryWait="30" successRC="0">
          <target>check_exists</target>
          <target>copy_to_archive</target>
          <target>rename_temps</target>
          <target>delete_files</target>
        </call>
      </transferSet>
    </managedCall>
  </request>
```

```

        <property name="trigger.filename" value="{FileName}"/>
        <property name="trigger.path" value="{FilePath}"/>
    </command>
</call>
</transferSet>
<job>
    <name>JOBCLEAN1</name>
</job>
</managedCall>
</request>

```

Element `<agent>` określa Managed File Transfer Agent , który jest skonfigurowany z nazwanym skryptem Ant na jego `commandPath`.

Struktura `<call><command>` . . . definiuje plik wykonywalny lub skrypt, który ma zostać uruchomiony. Komenda pobiera opcjonalny atrybut `type` , który może mieć jedną z następujących wartości:

Poprzednik

Uruchom skrypt Ant w osobnej maszynie JVM.

Wykonywalny

Wywołaj program wykonywalny.

JCL

Wywołaj zadanie JCL.

Jeśli atrybut `type` zostanie pominięty, zostanie użyta wartość domyślna wykonywalny .

Atrybut `name` określa nazwę skryptu Ant , pliku wykonywalnego lub zadania JCL, które ma zostać uruchomione bez żadnych informacji o ścieżce. Agent wyszukuje skrypt lub program w miejscach określonych za pomocą właściwości `commandPath` w pliku `agent.properties` agenta.

Atrybut `retrycount` określa liczbę prób ponownego wywołania programu, jeśli program nie zwraca kodu powrotu powodowanego przez powodzenie. Wartość przypisana do tego atrybutu nie może być ujemna. Jeśli atrybut `retrycount` nie zostanie określony, zostanie użyta wartość domyślna zero.

Atrybut `retrywait` określa czas oczekiwania (w sekundach) przed ponowną próbą wywołania programu. Wartość przypisana do tego atrybutu nie może być ujemna. Jeśli atrybut `retrywait` nie zostanie określony, zostanie użyta wartość domyślna zero.

Atrybut `successrc` jest wyrażeniem używanym do określenia, kiedy wywołanie programu powiodło się. Kod powrotu procesu dla komendy jest wartościowany przy użyciu tego wyrażenia. Wartość może składać się z jednego lub większej liczby wyrażeń w połączeniu z pionowym paskiem (`|`), oznaczając wartość boolowskim OR lub znak ampersand (`&`) oznacza, że oznaczanie boolowskie AND. Każde wyrażenie może być jednym z następujących typów wyrażeń:

- Liczba wskazująca na test równości między kodem powrotu procesu a liczbą.
- Liczba poprzedzona znakiem większym niż znak (`>`) w celu wskazania, że test jest większy niż test liczby i kodu powrotu procesu.
- Liczba poprzedzona znakiem mniejszym niż znak (`<`) w celu wskazania testu less-than między liczbą a kodem powrotu procesu.
- Liczba poprzedzona znakiem wykrzyknika (`!`) w celu wskazania, że między liczbą a kodem powrotu procesu nie jest test równy równości. Na przykład: `> 2 & < 7 &! 5 | 0 | 14` są interpretowane jako następujące kody powrotu, które są pomyślne: 0, 3, 4, 6, 14. Wszystkie pozostałe kody powrotu są interpretowane jako nieudane.

Jeśli atrybut `successrc` nie zostanie określony, zostanie użyta wartość domyślna zero. Oznacza to, że komenda jest oceniana jako pomyślnie uruchomiona, jeśli i tylko wtedy, gdy zwraca kod o wartości zero.

W przypadku skryptu Ant zwykle określone są elementy `<target>` i `<property>` . Wartości elementu `<target>` muszą być zgodne z nazwami docelowew skrypcie Ant .

W przypadku programów wykonywalnych można określić elementy programu `<argument>` . Zagnieżdżone elementy argumentów określają argumenty do przekazania do programu, który jest wywoływany jako część wywołania programu. Argumenty programu są budowane na podstawie wartości

określonych przez elementy argumentów w kolejności, w jakiej występują elementy argumentów. Jako elementy zagnieżdżone w wywołaniu programu można określić zero lub więcej elementów argumentu.

Administrator definiuje i uruchamia monitor jako normalny, korzystając z dokumentu XML definicji czynności, który zawiera element `<managedCall>`. Na przykład:

```
fteCreateMonitor -ma AGENT1 -mm QM1 -md /monitored -mn MONITOR01 -mt /tasks/cleanuptask.xml -pi 30 -pu seconds -tr match,*go
```

Ścieżka do dokumentu XML definicji przesyłania musi znajdować się w lokalnym systemie plików, z którego uruchamiana jest komenda **fteCreateMonitor** (w tym przykładzie `/tasks/cleanuptask.xml`). Dokument `cleanuptask.xml` jest używany tylko do tworzenia monitora zasobów. Wszystkie zadania, do których odwołuje się dokument produktu `cleanuptask.xml` (skrypty Ant lub zadania JCL), muszą znajdować się w ścieżce komend agenta monitorowania. Gdy warunek wyzwalacza monitora jest spełniony, wszystkie zmienne w kodzie XML definicji czynności są zastępowane rzeczywistymi wartościami z monitora. Na przykład wartość `${FilePath}` jest zastępowana w komunikacie żądania wysłanym do agenta z produktem `/monitored/cleanup.go`. Komunikat żądania jest umieszczany w kolejce komend agenta. Procesor komend wykrywa, że żądanie dotyczy wywołania programu i uruchamia określony program. Jeśli wywołana zostanie komenda typu `antscript`, uruchamiana jest nowa maszyna JVM, a zadanie Ant jest uruchamiane w nowej maszynie JVM. Więcej informacji na temat używania podstawiania zmiennych można znaleźć w sekcji [Dostosowywanie zadań przy użyciu podstawiania zmiennych](#).

Pojęcia pokrewne

[“Dostosowywanie zadań monitora zasobów produktu MFT przy użyciu podstawiania zmiennych” na stronie 234](#)

Gdy warunki wyzwalacza aktywnego monitora zasobów są spełnione, wywoływane jest zdefiniowane zadanie. Oprócz wywołania zadania przesyłania lub komendy z tym samym agentem docelowym lub tą samą nazwą pliku docelowego za każdym razem, można także zmodyfikować definicję zadania w czasie wykonywania. W tym celu należy wstawiać nazwy zmiennych do pliku XML definicji zadania. Gdy monitor stwierdzi, że warunki wyzwalacza są spełnione i że definicja zadania zawiera nazwy zmiennych, zastępuje nazwy zmiennych wartościami zmiennych, a następnie wywołuje zadanie.

Odsyłacze pokrewne

[Format komunikatu żądania przesyłania plików](#)

[Właściwość `commandPath` MFT](#)

Tworzenie dokumentu definicji czynności przez modyfikację wygenerowanego dokumentu

Dokument definicji zadania monitorowania można utworzyć, modyfikując dokument XML wygenerowany za pomocą opcji **-gt** produktu **fteCreateTransfer**.

O tym zadaniu

W wygenerowanym dokumencie znajduje się `<request>`, po którym następuje element `<managedTransfer>`. Aby przekształcić tę definicję zadania w poprawną strukturę `<managedCall>`, należy wykonać następujące czynności:

Procedura

1. Zastąp znaczniki rozpoczęcia i zakończenia produktu `<managedTransfer>` znacznikami `<managedCall>`.
2. Usuń dowolny element `<schedule>` i węzły podrzędne.
3. Zastąp znaczniki rozpoczęcia i zakończenia produktu `<sourceAgent>` znacznikiem `<agent>`, aby dopasować szczegóły konfiguracji agenta monitorowania.
4. Usuń elementy `<destinationAgent>` i `<trigger>`.
5. Usuń elementy `<item>`.

6. Usuń wszystkie elementy `preSourceCall`, `postSourceCall`, `preDestinationCall` lub `postDestinationCall`.
7. Wstaw nową strukturę `<call>...</call>` w obrębie elementu `<transferSet>`. Ta struktura zawiera definicję komendy w sposób przedstawiony w poniższym przykładzie:

```
<call>
  <command name="RunCleanup.xml" type="antscript" retryCount="2"
    retryWait="30" successRC="0">
    <target>check_exists</target>
    <target>copy_to_archive</target>
    <target>rename_temps</target>
    <target>delete_files</target>
    <property name="trigger.filename" value="{FileName}" />
    <property name="trigger.path" value="{FilePath}" />
  </command>
</call>
```

Przykład

Można również zachować element `<managedTransfer>`, w tym wszystkie szczegóły przesyłania plików, a także wstawić do czterech wywołań komend. W tym przypadku należy wstawić dowolny wybór następujących elementów wywołania między elementami `<metaDataSet>` i `<item>`:

preSourceCall

Przed rozpoczęciem przesyłania wywołaj program na agencie źródłowym.

postSourceCall

Po zakończeniu przesyłania wywołaj program na agencie źródłowym.

Wywołanie preDestinationCall

Przed rozpoczęciem przesyłania wywołaj program na agencie docelowym.

Wywołanie postDestinationCall

Po zakończeniu przesyłania wywołaj program na agencie docelowym.

Każdy z tych elementów przyjmuje strukturę elementu `<command>` zgodnie z opisem we wcześniejszym przykładzie. Schemat `FileTransfer.xsd` definiuje typy używane przez różne elementy wywołania.

W poniższym przykładzie przedstawiono właściwości `preSourceCall`, `postSourceCall`, `preDestinationCall` i `postDestinationCall` w dokumencie definicji zadania:

```
:
<transferSet priority="1">
  <metaDataSet>
    <metaData key="key1">value1</metaData>
  </metaDataSet>
  <preSourceCall>
    <command name="send.exe" retryCount="0" retryWait="0" successRC="0"
      type="executable">
      <argument>report1.pdf</argument>
      <argument>>true</argument>
    </command>
  </preSourceCall>
  <postSourceCall>
    <command name="//DO_IT.JCL" retryCount="0" retryWait="0" successRC="0"
      type="jcl">
      <argument>argument</argument>
    </command>
  </postSourceCall>
  <preDestinationCall>
    <command name="ant_script.xml" retryCount="0" retryWait="0" successRC="0"
      type="antscript">
      <target>step1</target>
      <property name="name" value="value" />
    </command>
  </preDestinationCall>
  <postDestinationCall>
    <command name="runit.cmd" retryCount="0" retryWait="0" successRC="0" />
  </postDestinationCall>
  <item checksumMethod="none" mode="binary">
:
```

Do przesyłania można mieszać różne typy komend. Argumenty, elementy docelowe i elementy właściwości są opcjonalne.

Monitorowanie katalogu i używanie podstawiania zmiennych

Katalog można monitorować za pomocą komendy **fteCreateMonitor**. Wartość zmiennej podstawianej można zastąpić definicją XML zadania, a następnie użyć jej do zdefiniowania zachowania przesyłania.

O tym zadaniu

W tym przykładzie agent źródłowy nosi nazwę AGENT_HOP. Katalog, w którym monitory AGENT_HOP mają nazwę /test/monitored. Agent odpytuje katalog co 5 minut.

Po zapisaniu pliku .zip w katalogu aplikacja, która zapisuje ten plik w katalogu, zapisuje plik wyzwalacza w tym samym katalogu. Nazwa pliku wyzwalacza jest taka sama, jak nazwa pliku .zip, ale ma inne rozszerzenie nazwy pliku. Na przykład, po zapisaniu pliku file1.zip w katalogu plik file1.go jest zapisywany w katalogu. Monitor zasobów monitoruje katalog dla plików, które są zgodne ze wzorcem *.go, a następnie używa podstawiania zmiennych do żądania przestania powiązanego pliku .zip.

Procedura

1. Utwórz plik XML zadania, który definiuje zadanie wykonywane przez monitor po jego wyzwoleniu.

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>blue.example.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_HOP" QMgr="QM_HOP" />
    <destinationAgent agent="AGENT_SKIP" QMgr="QM_SKIP" />
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <file>/test/monitored/${fileName}{token=1}{separator=.}.zip</file>
        </source>
        <destination type="file" exist="overwrite">
          <file>/out/${fileName}{token=1}{separator=.}.zip</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Zmienne, które są zastępowane wartościami powiązаныmi z plikiem wyzwalacza, są podświetlone na **pogrubienie**. Ten kod XML zadania jest zapisywany w pliku /home/USER1/task.xml

2. Utwórz monitor zasobów, aby monitorować katalog /test/monitored.

Wykonaj następującą komendę:

```
fteCreateMonitor -ma AGENT_HOP -mm QM_HOP -md /test/monitored
                 -mn myMonitor -mt /home/USER1/task.xml
                 -tr match,*.go -pi 5 -pu minutes
```

3. Użytkownik lub program zapisuje plik jump.zip do katalogu /test/monitored, a następnie zapisuje plik jump.go w katalogu.
4. Monitor jest wyzwolany przez istnienie pliku jump.go. Agent substytuuje informacje o pliku wyzwalacza w pliku XML zadania.

Wynikiem tego jest transformacja kodu XML zadania na:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
```



```

        xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
<managedTransfer>
  <originator>
    <hostName>blue.example.com</hostName>
    <userID>USER1</userID>
  </originator>
  <sourceAgent agent="AGENT_HOP" QMgr="QM_HOP" />
  <destinationAgent agent="AGENT_SKIP" QMgr="QM_SKIP" />
  <transferSet>
    <item mode="binary" checksumMethod="none">
      <source>
        <file>/test/monitored/jump.zip</file>
      </source>
      <destination type="file" exist="overwrite">
        <file>/out/jump.zip</file>
      </destination>
    </item>
  </transferSet>
</managedTransfer>
</request>

```

Wyniki

Operacja przesyłania zdefiniowana przez XML zadania jest wykonywana. Plik `jump.zip` jest odczytywany z katalogu `/test/monitored` przez `AGENT_HOP` i jest przesyłany do pliku o nazwie `/out/jump.zip`, który znajduje się w systemie, w którym działa `AGENT_SKIP`.

Pojęcia pokrewne

[“Dostosowywanie zadań monitora zasobów produktu MFT przy użyciu podstawiania zmiennych” na stronie 234](#)

Gdy warunki wyzwalacza aktywnego monitora zasobów są spełnione, wywołane jest zdefiniowane zadanie. Oprócz wywołania zadania przesyłania lub komendy z tym samym agentem docelowym lub tą samą nazwą pliku docelowego za każdym razem, można także zmodyfikować definicję zadania w czasie wykonywania. W tym celu należy wstawiać nazwy zmiennych do pliku XML definicji zadania. Gdy monitor stwierdzi, że warunki wyzwalacza są spełnione i że definicja zadania zawiera nazwy zmiennych, zastępuje nazwy zmiennych wartościami zmiennych, a następnie wywołuje zadanie.

Zadania pokrewne

[“Konfigurowanie zadań monitora programu MFT w celu uruchamiania komend i skryptów” na stronie 227](#)
 Monitory zasobów nie ograniczają się do wykonywania operacji przesyłania plików jako powiązane z nimi zadanie. Można również skonfigurować monitor tak, aby wywołał inne komendy z agenta monitorowania, w tym programy wykonywalne, skrypty Ant lub zadania JCL. Aby wywołać komendy, należy zmodyfikować plik XML definicji zadania monitorowania w taki sposób, aby zawierał jeden lub więcej elementów komendy z odpowiednimi parametrami wywołania komendy, takimi jak argumenty i właściwości.

Odsyłacze pokrewne

[fteCreateMonitor](#): tworzenie monitora zasobów MFT

Przykład: konfigurowanie zasobu MFT

Kolejkę IBM MQ można określić jako zasób, który ma być monitorowany przez monitor zasobów za pomocą parametru `-mq` za pomocą komendy **`fteCreateMonitor`**.

O tym zadaniu

W tym przykładzie zasobem, który ma być monitorowany, jest kolejka `MONITORED_QUEUE`. Ta kolejka musi znajdować się w menedżerze kolejek agenta monitorowania `QM_NEPTUNE`. Warunek, dla którego monitorowana jest kolejka, jest obecność kompletnej grupy komunikatów. Zadanie, które ma zostać wykonane, jeśli warunek jest spełniony, jest zdefiniowane w pliku `task.xml`.

Uwaga: W celu monitorowania pojedynczej kolejki nie należy tworzyć więcej niż jednego monitora zasobów. W takim przypadku występuje nieprzewidywalne zachowanie.

Procedura

Wywołaj następującą komendę:

```
fteCreateMonitor -ma AGENT_NEPTUNE -mn myMonitor -mm QM_NEPTUNE -mq MONITORED_QUEUE  
-mt task.xml -tr completeGroups -pi 5 -pu minutes
```

Monitor sprawdza kolejkę co pięć minut, aby sprawdzić, czy warunek `completeGroups` jest prawdziwy. Jeśli w kolejce znajduje się co najmniej jedna kompletna grupa, monitor uruchamia zadanie zdefiniowane w pliku `task.xml` raz dla każdej kompletnej grupy.

Dostosowywanie zadań monitora zasobów produktu MFT przy użyciu podstawiania zmiennych

Gdy warunki wyzwalacza aktywnego monitora zasobów są spełnione, wywoływane jest zdefiniowane zadanie. Oprócz wywołania zadania przesyłania lub komendy z tym samym agentem docelowym lub tą samą nazwą pliku docelowego za każdym razem, można także zmodyfikować definicję zadania w czasie wykonywania. W tym celu należy wstawiać nazwy zmiennych do pliku XML definicji zadania. Gdy monitor stwierdzi, że warunki wyzwalacza są spełnione i że definicja zadania zawiera nazwy zmiennych, zastępuje nazwy zmiennych wartościami zmiennych, a następnie wywołuje zadanie.



Ostrzeżenie: W nazwach zmiennych nie jest rozróżniana wielkość liter.

Zmienne używane do podstawiania są dostępne tylko dla dodatnich warunków wyzwalaczy. Tylko zmienne `match` i `fileSize` powodują podstawienia zmiennych, które mają być podstawiane. Jeśli zostanie użyty warunek `noMatch`, a w definicji zadania istnieją nazwy zmiennych podstawianych, to zadanie nie zostanie wywołane, a monitor zgłosi kod powrotu 110 i komunikat o błędzie BFGDM0060E.

Jeśli monitorowanym zasobem jest kolejka

Wartość dowolnej właściwości komunikatu IBM MQ w pierwszym komunikacie, która ma zostać odczyta z monitorowanej kolejki, może zostać podstawiona w definicji XML zadania.

Właściwości komunikatu zdefiniowane przez użytkownika są poprzedzane przedrostkiem `usr.`, ale nie zawierają tego przedrostka w nazwie zmiennej. Nazwy zmiennych muszą być poprzedzone znakiem dolara (\$) i ujęte w nawiasy klamrowe {}.

Na przykład wartość `${destFileName}` jest zastępowana wartością właściwości komunikatu `usr.destFileName` pierwszego komunikatu, który ma zostać odczytany z kolejki źródłowej. Więcej informacji na ten temat zawiera sekcja [Właściwości komunikatu produktuMQ odczytane przez MFT z komunikatów w kolejkach źródłowych](#) i [“Monitorowanie kolejki i używanie podstawiania zmiennych” na stronie 239](#).

Jeśli zmienna nie została zdefiniowana jako właściwość komunikatu, monitor zgłosi błąd BFGDM0060E i zwróci kod powrotu 110 (podstawienie zmiennej zadania monitora nie powiodło się). Oprócz tego agent zapisuje następujący komunikat o błędzie w swoim dzienniku zdarzeń (`outputN.log`):

```
BFGDM0113W: Trigger failure for <monitor name> for reason BFGDM0060E: A monitor task could  
not complete as a variable substitution <variable name> was not present.
```

Jeśli dla monitora włączone jest rejestrowanie w trybie umiarkowanym lub szczegółowym, monitor zapisuje następujący komunikat w dzienniku zdarzeń monitora zasobów agenta (`resmoneventN.log`):

```
BFGDM0060E: A monitor task could not complete as a variable substitution  
<variable name> was not present.
```

Więcej informacji na temat rejestrowania monitora zasobów można znaleźć w sekcji [“Rejestrowanie monitorów zasobów MFT” na stronie 245](#).

W poniższej tabeli przedstawiono zmienne podstawiane, które są udostępniane domyślnie. Na przykład `${AGENTNAME}` jest zastępowany nazwą agenta monitora zasobów.

Tabela 10. Zmienne podstawiane udostępniane domyślnie

Zmienna	Opis
nazwa_agenta	Nazwa agenta monitora zasobów.
queueName	Nazwa kolejki, która jest monitorowana.
ENCODING	Kodowanie znaków pierwszego komunikatu w kolejce lub pierwszego komunikatu w grupie.
MessageID	Identyfikator komunikatu IBM MQ pierwszego komunikatu w kolejce lub pierwszego komunikatu w grupie.
groupID	Identyfikator grupy IBM MQ grupy lub identyfikator komunikatu, jeśli zostanie znaleziony tylko jeden komunikat. Ta zmienna jest ustawiana tylko wtedy, gdy monitorowane są kompletne grupy.
CurrentTimeStempel	Znacznik czasu oparty o czas lokalny, który został wyzwolony przez monitor. Wartość znacznika czasu jest unikalna dla agenta.
CurrentTimeStampUTC	Datownik, który jest oparty na czasie, w strefie czasowej UTC, w którym monitor został wyzwolony. Wartość znacznika czasu jest unikalna dla agenta.

Jeśli monitorowanym zasobem jest katalog

W poniższej tabeli przedstawiono zestaw nazw zmiennych, które można zastąpić w definicji XML zadania.

Tabela 11. Zmienne, które mogą być podstawiane

Zmienna	Opis
FilePath	Pełna nazwa ścieżki do pliku wyzwalacza.
FileName	Część będąca nazwą pliku wyzwalacza.
Czas LastModified	Godzina ostatniej modyfikacji pliku wyzwalacza. Ten czas jest wyrażony jako czas lokalny strefy czasowej, w której agent jest uruchomiony, i jest sformatowany jako czas ISO 8601.
LastModified	Data ostatniej modyfikacji pliku wyzwalacza. Ta data jest wyrażona jako data lokalna strefy czasowej, w której agent jest uruchomiony, i jest sformatowany jako data ISO 8601.
LastModifiedTimeUTC	Godzina ostatniej modyfikacji pliku wyzwalacza. Ten czas jest wyrażony jako czas lokalny przekształcony w strefę czasową UTC i jest sformatowany jako czas ISO 8601.
LastModifiedDateUTC	Data ostatniej modyfikacji pliku wyzwalacza. Data jest wyrażona jako data lokalna przekształcona w datę UTC i ma format daty ISO 8601.
AgentName	Nazwa agenta monitora zasobów.
CurrentTimeStempel	Datownik, który jest oparty na czasie lokalnym, który został wyzwolony przez monitor. Wartość znacznika czasu jest unikalna dla agenta.
CurrentTimeStampUTC	Datownik, który jest oparty na czasie w strefie czasowej UTC, w której monitor został wyzwolony. Wartość znacznika czasu jest unikalna dla agenta.

Jeśli monitorowanym zasobem jest plik wyzwalacza

W poniższej tabeli przedstawiono zestaw nazw zmiennych, które można zastąpić, gdy monitor zasobów używa zawartości pliku wyzwalacza w celu określenia plików, które muszą zostać przestane.

Tabela 12. Zmienne, które mogą być podstawiane podczas używania pliku wyzwalacza

Zmienna	Opis
contentSource	Pełna nazwa ścieżki do pliku źródłowego.
contentDestination	Pełna nazwa ścieżki do pliku docelowego.

Nazwy zmiennych muszą być poprzedzone znakiem dolara (\$) i ujęte w nawiasy klamrowe, {}. Na przykład, `${FilePath}` jest zastępowana pełną ścieżką do pliku zgodnego wyzwalacza.

Istnieją dwa specjalne słowa kluczowe, które można zastosować do nazw zmiennych w celu zapewnienia dalszego dopracowania. Są to:

znacznik

Indeks tokenu do podstawienia (zaczynając od 1 od lewej i zaczynając od -1 od prawej)

separator

Pojedynczy znak, który umożliwi tokenizację zmiennej wartości. Wartością domyślną jest ukośnik (/) na platformach AIX and Linux lub ukośnik odwrotny (\) na platformach Windows, ale separatorem może być dowolny poprawny znak, który może pojawić się w wartości zmiennej.

Jeśli słowo kluczowe separatora jest określone w nazwie zmiennej, to wartość zmiennej jest dzielona na leksemy zgodnie z separatorem.

Wartość przypisana do słowa kluczowego znacznika jest używana jako indeks w celu wybrania znacznika, który ma zostać użyty do zastąpienia nazwy zmiennej. Indeks leksemu jest względny w stosunku do pierwszego znaku w zmiennej i rozpoczyna się od 1. Jeśli słowo kluczowe znacznika nie zostanie określone, wstawiana jest cała zmienna.

Wszystkie wartości, które są zastępowane przez nazwę agenta w kodzie XML komunikatu, są traktowane w sposób bez rozróżniania wielkości liter. Wszystkie nazwy Managed File Transfer Agent są pisane wielkimi literami. Jeśli wartość `Paris` jest podstawiana do atrybutu agenta w kodzie XML komunikatu, ta wartość jest interpretowana jako odwołanie do agenta `PARIS`.

Pojęcia pokrewne

“Przykłady: podstawianie zmiennych dla definicji monitora zasobów” na stronie 236

Przykłady podstawiania zmiennych dla definicji monitora zasobów za pomocą języka XML i produktu IBM MQ Explorer.

Zadania pokrewne

[Co zrobić, jeśli podstawianie zmiennych spowoduje, że wiele plików będzie przechowywać do jednej nazwy pliku](#)

Przykłady: podstawianie zmiennych dla definicji monitora zasobów

Przykłady podstawiania zmiennych dla definicji monitora zasobów za pomocą języka XML i produktu IBM MQ Explorer.

Przykłady ilustrujący sposób działania podstawiania zmiennych

Assuming that the file path to the matching trigger file is

`c:\MONITOR\REPORTS\Paris\Report2009.doc` on Windows and `/MONITOR/REPORTS/Paris/Report2009.doc` on AIX and Linux platforms, the variables are substituted as shown in the following table.

<i>Tabela 13. Sposób podstawiania zmiennych</i>	
Specyfikacja zmiennej	Po podstawieniu zmiennych
<code>\${FilePath}</code>	Windows :c:\MONITOR\REPORTS\Paris\Report2009.doc AIX and Linux :/MONITOR/REPORTS/Paris/Report2009.doc
<code>\${FilePath{token=1}{separator=.}}</code>	Windows :c:\MONITOR\REPORTS\Paris\Report2009 AIX and Linux :/MONITOR/REPORTS/Paris/Report2009
<code>\${FilePath{token=2}{separator=.}}</code>	Windows : doc AIX and Linux : doc
<code>\${FilePath{token=3}}</code>	Raporty Windows AIX and Linux : Paryż

Można również określić ujemny indeks znacznika, aby wybrać znaczniki względem ostatniego znaku zmiennej, jak pokazano w poniższej tabeli. The examples in the table use the same variable value, c:\MONITOR\REPORTS\Paris\Report2009.doc on Windows and /MONITOR/REPORTS/Paris/Report2009.doc on AIX and Linux.

<i>Tabela 14. Przykłady użycia ujemnego indeksu tokenów</i>	
Specyfikacja zmiennej	Po podstawieniu zmiennych
<code>\${FilePath}</code>	Windows :c:\MONITOR\REPORTS\Paris\Report2009.doc AIX and Linux :/MONITOR/REPORTS/Paris/Report2009.doc
<code>\${FilePath{token=-2}{separator=.}}</code>	Windows :c:\MONITOR\REPORTS\Paris\Report2009 AIX and Linux :/MONITOR/REPORTS/Paris/Report2009
<code>\${FilePath{token=-2}{separator=\}}</code>	Windows : Paryż AIX and Linux : Paryż
<code>\${FilePath{token=-4}}</code>	Windows : MONITOR AIX and Linux : MONITOR

Zmienne używane do podstawienia są dostępne tylko dla następujących dodatnich warunków wyzwalaczy i opcji `noSizeChange`, która jest wyjątkiem reguły warunku wyzwalacza dodatniego:

- jest zgodne
- `fileSize`
- `noSizeZmień`

Jeśli zostanie użyty warunek `noMatch`, a w definicji zadania istnieją nazwy zmiennych podstawianych, to zadanie nie zostanie wywołane, a monitor zgłosi kod powrotu 110 i komunikat o błędzie BFGDM0060E.

Przykład użycia kodu XML

Poniższy przykładowy kod XML definicji zadania używa nazwy agenta monitorowania jako agenta źródłowego dla przesyłania (Paris), używa przedostatniej nazwy katalogu w ścieżce do pliku jako nazwy agenta docelowego dla przesyłania (Report2009) i zmienia nazwę przestanego pliku, aby był głównym elementem nazwy pliku wyzwalacza z rozszerzeniem .rpt.

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="{AgentName}" QMgr="QM1" />
    <destinationAgent agent="{FilePath{token=-2}}" QMgr="QMD" />
    <transferSet>
      <item mode="binary" checksumMethod="MD5">
        <source recursive="false" disposition="leave">
          <file>c:/incoming/reports/summary/report.doc</file>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/{FileName{token=1}{separator=.}}.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Wynikiem tego jest transformacja kodu XML zadania na:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT1" QMgr="QM1" />
    <destinationAgent agent="Paris" QMgr="QMD" />
    <transferSet>
      <item mode="binary" checksumMethod="MD5">
        <source recursive="false" disposition="leave">
          <file>c:/incoming/reports/summary/report.doc</file>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/Report2009.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Zmienna `{FilePath{token=-2}}` w atrybucie agent elementu `<destinationAgent>` jest zastępowana wartością Paris. Ta wartość jest traktowana w sposób bez rozróżniania wielkości liter i interpretowana jako odwołanie do agenta PARIS.

Przykłady z wykorzystaniem IBM MQ Explorer

Podczas tworzenia monitora zasobów za pomocą IBM MQ Explorer oraz po określeniu właściwości monitora i warunków wyzwalacza, opcja ta jest podawana w celu dodania elementów przesyłania do monitora. W poniższych przykładach przedstawiono, w jaki sposób zmienne `{FilePath}` i `{FileName}` mogą być używane w panelu "**Dodaj panel elementu przesyłania**" w celu dostosowania transferów wynikających z dopasowania monitora zasobów.

Przykład 1

Aby po prostu przesłać plik źródłowy do innego miejsca, w którym spełniony jest warunek wyzwalacza, można użyć zmiennej `{FilePath}`:

- Ustaw źródło **File name** (Nazwa pliku) na wartość `${FilePath}`.
- Z menu rozwijanego **Typ** dla miejsca docelowego wybierz opcję **Katalog**.
- Ustaw jako miejsce docelowe **File name** (Nazwa pliku) położenie, do którego ma zostać przesłany plik źródłowy, na przykład `C:\MFT\out\`.

Przykład 2

W celu przesłania pliku źródłowego do innego położenia i zmiany rozszerzenia pliku, zmienna `${FileName}` może być używana w połączeniu ze zmienną `${FilePath}`:

W poniższym przykładzie założono, że ścieżka do pliku źródłowego jest równa `C:\MONITOR\REPORTS\Paris\Report2009.doc`:

- Ustaw źródło **File name** (Nazwa pliku) na wartość `${FilePath}`.
- Ustaw dla miejsca docelowego **File name** (Nazwa pliku) położenie, do którego ma zostać przesłany plik źródłowy, a następnie `${FileName}{token=1}{separator=.}`, po którym następuje nowe rozszerzenie pliku. Może to być na przykład wartość `C:\MFT\out\${FileName}{token=1}{separator=.}.rpt`, która będzie równa `C:\MFT\out\Report2009.rpt` z nazwą pliku źródłowego.

Przykład 3

Aby można było użyć części ścieżki do pliku źródłowego w celu określenia miejsca docelowego przesyłania, zmienna `${FilePath}` może być używana w połączeniu ze specyfikacjami znacznika i separatora.

W poniższym przykładzie założono, że ścieżka do pliku źródłowego jest równa `C:\MONITOR\REPORTS\Paris\Report2009.doc`.

Możliwe jest użycie części ścieżki do pliku źródłowego w celu określenia miejsca docelowego pliku. Jeśli plik ma zostać przesłany do folderu w zależności od położenia pliku źródłowego, w zależności od położenia pliku źródłowego, to znaczy `Paris` w tym przykładzie, można użyć następującego przykładu ścieżki pliku: `C:\MONITOR\REPORTS\Paris\Report2009.doc`:

- Ustaw źródło **File name** (Nazwa pliku) na wartość `${FilePath}`.
- Ustaw miejsce docelowe **File name** (Nazwa pliku), aby było miejscem docelowym, w którym znajdują się foldery dla każdego położenia, a następnie dołącz do miejsca docelowego w ścieżce i nazwę pliku. Może to być na przykład wartość `C:\MFT\out\${FilePath}{token=-2}{separator=\}\${FileName}`, która będzie równa `C:\MFT\out\Paris\Report2009.doc` z nazwą pliku źródłowego.

Pojęcia pokrewne

[“Dostosowywanie zadań monitora zasobów produktu MFT przy użyciu podstawiania zmiennych” na stronie 234](#)

Gdy warunki wyzwalacza aktywnego monitora zasobów są spełnione, wywoływane jest zdefiniowane zadanie. Oprócz wywołania zadania przesyłania lub komendy z tym samym agentem docelowym lub tą samą nazwą pliku docelowego za każdym razem, można także zmodyfikować definicję zadania w czasie wykonywania. W tym celu należy wstawiać nazwy zmiennych do pliku XML definicji zadania. Gdy monitor stwierdzi, że warunki wyzwalacza są spełnione i że definicja zadania zawiera nazwy zmiennych, zastępuje nazwy zmiennych wartościami zmiennych, a następnie wywołuje zadanie.

Zadania pokrewne

[Co zrobić, jeśli podstawianie zmiennych spowoduje, że wiele plików będzie przechowywać do jednej nazwy pliku](#)

Monitorowanie kolejki i używanie podstawiania zmiennych

Za pomocą komendy **fteCreateMonitor** można monitorować kolejkę i przysyłać komunikaty z monitorowanej kolejki do pliku. Wartość dowolnej właściwości komunikatu IBM MQ w pierwszym komunikacie, która ma zostać odczyta z monitorowanej kolejki, może zostać podstawiona w definicji XML zadania i użyta do zdefiniowania zachowania przesyłania.

O tym zadaniu

W tym przykładzie agent źródłowy jest nazywany AGENT_VENUS, który łączy się z QM_VENUS. Kolejka monitorów AGENT_VENUS nosi nazwę START_QUEUE i znajduje się w systemie QM_VENUS. Agent odpytuje kolejkę co 30 minut.

Gdy pełna grupa komunikatów jest zapisywana w kolejce, zadanie monitora wysyła grupę komunikatów do pliku w jednym z kilku agentów docelowych, z których wszystkie łączą się z menedżerem kolejek QM_MARS. The name of the file that the group of messages is transferred to is defined by the IBM MQ message property `usr.fileName` on the first message in the group. Nazwa agenta, do którego wysyłana jest grupa komunikatów, jest zdefiniowana przez właściwość komunikatu IBM MQ `usr.toAgent` w pierwszym komunikacie w grupie. Jeśli nagłówek `usr.toAgent` nie jest ustawiony, domyślną wartością, która ma być używana dla agenta docelowego, jest AGENT_MAGENTA.

Jeśli użytkownik określi opcję `useGroups="true"`, jeśli nie zostanie również określony parametr `groupId="{GROUPID}"`, zostanie on wyświetlony po pierwszym komunikacie w kolejce. Jeśli na przykład podstawianie zmiennych jest używane do generowania pliku `fileName`, możliwe jest, że zawartość pliku `a.txt` nie będzie poprawna. Dzieje się tak dlatego, że `fileName` monitor, ale w rzeczywistości zostanie wyświetlony komunikat, który nie powinien wygenerować pliku o nazwie `fileName`.

Procedura

1. Utwórz plik XML zadania, który definiuje zadanie wykonywane przez monitor po jego wyzwoleniu.

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_VENUS" QMgr="QM_VENUS" />
    <destinationAgent agent="{toAgent}" QMgr="QM_MARS" />
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <queue useGroups="true" groupId="{GROUPID}">START_QUEUE</queue>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/{fileName}.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Zmienne, które są zastępowane wartościami nagłówek komunikatów produktu IBM MQ, są podświetlone na **pogrubienie**. Ten kod XML zadania jest zapisywany w pliku `/home/USER1/task.xml`

2. Utwórz monitor zasobów, aby monitorować kolejkę START_QUEUE.

Wykonaj następującą komendę:

```
fteCreateMonitor -ma AGENT_VENUS -mm QM_VENUS -mq START_QUEUE
                 -mn myMonitor -mt /home/USER1/task.xml
                 -tr completeGroups -pi 30 -pu minutes -dv toAgent=AGENT_MAGENTA
```

3. Użytkownik lub program zapisuje grupę komunikatów do kolejki START_QUEUE.

Pierwszy komunikat w tej grupie zawiera następujące zestaw właściwości komunikatów produktu IBM MQ:

```
usr.fileName=larmer
usr.toAgent=AGENT_VIOLET
```


4. Monitor jest wyzwalany po zapisaniu kompletnej grupy. Agent substytuuje właściwości komunikatu produktu IBM MQ do pliku XML zadania.

Wynikiem tego jest transformacja kodu XML zadania na:

```
<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_VENUS" QMgr="QM_VENUS" />
    <destinationAgent agent="AGENT_VIOLET" QMgr="QM_MARS" />
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <queue useGroups="true" groupId="{GROUPID}">START_QUEUE</queue>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/larmer.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>
```

Wyniki

Operacja przesyłania zdefiniowana przez kod XML zadania jest wykonywana. Pełna grupa komunikatów odczytanych ze zmiennej `START_QUEUE` przez `AGENT_VENUS` jest zapisywana do pliku o nazwie `/reports/larmer.rpt` w systemie, w którym działa `AGENT_VIOLET`.

Co dalej

Przesyłanie poszczególnych komunikatów do osobnego pliku

Jeśli chcesz monitorować kolejkę i mieć każdy komunikat przesyłany do osobnego pliku, możesz użyć podobnej techniki do opisanej wcześniej w tym temacie.

1. Utwórz monitor zgodnie z opisem poprzednio, podając parametr **-tr completeGroups** w komendzie **fteCreateMonitor**.
2. W pliku XML zadania podaj następujące informacje:

```
<queue useGroups="true" groupId="{GROUPID}">START_QUEUE</queue>
```

Jednak po umieszczeniu komunikatów w kolejce źródłowej nie należy umieszczać ich w grupie IBM MQ. Dodaj właściwości komunikatu IBM MQ do każdego komunikatu. Na przykład można określić właściwość `usr.filename` z unikalną wartością nazwy pliku dla każdego komunikatu. Spowoduje to, że program Managed File Transfer Agent będzie traktował każdy komunikat w kolejce źródłowej jako oddzielną grupę.

Konfigurowanie zachowania ponowienia monitora dla przesyłania komunikatów do pliku

Jeśli operacja przesyłania komunikatów na plik, która jest wyzwalana przez monitor zasobów, nie powiedzie się i pozostaw grupę komunikatów, która wywołała monitor w kolejce, ten transfer jest ponownie wprowadzany w kolejnych odstępach czasu odpytywania. Liczba ponownych nadesłanych operacji przesyłania jest ograniczona przez właściwość **monitorGroupRetryLimit** agenta monitorowania.

O tym zadaniu

Za każdym razem, gdy nowe przesłanie komunikatów jest wyzwalane, dla zadania przesyłania generowany jest nowy identyfikator przesyłania.

Jeśli agent zostanie zrestartowany, monitor ponownie wyzwole przesyłanie, nawet jeśli liczba wyzwolonych operacji przesyłania przekroczyło wartość **monitorGroupRetryLimit** w pliku `agent.properties`. Wartość właściwości **monitorGroupRetryLimit** to maksymalna liczba operacji, które monitor wyzwole przesyłanie komunikatów ponownie, jeśli grupa komunikatów nadal istnieje w kolejce. Wartością domyślną tej właściwości jest 10. Wartość tej właściwości może być ustawiona na dowolną dodatnią liczbę całkowitą lub -1. Jeśli wartość -1 zostanie określona dla tej właściwości, monitor ponownie wyzwole operację przesyłania nieograniczoną liczbę razy, dopóki warunek wyzwalacza nie zostanie spełniony.

Jeśli operacja przesyłania spowoduje, że liczba wyzwolonych operacji przesyłania przekroczy wartość **monitorGroupRetryLimit**, agent zapisze błąd w dzienniku zdarzeń.

Pojedynczy komunikat jest traktowany tak, jakby był pojedynczą grupą, a przesyłanie jest wyzwalane ponownie w każdym przedziale czasu odpytywania, gdy komunikat pozostaje w kolejce, a liczba wyzwolonych operacji przesyłania jest mniejsza niż wartość **monitorGroupRetryLimit**.

Aby ustawić właściwość **monitorGroupRetryLimit** na agencie monitorowania, wykonaj następujące kroki:

Procedura

1. Zatrzymaj agenta monitorowania za pomocą komendy **fteStopAgent**.
2. Zmodyfikuj plik `agent.properties` dla agenta monitorowania, tak aby zawierał następujący wiersz:

```
monitorGroupRetryLimit=number_of_retries
```

Plik `agent.properties` znajduje się w katalogu `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/monitoring_agent_name`.

3. Uruchom agenta monitorowania za pomocą komendy **fteStartAgent**.

Zadania pokrewne

“Przykład: konfigurowanie zasobu MFT” na stronie 233

Kolejkę IBM MQ można określić jako zasób, który ma być monitorowany przez monitor zasobów za pomocą parametru **-mq** za pomocą komendy **fteCreateMonitor**.

Korzystanie z pliku wyzwalacza

Za pomocą zawartości pliku wyzwalacza w monitorze zasobów można zdefiniować zestaw plików do przestania w pojedynczym żądaniu transferu. Za każdym razem, gdy wykryty zostanie zgodny plik wyzwalacza, jego zawartość jest analizowana dla ścieżek do plików źródłowych i opcjonalnie dla docelowych ścieżek plików. Te ścieżki do plików są następnie używane do definiowania elementów plików w określonym pliku XML przesyłania zadań, który jest wprowadzany jako pojedyncze żądanie przesyłania do agenta. Definicja monitora zasobów określa, czy treść wyzwalacza jest włączona.

Wyzwalanie treści pliku można włączyć podczas tworzenia monitora, określając parametr **-tc** (treść wyzwalacza). Ten parametr **-tc** ma zastosowanie tylko do opcji wyzwalacza pliku `match` i `noSizeChange`. Więcej informacji na temat tworzenia monitora można znaleźć w sekcji **fteCreateMonitor**: tworzenie monitora zasobów produktu MFT.

W przypadku korzystania z pliku treści wyzwalacza domyślny format każdego wiersza jest następujący:

- Ścieżka do pojedynczego pliku źródłowego, lub
- Ścieżka do pliku źródłowego i ścieżka do pliku docelowego, oddzielone przecinkiem.

gdzie znaki białych znaków są obsługiwane jako część ścieżek do plików. Domyślny format linii można zmienić, podając parametry **-tcr** i **-tcc** w komendzie **fteCreateMonitor**. Więcej informacji na ten temat zawiera sekcja “Opcje zaawansowane” na stronie 244.

Po przeanalizowaniu pliku wyzwalacza zostanie wygenerowana lista ścieżek plików i zostanie ona zastosowana do podanego pliku XML zadania przesyłania. Podobnie jak w przypadku wszystkich monitorów, format XML zadania przesyłania jest kompletnym plikiem XML zadania przesyłania wygenerowanym przez komendę **fteCreateTransfer** z pojedynczym elementem lub zdefiniowanym plikiem. Pojedynczy element musi używać zmiennych podstawianych `${contentSource}` i opcjonalnie `${contentDestination}` jako zamienników dla ścieżek do plików źródłowych i docelowych. Monitor rozszerzy plik XML zadania przesyłania w taki sposób, aby zawierał element pliku dla każdego wiersza (ścieżki pliku) w pliku wyzwalacza.

Nie można użyć funkcji wyzwalającej treść pliku z parametrem **-bs**, ponieważ parametr **-tc** oznacza jedno żądanie transferu dla każdego pliku wyzwalacza.

Przykład

W poniższym przykładzie zdefiniowano monitor w celu wyzwolenia pliku, który kończy się w programie `trig`, i odczyta ścieżki do plików w tym pliku.

```
fteCreateTransfer -gt task.xml -sa SrcAgent -da DestAgent -dd /file/destdir ${contentSource}
fteCreateMonitor -mn TrigMonitor -md /home/trigdir -mt task.xml -ma SrcAgent -tr "match,*.trig"
-tc
```

Komenda **fteCreateTransfer** tworzy plik o nazwie `task.xml` dla pojedynczego pliku ze ścieżką do pliku źródłowego `${contentSource}`. Na przykład:

```
<item checksumMethod="MD5" mode="binary">
  <source disposition="leave" recursive="false">
    <file>${contentSource}</file>
  </source>
</item>
```

Komenda **fteCreateMonitor** skanuje dla plików, które kończą się w katalogu `trig` w katalogu `/home/trigdir`, i używa ich do utworzenia pojedynczego żądania transferu, które jest oparte na `task.xml` dla wszystkich ścieżek w tym pliku wyzwalacza. Format pliku wyzwalacza musi być jedną ścieżką do pliku (tylko dla źródła) w każdym wierszu, bez przecinka. Na przykład:

```
/home/file/first.txt
/home/file/second.txt
/home/different/third.txt
:
```

Wszystkie pliki są dostarczane do katalogu `/file/destdir` z nazwą pliku, a nie ścieżką do niego, to znaczy `/home/file/first.txt` jest dostarczana do produktu `/file/destdir/first.txt`.

Alternatywnie, jeśli parametr **-dd /file/destdir** zostanie zmieniony w komendzie **fteCreateTransfer** na **-df \${contentDestination}**, a format treści pliku wyzwalacza to *ścieżka do pliku źródłowego, ścieżka do pliku docelowego*, można zdefiniować różne ścieżki docelowe dla tego samego agenta docelowego. Na przykład:

```
/home/file/first.txt,/home/other/sixth.txt
```

Następnie miejsce docelowe staje się `/home/other/sixth.txt`.

Zmienne podstawiane mogą być tokenizowane. Na przykład można oddzielić część nazwy pliku od podanej ścieżki za pomocą `${contentDestination{token=-1}}`. Oznacza to, że jeśli miejsce docelowe **fteCreateTransfer** jest zdefiniowane jako **-df /file/destdir/\${contentDestination{token=-1}}**, nowym miejscem docelowym dla `/home/file/first.txt` jest `/file/destdir/sixth.txt`.

Opcje zaawansowane

Domyślny format linii dla treści pliku wyzwalacza można zmienić za pomocą parametru **-tcr** *regex* . Podaj wyrażenie regularne, które jest zgodne z wymaganym formatem wiersza i dostarcza jedną lub dwie grupy przechwytywania. Pierwsza grupa przechwytywania jest źródłem, a druga-opcjonalna, grupa przechwytywania jest miejscem docelowym. Na przykład:

- Ścieżka źródłowa i docelowa są rozdzielane łącznikiem:

```
((?:[^-]+)-((?:[^-]+)+)
```

W tym przykładzie separator jest zdefiniowany w trzech położeniach i wszystkie trzy instancje łącznika, można je zmienić na dowolny znak. Upewnij się, że nie ma żadnych specjalnych znaków.

- Ścieżki źródłowe i docelowe są rozdzielane przecinkami z spacjami kończącymi. Komentarze oznaczone znakiem # (#) są ignorowane.

```
((?:[^, ]+),((?:[^\s, ]+)+) *(?:#.*)+
```

Ścieżki do plików nie mogą zawierać znaku liczby (#). Zwykle wpis jest następujący: */home/source/from.txt,/home/destination/to.txt # some comment*.

Jeśli używany jest parametr **-tcr** , należy upewnić się, że wyrażenie regularne jest dobrze zaprojektowane i przetestowane w taki sposób, że wyrażenie może wykryć błędy i poprawnie przeanalizować pliki wyzwalaczy.

Kolejność przechwytywania można odwrócić za pomocą parametru **-tcc destSrc** . Jeśli ten parametr zostanie określony, pierwsza grupa przechwytywania będzie ścieżką do pliku docelowego, a druga-źródłowa ścieżka do pliku.

Jak obsługiwane są błędy

Pusty plik wyzwalacza

Jeśli plik wyzwalacza jest pusty, wynik nie jest transferem plików. Oznacza to, że monitor tworzy żądanie transferu, ale nie są określone żadne elementy pliku.

Plik wyzwalacza z błędami

Jeśli dowolna pozycja w pliku wyzwalacza nie zostanie przeanalizowana względem oczekiwanego formatu, nie zostanie wygenerowane żądanie transferu. Dziennik błędów monitora jest publikowany, a błąd jest rejestrowany w dzienniku zdarzeń. Plik wyzwalacza jest oznaczony jako przetworzony, a monitor nie podejmuje próby ponownego przetworzenia pliku, dopóki plik nie zostanie zaktualizowany.

Niezgodne zadanie przesyłania XML

Kod XML zadania przesyłania musi być zgodny z plikiem wyzwalacza, to znaczy, jeśli kod XML zadania przesyłania ma zarówno `${contentSource}` , jak i `${contentDestination}`, wszystkie pliki wyzwalaczy dla tego monitora muszą mieć ścieżki do plików źródłowych i docelowych oraz podobnie do odwrotnego. W pierwszym przypadku monitor zgłasza niepowodzenie podstawienia `${contentDestination}` , jeśli plik wyzwalacza dostarcza tylko ścieżkę do pliku źródłowego.

Przykłady

Poniższy przykład jest podstawowym wyzwalaczem treści, w którym zawartość pliku wyzwalacza ma tylko ścieżkę do pliku źródłowego:

```
fteCreateTransfer -gt task.xml -sa SrcAgent -da DestAgent -dd /file/destdir ${contentSource}
fteCreateMonitor -mn TrigMonitor -md /home/trigdir -mt task.xml -ma SrcAgent -tr "match,*.trig"
-tc
```

Parametr **-tcr** definiuje dwie grupy przechwytywania sekwencji dowolnych znaków, które są rozdzielone znakiem spacji. Parametr i opcja **-tcc destSrc** wskazują, że grupy przechwytywania mają być przetwarzane jako miejsce docelowe, a następnie źródło.

```
fteCreateTransfer -gt task.xml -sa SrcAgent -da DestAgent -df ${contentDestination} $
{contentSource}
fteCreateMonitor -mn TrigMonitor -md /home/trigdir -mt task.xml -ma SrcAgent -tr "match,*.trig"
-tc
    -tcr "((?:[^\ ])+) ((?:[^\ ])+)" -tcc destSrc
```

Rejestrowanie monitorów zasobów MFT

Informacje diagnostyczne na temat monitorów zasobów można uzyskać, używając funkcji rejestrowania.

O tym zadaniu

Funkcji rejestrowania dla monitorów zasobów można użyć za pomocą komendy **fteSetAgentLogLevel** lub pliku `agent.properties` w celu sterowania rejestrowaniem monitora zasobów.

Należy zauważyć, że istniejące punkty śledzenia są nadal używane do przechwytywania informacji.

Dzienniki monitora zasobów są zapisywane w pliku o nazwie `resmoneventN.log`, gdzie *N* oznacza liczbę, na przykład `resmonevent0.log`. Pliki dziennika zdarzeń rejestrują kilka działań, które mają miejsce, gdy monitor odpytuje zasób, na przykład, katalog lub kolejkę.



Ostrzeżenie: Wszystkie monitory zasobów agenta zapisują się do tego samego pliku dziennika.

Przykładowe dane wyjściowe pliku `resmoneventN.log` można znaleźć w sekcji [Co zrobić, jeśli monitor zasobów katalogu MFT nie jest wyzwalany przez pliki](#).

Poniższa tabela zawiera listę typów zdarzeń, które monitor zasobów zapisuje w pliku dziennika. Trzecia kolumna opisuje poziom rejestrowania potrzebny do przechwycenia każdego zdarzenia, dla którego najniższy poziom to INFO (INFO), a najwyższy (najwyższy), VERBOSE (szczegółowy).

Należy zauważyć, że ustawienie wyższego poziomu rejestrowania powoduje również zapisywanie zdarzeń niższego poziomu. Na przykład ustawienie poziomu rejestrowania na wartość UMIARKOWANE powoduje również zapisanie zdarzeń poziomu INFO, ale nie zdarzeń poziomu VERBOSE.

Numer	Zdarzenie	Poziom rejestrowania	Opis
1	Monitor został utworzony	INFORMACJA	Monitor zasobów został utworzony.
2	Usunięto monitor	INFORMACJA	Monitor zasobów został usunięty.
3	Monitor zatrzymany	INFORMACJA	Monitor zasobów został zatrzymany.
4	Monitor został uruchomiony	INFORMACJA	Monitor zasobów został uruchomiony.
5	Uruchom odpytywanie	INFORMACJA	Monitor zasobów uruchomił nowy cykl odpytywania.
6	Zakończ ankietę	INFORMACJA	Cykl odpytywania monitora zasobów zakończył się.
7	Dopasuj wzorzec	VERBOSE	Znaleziono plik w katalogu monitora wyzwalacza lub komunikat w kolejce, który jest zgodny z określonym wzorcem.
8	Niezgodność wzorca	VERBOSE	Znaleziono niedopasowany plik w katalogu monitora wyzwalacza lub komunikat

Numer	Zdarzenie	Poziom rejestrowania	Opis
			w kolejce, który nie jest zgodny z określonym wzorcem.
9	Żądanie przestania	INFORMACJA	Monitor zasobów zainicjował operację przesyłania.
10	Katalog jest zbyt głęboki	VERBOSE	Katalog monitorowany przez monitor zasobów zawiera więcej podkatalogów do odpytywania, niż liczba określona w konfiguracji monitora zasobów.
11	Zablokowano plik	umiarkowany	Plik wyzwalacza monitorowany przez monitor zasobów jest zablokowany przez inny proces.
12	Maća wielkość pliku	umiarkowany	Plik wyzwalacza jest mniejszy niż wielkość określona w konfiguracji monitora zasobów.
13	Niestabilna wielkość pliku	umiarkowany	Plik wyzwalacza jest zmieniany częściej, niż jest to oczekiwane przez konfigurację monitora zasobów.
14	Zbyt wiele odpytywań	umiarkowany	Monitor zasobów odpytał niestabilny plik wyzwalacza zbyt wiele razy.
15	Elementy zgodne	INFORMACJA	Łączna liczba plików wyzwalaczy znajdujących się w katalogu odpytywanym przez monitor zasobów.
16	Elementy operacji przesyłania	INFORMACJA	Łączna liczba pozycji w żądaniu transferu.
17	Wygenerowane FDC	umiarkowany	Monitor zasobów wygenerował wyjątek.
18	Żądanie przestania	INFORMACJA	Żądanie przestania wprowadzone przez monitor zasobów.
19	Uruchomienie monitora nie powiodło się	umiarkowany	Nie powiodło się uruchomienie monitora zasobów.
20	Wyczyszczono historię	INFORMACJA	Informacje o historii monitora zostały wyczyszczone.
21	Czyszczenie historii monitora nie powiodło się	INFORMACJA	Próba skasowania informacji o historii monitora nie powiodła się.
22	Identyfikator przesyłania	INFORMACJA	Identyfikator żądania przesyłania został wystany przez monitor.
23	Grupowanie	INFORMACJA	Łączna liczba żądań transferu dla dopasowanych elementów: N , gdzie N jest liczbą.

Procedura

- Aby użyć programu **fteSetAgentLogLevel** do włączenia i wyłączenia rejestrowania monitora zasobów, należy zapoznać się z opisem parametru **logMonitor** w sekcji [fteSetAgentLogLevel](#), a także przykładami sposobu użycia różnych opcji.
- Aby użyć pliku `agent.properties` do sterowania rejestrowaniem monitora zasobów, należy zapoznać się z opisem w sekcji [Plik MFT agent.properties](#), aby uzyskać opis dodatkowych właściwości, które umożliwiają przeprowadzenie następujących działań rejestrowania:

- Włączanie lub wyłączenie rejestrowania
- Ogranicz wielkość każdego pliku dziennika
- Ogranicz liczbę dzienników, które mogą być generowane przez monitory zasobów

Przykład

Poniższy przykładowy komunikat ustawia rejestrowanie na poziomie verbose dla agenta HA2, w menedżerze kolejek MFTDEMO:

```
<?xml version="1.0"?>
<log:log version="6.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xmlns:log="https://www.ibm.com/log">
  <log:originator>
    <log:request>
      <log:hostname>192.168.7.1</log:hostname>
      <log:userid>johndoe</log:userid>
    </log:request>
  </log:originator>
  <log:endpoint agent="HA2" QMgr="MFTDEMO"/>
  <log:logMonitor>MON1="verbose"</log:logMonitor>
</log:log>
```

Odsyłacze pokrewne

[Komenda fteSetAgentLogLevel](#)

[Plik MFT agent.properties](#)

V 9.2.2 Uruchamianie monitora zasobów MFT

W programie IBM MQ 9.2.2 można uruchomić monitory zasobów bez konieczności zatrzymywania lub restartowania agenta za pomocą komendy **fteStartMonitor**.

Zanim rozpoczniesz

Jeśli zarządzanie uprawnieniami użytkowników zostało włączone przez ustawienie atrybutu **authorityChecking** na wartość true w pliku `agent.properties`, do uruchomienia monitora zasobów wymagane jest uprawnienie Monitorowanie lub Monitorowanie operacji. Więcej informacji na temat zarządzania uprawnieniami użytkowników zawiera sekcja [Ograniczanie uprawnień użytkowników w działaniach agenta MFT](#).

O tym zadaniu

Komendę **fteStartMonitor** można uruchomić z dowolnego systemu, w którym jest zainstalowany komponent komend Managed File Transfer, co oznacza, że monitor zasobów można uruchomić z dowolnego miejsca i nie jest on ograniczony do systemu, w którym działa agent, który jest właścicielem monitora zasobów. Więcej informacji na temat wymaganych i opcjonalnych parametrów tej komendy zawiera sekcja [fteStartMonitor \(uruchamianie monitora zasobów produktu MFT\)](#).

Procedura

- Aby określić stan agenta przed lub po uruchomieniu komendy **fteStartMonitor**, należy użyć komendy **fteListMonitors** z parametrem **-v**, jak pokazano w poniższym przykładzie:

```
fteListMonitors -ma monitoring_agent_name -v
```

- Aby uruchomić monitor zasobów w agencie działającym na tym samym komputerze, wprowadź komendę **fteStartMonitor** w następujący sposób:

```
fteStartMonitor -mn monitor_name -ma agent_name
```

- Aby uruchomić monitor zasobów w agencie działającym na innym komputerze, wprowadź komendę **fteStartMonitor** w następujący sposób:

```
fteStartMonitor -mn monitor_name -ma agent_name -mm AgentQueueManager
```

Jeśli menedżer kolejek komend jest również menedżerem kolejek agenta dla agenta monitorowania, parametr **-mm** jest opcjonalny. W przeciwnym razie należy określić menedżer kolejek agenta za pomocą parametru **-mm**.

Wyniki

Jeśli agent jest uruchomiony, monitor zasobów jest uruchamiany, jeśli jest obecnie zatrzymany. Komenda wyświetla następujące komunikaty i rejestruje zdarzenie w pliku `output0.log` agenta.

```
BFGCL0816I: wydano żądanie uruchomienia monitora zasobów 'nazwa_monitora' agenta 'nazwa_agenta'.  
BFGCL0251I: Żądanie zostało pomyślnie zakończone.
```

Informacje o komunikatach wyprowadzanych przez komendę, jeśli nie może ona uruchomić monitora zasobów, zawiera sekcja [MonitorfteStart\(uruchamianie monitora zasobów MFT\)](#).

Pojęcia pokrewne

“Pojęcia dotyczące monitorowania zasobów produktu MFT” na stronie 223

Przegląd kluczowych pojęć związanych z funkcją monitorowania zasobów produktu Managed File Transfer.

Zadania pokrewne

“Zatrzymywanie monitora zasobów MFT” na stronie 248

W programie IBM MQ 9.2.2 można zatrzymać monitory zasobów bez konieczności zatrzymywania lub restartowania agenta za pomocą komendy **fteStopMonitor**.

Odsyłacze pokrewne

[Monitor fteStart\(uruchamianie monitora zasobów produktu MFT\)](#)

V 9.2.2 Zatrzymywanie monitora zasobów MFT

W programie IBM MQ 9.2.2 można zatrzymać monitory zasobów bez konieczności zatrzymywania lub restartowania agenta za pomocą komendy **fteStopMonitor**.

Zanim rozpocznie

Jeśli zarządzanie uprawnieniami użytkowników zostało włączone przez ustawienie atrybutu **authorityChecking** na wartość `true` w pliku `agent.properties`, do zatrzymania monitora zasobów wymagane jest uprawnienie `Monitoruj` lub `Monitoruj operacje`. Więcej informacji na temat zarządzania uprawnieniami użytkowników zawiera sekcja [Ograniczanie uprawnień użytkowników w działaniach agenta MFT](#).

O tym zadaniu

Komendę **fteStopMonitor** można uruchomić z dowolnego systemu, w którym zainstalowany jest komponent komend Managed File Transfer, co oznacza, że można zatrzymać monitor zasobów z dowolnego miejsca i nie ograniczać się do systemu, w którym działa agent, do którego należy monitor zasobów. Więcej informacji na temat wymaganych i opcjonalnych parametrów tej komendy zawiera sekcja [fteStopMonitor \(zatrzymywanie monitora zasobów produktu MFT\)](#).

Gdy monitor zasobów jest zatrzymany, zapisuje komunikat w dzienniku zdarzeń monitora zasobów agenta, `resmoneventnumber.log`. Jeśli monitor zasobów został zatrzymany za pomocą komendy **fteStopMonitor**, komunikat zawiera nazwę użytkownika, który wysłał żądanie zatrzymania:

```
Monitor zasobów został zatrzymany przez użytkownika '< muserid >'
```

Monitor zasobów jest automatycznie uruchamiany po zrestartowaniu jego agenta, nawet jeśli monitor zasobów został wcześniej zatrzymany za pomocą komendy **fteStopMonitor**.

Agenty przetwarzają żądania zatrzymania monitora szeregowo, a nie równolegle, dlatego na przykład, jeśli agent otrzyma żądanie zatrzymania monitora M1 , a następnie kolejne żądanie zatrzymania monitora M2 w krótkim odstępie czasu, zatrzymuje M1 przed próbą zatrzymania M2.

Procedura

- Aby określić stan agenta przed lub po uruchomieniu komendy **fteStopMonitor** , należy użyć komendy **fteListMonitors** z parametrem **-v** , jak pokazano w poniższym przykładzie:

```
fteListMonitors -ma monitoring_agent_name -v
```

- Aby zatrzymać monitor zasobów w agencie działającym na tym samym komputerze, wprowadź komendę **fteStopMonitor** w następujący sposób:

```
fteStopMonitor -mn monitor_name -ma agent_name
```

- Aby zatrzymać monitor zasobów w agencie działającym na innym komputerze, wprowadź komendę **fteStopMonitor** w następujący sposób:

```
fteStopMonitor -mn monitor_name -ma agent_name -mm AgentQueueManager
```

Jeśli menedżer kolejek komend jest również menedżerem kolejek agenta dla agenta monitorowania, parametr **-mm** jest opcjonalny. W przeciwnym razie należy określić menedżer kolejek agenta za pomocą parametru **-mm** .

Wyniki

Jeśli agent jest uruchomiony, monitor zasobów zostanie zatrzymany, jeśli jest uruchomiony. Komenda wyświetla następujące komunikaty i rejestruje zdarzenie w pliku `output0.log` agenta.

```
BFGCL0813I: wydano żądanie zatrzymania monitora zasobów 'MNTR' agenta 'SOURCE'.  
BFGCL0251I: Żądanie zostało pomyślnie zakończone.
```

Więcej informacji na temat komunikatów wysyłanych przez komendę, jeśli nie można zatrzymać monitora zasobów, zawiera sekcja [fteStopMonitor \(zatrzymywanie monitora zasobów MFT\)](#)

Pojęcia pokrewne

“[Pojęcia dotyczące monitorowania zasobów produktu MFT](#)” na stronie 223

Przegląd kluczowych pojęć związanych z funkcją monitorowania zasobów produktu Managed File Transfer .

Zadania pokrewne

“[Uruchamianie monitora zasobów MFT](#)” na stronie 247

W programie IBM MQ 9.2.2 można uruchomić monitory zasobów bez konieczności zatrzymywania lub restartowania agenta za pomocą komendy **fteStartMonitor** .

Odsyłacze pokrewne

[Monitor fteStop\(zatrzymaj monitor zasobów MFT\)](#)

Tworzenie i odtwarzanie kopii zapasowych monitorów zasobów MFT

Można utworzyć kopię zapasową monitorów zasobów, które mają być dostępne do użycia w przyszłości, eksportując ich definicje do pliku XML, który można następnie zaimportować w celu utworzenia nowego monitora zasobów z kopii zapasowej.

O tym zadaniu

Może być konieczne utworzenie kopii zapasowej wcześniej zdefiniowanych monitorów zasobów, aby można było ponownie wykorzystać ich definicje w przyszłości, na przykład w celu ponownego utworzenia monitorów zasobów w innej infrastrukturze lub w przypadku konieczności ponownego utworzenia monitora zasobów z powodu problemów z menedżerem kolejek.

Kopię zapasową pojedynczej definicji menedżera zasobów można utworzyć za pomocą komendy **fteCreateMonitor** lub komendy **fteListMonitors** z parametrem **-ox**. W obu przypadkach kopia zapasowa definicji menedżera zasobów jest eksportowana do pliku XML. Następnie można użyć parametru **-ix** komendy **fteCreateMonitor** do utworzenia nowego menedżera zasobów, importując definicję z pliku XML.

Za pomocą parametru **-ox** można utworzyć kopię zapasową tylko jednej definicji monitora zasobów w danym momencie.

W katalogu IBM MQ 9.1 parametr **-od** jest dodawany do komendy **fteListMonitors**. Po określeniu tego parametru można utworzyć kopię zapasową więcej niż jednego monitora zasobów jednocześnie, eksportując ich definicje masowo do określonego katalogu. Każda definicja monitora zasobów jest zapisywana w osobnym pliku XML z nazwą w formacie *agent name.monitor name.xml*.

Parametr **-od** jest szczególnie przydatny w przypadku dużej liczby monitorów zasobów, dla których ma zostać utworzona kopia zapasowa, ponieważ komenda **fteListMonitors -od** musi zostać uruchomiona tylko raz, bez konieczności uruchamiania komendy **fteListMonitors -ox** osobno dla każdej definicji zasobu, lub należy użyć oddzielnego skryptu w celu uruchomienia komendy **fteListMonitors -ox** dla każdego monitora zasobów.

Procedura

- Aby utworzyć kopię zapasową definicji jednego monitora zasobów, eksportując ją do pliku XML, należy użyć jednej z następujących komend:
 - Komenda **fteCreateMonitor** z parametrem **-ox**.
 - Komenda **fteListMonitors** z parametrem **-ox**.

Jeśli używany jest parametr **-ox**, należy również określić parametry **-ma** i **-mn**, jak pokazano w poniższym przykładzie:

```
fteListMonitors -ma AGENT1 -mn MONITOR1 -ox filename1.xml
```

- Aby utworzyć kopię zapasową wielu definicji monitorów zasobów, eksportując je do plików XML w określonym katalogu, należy użyć komendy **fteListMonitors** z parametrem **-od**, jak pokazano w poniższym przykładzie:

```
fteListMonitors -od /usr/mft/resmonbackup
```

Podczas tworzenia masowej kopii zapasowej monitorów zasobów należy podać poprawny katalog docelowy. Jeśli ścieżka docelowa nie zostanie określona, zostanie wyświetlony komunikat o błędzie, jak pokazano w poniższym przykładzie:

```
BFGCL0762E: Nie określono katalogu wyjściowego. Uruchom komendę ponownie, określając poprawną ścieżkę.
```

Parametru **-od** nie można łączyć z parametrem **-ox**, w przeciwnym razie zostanie wyświetlony następujący komunikat o błędzie:

```
BFGCL0761E: Nie jest poprawne jednoczesne określenie parametrów '-od' i '-ox'.
```

Można zdefiniować konkretny zestaw monitorów zasobów, które mają być uwzględnione w kopii zapasowej. Na przykład, używając parametru **-ma** do określenia nazwy agenta, można utworzyć kopię zapasową wszystkich monitorów zasobów dla tego agenta, jak pokazano w poniższym przykładzie:

```
fteListMonitors -ma AGENT1 -od /usr/mft/resmonbackup
```

Podczas definiowania wzorca, który ma być używany do dopasowywania nazw agentów i/lub nazw monitorów, można również użyć znaku wieloznacznego, dołączając znak gwiazdki (*). W poniższym przykładzie tworzona jest kopia zapasowa wszystkich monitorów zasobów o nazwach zgodnych z podanym wzorcem i znajdujących się w agencie o nazwie zgodnej z podanym wzorcem:

```
fteListMonitors -ma AGENT* -mn MON* -od /usr/mft/resmonbackup
```

Podczas działania komendy wyświetlane są następujące komunikaty raportu o postępie:

Łączna liczba znalezionych zgodnych definicji monitorów zasobów: *liczba* .
indeks z *number* definicji monitora zasobów zapisanych w systemie plików.

Jeśli używana jest opcja verbose, suma jest nadal wyświetlana, ale nie jest wyświetlana.

indeks z *number* definicji monitora zasobów zapisanych w systemie plików

Komenda wyświetla nazwę zapisanej definicji monitora, na przykład:

```
BFGCL0762I: Definicja monitora 'FILEMON' agenta 'XFERAGENT' zapisana jako  
FILEMON.XFERAGENT.XML do systemu plików.
```

- Aby utworzyć kopię zapasową jednego monitora zasobów dla konkretnego agenta, eksportując go do pliku XML w określonym katalogu, należy użyć komendy **fteListMonitors** z parametrem **-od** :

```
fteListMonitors -ma AGENT1 -mn MONITOR1 -od /usr/mft/resmonbackup
```

Użycie parametru **-od** do utworzenia kopii zapasowej pojedynczego monitora zasobów jest podobne do użycia parametru **-ox** , z tą różnicą, że nazwa pliku wyjściowego ma format *agent name.monitor name.xml*.

- Aby odtworzyć definicje monitora zasobów z kopii zapasowej, należy użyć komendy **fteCreateMonitor** z parametrem **-ix** , jak pokazano w poniższym przykładzie:

```
fteCreateMonitor -ix file name
```

Więcej przykładów użycia parametru **-od** zawiera temat [fteListMonitory: list MFT monitory zasobów](#).

Odsyłacze pokrewne

fteCreateMonitor: utwórz monitor zasobów MFT

[fteListMonitory](#): lista MFT monitorów zasobów

V9.2.0 Czyszczenie historii monitora zasobów

Można wyczyścić historię monitora zasobów, aby można było przestać inne żądanie przesyłania plików dla pliku, który nie został wcześniej przesłany z powodu niepowodzenia. Aby wyczyścić historię monitora zasobów, można użyć komendy **fteClearMonitorHistory** lub komendy IBM MQ Explorer.

Zanim rozpocznie

Jeśli zarządzanie uprawnieniami użytkowników zostało włączone przez ustawienie atrybutu **authorityChecking** na wartość **true** w pliku [agent.properties](#) , użytkownik, który czyści historię monitorowania, musi mieć odpowiednie uprawnienia, jak pokazano w poniższej tabeli.

Tabela 15. Uprawnienia użytkownika wymagane do uruchomienia komendy fteClearMonitorHistory		
Użytkownik czyści historię monitora	Uprawnienie dostępu MFT	Wymagane uprawnienia
Ten sam użytkownik, który utworzył monitor zasobów.	Monitor	BROWSE on SYSTEM.FTE.AUTHMON1. <nazwa_agenta_monitorowania> Jest to takie samo uprawnienie, jak uprawnienie wymagane do utworzenia lub usunięcia monitora zasobów.
Dowolny użytkownik inny niż użytkownik, który utworzył monitor zasobów.	Monitorowanie operacji	SET on SYSTEM.FTE.AUTHPS1. <nazwa_agenta> Jest to takie samo uprawnienie, jak uprawnienie wymagane do usunięcia monitora zasobów.

Więcej informacji na temat zarządzania uprawnieniami użytkowników zawiera sekcja [Ograniczanie uprawnień użytkowników w działaniach agenta MFT](#).

Jeśli użytkownik bez wymaganych uprawnień podejmie próbę wyczyszczenia historii monitora zasobów, komenda **fteClearMonitorHistory** wyświetli komunikat o błędzie i zarejestruje niepowodzenie w pliku `output0.log` agenta. Więcej informacji na ten temat zawiera sekcja [fteClearMonitorHistory: czyszczenie historii monitora zasobów](#).

O tym zadaniu

Jeśli przesyłanie plików zostało zainicjowane i nie można przestać pliku z jakiegokolwiek powodu, monitor zasobów nie wybierze tego pliku ponownie do przestania podczas następnego odpytywania, ponieważ historia monitora wskazuje, że plik był widoczny we wcześniejszym odpytywaniu i od tego czasu nie był modyfikowany (patrz sekcja [“Pojęcia dotyczące monitorowania zasobów produktu MFT”](#) na stronie 223).

Przed IBM MQ 9.1.3, jeśli przesyłanie pliku nie powiedzie się, przesyłanie pliku można zainicjować ponownie tylko wtedy, gdy plik zostanie usunięty, a następnie ponownie umieszczony w katalogu lub jeśli plik zostanie zaktualizowany w taki sposób, że atrybut daty ostatniej modyfikacji zostanie zmieniony, albo gdy monitor zasobów zostanie ponownie utworzony.

Jednak w produkcie IBM MQ 9.1.3 można wyczyścić historię monitora zasobów za pomocą komendy **fteClearMonitorHistory** lub za pomocą komendy IBM MQ Explorer. Usunięcie historii umożliwia przestanie innego żądania przesyłania dla pliku, którego przestanie nie powiodło się, bez konieczności usuwania pliku, a następnie ponownego umieszczenia go w katalogu, lub zaktualizowania pliku w celu zmiany jego atrybutu daty ostatniej modyfikacji, co jest przydatne na przykład w sytuacjach, gdy istnieje potrzeba przestania pliku, ale zmodyfikowanie pliku nie jest możliwe. Możliwość wyczyszczenia historii monitora zasobów oznacza również, że nie ma potrzeby ponownego tworzenia monitora zasobów w celu wprowadzenia innego żądania przesyłania dla pliku, którego przestanie nie powiodło się.

 Przykładowy element SCSQCFCMD dostarczany z programem Managed File Transfer w systemie z/OS zawiera skrypt JCL służący do czyszczenia historii monitora.

Procedura

- Aby użyć komendy **fteClearMonitorHistory** do wyczyszczenia historii monitora zasobów, należy wprowadzić komendę w następującym formacie:

```
fteClearMonitorHistory -p <configuration> -ma <agent name> -mn <monitor name> -w 1000
```

Wymagane są tylko parametry **-ma** i **-mn**. Wszystkie pozostałe parametry są opcjonalne. Więcej informacji na temat używania komendy **fteClearMonitorHistory**, w tym przykłady, zawiera sekcja [fteClearMonitorHistory: czyszczenie historii monitora zasobów](#).

Jeśli historia zostanie pomyślnie wyczyszczona, komenda wyświetli następujący komunikat:

```
BFGCL0780I: wydano żądanie wyczyszczenia historii monitora zasobów 'nazwa monitora' dla agenta 'nazwa agenta'.  
BFGCL0251I: Żądanie zostało pomyślnie zakończone.
```

i rejestruje powodzenie w pliku `output0.log` agenta.

Jeśli próba wyczyszczenia historii monitora zasobów nie powiedzie się, program **fteClearMonitorHistory** wyświetli komunikat o błędzie i zarejestruje niepowodzenie w pliku `output0.log` agenta.

- Aby za pomocą widoku monitora zasobów we wtyczce IBM MQ Explorer MFT wyczyścić historię monitora zasobów, należy kliknąć prawym przyciskiem myszy monitor zasobów i z menu rozwijanego wybrać opcję **Wyczyść historię**.

Jeśli historia zostanie pomyślnie wyczyszczona, zostanie wyświetlony następujący komunikat:

```
BFGUI00171: Historia monitora zasobów została pomyślnie wyczyszczona.
```

Jeśli próba wyczyszczenia historii nie powiedzie się, zostanie wyświetlony komunikat o błędzie. Na przykład:

Praca z szablonami przesyłania plików

Szablonów przesyłania plików można używać do przechowywania wspólnych ustawień przesyłania plików dla powtarzających się lub złożonych operacji przesyłania. Utwórz szablon przesyłania z wiersza komend za pomocą komendy **fteCreateTemplate** lub użyj komendy IBM MQ Explorer , aby utworzyć szablon przesyłania przy użyciu kreatora **Utwórz nowy szablon dla zarządzanego przesyłania plików** lub zapisz szablon podczas tworzenia przesyłania plików, zaznaczając pole wyboru **Zapisz ustawienia przesyłania jako szablon** . W oknie **Szablony przesyłania** wyświetlane są wszystkie szablony przesyłania utworzone w sieci produktu Managed File Transfer .

O tym zadaniu

Aby utworzyć szablon przesyłania z poziomu wiersza komend, należy użyć komendy `fteCreateTemplate` . Następnie, jeśli chcesz wysłać szablon przesyłania utworzony w wierszu komend, kliknij przycisk **Wyślij** w IBM MQ Explorer .

Aby wyświetlić szablony przesyłania w produkcie IBM MQ Explorer, wykonaj następujące kroki:

Procedura

1. Rozwiń pozycję **Managed File Transfer** (Managed File Transfer) w widoku Navigator . **Centrala przesyłania plików zarządzanych** jest wyświetlana w widoku zawartości.
2. Wszystkie menedżery kolejek koordynacji są wymienione w widoku Navigator . Rozwiń nazwę menedżera kolejek koordynacji, który został użyty do zaplanowanego przesyłania. Aby zmienić menedżer kolejek koordynacji, z którym nawiązano połączenie, kliknij prawym przyciskiem myszy nazwę menedżera kolejek koordynacji, który ma być używany w widoku Navigator , a następnie kliknij opcję **Połącz**.
3. Kliknij opcję **Szablony przesyłania**. W widoku Zawartość zostanie wyświetlone okno **Szablony przesyłania** .
4. W oknie **Szablony przesyłania** wyświetlane są następujące szczegóły dotyczące przesyłania plików:
 - a) **Nazwa** Nazwa szablonu przesyłania plików.
 - b) **Źródło** Nazwa agenta używanego do przesyłania pliku z systemu źródłowego.
 - c) **Plik źródłowy** Nazwa pliku, który ma zostać przesłany w jego systemie hosta.
Aby wyświetlić to pole, należy rozwinąć informacje o szablonie przesyłania.
 - d) **Miejsce docelowe** Nazwa agenta używanego do odbierania pliku w systemie docelowym.
 - e) **Plik docelowy** Nazwa pliku po przestaniu go do systemu docelowego.
Aby wyświetlić to pole, należy rozwinąć informacje o szablonie przesyłania.
 - f) **Zaplanowane uruchomienie (wybrana strefa czasowa)** Godzina i godzina planowanego rozpoczęcia przesyłania plików w strefie czasowej używanej przez administratora. Aby zmienić wyświetlaną strefę czasową, należy kliknąć opcję **Okna > Preferencje > IBM MQ Explorer > Managed File Transfer** , a następnie wybrać alternatywną strefę czasową z listy **Strefa czasowa:** . Kliknij przycisk **OK**.
 - g) **Zdarzenia wyzwalacza** Typ zdarzenia, które wyzwala proces przesyłania plików do uruchomienia. Typ może mieć jedną z następujących wartości: `exists`, `does not exist` lub `przekracza`.

Wyniki

Aby odświeżyć elementy wyświetlane w oknie **Szablony przesyłania** , należy kliknąć przycisk Odśwież  na pasku narzędzi widoku zawartości.

Aby przestać szablon przesyłania i rozpocząć transfer zdefiniowany w szablonie, kliknij prawym przyciskiem myszy nazwę szablonu i kliknij przycisk **Wyślij**.

Aby zmienić szablon przesyłania, kliknij prawym przyciskiem myszy nazwę szablonu i kliknij opcję **Edytuj**. Wszystkie pliki zawarte w oryginalnym szablonie są wymienione jako część grupy przesyłania, nawet jeśli nie zostały uwzględnione jako część grupy w oryginalnym szablonie. Aby usunąć plik z szablonu, należy wybrać specyfikację pliku z grupy, a następnie kliknąć opcję **Usuń wybrane**. Aby dodać nowe specyfikacje plików do szablonu, należy użyć pól na panelu szablonu i kliknąć przycisk **Dodaj do grupy**. Po dokonanej edycji zostanie wyświetlona prośba o nadanie edytowanego szablonu nowej nazwy.

Aby utworzyć przesyłanie plików z szablonu przesyłania, kliknij prawym przyciskiem myszy nazwę szablonu i kliknij opcję **Edytuj jako nowe przesyłanie**.

Aby utworzyć duplikat kopii szablonu przesyłania, kliknij prawym przyciskiem myszy nazwę szablonu i kliknij opcję **Duplikuj**. Zduplikowany szablon przesyłania jest automatycznie zapisywany o tej samej nazwie co oryginalny szablon, dołączany za pomocą opcji "(kopia)".

Aby usunąć szablon przesyłania, kliknij prawym przyciskiem myszy nazwę szablonu i kliknij opcję **Usuń**.

Zadania pokrewne

“Tworzenie szablonu przesyłania plików za pomocą IBM MQ Explorer” na stronie 254

Szablon przesyłania plików można utworzyć z poziomu produktu IBM MQ Explorer lub z poziomu wiersza komend. Następnie można użyć tego szablonu w celu utworzenia nowych operacji przesyłania plików przy użyciu szczegółów szablonu lub wysłać szablon w celu uruchomienia przesyłania plików.

Odsyłacze pokrewne

fteCreateTemplate: tworzenie nowego szablonu przesyłania plików

[Szablony fteList](#)

[Szablony fteDelete](#)

Tworzenie szablonu przesyłania plików za pomocą IBM MQ Explorer

Szablon przesyłania plików można utworzyć z poziomu produktu IBM MQ Explorer lub z poziomu wiersza komend. Następnie można użyć tego szablonu w celu utworzenia nowych operacji przesyłania plików przy użyciu szczegółów szablonu lub wysłać szablon w celu uruchomienia przesyłania plików.

O tym zadaniu

Aby utworzyć szablon przesyłania plików z poziomu wiersza komend, należy użyć komendy [fteCreateTemplate](#).

Aby utworzyć szablon przesyłania plików przy użyciu kreatora **Utwórz nowy szablon dla zarządzanego przesyłania plików** w programie IBM MQ Explorer, wykonaj następujące kroki:

Procedura

1. W widoku Navigator kliknij opcję **Managed File Transfer**(Przesyłanie plików zarządzanych). **Centrala przesyłania plików zarządzanych** jest wyświetlana w widoku zawartości.
2. Wszystkie menedżery kolejek koordynacji są wyświetlane w widoku Navigator. Rozwiń nazwę menedżera kolejek koordynacji, który został użyty do zaplanowanego przesyłania. Aby zmienić menedżer kolejek koordynacji, z którym nawiązano połączenie, kliknij prawym przyciskiem myszy nazwę menedżera kolejek koordynacji, który ma być używany w widoku Navigator, a następnie kliknij opcję **Połącz**.
3. Uruchom kreator **Utwórz nowy szablon dla produktu Managed File Transfer**, klikając prawym przyciskiem myszy opcję **Szablony przesyłania**, a następnie klikając opcję **Nowy szablon**.
4. Postępuj zgodnie z instrukcjami wyświetlanym na panelach kreatora. Dla każdego panelu dostępna jest pomoc kontekstowa. Aby uzyskać dostęp do pomocy kontekstowej w systemie Windows, naciśnij klawisz F1. W systemie Linux naciśnij klawisz Ctrl+F1 lub Shift+F1.

Jeśli został utworzony szablon zawierający wszystkie wymagane szczegóły przesyłania, należy zaznaczyć pole wyboru **Zapisz ustawienia przesyłania jako szablon** na stronie **Podsumowanie przesyłania**, jeśli to pole wyboru nie jest jeszcze zaznaczone. Należy również wprowadzić nazwę szablonu w polu Nazwa. Jeśli zostanie utworzony szablon, który nie zawiera jeszcze wszystkich

wymaganych szczegółów przesyłania, pole wyboru **Zapisz ustawienia przesyłania jako szablon** zostanie automatycznie zaznaczone.

Zadania pokrewne

“Praca z szablonami przesyłania plików” na stronie 253

Szablonów przesyłania plików można używać do przechowywania wspólnych ustawień przesyłania plików dla powtarzających się lub złożonych operacji przesyłania. Utwórz szablon przesyłania z wiersza komend za pomocą komendy **fteCreateTemplate** lub użyj komendy IBM MQ Explorer , aby utworzyć szablon przesyłania przy użyciu kreatora **Utwórz nowy szablon dla zarządzanego przesyłania plików** lub zapisz szablon podczas tworzenia przesyłania plików, zaznaczając pole wyboru **Zapisz ustawienia przesyłania jako szablon** . W oknie **Szablony przesyłania** wyświetlane są wszystkie szablony przesyłania utworzone w sieci produktu Managed File Transfer .

Odsyłacze pokrewne

fteCreateTemplate: tworzenie nowego szablonu przesyłania plików

[Szablony fteList](#)

[Szablony fteDelete](#)

Tworzenie kopii zapasowej definicji szablonu przesyłania plików

Szablony przesyłania plików zawierają dokument XML, który definiuje specyfikację pliku źródłowego i docelowego dla przesyłania. Tego pliku XML można użyć jako danych wejściowych do komendy **fteCreateTemplate** w celu ponownego utworzenia szablonu przesyłania plików.

O tym zadaniu

Aby utworzyć kopię zapasową dokumentu XML zawierającego specyfikację pliku źródłowego i docelowego dla szablonu przesyłania, należy użyć komendy [fteCreateTransfer](#) lub IBM MQ Explorer. Aby utworzyć plik kopii zapasowej w formacie XML szablonu przesyłania, wykonaj następujące kroki:

Procedura

- Metoda pierwsza: należy użyć parametru **-gt** w komendzie [fteCreateTransfer](#) w celu wygenerowania komunikatu XML szablonu przesyłania do nowego pliku.
- Metoda druga: utwórz szablon za pomocą programu IBM MQ Explorer.
Po dostaniu się do strony *Podsumowanie szablonu przesyłania* :
 - a) Skopiuj *podgląd XML komunikatu żądania*.
 - b) Zapisz ten komunikat XML szablonu przesyłania w nowym pliku.
- Metoda trzecia: użyj IBM MQ Explorer , aby utworzyć kopię zapasową istniejących szablonów.
 - a) Przejdź do sekcji **Zarządzane przesyłanie plików** > **Nazwa menedżera kolejek** > **Szablony przesyłania**.
 - b) W panelu Przesyłanie podświetl szablon, który ma zostać utworzony, kliknij prawym przyciskiem myszy i wybierz opcję **Edytuj** z menu podręcznego.
 - c) Kliknij przycisk **Dalej** , aby przejść do strony *Podsumowanie szablonu przesyłania* .
 - d) Skopiuj *podgląd XML komunikatu żądania*.
 - e) Zapisz ten komunikat XML szablonu przesyłania w nowym pliku.

Wyniki

Można użyć pliku komunikatów XML szablonu przesyłania utworzonego przy użyciu jednej z powyższych metod jako danych wejściowych dla komendy [fteCreateTemplate](#) . Szczegółowe informacje na temat korzystania z tej komendy można znaleźć w komendzie **fteCreateTemplate** .

Odsyłacze pokrewne

[Komenda fteCreateTemplate](#)

Przesyłanie danych z plików do wiadomości

Funkcja `file-to-message` produktu Managed File Transfer umożliwia przesyłanie danych z pliku do jednego komunikatu lub wielu komunikatów w kolejce produktu IBM MQ .

Aby możliwe było przesłanie pliku z komunikatem i przesyłanie komunikatów do pliku, zarówno źródłowy, jak i docelowy agent przesyłania muszą być w wersji IBM WebSphere MQ 7.5 lub nowszej albo w wersji IBM WebSphere MQ File Transfer Edition 7.0.3 lub nowszej. Informacje na temat przesyłania komunikatów do pliku zawiera sekcja [“Przesyłanie danych z komunikatów do plików”](#) na stronie 264.

Agent docelowy dla przesyłania plików do wiadomości nie może być agentem mostu protokołu ani agentem mostu `Connect:Direct` .

Dane pliku można przestać do danych komunikatu programu IBM MQ . Komunikaty produktu IBM MQ mogą być odczytywane i używane przez aplikacje. Obsługiwane są następujące typy przesyłania plików do komunikatów:

- Z jednego pliku do pojedynczego komunikatu. Komunikat nie ma ustawionego identyfikatora grupy IBM MQ .
- Z pojedynczego pliku do wielu komunikatów, rozdzielając ten plik na komunikaty o określonej długości. Wszystkie komunikaty mają ten sam identyfikator grupy produktu IBM MQ .
- Z jednego pliku do wielu komunikatów, rozdzielając plik tekstowy na ogranicznik wyrażenia regularnego Java . Wszystkie komunikaty mają ten sam identyfikator grupy produktu IBM MQ .
- Z jednego pliku do wielu komunikatów, rozdzielając plik binarny pod separatorem szesnastkowym. Wszystkie komunikaty mają ten sam identyfikator grupy produktu IBM MQ .

Aby podzielić plik binarny przy użyciu sekwencji bajtów jako separatora, należy użyć parametru **-sqdb** komendy **fteCreateTransfer** . Więcej informacji na ten temat zawiera sekcja [Parametr -sqdb](#).

Domyślnie komunikaty tworzone za pomocą przesyłania plików do komunikatów są trwałe. Komunikaty można ustawić tak, aby były nietrwałe lub miały wartość trwałości zdefiniowaną przez kolejkę docelową.

Jeśli zostanie określone, że plik zostanie podzielony na wiele komunikatów, wszystkie komunikaty utworzone z tego pliku będą miały ten sam identyfikator grupy IBM MQ . Jeśli użytkownik nie poda, że plik zostanie podzielony na wiele komunikatów, z pliku zostanie utworzony tylko jeden komunikat, a ten komunikat nie będzie miał ustawionego identyfikatora grupy IBM MQ .

W przypadku przesyłania plików do dużych komunikatów lub wielu małych komunikatów może być konieczna zmiana niektórych właściwości produktu IBM MQ lub Managed File Transfer . Więcej informacji na ten temat zawiera sekcja [Wskazówki dotyczące ustawiania atrybutów MQ i właściwości MFT powiązanych z wielkością komunikatu](#).

Uwaga: Jeśli kolejka docelowa jest kolejką w klastrze lub jest aliasem w kolejce klastrowej, podczas przesyłania pliku do kolejki zostanie wyświetlony komunikat o błędzie, jeśli właściwość agenta `enableClusterQueueInput` została ustawiona na wartość `true`. Więcej informacji na ten temat zawiera sekcja [Co zrobić, jeśli kolejka docelowa jest kolejką grupowaną, lub alias kolejki klastrowej](#) .

Konfigurowanie agenta do wykonywania operacji przesyłania plików do komunikatów

Domyślnie agenty nie mogą wykonywać operacji przesyłania plików z komunikatami lub komunikatów do pliku. Aby włączyć tę funkcję, należy ustawić właściwość agenta `enableQueueInputOutput` na wartość `true`. Aby włączyć zapisywanie w kolejkach klastrowych IBM MQ , należy również ustawić właściwość agenta `enableClusterQueueInput` na wartość `true`.

O tym zadaniu

W przypadku próby wykonania operacji przesyłania plików z komunikatami do agenta docelowego, który nie ma ustawionej właściwości `enableQueueInputOutput` na wartość `true`, przesyłanie nie powiedzie się. Komunikat dziennika przesyłania, który jest publikowany w menedżerze kolejek koordynacji, zawiera następujący komunikat:

```
BFGI00197E: An attempt to write to a queue was rejected by the destination agent. The agent must have enableQueueInputOutput=true set in the agent.properties file to support transferring to a queue.
```

Aby umożliwić agentowi zapis i odczyt z kolejek, wykonaj następujące kroki:

Procedura

1. Zatrzymaj agenta docelowego za pomocą komendy **Agent fteStop** .
2. Zmodyfikuj plik `agent.properties` , tak aby zawierał wiersz `enableQueueInputOutput=true`.
Plik `agent.properties` znajduje się w katalogu `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/destination_agent_name`.
3. Opcjonalne: Zmodyfikuj plik `agent.properties` , tak aby zawierał wiersz `enableClusterQueueInputOutput=true`. Plik `agent.properties` znajduje się w katalogu `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/destination_agent_name`.
4. Uruchom agenta docelowego za pomocą komendy **Agent fteStart** .

Przykład: przesyłanie pojedynczego pliku do pojedynczego komunikatu

Kolejkę jako miejsce docelowe operacji przesyłania plików można określić za pomocą parametru **-dq** za pomocą komendy **fteCreateTransfer** . Plik źródłowy musi być mniejszy niż maksymalna długość komunikatu ustawiona w kolejce docelowej. Kolejka docelowa nie musi znajdować się w tym samym menedżerze kolejek, co menedżer kolejek, z którym łączy się agent docelowy, ale te dwa menedżery kolejek muszą być w stanie komunikować się.

O tym zadaniu

Plik źródłowy nosi nazwę `/tmp/single_record.txt` i znajduje się w tym samym systemie, co agent źródłowy `AGENT_NEPTUNE`. Agent źródłowy `AGENT_NEPTUNE` korzysta z menedżera kolejek `QM_NEPTUNE`. Agent docelowy ma wartość `AGENT_VENUS`, a agent ten łączy się z menedżerem kolejek `QM_VENUS`. Kolejka docelowa, `RECEIVING_QUEUE`, znajduje się w menedżerze kolejek `QM_MERCURY`. `QM_MERCURY` znajduje się w tej samej sieci IBM MQ co menedżer kolejek `QM_VENUS` i może być do niego dostęp.

Procedura

Wywołaj następującą komendę:

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_VENUS -dm QM_VENUS  
-dq RECEIVING_QUEUE@QM_MERCURY /tmp/single_record.txt
```

Jeśli kolejka docelowa znajduje się w innym menedżerze kolejek, do menedżera kolejek używanego przez agenta docelowego, należy określić wartość parametru **-dq** w następującym formacie: `nazwa_kolejki@nazwa_menedzera_kolejek`. Jeśli wartość `@nazwa_menedzera_kolejek` nie zostanie podana w wartości, agent docelowy przyjmie, że kolejka docelowa znajduje się w menedżerze kolejek agenta docelowego. Wyjątkiem jest sytuacja, gdy właściwość agenta wyjściowego `enableClusterQueueInputOutput` została ustawiona na wartość `true`. W tym przypadku agent docelowy będzie używać standardowych procedur rozstrzygania IBM MQ w celu określenia miejsca, w którym znajduje się kolejka.

Agent źródłowy, `AGENT_NEPTUNE`, odczytuje dane z pliku `/tmp/single_record.txt` i przesyła te dane do agenta docelowego `AGENT_VENUS`. Agent docelowy `AGENT_VENUS` wysyła dane do trwałego

komunikatu w kolejce RECEIVING_QUEUE@QM_MERCURY. Komunikat nie ma ustawionego identyfikatora grupy IBM MQ .

Przykład: dzielenie pojedynczego pliku na wiele komunikatów według długości

Plik można podzielić na wiele komunikatów programu IBM MQ za pomocą parametru **-qs** komendy **fteCreateTransfer** . Plik jest dzielony na sekcje o stałej długości, z których każdy jest zapisywany do pojedynczego komunikatu.

O tym zadaniu

Plik źródłowy nosi nazwę `/tmp/source.file` i jest o 36 kB wielkości. Plik źródłowy znajduje się w tym samym systemie, w którym znajduje się agent źródłowy AGENT_NEPTUNE. Agent źródłowy, AGENT_NEPTUNE, łączy się z menedżerem kolejek QM_NEPTUNE. Agent docelowy to AGENT_MERCURY, który łączy się z menedżerem kolejek QM_MERCURY. Kolejka docelowa, RECEIVING_QUEUE, znajduje się również w menedżerze kolejek QM_MERCURY. Operacja przesyłania powoduje rozdzielający plik źródłowy na sekcje o wielkości 1 kB i zapisuje każdą z tych sekcji na komunikat w sekcji RECEIVING_QUEUE.

Procedura

Wywołaj następującą komendę:

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_MERCURY -dm QM_MERCURY  
-dq RECEIVING_QUEUE -qs 1K /tmp/source.file
```

Agent źródłowy AGENT_NEPTUNE odczytuje dane z pliku `/tmp/source.file` i przesyła te dane do agenta docelowego, AGENT_MERCURY. Agent docelowy, AGENT_MERCURY, zapisuje dane do trzydziestu sześciu komunikatów trwałych o wielkości 1 kB w kolejce RECEIVING_QUEUE@QM_MERCURY. Wszystkie te komunikaty mają ten sam identyfikator grupy IBM MQ , a ostatni komunikat w grupie ma ustawioną flagę IBM MQ LAST_MSG_IN_GROUP.

Przykład: podział pliku tekstowego na wiele komunikatów przy użyciu separatora wyrażenia regularnego

Prześlij pojedynczy plik tekstowy do wielu komunikatów, dzieląc go na każdy zgodny z danym wyrażeniem regularnym Java . W tym celu należy użyć parametru **-dqdt** komendy **fteCreateTransfer** .

O tym zadaniu

Plik jest dzielony na sekcje o zmiennej długości, z których każdy jest zapisywany do pojedynczego komunikatu. Plik tekstowy jest dzielony w każdym punkcie, w którym tekst w pliku jest zgodny z danym wyrażeniem regularnym. Plik źródłowy nosi nazwę `/tmp/names.text` i ma następującą treść:

```
Jenny Jones,John Smith,Jane Brown
```

Wyrażenie regularne, które określa miejsce, w którym ma zostać podzielony plik, to znak przecinka (,).

Plik źródłowy znajduje się w tym samym systemie, w którym znajduje się agent źródłowy AGENT_NEPTUNE, który łączy się z menedżerem kolejek QM_NEPTUNE. Kolejka docelowa, RECEIVING_QUEUE, znajduje się w menedżerze kolejek QM_MERCURY. QM_MERCURY jest również menedżerem kolejek używanym przez agenta docelowego AGENT_MERCURY. Operacja przesyłania powoduje rozdzielający plik źródłowy na sekcje i zapisuje każdą z tych sekcji na komunikat w sekcji RECEIVING_QUEUE.

Procedura

Wywołaj następującą komendę:

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_MERCURY -dm QM_MERCURY
-dq RECEIVING_QUEUE -t text -dqdp postfix -dqdt "," /tmp/names.text
```

Agent źródłowy AGENT_NEPTUNE odczytuje dane z pliku /tmp/names.text i przesyła te dane do agenta docelowego, AGENT_MERCURY. Agent docelowy, AGENT_MERCURY, zapisuje dane do trzech trwałych komunikatów w kolejce RECEIVING_QUEUE. Wszystkie te komunikaty mają ten sam identyfikator grupy IBM MQ, a ostatni komunikat w grupie ma ustawioną flagę IBM MQ LAST_MSG_IN_GROUP.

Dane w komunikatach są następujące.

- Pierwsza wiadomość:

```
Jenny Jones
```

- Drugi komunikat:

```
John Smith
```

- Trzeci komunikat:

```
Jane Brown
```

Przykład: dzielenie pliku tekstowego z ogranicznikiem wyrażenia regularnego, w tym ogranicznikiem w komunikatach

Prześlij pojedynczy plik tekstowy do wielu komunikatów, dzieląc go na każdy zgodny z danym wyrażeniem regularnym Java i dołączając wyrażenie regularne do wynikowych komunikatów. W tym celu należy użyć parametrów **-dqdt** i **-qi** komendy **fteCreateTransfer**.

O tym zadaniu

Przesyłanie pojedynczego pliku tekstowego do wielu komunikatów w kolejce. Plik jest dzielony na sekcje o zmiennej długości, z których każdy jest zapisywany do pojedynczego komunikatu. Plik tekstowy jest dzielony w każdym punkcie, w którym tekst w pliku jest zgodny z danym wyrażeniem regularnym. Plik źródłowy nosi nazwę /tmp/customers.text i ma następującą treść:

```
Customer name: John Smith
Customer contact details: john@example.net
Customer number: 314

Customer name: Jane Brown
Customer contact details: jane@example.com
Customer number: 42

Customer name: James Jones
Customer contact details: jjones@example.net
Customer number: 26
```

Wyrażenie regularne, które określa, gdzie ma zostać podzielony plik, to `Customer\snumber:\s\d+`, który jest zgodny z tekstem "Numer klienta: ", po którym następuje dowolna liczba cyfr. Wyrażenia regularne określone w wierszu komend muszą być ujęte w znaki podwójnego cudzysłowu, aby zapobiec wartościowaniu wyrażenia regularnego przez powłokę komend. Wyrażenie regularne jest wartościowane jako wyrażenie regularne Java. Więcej informacji na ten temat zawiera sekcja [Wyrażenia regularne używane przez MFT](#).

Domyślnie liczba znaków, które może być zgodne z wyrażeniem regularnym, jest ustawiona na pięć. Wyrażenie regularne używane w tym przykładzie jest zgodne z łańcuchami dłuższymi niż pięć znaków. Aby włączyć zgodne elementy, które są dłuższe niż pięć znaków, należy zmodyfikować plik właściwości agenta w celu uwzględnienia właściwości **maxDelimiterMatchLength**.

Domyślnie tekst, który jest zgodny z wyrażeniem regularnym, nie jest dołączany do komunikatów. Aby dołączyć tekst zgodny z wyrażeniem regularnym w komunikatach, tak jak w tym przykładzie, należy użyć parametru **-qi**. Plik źródłowy znajduje się w tym samym systemie, w którym znajduje się agent źródłowy AGENT_NEPTUNE, który łączy się z menedżerem kolejek QM_NEPTUNE. Kolejka docelowa, RECEIVING_QUEUE, znajduje się w menedżerze kolejek QM_MERCURY. QM_MERCURY jest również menedżerem kolejek używanym przez agenta docelowego AGENT_MERCURY. Operacja przesyłania powoduje rozdzielenie pliku źródłowego na sekcje i zapisuje każdą z tych sekcji na komunikat dotyczący RECEIVING_QUEUE.

Procedura

1. Zatrzymaj agenta docelowego za pomocą następującej komendy:

```
fteStopAgent AGENT_MERCURY
```

2. Dodaj następujący wiersz do pliku właściwości agenta dla AGENT_MERCURY:

```
maxDelimiterMatchLength=25
```

Uwaga: Zwiększenie wartości **maxDelimiterMatchLength** może zmniejszyć wydajność.

3. Uruchom agenta docelowego za pomocą następującej komendy:

```
fteStartAgent AGENT_MERCURY
```

4. Wywołaj następującą komendę:

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_MERCURY -dm QM_MERCURY  
-dq RECEIVING_QUEUE  
text -dqdt "Customer\snumber:\s\d+" -qi -dqdp postfix /tmp/customers.text
```

Agent źródłowy AGENT_NEPTUNE odczytuje dane z pliku /tmp/customers.text i przesyła te dane do agenta docelowego, AGENT_MERCURY. Agent docelowy, AGENT_MERCURY, zapisuje dane do trzech trwałych komunikatów w kolejce RECEIVING_QUEUE. Wszystkie te komunikaty mają ten sam identyfikator grupy IBM MQ, a ostatni komunikat w grupie ma ustawioną flagę IBM MQ LAST_MSG_IN_GROUP.

Dane w komunikatach są następujące.

- Pierwsza wiadomość:

```
Customer name: John Smith  
Customer contact details: john@example.net  
Customer number: 314
```

- Drugi komunikat:

```
Customer name: Jane Brown  
Customer contact details: jane@example.com  
Customer number: 42
```

- Trzeci komunikat:

```
Customer name: James Jones  
Customer contact details: jjones@example.net  
Customer number: 26
```

Przykład: Ustawianie właściwości komunikatu produktu IBM MQ w przesyłaniu plików do wiadomości

Za pomocą parametru **-qmp** w komendzie **fteCreateTransfer** można określić, czy właściwości komunikatu produktu IBM MQ są ustawiane w pierwszym komunikacie zapisanej do kolejki docelowej przy użyciu przesyłania. Właściwości komunikatu produktu IBM MQ umożliwiają aplikacji wybór komunikatów do przetwarzania lub pobieranie informacji o komunikacie bez uzyskiwania dostępu do deskryptora komunikatu produktu IBM MQ (MQMD) lub nagłówków MQRFH2 .

O tym zadaniu

Dołącz parametr **-qmp true** do komendy **fteCreateTransfer** . W tym przykładzie identyfikator użytkownika MQMD użytkownika, który wprowadzał komendę, to **larmer**.

Procedura

Wywołaj następującą komendę:

```
fteCreateTransfer -sa AGENT_JUPITER -da AGENT_SATURN -dq MY_QUEUE@MyQM -qmp true
-t text /tmp/source_file.txt
```

Właściwości komunikatu IBM MQ pierwszego komunikatu napisanego przez agenta docelowego **AGENT_SATURN** do kolejki **MY_QUEUE**, w menedżerze kolejek **MyQM** są ustawione na następujące wartości:

```
usr.WMQFTETTransferId=414cbaedefa234889d999a8ed09782395ea213ebbc9377cd
usr.WMQFTETTransferMode=text
usr.WMQFTESourceAgent=AGENT_JUPITER
usr.WMQFTEDestinationAgent=AGENT_SATURN
usr.WMQFTEFileName=source_file.txt
usr.WMQFTEFileSize=1024
usr.WMQFTEFileLastModified=1273740879040
usr.WMQFTEFileIndex=0
usr.WMQFTEMqmdUser=larmer
```

Przykład: Ustawianie właściwości zdefiniowanych przez użytkownika w przesyłaniu plików do wiadomości

Metadane zdefiniowane przez użytkownika są ustawiane jako właściwość komunikatu IBM MQ w pierwszym komunikacie zapisanej do kolejki docelowej przy użyciu przesyłania. Właściwości komunikatu produktu IBM MQ umożliwiają aplikacji wybór komunikatów do przetwarzania lub pobieranie informacji o komunikacie bez uzyskiwania dostępu do deskryptora komunikatu produktu IBM MQ (MQMD) lub nagłówków MQRFH2 .

O tym zadaniu

Należy uwzględnić parametry **-qmp true** i **-md account=123456** w komendzie **fteCreateTransfer** , aby ustawić właściwość **usr.account** na wartość **123456** w nagłówku RFH2 .

Procedura

Wywołaj następującą komendę:

```
fteCreateTransfer -sa AGENT_JUPITER -da AGENT_SATURN -dq MY_QUEUE@MyQM
-qmp true -md account=123456 /tmp/source_file.txt
```

Oprócz standardowego zestawu właściwości komunikatu IBM MQ właściwość zdefiniowana przez użytkownika jest ustawiana w nagłówku komunikatu pierwszego komunikatu napisanego przez agenta

docelowego AGENT_SATURN do kolejki MY_QUEUE, menedżera kolejek MyQM. Nagłówek jest ustawiony na następującą wartość:

```
usr.account=123456
```

Przedrostek `usr` jest dodawany na początku nazwy metadanych definiowanych przez użytkownika.

Przykład: dodawanie właściwości komunikatu definiowanego przez użytkownika dla przesyłania plików z komunikatami

W przypadku korzystania z produktu Managed File Transfer w przypadku przesyłania zarządzanego przez komunikat można uwzględnić właściwość komunikatu zdefiniowaną przez użytkownika dla wynikowego komunikatu.

O tym zadaniu

Aby zdefiniować niestandardową właściwość komunikatu, można użyć dowolnej z następujących metod:

- Podaj parametr `-md` w żądaniu transferu. Więcej informacji na ten temat zawiera sekcja [“Przykład: Ustawianie właściwości zdefiniowanych przez użytkownika w przesyłaniu plików do wiadomości”](#) na stronie 261.
- Użyj zadania Ant. Można użyć dowolnego z następujących elementów: `fte:filecopy` lub `fte:filemove`. Poniżej przedstawiono przykład zadania `fte:filecopy`:

```
<project xmlns:fte="antlib:com.ibm.wmqfte.ant.taskdefs" default="complete">
<!-- Initialise the properties used in this script.-->

<target name="init" description="initialise task properties">
    <property name="src.file" value="/home/user/file1.bin"/>
    <property name="dst.queue" value="TEST.QUEUE@qm2"/>
    <fte:uuid property="job.name" length="8"
prefix="copyjob#"/>
</target>
<target name="step1" depends="init" description="transfer file">

<fte:filecopy cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
src="agent1@qm1" dst="agent2@qm2"
rcproperty="copy.result">

<fte:metadata>
<fte:entry name="fileName" value="{FileName}"/>
</fte:metadata>

<fte:filespec srcfilespec="{src.file}" dstqueue="{dst.queue}"
dstmsgprops="true"/>

</fte:filecopy>

</target>
</project>
```

- Użyj monitora zasobów i podstawienia zmiennych. W poniższym przykładzie przedstawiono niektóre dane XML zadania przesyłania:

```
<?xml version="1.0" encoding="UTF-8"?>
<monitor:monitor
xmlns:monitor="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinition"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance" version="5.00"
xsi:schemaLocation="https://www.ibm.com/xmlns/wmqfte/7.0.1/MonitorDefinit ion ./Monitor.xsd">
<name>METADATA</name>
<pollInterval units="minutes">5</pollInterval>
<batch maxSize="5"/>
<agent>AGENT1</agent>
<resources>
<directory recursionLevel="0">e:\temp</directory>
</resources>
<triggerMatch>
```

```

<conditions>
  <allOf>
    <condition>
      <fileMatch>
        <pattern>*.txt</pattern>
      </fileMatch>
    </condition>
  </allOf>
</conditions>
</triggerMatch>
<tasks>
  <task>
    <name/>
    <transfer>
      <request version="5.00"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
        <managedTransfer>
          <originator>
            <hostName>mqjason.raleigh.ibm.com.</hostName>
            <userID>administrator</userID>
          </originator>
          <sourceAgent QMgr="AGENTQM" agent="AGENT1"/>
          <destinationAgent QMgr="AGENTQM" agent="AGENT2"/>
          <transferSet priority="0">
            <metaDataSet>
              <metaData key="FileName">${FileName}</metaData>
            </metaDataSet>
            <item checksumMethod="MD5" mode="text">
              <source disposition="delete" recursive="false">
                <file>${FilePath}</file>
              </source>
              <destination type="queue">
                <queue persistent="true"
setMqProps="true">TEST.QUEUE@AGENTQM</queue>
              </destination>
            </item>
          </transferSet>
        </job>
      </managedTransfer>
    </request>
  </transfer>
</task>
</tasks>
<originator>
  <hostName>mqjason.raleigh.ibm.com.</hostName>
  <userID>administrator</userID>
</originator>
</monitor:monitor>

```

Zadania pokrewne

“Przykład: Ustawianie właściwości komunikatu produktu IBM MQ w przesyłaniu plików do wiadomości”
na stronie 261

Za pomocą parametru **-qmp** w komendzie **fteCreateTransfer** można określić, czy właściwości komunikatu produktu IBM MQ są ustawiane w pierwszym komunikacie zapisanej do kolejki docelowej przy użyciu przesyłania. Właściwości komunikatu produktu IBM MQ umożliwiają aplikacji wybór komunikatów do przetwarzania lub pobieranie informacji o komunikacie bez uzyskiwania dostępu do deskryptora komunikatu produktu IBM MQ (MQMD) lub nagłówków MQRFH2 .

Odsyłacze pokrewne

[fte: filecopy Ant zadanie](#)

[fte: filemove Ant , zadanie](#)

Niepowodzenie przestania pliku do wiadomości

Jeśli operacja przesyłania pliku do komunikatu nie powiedzie się po tym, jak agent rozpoczął zapisywanie danych pliku w kolejce docelowej, agent zapisze komunikat w kolejce w celu wskazania aplikacji konsumująca komunikaty, które wystąpiły w przypadku niepowodzenia.

Komunikat zapisany w kolejce docelowej, jeśli wystąpi awaria:

- Jest pusty

- Ma ten sam identyfikator grupy produktu IBM MQ , co poprzedni komunikat zapisany w kolejce docelowej przez agenta.
- Ma ustawioną flagę IBM MQ LAST_MSG_IN_GROUP
- Zawiera dodatkowe właściwości komunikatu produktu IBM MQ , jeśli właściwości komunikatu są włączone. Więcej informacji na ten temat można znaleźć w temacie [Właściwości komunikatuMQ ustawionym przez MFT w komunikatach zapisanych w kolejkach docelowych](#).

Przykład

Żądanie transferu jest wymagane przez uruchomienie następującej komendy:

```
fteCreateTransfer -sa AGENT_JUPITER -da AGENT_SATURN -dq RECEIVING_QUEUE
-qmp true -qs 1K /tmp/source1.txt
```

Plik source1.txt to 48 kB. Operacja przesyłania powoduje rozdzielenie tego pliku na 1 kB i zapisuje te komunikaty w kolejce RECEIVING_QUEUE kolejki docelowej.

Gdy operacja przesyłania jest w toku, po zapisaniu przez agenta 16 komunikatów do RECEIVING_QUEUE, w agencji źródłowym wystąpi błąd.

Agent zapisze pusty komunikat do RECEIVING_QUEUE. Oprócz standardowego zestawu właściwości komunikatu, pusty komunikat ma ustawione następujące właściwości komunikatu:

```
usr.WMQFTEResultCode = 40
usr.WMQFTESupplement = BFGTR0036I: The transfer failed to complete successfully.
```

V9.2.0.2 → **V9.2.2** From IBM MQ 9.2.0 Fix Pack 2 i IBM MQ 9.2.2, when a transfer from a file fails, because of a delimiter size check error, only one empty message is sent. Ponadto właściwości użytkownika są dodawane do tego komunikatu, jeśli niepowodzenie przesyłania było spowodowane ogranicznikiem przekraczającym wielkość ustawioną w agencji docelowym.

Przesyłanie danych z komunikatów do plików

Opcja "message-to-file" produktu Managed File Transfer umożliwia przesyłanie danych z jednego lub większej liczby komunikatów w kolejce IBM MQ do pliku, zestawu danych (w systemie z/OS), lub przestrzeni plików użytkownika. Jeśli istnieje aplikacja, która tworzy lub przetwarza komunikaty produktu IBM MQ , można użyć funkcji message-to-file produktu Managed File Transfer w celu przesłania tych komunikatów do pliku w dowolnym systemie w sieci Managed File Transfer .

Więcej informacji na temat przesyłania plików do komunikatów zawiera sekcja [“Przesyłanie danych z plików do wiadomości”](#) na stronie 256.



Ostrzeżenie: Agent źródłowy dla przesyłania komunikatów do pliku nie może być agentem mostu protokołu ani agentem mostu Connect:Direct .

Dane komunikatu programu IBM MQ można przesłać do pliku. Obsługiwane są następujące typy przesyłania komunikatów do pliku:

- Z pojedynczego komunikatu do pojedynczego pliku
- Z wielu komunikatów do jednego pliku
- Z wielu komunikatów o tym samym identyfikatorze grupy IBM MQ do pojedynczego pliku.
- Z wielu komunikatów do jednego pliku, włącznie z separatorem tekstowym lub binarnym między danymi z każdego komunikatu zapisanego do pliku.

W przypadku przesyłania plików z dużych komunikatów lub wielu małych komunikatów może być konieczna zmiana niektórych właściwości produktu IBM MQ lub Managed File Transfer . Więcej informacji na ten temat zawiera sekcja [Wskazówki dotyczące ustawiania atrybutów MQ i właściwości MFT powiązanych z wielkością komunikatu](#).

Z poziomu produktu IBM MQ 9.1.0w komunikacie do przesyłania plików agent źródłowy przegląda komunikaty z kolejki źródłowej, w przeciwieństwie do destrukcyjnego GET we wcześniejszych wersjach produktu IBM MQ. Komunikaty są usuwane z kolejki źródłowej po przeglądowaniu wszystkich komunikatów (w grupie, jeśli używane są grupowanie komunikatów), a także o danych zapisanych w pliku docelowym. Dzięki temu komunikaty pozostają w kolejce źródłowej, jeśli operacja przesyłania nie powiedzie się lub zostanie anulowana. Ze względu na tę zmianę należy również zapewnić uprawnienie do BROWSE wraz z uprawnieniem GET w celu uruchomienia przesyłania komunikatów do przesyłania plików.

Z produktów IBM MQ 9.0.0 Fix Pack 2 i IBM MQ 9.0.4zaktualizowano Managed File Transfer , aby przywrócić sprawdzenie porównania, które zostało wcześniej usunięte przez APAR IT18213, identyfikatora przesyłania i wartości atrybutu groupId w ładunku XML żądania przesyłania. Jeśli te dwa identyfikatory są równoważne, agent źródłowy używa identyfikatora jako opcji dopasowania identyfikatora komunikatu (w przeciwieństwie do opcji dopasowania identyfikatora grupy) dla pierwszej próby MQGET, która jest podejmowana w kolejce wejściowej dla przesyłania komunikatów w pliku komunikatów.

Konfigurowanie agenta do wykonywania operacji przesyłania komunikatów w pliku

Domyślnie agenty nie mogą wykonywać operacji przesyłania komunikatów do pliku lub do przesyłania komunikatów. Aby włączyć tę funkcję, należy ustawić właściwość agenta enableQueueInputOutput na wartość true.

O tym zadaniu

W przypadku próby wykonania komunikatu w celu przesyłania pliku z agenta źródłowego, który nie ma ustawionej właściwości enableQueueInputOutput na wartość true, przesyłanie nie powiedzie się. Komunikat dziennika przesyłania, który jest publikowany w menedżerze kolejek koordynacji, zawiera następujący komunikat:

```
BFGI00197E: An attempt to read from a queue was rejected by the source agent.  
The agent must have enableQueueInputOutput=true set in the agent.properties file  
to support transferring from a queue.
```

Aby umożliwić agentowi zapis i odczyt z kolejek, wykonaj następujące kroki:

Procedura

1. Zatrzymaj agenta źródłowego za pomocą komendy **Agent fteStop** .
2. Zmodyfikuj plik `agent.properties` , tak aby zawierał wiersz `enableQueueInputOutput=true`.
Plik `agent.properties` znajduje się w katalogu `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/source_agent_name`.
3. Uruchom agenta źródłowego za pomocą komendy **Agent fteStart** .

Przykład: przesyłanie z kolejki do pojedynczego pliku

Kolejkę IBM MQ można określić jako źródło przesyłania plików za pomocą parametru **-sq** za pomocą komendy **fteCreateTransfer** .

O tym zadaniu

Dane źródłowe są zawarte w trzech komunikatach w kolejce START_QUEUE. Ta kolejka musi znajdować się w menedżerze kolejek agenta źródłowego, `QM_NEPTUNE`.

Procedura

Wywołaj następującą komendę:

Ogranicznik tekstu jest dodawany na początku danych z każdego z czterech komunikatów w START_QUEUE przez agenta źródłowego AGENT_NEPTUNE. Dane te są zapisywane w pliku docelowym /out/output.txt.

Przykład: wstawianie separatora binarnego po danych z każdego komunikatu

Podczas przesyłania w trybie binarnym z kolejki źródłowej do pliku można określić, że separator binarny jest wstawiany po danych z poszczególnych komunikatów za pomocą parametrów **-sq**, **-sqdbi** **-sqdp** za pomocą komendy **fteCreateTransfer**.

O tym zadaniu

W tym przykładzie w kolejce START_QUEUE znajdują się trzy komunikaty. Ta kolejka znajduje się w menedżerze kolejek agenta źródłowego, *QM_NEPTUNE*. Separator binarny, który ma zostać wstawiony po danych z każdego komunikatu, musi być wyrażony jako rozdzielona przecinkami lista bajtów szesnastkowych, na przykład: x34,xE7,xAE.

Procedura

Wywołaj następującą komendę:

```
fteCreateTransfer -sa AGENT_NEPTUNE -sm QM_NEPTUNE -da AGENT_VENUS -df /out/binary.file  
-sqdp postfix -sqdb x34,xE7,xAE -sq START_QUEUE
```

Ogranicznik binarny jest dodawany do danych z każdego z trzech komunikatów w START_QUEUE przez agenta źródłowego AGENT_NEPTUNE. Dane te są zapisywane w pliku docelowym /out/binary.file.

Monitorowanie kolejki i używanie podstawiania zmiennych

Za pomocą komendy **fteCreateMonitor** można monitorować kolejkę i przysyłać komunikaty z monitorowanej kolejki do pliku. Wartość dowolnej właściwości komunikatu IBM MQ w pierwszym komunikacie, która ma zostać odczyta z monitorowanej kolejki, może zostać podstawiona w definicji XML zadania i użyta do zdefiniowania zachowania przesyłania.

O tym zadaniu

W tym przykładzie agent źródłowy jest nazywany AGENT_VENUS, który łączy się z QM_VENUS. Kolejka monitorów AGENT_VENUS nosi nazwę START_QUEUE i znajduje się w systemie QM_VENUS. Agent odpytuje kolejkę co 30 minut.

Gdy pełna grupa komunikatów jest zapisywana w kolejce, zadanie monitora wysyła grupę komunikatów do pliku w jednym z kilku agentów docelowych, z których wszystkie łączą się z menedżerem kolejek QM_MARS. The name of the file that the group of messages is transferred to is defined by the IBM MQ message property `usr.fileName` on the first message in the group. Nazwa agenta, do którego wysyłana jest grupa komunikatów, jest zdefiniowana przez właściwość komunikatu IBM MQ `usr.toAgent` w pierwszym komunikacie w grupie. Jeśli nagłówek `usr.toAgent` nie jest ustawiony, domyślną wartością, która ma być używana dla agenta docelowego, jest AGENT_MAGENTA.

Jeśli użytkownik określi opcję `useGroups="true"`, jeśli nie zostanie również określony parametr `groupId="${GROUPID}"`, zostanie on wyświetlony po pierwszym komunikacie w kolejce. Jeśli na przykład podstawianie zmiennych jest używane do generowania pliku `fileName`, możliwe jest, że zawartość pliku `a.txt` nie będzie poprawna. Dzieje się tak dlatego, że `fileName` monitor, ale w rzeczywistości zostanie wyświetlony komunikat, który nie powinien wygenerować pliku o nazwie `fileName`.

Procedura

1. Utwórz plik XML zadania, który definiuje zadanie wykonywane przez monitor po jego wyzwoleniu.

```

<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_VENUS" QMgr="QM_VENUS" />
    <destinationAgent agent="toAgent" QMgr="QM_MARS" />
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <queue useGroups="true" groupId="{GROUPID}">START_QUEUE</queue>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/{fileName}.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>

```

Zmienne, które są zastępowane wartościami nagłówek komunikatów produktu IBM MQ, są podświetlone na **pogrubienie**. Ten kod XML zadania jest zapisywany w pliku /home/USER1/task.xml

2. Utwórz monitor zasobów, aby monitorować kolejkę START_QUEUE.

Wykonaj następującą komendę:

```

fteCreateMonitor -ma AGENT_VENUS -mm QM_VENUS -mq START_QUEUE
                 -mn myMonitor -mt /home/USER1/task.xml
                 -tr completeGroups -pi 30 -pu minutes -dv toAgent=AGENT_MAGENTA

```

3. Użytkownik lub program zapisuje grupę komunikatów do kolejki START_QUEUE.

Pierwszy komunikat w tej grupie zawiera następujące zestaw właściwości komunikatów produktu IBM MQ:

```

usr.fileName=larmer
usr.toAgent=AGENT_VIOLET

```

4. Monitor jest wyzwalany po zapisaniu kompletnej grupy. Agent substytuuje właściwości komunikatu produktu IBM MQ do pliku XML zadania.

Wynikiem tego jest transformacja kodu XML zadania na:

```

<?xml version="1.0" encoding="UTF-8" ?>
<request version="4.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
  <managedTransfer>
    <originator>
      <hostName>reportserver.com</hostName>
      <userID>USER1</userID>
    </originator>
    <sourceAgent agent="AGENT_VENUS" QMgr="QM_VENUS" />
    <destinationAgent agent="AGENT_VIOLET" QMgr="QM_MARS" />
    <transferSet>
      <item mode="binary" checksumMethod="none">
        <source>
          <queue useGroups="true" groupId="{GROUPID}">START_QUEUE</queue>
        </source>
        <destination type="file" exist="overwrite">
          <file>/reports/larmer.rpt</file>
        </destination>
      </item>
    </transferSet>
  </managedTransfer>
</request>

```

Wyniki

Operacja przesyłania zdefiniowana przez kod XML zadania jest wykonywana. Pełna grupa komunikatów odczytanych ze zmiennej `START_QUEUE` przez `AGENT_VENUS` jest zapisywana do pliku o nazwie `/reports/larmer.rpt` w systemie, w którym działa `AGENT_VIOLET`.

Co dalej

Przesyłanie poszczególnych komunikatów do osobnego pliku

Jeśli chcesz monitorować kolejkę i mieć każdy komunikat przesyłany do osobnego pliku, możesz użyć podobnej techniki do opisanej wcześniej w tym temacie.

1. Utwórz monitor zgodnie z opisem poprzednio, podając parametr **-tr completeGroups** w komendzie **fteCreateMonitor**.
2. W pliku XML zadania podaj następujące informacje:

```
<queue useGroups="true" groupId="${GROUPID}">START_QUEUE</queue>
```

Jednak po umieszczeniu komunikatów w kolejce źródłowej nie należy umieszczać ich w grupie IBM MQ. Dodaj właściwości komunikatu IBM MQ do każdego komunikatu. Na przykład można określić właściwość `usr.filename` z unikalną wartością nazwy pliku dla każdego komunikatu. Spowoduje to, że program Managed File Transfer Agent będzie traktował każdy komunikat w kolejce źródłowej jako oddzielną grupę.

Przykład: niepowodzenie przesyłania komunikatów z plikiem przy użyciu właściwości komunikatu IBM MQ

Można spowodować niepowodzenie operacji przesyłania komunikatów, ustawiając właściwość komunikatu `usr.UserReturnCode` IBM MQ na wartość niezerową. Można również określić dodatkowe informacje o przyczynie niepowodzenia, ustawiając właściwość komunikatu `usr.UserSupplement` IBM MQ.

O tym zadaniu

W tym przykładzie trwa przesyłanie między kolejką `INPUT_QUEUE` i plikiem `/home/user/output.file`.

Użytkownik tworzy komunikaty i umieszcza je w kolejce `INPUT_QUEUE`. Agent źródłowy konsumuje komunikaty z kolejki `INPUT_QUEUE` i wysyła dane przesyłania do agenta docelowego. Agent docelowy zapisuje te dane w pliku `/home/user/output.file`.

Użytkownik pisząc komunikaty do kolejki `INPUT_QUEUE` chce zatrzymać przesyłanie, które jest w toku, i usunąć wszystkie dane, które zostały już zapisane w pliku docelowym.

Procedura

1. Użytkownik zapisze komunikat do kolejki `INPUT_QUEUE`, która ma ustawione następujące właściwości komunikatów programu IBM MQ:

```
usr.UserReturnCode=1  
usr.UserSupplement="Cancelling transfer - sent wrong data."
```

2. Agent źródłowy odczytuje właściwości komunikatu produktu IBM MQ i zatrzymuje przetwarzanie komunikatów z kolejki. Agent docelowy usuwa wszystkie dane pliku, które zostały zapisane w katalogu docelowym.
3. Agent źródłowy wysyła komunikat dziennika przesyłania do menedżera kolejek koordynacji, który zgłasza niepowodzenie przesyłania.

Komunikat zawiera następujące informacje:

```
<?xml version="1.0" encoding="UTF-8"?>  
<transaction version="1.00"  
  ID="414d5120514d3120202020202020202020202020207e970d4920008702" agentRole="sourceAgent"
```

```

        xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
        xsi:noNamespaceSchemaLocation="TransferLog.xsd"
        xmlns="">
<action time="2008-11-02T21:28:09.593Z">progress</action>
<sourceAgent agent="FTEAGENT" QMgr="QM1">
  <systemInfo architecture="x86" name="Windows 7"
    version="6.1 build 7601 Service Pack 1" />
</sourceAgent>
<destinationAgent agent="FTEAGENT" QMgr="QM1">
  <systemInfo architecture="x86" name="Windows 7"
    version="6.1 build 7601 Service Pack 1" />
</destinationAgent>
<originator>
  <hostName>reportserver.com</hostName>
  <userID>USER1</userID>
  <mqmdUserID>USER1 </mqmdUserID>
</originator>
<transferSet index="0" size="1"
  startTime="2008-11-02T21:28:09.281Z"
  total="1">
  <item mode="binary">
    <source>
      <queue>INPUT_QUEUE@QM1</queue>
    </source>
    <destination exist="error">
      <file>/home/user/output.file</file>
    </destination>
    <status resultCode="1">
      <supplement>Cancelling transfer - sent wrong data.</supplement>
    </status>
  </item>
</transferSet>
</transaction>

```

Most protokołu

Most protokołu umożliwia sieci Managed File Transfer (MFT) dostęp do plików przechowywanych na serwerze plików poza siecią produktu MFT, zarówno w domenie lokalnej, jak i w zdalnej lokalizacji. Ten serwer plików może korzystać z protokołów sieciowych FTP, FTPS lub SFTP. Każdy serwer plików wymaga co najmniej jednego dedykowanego agenta. Dedykowany agent jest znany jako agent mostu protokołu. Agent mostu może wchodzić w interakcje z wieloma serwerami plików.

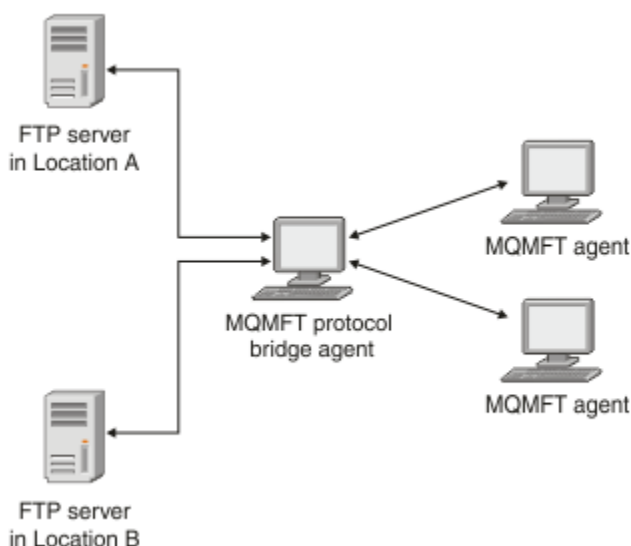
Most protokołu jest dostępny jako część komponentu usługi produktu Managed File Transfer. Istnieje możliwość posiadania wielu dedykowanych agentów w jednym systemie z uruchomionym MFT, które łączą się z różnymi serwerami plików.

Agenta mostu protokołu można używać do przesyłania plików do wielu punktów końcowych jednocześnie. Program MFT udostępnia plik o nazwie `ProtocolBridgeProperties.xml`, który można edytować w celu zdefiniowania różnych serwerów plików protokołów, do których mają być przesyłane pliki. Komenda **fteCreateBridgeAgent** dodaje szczegółowe informacje na temat domyślnego serwera plików protokołu do programu `ProtocolBridgeProperties.xml`. Ten plik jest opisany w sekcji [Format pliku właściwości mostu protokołu](#).

Agent mostu protokołu może być używany do wykonywania następujących czynności:

- Przesyłanie plików z sieci MFT do zdalnego serwera przy użyciu protokołu FTP, FTPS lub SFTP.
- Pobieranie plików z serwera zdalnego przy użyciu protokołu FTP, FTPS lub SFTP do sieci produktu MFT.

Uwaga: Agent mostu protokołu może obsługiwać tylko serwery FTP, FTPS lub SFTP, które umożliwiają dostęp do plików za pomocą ich bezwzględnej ścieżki do plików. Jeśli w żądaniu przestania określono względną ścieżkę do pliku, agent mostu protokołu podejmie próbę przekształcenia ścieżki względnej w bezwzględną ścieżkę do pliku w oparciu o katalog osobisty używany do logowania się do serwera protokołu. Te serwery protokołów, które zezwalają na dostęp do plików tylko na podstawie bieżącego katalogu, nie są obsługiwane przez agenta mostu protokołu.



Na diagramie przedstawiono dwa serwery FTP w różnych lokalizacjach. Serwery FTP są używane do wymiany plików za pomocą agentów Managed File Transfer. Agent mostu protokołu znajduje się między serwerami FTP a resztą sieci MFT i jest skonfigurowany do komunikacji z obydwoma serwerami FTP.

Oprócz agenta mostu protokołu należy upewnić się, że w sieci produktu MFT jest inny agent. Agent mostu protokołu jest mostem tylko dla serwera FTP, FTPS lub serwera SFTP i nie zapisuje przestanych plików na dysku lokalnym. Aby przestać pliki do lub z serwera FTP, FTPS lub SFTP, należy użyć agenta mostu protokołu jako miejsca docelowego lub źródła dla przesyłania plików (reprezentującego serwer FTP, FTPS lub serwer SFTP) i innego standardowego agenta jako odpowiadającego mu źródła lub miejsca docelowego.

Podczas przesyłania plików za pomocą mostu protokołu, most musi mieć uprawnienia do odczytu katalogu źródłowego lub docelowego zawierającego pliki, które mają zostać przesunięte. Na przykład, jeśli mają być przesyłane pliki z katalogu `/home/ftc/bridge`, który ma uprawnienia do wykonywania (`d -- x -- x -- x`), wszelkie próby przeniesienia z tego katalogu nie powiedzą się z następującym komunikatem o błędzie:

```
BFGBR0032E: Attempt to read filename from the protocol file server
has failed with server error 550. Failed to open file.
```

Konfigurowanie agenta mostu protokołu

Agent mostu protokołu jest podobny do standardowego agenta MFT. Utwórz agent mostu protokołu za pomocą komendy `fteCreateBridgeAgent`. Agent mostu protokołu można skonfigurować przy użyciu pliku `ProtocolBridgeProperties.xml`, który jest opisany w sekcji [Format pliku właściwości mostu protokołu](#). Jeśli używana jest wcześniejsza wersja, należy skonfigurować agenta przy użyciu specyficznych właściwości mostu protokołu opisanych w sekcji [Zaawansowane właściwości agenta: most protokołu](#) i [Zaawansowane właściwości agenta: Rejestrowanie agenta mostu protokołu](#). Dla wszystkich wersji można również skonfigurować odwzorowanie referencji zgodnie z opisem w sekcji ["Odwzorowywanie informacji autoryzacyjnych dla serwera plików"](#) na stronie 279. Po skonfigurowaniu agenta mostu protokołu dla określonego serwera plików protokołu, można użyć tego agenta tylko w tym celu.

Odtwarzanie mostu protokołu

Jeśli agent mostu protokołu nie może połączyć się z serwerem plików, ponieważ serwer plików jest niedostępny, wszystkie żądania przesyłania plików są umieszczane w kolejce do momentu, gdy serwer plików stanie się dostępny. Jeśli agent mostu protokołu nie może połączyć się z serwerem plików, ponieważ agent używa niewłaściwych referencji, przesyłanie nie powiedzie się, a komunikat dziennika przesyłania będzie odzwierciedlał ten błąd. Jeśli agent mostu protokołu zostanie zakończony

z dowolnej przyczyny, wszystkie żądane transfery plików zostaną zachowane, a most protokołu zostanie zrestartowany.

Podczas przesyłania plików pliki są zwykle zapisywane jako pliki tymczasowe w miejscu docelowym, a następnie zmieniane są po zakończeniu przesyłania plików. Jeśli jednak miejscem docelowym przesyłania jest serwer plików protokołu, który jest skonfigurowany jako ograniczony zapis (użytkownicy mogą przysyłać pliki do serwera plików protokołu, ale nie mogą w żaden sposób zmieniać tych przesyłanych plików; użytkownicy mogą zapisywać tylko jeden raz), przesłane pliki są zapisywane bezpośrednio w miejscu docelowym. Oznacza to, że jeśli podczas przesyłania wystąpi problem, częściowo zapisane pliki pozostaną na docelowym serwerze plików protokołu, a program Managed File Transfer nie może usunąć ani edytować tych plików. W takiej sytuacji przesyłanie nie powiedzie się.

Definiowanie właściwości dla serwerów plików protokołu przy użyciu pliku **ProtocolBridgeProperties.xml**

Zdefiniuj właściwości jednego lub większej liczby serwerów plików protokołu, do których mają być przesyłane pliki, i z pliku `ProtocolBridgeProperties.xml`, który jest udostępniany przez produkt Managed File Transfer w katalogu konfiguracji agenta.

O tym zadaniu

Komenda **fteCreateBridgeAgent** tworzy plik `ProtocolBridgeProperties.xml` w katalogu konfiguracji agenta `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name`. Komenda ta tworzy również wpis w pliku dla domyślnego serwera plików protokołu, jeśli podczas uruchamiania komendy określono wartość domyślną.

W komunikacie BFGCL0392I znajduje się położenie pliku `ProtocolBridgeProperties.xml`.

```
<?xml version="1.0" encoding="IBM-1047"?>
<!--
This ProtocolBridgeProperties.xml file determines the protocol servers that will be accessed by
the
MQMFT protocol bridge agent.

Each protocol server is defined using either a <tns:ftpServer>, <tns:ftpsServer>, or
<tns:sftpServer>
element - depending on the protocol used to communicate with the server. When the protocol
bridge agent participates in a managed file transfer it will determine which server to used
based on
the prefix (if any) present on the file path. For example a file path of 'server1:/home/user/
file.txt' would
be interpreted as a request to transfer /home/user/file.txt using 'server1'. The server name
is compared
to the 'name' attribute of each <tns:ftpServer>, <tns:ftpsServer> or <tns:sftpServer> element
in this
XML document and the first match is used to determine which protocol server the protocol bridge
agent will connect to. If no match is found then the managed file transfer operation will fail.

If a file path is not prefixed with a server name, for example '/home/user/file.txt' then this
XML
document can specify a default server to use for the managed file transfer. To specify a
default server use the <tns:defaultServer> element as the first element inside the
<tns:serverProperties>
element. The default server will be used whenever the protocol bridge agent participates in
a managed file transfer for file names which do not specify a prefix.

An optional <tns:limits> element can be specified within each server definition. This element
contains
attributes that govern the amount of resources used by each defined server.

An optional <tns:credentialsFile> element can be specified within each serverProperties
definition. This
element contains a path to a file containing credentials to be used when connecting to defined
servers.

An example ProtocolBridgeProperties.xml file is as follows:

<?xml version="1.0" encoding="UTF-8"?>
<tns:serverProperties xmlns:tns="http://wmmqfte.ibm.com/ProtocolBridgeProperties"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmmqfte.ibm.com/ProtocolBridgeProperties
  ProtocolBridgeProperties.xsd">
```



```

<tns:credentialsFile path="$HOME/ProtocolBridgeCredentials.xml" />
<tns:defaultServer name="myFTPserver" />
  <tns:ftpServer name="myFTPserver" host="windows.hursley.ibm.com" port="1234"
platform="windows"
  timeZone="Europe/London" locale="en_GB" fileEncoding="UTF-8"
  listFormat="unix" limitedWrite="false">
    <tns:limits maxListFileNames="100" maxListDirectoryLevels="999999999"
      maxReconnectRetry="2" reconnectWaitPeriod="10"
      maxSessions="60" socketTimeout="30" />
  </tns:ftpServer>
  <tns:ftpsServer name="myFTPSserver" host="unix.hursley.ibm.com" platform="unix"
  timeZone="Europe/London" locale="en_GB" fileEncoding="UTF8"
  listFormat="unix" limitedWrite="false" ftpsType="explicit"
  trustStore="C:\FTE\keystores\myFTPSServer\FTPSKeyStore.jks"
  trustStorePassword="password">
    <tns:limits maxReconnectRetry="10" connectionTimeout="10"/>
  </tns:ftpsServer>
  <tns:sftpServer name="mySFTPserver" host="windows.hursley.ibm.com" platform="windows"
  timeZone="Europe/London" locale="en_GB" fileEncoding="UTF-8"
  limitedWrite="false">
    <tns:limits connectionTimeout="60"/>
  </tns:sftpServer>
</tns:serverProperties>

```

This example shows the outermost <tns:serverProperties> element which must exist for the document to be valid, an optional <tns:defaultServer> element, as well as definitions for an FTP, FTPS and SFTP server.

The attributes of the <tns:ftpServer>, <tns:ftpsServer> and <tns:sftpServer> elements determine the characteristics of the connection established to the server. These attributes correspond to the command line parameters for the 'fteCreateBridgeAgent' command.

The following attributes are valid for all of the <tns:ftpServer>, <tns:ftpsServer> and <tns:sftpServer> elements: name, host, port, platform, fileEncoding, limitedWrite and controlEncoding.

The following attributes are valid for the <tns:ftpServer> and <tns:ftpsServer> elements: timeZone, locale, listFormat, listFileRecentDateFormat, listFileOldDateFormat, and monthShortNames.

The following attributes are valid for the <tns:ftpServer> element only: passiveMode

The following attributes are valid for the <tns:ftpsServer> element only: ftpsType, trustStore, trustStorePassword, trustStoreType, keyStore, keyStorePassword, keyStoreType, ccc, protFirst, auth, and connectTimeout.

The following attributes are valid for the <tns:limits> element within all of the <tns:ftpServer>, <tns:ftpsServer> and <tns:sftpServer> elements: maxListFileNames, maxListDirectoryLevels, maxReconnectRetry, reconnectWaitPeriod, maxSessions and socketTimeout

```

-->
<tns:serverProperties xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeProperties"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeProperties
ProtocolBridgeProperties.xsd">

```

<!-- By default the location of the credentials file is in the home directory of the user that started the -->

<!-- protocol bridge agent. If you wish to specify a different location use the credentialsFile element to -->

<!-- do this. For

example:

```

<!-- <tns:credentialsFile path="/test/
ProtocolBridgeCredentials.xml"/>

```

-->

-->

```

    <tns:defaultServer name="WINMVSCA.HURSLEY.IBM.COM" />
    <tns:ftpServer name="WINMVSCA.HURSLEY.IBM.COM" host="WINMVSCA.HURSLEY.IBM.COM"
platform="unix"
        timeZone="Europe/London" locale="en-GB" fileEncoding="US-ASCII"
        listFormat="unix" limitedWrite="false" />

    <!-- Define servers here -->
</tns:serverProperties>

```

Komenda może wygenerować następujący komunikat: BFGCL0532I:

Do działania tego agenta wymagane jest ręczne utworzenie dodatkowego pliku referencji. Domyślnie ten plik nosi nazwę ProtocolBridgeCredentials.xml i znajduje się w katalogu głównym. Katalog użytkownika, który uruchamia agenta. Na przykład, jeśli ten użytkownik uruchomił agenta Położenie to: \$HOME/ProtocolBridgeCredentials.xml

Jeśli używany jest plik referencji:

1. Więcej informacji na temat sposobu tworzenia można znaleźć w poniższym tekście.
2. Plik referencji musi znajdować się w katalogu z ograniczonymi uprawnieniami. Na przykład nie może być żadnego prawa do odczytu dla innych użytkowników.
3. Podaj położenie katalogu dla pliku referencji w zmiennej środowiskowej \$HOME dla ID użytkownika uruchomionego agenta lub zmodyfikuj plik ProtocolBridgeProperties.xml i określ położenie w:

```
<tns:credentialsFile path="/test/ProtocolBridgeCredentials.xml"/>
```

Jeśli chcesz dodać kolejne niedomyślne serwery protokołów, edytuj ten plik, aby zdefiniować ich właściwości. W tym przykładzie dodawany jest dodatkowy serwer FTP.

Uwaga: Agent mostu protokołu nie obsługuje blokowania plików. Wynika to z faktu, że produkt Managed File Transfer nie obsługuje mechanizmu blokowania plików na serwerze plików.

Procedura

1. Zdefiniuj serwer plików protokołu, wstawiając do pliku następujące wiersze jako element potomny produktu <tns:serverProperties>:

```

<tns:ftpServer name="myserver" host="myhost.hursley.ibm.com" port="1234"
platform="windows"
        timeZone="Europe/London" locale="en-GB" fileEncoding="UTF-8"
        listFormat="unix" limitedWrite="false" >
<tns:limits maxListFileNames="10" maxListDirectoryLevels="500"/>

```

2. Następnie zmień wartość atrybutów:

- name to nazwa serwera plików protokołu.
- host to nazwa hosta lub adres IP serwera plików protokołu.
- port to numer portu serwera plików protokołu.
- platform to platforma, na której działa serwer plików protokołu.
- timeZone jest strefą czasową, w której działa serwer plików protokołu.
- locale jest językiem używanym na serwerze plików protokołu.
- fileEncoding jest kodowaniem znaków dla serwera plików protokołu.
- listFormat jest formatem listingu plików zwróconego z serwera plików protokołu.
- Produkt limitedWrite określa, czy podczas zapisu do serwera plików ma być używany tryb domyślny, który ma utworzyć plik tymczasowy, a następnie zmienić nazwę tego pliku po zakończeniu przesyłania. W przypadku serwera plików, który jest skonfigurowany tylko jako zapis, plik jest tworzony bezpośrednio z jego nazwą końcową. Wartość tej właściwości może mieć wartość true (prawda) lub false (fałsz). Atrybut limitedWrite i właściwość agenta doNotUseTempOutputFile są używane razem w przypadku agentów mostu protokołu. Jeśli mają być używane pliki tymczasowe, nie należy ustawiać wartości parametru doNotUseTempOutputFile, a wartość parametru limitedWrite należy ustawić na wartość false. Każda inna kombinacja ustawień oznacza, że pliki tymczasowe nie będą używane.

- `maxListFileNames` to maksymalna liczba zebranych nazw podczas skanowania katalogu na serwerze plików protokołu dla nazw plików.
- `maxListDirectoryLevels` to maksymalna liczba poziomów katalogów, które mają być rekurencyjne podczas skanowania katalogu na serwerze plików protokołu dla nazw plików.

Więcej szczegółowych informacji na temat tych atrybutów, w tym informacje o tym, czy są one wymagane, czy opcjonalne, oraz ich wartości domyślne zawiera sekcja [Format pliku właściwości mostu protokołu](#).

Odsyłacze pokrewne

[Format pliku właściwości mostu protokołu](#)

[Wyrażenia regularne używane przez MFT](#)

Wyszukiwanie właściwości serwera plików protokołu: ProtocolBridgePropertiesExit2

Jeśli istnieje duża liczba serwerów plików protokołu, można zaimplementować interfejs `com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit2` w celu wyszukania właściwości serwera plików protokołu, które są przywoływane w transferach. Ten interfejs można zaimplementować w preferencjach w celu zachowania pliku `ProtocolBridgeProperties.xml`.

O tym zadaniu

Produkt Managed File Transfer udostępnia przykładowe wyjście użytkownika, które wyszukuje właściwości serwera plików protokołu. Więcej informacji na ten temat zawiera sekcja [“Korzystanie z przykładowego programu zewnętrznego do wyszukiwania właściwości serwera plików protokołu” na stronie 276](#).

Każde wyjście użytkownika, które wyszukuje właściwości mostu protokołu, musi implementować interfejs `com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit2`. Więcej informacji na ten temat zawiera dokument [ProtocolBridgePropertiesExit.java interface](#).

Można połączyć wiele właściwości serwera protokołu w podobny sposób, jak inne wyjścia użytkownika. Wyjścia są wywoływane w kolejności, w jakiej są one określane przy użyciu właściwości `protocolBridgePropertiesExitClasses` w pliku właściwości agenta. Wszystkie metody inicjowania są zwracane oddzielnie, a jeśli jeden lub więcej zwraca wartość `false`, agent nie uruchamia się. Błąd jest zgłaszany w dzienniku zdarzeń agenta.

Tylko jeden wynik ogólny jest zwracany dla metod `getProtocolServerProperties` wszystkich wyjść. Jeśli metoda zwraca obiekt właściwości jako kod wyniku, ta wartość jest zwróconego wyniku, a metody `getProtocolServerProperties` dla kolejnych wyjść nie są wywoływane. Jeśli metoda zwraca wartość `null` jako kod wyniku, wywoływana jest metoda `getProtocolServerProperties` następnego wyjścia. Jeśli nie ma kolejnego wyjścia, zwracany jest wynik zerowy. Ogólny kod wyniku o wartości `NULL` jest uznawany za niepowodzenie wyszukiwania przez agenta mostu protokołu.

Zalecane jest użycie interfejsu `ProtocolBridgePropertiesExit2.java`, ale w celu uzyskania informacji na temat interfejsu `ProtocolBridgePropertiesExit.java`, należy zapoznać się z informacjami w sekcji [“Wyszukiwanie właściwości serwera plików protokołu: ProtocolBridgePropertiesExit” na stronie 277](#).

Aby uruchomić wyjście, wykonaj następujące kroki:

Procedura

1. Skompiluj wyjście użytkownika właściwości serwera protokołu.
2. Utwórz plik archiwum Java (JAR) zawierający skompilowany program obsługi wyjścia i jego strukturę pakietu.
3. Umieść plik JAR zawierający klasę wyjścia w katalogu `exits` agenta mostu protokołu. Katalog ten znajduje się w katalogu `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name`.

4. Zmodyfikuj plik właściwości agenta mostu protokołu, aby zawierał właściwość `protocolBridgePropertiesExitClasses`. W przypadku wartości tej właściwości należy określić rozdzielaną przecinkami listę klas, które implementują wyjście użytkownika właściwości serwera mostu protokołu. Klasy wyjścia są wywoływane w kolejności, w jakiej są określone na tej liście. Więcej informacji na ten temat zawiera sekcja [Plik MFT agent.properties](#).
5. Opcjonalnie można określić właściwość `protocolBridgePropertiesConfiguration`. Wartość określona dla tej właściwości jest przekazywana w postaci łańcucha do metody `initialize()` klas wyjścia określonych przez produkt `protocolBridgePropertiesExitClasses`. Więcej informacji na ten temat zawiera sekcja [Plik MFT agent.properties](#).

Korzystanie z przykładowego programu zewnętrznego do wyszukiwania właściwości serwera plików protokołu

Produkt Managed File Transfer udostępnia przykładowe wyjście użytkownika, które wyszukuje właściwości serwera plików protokołu.

O tym zadaniu

Przykładowe wyjście użytkownika, które wyszukuje właściwości mostu protokołu, znajduje się w katalogu `MQ_INSTALLATION_PATH/mqft/samples/protocolBridge` i w temacie [Przykładowy program obsługi wyjścia właściwości mostu protokołu](#).

Program zewnętrzny `SamplePropertiesExit2.java` odczytuje plik właściwości, który zawiera właściwości serwerów protokołów. Format każdej pozycji w pliku właściwości jest następujący:

```
serverName=type://host:port
```

Położenie pliku właściwości jest pobierane z właściwości agenta mostu protokołu `protocolBridgePropertiesConfiguration`.

Aby uruchomić przykładowe wyjście użytkownika, wykonaj następujące kroki:

Procedura

1. Skompiluj plik `SamplePropertiesExit2.java`.
2. Utwórz plik JAR zawierający skompilowane wyjście i jego strukturę pakietu.
3. Umieść plik JAR w katalogu `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent/exits`.
4. Zmodyfikuj plik `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name/agent.properties`, tak aby zawierał wiersz:

```
protocolBridgePropertiesExitClasses=SamplePropertiesExit2
```

5. Utwórz plik właściwości mostu protokołu, na przykład `protocol_bridge_properties.properties`, w katalogu `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent`. Zmodyfikuj ten plik, tak aby zawierał wpisy w formacie:

```
serverName=type://host:port
```

6. Zmodyfikuj plik `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent/agent.properties`, tak aby zawierał wiersz:

```
protocolBridgePropertiesConfiguration=MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent/protocol_bridge_properties.properties
```

Należy użyć ścieżki bezwzględnej do pliku `protocol_bridge_properties.properties`.

7. Uruchom agenta mostu protokołu za pomocą komendy `fteStartAgent`.

Pojęcia pokrewne

[“Most protokołu” na stronie 270](#)

Most protokołu umożliwia sieci Managed File Transfer (MFT) dostęp do plików przechowywanych na serwerze plików poza siecią produktu MFT , zarówno w domenie lokalnej, jak i w zdalnej lokalizacji. Ten serwer plików może korzystać z protokołów sieciowych FTP, FTPS lub SFTP. Każdy serwer plików wymaga co najmniej jednego dedykowanego agenta. Dedykowany agent jest znany jako agent mostu protokołu. Agent mostu może wchodzić w interakcje z wieloma serwerami plików.

Odsyłacze pokrewne

[Interfejs ProtocolBridgePropertiesExit.java](#)

[Wyjście użytkownika właściwości mostu przykładowego protokołu](#)

[Plik MFT agent.properties](#)

[fteCreateBridgeAgent \(tworzenie i konfigurowanie agenta mostu protokołu MFT\)](#)

Wyszukiwanie właściwości serwera plików protokołu: ProtocolBridgePropertiesExit

Jeśli istnieje duża liczba serwerów plików protokołu, można zaimplementować interfejs `com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit` w celu wyszukania właściwości serwera plików protokołu, które są przywoływane w transferach.

O tym zadaniu

Interfejs `com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit` można zaimplementować w preferencjach w celu zachowania pliku `ProtocolBridgeProperties.xml` . Zalecane jest użycie interfejsu `ProtocolBridgePropertiesExit2.java` , ale obsługiwany jest również interfejs `ProtocolBridgePropertiesExit.java`. Jeśli istnieje implementacja interfejsu `ProtocolBridgePropertiesExit.java` z produktu IBM WebSphere MQ File Transfer Edition, można użyć go w produkcie IBM WebSphere MQ 7.5 lub nowszym. Metoda `getCredentialLocation` w pliku `ProtocolBridgePropertiesExit2.java` korzysta z domyślnego położenia pliku `ProtocolBridgeCredentials.xml` , który jest katalogiem osobistym użytkownika.

Uwaga: IBM WebSphere MQ File Transfer Edition (FTE) nie jest już obsługiwany produktem. Informacje o migrowaniu FTE do składnika Managed File Transfer w programie IBM MQ można znaleźć w sekcji [Migrowanie produktu Managed File Transfer](#).

Każde wyjście użytkownika, które wyszukuje właściwości mostu protokołu, musi implementować interfejs `com.ibm.wmqfte.exitroutine.api.ProtocolBridgePropertiesExit`:

```
package com.ibm.wmqfte.exitroutine.api;

import java.util.Map;
import java.util.Properties;

/**
 * An interface that is implemented by classes that are to be invoked as part of
 * user exit routine processing. This interface defines methods that will be
 * invoked by a protocol bridge agent to look up properties for protocol servers
 * that are referenced in transfers.
 * <p>
 * There will be one instance of each implementation class for each protocol
 * bridge agent. The methods can be called from different threads so the methods
 * must be synchronised.
 */
public interface ProtocolBridgePropertiesExit {

    /**
     * Invoked once when a protocol bridge agent is started. It is intended to
     * initialize any resources that are required by the exit.
     *
     * @param bridgeProperties
     *         The values of properties defined for the protocol bridge.
     *         These values can only be read, they cannot be updated by the
     *         implementation.
     * @return {@code true} if the initialization is successful and {@code
     *         false} if unsuccessful. If {@code false} is returned from an exit
     *         the protocol bridge agent will not start.
     */
}
```

```

public boolean initialize(final Map<String, String> bridgeProperties);

/**
 * Obtains a set of properties for the specified protocol server name.
 * <p>
 * The returned {@link Properties} must contain entries with key names
 * corresponding to the constants defined in
 * {@link ProtocolServerPropertyConstants} and in particular must include an
 * entry for all appropriate constants described as required.
 *
 * @param protocolServerName
 *     The name of the protocol server whose properties are to be
 *     returned. If a null or a blank value is specified, properties
 *     for the default protocol server are to be returned.
 * @return The {@link Properties} for the specified protocol server, or null
 *     if the server cannot be found.
 */
public Properties getProtocolServerProperties(
    final String protocolServerName);

/**
 * Invoked once when a protocol bridge agent is shut down. It is intended to
 * release any resources that were allocated by the exit.
 *
 * @param bridgeProperties
 *     The values of properties defined for the protocol bridge.
 *     These values can only be read, they cannot be updated by the
 *     implementation.
 */
public void shutdown(final Map<String, String> bridgeProperties);
}

```

Można połączyć wiele właściwości serwera protokołu w podobny sposób, jak inne wyjścia użytkownika. Wyjścia są wywoływane w kolejności, w jakiej są one określone przy użyciu właściwości `protocolBridgePropertiesExitClasses` w pliku właściwości agenta. Wszystkie metody inicjowania są zwracane oddzielnie, a jeśli jeden lub więcej zwraca wartość `false`, agent nie uruchamia się. Błąd jest zgłaszany w dzienniku zdarzeń agenta.

Tylko jeden wynik ogólny jest zwracany dla metod `getProtocolServerProperties` wszystkich wyjść. Jeśli metoda zwraca obiekt właściwości jako kod wyniku, ta wartość jest zwróconego wyniku, a metody `getProtocolServerProperties` dla kolejnych wyjść nie są wywoływane. Jeśli metoda zwraca wartość `null` jako kod wyniku, wywoływana jest metoda `getProtocolServerProperties` następnego wyjścia. Jeśli nie ma kolejnego wyjścia, zwracany jest wynik zerowy. Ogólny kod wyniku o wartości `NULL` jest uznawany za niepowodzenie wyszukiwania przez agenta mostu protokołu.

Procedura

Aby uruchomić wyjście, wykonaj następujące kroki:

1. Skompiluj wyjście użytkownika właściwości serwera protokołu.
2. Utwórz plik archiwum Java (JAR) zawierający skompilowany program obsługi wyjścia i jego strukturę pakietu.
3. Umieść plik JAR zawierający klasę wyjścia w katalogu `exits` agenta mostu protokołu.
Katalog ten znajduje się w katalogu `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name`.
4. Zmodyfikuj plik właściwości agenta mostu protokołu, aby zawierał właściwość `protocolBridgePropertiesExitClasses`.
W przypadku wartości tej właściwości należy określić rozdzielaną przecinkami listę klas, które implementują wyjście użytkownika właściwości serwera mostu protokołu. Klasy wyjścia są wywoływane w kolejności, w jakiej są określone na tej liście. Więcej informacji na ten temat zawiera sekcja [Plik MFT agent.properties](#).
5. Opcjonalnie można określić właściwość `protocolBridgePropertiesConfiguration`.
Wartość określona dla tej właściwości jest przekazywana w postaci łańcucha do metody `initialize()` klas wyjścia określonych przez produkt `protocolBridgePropertiesExitClasses`. Więcej informacji na ten temat zawiera sekcja [Plik MFT agent.properties](#).

Odwzorowywanie informacji autoryzacyjnych dla serwera plików

Odwzoruj informacje autoryzacyjne użytkownika w programie Managed File Transfer na referencje użytkownika na serwerze plików za pomocą domyślnej funkcji odwzorowania referencji agenta mostu protokołu lub poprzez napisanie własnego wyjścia użytkownika. Produkt Managed File Transfer udostępnia przykładowe wyjście użytkownika, które wykonuje odwzorowanie referencji użytkownika.

Odwzorowywanie informacji autoryzacyjnych dla serwera plików za pomocą pliku `ProtocolBridgeCredentials.xml`

Odwzoruj informacje autoryzacyjne użytkownika w programie Managed File Transfer na referencje użytkownika na serwerze plików przy użyciu domyślnej funkcji odwzorowania referencji agenta mostu protokołu. Produkt Managed File Transfer udostępnia plik XML, który można edytować w celu uwzględnienia informacji autoryzacyjnych.

O tym zadaniu

Plik `ProtocolBridgeCredentials.xml` musi zostać ręcznie utworzony przez użytkownika. Domyślnie położenie tego pliku to katalog osobisty użytkownika, który uruchomił agent mostu protokołu, ale może on być przechowywany w dowolnym miejscu w systemie plików dostępnym dla agenta. Aby określić inne położenie, należy dodać element `<credentialsFile>` do pliku `ProtocolBridgeProperties.xml`. Na przykład składnia

```
<tns:credentialsFile path="/example/path/to/ProtocolBridgeCredentials.xml"/>
```

Przed użyciem agenta mostu protokołu skonfiguruj odwzorowanie referencji, edytując ten plik w celu uwzględnienia informacji o hoście, użytkownika i informacji autoryzacyjnych. Więcej informacji i przykłady można znaleźć w sekcji [Format pliku referencji mostu protokołu](#).

z/OS Podczas tworzenia pliku `ProtocolBridgeCredentials.xml` na platformie z/OS przy użyciu produktu IBM WebSphere MQ 7.5 lub wcześniejszej należy ustawić znacznik pliku przed rozpoczęciem edycji pliku. Uruchom następującą komendę, aby oznaczyć plik jako posiadający treść ASCII:

```
chtag -t -c IS08859-1 ProtocolBridgeCredentials.xml
```

Uwaga: **z/OS** W systemie z/OS można przechowywać plik referencji mostu protokołu w zestawie danych, w którym nazwa pliku `.xml` może być określona przez użytkownika.

Procedura

1. Zmodyfikuj wiersz `<tns:server name="server name">`, aby zmienić wartość atrybutu nazwy na nazwę serwera w pliku `ProtocolBridgeProperties.xml`.

Agenty mostu protokołu utworzone dla produktu IBM WebSphere MQ File Transfer Edition 7.0.4 i wcześniejszych nie mają pliku `ProtocolBridgeProperties.xml` (lub pokrewnych programów zewnętrznych), dlatego w przypadku systemu IBM WebSphere MQ File Transfer Edition 7.0.4 Fix Pack 1 i nowszych nazwa serwera jest automatycznie przypisywany do nazwy hosta serwera. Oznacza to, że jeśli zostanie użyty zaktualizowany plik `ProtocolBridgeCredentials.xml` z pozycjami `<server>`, zostanie dopasowana nazwa odpowiadająca nazwie hosta serwera.

Za pomocą atrybutu `pattern` można określić, że używana jest nazwa serwera, która zawiera znaki wieloznaczne lub wyrażenia regularne. Na przykład składnia

```
<tns:server name="serverA*" pattern="wildcard">
```

2. Wstaw identyfikator użytkownika i informacje autoryzacyjne do pliku jako elementy potomne produktu `<tns:server>`.

Do pliku można wstawić jeden lub wiele z następujących elementów:

- Jeśli serwer plików protokołu jest serwerem FTP, FTPS lub SFTP, można użyć haseł do uwierzytelniania użytkownika żądającego transferu. Wstaw następujące wiersze do pliku:

```
<tns:user name="FTE User ID"
  serverUserId="Server User ID"
  serverPassword="Server Password">
</tns:user>
```

Następnie należy zmienić wartość atrybutów.

- name jest wyrażeniem regularnym Java , które jest zgodne z identyfikatorem użytkownika MQMD powiązany z żądaniem transferu MFT .
- serverUserId jest wartością, która jest przekazywana do serwera plików protokołu jako identyfikator użytkownika logowania. Jeśli atrybut serverUserId nie jest określony, zamiast tego używany jest identyfikator użytkownika MQMD powiązany z żądaniem przesyłania MFT .
- serverPassword jest hasłem, które jest powiązane z serverUserId.

Atrybut name może zawierać wyrażenie regularne Java . Program odwzorowujący referencje próbuje dopasować identyfikator użytkownika MQMD żądania przesyłania MFT do tego wyrażenia regularnego. Agent mostu protokołu próbuje dopasować identyfikator użytkownika MQMD do wyrażenia regularnego w atrybucie name elementów < tns: user> w kolejności, w jakiej występują elementy w pliku. Jeśli zostanie znaleziony zgodny agent mostu protokołu, nie będzie on poszukiwać więcej dopasowań. Jeśli zostanie znalezione dopasowanie, odpowiadające im wartości serverUserId i serverPassword są przekazywane do serwera plików protokołu jako identyfikator użytkownika i hasło logowania. W przypadku zgodności z ID użytkownika MQMD jest rozróżniana wielkość liter.

- Jeśli serwer plików protokołu jest serwerem SFTP, można użyć kluczy publicznych i prywatnych w celu uwierzytelnienia użytkownika żądającego przesyłania. Wstaw następujące wiersze do pliku i zmień wartość atrybutów. Element < tns: user> może zawierać jeden lub wiele elementów < tns:privateKey>.

```
<tns:user name="FTE User ID"
  serverUserId="Server User ID"
  hostKey="Host Key">
  <tns:privateKey associationName="association"
    keyPassword="Private key password">
    Private key file text
  </tns:privateKey>
</tns:user>
```

- name jest wyrażeniem regularnym Java , które jest zgodne z identyfikatorem użytkownika MQMD powiązany z żądaniem transferu MFT .
- serverUserId jest wartością, która jest przekazywana do serwera plików protokołu jako identyfikator użytkownika logowania. Jeśli atrybut serverUserId nie jest określony, zamiast tego używany jest identyfikator użytkownika MQMD powiązany z żądaniem przesyłania MFT .
- hostKey jest oczekiwanym kluczem, który jest zwracany z serwera podczas logowania
- key jest kluczem prywatnym serverUserId
- keyPassword to hasło klucza do generowania kluczy publicznych.
- associationName jest wartością, która jest używana do identyfikowania celu śledzenia i rejestrowania.

Atrybut name może zawierać wyrażenie regularne Java . Program odwzorowujący referencje próbuje dopasować identyfikator użytkownika MQMD żądania przesyłania MFT do tego wyrażenia regularnego. Agent mostu protokołu próbuje dopasować identyfikator użytkownika MQMD do wyrażenia regularnego w atrybucie name elementów < tns: user> w kolejności, w jakiej występują elementy w pliku. Jeśli zostanie znaleziony zgodny agent mostu protokołu, nie będzie on poszukiwać więcej dopasowań. Jeśli zostanie znalezione dopasowanie, odpowiadające im wartości serverUserId i key są używane do uwierzytelnienia użytkownika MFT z serwerem plików protokołu. W przypadku zgodności z ID użytkownika MQMD jest rozróżniana wielkość liter.

Więcej informacji na temat używania kluczy prywatnych z agentem mostu protokołu znajduje się w sekcji [“Przykład: sposób konfigurowania agenta mostu protokołu w celu użycia referencji klucza prywatnego z serwerem SFTP UNIX”](#) na stronie 283.

Uwaga: 

Gdy żądanie przesyłania zostanie zapisane w kolejce komend, identyfikator użytkownika MQMD może zostać przekształcony na wielkie litery, jeśli kolejka komend agenta źródłowego znajduje się w systemie z/OS lub IBM i . W wyniku tego identyfikator użytkownika MQMD dla tego samego użytkownika źródłowego może zostać dostarczony do wyjścia referencji w oryginalnej wielkości liter lub zostać przekształcony na wielkie litery w zależności od agenta źródłowego określonego w żądaniu transferu. Domyślne wyjście odwzorowania referencji przeprowadza rozróżnianie wielkości liter w odniesieniu do podanego identyfikatora użytkownika MQMD, który może być potrzebny w pliku odwzorowania.

Odwzorowywanie informacji autoryzacyjnych dla serwera plików za pomocą klas wyjścia

Jeśli nie ma być używana domyślna funkcja odwzorowania referencji agenta mostu protokołu, można odwzorować referencje użytkownika w programie Managed File Transfer na referencje użytkownika na serwerze plików, zapisując własne wyjście użytkownika. Jeśli zostanie skonfigurowane wyjście użytkownika odwzorowania referencji, zostaną one umieszczone w miejscu domyślnej funkcji odwzorowania referencji.

O tym zadaniu

Produkt Managed File Transfer udostępnia przykładowe wyjście użytkownika, które wykonuje odwzorowanie referencji użytkownika. Więcej informacji na ten temat zawiera sekcja [“Korzystanie z przykładowego programu obsługi wyjścia referencji mostu protokołu”](#) na stronie 282.

Wyjście użytkownika służące do odwzorowywania referencji mostu protokołu musi implementować jeden z następujących interfejsów:

- `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit`, który umożliwia agentowi mostu protokołu przesyłanie plików do i z jednego domyślnego serwera plików protokołu.
- `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit2`, który umożliwia przesyłanie plików do i z wielu punktów końcowych.

Interfejs `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit2` zawiera tę samą funkcję, co `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit`, a ponadto zawiera funkcję rozszerzoną. Więcej informacji na ten temat zawierają pliki [ProtocolBridgeCredentialExit.java](#) interface i [ProtocolBridgeCredentialExit2.java](#) interface.

Wyjścia referencji mogą być połączone w podobny sposób do innych wyjść użytkownika. Wyjścia są wywoływane w kolejności, w jakiej są one określane przy użyciu właściwości `protocolBridgeCredentialConfiguration` w pliku właściwości agenta. Wszystkie metody inicjowania są zwracane oddzielnie, a jeśli jeden lub więcej zwraca wartość `false`, agent nie uruchamia się. Błąd jest zgłaszany w dzienniku zdarzeń agenta.

Tylko jeden wynik ogólny jest zwracany dla metod `mapMQUserId` wszystkich wyjść w następujący sposób:

- Jeśli metoda zwraca wartość `USER_SUCCESSFULLY_MAPPED` lub `USER_DENIED_ACCESS` jako kod wyniku, ta wartość jest zwróconego wyniku, a metody `mapMQUserId` kolejnych wyjść nie są wywoływane.
- Jeśli metoda zwraca wartość `NO_MAPPING_FOUND` jako kod wyniku, wywoływana jest metoda `mqMQUsero` identyfikatorze następnego wyjścia.
- Jeśli nie ma kolejnego wyjścia, zwracany jest wynik `NO_MAPPING_FOUND`.
- Ogólny kod wyniku `USER_DENIED_ACCESS` lub `NO_MAPPING_FOUND` jest traktowany jako niepowodzenie transferu przez agent mostu.

Aby uruchomić wyjście, wykonaj następujące kroki:

Procedura

1. Skompiluj wyjście użytkownika referencji mostu protokołu.
2. Utwórz plik archiwum Java (JAR) zawierający skompilowany program obsługi wyjścia i jego strukturę pakietu.
3. Umieść plik JAR, który zawiera klasę wyjścia w katalogu `exits` agenta mostu. Katalog znajduje się w katalogu `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name`.
4. Zmodyfikuj plik właściwości agenta mostu protokołu, aby zawierał właściwość `protocolBridgeCredentialExitClasses`. W przypadku wartości tej właściwości należy określić rozdzielaną przecinkami listę klas, które implementują procedurę wyjścia referencji mostu protokołu. Klasy wyjścia są wywoływane w kolejności, w jakiej są określone na tej liście. Więcej informacji na ten temat zawiera sekcja [Plik MFT agent.properties](#).
5. Zmodyfikuj plik właściwości agenta mostu protokołu, aby zawierał następujące elementy:

```
exitClassPath=IBM MQ
installation_directory\mqft\config\configuration_queue_manager\agents\protocol_bridge_agent_name\exits\SampleCredentialExit.jar
```

Plik `agent.properties` dla agenta znajduje się w katalogu `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/bridge_agent_name`.

Jeśli zostanie zmieniony plik `agent.properties`, należy zrestartować agenta, aby zmiany zostały wprowadzone.

6. Opcjonalnie można określić właściwość `protocolBridgeCredentialConfiguration`. Wartość określona dla tej właściwości jest przekazywana jako obiekt typu `String` do metody `initialize()` klasy wyjścia określonych przez produkt `protocolBridgeCredentialExitClasses`. Więcej informacji na ten temat zawiera sekcja [Plik MFT agent.properties](#).
7. Uruchom agenta mostu protokołu za pomocą komendy **`fteStartAgent`**.

Korzystanie z przykładowego programu obsługi wyjścia referencji mostu protokołu

Produkt Managed File Transfer udostępnia przykładowe wyjście użytkownika, które wykonuje odwzorowanie referencji użytkownika.

O tym zadaniu

Przykładowe wyjście referencji mostu protokołu jest dostępne w katalogu `MQ_INSTALLATION_PATH/mqft/samples/protocolBridge` i w temacie [Sample protocol bridge credential user exit](#) (Wyjście użytkownika referencji mostu protokołu). Ten przykład jest oparty na interfejsie produktu `com.ibm.wmqfte.exitroutine.api.ProtocolBridgeCredentialExit`.

Program zewnętrzny `SampleCredentialExit.java` odczytuje plik właściwości, który odwzorowuje identyfikatory użytkowników MQMD powiązane z żądaniami przesyłania na identyfikatory użytkowników serwera i hasła serwera. Położenie pliku właściwości jest pobierane z właściwości agenta mostu protokołu `protocolBridgeCredentialConfiguration`.

Aby uruchomić przykładowe wyjście użytkownika, wykonaj następujące kroki:

Procedura

1. Skompiluj plik `SampleCredentialExit.java`.
2. Utwórz plik JAR zawierający skompilowane wyjście i jego strukturę pakietu.
3. Umieść plik JAR w katalogu `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name/exits`.
4. Zmodyfikuj plik `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name/agent.properties`, tak aby zawierał wiersz:

```
protocolBridgeCredentialExitClasses=SampleCredentialExit
```

5. Zmodyfikuj plik właściwości agenta mostu protokołu, aby zawierał następujące elementy:

```
exitClassPath=IBM MQ
installation_directory\mqft\config\configuration_queue_manager\agents\protocol_bridge_agent_name\exits\SampleCredentialExit.jar
```

Plik `agent.properties` dla agenta znajduje się w katalogu `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name`.

Jeśli zostanie zmieniony plik `agent.properties`, należy zrestartować agenta, aby zmiany zostały wprowadzone.

6. Utwórz plik właściwości referencji (`credentials.properties`) w katalogu `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent` i zmodyfikuj go, tak aby zawierał wpisy w formacie:

```
mqUserId=serverUserId,serverPassword
```

7. Zmodyfikuj plik `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name/agent.properties`, tak aby zawierał wiersz:

```
protocolBridgeCredentialConfiguration=MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/bridge_agent_name/credentials.properties
```

Należy użyć ścieżki bezwzględnej do pliku `credentials.properties`.

8. Uruchom agenta mostu protokołu za pomocą komendy **fteStartAgent**.

Przykład: sposób konfigurowania agenta mostu protokołu w celu użycia referencji klucza prywatnego z serwerem SFTP UNIX

W tym przykładzie pokazano, w jaki sposób można wygenerować i skonfigurować plik `ProtocolBridgeCredentials.xml`. Ten przykład jest typowym przykładem, a szczegóły mogą się różnić w zależności od używanej platformy, ale zasady pozostają takie same.

O tym zadaniu

Procedura

1. Wygeneruj klucz publiczny i prywatny, który ma być używany do uwierzytelniania na serwerze SFTP. Na przykład w systemie hosta Linux można użyć narzędzia **ssh-keygen**, które jest dostarczane jako część pakietu 'openssh', w celu utworzenia pary kluczy publiczny/prywatny. Domyślnie, bez argumentów, komenda **ssh-keygen** pyta o położenie i frazę hasła dla dwóch plików kluczy, które domyślnie są nazwami:

```
id_rsa      <-- Private key
id_rsa.pub  <-- Public key
```



Ostrzeżenie: Jeśli używana jest komenda **ssh-keygen** z najnowszej wersji programu OpenSSH, na przykład dostarczona z systemem RHEL 8, używany format klucza nie jest zgodny z agentem mostu protokołu, a próby przeniesienia do serwera SFTP nie powiodą się z komunikatem:

```
BFGBR0216E: Authentication to protocol server 'sftp.host.address' failed
because of invalid private key.
```

Aby utworzyć zgodny klucz prywatny z tymi nowszymi wersjami pakietu OpenSSH, należy podać format klucza z następującym argumentem komendy **ssh-keygen** :

```
ssh-keygen -m PEM
```

Następnie zawartość klucza prywatnego `id_rsa` będzie zawierać pierwsze i ostatnie wiersze:

```
-----BEGIN RSA PRIVATE KEY-----  
.....  
-----END RSA PRIVATE KEY-----
```

co jest zgodne z agentem mostu protokołu.

2. Skopiuj całą zawartość pliku `id_rsa.pub` do pliku `~/.ssh/authorized_keys` użytkownika SFTP na serwerze SFTP.

Upewnij się, że uprawnienia do pliku w tym pliku oraz katalog `~/.ssh` są odpowiednio ustawione dla serwera SFTP, aby umożliwić uwierzytelnianie za pomocą klucza. Te uprawnienia są zwykle następujące:

```
~/.ssh          Mode 700  
~/.ssh/authorized_keys  Mode 600
```

3. Program Managed File Transfer wymaga, aby za pomocą algorytmu MD5 został wygenerowany odcisk hosta ssh hosta. Uruchom jedną z następujących komend, aby uzyskać dostęp do odcisku ssh hosta serwera SFTP.

- W przypadku produktu Red Hat® Enterprise Linux w wersji 6.x i poniżej oraz Linux Ubuntu 14.04 uruchom następującą komendę:

```
ssh-keygen -l -f /etc/ssh/ssh_host_rsa_key.pub
```

- Począwszy od wersji Red Hat Enterprise Linux 7.x, Linux Ubuntu 16.04 i SuSE Linux 12.4, komenda `ssh-keygen` generuje domyślnie odcisk klucza ssh przy użyciu algorytmu SHA256 . Aby wygenerować odcisk ssh przy użyciu algorytmu MD5 , uruchom następującą komendę:

```
ssh-keygen -l -E MD5 -f /etc/ssh/ssh_host_rsa_key.pub
```

Dane wyjściowe komendy będą podobne do poniższego przykładu:

```
2048 MD5:64:39:f5:49:41:10:55:d2:0b:81:42:5c:87:62:9d:27 no comment (RSA)
```

Wyodrębni część szesnastkową tylko danych wyjściowych, które mają być używane jako `hostKey` w pliku `ProtocolBridgeCredentials.xml` (patrz krok "4" na [stronie 284](#)). W związku z tym w tym przykładzie wyodrębniłby się produkt `64:39:f5:49:41:10:55:d2:0b:81:42:5c:87:62:9d:27`.

4. W systemie agenta mostu protokołu zmodyfikuj plik `ProtocolBridgeCredentials.xml` . Zastąp wartości przedstawione kursywą w poniższym przykładzie, podając własne wartości:

```
<tns:credentials xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeCredentials"  
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"  
xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeCredentials  
ProtocolBridgeCredentials.xsd ">  
  
<tns:agent name="Agent_name">  
  
<tns:server name="SFTP_name">  
  
<tns:user name="mq_User_ID" serverUserId="SFTP_user_ID"  
hostKey="ssh_host_finger">  
<tns:privateKey associationName="name" keyPassword="pass_phrase">  
Complete contents of the id_rsa file including the entries  
-----BEGIN RSA PRIVATE KEY-----  
  
-----END RSA PRIVATE KEY-----  
</tns:privateKey>  
</tns:user>
```

```
</tns:server>
</tns:agent>
</tns:credentials>
```

gdzie:

- *nazwa_agenta* to nazwa agenta mostu protokołu.
- *SFTP_host_name* to nazwa serwera SFTP, która jest wyświetlana w pliku `ProtocolBridgeProperties.xml`.
- *mq_ID_użytkownika* to identyfikator użytkownika MQMD powiązany z żądaniem przesłania.
- *SFTP_user_ID* to identyfikator użytkownika SFTP, który został użyty w kroku 2. Jest to wartość przekazana do SFTP jako identyfikator użytkownika logowania.
- *ssh_host_finger* jest odciskiem pobranym w kroku 3.
- *nazwa* to nazwa, która może być używana na potrzeby śledzenia i rejestrowania.
- *fra_hasła* jest frazą przekazanej w kroku 1 w programie `ssh-keygen`.
- *Pełna treść pliku id_rsa* to pełna treść wygenerowanego pliku `id_rsa` z kroku 1. Aby zapobiec wystąpieniu błędu połączenia, należy uwzględnić oba następujące wpisy:

```
-----BEGIN RSA PRIVATE KEY-----
-----END RSA PRIVATE KEY-----
```

Aby dodać dodatkowe klucze, należy zduplikować element `<tns:privatekey>`.

5. Jeśli agent nie jest jeszcze uruchomiony, uruchom agenta mostu protokołu. Alternatywnie agent mostu protokołu okresowo odpytuje plik `ProtocolBridgeCredentials.xml` i pobiera zmiany.

Konfigurowanie mostu protokołu dla serwera FTPS

Skonfiguruj serwer FTPS w podobny sposób, jak konfigurowany jest serwer FTP: utwórz agenta mostu dla serwera, zdefiniuj właściwości serwera i odwzoruj referencje użytkownika.

O tym zadaniu

Aby skonfigurować serwer FTPS, wykonaj następujące kroki:

Procedura

1. Utwórz agent mostu protokołu dla serwera FTPS za pomocą komendy **fteCreateBridgeAgent**. Parametry, które mają zastosowanie do FTP mają zastosowanie także do protokołu FTPS, ale istnieją również trzy wymagane parametry specyficzne dla protokołu FTPS:
 - a) Parametr **-bt**. Jako wartość tego parametru należy podać wartość FTPS.
 - b) Parametr **-bts** dla pliku zaufanych certyfikatów. W komendzie założono, że wymagane jest tylko uwierzytelnianie serwera, a użytkownik musi określić położenie pliku zaufanych certyfikatów.

Jawna forma protokołu FTPS jest domyślnie skonfigurowana za pomocą komendy **fteCreateBridgeAgent**, ale można skonfigurować niejawną formę poprzez zmianę pliku właściwości mostu protokołu. Most protokołu zawsze łączy się z serwerami FTPS w trybie pasywnym.

Więcej informacji na temat komendy **fteCreateBridgeAgent** można znaleźć w sekcji [fteCreateBridgeAgent \(tworzenie i konfigurowanie agenta mostu protokołu MFT\)](#).

Aby uzyskać instrukcje dotyczące tworzenia plików zaufanych certyfikatów, należy zapoznać się z artykułem IBM Developer ([Konfigurowanie połączeń SSL \(Secure Sockets Layer\) w produkcie IBM WebSphere MQ File Transfer Edition](#)) lub zapoznać się z informacjami na temat narzędzia `keytool` w [Dokumentacja narzędzia klawiszy Oracle](#).

Uwaga: IBM WebSphere MQ File Transfer Edition (FTE) nie jest już obsługiwany produktem. Informacje o migrowaniu FTE do składnika Managed File Transfer w programie IBM MQ można znaleźć w sekcji [Migrowanie produktu Managed File Transfer](#).

2. Zdefiniuj właściwości serwera FTPS w obrębie elementu `<ftpsServer>` w pliku właściwości mostu protokołu: `ProtocolBridgeProperties.xml`. Więcej informacji na ten temat zawiera sekcja [“Definiowanie właściwości dla serwerów plików protokołu przy użyciu pliku ProtocolBridgeProperties.xml” na stronie 272](#). Uwierzytelnianie klienta można również włączyć, edytując plik właściwości mostu protokołu. Szczegółowe informacje na temat wszystkich opcji konfiguracyjnych można znaleźć w sekcji [Format pliku właściwości mostu protokołu](#).
3. Odwzoruj informacje autoryzacyjne użytkownika w programie Managed File Transfer na referencje użytkownika na serwerze FTPS, korzystając z domyślnej funkcji odwzorowania referencji agenta mostu protokołu lub zapisując własne wyjście użytkownika. Więcej informacji na ten temat zawiera sekcja [“Odwzorowywanie informacji autoryzacyjnych dla serwera plików” na stronie 279](#).
4. Domyślnie plik zaufanych certyfikatów jest skonfigurowany jako posiadający format JKS. Jeśli ma zostać zmieniony format, należy zmodyfikować plik właściwości mostu protokołu.

Przykład

Przykładowa pozycja dla serwera FTPS w pliku właściwości mostu protokołu jest wyświetlana w następujący sposób:

```
<tns:serverProperties xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeProperties"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeProperties
  ProtocolBridgeProperties.xsd">
  <tns:defaultServer name="ftpsserver.mycompany.com" />

  <tns:ftpsServer name="ftpsserver.mycompany.com" host="ftpsserver.mycompany.com" port="990"
  platform="windows"
  timeZone="Europe/London" locale="en_US" fileEncoding="UTF8"
  listFormat="unix" limitedWrite="false"
  trustStore="c:\mydirec\truststore.jks" />

  <!-- Define servers here -->
</tns:serverProperties>
```

Co dalej

Informacje na temat obsługiwanych i nieobsługiwanych części protokołu FTPS można znaleźć w sekcji [Obsługa serwera FTPS przez most protokołu](#).

V 9.2.1 Scenariusze i przykłady ograniczania liczby transferów plików do poszczególnych serwerów plików

W jaki sposób poprawiony agent mostu protokołu współpracuje z atrybutami `maxActiveDestinationTransfers` i `failTransferWhenCapacityReached`, wraz z przykładami.

Scenariusze pokazujące pracę agenta mostu protokołu w oparciu o wartość `maxActiveDestinationTransfers`

Scenariusz 1

Plik `ProtocolBridgeProperties.xml` dla agenta mostu protokołu zawiera dwie definicje serwera plików:

- Nie ustawiono atrybutu globalnego `maxActiveDestinationTransfers`.
- Nie ustawiono atrybutu `maxActiveDestinationTransfers` zarówno na serwerze `fileServerA`, jak i w produkcie `FileServerB`.
- Atrybut `maxDestinationTransfers` agenta mostu protokołu został ustawiony na wartość domyślną.

Jeśli atrybut **maxDestinationTransfers** agenta mostu protokołu jest ustawiony na wartość domyślną 25, to:

- Agent docelowy rozpoczyna przetwarzanie dwóch zarządzanych transferów do programu `fileServerA`.
- Zarówno przelewy zakończone.

W tym momencie klient zdaje sobie sprawę z tego, że produkt `fileServerA` uległ awarii i ustawia następujące wartości dla `fileServerA` w pliku `ProtocolBridgeProperties.xml`:

```
maxActiveDestinationTransfers = 0  
failTransferWhenCapacityReached = true
```

- Dla serwera `fileServerA` i kilku dla systemu `fileServerB` pojawia się inny transfer:
Na podstawie właściwości ustawionych w poprzednim kroku, zarządzany transfer do produktu `fileServerA` jest odrzucany i oznaczony jako uszkodzony, podczas gdy przesunięcia dla partycji `fileServerB` są obsługiwane w standardowym istniejącym przepływie.
- Po pewnym czasie klient dowiaduje się, że `fileServerA` działa ponownie, dlatego klient usuwa lub przekształca poprzednio dodaną wartość w `ProtocolBridgeProperties.xml`. Nowy zarządzany transfer jest dostarczany dla `fileServerA` i jest obsługiwany w standardowym istniejącym przepływie.

Scenariusz2

- Ustawiono atrybut **maxActiveDestinationTransfers** dla serwera plików i nie ustawiono atrybutu **failTransferWhenCapacityReached**.
- Agent mostu protokołu działa jako agent docelowy dla tej liczby operacji przesyłania zarządzanych na serwer plików.
- Wartość atrybutu **maxActiveDestinationTransfers** jest zmniejszana o 1.

Agent mostu protokołu dynamicznie aktualizuje swoją konfigurację, a następnie ustawia wartość **maxActiveDestinationTransfers** na nową wartość, gdy jest ona nadal aktywna. Ta aktualizacja nie ma wpływu na transfery zarządzane w toku i są one dozwolone do zakończenia.

Scenariusz 3

Plik `ProtocolBridgeProperties.xml` dla agenta mostu protokołu zawiera dwie definicje serwera plików:

- Nie ustawiono atrybutu globalnego **maxActiveDestinationTransfers**.
- Nie ustawiono atrybutu **failTransferWhenCapacityReached**.
- Użytkownik ustawił **maxActiveDestinationTransfers** na wartość 1 w systemie `fileServerA`.
- Nie ustawiono atrybutu **maxActiveDestinationTransfers** w systemie `fileServerB`.

Jeśli agent mostu protokołu ma atrybut **maxDestinationTransfers** ustawiony na wartość 5:

- Maksymalna liczba aktywnych transferów miejsca docelowego z agenta mostu protokołu do `fileServerA` wynosi 1 (choć agent docelowy ma 5 docelowych gniazd przesyłania, tylko 1 może być używany dla zarządzanych operacji przesyłania do produktu `fileServerA`).

Jest to przydatne w przypadku niepowodzenia `fileServerA`. Po ponownym uruchomieniu programu `fileServerA` wartość **maxActiveDestinationTransfers** może zostać zwiększona do 5, aby umożliwić pełne wykorzystanie dozwolonych transferów docelowych.

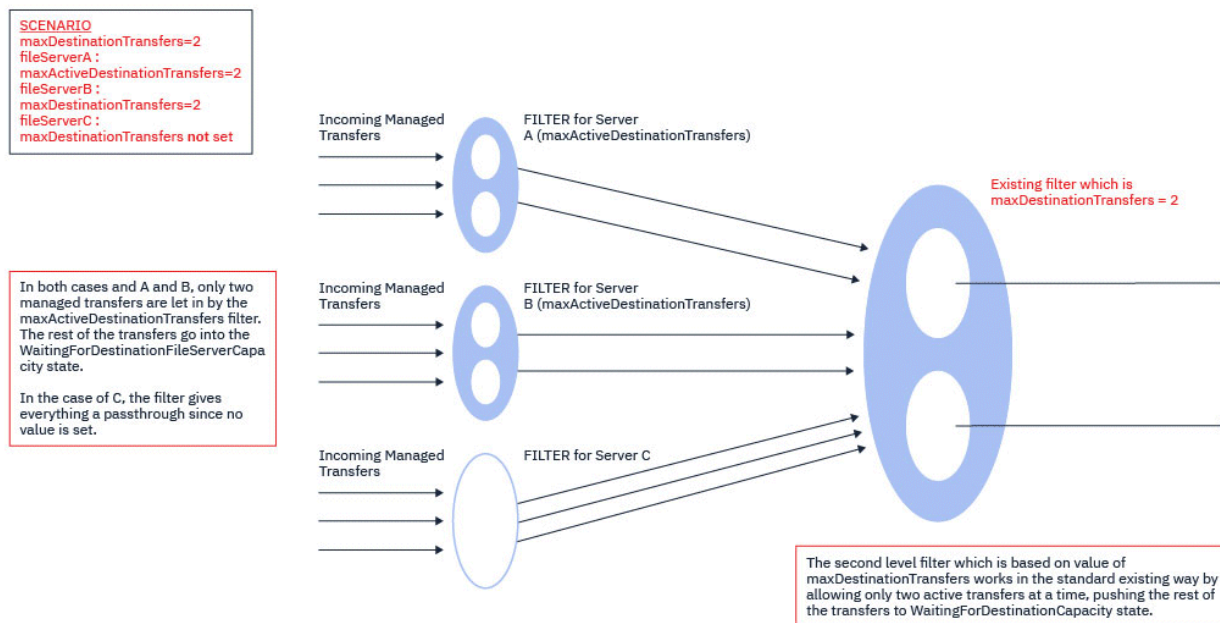
- Maksymalna liczba aktywnych transferów docelowych z agenta mostu protokołu do `fileServerB` wynosi 5.

Ponieważ produkt **maxActiveDestinationTransfers** nie jest ustawiony dla tego serwera plików, agent mostu protokołu może używać wszystkich 5 swoich gniazd przesyłania dla zarządzanych transferów do tego serwera.

Scenariusz 4

Na poniższym diagramie:

- Atrybut **maxDestinationTransfers** został ustawiony na wartość 2 w pliku `agent.properties`.
- You have set the **maxActiveDestinationTransfers** to 2 on `fileServerA`.
- Atrybut **maxActiveDestinationTransfers** został ustawiony na wartość 2 w systemie `fileServerB`.
- Nie ustawiono atrybutu **maxActiveDestinationTransfers** w systemie `fileServerC`.



Jak pokazano na diagramie, atrybuty **maxActiveDestinationTransfers** i **maxDestinationTransfers** są niezależne od siebie.

Sprawdzone są wartości dla **maxActiveDestinationTransfers** dla każdego z serwerów. W oparciu o tę wartość transfery mogą być dalej kontynuowane lub przekazywane do stanu **WaitingForDestinationFileServerCapacity**.

Transfery, które są dozwolone, przechodzą przez istniejący standardowy przepływ sprawdzający dla **maxDestinationTransfers**.

Scenariusz 5



Ostrzeżenie: Należy zachować ostrożność podczas ustawiania wartości atrybutów **maxActiveDestinationTransfers**, ponieważ należy pamiętać o tym, że należy zachować wartość atrybutu **maxDestinationTransfers**.

Jeśli tego nie zrobisz, może wystąpić sytuacja opisana w następującym tekście:

- Nie ustawiono wartości dla globalnego atrybutu **maxActiveDestinationTransfers**.
- W pliku `agent.properties` ustawiono wartość **maxDestinationTransfers=2**.
- W systemie `fileServerA` ustawiono wartość **maxActiveDestinationTransfers=2**.
- Nie ustawiono wartości dla **maxActiveDestinationTransfers** w systemie `fileServerB`.

Założmy, że występuje następująca sekwencja zdarzeń:

- Agent mostu protokołu odbiera żądanie przestania pliku do programu `fileServerA`. Agent mostu protokołu nie wykonuje obecnie żadnych działań, dlatego akceptuje to żądanie zarządzanego przesyłania.

Szczeliny transferowe wyglądają teraz następująco:

- Transfery docelowe: 1
- Transfery docelowe dla fileServerA: 1
- Transfery docelowe dla fileServerB: 0
- Agent mostu protokołu odbiera kolejne żądanie, aby działać jako agent docelowy dla zarządzanego przesyłania z udziałem fileServerA. Po raz kolejny akceptuje to żądanie, a więc gniazda transferowe wyglądają w ten sposób:
 - Transfery docelowe: 2
 - Transfery docelowe dla fileServerA: 2
 - Transfery docelowe dla fileServerB: 0

Dwa gniazda Destination Transfer w agencji są teraz zajęte, a więc agent nie może uczestniczyć w kolejnych operacjach przesyłania, dopóki nie zostanie zakończony jeden z operacji przesyłania do produktu fileServerA .

- Krótce później fileServerA kończy się niepowodzeniem, co powoduje, że oba zarządzane transfery mają przejść do odtwarzania. Gniazda Destination transfer , które są używane przez te zarządzane transfery, pozostają w użyciu w tym czasie.
- Następnie agent mostu protokołu odbiera żądanie przesłania pliku do produktu fileServerB. Istnieje spacja dla tego przesunięcia w gniazdach Destination Transfers for fileServerB , jednak wszystkie szczeliny Destination Transfer dla agenta są używane, a więc przesyłanie jest umieszczane w dzienniku, aby można było go później ponowierwać.

W rezultacie przesyłanie do produktu fileServerB jest blokowane do momentu zakończenia i zwolnienia gniazda Destination Transfer przez co najmniej jeden z transferów do produktu fileServerA .

Aby zapobiec takiej sytuacji:

- Ustaw wartość parametru **maxActiveDestinationTransfers** na serwerach plików, aby była mniejsza niż wartość **maxDestinationTransfers** , tak aby wolne szczeliny pozostały, lub
- Równomiernie rozdziel wartość atrybutu **maxActiveDestinationTransfers** spośród wszystkich serwerów punktów końcowych.

Zachowanie agenta mostu protokołu w oparciu o wartości atrybutu **maxActiveDestinationTransfers** .

Uwaga: We wszystkich przypadkach błędów wymienionych w poniższej tabeli, jeśli atrybut **maxActiveDestinationTransfers** jest ustawiony na wartość, która nie jest poprawna, agent mostu protokołu zakłada, że ten atrybut nie jest ustawiony.

maxActiveDestinationTransfers	Wartość przykładowa	Opis
Nie wskazano	Nie wskazano	Przelewy idą jak zwykle. Nie ma limitu na liczbę transferów dla punktu końcowego * ftp *.
Określono	0	Dla tego konkretnego punktu końcowego * ftp * nie są dozwolone żadne transfery.
Ujemna wartość	-1	Błąd rejestrowany w output0.log Wartość -1 nie jest poprawny dla nieujemnej liczby całkowitej. Agent mostu protokołu zakłada, że atrybut nie jest ustawiony.
Wartość inna niż liczba całkowita	abc	Błąd rejestrowany w output0.log Wartość abc nie jest poprawny w przypadku liczby całkowitej.

maxActiveDestinationTransfers	Wartość przykładowa	Opis
		Agent mostu protokołu zakłada, że atrybut nie jest ustawiony.
Pusty	""	Wartość ' ' atrybutu maxActiveDestinationTransfers nie jest poprawna dla nieujemnej liczby całkowitej.
Określono	5	Pozwala na uruchomienie tylko pięciu aktywnych transferów w dowolnym momencie dla tego punktu końcowego * ftp *. Nadmierne transfery są ponawiane lub odrzucane w zależności od wartości atrybutu failTransferWhenCapacityReached.

Zachowanie agenta mostu protokołu dla kombinacji atrybutów maxActiveDestinationTransfers i failTransferWhenCapacityReached

failTransferWhenCapacityWartość Relona	Wartość maxActiveDestinationTransfers	Wynik
Fałszywe	3	Dla tego serwera punktu końcowego dozwolone są trzy aktywne transfery. Kolejne transfery są ponawiane.
Prawdziwe	3	Dla tego serwera punktu końcowego dozwolone są trzy aktywne transfery. Wszystkie kolejne transfery są odrzucane i oznaczone jako zakończone niepowodzeniem.
Nie wskazano	3	Wartość domyślna false dla opcji failTransferWhenCapacity jest uważana za wartość. W rezultacie trzy aktywne transfery są dozwolone dla tego serwera punktu końcowego. Kolejne transfery są ponawiane.
Wartości inne niż wartość boolowska	Określono	Błąd rejestrowany w pliku output.log. Wartość podana dla opcji failTransferWhenCapacityReached nie jest wartością boolowskim. Domyślnie rozważana jest wartość domyślna failTransferWhenCapacity.

Zachowanie agenta mostu protokołu dla kombinacji atrybutów **maxDestinationTransfers** i **failTransferWhenCapacityReached**

failTransferWhenCapacityWartość Relona	maxDestinationTransfery wartości	Wynik
Prawdziwe	10	Gdy liczba współbieżnych aktywnych operacji przesyłania osiągnie wartość 10, agent mostu protokołu nie powiedzie się z 11 ^{ty} m przeniesieniem zarządzanym.
Fałszywe	10	Istniejące zachowanie. Gdy liczba współbieżnych aktywnych transferów osiągnie wartość 10, liczba 11 th zarządzanego przesyłania jest kolejkowana w oczekiwaniu na zwolnienie szczeliny.
Nie wskazano	10	Istniejące zachowanie

Komunikaty o błędzie

Istniejący komunikat:

BFGS0082I

Jest rejestrowany w pliku output0.log agenta źródłowego, gdy agent mostu protokołu odrzuca operację przesyłania, gdy agent mostu protokołu jest już uruchomiony maksymalną liczbę transferów zdefiniowanych w atrybucie **maxDestinationTransfers**.

Nowe wiadomości:

BFGSS0085I

Jest protokołowany w pliku output0.log agenta źródłowego, gdy agent mostu protokołu odrzuca i ponawia próbę zarządzanego przesyłania,

BFGSS0086I

Jest protokołowany w pliku output0.log agenta źródłowego, gdy agent mostu protokołu odrzuca i ponawia operację przesyłania zarządzanego, a element docelowy nie zawiera nazwy serwera plików.

BFGSS0084E

Jest rejestrowany w Eksploratorze i pliku audit.xml, gdy agent mostu protokołu odrzuca, dla przekroczenia maksymalnej liczby współbieżnych transferów określonych w atrybucie **maxActiveDestinationTransfers**, i oznacza, że zarządzany transfer nie powiódł się.

BFGSS0087E

Jest rejestrowany w Eksploratorze i pliku audit.xml, gdy agent mostu protokołu odrzuca, dla przekroczenia maksymalnej liczby transferów docelowych określonych w atrybucie **maxActiveDestinationTransfers**, i oznacza, że zarządzany transfer nie powiódł się.

BFGSS0088W

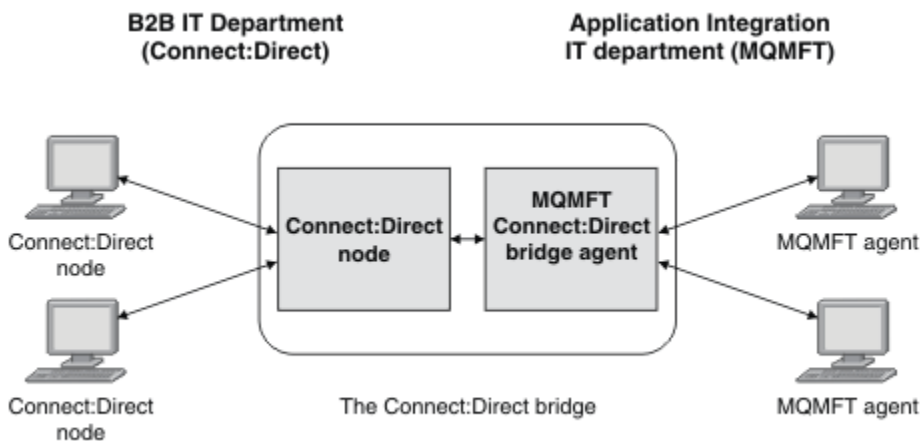
Jest rejestrowany w pliku output0.log, gdy wartość atrybutu **maxActiveDestinationTransfers** przekracza wartość atrybutu **maxDestinationTransfers**.

BFGSS0089I

Jest rejestrowany w pliku output0.log agenta mostu protokołu docelowego, gdy działa agent źródłowy, który nie znajduje się w produkcie IBM MQ 9.2.1 lub w późniejszym czasie.

Most Connect:Direct

Istnieje możliwość przesyłania plików do i z istniejącej sieci produktu IBM Sterling Connect:Direct . Użyj mostu Connect:Direct , który jest komponentem produktu Managed File Transfer, do przesyłania plików między MFT i IBM Sterling Connect:Direct.



Na diagramie przedstawiono most MFT Connect:Direct między dwoma działami, dział IT B2B oraz dział IT integracji aplikacji. Dział IT B2B używa produktu Connect:Direct do przesyłania plików do i z partnerów biznesowych firmy. Dział IT Application Integration używa produktu IBM MQ jako infrastruktury przesyłania komunikatów, dlatego w ostatnim czasie wybrano produkt Managed File Transfer jako rozwiązanie przesyłania plików.

Korzystając z mostu MFT Connect:Direct , oba działy mogą przysyłać pliki między siecią Connect:Direct w dziale IT B2B oraz siecią MFT w dziale IT Application Integration. Most Connect:Direct jest komponentem produktu Managed File Transfer, który zawiera agenta MFT , który komunikuje się z węzłem produktu Connect:Direct . Agent MFT jest dedykowany do przesyłania z węzłem Connect:Direct i jest znany jako agent mostu Connect:Direct .

Most Connect:Direct jest dostępny jako część komponentów Service and Agent produktu Managed File Transferi może być używany do wykonywania następujących zadań:

1. Komenda Managed File Transfer służy do inicjowania przesyłania pliku lub wielu plików z agenta MFT do węzła Connect:Direct .
2. Za pomocą komend Managed File Transfer można zainicjować przesyłanie pliku lub wielu plików z węzła Connect:Direct do agenta MFT .
3. Komendy Managed File Transfer umożliwiają zainicjowanie operacji przesyłania plików, która uruchamia proces Connect:Direct zdefiniowany przez użytkownika.
4. Użyj procesu Connect:Direct , aby wysłać żądanie przesłania pliku MFT .

Most Connect:Direct może przysyłać pliki do lub z tylko węzłów Connect:Direct . Most Connect:Direct może przysyłać pliki do lokalnego systemu plików lub z niego tylko w ramach przesyłania przekazanego przez proces Connect:Direct .


► **z/OS** Mostu Connect:Direct można użyć do przesyłania danych do lub z zestawu danych znajdującego się w węzle produktu Connect:Direct w systemie z/OS . Istnieją pewne różnice w działaniu w porównaniu z transferami zestawu danych, które dotyczą tylko agentów Managed File Transfer .

Więcej informacji na ten temat zawiera sekcja ► **z/OS** [Przesyłanie zestawów danych do i z węzłów Connect:Direct](#).

Obsługiwane platformy

Most Connect:Direct składa się z agenta mostu MFT Connect:Direct i węzła Connect:Direct . The agent is supported on Windows and Linux for x86-64. The node is supported on the platforms that are supported

for IBM Sterling Connect:Direct for Windows and IBM Sterling Connect:Direct for UNIX. Instrukcje dotyczące tworzenia agenta mostu Connect:Direct i konfigurowania węzła Connect:Direct dla agenta, z którym można się komunikować, zawiera sekcja [Konfigurowanie mostu Connect:Direct](#).

The Connect:Direct bridge can transfer files to and from Connect:Direct nodes that are running as part of a Connect:Direct for Windows or Connect:Direct for UNIX , lub Connect:Direct dla z/OS Service installation. Szczegółowe informacje na temat obsługiwanych wersji produktu Connect:Direct można znaleźć na stronie WWW [Wymagania systemowe dla produktu IBM MQ](#).

Agent i węzeł tworzące most Connect:Direct muszą znajdować się w tym samym systemie lub mieć dostęp do tego samego systemu plików, na przykład za pośrednictwem współużytkowanego podłączonego systemu plików NFS. Ten system plików służy do tymczasowego przechowywania plików podczas przesyłania plików, które dotyczą mostu Connect:Direct, w katalogu definiowanym przez parametr **cdTmpDir**. Agent mostu Connect:Direct i węzeł mostu Connect:Direct muszą mieć możliwość dostępu do tego katalogu przy użyciu ścieżki o tej samej nazwie. Jeśli na przykład agent i węzeł znajdują się w oddzielnych systemach Windows, współużytkowany system plików musi być podłączony w tych systemach za pomocą tej samej litery napędu. Agent i węzeł mogą używać tej samej nazwy ścieżki, jeśli używane są następujące konfiguracje:

- Agent i węzeł znajdują się w tym samym systemie Windows lub Linux dla platformy x86-64.
- Agent znajduje się w systemie Linux dla platformy x86-64, a węzeł znajduje się w systemie AIX.
- Agent znajduje się w jednym systemie Windows, a węzeł znajduje się w innym systemie Windows.

Agent i węzeł nie mogą używać tej samej nazwy ścieżki, jeśli używane są następujące konfiguracje:

- Agent znajduje się w systemie Linux dla platformy x86-64, a węzeł znajduje się w systemie Windows.
- Agent znajduje się w systemie Windows, a węzeł znajduje się w systemie UNIX.

Te ograniczenia należy wziąć pod uwagę podczas planowania instalacji mostu Connect:Direct.

Przesyłanie pliku do węzła Connect:Direct

Za pomocą mostu Connect:Direct można przestać plik z agenta Managed File Transfer do węzła Connect:Direct. Określ węzeł Connect:Direct jako miejsce docelowe przesyłania, określając agenta mostu Connect:Direct jako agenta docelowego i określając plik docelowy w formularzu `connect_direct_node_name:file_path`.

Zanim rozpoczniesz

Przed przesyłaniem pliku należy skonfigurować most Connect:Direct, który jest komponentem produktu Managed File Transfer. Więcej informacji na ten temat zawiera sekcja [Konfigurowanie mostu Connect:Direct](#).

O tym zadaniu

W tym przykładzie agent mostu Connect:Direct nosi nazwę CD_BRIDGE. Agent źródłowy nosi nazwę FTE_AGENT i może być dowolną wersją produktu WMQFTE. Docelowy węzeł Connect:Direct ma nazwę CD_NODE1. Plik, który ma zostać przesyłany, znajduje się w ścieżce pliku /home/helen/file.log w systemie, w którym znajduje się FTE_AGENT. Plik jest przesyłany do ścieżki do pliku /files/data.log w systemie, w którym działa CD_NODE1.

Procedura

1. Użyj komendy `fteCreateTransfer` z wartością parametru **-df** (plik docelowy) w formularzu `connect_direct_node_name:file_path` i wartością parametru **-da** (agent docelowy) określoną jako nazwa agenta mostu Connect:Direct.

Uwaga: Węzeł Connect:Direct określony przez produkt `connect_direct_node_name` jest węzłem, do którego ma zostać przesłany plik, a nie do węzła Connect:Direct działającego jako część mostu Connect:Direct .

```
fteCreateTransfer -sa FTE_AGENT -da CD_BRIDGE
-df CD_NODE1:/files/data.log /home/helen/file.log
```

Więcej informacji na ten temat zawiera sekcja **fteCreateTransfer**: uruchomienie nowego przesyłania plików.

2. Agent źródłowy FTE_AGENT przekazuje ten plik do agenta mostu Connect:Direct CD_BRIDGE. Plik jest tymczasowo przechowywany w systemie, w którym działa agent mostu Connect:Direct , w położeniu zdefiniowanym przez właściwość agenta `cdTmpDir`. Agent mostu Connect:Direct przesyła plik do węzła Connect:Direct CD_NODE1.

Pojęcia pokrewne

[“Most Connect:Direct” na stronie 292](#)

Istnieje możliwość przesyłania plików do i z istniejącej sieci produktu IBM Sterling Connect:Direct . Użyj mostu Connect:Direct , który jest komponentem produktu Managed File Transfer, do przesyłania plików między MFT i IBM Sterling Connect:Direct.

Zadania pokrewne

[“Przesyłanie pliku z węzła Connect:Direct” na stronie 294](#)

Za pomocą mostu Connect:Direct można przesłać plik z węzła Connect:Direct do partycji Managed File Transfer Agent . Jako źródło operacji przesyłania można określić węzeł Connect:Direct , określając agenta mostu Connect:Direct jako agenta źródłowego i określając specyfikację źródła w formularzu `connect_direct_node_name:file_path`.

Odsyłacze pokrewne

[Plik MFT agent.properties](#)

Przesyłanie pliku z węzła Connect:Direct

Za pomocą mostu Connect:Direct można przesłać plik z węzła Connect:Direct do partycji Managed File Transfer Agent . Jako źródło operacji przesyłania można określić węzeł Connect:Direct , określając agenta mostu Connect:Direct jako agenta źródłowego i określając specyfikację źródła w formularzu `connect_direct_node_name:file_path`.

Zanim rozpoczniesz

Przed przesyłaniem pliku należy skonfigurować most Connect:Direct , który jest komponentem produktu Managed File Transfer. Patrz sekcja [Konfigurowanie mostu Connect:Direct](#).

O tym zadaniu

W tym przykładzie agent mostu Connect:Direct nosi nazwę CD_BRIDGE. Agent docelowy ma nazwę FTE_AGENT i może być dowolną wersją produktu Managed File Transfer. Źródłowy węzeł Connect:Direct ma nazwę CD_NODE1. Plik, który ma zostać przesłany, znajduje się w ścieżce pliku `/home/brian/in.file` w systemie, w którym znajduje się zmienna CD_NODE1 . Plik jest przesyłany do ścieżki do pliku `/files/out.file` w systemie, w którym działa FTE_AGENT.

Procedura

Użyj komendy **fteCreateTransfer** z wartością dla specyfikacji źródła w formularzu `connect_direct_node_name:file_path` i wartością parametru **-sa** określoną jako nazwa agenta mostu Connect:Direct .

Uwaga: Węzeł Connect:Direct określony przez `connect_direct_node_name` jest węzłem, z którego ma zostać przesyłany plik, a nie z węzłem Connect:Direct działającym jako część mostu Connect:Direct. Na przykład:

```
fteCreateTransfer -sa CD_BRIDGE -da FTE_AGENT  
-df /files/out.file CD_NODE1:/home/brian/in.file
```

Więcej informacji na ten temat zawiera sekcja **fteCreateTransfer**: uruchomienie nowego przesyłania plików.

Wyniki

Agent mostu Connect:Direct CD_BRIDGE żąda pliku z węzła Connect:Direct CD_NODE1. Węzeł Connect:Direct wysyła plik do mostu Connect:Direct. Gdy plik jest przesyłany z węzła Connect:Direct, most Connect:Direct przechowuje plik tymczasowo w położeniu zdefiniowanym przez właściwość agenta `cdTmpDir`. Po zakończeniu przesyłania pliku z węzła Connect:Direct do mostu Connect:Direct most Connect:Direct wysyła ten plik do agenta docelowego FTE_AGENT i usuwa plik z położenia tymczasowego.

Pojęcia pokrewne

“Most Connect:Direct” na stronie 292

Istnieje możliwość przesyłania plików do i z istniejącej sieci produktu IBM Sterling Connect:Direct. Użyj mostu Connect:Direct, który jest komponentem produktu Managed File Transfer, do przesyłania plików między MFT i IBM Sterling Connect:Direct.

Odsyłacze pokrewne

Plik MFT `agent.properties`

Przesyłanie zestawu danych do węzła Connect:Direct w systemie z/OS

Istnieje możliwość przesłania zestawu danych z agenta Managed File Transfer w systemie z/OS do węzła Connect:Direct w systemie z/OS przy użyciu mostu Connect:Direct znajdującego się w systemie Windows lub Linux.

Zanim rozpoczniesz

Przed przesłaniem pliku należy skonfigurować most Connect:Direct, który jest komponentem produktu Managed File Transfer. Patrz sekcja Konfigurowanie mostu Connect:Direct.

O tym zadaniu

W tym przykładzie parametr `-df` jest używany do określenia miejsca docelowego przesyłania. Parametr `-df` jest poprawny w przypadku użycia, gdy agent źródłowy przesyłania jest dowolną wersją produktu Managed File Transfer. Jeśli agent źródłowy jest w wersji IBM WebSphere MQ File Transfer Edition 7.0.4 lub nowszej, można zamiast niego użyć parametru `-ds`. Agent źródłowy nosi nazwę FTE_ZOS1 i jest agentem IBM WebSphere MQ File Transfer Edition 7.0.3. Agent mostu Connect:Direct nosi nazwę CD_BRIDGE i znajduje się w systemie Linux. Docelowy węzeł Connect:Direct ma nazwę CD_ZOS2. Zarówno agent źródłowy, jak i docelowy węzeł Connect:Direct znajdują się w systemach z/OS. Zestaw danych, który ma zostać przesłany, znajduje się w systemie `//FTEUSER.SOURCE.LIB` w systemie, w którym znajduje się FTE_ZOS1. Zestaw danych jest przesyłany do zestawu danych `//CDUSER.DEST.LIB` w systemie, w którym znajduje się CD_ZOS2.

Uwaga: IBM WebSphere MQ File Transfer Edition (FTE) nie jest już obsługiwany produktem. Informacje o migrowaniu FTE do składnika Managed File Transfer w programie IBM MQ można znaleźć w sekcji Migrowanie produktu Managed File Transfer.

Procedura

1. Użyj komendy `fteCreateTransfer` z wartością parametru **-df** w formularzu: `connect_direct_node_name:data_set_name;attributes` i wartością parametru **-da** (agent docelowy) określoną jako nazwa agenta mostu `Connect:Direct`.

Węzeł `Connect:Direct` określony przez produkt `connect_direct_node_name` jest węzłem, do którego ma zostać przesłany zestaw danych, a nie węzeł `Connect:Direct`, który działa jako część mostu `Connect:Direct`.

Nazwa zestawu danych określona przez parametr `nazwa_zestawu_danych` musi być bezwzględna, a nie względna. `Connect:Direct` nie jest przedrostkiem nazwy zestawu danych z nazwą użytkownika.

```
fteCreateTransfer -sa FTE_ZOS1 -sm QM_ZOS
                 -da CD_BRIDGE -dm QM_BRIDGE
                 -df CD_ZOS2:/'CDUSER.DEST.LIB;BLKSIZE(8000);LRECL(80)'
                 //'FTEUSER.SOURCE.LIB'
```

Więcej informacji na ten temat zawiera sekcja **fteCreateTransfer: uruchomienie nowego przesyłania plików**.


2. Agent źródłowy `FTE_ZOS1` przesyła dane w zestawie danych do agenta mostu `Connect:Direct` `CD_BRIDGE`. Dane są tymczasowo przechowywane jako plik tekstowy w systemie, w którym działa agent mostu `Connect:Direct`, w położeniu zdefiniowanym przez właściwość agenta `cdTmpDir`. Agent mostu `Connect:Direct` przesyła dane do węzła `Connect:Direct` `CD_ZOS2`. Po zakończeniu przesyłania plik płaski jest usuwany z systemu, w którym działa agent mostu `Connect:Direct`.

Pojęcia pokrewne

[“Most Connect:Direct” na stronie 292](#)

Istnieje możliwość przesyłania plików do i z istniejącej sieci produktu IBM Sterling Connect:Direct. Użyj mostu `Connect:Direct`, który jest komponentem produktu Managed File Transfer, do przesyłania plików między MFT i IBM Sterling Connect:Direct.

Zadania pokrewne

 [Przesyłanie zestawów danych do i z węzłów Connect:Direct](#)

Odsyłacze pokrewne

 [Właściwości BPXWDYN nie mogą być używane z produktem MFT](#)

Przesyłanie wielu plików do węzła Connect:Direct

Za pomocą mostu `Connect:Direct` można przestać wiele plików z Managed File Transfer Agent do węzła `Connect:Direct`. Aby użyć węzła `Connect:Direct` jako miejsca docelowego przesyłania wielu plików, należy określić agenta mostu `Connect:Direct` jako agenta docelowego i określić katalog docelowy w formularzu `connect_direct_node_name:directory_path`.

Zanim rozpocznie

Przed przesłaniem plików należy skonfigurować most `Connect:Direct`, który jest komponentem produktu Managed File Transfer. Patrz sekcja [Konfigurowanie mostu Connect:Direct](#).

O tym zadaniu

W tym przykładzie agent źródłowy nosi nazwę `FTE_AGENT`. Agent mostu `Connect:Direct` nosi nazwę `CD_BRIDGE`. Docelowy węzeł `Connect:Direct` ma nazwę `CD_NODE1`. Pliki, które mają zostać przesłane, to: `/home/jack/data.log`, `/logs/log1.txt`, `/results/latest` w systemie, w którym znajduje się `FTE_AGENT`. Pliki są przesyłane do katalogu `/in/files` w systemie, w którym działa `CD_NODE1`.

Procedura

Użyj komendy `fteCreateTransfer` z wartością parametru **-dd** (katalog docelowy) w formularzu `connect_direct_node_name:directory_path`. Podaj wartość parametru **-da** (agent docelowy) jako nazwę agenta mostu `Connect:Direct`.

Uwaga: Węzeł Connect:Direct określony przez produkt `connect_direct_node_name` jest węzłem, do którego mają zostać przesłane pliki, a nie do węzła Connect:Direct, który działa jako część mostu Connect:Direct.

```
fteCreateTransfer -sa FTE_AGENT -da CD_BRIDGE
                  -dd CD_NODE1:/in/files /home/jack/data.log
                  /logs/log1.txt /results/latest
```

Więcej informacji na ten temat zawiera sekcja **fteCreateTransfer**: uruchomienie nowego przesyłania plików.

Wyniki

Agent źródłowy FTE_AGENT przekazuje pierwszy plik do agenta mostu Connect:Direct CD_BRIDGE. Agent mostu Connect:Direct tymczasowo przechowuje plik w położeniu zdefiniowanym przez właściwość `cdTmpDir`. Gdy plik został całkowicie przesłany z agenta źródłowego do mostu Connect:Direct, agent mostu Connect:Direct wysyła ten plik do węzła Connect:Direct, który jest zdefiniowany przez właściwość agenta `cdNode`. Ten węzeł wysyła plik do docelowego węzła Connect:Direct CD_NODE1. Agent mostu Connect:Direct usuwa plik z położenia tymczasowego, gdy operacja przesyłania między dwoma węzłami programu Connect:Direct została zakończona. Ten proces jest powtarzany dla każdego określonego pliku źródłowego.

Pojęcia pokrewne

[“Most Connect:Direct” na stronie 292](#)

Istnieje możliwość przesyłania plików do i z istniejącej sieci produktu IBM Sterling Connect:Direct. Użyj mostu Connect:Direct, który jest komponentem produktu Managed File Transfer, do przesyłania plików między MFT i IBM Sterling Connect:Direct.

Zadania pokrewne

[“Przesyłanie pliku do węzła Connect:Direct” na stronie 293](#)

Za pomocą mostu Connect:Direct można przestać plik z agenta Managed File Transfer do węzła Connect:Direct. Określ węzeł Connect:Direct jako miejsce docelowe przesyłania, określając agenta mostu Connect:Direct jako agenta docelowego i określając plik docelowy w formularzu `connect_direct_node_name:file_path`.

[“Przesyłanie wielu plików do programu Connect:Direct przy użyciu znaków wieloznacznych” na stronie 298](#)

Aby przestać wiele plików z agenta Managed File Transfer do węzła Connect:Direct, należy użyć mostu Connect:Direct. W specyfikacji źródłowej można używać znaków wieloznacznych, które są dostępne w komendzie **fteCreateTransfer**. Podobnie jak w przypadku wszystkich transferów Managed File Transfer obejmujących znaki wieloznaczne, tylko ostatnia część ścieżki do pliku może zawierać znak wieloznaczny. Na przykład `/abc/def*` jest poprawną ścieżką do pliku, a `/abc*/def` nie jest poprawna.

[“Przesyłanie pliku z węzła Connect:Direct” na stronie 294](#)

Za pomocą mostu Connect:Direct można przestać plik z węzła Connect:Direct do partycji Managed File Transfer Agent. Jako źródło operacji przesyłania można określić węzeł Connect:Direct, określając agenta mostu Connect:Direct jako agenta źródłowego i określając specyfikację źródła w formularzu `connect_direct_node_name:file_path`.

[“Przesyłanie wielu plików z węzła Connect:Direct” na stronie 297](#)

Za pomocą mostu Connect:Direct można przestać wiele plików z węzła Connect:Direct do Managed File Transfer Agent. Węzeł Connect:Direct można określić jako źródło wielu operacji przesyłania plików, określając agenta mostu Connect:Direct jako agenta źródłowego i określając jedną lub większą liczbę specyfikacji źródłowych w formularzu `connect_direct_node_name:file_path`.

Odsyłacze pokrewne

[Plik MFT agent.properties](#)

Przesyłanie wielu plików z węzła Connect:Direct

Za pomocą mostu Connect:Direct można przestać wiele plików z węzła Connect:Direct do Managed File Transfer Agent. Węzeł Connect:Direct można określić jako źródło wielu operacji przesyłania plików,

określając agenta mostu Connect:Direct jako agenta źródłowego i określając jedną lub większą liczbę specyfikacji źródłowych w formularzu `connect_direct_node_name:file_path`.

Zanim rozpoczniesz

Przed przesłaniem pliku należy skonfigurować most Connect:Direct, który jest komponentem produktu Managed File Transfer. Więcej informacji na ten temat zawiera sekcja [Konfigurowanie mostu Connect:Direct](#).

O tym zadaniu

W tym przykładzie agent mostu Connect:Direct nosi nazwę CD_BRIDGE. Agent docelowy jest nazywany FTE_Z i jest uruchomiony w systemie z/OS. Źródłowy węzeł Connect:Direct ma nazwę CD_NODE1. Pliki, które mają zostać przesłane, znajdują się w ścieżkach plików `/in/file1`, `/in/file2` i `/in/file3` w systemie, w którym znajduje się zmienna CD_NODE1. Pliki są przesyłane do partycjonowanego zestawu danych `//OBJECT.LIB` w systemie, w którym działa FTE_Z.

Procedura

Użyj komendy `fteCreateTransfer` z wartościami dla specyfikacji źródłowych w formularzu `connect_direct_node_name:file_path` i wartością parametru `-sa` określoną jako nazwa agenta mostu Connect:Direct.

Uwaga: Węzeł Connect:Direct określony przez produkt `connect_direct_node_name` jest węzłem, z którego pliki mają zostać przesłane, a nie z węzłem Connect:Direct działającym jako część mostu Connect:Direct.

```
fteCreateTransfer -sa CD_BRIDGE -da FTE_Z
                  -dp '//OBJECT.LIB' CD_NODE1:/in/file1
                  CD_NODE1:/in/file2 CD_NODE1:/in/file3
```

Więcej informacji na ten temat zawiera sekcja [fteCreateTransfer: uruchomienie nowego przesyłania plików](#).

Wyniki

Agent mostu Connect:Direct CD_BRIDGE żąda pierwszego pliku z węzła Connect:Direct CD_NODE1. Węzeł Connect:Direct wysyła plik do mostu Connect:Direct. Gdy plik jest przesyłany z węzła Connect:Direct, most Connect:Direct przechowuje plik tymczasowo w położeniu zdefiniowanym przez właściwość agenta `cdTmpDir`. Po zakończeniu przesyłania pliku z węzła Connect:Direct do mostu Connect:Direct most Connect:Direct wysyła ten plik do agenta docelowego FTE_Z, a następnie usuwa ten plik z położenia tymczasowego. Ten proces jest powtarzany dla każdego określonego pliku źródłowego.

Pojęcia pokrewne

“Most Connect:Direct” na stronie 292

Istnieje możliwość przesyłania plików do i z istniejącej sieci produktu IBM Sterling Connect:Direct. Użyj mostu Connect:Direct, który jest komponentem produktu Managed File Transfer, do przesyłania plików między MFT i IBM Sterling Connect:Direct.

Odsyłacze pokrewne

[Plik MFT agent.properties](#)

Przesyłanie wielu plików do programu Connect:Direct przy użyciu znaków wieloznacznych

Aby przesłać wiele plików z agenta Managed File Transfer do węzła Connect:Direct, należy użyć mostu Connect:Direct. W specyfikacji źródłowej można używać znaków wieloznacznych, które są dostępne w komendzie `fteCreateTransfer`. Podobnie jak w przypadku wszystkich transferów Managed File Transfer obejmujących znaki wieloznaczne, tylko ostatnia część ścieżki do pliku może zawierać znak wieloznaczny. Na przykład `/abc/def*` jest poprawną ścieżką do pliku, a `/abc*/def` nie jest poprawna.

Zanim rozpoczniesz

Przed przestaniem pliku należy skonfigurować most Connect:Direct, który jest komponentem produktu Managed File Transfer. Więcej informacji na ten temat zawiera sekcja [Konfigurowanie mostu Connect:Direct](#).

O tym zadaniu

W tym przykładzie agent źródłowy nosi nazwę FTE_AGENT, a agent mostu Connect:Direct nosi nazwę CD_BRIDGE. Docelowy węzeł Connect:Direct ma nazwę CD_NODE1. Pliki, które mają zostać przesłane, znajdują się w katalogu /reports w systemie, w którym znajduje się FTE_AGENT. Przesyłane są tylko pliki o nazwach zaczynający się od report, po których następują dwa znaki i przyrostek .log. Na przykład plik /reports/report01.log jest przesyłany, ale plik /reports/report1.log nie jest przesyłany. Pliki są przesyłane do katalogu /home/fred w systemie, w którym działa CD_NODE1.

Procedura

1. Użyj komendy fteCreateTransfer z wartością parametru **-dd** (katalog docelowy) w formularzu `connect_direct_node_name:directory_path`. W przypadku parametru **-da** (agent docelowy) podaj agenta mostu Connect:Direct.

Uwaga: Węzeł Connect:Direct określony przez produkt `connect_direct_node_name` jest węzłem, do którego mają zostać przesłane pliki, a nie do węzła Connect:Direct, który działa jako część mostu Connect:Direct.

```
fteCreateTransfer -sa FTE_AGENT -da CD_BRIDGE
                  -dd CD_NODE1:/home/fred "/reports/report??.log"
```

Więcej informacji na ten temat zawiera sekcja [fteCreateTransfer: uruchomienie nowego przesyłania plików](#).

2. Agent źródłowy FTE_AGENT przekazuje pierwszy plik, który jest zgodny ze wzorcem /reports/report??.log, do agenta mostu Connect:Direct CD_BRIDGE. Agent mostu Connect:Direct tymczasowo przechowuje plik w położeniu zdefiniowanym przez właściwość cdTmpDir. Gdy plik został całkowicie przesłany z agenta źródłowego do mostu Connect:Direct, agent mostu Connect:Direct wysyła ten plik do węzła Connect:Direct, który jest zdefiniowany przez właściwość agenta cdNode. Ten węzeł wysyła plik do docelowego węzła Connect:Direct CD_NODE1. Agent mostu Connect:Direct usuwa plik z położenia tymczasowego, gdy operacja przesyłania między dwoma węzłami programu Connect:Direct została zakończona. Proces ten jest powtarzany dla każdego pliku źródłowego, który jest zgodny ze wzorcem wieloznacznym /reports/report??.log.

Uwaga: Lista plików zgodnych ze wzorcem /reports/report??.log różni się w zależności od systemu operacyjnego systemu, w którym znajduje się agent źródłowy FTE_AGENT.

- Jeśli agent źródłowy znajduje się w systemie z systemem operacyjnym Windows, w dopasowaniu wzorca nie jest rozróżniana wielkość liter. Wzorec dopasowuje wszystkie pliki w katalogu /reports z nazwą pliku o nazwie report, po której następują dwa znaki i przyrostek .log, niezależnie od przypadku, w którym znajdują się litery. Na przykład Report99.Log jest zgodne.
- Jeśli agent źródłowy znajduje się w systemie z systemem operacyjnym Linux lub UNIX, w dopasowaniu wzorca rozróżniana jest wielkość liter. Wzorec jest zgodny tylko z tymi plikami w katalogu /reports z nazwą pliku o nazwie report, po której następują dwa znaki i przyrostek .log. Na przykład reportAB.log jest zgodne, ale reportAB.LOG i Report99.Log nie są zgodne.

Odtwarzanie i restartowanie w przypadku przesyłania do i z węzłów Connect:Direct

Program Managed File Transfer może nie być w stanie połączyć się z węzłem IBM Sterling Connect:Direct podczas przesyłania, na przykład, jeśli węzeł stanie się niedostępny. Managed File Transfer podejmuje próbę odtworzenia operacji przesyłania lub operacja przesyłania nie powiedzie się i zostanie wygenerowany komunikat o błędzie.

Jeśli węzeł Connect:Direct stanie się niedostępny

Jeśli węzeł Connect:Direct stanie się niedostępny, na przykład z powodu sieci lub wyłączenia zasilania, program Managed File Transfer odtwarza plik w jeden z następujących sposobów:

- Jeśli produkt Managed File Transfer nie został wcześniej pomyślnie połączony z węzłem Connect:Direct w ramach tego żądania przesyłania, operacja przesyłania jest podejmowana ponownie przez pewien czas określony przez wartości parametrów **cdMaxConnectionRetries** i **recoverableTransferRetryInterval properties**. Te właściwości są określone w pliku `agent.properties` dla agenta mostu Connect:Direct. Operacja przesyłania nie powiedzie się i zostanie wygenerowany komunikat o błędzie, po osiągnięciu przez liczbę nieudanych prób osiągnięcia wartości **cdMaxConnectionRetries property**. Domyślnie operacja przesyłania jest podejmowana w nieskończoność, z 60 sekund między próbami.
- Jeśli produkt Managed File Transfer został wcześniej pomyślnie połączony z węzłem Connect:Direct w ramach tego żądania przesyłania, operacja przesyłania jest podejmowana ponownie przez pewien czas określony przez wartości właściwości **cdMaxPartialWorkConnectionRetries** i **recoverableTransferRetryInterval**. Przesyłanie nie powiodło się i zostanie wygenerowany komunikat o błędzie, po osiągnięciu przez liczbę nieudanych prób osiągnięcia wartości właściwości **cdMaxPartialWorkConnectionRetries**. Domyślnie operacja przesyłania jest podejmowana w nieskończoność, z 60 sekund między próbami.
- W przypadku niektórych typów awarii węzła Connect:Direct, na przykład w przypadku wymuszonego zatrzymania węzła, procesy produktu Connect:Direct mają status `HEld Due to Error (HE)`, gdy węzeł jest odtwarzalny. Po odzyskaniu węzła produkt Managed File Transfer automatycznie wznawia wszystkie procesy produktu Connect:Direct, które są powiązane z przesyłaniem plików i mają status HE.
- Jeśli operacja przesyłania nie powiedzie się, wszystkie pliki tymczasowe związane z przesyłaniem zostaną usunięte z systemu, który udostępni most Connect:Direct. Położenie tych plików tymczasowych jest definiowane przez właściwość **cdTmpDir**.
- Jeśli przesyłanie pochodzi z Managed File Transfer do Connect:Direct, a określono dyspozycję źródłową usuwania, wówczas pliki źródłowe nie zostaną usunięte, jeśli operacja przesyłania nie powiedzie się.

Jeśli referencje użytkownika węzła Connect:Direct są niepoprawne

Jeśli program Managed File Transfer nie połączy się z węzłem Connect:Direct, ponieważ referencje użytkownika zostały odrzucone przez węzeł, przesyłanie nie powiedzie się i zostanie wygenerowany komunikat o błędzie. W takiej sytuacji należy sprawdzić, czy podano poprawne referencje użytkownika dla węzła Connect:Direct. Więcej informacji na ten temat zawiera sekcja [Odwzorowanie referencji dla produktu Connect:Direct](#).

Jeśli agent mostu Connect:Direct stanie się niedostępny

Jeśli agent mostu Connect:Direct stanie się niedostępny, wszystkie bieżące operacje przesyłania plików zostaną odzyskane w taki sam sposób, jak standardowe transfery produktu Managed File Transfer. Więcej informacji na ten temat zawiera sekcja ["Odtwarzanie i restartowanie produktu MFT" na stronie 307](#).

Pojęcia pokrewne

["Most Connect:Direct" na stronie 292](#)

Istnieje możliwość przesyłania plików do i z istniejącej sieci produktu IBM Sterling Connect:Direct. Użyj mostu Connect:Direct, który jest komponentem produktu Managed File Transfer, do przesyłania plików między MFT i IBM Sterling Connect:Direct.

["Odtwarzanie i restartowanie produktu MFT" na stronie 307](#)

If your agent or queue manager are unavailable for any reason, for example because of a power or network failure, Managed File Transfer recovers as follows in these scenarios:

Zadania pokrewne

[Konfigurowanie mostu Connect:Direct](#)

Odsyłacze pokrewne

[Plik MFT `agent.properties`](#)

Wysyłanie zdefiniowanego przez użytkownika procesu Connect:Direct z żądania przesyłania plików

Istnieje możliwość wprowadzenia żądania przesyłania dla operacji przesyłania, która przechodzi przez agent mostu Connect:Direct, który wywołuje zdefiniowany przez użytkownika proces Connect:Direct w ramach przesyłania plików.

Domyślnie po wystaniu żądania przesyłania plików dla operacji przesyłania, która przechodzi przez most Connect:Direct, agent mostu Connect:Direct generuje proces Connect:Direct używany do przesyłania pliku do lub ze zdalnego węzła Connect:Direct.

Można jednak skonfigurować agenta mostu Connect:Direct w taki sposób, aby wywołał proces Connect:Direct zdefiniowany przez użytkownika przy użyciu pliku `ConnectDirectProcessDefinition.xml`.

Plik `ConnectDirectProcessDefinition.xml`

Komenda `fteCreateCDAgent` tworzy plik `ConnectDirectProcessDefinitions.xml` w katalogu konfiguracji agenta `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent_name`. Zanim możliwe będzie wywołanie procesów Connect:Direct zdefiniowanych przez użytkownika z agenta mostu Connect:Direct, należy skonfigurować definicje procesów, edytując ten plik.

Plik definiuje jeden lub więcej zestawów procesów, które zawierają położenie jednego lub większej liczby procesów produktu Connect:Direct wywołanych jako część przesyłania. Każdy zestaw procesów zawiera pewną liczbę warunków. Jeśli transfer spełnia wszystkie warunki zestawu procesów, zestaw procesów jest używany do określania, które procesy produktu Connect:Direct są wywoływane przez operację przesyłania. Więcej informacji na ten temat zawiera sekcja [“Określanie procesu Connect:Direct, który ma zostać uruchomiony za pomocą pliku ConnectDirectProcessDefinition.xml”](#) na stronie 301.

Wbudowane zmienne symboliczne

Istnieje możliwość użycia wbudowanych zmiennych symbolicznych, które są definiowane przez program Managed File Transfer w celu podstawienia wartości do zdefiniowanych przez użytkownika procesów Connect:Direct. Aby postępować zgodnie z konwencją nazewnictwa Connect:Direct, wszystkie wewnętrzne zmienne symboliczne używane przez program Managed File Transfer mają format `%FTE`, po którym następują pięć wielkich liter.

Gdy tworzysz proces w celu przesyłania plików z węzła Connect:Direct do systemu mostu Connect:Direct, należy użyć wbudowanej zmiennej `%FTETFILE` jako wartości parametru `TO FILE` w procesie Connect:Direct. Gdy tworzysz proces w celu przesyłania plików do węzła Connect:Direct z systemu mostu Connect:Direct, należy użyć wbudowanej zmiennej `%FTEFFILE` jako wartości parametru `FROM FILE` w procesie Connect:Direct. Te zmienne zawierają ścieżki do plików tymczasowych, które są używane przez agent mostu Connect:Direct do przesyłania danych do i z sieci produktu Managed File Transfer.

Więcej informacji na temat wewnętrznych zmiennych symbolicznych można znaleźć w dokumentacji produktu Connect:Direct.

Przykładowe procesy produktu Connect:Direct

Produkt Managed File Transfer udostępnia przykładowe procesy produktu Connect:Direct. Przykłady te znajdują się w następującym katalogu: `MQ_INSTALLATION_PATH/mqft/samples/ConnectDirectProcessTemplates`.

Określanie procesu Connect:Direct, który ma zostać uruchomiony za pomocą pliku `ConnectDirectProcessDefinition.xml`

Określ, który proces Connect:Direct ma być uruchamiany jako część przesyłania Managed File Transfer. Produkt Managed File Transfer udostępnia plik XML, który można edytować w celu określenia definicji procesów.

O tym zadaniu

Komenda **fteCreateCDAgent** tworzy plik `ConnectDirectProcessDefinitions.xml` w katalogu konfiguracji agenta `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent_name`. Zanim możliwe będzie wywołanie procesów `Connect:Direct` zdefiniowanych przez użytkownika z agenta mostu `Connect:Direct`, należy skonfigurować definicje procesów, edytując ten plik.

Dla każdego procesu, który ma zostać określony jako część operacji przesyłania za pomocą mostu `Connect:Direct`, wykonaj następujące kroki:

Procedura

1. Zdefiniuj proces `Connect:Direct`, który agent mostu `Connect:Direct` ma wywoływać jako część przesyłania, a następnie zapisz szablon procesu w pliku.
2. Otwórz plik `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent_name/ConnectDirectProcessDefinitions.xml` w edytorze tekstu.
3. Utwórz element `<processSet>`.
4. Wewnątrz elementu `<processSet>` utwórz element `<condition>`.
5. Wewnątrz elementu `<condition>` utwórz co najmniej jeden element, który definiuje warunek, który musi być zgodny z żądaniem przesyłania w celu wywołania procesu `Connect:Direct` zdefiniowanego w kroku 1. Te elementy mogą być elementami `<match>` lub elementami `<defined>`.
 - Użyj elementu `<match>`, aby określić, że wartość zmiennej musi być zgodna ze wzorcem. Utwórz element `<match>` z następującymi atrybutami:
 - `variable` -nazwa zmiennej, której wartość jest porównywana. Zmienna jest nieodłącznym symbolem. Więcej informacji na ten temat zawiera sekcja [Zmienne podstawiania w celu użycia z procesami Connect:Direct zdefiniowanymi przez użytkownika](#).
 - `value` -wzorec do porównania z wartością określonej zmiennej.
 - Opcjonalnie: `pattern` -typ wzorca używany przez wartość atrybutu `value`. Ten typ wzorca może mieć wartość `wildcard` lub `regex`. Ten atrybut jest opcjonalny, a wartością domyślną jest `wildcard`.
 - Użyj elementu `<defined>`, aby określić, że zmienna musi mieć zdefiniowaną wartość. Utwórz element `<defined>` z następującym atrybutem:
 - `variable` -nazwa zmiennej, która musi mieć zdefiniowaną wartość. Zmienna jest nieodłącznym symbolem. Więcej informacji na ten temat zawiera sekcja [Zmienne podstawiania w celu użycia z procesami Connect:Direct zdefiniowanymi przez użytkownika](#).

Warunki określone w elemencie `<condition>` są łączone z logicznym AND. Aby agent mostu `Connect:Direct` mógł wywołać proces określony przez ten element `<processSet>`, muszą być spełnione wszystkie warunki. Jeśli element `<condition>` nie zostanie określony, zestaw procesów będzie zgodny z wszystkimi transferami.

6. Wewnątrz elementu `<processSet>` utwórz element `<process>`.
7. Wewnątrz elementu `<process>` utwórz element `<transfer>`.

Element przesyłania określa proces `Connect:Direct`, który jest używany przez agenta mostu `Connect:Direct` jako część przesyłania. Utwórz element `<transfer>` z następującym atrybutem:

 - `process` - położenie procesu `Connect:Direct` zdefiniowanego w kroku 1. Położenie tego pliku jest określane przy użyciu ścieżki bezwzględnej lub względnej w stosunku do katalogu `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent_name`.

Wyniki

Podczas wyszukiwania zgodności warunku agent mostu `Connect:Direct` wyszukuje od początku pliku na końcu pliku. Pierwszy znaleziony zgodny element to ten, który jest używany.

Zadania pokrewne

[Konfigurowanie mostu Connect:Direct](#)

Odsyłacze pokrewne

[Format pliku definicji procesu Connect:Direct](#)

[fteCreateCDAgent: tworzenie agenta mostu Connect:Direct](#)

Używanie wbudowanych zmiennych symbolicznych w procesach produktu

Connect:Direct wywołanych przez produkt Managed File Transfer

Proces Connect:Direct definiowany przez użytkownika można wywołać z operacji przesyłania Managed File Transfer i przekazać informacje z operacji przesyłania do procesu Connect:Direct za pomocą wbudowanych zmiennych symbolicznych w definicji procesu.

O tym zadaniu

W tym przykładzie używane są wbudowane zmienne symboliczne w celu przekazania informacji z operacji przesyłania Managed File Transfer do procesu Connect:Direct zdefiniowanego przez użytkownika. Więcej informacji na temat wbudowanych zmiennych symbolicznych używanych przez produkt Managed File Transfer zawiera sekcja [Zmienne podstawiania używane z procesami Connect:Direct zdefiniowanymi przez użytkownika](#).

W tym przykładzie plik jest przesyłany z partycji Managed File Transfer Agent do węzła mostu Connect:Direct. Pierwsza część przesyłania jest wykonywana przez produkt Managed File Transfer. Druga część przesyłania jest wykonywana przez proces Connect:Direct zdefiniowany przez użytkownika.

Procedura

1. Utwórz proces produktu Connect:Direct, który korzysta z wbudowanych zmiennych symbolicznych.

```
%FTEPNAME PROCESS
  SNODE=%FTESNODE
  PNODEID=(%FTEPUSER,%FTEPPASS)
  SNODEID=(%FTESUSER,%FTESPASS)

COPY001 COPY
  FROM (
    FILE=%FTEFFILE
    DISP=%FTEFDISP
  )
  TO (
    FILE=%FTETFILE
    DISP=%FTETDISP
  )
PEND
```

2. Zapisz ten proces w pliku tekstowym w następującym miejscu: `MQ_DATA_PATH/mqft/config/coordination_queue_manager/agents/cd_bridge_agent/Example.cdp`
3. Zmodyfikuj plik `ConnectDirectProcessDefinition.xml` w taki sposób, aby zawierał regułę, która wywołuje proces Connect:Direct utworzony w kroku 1.

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:cdprocess xmlns:tns="http://wmqfte.ibm.com/ConnectDirectProcessDefinitions"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/
  ConnectDirectProcessDefinitions ConnectDirectProcessDefinitions.xsd">

  <tns:processSet>
    <tns:condition>
      <tns:match variable="%FTESNODE" value="TOBERMORY" pattern="wildcard" />
    </tns:condition>
    <tns:process>
      <tns:transfer process="Example.cdp" />
    </tns:process>
  </tns:processSet>

</tns:cdprocess>
```

W tym przykładzie, jeśli żądanie przesyłania zostanie wysłane do agenta mostu Connect:Direct, który ma TOBERMORY jako jego źródłowy lub docelowy węzeł Connect:Direct, zostanie wywołany proces Example.cdp Connect:Direct.

4. Wyślij żądanie przesyłania pliku, które spełnia warunki zdefiniowane w pliku ConnectDirectProcessDefinition.xml w kroku 3.

Na przykład składnia

```
fteCreateTransfer -sa ORINOCO -da CD_BRIDGE
                 -sm QM_WIMBLEDON -dm QM_COMMON
                 -de overwrite -df TOBERMORY:/home/bulgaria/destination.txt
                 -sd leave c:\bungo\source.txt
```

W tym przykładzie docelowym węzłem produktu Connect:Direct jest TOBERMORY. Ten węzeł jest węzłem drugorzędym w operacji przesyłania, a wartość parametru %FTESNODE jest ustawiona na TOBERMORY. Ta komenda jest zgodna z warunkiem, który jest ustawiony w pliku ConnectDirectProcessDefinition.xml.

5. Program Managed File Transfer przesyła plik źródłowy do tymczasowego położenia w tym samym systemie, co agent mostu Connect:Direct.
6. Agent mostu Connect:Direct ustawia wartości wewnętrznych zmiennych symbolicznych na podstawie informacji znajdujących się w żądaniu przesyłania i informacji o konfiguracji.

Wewnętrzne zmienne symboliczne są ustawiane na następujące wartości:

- %FTEPNAME=*nazwa_procesu* -ta wartość jest 8-znakową nazwą procesu wygenerowaną przez agenta mostu Connect:Direct.
- %FTESNODE=TOBERMORY-ta wartość jest ustawiana na podstawie parametru **-df** komendy **fteCreateTransfer**.
- %FTEPUSER,=*użytkownik_węzła_podstawowego* -informacje te są pobierane z pliku ConnectDirectCredentials.xml.
- %FTEPPASS=*hasło_użytkownika_podstawowego* -te informacje są pobierane z pliku ConnectDirectCredentials.xml.
- %FTESUSER,=*użytkownik_węzła_drugiego* -te informacje są pobierane z pliku ConnectDirectCredentials.xml.
- %FTESPASS=*hasło_użytkownika_wtórniego* -informacje te są pobierane z pliku ConnectDirectCredentials.xml.
- %FTEFFILE =*miejsca_tymczasowa* -ta wartość jest tymczasową lokalizacją pliku w tym samym systemie, co agent mostu Connect:Direct.
- %FTEFDISP=leave-Ta wartość jest ustawiana na podstawie parametru **-sd** komendy **fteCreateTransfer**.
- %FTETFILE=/home/bulgaria/destination.txt -Ta wartość jest ustawiana na podstawie parametru **-df** komendy **fteCreateTransfer**.
- %FTETDISP=overwrite-ta wartość jest ustawiana za pomocą parametru **-de** komendy **fteCreateTransfer**.

7. Proces Connect:Direct jest uruchamiany w węźle mostu Connect:Direct. Program Connect:Direct przesyła plik z położenia tymczasowego w systemie mostu Connect:Direct do miejsca docelowego /home/bulgaria/destination.txt w systemie, w którym działa węzeł TOBERMORY produktu Connect:Direct.

Pojęcia pokrewne

[“Wysyłanie zdefiniowanego przez użytkownika procesu Connect:Direct z żądania przesyłania plików” na stronie 301](#)

Istnieje możliwość wprowadzenia żądania przesyłania dla operacji przesyłania, która przechodzi przez agent mostu Connect:Direct, który wywołuje zdefiniowany przez użytkownika proces Connect:Direct w ramach przesyłania plików.

Odsyłacze pokrewne

Zmienne podstawiane do użycia z procesami Connect:Direct zdefiniowanymi przez użytkownika

Korzystanie z procesów produktu Connect:Direct do przesyłania żądań transferu produktu Managed File Transfer

Żądanie transferu można wysłać do agenta mostu Connect:Direct z procesu Connect:Direct . Produkt Managed File Transfer udostępnia komendy, które można wywoływać z instrukcji **RUN TASK** w procesie produktu Connect:Direct .

Produkt Managed File Transfer udostępnia następujące komendy do użycia z procesami produktu Connect:Direct :

ftetag

Tę komendę należy określić w kroku poprzedzającym komendę **ftebxfer** lub **ftecxfer** w celu utworzenia wymaganych informacji kontrolnych dla operacji przesyłania. Ta komenda przyjmuje specyfikację źródła przesyłania jako parametr. Więcej informacji na temat formatu specyfikacji źródła zawiera sekcja **fteCreateTransfer: uruchomienie nowego przesyłania plików**.

ftebxfer

Tę komendę należy określić, aby utworzyć żądanie przesyłania plików, gdy menedżer kolejek, do którego wysyłane jest żądanie przesyłania, znajduje się w tym samym systemie, co węzeł Connect:Direct , który wprowadza komendę. Ta komenda przyjmuje te same parametry, co komenda **fteCreateTransfer** . Więcej informacji na temat tych parametrów zawiera sekcja **fteCreateTransfer: uruchomienie nowego przesyłania plików**. Ta komenda ma również dodatkowy parametr:

-qmgrname

Wymagane. Nazwa menedżera kolejek, do którego ma zostać nadana komenda.

ftecxfer

Tę komendę należy określić, aby utworzyć żądanie przesyłania plików, gdy menedżer kolejek, do którego wysyłane jest żądanie przesyłania, znajduje się w innym systemie niż węzeł Connect:Direct , który wprowadza komendę. Ta komenda przyjmuje te same parametry, co komenda **fteCreateTransfer** . Więcej informacji na temat parametrów zawiera sekcja **fteCreateTransfer: uruchomienie nowego przesyłania plików**. Ta komenda ma również trzy dodatkowe parametry:

-qmgrname

Wymagane. Nazwa menedżera kolejek, do którego ma zostać nadana komenda.

-connname

Wymagane. Host i port menedżera kolejek, do którego należy wprowadzić komendę, określoną w formacie CONNAME IBM MQ . Na przykład: host . example . com (1337) .

-channelname

Opcjonalne. Nazwa kanału, który ma być używany do łączenia się z menedżerem kolejek w celu wprowadzenia komendy. Jeśli opcja ta nie zostanie podana, zostanie użyta wartość domyślna SYSTEM.DEF.SVRCONN jest używany.

Zadania pokrewne

“Tworzenie i wprowadzanie procesu produktu Connect:Direct , który wywołuje produkt Managed File Transfer przy użyciu requestera Connect:Direct” na stronie 305

Program Connect:Direct Requester jest graficznym interfejsem użytkownika, którego można użyć do utworzenia i wysłania procesu Connect:Direct , który wywołuje Managed File Transfer.

Odsyłacze pokrewne

Przykład: plik procesu Connect:Direct , który wywołuje komendy MFT.

Tworzenie i wprowadzanie procesu produktu Connect:Direct , który wywołuje produkt Managed File Transfer przy użyciu requestera Connect:Direct

Program Connect:Direct Requester jest graficznym interfejsem użytkownika, którego można użyć do utworzenia i wysłania procesu Connect:Direct , który wywołuje Managed File Transfer.

O tym zadaniu

W tym zadaniu opisano sposób tworzenia procesu Connect:Direct , który wywołuje komendę Managed File Transfer **ftecxfer** lub komendę **ftebxfer** . Użyj komendy **ftecxfer** , gdy menedżer kolejek, do którego wysłano żądanie przesyłania, znajduje się w innym systemie, do węzła Connect:Direct , który wprowadza komendę. Użyj komendy **ftebxfer** , gdy menedżer kolejek, do którego wysłano żądanie przesyłania, znajduje się w tym samym systemie, co węzeł Connect:Direct , który wprowadza komendę. Komenda **ftecxfer** umożliwia nawiązanie połączenia klienta z menedżerem kolejek agenta źródłowego agenta przesyłania. Przed wywołaniem komendy **ftecxfer** należy wywołać komendę **ftetag** i przekazać ją do informacji o specyfikacji źródłowej. Umożliwia to protokołowanie i kontrolę procesu w taki sam sposób, jak operacje przesyłania zainicjowane z produktu Managed File Transfer.

Procedura

1. Uruchom program Requester Connect:Direct .
2. Na karcie **Węzły** panelu wybierz węzeł Connect:Direct , który jest używany jako podstawowy węzeł procesu.
3. Wybierz opcję **Plik > Nowy > Proces**. Zostanie otwarte okno **Właściwości procesu** .
4. W polu **Nazwa:** wpisz nazwę procesu.
5. Wybierz węzeł dodatkowy z listy **Snode > Nazwa:** .
6. Z listy **Snode > Operating System:** wybierz system operacyjny węzła drugorzędneho.
7. Opcjonalne: W tym oknie należy wypełnić wszelkie wymagane informacje.
8. Kliknij przycisk **OK**. Okno **Właściwości procesu** zostanie zamknięte.
9. Utwórz instrukcję, która uruchamia komendę Managed File Transfer **ftetag** .
 - a) Kliknij prawym przyciskiem myszy w oknie **Proces** w instrukcji **End** .
 - b) Wybierz opcję **Wstaw > Uruchom zadanie**. Zostanie otwarte okno **Uruchom instrukcję zadania** .
 - c) W polu **Etykieta:** wpisz Tag.
 - d) W polu **Parametry opcjonalne lub komendy** wpisz `pgm(MQ_INSTALLATION_PATH/bin/ftetag) args(source_specification)`. Więcej informacji na temat formatu specyfikacja_źródła zawiera sekcja **ftecCreateTransfer: uruchomienie nowego przesyłania plików**.
 - e) Kliknij przycisk **OK**. Okno **Uruchom instrukcję zadania** zostanie zamknięte.
10. Utwórz instrukcję, która uruchamia komendę Managed File Transfer **ftecxfer** lub **ftebxfer** .
 - a) Kliknij prawym przyciskiem myszy w oknie **Proces** w instrukcji **End** .
 - b) Wybierz opcję **Wstaw > Uruchom zadanie**. Zostanie otwarte okno **Uruchom instrukcję zadania** .
 - c) W polu **Etykieta:** wpisz Transfer.
 - d) W polu **Parametry opcjonalne lub komendy** wpisz `pgm(MQ_INSTALLATION_PATH/bin/ftecxfer) args(parameters)` lub `pgm(MQ_INSTALLATION_PATH/bin/ftebxfer) args(parameters)` , w zależności od wybranej komendy. Parametry używane przez komendy **ftecxfer** i **ftebxfer** są takie same, jak parametry używane przez komendę **ftecCreateTransfer** oraz dodatkowe parametry specyficzne dla produktów **ftecxfer** i **ftebxfer**. Więcej informacji na ten temat zawiera sekcja **ftecCreateTransfer: uruchamianie nowego przesyłania plików** i “Korzystanie z procesów produktu Connect:Direct do przesyłania żądań transferu produktu Managed File Transfer” na stronie 305.
 - e) Kliknij przycisk **OK**. Okno **Uruchom instrukcję zadania** zostanie zamknięte.
11. Opcjonalne: Utwórz dodatkowe instrukcje wymagane przez użytkownika.
12. Wyślij proces.
 - a) Kliknij prawym przyciskiem myszy w oknie **Proces** .
 - b) Wybierz opcję **Wyślij**. Zostanie otwarte okno **Connect:Direct Attach** (Połączenie: bezpośrednie).
 - c) Wprowadź nazwę użytkownika i hasło, które mają być używane do uruchamiania procesu.

d) Kliknij przycisk **OK**.

Pojęcia pokrewne

“Korzystanie z procesów produktu Connect:Direct do przesyłania żądań transferu produktu Managed File Transfer” na stronie 305

Żądanie transferu można wysłać do agenta mostu Connect:Direct z procesu Connect:Direct . Produkt Managed File Transfer udostępnia komendy, które można wywoływać z instrukcji **RUN TASK** w procesie produktu Connect:Direct .

Praca z MFT z IBM Integration Bus

Produkt Managed File Transfer można pracować z produktem IBM Integration Bus , korzystając z węzłów FTEOutput i FTEInput.

- Węzeł FTEInput służy do przesyłania pliku w sieci za pomocą programu Managed File Transfer , a następnie do przetwarzania tego pliku jako części przepływu Integration Bus .
- Węzeł FTEOutput służy do przesyłania pliku, który został wyjściowy przez przepływ Integration Bus do innego miejsca w sieci.

Agenty, które przesyłają pliki do lub z agenta brokera, mogą być na dowolnym poziomie produktu Managed File Transfer.

Więcej informacji na ten temat można znaleźć w dokumentacji produktu IBM Integration Bus.

Odtwarzanie i restartowanie produktu MFT

If your agent or queue manager are unavailable for any reason, for example because of a power or network failure, Managed File Transfer recovers as follows in these scenarios:

- Zwykle, jeśli wystąpił problem podczas przesyłania pliku, program Managed File Transfer odtwarza i restartuje ten transfer plików po naprawieniu problemu.
- Jeśli plik, który był w trakcie przesyłania, zostanie usunięty lub zmieniony w czasie, gdy agent lub menedżer kolejek są niedostępne, operacja przesyłania nie powiedzie się i zostanie wyświetlony komunikat w dzienniku przesyłania zawierający szczegółowe informacje na temat tego niepowodzenia.
- Jeśli proces agenta nie powiedzie się podczas przesyłania pliku, operacja przesyłania będzie kontynuowana po zrestartowaniu agenta.
- Jeśli agent utraci połączenie z menedżerem kolejek agenta, agent czeka podczas próby ponownego nawiązania połączenia z menedżerem kolejek. Gdy agent pomyślnie ponownie nawiąże połączenie z menedżerem kolejek, bieżące przesyłanie będzie kontynuowane.
- Jeśli agent jest zatrzymany z dowolnej przyczyny, wszystkie monitory zasobów powiązane z odpytywaniem zatrzymują agenta. Gdy agent odtwarza, monitory są także restartowane, a odpytywanie zasobów jest wznowiane.
- W przypadku operacji przesyłania plików z dyspozycją `delete`, jeśli odtwarzanie następuje po wysłaniu wszystkich danych z agenta źródłowego do agenta docelowego, plik źródłowy zostanie odblokowany przed usunięciem. Odblokowanie to oznacza, że plik źródłowy może być modyfikowany przed usunięciem pliku. Dlatego też uważa się, że usunięcie pliku źródłowego jest niebezpieczne i wyświetlane jest następujące ostrzeżenie:

```
BFGTR0075W: The source file has not been deleted because it is possible that the source file was modified after the source file was transferred.
```

W takim przypadku należy sprawdzić, czy treść pliku źródłowego nie została zmodyfikowana, a następnie ręcznie usunąć plik źródłowy.

Status transferów można sprawdzić w IBM MQ Explorer. Jeśli jakiegokolwiek transfery są wyświetlane jako `Stalled`, może być konieczne podjęcie działań naprawczych, ponieważ status wstrzymany oznacza problem z agentem lub między dwoma agentami zaangażowanymi w przesyłanie.

Zadania pokrewne

[“Ustawianie limitu czasu na odzyskiwanie wstrzymanych transferów”](#) na stronie 308

Można ustawić limit czasu odzyskiwania transferu dla wstrzymanych transferów plików, który ma zastosowanie do wszystkich transferów dla agenta źródłowego. Można również ustawić limit czasu odzyskiwania transferu dla pojedynczego przesyłania. Jeśli zostanie ustawiony określony czas (w sekundach), podczas którego agent źródłowy próbuje odzyskać wstrzymane przesyłanie plików, a operacja przesyłania nie powiedzie się, gdy agent osiągnie limit czasu, przesyłanie nie powiedzie się.

Ustawianie limitu czasu na odzyskiwanie wstrzymanych transferów

Można ustawić limit czasu odzyskiwania transferu dla wstrzymanych transferów plików, który ma zastosowanie do wszystkich transferów dla agenta źródłowego. Można również ustawić limit czasu odzyskiwania transferu dla pojedynczego przesyłania. Jeśli zostanie ustawiony określony czas (w sekundach), podczas którego agent źródłowy próbuje odzyskać wstrzymane przesyłanie plików, a operacja przesyłania nie powiedzie się, gdy agent osiągnie limit czasu, przesyłanie nie powiedzie się.

O tym zadaniu

W programie IBM MQ 9.1 można ustawić limit czasu odtwarzania transferu, który ma zastosowanie do wszystkich operacji przesyłania dla agenta źródłowego, dodając parametr limitu czasu odtwarzania przesyłania do pliku `agent.properties` agenta. Można również ustawić limit czasu odzyskiwania transferu dla pojedynczego przesyłania z wiersza komend lub za pomocą programu IBM MQ Explorer lub za pomocą zadań Apache Ant. Jeśli w pliku `agent.properties` ustawiono wartość limitu czasu odtwarzania przesyłania, to ustawienie limitu czasu odtwarzania przesyłania dla pojedynczego przesyłania wartość w pliku `agent.properties`.

Istnieją trzy opcje limitu czasu odtwarzania przesyłania:

- Agent będzie nadal próbował odtworzyć wstrzymany transfer do czasu jego pomyślnego zakończenia. Jest to takie samo działanie, jak domyślne działanie agenta, jeśli nie ustawiono limitu czasu odtwarzania przesyłania.
- Agent zaznacza, że przesyłanie nie powiodło się natychmiast po wprowadzeniu odtwarzania.
- Agent ciągle ponawia wstrzymany transfer przez określoną ilość czasu, zanim operacja przesyłania zostanie oznaczona jako zakończona niepowodzeniem.

Ustawienie limitu czasu odtwarzania przesyłania plików jest opcjonalne. Jeśli ta opcja nie zostanie ustawiona, transfery są zgodne z domyślnym zachowaniem. Jest to takie samo działanie, jak domyślne działanie agenta źródłowego Managed File Transfer przed IBM MQ 9.1, w którym agent próbuje odzyskać wstrzymany transfer do czasu, aż zakończy się pomyślnie.

Pojęcia pokrewne

[“Odtwarzanie i restartowanie produktu MFT”](#) na stronie 307

If your agent or queue manager are unavailable for any reason, for example because of a power or network failure, Managed File Transfer recovers as follows in these scenarios:

Pojęcia dotyczące limitu czasu odtwarzania transferu

Można ustawić czas (w sekundach), podczas którego agent źródłowy próbuje odzyskać wstrzymany plik przesyłania plików. Jeśli operacja przesyłania nie powiedzie się, gdy agent osiągnie limit czasu dla odstępu czasu między ponownymi próbami, przesyłanie nie powiedzie się.

Kolejność przekroczenia limitu czasu odtwarzania

Wartość limitu czasu odtwarzania przesyłania dla pojedynczego transferu określonego za pomocą komend `fteCreateTransfer`, `fteCreateTemplate` lub `fteCreateMonitor` lub za pomocą składnika IBM MQ Explorer lub określonego w zagnieżdżonym elemencie `fte:filespec` ma pierwszeństwo przed wartością określoną dla parametru `transferRecoveryTimeout` w pliku `agent.properties` dla agenta źródłowego.

Na przykład, jeśli komenda **fteCreateTransfer** zostanie uruchomiona bez pary parametrów **-rt** i wartości, agent źródłowy AGENT1 sprawdza plik `agent.properties` pod kątem wartości **transferRecoveryTimeout**, aby określić zachowanie limitu czasu odtwarzania:

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -df C:\import\transferredfile.txt
C:\export\originalfile.txt
```

Jeśli parametr **transferRecoveryTimeout** w pliku `agent.properties` nie jest ustawiony lub jest ustawiony na wartość **-1**, agent postępuje zgodnie z domyślnym zachowaniem i próbuje odzyskać transfer do czasu, aż zakończy się on pomyślnie.

Jeśli jednak komenda **fteCreateTransfer** zawiera parametr **-rt**, to wartość tego parametru ma pierwszeństwo przed wartością w pliku `agent.properties` i jest używana jako ustawienie limitu czasu odtwarzania dla operacji przesyłania:

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -rt 21600 -df C:\import\transferredfile.txt
C:\export\originalfile.txt
```

Licznik limitu czasu odtwarzania

Licznik limitu czasu odtwarzania rozpoczyna się, gdy operacja przesyłania przechodzi do stanu odtwarzania. Komunikat dziennika przesyłania jest publikowany w systemie SYSTEM.FTE z łańcuchem tematu `Log/agent_name/transfer_ID`, który wskazuje, że status przesyłania jest zmieniany na odzyskiwanie oraz czas zegara agenta źródłowego, o którym zmieniony został status. Jeśli operacja przesyłania zostanie wznowiona w przedziale czasu między ponownymi próbami i nie osiągnie limitu czasu odtwarzania (licznik <= limit czasu odtwarzania), licznik zostanie zresetowany do wartości 0, a następnie będzie gotowy do ponownego uruchomienia, jeśli operacja przesyłania zostanie przywróta.

Jeśli licznik osiągnie maksymalną wartość ustawioną dla limitu czasu odtwarzania (licznik == przekroczenie limitu czasu odtwarzania), odtwarzanie zostanie zatrzymane, a agent źródłowy zgłosi przesyłanie jako nieudane. Ten typ niepowodzenia przesyłania spowodowany przez to, że operacja przesyłania osiągnęła limit czasu odtwarzania, jest wskazywany przez kod komunikatu LIMIT CZASU ODTWARZANIA (69). Inny komunikat dziennika przesyłania jest publikowany w systemie SYSTEM.FTE, z łańcuchem tematu `Log/agent_name/transfer_ID`, który wskazuje, że operacja przesyłania nie powiodła się i zawiera komunikat, kod powrotu i dziennik zdarzeń agenta źródłowego. Dziennik zdarzeń agenta źródłowego jest aktualizowany za pomocą komunikatu, jeśli podczas odtwarzania wystąpią dowolne z następujących zdarzeń:

- Jeśli parametr limitu czasu odtwarzania jest ustawiony na wartość większą niż **-1**, transfer zostanie wprowadzony do odtwarzania. Dziennik zdarzeń agenta jest aktualizowany w celu wskazania uruchomienia licznika czasu odtwarzania dla **TransferId** oraz czasu oczekiwania przez agenta źródłowego, zanim zainicjuje przetwarzanie limitu czasu odtwarzania.
- Po wznowieniu odtwarzania dziennik zdarzeń agenta źródłowego zostanie zaktualizowany z nowym komunikatem w celu wskazania, że **TransferId**, które było w odtwarzaniu, jest wznowiane.
- Po przekroczeniu limitu czasu transferu z powodu przekroczenia limitu czasu dziennik zdarzeń agenta źródłowego jest aktualizowany w celu wskazania **TransferId**, który uległ awarii podczas odtwarzania z powodu przekroczenia limitu czasu odtwarzania.

Te komunikaty dziennika umożliwiają użytkownikom (subskrybentom i loggerom) identyfikowanie operacji przesyłania, które nie powiodły się z powodu przekroczenia limitu czasu odtwarzania przesyłania.

Licznik limitu czasu odtwarzania jest zawsze w agencji źródłowym. Jeśli jednak agent docelowy nie odbierze informacji od agenta źródłowego w odpowiednim czasie, może wystać żądanie do agenta źródłowego, aby umieścić go w odtwarzaczy. W przypadku operacji przesyłania, w której ustawiona jest opcja limitu czasu odtwarzania, agent źródłowy uruchamia licznik limitu czasu odtwarzania po odebraniu żądania od agenta docelowego.

Obsługa ręczna jest nadal wymagana dla transferów, które nie korzystają z opcji limitu czasu odtwarzania, zakończonych niepowodzeniem i częściowo kompletnych transferów.

W przypadku zestawów przesyłania, w których pojedyncze żądanie przesyłania jest wysyłane dla wielu plików, a niektóre z nich zostały pomyślnie zakończone, ale jedno zostało zakończone tylko częściowo, przesyłanie jest nadal oznaczone jako zakończone niepowodzeniem, ponieważ nie zostało zakończone zgodnie z oczekiwaniami. Agent źródłowy mógł zostać przekroczony podczas przesyłania częściowo ukończonych plików.

Upewnij się, że agent docelowy i serwer plików są gotowe i w stanie akceptować transfery plików.

Należy ponownie wprowadzić żądanie przesyłania dla całego zestawu, ale aby uniknąć problemów, ponieważ niektóre pliki pozostają w miejscu docelowym od początkowej próby przesyłania, można wprowadzić nowe żądanie z określoną opcją `overwrite if existing`. Gwarantuje to, że niekompletny zestaw plików z poprzedniej próby przesyłania zostanie wyczyszczony jako część nowego transferu, zanim pliki zostaną ponownie zapisane w miejscu docelowym.

V 9.2.0 W produkcie IBM MQ 9.1.5 nie jest już konieczne ręczne usuwanie plików części pozostawionych w miejscu docelowym po nieudanej próbie przesyłania początkowego. Jeśli dla operacji przesyłania zostanie ustawiony limit czasu odtwarzania przesyłania, agent źródłowy przesuwa operację przesyłania do stanu `RecoveryTimedwyjściowe`, jeśli nastąpi przekroczenie limitu czasu odtwarzania. Po ponownym zsynchronizowaniu przesyłania agent docelowy usuwa wszystkie pliki części, które zostały utworzone podczas przesyłania, i wysyła komunikat o zakończeniu do agenta źródłowego.

Dane śledzenia i komunikaty

Punkty śledzenia są uwzględniane w celach diagnostycznych. Rejestrowana jest wartość limitu czasu odtwarzania, początek okresu ponawiania, początek okresu wznowienia i licznika licznika oraz informacja o tym, czy nastąpiło przekroczenie limitu czasu operacji przesyłania, czy też nie powiodło się. W przypadku wystąpienia problemu lub nieoczekiwanego zachowania można zgromadzić dzienniki wyjściowe agenta źródłowego i pliki śledzenia, a następnie udostępnić je na żądanie działu wsparcia produktu IBM, aby pomóc w rozwiązywaniu problemów.

Komunikaty powiadamiają o następujących przypadkach:

- Operacja przesyłania jest wprowadzana do odtwarzania (`BFGTR0081I`)
- Przesyłanie zostało zakończone, ponieważ nastąpiło przekroczenie limitu czasu z poziomu odtwarzania (`BFGSS0081E`)
- Po odtwarzaniu Atransfer jest wznawiany (`BFGTR0082I`)

Pojęcia pokrewne

“Odtwarzanie i restartowanie produktu MFT” na stronie 307

If your agent or queue manager are unavailable for any reason, for example because of a power or network failure, Managed File Transfer recovers as follows in these scenarios:

Ustawianie limitu czasu odzyskiwania transferu dla wszystkich operacji przesyłania dla jednego agenta źródłowego

Można ustawić limit czasu odzyskiwania transferu, który ma zastosowanie do wszystkich operacji przesyłania dla agenta źródłowego, dodając parametr `transferRecoveryTimeout` do pliku `agent.properties`.

O tym zadaniu

Aby ustawić limit czasu odtwarzania dla agenta źródłowego, który ma zastosowanie do wszystkich operacji przesyłania dla agenta źródłowego, należy dodać parę parametrów i wartości dla `transferRecoveryTimeout` do pliku `agent.properties`.

Istnieją trzy opcje dla parametru `transferRecoveryTimeout` :

-1

Agent będzie kontynuował próby odzyskania wstrzymanego przesyłania, dopóki operacja przesyłania nie powiedzie się. Użycie tej opcji jest równoznaczne z domyślnym zachowaniem agenta, gdy właściwość nie jest ustawiona.

0

Agent zatrzymuje przesyłanie pliku natychmiast po wejściu w proces odtwarzania.

>0

Agent będzie kontynuował próbę odzyskania wstrzymanego przesyłania przez ilość czasu w sekundach określoną przez określoną dodatnią liczbę całkowitą.

Wszelkie zmiany wprowadzone w pliku `agent.properties` są wprowadzane w życie dopiero po zrestartowaniu agenta.

Jeśli jest to wymagane, można przesłonić wartość limitu czasu odtwarzania przesyłania w pliku `agent.properties` dla pojedynczego przesyłania. Więcej informacji na ten temat zawiera sekcja [“Ustawianie limitu czasu odzyskiwania transferu dla poszczególnych transferów”](#) na stronie 311.

Procedura

- Aby określić, że agent będzie nadal próbował odtworzyć wstrzymany transfer do czasu jego pomyślnego zakończenia, należy ustawić wartość limitu czasu odtwarzania przesyłania -1, jak to pokazano w poniższym przykładzie:

```
transferRecoveryTimeout=-1
```

- Aby określić, że agent oznacza przesyłanie jako nieudane od razu po wprowadzeniu odtwarzania, należy ustawić wartość limitu czasu odtwarzania przesyłania 0, jak to pokazano w poniższym przykładzie:

```
transferRecoveryTimeout=0
```

- Aby określić, że agent będzie ponawiał próby wstrzymanego przesyłania przez określoną ilość czasu, zanim operacja przesyłania zostanie oznaczona jako nieudana, należy ustawić wartość limitu czasu odtwarzania transferu na czas (w sekundach), który ma być ponawiany.

Na przykład ustawienie wartości 21600 limitu czasu odtwarzania przesyłania oznacza, że agent nadal próbuje odzyskać transfer przez sześć godzin od momentu, w którym następuje odtwarzanie:

```
transferRecoveryTimeout=21600
```

Maksymalna wartość tego parametru to 999999999.

Ustawianie limitu czasu odzyskiwania transferu dla poszczególnych transferów

Limit czasu odzyskiwania przesyłania można ustawić dla pojedynczego przesyłania z wiersza komend lub za pomocą programu IBM MQ Explorer lub za pomocą zadań Apache Ant. Jeśli w pliku `agent.properties` ustawiono wartość limitu czasu odtwarzania przesyłania, to ustawienie limitu czasu odtwarzania dla operacji przesyłania dla pojedynczego przesyłania powoduje nadpisanie wartości ustawionej w pliku `agent.properties`.

O tym zadaniu

Parametr limitu czasu odtwarzania przesyłania można ustawić dla pojedynczego przesyłania w następujących przypadkach:

- Operację przesyłania można utworzyć za pomocą komendy **fteCreateTransfer** lub za pomocą programu IBM MQ Explorer.
- Tworzenie szablonu przesyłania za pomocą komendy **fteCreateTemplate** lub za pomocą programu IBM MQ Explorer.

- Tworzenie monitora zasobów za pomocą komendy **fteCreateMonitor** lub za pomocą programu IBM MQ Explorer.
- Kopiowanie lub przenoszenie plików przy użyciu zadań `fte: filecopy` lub `fte: filemove` Ant .

Jeśli dla pojedynczego transferu zostanie ustawiona wartość limitu czasu odtwarzania przesyłania, ta wartość nadpisuje wartość limitu czasu odtwarzania transferu ustawioną w pliku `agent.properties` (patrz "[Ustawianie limitu czasu odzyskiwania transferu dla wszystkich operacji przesyłania dla jednego agenta źródłowego](#)" na stronie 310).

Procedura

- Aby użyć komendy **fteCreateTransfer** lub **fteCreateTemplate** w celu ustawienia limitu czasu odtwarzania przesyłania, należy określić odpowiednią opcję dla parametru **-rt** :

-1

Agent będzie kontynuował próby odzyskania wstrzymanego przesyłania, dopóki operacja przesyłania nie powiedzie się. Użycie tej opcji jest równoznaczne z domyślnym zachowaniem agenta, gdy właściwość nie jest ustawiona.

0

Agent zatrzymuje przesyłanie pliku natychmiast po wejściu w proces odtwarzania.

>0

Agent będzie kontynuował próby odzyskania wstrzymanego przesyłania przez określony czas w sekundach.

Przykłady komendy **fteCreateTransfer**

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -rt -1 -df C:\import\transferredfile.txt
C:\export\originalfile.txt
```

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -rt 0 -df C:\import\transferredfile.txt
C:\export\originalfile.txt
```

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -rt 21600 -df C:\import\transferredfile.txt
C:\export\originalfile.txt
```

Przykłady komendy **fteCreateTemplate**

```
fteCreateTemplate -tn "payroll accounts monthly report template" -rt -1 -sa PAYROLL -sm
QM_PAYROLL1 -da ACCOUNTS
-dm QM_ACCOUNTS -df C:\payroll_reports\*.xls C:\out\*.xls
```

```
fteCreateTemplate -tn "payroll accounts monthly report template" -rt 0 -sa PAYROLL -sm
QM_PAYROLL1 -da ACCOUNTS
-dm QM_ACCOUNTS -df C:\payroll_reports\*.xls C:\out\*.xls
```

```
fteCreateTemplate -tn "payroll accounts monthly report template" -rt 21600 -sa PAYROLL -sm
QM_PAYROLL1 -da ACCOUNTS
-dm QM_ACCOUNTS -df C:\payroll_reports\*.xls C:\out\*.xls
```

Dla komendy **fteCreateMonitor** nie ma parametru **-rt** . Jeśli parametr **-rt** zostanie ustawiony za pomocą komendy **fteCreateTransfer** , a także zostanie ustawiony parametr **-gt** , parametr limitu czasu odtwarzania zostanie dołączony do dokumentu XML z definicją przesyłania wygenerowaną podczas uruchamiania komendy **fteCreateTransfer** . Następnie monitor zasobów używa tego dokumentu XML podczas uruchamiania komendy **fteCreateMonitor** . W poniższym przykładzie szczegóły limitu czasu odtwarzania transferu zostaną uwzględnione w pliku `task.xml` :

```
fteCreateMonitor -ma AgentName -md C:\mqmft\monitors -mn Monitor_Name -mt task.xml -tr
"fileSize>=5MB,*.zip"
```

- Aby użyć strony IBM MQ Explorer Nowy transfer, Nowy monitor lub Nowy szablon w celu ustawienia limitu czasu odtwarzania przesyłania, należy wybrać wymaganą opcję w polu **Limit czasu odtwarzania przesyłania** (sekundy):

Jako agent źródłowy

Jeśli zostanie wybrana opcja **Jako agent źródłowy**, zostanie użyta wartość parametru **transferRecoveryTimeout** z pliku `agent.properties`, jeśli zostanie ona ustawiona. W przeciwnym razie zostanie zastosowane zachowanie domyślne dla limitu czasu odtwarzania przesyłania.

Pole listy liczbowej

Jeśli w polu listy numerycznej zostanie wprowadzony czas (w sekundach), agent będzie kontynuował próby odzyskania wstrzymanego transferu przez określony czas.

Brak

Jeśli zostanie wybrana opcja **Brak**, nie zostanie ustawiony limit czasu odtwarzania po przeniesieniu, a agent będzie kontynuował próbę odzyskania wstrzymanego przesyłania do czasu, aż operacja przesyłania zakończy się pomyślnie.

- Aby ustawić limit czasu odtwarzania, należy użyć zadań Ant . zawiera opcję i wartość **transferRecoveryTimeout**, z elementami **fte:filecopy** lub **fte:filemove** w celu przenoszenia lub kopiowania plików, na przykład:

Przykład dla **fte:filecopy**

```
<fte:filecopy cmdqmq="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
src="agent1@qm1" dst="agent2@qm2"
rcproperty="copy.result" transferRecoveryTimeout="0">

  <fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/file.bin"/>
</fte:filecopy>
```

Przykład dla **fte:filemove**

```
<fte:filemove cmdqmq="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
src=agent1@qm1 dst="agent2@qm2"
rcproperty="move.result" transferRecoveryTimeout="21600">

  <fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/file.bin"/>
</fte:filemove>
```

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 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](#) .

Pojęcia pokrewne

[MQ Telemetry](#)

Odsyłacze pokrewne

[Właściwości MQXR](#)

Konfigurowanie menedżera kolejek dla telemetrii w systemach Linux i AIX

Aby ręcznie skonfigurować produkt MQ Telemetry, należy wykonać następujące kroki. Jeśli potrzebna jest tylko prosta konfiguracja, która korzysta z identyfikatora użytkownika gościa, można zamiast tego uruchomić kreatora obsługi produktu MQ Telemetry w produkcie IBM MQ Explorer.

Zanim rozpoczniesz

Jeśli potrzebna jest tylko prosta konfiguracja, należy rozważyć użycie obsługi produktu MQ Telemetry w produkcie IBM MQ Explorer. Obsługa ta obejmuje kreatora i przykładową procedurę komendy `sampleMQM`. Te zasoby ustawiają konfigurację początkową przy użyciu identyfikatora użytkownika gościa. Więcej informacji na ten temat zawiera sekcja [Weryfikowanie instalacji produktu MQ Telemetry przy użyciu programu IBM MQ Explorer i programów przykładowych IBM MQ Telemetry Transport](#).

Jeśli potrzebna jest bardziej złożona konfiguracja, która korzysta z innej metody uwierzytelniania, należy wykonać kroki opisane w tej czynności. Rozpocznij od następujących kroków początkowych:

1. Informacje na temat instalowania produktu IBM MQ i opcji MQ Telemetry zawiera sekcja [Uwagi dotyczące instalacji 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

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.

Jeśli produkt `SYSTEM.MQTT.TRANSMIT.QUEUE` nie istnieje, jest tworzony automatycznie, gdy usługa telemetryczna (MQXR) jest uruchamiana po raz pierwszy, i ustawiona jest na użycie identyfikatora użytkownika gościa. To zadanie konfiguruje jednak program MQ Telemetry w taki sposób, aby korzystała z innego schematu autoryzacji. Dla tego zadania należy utworzyć produkt `SYSTEM.MQTT.TRANSMIT.QUEUE` i skonfigurować dostęp do niego przed uruchomieniem usługi telemetrycznej (MQXR).

Uruchom następującą komendę:

```
echo "DEFINE QLOCAL('SYSTEM.MQTT.TRANSMIT.QUEUE') USAGE(XMITQ) MAXDEPTH(100000) | runmqsc qMgr"
```

3. Ustaw domyślną kolejkę transmisji.

Łatwiej jest wysyłać komunikaty bezpośrednio do klientów MQTT, jeśli `SYSTEM.MQTT.TRANSMIT.QUEUE` jest domyślną kolejką transmisji. W przeciwnym razie konieczne jest dodanie definicji kolejki zdalnej dla każdego klienta, który odbiera komunikaty produktu IBM MQ; patrz ["Bezpośrednie wysyłanie wiadomości do klienta"](#) na stronie 319. Należy pamiętać, że zmiana domyślnej kolejki transmisji może kolidować z istniejącą konfiguracją.

Po pierwszym uruchomieniu usługi telemetrycznej (MQXR) nie ustawia ona SYSTEM.MQTT.TRANSMIT.QUEUE jako domyślnej kolejki transmisji dla menedżera kolejek. Aby skonfigurować to ustawienie, należy zmienić domyślną właściwość kolejki transmisji. Można to zrobić za pomocą konsoli IBM MQ Explorer lub za pomocą następującej komendy:

```
echo "ALTER QMGR DEFXMITQ('SYSTEM.MQTT.TRANSMIT.QUEUE')" | runmqsc qMgr
```

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 322 . 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 [“Tworzenie SYSTEM.MQXR.SERVICE”](#) na stronie 315.

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 tak, aby ich tożsamości były jednym z identyfikatorów użytkowników zdefiniowanych w kroku [“4”](#) na stronie 315.

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](#).

Tworzenie SYSTEM.MQXR.SERVICE

Aby utworzyć SYSTEM.MQXR.SERVICE, należy użyć komendy **runMQXRService** .

LTS

```
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')
```

V9.2.4

```
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+" -sf "[DEFAULT]"') +
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')
```

Uwaga: **V 9.2.4** Począwszy od wersji IBM MQ 9.2.4, opcja **-sf** udostępnia nazwę pliku kluczy informacji autoryzacyjnych, który zawiera klucz używany do szyfrowania hasła kanału TLS. Więcej informacji na ten temat zawiera sekcja [Szyfrowanie hasła dla kanałów TLS MQTT](#).

Windows Konfigurowanie menedżera kolejek dla telemetrii w systemie Windows

Aby ręcznie skonfigurować produkt MQ Telemetry, należy wykonać następujące kroki. Jeśli potrzebna jest tylko prosta konfiguracja, która korzysta z identyfikatora użytkownika gościa, można zamiast tego uruchomić kreatora obsługi produktu MQ Telemetry w produkcie IBM MQ Explorer.

Zanim rozpoczniesz

Jeśli potrzebna jest tylko prosta konfiguracja, należy rozważyć użycie obsługi produktu MQ Telemetry w produkcie IBM MQ Explorer. Obsługa ta obejmuje kreatora i przykładową procedurę komendy `sampleMQM`. Te zasoby ustawiają konfigurację początkową przy użyciu identyfikatora użytkownika gościa. Więcej informacji na ten temat zawiera sekcja [Weryfikowanie instalacji produktu MQ Telemetry przy użyciu programu IBM MQ Explorer i programów przykładowych IBM MQ Telemetry Transport](#).

Jeśli potrzebna jest bardziej złożona konfiguracja, która korzysta z innej metody uwierzytelniania, należy wykonać kroki opisane w tej czynności. Rozpocznij od następujących kroków początkowych:

1. Informacje na temat instalowania produktu IBM MQ i opcji MQ Telemetry zawiera sekcja [Uwagi dotyczące instalacji 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

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.

Jeśli produkt `SYSTEM.MQTT.TRANSMIT.QUEUE` nie istnieje, jest tworzony automatycznie, gdy usługa telemetryczna (MQXR) jest uruchamiana po raz pierwszy, i ustawiona jest na użycie identyfikatora użytkownika gościa. To zadanie konfiguruje jednak program MQ Telemetry w taki sposób, aby korzystała z innego schematu autoryzacji. Dla tego zadania należy utworzyć produkt `SYSTEM.MQTT.TRANSMIT.QUEUE` i skonfigurować dostęp do niego przed uruchomieniem usługi telemetrycznej (MQXR).

Uruchom następującą komendę:

```
echo DEFINE QLOCAL('SYSTEM.MQTT.TRANSMIT.QUEUE') USAGE(XMITQ) MAXDEPTH(100000) | runmqsc qMgr
```

3. Ustaw domyślną kolejkę transmisji.

Łatwiej jest wysyłać komunikaty bezpośrednio do klientów MQTT, jeśli `SYSTEM.MQTT.TRANSMIT.QUEUE` jest domyślną kolejką transmisji. W przeciwnym razie konieczne jest dodanie definicji kolejki zdalnej dla każdego klienta, który odbiera komunikaty produktu IBM MQ; patrz [“Bezpośrednie wysyłanie wiadomości do klienta”](#) na stronie 319. Należy pamiętać, że zmiana domyślnej kolejki transmisji może kolidować z istniejącą konfiguracją.

Po pierwszym uruchomieniu usługi telemetrycznej (MQXR) nie ustawia ona `SYSTEM.MQTT.TRANSMIT.QUEUE` jako domyślnej kolejki transmisji dla menedżera kolejek. Aby skonfigurować to ustawienie, należy zmienić domyślną właściwość kolejki transmisji. Można to zrobić za pomocą konsoli IBM MQ Explorer lub za pomocą następującej komendy:

```
echo ALTER QMGR DEFXMITQ('SYSTEM.MQTT.TRANSMIT.QUEUE') | runmqsc qMgr
```

4. Aby utworzyć jeden lub więcej identyfikatorów użytkowników, należy wykonać procedurę w produkcji [“Autoryzowanie klientów MQTT do uzyskiwania dostępu do obiektów IBM MQ”](#) na stronie 322. Identyfikatory użytkowników mają uprawnienia do publikowania, subskrybowania i wysyłania publikacji do klientów MQTT.
5. Zainstaluj usługę telemetryczną (MQXR).

```
type installMQXRService_win.mqsc | runmqsc qMgr
```

Patrz także przykładowy kod w produkcji [“Tworzenie SYSTEM.MQXR.SERVICE”](#) na stronie 317.

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 tak, aby ich tożsamości były jednym z identyfikatorów użytkowników zdefiniowanych w kroku [“4”](#) na stronie 317.

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](#).

Tworzenie SYSTEM.MQXR.SERVICE

Aby utworzyć `SYSTEM.MQXR.SERVICE`, należy użyć komendy `runMQXRService`.

```
V 9.2.4
```

```
DEF SERVICE(SYSTEM.MQXR.SERVICE) +  
CONTROL(QMGR) +  
DESCR('Manages clients using MQXR protocols such as MQTT') +  
SERVTYPE(SERVER) +  
STARTCMD('+MQ_INSTALL_PATH+mqx\bin\runMQXRService.bat') +  
STARTARG('-m +QMNAME+ -d "+MQ_Q_MGR_DATA_PATH+\" -g "+MQ_DATA_PATH+\" -sf "[DEFAULT]"') +  
STOPCMD('+MQ_INSTALL_PATH+mqx\bin\endMQXRService.bat') +  
STOPARG('-m +QMNAME+') +  
STDOUT('+MQ_Q_MGR_DATA_PATH+mqx.stdout') +  
STDERR('+MQ_Q_MGR_DATA_PATH+mqx.stderr')
```

Uwaga: **V 9.2.4** Począwszy od wersji IBM MQ 9.2.4, opcja **-sf** udostępnia nazwę pliku kluczy informacji autoryzacyjnych, który zawiera klucz używany do szyfrowania haseł kanału TLS. Więcej informacji na ten temat zawiera sekcja [Szyfrowanie haseł dla kanałów TLS MQTT](#).

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.QUEUEi 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 16 na stronie 319](#).

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 16. 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. SYSTEM.MQTT.TRANSMIT.QUEUE
Jest zgodny z aliasem menedżera kolejek o nazwie <i>ClientIdentifier</i>	<i>ClientIdentifier</i>	<i>niezdefiniowane</i>	<i>ClientIdentifier</i>	<i>niezdefiniowane</i>	SYSTEM.MQTT.TRANSMIT.QUEUE

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 16. 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 wysłanie 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 17. 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 18. 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 17 na stronie 320. Nazwa kolejki zdalnej może być dowolna. Klient otrzymuje go jako łańcuch tematu.

Tabela 17. 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	<code>ClientIdentifier</code>	<code>SYSTEM.MQTT.TRANSMIT.QUEUE</code>

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 bezstanowe i stanowe sesje](#). 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 konieczne 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 dostępnego 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

dostęp 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 323](#)
- [“Kontrola dostępu gruboziarnistego” na stronie 323](#)
- [“Średnioziarnista kontrola dostępu” na stronie 323](#)
- [“Precyzyjna kontrola dostępu” na stronie 323](#)

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 [Tworzenie grup i zarządzanie nimi 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 wysłać 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 Uwierzelnianie klienta MQTT przy użyciu hasła

Uwierzelniaj nazwę użytkownika przy użyciu hasła klienta. Istnieje możliwość uwierzelniania 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 uwierzelniania klienta Nazwa użytkownika. Usługa JAAS korzysta z hasła dostarczonego przez klienta MQTT.

Administrator produktu IBM MQ podejmuje decyzję, czy uwierzelnić nazwę użytkownika, czy nie w ogóle uwierzelnić, konfigurując kanał MQTT, z którym łączy się klient. Klienci mogą być przypisani do różnych kanałów, a każdy kanał może być skonfigurowany do uwierzelniania swoich klientów na różne sposoby. Za pomocą usługi JAAS można skonfigurować, które metody muszą uwierzelniać klienta, i które opcjonalnie mogą uwierzelniać klienta.

Wybór tożsamości do uwierzelniania nie ma wpływu na wybór tożsamości do autoryzacji. Użytkownik może chcieć skonfigurować wspólną tożsamość w celu autoryzacji dla wygody administratora, ale uwierzelnia każdego użytkownika w celu użycia tej tożsamości. Poniższa procedura przedstawia kroki w celu uwierzelniania 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 klientowi prywatnego podpisanego certyfikatu cyfrowego 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 uwierzytelnić 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 326](#)

Administrator produktu IBM MQ konfiguruje kanały pomiarowe 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.

Zadania 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 pomiarowe 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 jest podpisany przez certyfikat pochodzący z ośrodka certyfikacji, publicznie podpisane certyfikaty klienta lub ośrodka certyfikacji muszą być bezpiecznie przechowywane 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, to klient jest 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 328.

Pojęcia pokrewne

[“Konfiguracja kanału telemetrycznego do uwierzytelniania kanału przy użyciu protokołu TLS”](#) na stronie 327

Administrator produktu IBM MQ konfiguruje kanały pomiarowe 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.

Odsyłacze 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 sprawdza zgodność certyfikatów wystanych z serwera z certyfikatami umieszczonymi w jego magazynie zaufanych certyfikatów lub w bazie cacerts środowiska JRE.

Magazyn certyfikatów środowiska JRE jest plikiem 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 OGSi, które korzystały z klienta MQTT, środowisko udostępnia innego dostawcę zaufania.

Aby móc uwierzytelnić kanał pomiarowy przy użyciu protokołu TLS, należy skonfigurować serwer i klient.

Konfiguracja kanału telemetrycznego do uwierzytelniania kanału przy użyciu protokołu TLS

Administrator produktu IBM MQ konfiguruje kanały pomiarowe 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 łańcuch 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 328.

Pojęcia pokrewne

[“Konfiguracja kanału telemetrycznego dla uwierzytelniania klienta MQTT przy użyciu protokołu TLS”](#) na stronie 326

Administrator produktu IBM MQ konfiguruje kanały pomiarowe 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.

Odsyłacze pokrewne

[ZDEFINIUJ KANAŁ \(MQTT\)](#)

[ZMIENŃ KANAŁ \(MQTT\)](#)

[CipherSpecs i CipherSuites](#)

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

[Usługa telemetryczna \(MQXR\)](#)

[“Konfiguracja kanału telemetrycznego do uwierzytelniania kanału przy użyciu protokołu TLS”](#) na stronie 327

Administrator produktu IBM MQ konfiguruje kanały pomiarowe 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.

Odsyłacze pokrewne

[ZDEFINIUJ KANAŁ \(MQTT\)](#)

[ZMIENŃ KANAŁ \(MQTT\)](#)

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 327.

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 uwierzytelnia 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).

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 telemetryczny. Szyfruj komunikaty między klientem i menedżerem kolejek; patrz [“Uwierzytelnianie klienta MQTT przy użyciu protokołu TLS”](#) na stronie 325
2. Połącz klienta MQTT tylko z zaufanym kanałem telemetryczny. Szyfruj komunikaty między klientem i menedżerem kolejek; patrz [“Uwierzytelnianie kanału telemetrycznego przy użyciu protokołu TLS”](#) na stronie 327.
3. Szyfruj komunikaty między klientem i menedżerem kolejek; patrz [“Prywatność publikacji na kanałach telemetrycznych”](#) na stronie 329.

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", "clientId1");
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 połączeń 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.MqttConnectionOptions conOptions = new MqttConnectionOptions();
conOptions.setSSLProperties(sslClientProperties);
```

Nazwy i wartości konkretnych właściwości są opisane w klasie `MqttConnectionOptions`. Odsyłacze do dokumentacji interfejsu API klienta dla bibliotek klienta MQTT zawiera sekcja [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:

Tabela 18. Ścieżki do plików przy użyciu platformy dla plików konfiguracyjnych środowiska JRE TLS

Platforma	Ścieżka do pliku
Windows	<code>WMQ Installation Directory\java\jre\lib\security</code>
AIX and Linux platformy	<code>WMQ Installation Directory/java/jre64/jre/lib/security</code>

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 `strmqikm`, 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 19 na stronie 333](#) 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 19. 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 20. *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 21. *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 22. *AcceptAllX509TrustManager.java*

Windows

Linux

AIX

Konfiguracja kanału telemetrycznego JAAS

Skonfiguruj usługę JAAS w taki sposób, aby uwierzytliła 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 [“Przykładowy plik jaas.config” na stronie 335](#).

JAAS udostępnia cztery standardowe moduły logowania. Standardowe moduły logowania NT i UNIX mają ograniczoną wartość.

Moduł JndiLogin

Uwierzytelnia się przed usługą katalogową skonfigurowaną w interfejsie JNDI (Java Naming and Directory Interface).

Krb5LoginModule

Uwierzytelnia się przy użyciu protokołów Kerberos .

NTLoginModule

Uwierzytelnia 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 telemetrycznej (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 Hasłow sekcji [“Przykładowa metoda JAASLoginModule.Login\(\)” na stronie 335](#) .

Przykładowy plik jaas . config

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";
};
```

Przykładowa metoda JAASLoginModule.Login()

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;
}

```

Zadania pokrewne

Rozwiązywanie problemu: moduł logowania JAAS nie jest wywoływany przez usługę telemetryczną

Odsyłacze pokrewne

[Klasa MQXR AuthCallback](#)

Administrowanie klientem AMQP

Klientem AMQP można administrować przy użyciu produktu IBM MQ Explorer lub w wierszu komend. Eksplorator służy do konfigurowania kanałów i monitorowania klientów AMQP, które są połączone z produktem IBM MQ. Skonfiguruj zabezpieczenia klientów AMQP przy użyciu protokołów TLS i JAAS.

Zanim zaczniesz

Informacje na temat instalowania produktu AMQP na platformie użytkownika zawiera sekcja [Wybór elementu do zainstalowania](#).

Administrowanie przy użyciu programu IBM MQ Explorer

Użyj eksploratora, aby skonfigurować kanały AMQP i monitorować klienty AMQP, które są połączone z produktem IBM MQ. Zabezpieczenia klientów AMQP można skonfigurować przy użyciu protokołów TLS i JAAS.

Administrowanie za pomocą wiersza komend

Klientem AMQP można administrować w wierszu komend [przy użyciu komend MQSC](#).

Wyświetlanie obiektów produktu IBM MQ w użyciu przez klienty AMQP

Istnieje możliwość wyświetlenia różnych zasobów produktu IBM MQ, które są używane przez klienty AMQP, 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 klienty AMQP publikują komunikaty. Serwer Hconns można wyświetlić za pomocą komendy **DISPLAY CONN**. Na przykład:

```

DISPLAY CONN(*) TYPE(CONN) WHERE (APPLDESC LK 'IBM 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 AMQP łączy się z kanałem AMQP, nowe Hconn jest połączone z menedżerem kolejek. Ten Hconn jest używany do asynchronicznego konsumowania komunikatów dla subskrypcji utworzonych przez klienta AMQP. Istnieje możliwość wyświetlenia Hconn używanego przez konkretnego klienta AMQP za pomocą komendy **DISPLAY CONN** . Na przykład:

```
DISPLAY CONN(*) TYPE(CONN) WHERE (CLIENTID EQ 'recv_abcd1234')
```

Subskrypcje utworzone przez klienty

Gdy klient AMQP subskrybuje dany temat, tworzona jest nowa subskrypcja produktu 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 używane przez konkretnego klienta AMQP, uruchom komendę **DISPLAY SUB** i przefiltruj 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 AMQP, 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(414D5143514D31202020202020202020)
TYPE(HANDLE)

OBJNAME(SYSTEM.BASE.TOPIC)          OBJTYPE(TOPIC)
```

```
OBJNAME(SYSTEM.MANAGED.DURABLE.560A7E7020002961)
OBJTYPE(QEUE)
```

3. Dla każdego z uchwytów wyświetl identyfikator klienta AMQP, który ma otwarty uchwyt:

```
DISPLAY CONN(707E0A56642B0020) CLIENTID
23 : DISPLAY CONN(707E0A56642B0020) CLIENTID

AMQ8276: Display Connection details.
CONN(707E0A56642B0020)
EXTCONN(414D5143514D31202020202020202020)
TYPE(CONN)
CLIENTID(recv_8f02c9d)
DISPLAY CONN(707E0A565F290020) CLIENTID
24 : DISPLAY CONN(707E0A565F290020) CLIENTID
AMQ8276: Display Connection details.
CONN(707E0A565F290020)
EXTCONN(414D5143514D31202020202020202020)
TYPE(CONN)
CLIENTID(recv_86d8888)
```

Identyfikacja, autoryzacja i uwierzytelnianie klienta AMQP

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 nawiązanie połączenie składa się z następujących dwóch kroków:

- Rozstrzyganie, 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 produktu IBM MQ zakłada, że aplikacja ma być sprawdzona przez uprawnienia:

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 produktu IBM MQ zakłada, że aplikacja ma być sprawdzona przez uprawnienia:

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 wzorzec tematu. Kroki określają, czy połączenie jest kontynuowane, a tożsamość produktu IBM MQ zakłada, że aplikacja ma być sprawdzona przez uprawnienia:

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.

Tożsamość klienta AMQP i autoryzacja

Aby uzyskać dostęp do obiektów produktu IBM MQ , należy użyć identyfikatora klienta AMQP, nazwy użytkownika AMQP 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 dostępnego 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
```

Uwierzytelnianie klienta AMQP przy użyciu hasła

Uwierzytelnij nazwę użytkownika klienta AMQP 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.

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 przy użyciu specyfikacji CipherSpec wymiany kluczy Diffie-Hellmana, 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 CipherSpec. Jeśli żądanie klienta dotyczące połączenia TLS zostanie zaakceptowane przez kanał AMQP, kanał kluczy musi być chroniony hasłem.

Domyślnie, ponieważ implementacje protokołu TLS nie używają anonimowych specyfikacji szyfrowania, magazyn kluczy musi zawierać podpisany przez klienta certyfikat, który może być uwierzytelniany przez klienta.

Jeśli używane są anonimowe specyfikacje CipherSpec, 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.

Konfigurowanie klientów AMQP za pomocą protokołu TLS

Klienci AMQP można skonfigurować w taki sposób, aby używały protokołu TLS w celu ochrony danych przepływających przez sieć, a także do uwierzytelniania tożsamości menedżera kolejek, z którym łączy się klient.

Aby użyć protokołu TLS dla połączenia z klientem AMQP 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 od klienta AMQP przedstawienia certyfikatu 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 AMQP łączy się z kanałem AMQP zabezpieczonym przez protokół TLS, klient weryfikuje tożsamość certyfikatu prezentowanego przez menedżer kolejek. W tym celu należy skonfigurować klienta AMQP przy użyciu magazynu zaufanych certyfikatów zawierającego certyfikat menedżera kolejek. Kroki do wykonania różnią się w zależności od używanego klienta AMQP. Więcej informacji na temat różnych klientów i interfejsów API AMQP można znaleźć w dokumentacji klienta AMQP.

Rozłączanie klientów AMQP z menedżera kolejek

Jeśli chcesz odłączyć klientów AMQP od menedżera kolejek, uruchom komendę PURGE CHANNEL lub zatrzymaj połączenie z klientem AMQP.

- Uruchom komendę **PURGE CHANNEL**. Na przykład:

```
PURGE CHANNEL(MYAMQP) CLIENTID('recv_28dbb7e')
```

- Można również zatrzymać połączenie, które jest używane przez klienta AMQP 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 komunikacie 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 343.

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 grupowego adresu IP sięga od 224.0.0.0 do 239.255.255.255, jednak niektóre z tych adresów są zarezerwowane. Aby uzyskać listę zastrzeżonych 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.

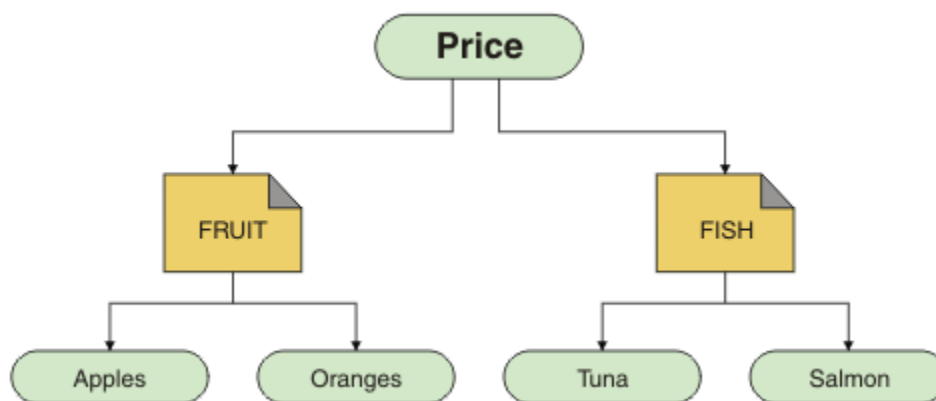
Na poniższym diagramie istnieją dwa możliwe strumienie danych rozsyłania grupowego:

```
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 deskrytorze 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 odbierająca zastosuje wartość domyślną. Domyślne wartości MQMD nie muszą być takie same, jak wartość MQMD_DEFAULT, i są opisane w Tabeli 19 na stronie 345.

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 .

V 9.2.0 **V 9.2.0** Funkcja przesyłania komunikatów opartej na rozsyłaniu grupowym XMS .NET (przy użyciu programu RMM) stała się nieaktualna od wersji IBM MQ 9.2 i zostanie usunięta w przyszłej wersji programu XMS .NET.

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 (opisanej wcześniej w tej sekcji), a także wartości domyślnej używanej, gdy atrybut nie jest wysyłany.

Atrybut	Działanie podczas korzystania z rozsyłania grupowego	Domyślna, jeśli nie przestano
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
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

Tabela 19. Atrybuty przesyłania komunikatów i sposób ich powiązania z rozsyłaniem grupowym (kontynuacja)

Atrybut	Działanie podczas korzystania z rozsyłania grupowego	Domyślna, jeśli nie przesłano
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](#)

[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 wysłanych 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 potwierżeń).

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.

Odsyłacze 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 włączony **MCAST**, a opcja **BRIDGE** jest włączona, menedżer kolejek przesyła komunikat z 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 opcja **BRIDGE** jest ustawiona na wartość **ENABLED**, a parametr **MCAST** tematu ma również wartość **ENABLED**, publikacje z aplikacji, które nie korzystają z rozsyłania grupowego, są połączone mostem do aplikacji, 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 odbiorcę lub odbiorcy 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:

Tabela 20. Właściwości komunikatu produktu IBM MQ do odwzorowań właściwości LLM produktu IBM MQ

IBM MQ właściwość	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
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](#).

administrowanieIBM 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 352](#)
- [“Alternatywne sposoby administrowania produktem IBM MQ for IBM i” na stronie 366](#)
- [“Zarządzanie pracą dla produktu IBM i” na stronie 371](#)

Pojęcia pokrewne

[“Dostępność, tworzenie kopii zapasowych, odtwarzanie i restartowanie w systemie IBM i” na stronie 378](#)
 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.

[Informacje o nazwach bibliotek menedżera kolejek produktu IBM MQ for IBM i](#)

Zadania pokrewne

[Zmiana informacji konfiguracyjnych w systemie IBM i](#)

[Konfigurowanie zabezpieczeń w systemie IBM i](#)

Odsyłacze pokrewne

[“WYCISZANIE IBM MQ for IBM i” na stronie 423](#)

W tej sekcji opisano sposób wyciszania (na koniec z wdziękiem) IBM MQ for IBM i.

[“Procedura obsługi kolejki niedostarczonych komunikatów w systemie IBM i” na stronie 153](#)

Co to jest procedura obsługi kolejki niedostarczonych komunikatów programu IBM i i jak ją wywołać?

[Określanie problemów z aplikacjami produktu IBM MQ for IBM i](#)

[Instalowalne usługi i komponenty w systemie IBM i](#)

[Obiekty systemowe i domyślne](#)

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 [“Administrowanie za pomocą komend MQSC” na stronie 12](#) i [“Korzystanie z formatów komend programowalnych IBM MQ” na stronie 26](#).

Odsyłacze 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 423.

Tworzenie obiektów produktu IBM MQ for IBM i

Te informacje umożliwiają zapoznanie się z metodami tworzenia obiektów IBM MQ dla produktu IBM i.

Zanim rozpocznie

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 odpowiedzi. Aby wyświetlić panel odpowiedzi 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 WRKMQM

Procedura

1. W wierszu komend wpisz WRKMQM .
2. Wprowadź nazwę menedżera kolejek.
3. Jeśli chcesz wyświetlić panel odpowiedzi, naciśnij klawisz F4. Panel odpowiedzi 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 354 . 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.

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)
MAXMSGLEN(2000)
USAGE(*NORMAL)
```

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 356](#).

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 zawiera sekcja [“Procedura obsługi kolejki niedostarczonych komunikatów w systemie IBM i”](#) na stronie 153.

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 355 zdefiniowano 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:

```
CLRMQM 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 `DLTMQM`.

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.

Informacje o produkcie 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 poziomego panelu komend `WRKMQM`.

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(MYQUEUEEMANAGER)
```

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(MYQUEUEEMANAGER)
```

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(MYQUEUEEMANAGER)

/* 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(MYQUEUEEMANAGER)
```

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(MYQUEUEEMANAGER)

/* 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(MYQUEUEEMANAGER)
```

IBM i

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. */
CHGMQMQ MQMNAME(MYQUEUEMANAGER) QNAME('blue.model.queue') PUTENBL(*YES)
```



Praca z wyzwaniem w systemie IBM i

Ta sekcja zawiera informacje na temat wyzwania 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 *wyzwaniem* i jest szczegółowo opisany w sekcji [Kanały wyzwaniane](#).

Co to jest wyzwanie?

Menedżer kolejek definiuje pewne warunki, które tworzą zdarzenia wyzwajające. Jeśli wyzwanie jest włączone dla kolejki i wystąpi zdarzenie wyzwajające, menedżer kolejek wysyła komunikat wyzwajacza do kolejki o nazwie kolejka inicjujący. Obecność komunikatu wyzwajacza w kolejce inicjuj. wskazuje, że wystąpiło zdarzenie wyzwajające.

Komunikaty wyzwajacza 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 przestaniania 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)
MAXMSGL(2000)
MAXDEPTH(1000)
```

Tworzenie definicji procesu

Aby utworzyć definicję procesu, należy użyć komendy **CRTMQMPCRC**. 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.quote.queue`. Następująca komenda tworzy wymagany proces, `motor.insurance.quote.process`, zidentyfikowany w tym przykładzie:

```
CRTMQMPCRC MQMNAME(MYQUEUEMANAGER) PRCNAME('motor.insurance.quote.process')
TEXT('Insurance request message processing')
APPCTYPE(*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 **DSPMQMPCRC**. Może to pomóc w określeniu, co proces ma. Jeśli w łańcuchu używane są spacje, łańcuch należy ująć w pojedynczy cudzysłów.

APPCTYPE(*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 **DSPMQMPCRC** . Na przykład:

```
MQMNAME(MYQUEUEMANAGER) DSPMQMPCRC('motor.insurance.quote.process')
```

Można również użyć komendy **CHGMQMPRC** , aby zmienić istniejącą definicję procesu, a następnie komendę **DLTMQMPRC** , aby usunąć definicję procesu.

IBM i

Komunikacja między dwoma systemami IBM MQ w 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') UDLMSGQ(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).
```

IBM i Przykładowe definicje zasobów w systemie IBM i

Ten przykład zawiera przykładowy program CL AMQSAMP4 IBM i.

```

/*****/
/*
/* 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, 2024. 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.
/*
/*****/
SNDPGMMSG 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 352.

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 :

IBM i Administrowanie lokalne i zdalne w systemie 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. Patrz sekcja [“Administrowanie przy użyciu komend PCF w systemie IBM i”](#) na stronie 368.
2. Pisanie programu za pomocą interfejsu MQAI, który wysyła komunikaty PCF. Patrz sekcja [“Korzystanie z interfejsu MQAI w celu uproszczenia korzystania z systemów PCF”](#) na stronie 37.
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 368.
4. Komenda **STRMQMQSC** służy do wysyłania komend pośrednio do zdalnego menedżera kolejek. Patrz [“Administrowanie za pomocą komend MQSC w systemie IBM i”](#) na stronie 367.

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 nasłuchiwania, 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:

```
CRTRCPDF 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 23 na stronie 367](#).

Aby zakończyć taką sesję MQSC, wpisz end.

Rysunek 23 na stronie 367 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 23. 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 one rozróżniane wielkością 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 MQSC](#).

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. Następnie aplikacja może przetworzyć odpowiedź i działać odpowiednio.

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 37.

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 (platformax86) i produkt IBM MQ for Linux (platformyx86 i x86-64) udostępniają interfejs administracyjny o nazwie IBM MQ Explorer, który umożliwia wykonywanie zadań administracyjnych jako alternatywa dla komend CL, kontrolnych 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 nasłuchiwania, 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 **STRMQMSVR**.
2. Odpowiedni program nasłuchujący TCP/IP istnieje dla każdego menedżera kolejek zdalnych. Jest to program nasłuchujący IBM MQ uruchomiony za pomocą komendy **STRMQMSR** .
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:

```
STRMQMSVR 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.

```
ENDMQMSVR 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.

```
ENDMQMSVR 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 kolejkowane 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:

```
DSPMQMSVR MQMNAME('saturn.queue.manager')
```

Wydaj tę komendę na komputerze docelowym. Jeśli serwer komend jest uruchomiony, zostanie wyświetlony panel wyświetlany w programie [Rysunek 24 na stronie 370](#) :

```
Display MQM Command Server (DSPMQMSVR)
```

```
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 24. Wyświetl panel serwera komend MQM

Należy skonfigurować środowisko zgodnie z opisem w poniższym tekście, aby komendy Qshell związane z konsolą WWW przebiegły poprawnie w systemie IBM MQ for IBM i.

O tym zadaniu

Po uruchomieniu Qshell inicjuje wewnętrzne tabele służące do przetwarzania komend w oparciu o CCSID zadania. Aby komendy Qshell związane z konsolą WWW przebiegły poprawnie w systemie IBM i, należy skonfigurować środowisko.

Ustawienia narodowe są ustawiane przez ustawienie zmiennej środowiskowej LANG na nazwę ścieżki do obiektu ustawień narodowych. Na przykład, aby ustawić ustawienia narodowe dla języka angielskiego (Stany Zjednoczone), zmienna środowiskowa LANG jest ustawiana w następujący sposób:

```
LANG=/QSYS.LIB/EN_US.LOCALE
```

W powłoce Qshell można sprawdzić ustawienie przy użyciu zestawu komend, aby wyświetlić listę wszystkich zmiennych środowiskowych. Zwykle jest to LANG, które może mieć wpływ na ustawienia narodowe dla środowiska wykonawczego. Może mieć również wartość LC_ALL.

Aby poprawnie uruchomić komendy Qshell, ustawienie środowiska ustawień narodowych musi być spójne z ustawieniem zadania.

Procedura

Należy użyć komendy CL DSPJOB JOB (JobNumber/USERProfile/JobName).

- a) Wybierz opcję 2, aby wyświetlić atrybuty definicji zadania.
- b) Następujące atrybuty powinny być spójne z ustawieniem środowiska LANG lub LC_ALL
 - Identyfikator języka
 - Identyfikator kraju lub rejonu
 - Identyfikator kodowanego zestawu znaków

Na przykład, jeśli

```
LANG=/QSYS.LIB/FR_FR.LOCALE
```

Atrybuty zadania powinny być następujące:

- ID języka..... FRA
- ID kraju lub rejonu..... FR
- Identyfikator kodowanego zestawu znaków... 297

Co dalej

For more information on national language support, see the IBM Documentation topic [Uwagi dotyczące obsługi języków narodowych \(NLS\)](#).

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 21 na stronie 372](#).

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 21. IBM MQ Zadania.</i>	
Nazwa zadania	Funkcja
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.

Tabela 21. IBM MQ Zadania. (kontynuacja)	
Nazwa zadania	Funkcja
AMQZMUR0	Menedżer programu narzędziowego. To zadanie wykonuje newralgiczne programy narzędziowe menedżera kolejek, na przykład menedżer łańcucha kroniki.
AMQFQPUB	Demon publikowania/subskrybowania w kolejce.
AMQFCXBA	Zadanie robocze brokera.
RUNMQBRK	Zadanie sterujące brokerem.
AMQRMPPA	Zadanie zestawiania procesów kanału.
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 22 na stronie 373](#), a obiekty utworzone dla menedżera kolejek są wymienione w sekcji [Tabela 23 na stronie 374](#).

Uwaga: Obiekty zarządzania pracą można znaleźć w bibliotece QMQM, a obiekty menedżera kolejek można znaleźć w bibliotece menedżera kolejek.

Tabela 22. Obiekty zarządzania pracą		
Nazwa	Typ	Opis
AMQZLAA0	*JOB	Opis zadania, który jest używany przez procesy agenta IBM MQ
AMQZLSA0	*JOB	Agent menedżera kolejek izolowanych powiązań

Tabela 22. Obiekty zarządzania pracą (kontynuacja)		
Nazwa	Typ	Opis
AMQZXMAO	*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
QMQMMSG	*MSGQ	Domyślna kolejka komunikatów dla zadań programu IBM MQ.
QMQMRUN20	*CLS	Opis klasy dla zadań o wysokim priorytecie IBM MQ
QMQMRUN35	*CLS	Opis klasy dla zadań o średnim priorytecie IBM MQ
QMQMRUN50	*CLS	Opis klasy dla zadań o niskim priorytecie IBM MQ

Tabela 23. Obiekty zarządzania pracą utworzone dla menedżera kolejek		
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
QMQMMSG	*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 udostępni przykłady konfiguracji.



Ostrzeżenie: Nie zmieniaj ustawień pozycji kolejki zadań w podsystemie QMQM, 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 nazwie takiej samej, 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 jedną z pozycji routingu zdefiniowanych w podsystemie QMQM, a to definiuje, które z dostarczonych klas (QMQMRUN20, QMQMRUN35 lub QMQMRUN50) są używane przez zadanie.

Uwaga: Jeśli zadania IBM MQ nie są 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 21 na stronie 372](#) można utworzyć opis zadania, korzystając z następującego arkusza jako przykładu:

```
What is the queue manager library name? _____
Does job description AMQZXA0 exist in the queue manager library? Yes No
Does job description QMQMJOB exist in the queue manager library? Yes No
Does job description AMQZXA0 exist in the QMQM library? Yes No
Does job description QMQMJOB 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.

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 QMQMJOBd.
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.

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/QMQMJOBd z nazwami procesów produktu IBM MQ , które mają być sterowane w bibliotece menedżera kolejek. Na przykład:

```
CRTDUPOBJ OBJ(QMQMJOBd) FROMLIB(QMQM) OBJTYPE(*JOBd) TOLIB(QMTESTQM)
NEWOBJ (RUNMQCHI)
CRTDUPOBJ OBJ(QMQMJOBd) FROMLIB(QMQM) OBJTYPE(*JOBd) TOLIB(QMTESTQM)
NEWOBJ (AMQRRMFA)
```

- b. Zmień parametr danych routingu w opisie zadania, aby upewnić się, że zadania korzystają z klasy QMQMRUN20 .

```
CHGJOBd JOBd(QMTESTQM/RUNMQCHI) RTGDTA('QMQMRUN20')
CHGJOBd 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 SBS(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:

```
CHGJOB JOB(QMQMlibname/QMQMJOB) RTGDTA('QMQMRUN10')
```

Teraz wszystkie zadania menedżera kolejek powiązane z biblioteką QMlibraryname 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/QMQMJOB w bibliotece menedżera kolejek za pomocą komendy

```
CRTDUPOBJ OBJ(QMQMJOB) FROMLIB(QMQM) OBJTYPE(*JOB) TOLIB(QMTESTQM)
```

- b. Zmień parametr kolejki zadań w opisie zadania, aby upewnić się, że zadania korzystają z kolejki zadań QBATCH.

```
CHGJOB JOB(QMTESTQM/QMQMJOB) 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 SBS. Użyj komendy DSPSBS 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 SBSB(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 SBSB(QMQM/QMQM2) JOBQ(QMQM/QMQM2) MAXACT(*NOMAX)
```

- e. Utwórz duplikat QMQMJOBd w bibliotece menedżera kolejek, wydając następującą komendę:

```
CRTDUPOBJ OBJ(QMQMJOBd) FROMLIB(QMQM) OBJTYPE(*JOBd) TOLIB(QMlibraryname)
```

- f. Zmień opis zadania, aby użyć nowej kolejki zadań, wydając następującą komendę:

```
CHGJOBd JOBd(QMlibraryname/QMQMJOBd) JOBQ(QMQM/QMQM2)
```

- g. Uruchom podsystem, wydając następującą komendę:

```
STRSBS SBSB(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.

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 wspomagania 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.

0

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 384).
 - 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 mqs.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 384 .

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) DSPJRNDTA(*YES)
```

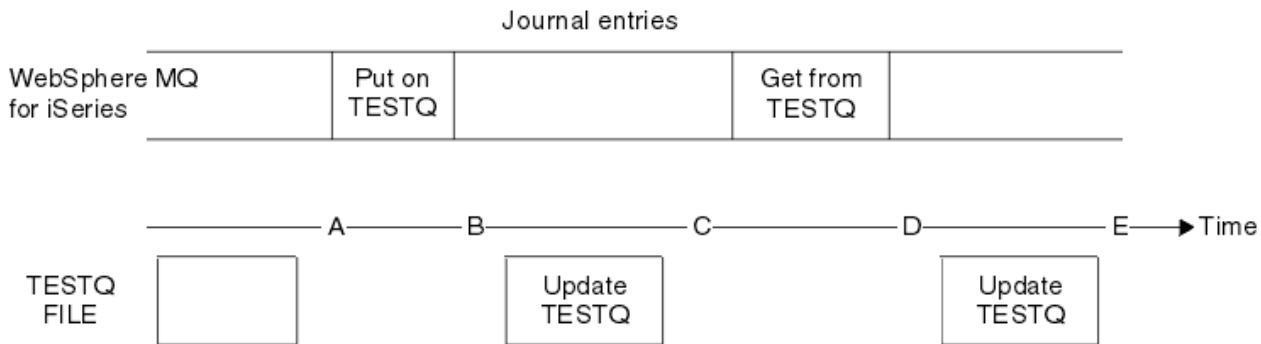
W miarę jak kolejki obsługują kolejne komunikaty, rekord punktu kontrolnego staje się niespójny z bieżącym stanem kolejek.

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 25 na stronie 381](#).



Rysunek 25. 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 378.

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 384).

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.

Wątek punktu kontrolnego w procesie AMQZMUC0 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

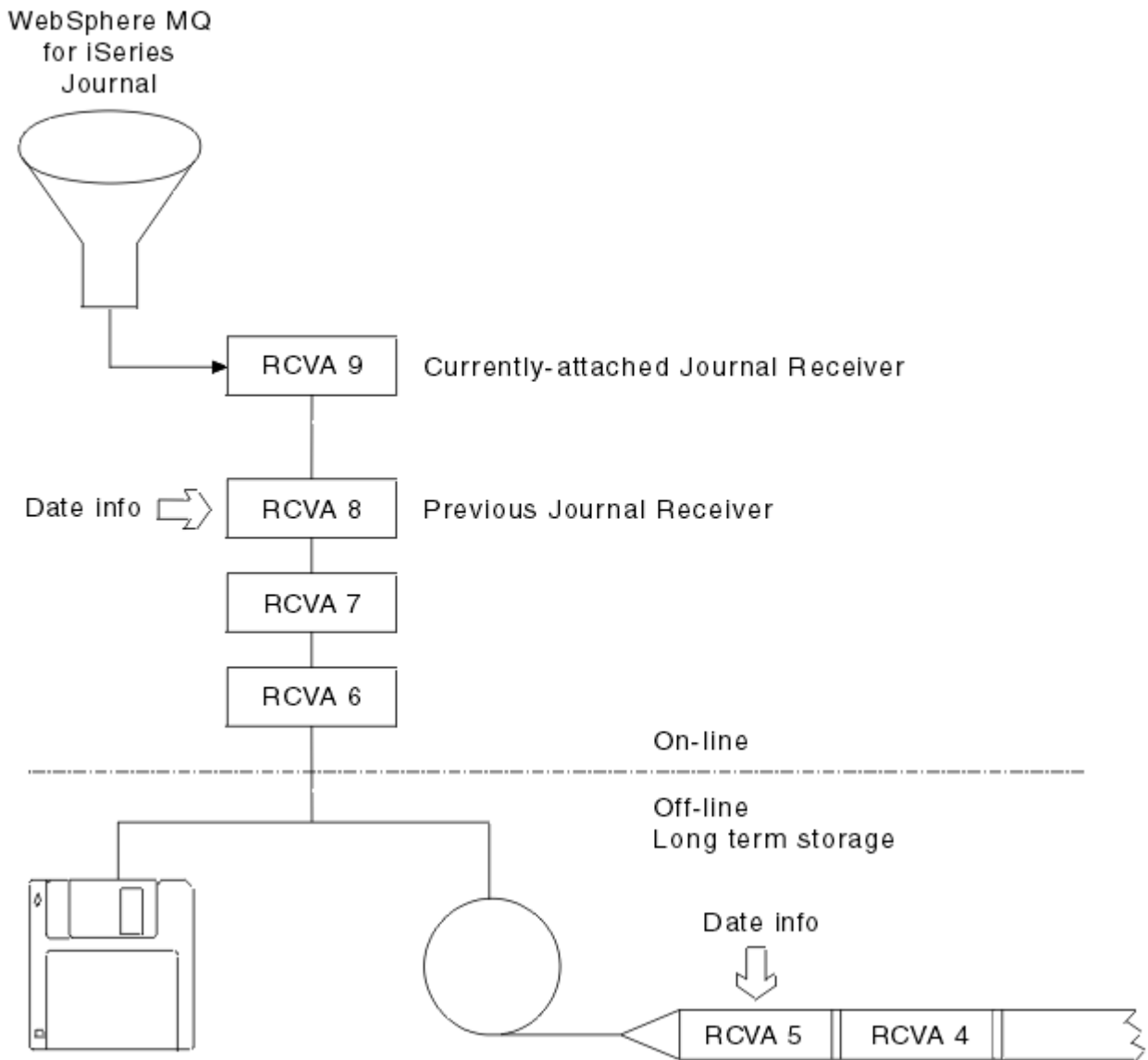
W ramach strategii składowania, należy zadbać o swoje dzienniki. Z różnych powodów użyteczne jest usuwanie dzienników z bibliotek produktu IBM MQ :

- 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 zeskładanych, pod warunkiem, że są one dostępne do odtworzenia, jeśli jest to konieczne dla operacji odtwarzania.

Pojęcie zarządzania kronikami jest przedstawione w sekcji [Rysunek 26 na stronie 385](#).



Rysunek 26. 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 zeskalowany, 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:

JJJJJJJJJ

Jest najstarszą nazwą dziennika, której nadal potrzebuje produkt IBM MQ .

LLLLLLLLL

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(*JRNRCV) .....
```

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 379):

```
CRTJRNRCV JRNRCV(QMQLIB/AMQA900001) THRESHOLD(XXXXX)
```

2. Ręcznie utwórz kronikę z dziennikiem:

```
CRTJRN JRN(QMGLIB/AMQAJRN) MNGRCV(*SYSTEM) +  
JRNRCV(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 rezerwowany, 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.

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](#). System plików Network File System 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 sieciowej. Aby odtworzyć dzienniki menedżera kolejek do stanu spójnego w punkcie awarii, należy przestać 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 informacjach o produkcie IBM i w produkcie IBM Documentation.

Aplikacje

Aby zbudować klient w celu automatycznego ponownego nawiązania połączenia z menedżerem kolejek, gdy rezerwowy 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](#) .

 *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 404](#) 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 408](#).

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 GID i UID, co eliminuje konieczność zmiany identyfikatorów UID i GID 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 395
2. Niezależna pula pamięci dyskowej (IASP), do której przenoszony jest menedżer kolejek; patrz [“Niezależne pule pamięci dyskowej \(IASPs\)”](#) na stronie 395
3. Grupa zasobów klastra (cluster resource group-CRG); patrz [“Grupy zasobów klastra urządzeń”](#) na stronie 395, w której definiuje się:
 - Domena odzyskiwania zasobów
 - IASP
 - Program obsługi wyjścia; patrz [“Program obsługi wyjścia CRG urządzenia”](#) na stronie 396

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.

Przykładowy program obsługi wyjścia AMQSCRG4 jest dostępny w produkcie IBM i z produktu IBM MQ 9.1.

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 397 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 396 zawiera przykładowy kod pomocny podczas wykonywania tej operacji.

Pojęcia pokrewne

“Niezależne ASP i wysoka dostępność” na stronie 417

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 przejęcia do wszystkich węzłów za pomocą komendy:

```
ADDTCPICF INTNETADR(' [TAKEOVER IP]') LIND([LINE DESC])
SUBNETMASK(' [SUBNET MASK]') AUTOSTART(*NO)
```

7. Uruchom adres IP przejęcia tylko w węźle podstawowym, korzystając z komendy:


```
STRTCPICF INTNETADR(' [TAKEOVER IP]')
```

8. Opcjonalne: Jeśli pula IASP jest przetączalna, 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(([IAPS NAME] *DEVD *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(([IAPS NAME] *DEVD *ONLINE ' [TAKEOVER IP]'))
```

 *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 trybie `varry on` 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 z 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:
- VRYCFG CFGOBJ ([IASP NAME]) CFGTYPE (*DEV) STATUS (*ON)
 - SETASGRP [NAZWA PULI IASP]
 - ENDMQM [NAZWA MENEDŻERA]
 - 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 wykonania aktywnej instancji lub nie przetwarzać komunikatów, które zostały odebrane przed nieudanym niepowodzeniem.

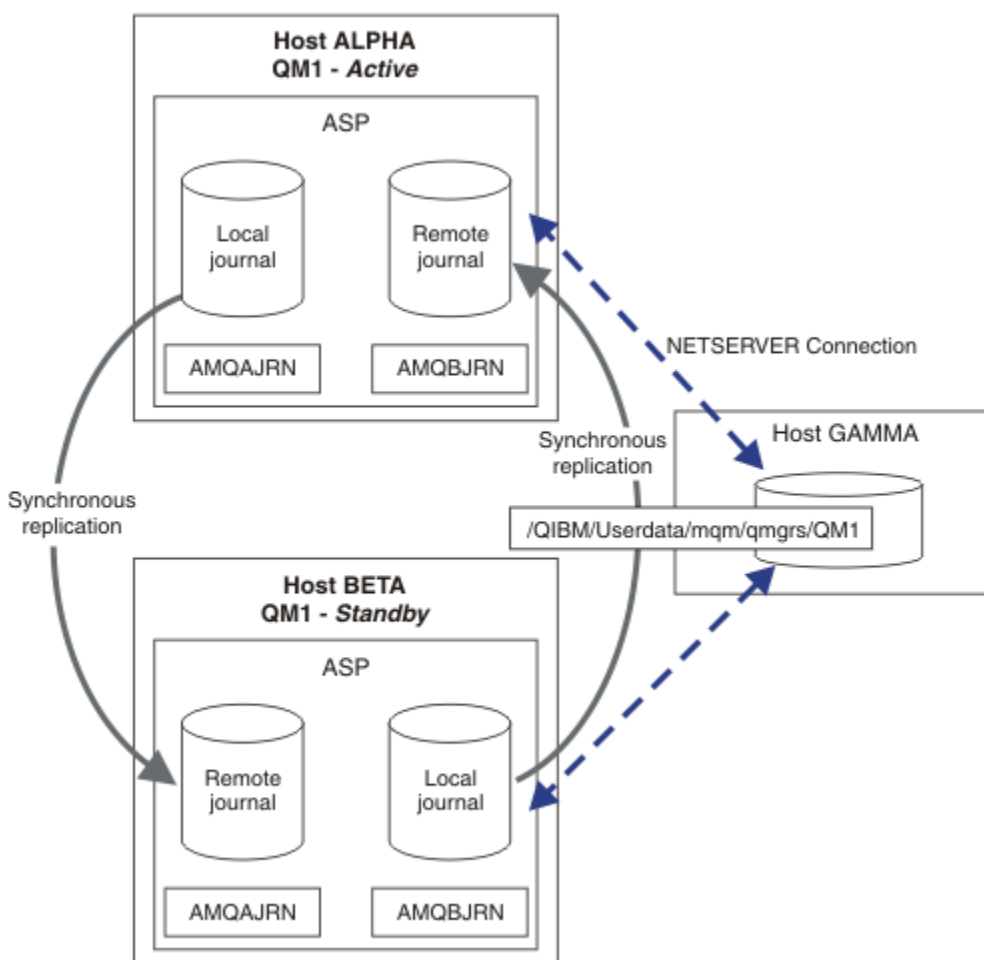
Inną, rzadką możliwością jest to, że system plików zawiera najnowszy rekord punktu kontrolnego, a niezsynchronizowana kronika zdalna w systemie rezerwowym nie jest dostępna. W takim przypadku menedżer kolejek nie jest restartowany automatycznie. Istnieje możliwość oczekiwania na zsynchronizowanie kroniki 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. Rezerwowy oczekuje na uzyskanie blokady, a po jej udostępnieniu staje się instancją aktywną. Blokada jest zwolniona 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 27 na stronie 400, jeśli działanie ALPHA nie powiedzie się, można ręcznie zrestartować produkt QM1 w systemie BETA, korzystając z kroniki kopii lustrzanej. Dodanie funkcji menedżera kolejek z wieloma instancjami do produktu QM1 powoduje, że instancja rezerwowa produktu QM1 automatycznie wznowia 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 27 na stronie 400 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 uwzględniał 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 379. Obecnie liczba działających instancji menedżera kolejek jest ograniczona do dwóch, jeden jest aktywny, a jeden jest w stanie gotowości.



Rysunek 27. 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 27 na stronie 400 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 wychwytywać dziennik lokalny. Jeśli zostanie wybrana opcja *SYNC, system lokalny oczekuje na kronikę zdalną przed powrotem z wywołania, które wymaga wykonania zakończonego zapisu. Dzienniki lokalne i zdalne zasadniczo 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 / *ASYN zilustrowany w programie Rysunek 27 na stronie 400 jest sposobem, w jaki instancja rezerwowego 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 powiedzie 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 *SYNC lub *ASYN jest wyłączonym transakowaniem. Tabela 24 na stronie 401 podsumowuje kompromisy między użyciem kronikowania *SYNC i *ASYN między parą menedżerów kolejek:

<i>Tabela 24. Opcje zdalnego kronikowania</i>			
Aktywny	W trybie gotowości	*SYNC	*ASYN
*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
*ASYN		<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

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 .

***SYNC / *SYNC**

Aktywna instancja menedżera kolejek używa kronikowania *SYNC i gdy instancja rezerwowego menedżera kolejek zostanie uruchomiona, natychmiast próbuje użyć kronikowania *SYNC .

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 zdalne kronikowanie 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 zdalne kronikowanie 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 wychwyt 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 zdalne kronikowanie 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, podczas 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 rezerwową 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 rezerwowego menedżera 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 rozpoczęta świeża dziennik, a transakcje niezapisane w danych menedżera kolejek w systemie plików zostaną utracone.

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 **ADDQMQRN** 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. produkt musi być co najmniej w IBM WebSphere MQ 7.0.1 Fix Pack 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 28 na stronie 407](#). 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 **ADDMQMJRN** 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 391.

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.

```
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.

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.

```
ADDMQMINF 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.

```
ADDMQMJRN 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 406](#).

```
ADDMQMJRN MQMNAME (QM1) RMTJNRDB (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 399](#) 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 zdalnego dziennika.

8. 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)
```

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.

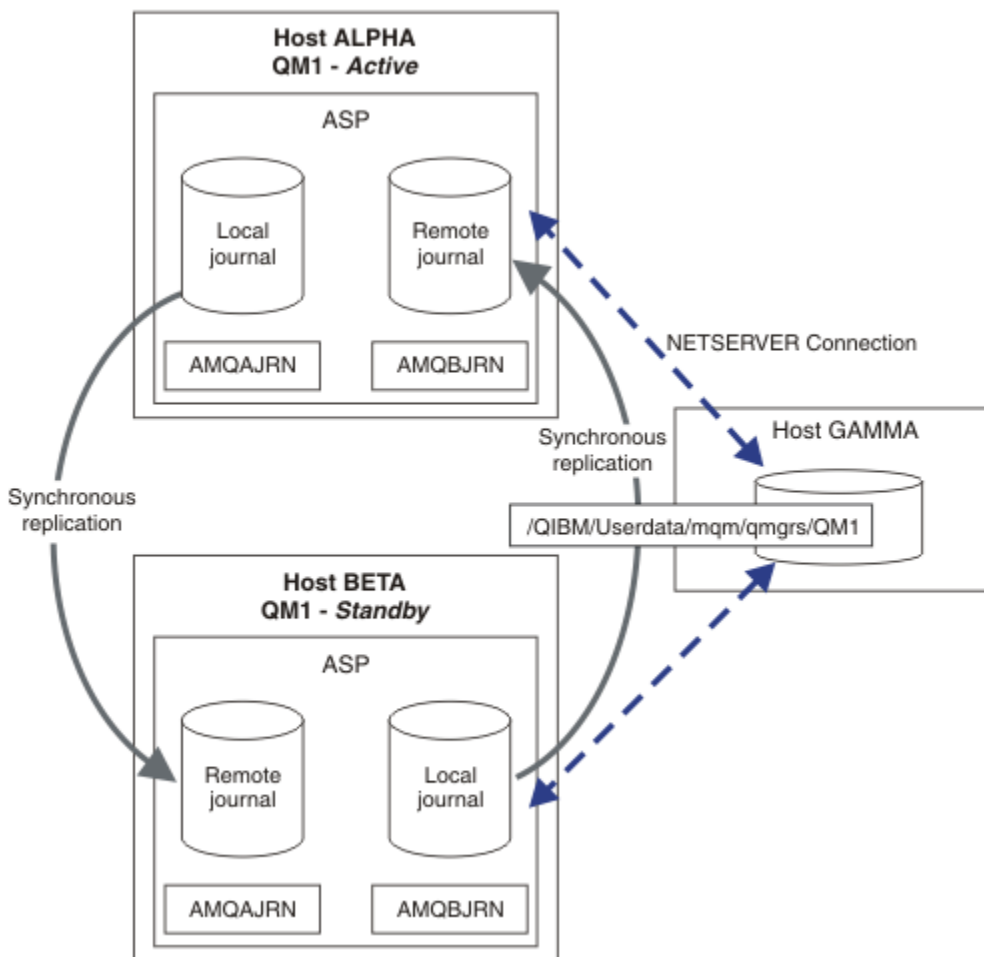
```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

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 28. 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 29 na stronie 412](#). 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 409](#) 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 409. Odtwórz bibliotekę menedżera kolejek po kroku “5” na stronie 409.

Uwaga:

W kroku “5” na stronie 409 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 391.

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.

```
QSAVEQMGR LCLQMGRNAM(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 406.

```
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 399 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 zdalnego dziennika.

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.

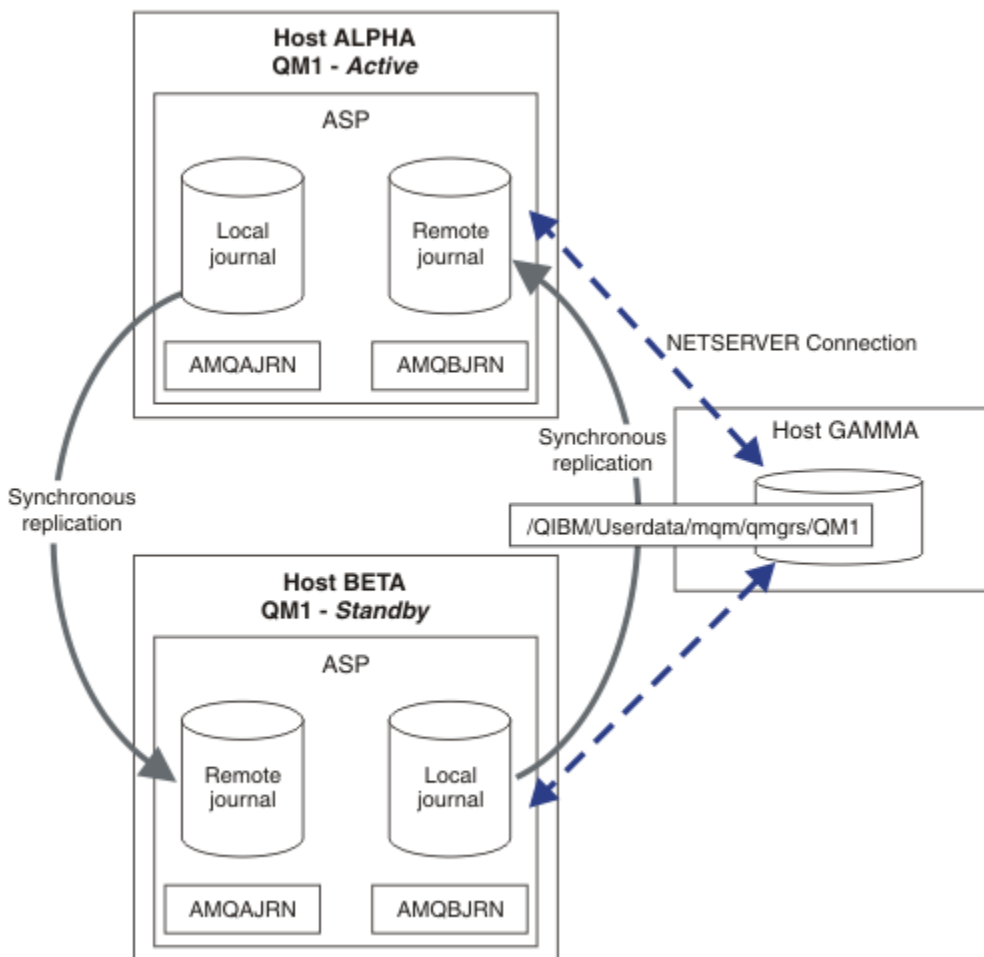
```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

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 29. 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:

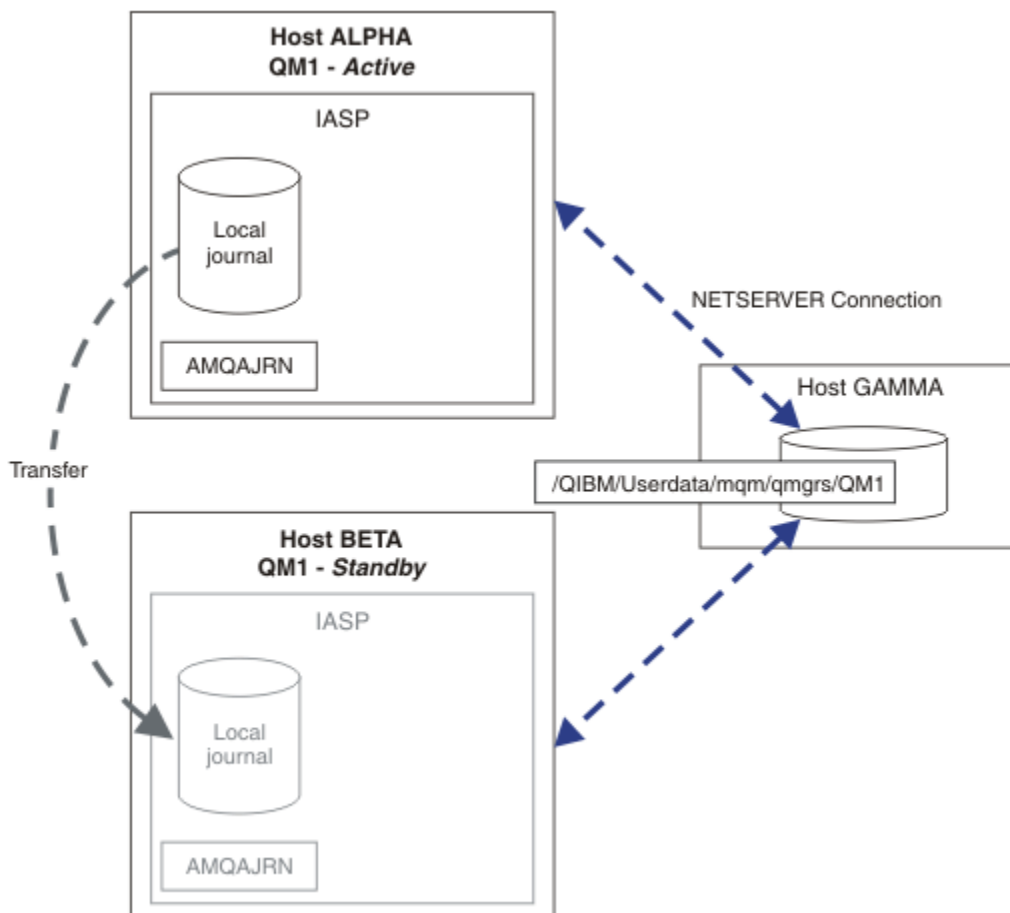
1. Użyj opcji NetServer, aby umieścić dane menedżera kolejek na serwerze Windows.

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 **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 przetącznika, lub można napisać procedurę komendy, aby programowo przesłać niezależną ASP.



Rysunek 30. 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 390.

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 417.

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.

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. produkt musi być co najmniej w IBM WebSphere MQ 7.0.1 Fix Pack 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 31 na stronie 416](#). 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 391.

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=QMQM1
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(QMQM1)
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.

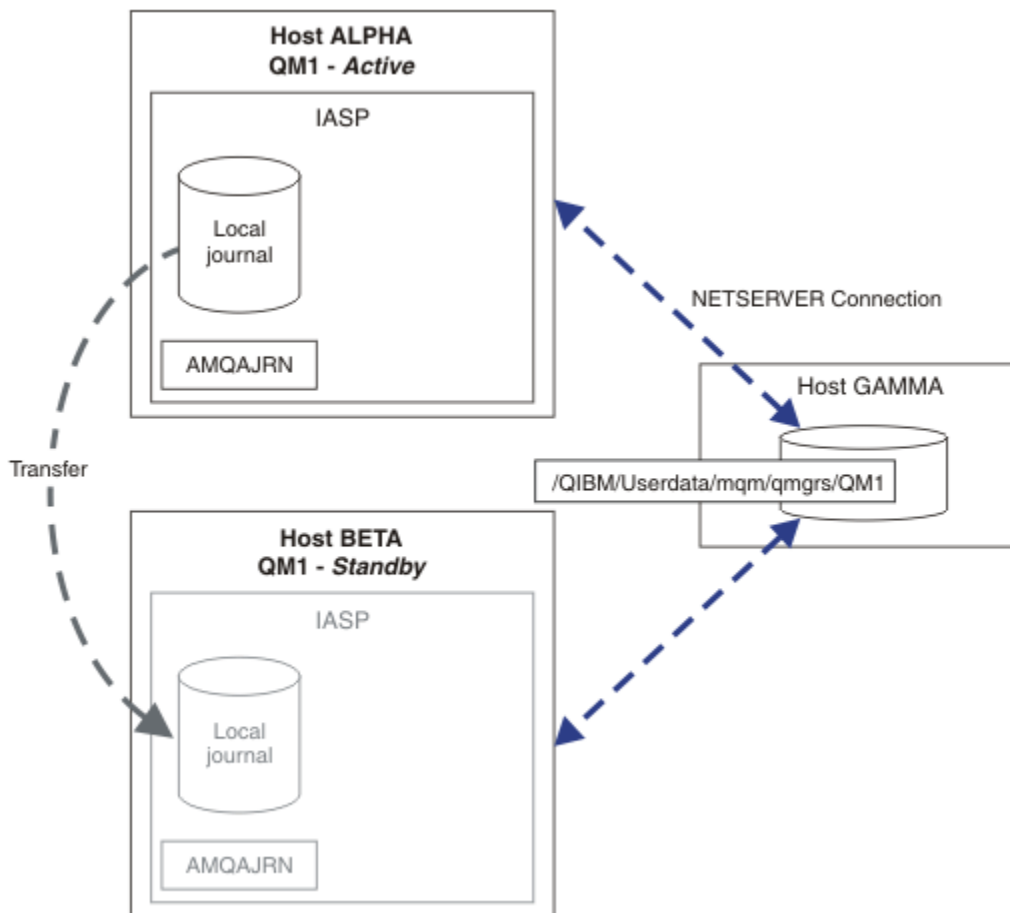
```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

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 31. 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.

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.

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 404.](#)
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 420 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 rozpocznie

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 404. 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](#) w dokumentacji produktu IBM i .

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 424) lub wszystkich menedżerów kolejek (patrz [“Zamykanie wszystkich menedżerów kolejek w systemie IBM MQ for IBM i”](#) na stronie 425).
5. Zamknij serwer mqweb, wprowadzając następującą komendę w powłoce qshell:

```
/QIBM/ProdData/mqm/bin/endmqweb
```

ENDMQM, parametr ENDCCTJOB (*YES)

Parametr ENDMQM ENDCCTJOB (*YES) działa inaczej w programie IBM MQ for IBM i V6.0 , 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 V6.0 lub nowszym, jeśli określono 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 zapobiegną 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.

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 równolegle.

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 równolegle.

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 równolegle.

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 równolegle.

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(QMg11)
```

Powtórz tę czynność dla każdego menedżera kolejek, który ma zostać uruchomiony.

Jeśli jakikolwiek 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

administrowanie IBM 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/OS można użyć komend IBM MQ Script (MQSC) lub Programmable Command Formats (PCFs). Informacje na temat używania komend dla produktu IBM MQ for z/OS zawiera sekcja [“Wysyłanie komend do programu IBM MQ for z/OS”](#) na stronie 428.

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 436.

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

[IBM MQ for z/OS pojęcia](#)

Zadania pokrewne

[“Praca z lokalnymi obiektami IBM MQ”](#) na stronie 126

Istnieje możliwość administrowania lokalnymi obiektami produktu IBM MQ w celu obsługi aplikacji, które używają interfejsu kolejki komunikatów (Message Queue Interface-MQI).

[“Praca ze zdalnymi obiektami IBM MQ”](#) na stronie 191

Obiekty IBM MQ można administrować w zdalnych menedżerach kolejek za pomocą komend MQSC, komend PCF lub administrative REST API. Zanim będzie można użyć dowolnej z tych metod, należy zdefiniować kolejki transmisji i kanały między lokalnym menedżerem kolejek a zdalnym menedżerem kolejek, aby komendy mogły być wysyłane do zdalnego menedżera kolejek, a odpowiedzi otrzymane przez lokalny menedżer kolejek. Alternatywnie można skonfigurować klaster menedżera kolejek, a następnie użyć tych samych metod zdalnego administrowania.

[“administrowanie IBM MQ”](#) na stronie 7

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.

[Planowanie](#)

[Planowanie środowiska IBM MQ w systemie z/OS](#)

[Konfigurowanie](#)

[Konfigurowanie produktu z/OS](#)

[Korzystanie z programów narzędziowych IBM MQ for z/OS](#)

Odsyłacze pokrewne

[“Źródła, z których można wydawać komendy MQSC w systemie z/OS” na stronie 21](#)

Komendy MQSC mogą być wydawane z różnych źródeł, w zależności od komendy.

[Skorowidz formatów komend programowalnych](#)

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).

Gdy używana jest konsola z/OS, należy dodać /cpf do początku komendy, gdzie cpf jest przedrostkiem komendy dla podsystemu menedżera kolejek.

- Wejściowe zestawy danych inicjowania.
- Dostarczony program narzędziowy do przetwarzania wsadowego, CSQUTIL, przetwarzający listę komend w sekwencyjnym zestawie danych.

W przypadku korzystania z wejściowych zestawów danych inicjowania lub dostarczonego programu narzędziowego wsadowego nie trzeba dodawać /cpf do początku komendy.

- 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 26 na stronie 432 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](#).

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 na niej komunikaty, odświeżenie, które usuwa tę kolejkę, nie powiedzie się.
- Jeśli kopia kolejki wymaga wykonania instrukcji ALTER z FORCE, aby ją zmienić.

W tych okolicznościach odświeżanie nie jest wykonywane dla 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 jako autonomiczne oraz jako element grupy współużytkowania kolejki.

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.

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 równolegle 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. Rozporządzanie 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 rozporządzania 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 25 na stronie 431 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 25. Podsumowanie głównych komend MQSC i PCF według typu obiektu</i>				
Komenda MQSC	Zmień	Definiowanie	WYŚWIETL	DELETE
Komenda PCF	Zmień	Utwórz/Kopuj	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
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 25 na stronie 431](#).

Tabela 26 na stronie 432 wyświetla każdą komendę MQSC i w przypadku, gdy każda komenda może zostać wydana z:

- Wejściowy zestaw danych inicjowania CSQINP1
- Wejściowy zestaw danych inicjowania CSQINP2
- Konsola programu z/OS (lub jej odpowiednik)
- SYSTEM.COMMAND.INPUT , kolejka i serwer komend (z aplikacji, CSQUTIL lub zestawu danych wejściowych inicjowania CSQINPX)

Tabela 26. 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
ZMIENŃ 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
ALTER QMGR		X	X	X
ALTER QMODEL		X	X	X
ALTER QREMOTE		X	X	X
ZMIENŃ 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

Tabela 26. Źródła, z których można uruchamiać komendy MQSC (kontynuacja)

Komenda	CSQINP1	CSQINP2	konsola z/OS	Kolejka wejściowa komend i serwer
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
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

Tabela 26. Źródła, z których można uruchamiać komendy MQSC (kontynuacja)

Komenda	CSQINP1	CSQINP2	konsola z/OS	Kolejka wejściowa komend i serwer
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
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
Rozstrzygnięcie kanału			X	X
ROZSTRZYGNIJ WĄTPLIWOŚĆ		X	X	X
WZNÓW MENEDŻERA KOLEJEK			X	X
ZABEZPIECZENIA RVERIFY		X	X	X

Tabela 26. Źródła, z których można uruchamiać komendy MQSC (kontynuacja)

Komenda	CSQINP1	CSQINP2	konsola z/OS	Kolejka wejściowa komend i serwer
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
Zamknij kanał			X	X
ZATRZYMAJ CHINIT			X	X
STOP CMDSERV	X	X	X	
Zatrzymaj proces nastuchujący			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 kolejkowania (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 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 menedżer 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.

Program narzędziowy qload

Z poziomu produktu IBM MQ 8.0 program narzędziowy **qload** dostarczany w programie IBM MQ Supportpac MO03 został zintegrowany z produktem IBM MQ jako program narzędziowy **dmpmqmsg**.

W systemie z/OS program narzędziowy jest dostępny jako moduł wykonywalny, CSQUDMSG w bibliotece SCSQLOAD, z aliasem QLOAD dla kompatybilności. Przykładowy kod JCL jest również udostępniany jako element CSQ4QLOAD w SCSQPROC.

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.

LOAD

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 zresetuj 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 przetączyć lub wystać zapytanie do kolejki transmisji powiązanej z kanałami nadawczym klastra.

XPARM

Aby przekształcić moduł ładujący parametr inicjatora kanału w atrybuty menedżera kolejek (dla celów migracji).

z/OS Operacyjny IBM MQ for z/OS

Te podstawowe procedury umożliwiają działanie produktu IBM MQ for z/OS.

Operacje opisane w tej sekcji można również wykonać za pomocą konsoli IBM MQ Explorer, która jest dystrybuowana z IBM MQ for Windows i IBM MQ dla serwerów Linux (platformy x86 i x86-64). Autonomiczny produkt IBM MQ Explorer jest dostępny do pobrania z produktu Fix Central. Więcej informacji na ten temat zawiera sekcja [“Administrowanie przy użyciu IBM MQ Explorer”](#) na stronie 117.

z/OS 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 26. 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 435).
- Konsola z/OS lub odpowiednik, taki jak SDSF
- Procedura komendy get dla systemu głównego z/OS, MGCRE (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 460.

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 445.

Więcej informacji na ten temat zawiera sekcja

- [“Wydawanie komend z konsoli programu z/OS lub jej odpowiedników”](#) na stronie 440
 - [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 32](#) na stronie 440.

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 32](#) na stronie 440. Jednak w przypadku złożonych komend lub w przypadku zestawów komend, które są często wydawane, inne metody wydawania komend są lepsze.

```
+CSQ1 DISPLAY QUEUE(TRANSMIT.QUEUE.PROD) TYPE(QLLOCAL)
```

Rysunek 32. 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, w związku z czym można używać 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 rozpoczniesz IBM MQ” na stronie 441](#)
- [“Uruchamianie menedżera kolejek” na stronie 441](#)
- [“Zatrzymywanie menedżera kolejek” na stronie 443](#)

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 rozpoczniesz 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, tj. grupy komend SYS systemu 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 33 na](#)

stronie 442 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 33. Uruchamianie menedżera kolejek za pomocą 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 współużytkownika 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.

W produkcie IBM MQ 9.1.0 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](#).

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 34](#) na stronie 443.

```

//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
//BDS1 DD DISP=SHR,DSN=myqual.BSDS01
//BDS2 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 34. 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.

Rozpoczynanie 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 komendy STOP QMGR system zakończy pracę w uporządkowany sposób i przed zatrzymaniem podejmuje 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 35 na stronie 444](#) kończy działanie działającego menedżera kolejek.

```
+CSQ1  STOP QMGR
+CSQ1  STOP QMGR MODE(QUIESCE)
+CSQ1  STOP QMGR MODE(FORCE)
+CSQ1  STOP QMGR MODE(RESTART)
```

Rysunek 35. 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) wyłuskiwać 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 523.

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
```

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 IBM MQ for z/OS (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

e. konsola WWW IBM MQ

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](#).

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; `th1qua1` to kwalifikator wysokiego poziomu dla używanych bibliotek IBM MQ, a `lang1etter` 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 różne proponowane 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.

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 CFSTATUS	Status struktury CF

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 wznowiania.
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			

Tabela 27. 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
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			
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

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.

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 pozostanie 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 wystane, zostaną wykonane we wszystkich menedżerach kolejek w grupie współużytkowania kolejek. Jeśli to pole pozostanie 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, to 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 w sposób zadowalający tylko z menedżerami kolejek uruchomionym w systemie z/OS w wersji IBM WebSphere MQ 7.10 lub nowszej.

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.

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 453](#)
 - [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 36 na stronie 454](#). 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 wróć 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 zapamiętywać dla kolejnych sesji z panelami operacyjnymi i kontrolnymi.

Korzystanie z operacji i paneli sterowania w systemie z/OS

Ten temat zawiera informacje dotyczące początkowego panelu sterującego wyświetlanego w języku CSQOREXX

Rysunek 36 na stronie 454 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, 2024. All rights reserved.

Command ==>
F1=Help      F2=Split      F3=Exit      F4=Prompt      F9=SwapNext F10=Messages
F12=Cancel
```

Rysunek 36. Panel początkowy operacji i sterowania systemu IBM MQ

Na tym panelu można wykonywać następujące działania:

- Wybierz żądany menedżer kolejek lokalnych i określ, czy komendy mają być wykonywane w tym menedżerze kolejek, w zdalnym menedżerze kolejek, czy w innym menedżerze kolejek w tej samej grupie współużytkowania kolejek, co menedżer kolejek lokalnych. Wpisz nazwę menedżera kolejek, jeśli chcesz ją zmienić.
- Wybierz działanie, które chcesz wykonać, wpisując odpowiedni numer w polu **Działanie**.
- Określ typ obiektu, z którym chcesz pracować. Naciśnij klawisz funkcyjny F1, aby uzyskać pomoc dotyczącą typów obiektów, jeśli nie masz pewności, czym one są.
- Określ dyspozycję typu obiektu, z którym chcesz pracować.
- Wyświetla 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 kolejności. Wszystkie działania są dostępne na liście.

Uwaga: Zaleca się dokonanie wyboru, który spowoduje wyświetlenie listy obiektów, a następnie pracę z tą listą. Użyj działania **Wyświetl**, ponieważ jest ono 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, przesuń 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:

- **WyjścieF3** 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 rozporządzania. 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 rozporządzania. 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 wystąpić 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 wystąpić 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 457](#).

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.
- Zaczynaj 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 wysłać 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.

z/OS 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 przełączania” na stronie 458.](#)
- Jeśli używane jest podejście etapowo, patrz [“Zmiana kanałów nadajnika klastra przy użyciu metody stopniowej” na stronie 458.](#)

z/OS Korzystanie z automatycznej definicji kolejek i przełączania

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
```

z/OS 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 wystanie.
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 .

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:

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 26.

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, nazywanych 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 skonfigurowanie 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.

Korzystanie z serwera komend

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 zawiera sekcja [“Interpretowanie komunikatów odpowiedzi z serwera komend”](#) na stronie 466. 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. Staże 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 tracone.

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 zakończonego 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 składa je 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 466.

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 466.

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

<i>Feedback</i>	MQFB_NONE
<i>Encoding</i>	MQENC_NATIVE
<i>Priority</i>	Tak jak w przypadku deskryptora MQMD w komunikacie, który został wygenerowany.
<i>Persistence</i>	Tak jak w przypadku deskryptora MQMD w komunikacie, który został wygenerowany.
<i>CorrelId</i>	Zależy od opcji raportu MQPUT .
<i>ReplyToQ</i>	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 przyczyn .

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 439.

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:

```
CSQN205I    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=00000000, 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 CSQMMSG ' 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 CSQMMSG ' 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 CSQMMSG ' 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 473](#)
- [“Zarządzanie zestawem danych programu startowego \(BSDS\)” na stronie 482](#)
- [“Zarządzanie zestawami stron” na stronie 491](#)
- [“Tworzenie kopii zapasowych i odtwarzanie zestawów stron” na stronie 497](#)

- [“Tworzenie kopii zapasowych i odtwarzanie kolejek za pomocą komendy CSQUTIL” na stronie 501](#)
- [“Zarządzanie pulami buforów” na stronie 502](#)
- [“Zarządzanie grupami współużytkowania kolejek i współużytkowanymi kolejkami w systemie z/OS” na stronie 503](#)

Pojęcia pokrewne

[IBM MQ for z/OS pojęcia](#)

[“administrowanie IBM MQ for z/OS” na stronie 427](#)

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 428](#)

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 513](#)

W tym temacie opisano mechanizmy odtwarzania i restartowania używane przez produkt IBM MQ.

Zadania pokrewne

[Planowanie środowiska IBM MQ w systemie z/OS](#)

[Konfigurowanie menedżerów kolejek w systemie z/OS](#)

[Korzystanie z programów narzędziowych IBM MQ for z/OS](#)

Odsyłacze pokrewne

[“Źródła, z których można wydawać komendy MQSC w systemie z/OS” na stronie 21](#)

Komendy MQSC mogą być wydawane z różnych źródeł, w zależności od komendy.

[“Programy narzędziowe serwera IBM MQ for z/OS” na stronie 436](#)

Produkt IBM MQ for z/OS udostępnia zestaw programów narzędziowych, które mogą być pomocne w administrowaniu systemem.

[Skorowidz formatów komend programowalnych](#)

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 zatwierdzenia, a wynikowy punkt spójności jest przechwytywany w bieżącym aktywnym dzienniku, zanim zostanie on odciążony.

Należy rozważyć użycie opcji **MODE (QUIESCE)** podczas planowania strategii składowania na potrzeby odtwarzania po stronie serwisu. 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** , to okres wyciszania będzie domyślnie ustawiony na wartość parametru **QUIESCE** makra CSQ6ARVP . Jeśli czas wymagany do zakończenia operacji ARCHIVE LOG MODE (QUIESCE) jest krótszy niż określony czas, komenda zakończy się pomyślnie. W przeciwnym razie wykonanie komendy nie powiedzie się, 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 przetwarzania **ARCHIVE LOG** .

Uwaga: Użycie opcji **TIME** w czasie, gdy czas jest krytyczny, może znacznie zakłócić dostępność produktu IBM MQ dla wszystkich zadań i użytkowników korzystających z zasobów produktu IBM MQ .

Domyślnie komenda jest przetwarzana asynchronicznie od momentu wprowadzenia komendy. (Aby przetworzyć komendę synchronicznie z innymi komendami IBM MQ , należy użyć opcji WAIT (YES) z produktem **QUIESCE**, ale należy pamiętać, że konsola z/OS jest zablokowana z poziomu danych wyjściowych komendy IBM MQ dla całego okresu **QUIESCE** .

W okresie wyciszania:

- Zadania i użytkownicy w menedżerze kolejek mogą przechodzić przez przetwarzanie zatwierdzania, ale są zawieszani, 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, którzy zostali zawieszani.
- Nowe zadania mogą być uruchamiane, ale nie mogą aktualizować danych.

Dane wyjściowe komendy **DISPLAY LOG** korzystają z komunikatu CSQV400I w celu wskazania, że wygaszenie jest aktywne.

V 9.2.0 Na przykład z programu IBM MQ 9.1.4:

```
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 zHyperWrite Encrypted DSName
  1    68 NO          NO          VICY.CSQ1.LOGCOPY1.DS01
  2    68 NO          NO          VICY.CSQ1.LOGCOPY2.DS01
Restarted at 2019-08-15 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
```

Na przykład w przypadku systemów Long Term Support i Continuous Delivery przed IBM MQ 9.1.2:

```
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
WRTHRSH     20
DEALLCT      0
COMPLG       NONE
ZHYWRITE     NO
              YES
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

```

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 odciążający został uruchomiony.

Jeśli aktualizacje nie mogą być wyciszane przed upływem okresu wyciszenia, program IBM MQ wysyła komunikat CSQJ317I, a przetwarzanie **ARCHIVE LOG** kończy działanie. Bieżące aktywne zestawy danych dziennika nie są obcinane ani przełączane na następne dostępne zestawy danych dziennika, a proces odciążania nie jest uruchomiony.

Niezależnie od tego, czy wyciszenie powiodło się, wszyscy zawieszoni użytkownicy i zadania są wznawiane, a program IBM MQ wysyła komunikat CSQJ312I, stwierdzając, że wyciszenie zostało zakończone, a działanie aktualizacji jest wznawiane.

Jeśli **ARCHIVE LOG** jest wydawana, 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 BIEŻĄCY AKTYWNY LOG DATA SET IS THE LAST
DOSTĘPNY AKTYWNY ZESTAW DANYCH DZIENNIKA. PRZETWARZANIE DZIENNIKA
ARCHIWALNEGO ZOSTANIE ZAKOŃCZONE

```

Jeśli **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 -KOMENDA ARCHIWIZACJI DZIENNIKA JEST JUŻ W TOKU

```

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. Proces archiwizacji można anulować, a następnie zrestartować go za pomocą następującej 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 komenda 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 [Tailor your system parameter module](#) (Moduł 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 28](#) na stronie 476:

<i>Tabela 28. Archiwizowanie i rejestrowanie parametrów, które można zmieniać w czasie działania menedżera kolejek</i>	
komenda SET	Parametry
LOG	WRTHRSR, 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 25](#)
- 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 przestonyty 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 484. 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 CSQI024I i CSQI025I. Ten RBA jest uzyskiwany za pomocą komendy DISPLAY USAGE, która została wydana podczas tworzenia punktu odtwarzania za pomocą 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 CSQI024I i CSQI025I. 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 486.

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 w 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 486.

Efekt rozdrabniania 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 przetaczanych 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 działania 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 zgłasza następujące trzy informacje o odtwarzaniu:

1. Ilość dziennika musi być zachowana dla jednostki odzyskiwania pracy.
2. Ilość dziennika musi być zachowana na potrzeby odtwarzania nośników w zestawach stron.
3. W przypadku menedżera kolejek w grupie współużytkowania kolejek, jaka część dziennika musi być przechowywana w celu odtworzenia nośnika struktur CF.

W przypadku każdego z tych informacji podejmowana jest próba odwzorowania najstarszych danych dziennika wymaganych do zestawu danych. Ponieważ nowe jednostki pracy są uruchamiane i zatrzymują się, (1) oczekuje się, że w dzienniku zostanie przesunięta do ostatniej pozycji. 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, kiedy ostatnio utworzono kopię zapasową zestawów stron lub może być używana kopia zapasowa, która może być używana, 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.

z/OŚ **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 nieodtworzalne. Koniec dziennika RBA jest albo wartością FFFFFFFFFF (jeśli używane jest 6-bajtowe RBAs), albo FFFFFFFFFFFFFFFF (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 FFF800000000 (jeśli używane są 6-bajtowe dzienniki RBAs) lub FFFFFFFC000000000 (jeśli używane są 8-bajtowe RBAs dziennika).

Jeśli używane są 6-bajtowe RBAs dziennika, należy rozważyć przekształcenie menedżera kolejek w taki sposób, aby używało 8-bajtowych dzienników RBA, a nie resetowanie dziennika menedżera kolejek, postępując zgodnie z procesem opisanym w sekcji “Implementowanie większego adresu bajtu względnego w dzienniku” na stronie 480. 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 do zestawów stron.

3. Jeśli menedżer kolejek jest częścią grupy współużytkownika kolejek, należy wykonać kopie zapasowe CFSTRUCT w innych menedżerach kolejek dla wszystkich struktur w grupie współużytkownika 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 może być resetowany niezależnie, dlatego można przedstawić wiele współbieżnych zadań (na przykład jeden na zestaw 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 480

Przed IBM MQ for z/OS 8.0 produkt IBM MQ for z/OS użył 6-bajтового dziennika RBA w celu zidentyfikowania położenia danych w dzienniku. W produkcie IBM MQ for z/OS 8.0 dziennik RBA dziennika może mieć długość 8 bajtów, co zwiększa czas przed zresetowaniem dziennika.

Implementowanie większego adresu bajtu względnego w dzienniku

Przed IBM MQ for z/OS 8.0 produkt IBM MQ for z/OS użył 6-bajтового dziennika RBA w celu zidentyfikowania położenia danych w dzienniku. W produkcie IBM MQ for z/OS 8.0 dziennik RBA dziennika może mieć długość 8 bajtów, co zwiększa czas przed zresetowaniem dziennika.

W przypadku menedżerów kolejek produktu IBM MQ 9.2.0 Long Term Support i dla menedżerów kolejek ciągłych dostarczania utworzonych przed produktem IBM MQ 9.2.4 ta funkcja musi być jawnie włączona.

V 9.2.5 W przypadku menedżerów kolejek utworzonych w systemie IBM MQ 9.2.5 i nowszych ten składnik jest już włączony.

Sekcja Planowanie zwiększenia maksymalnego zakresu dziennika adresowalnego zawiera uwagi dotyczące planowania włączenia 8-bajowego dziennika RBA.

Ważne: **V 9.2.0** Jeśli menedżer kolejek nie jest częścią grupy współużytkowania kolejek, a następnie migrujesz z powrotem do programu IBM MQ for z/OS 9.0.0, po włączeniu 8-bajowego dziennika RBA, należy upewnić się, że w tych wersjach jest używany produkt **OPMODE=NEWFUNC, 900**, w przeciwnym razie uruchomienie menedżera kolejek nie powiedzie się.

Aby włączyć 8-bajowy dziennik RBA w jednym menedżerze kolejek produktu IBM MQ for z/OS, wykonaj poniższe instrukcje w wyświetlonej kolejności:

1. **V 9.2.0** Jeśli menedżer kolejek znajduje się w grupie współużytkowania kolejek, przed przejściem do kroku "2" na stronie 481 upewnij się, że wszystkie menedżery kolejek w grupie współużytkującej kolejkę, które znajdują się w IBM MQ for z/OS 9.0.0, są uruchomione z programem **OPMODE= (NEWFUNC, 900)**.

Aby to zrobić, nie trzeba wykonywać całkowitego wyłączenia grupy współużytkowania kolejki. Można zatrzymać każdy menedżer kolejek, który jest w stanie IBM MQ for z/OS 9.0.0 z kolei, zmienić go na **OPMODE= (NEWFUNC, 900)** i zrestartować go.

Gdy wszystkie menedżery kolejek produktu IBM MQ for z/OS 9.0.0 w grupie współużytkowania kolejek są uruchomione z programem **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 serwerem 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. Przed wystaniem zadania należy sprawdzić atrybuty nowego BSDS, aby przydzielić nowe 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. Przykładowe JCL w pliku thlqual.SCSQPROC(CSQ4BSDS) zawiera rekomendowane wartości podczas definiowania nowego BSDS.
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-bajowego 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 stać się bieżącym BSDS.
- c. Restartowanie menedżera kolejek.

Po pomyślnym zrestartowaniu menedżera kolejek przy użyciu przekształconego serwera BSDS nie należy uruchamiać menedżera kolejek przy użyciu starego 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 dla nowego menedżera kolejek, przed pierwszym uruchomieniem, należy najpierw utworzyć pusty format BSDS w wersji 1 i użyć go jako danych wejściowych do 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 .

Pojęcia pokrewne

Względny adres w bajtach względnego dziennika

Zadania pokrewne

Planowanie zwiększenia maksymalnego zakresu dziennika adresowalnego

Odsyłacze pokrewne

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 482
- “Zmiana BSDS” na stronie 484
- “Odtwarzanie BSDS” na stronie 488

Znajdowanie informacji o tym, co zawiera BSDS

Aby sprawdzić zawartość BSDS, można użyć programu narzędziowego do map 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 nie może być aktualizowane przez kilka sekund, minut lub nawet godzin. Szczegółowe informacje na temat progu zapisu można znaleźć w parametrze WRTHRSH 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 obciążonym 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.

System 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.

Zmiany w dziennikach aktywnych

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 DSNAMES=MQM111.LOGCOPY1.DS10,COPY1
NEWLOG DSNAMES=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 DSNAMES=MQM111.LOGCOPY1.DS99
DELETE DSNAMES=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 DSNAMES=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 przedsięwzięcie 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.

Zmiany w dziennikach archiwalnych

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 DSNAME=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 DSNAME=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.

Zmiana kwalifikatora wysokiego poziomu (HLQ) dla dzienników i BSDS

Ten temat zawiera opis procedury wymaganej do zmiany kwalifikatora wysokiego poziomu (HLQ).

Zanim rozpoczniesz

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. Aby to zrobić, wykonaj następujące kroki:

Procedura

1. Uruchom program narzędziowy do drukowania dziennika CSQJU004 , aby zarejestrować informacje o zestawie danych dziennika. Te informacje są potrzebne później.
2. Dostępne są następujące 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 zmieniła.

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.

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 usług metod dostępu, aby zmienić nazwę uszkodzonego BSDS lub usunąć uszkodzone BSDS i zdefiniować nowe BSDS o tej samej nazwie, co zniszczony BSDS. Przykładowe instrukcje sterujące można znaleźć w zadaniu CSQ4BREC w pliku thlqual.SCSQPROC.
2. Wydać 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 archiwum

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), należy znaleźć 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 489. Nie należy kopiować żadnych danych do drugiego zastąpienia BSDS, należy to zrobić w kroku [“5”](#) na stronie 490.

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 skopiowano 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 na 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 zapewnienia. 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

W tym temacie opisano sposób zarządzania zestawami stron powiązanych z menedżerem kolejek.

W tym temacie opisano sposób dodawania, kopiowania i ogólnego zarządzania zestawami stron powiązanych z menedżerem kolejek. Zawiera on następujące sekcje:

- [“Jak zmienić kwalifikator wysokiego poziomu \(HLQ\) dla zestawów stron” na stronie 491](#)
- [“Dodawanie zestawu stron do menedżera kolejek” na stronie 491](#)
- [“Co zrobić, gdy jeden z zestawów stron zostanie zapelniony” na stronie 492](#)
- [“Jak zrównoważyć obciążenie zestawów stron” na stronie 492](#)
- [Jak zwiększyć wielkość zestawu stron](#)
- [“Jak zmniejszyć zestaw stron” na stronie 496](#)
- [“Jak ponownie wprowadzić zestaw stron” na stronie 496](#)
- [“Tworzenie kopii zapasowych i odtwarzanie zestawów stron” na stronie 497](#)
- [“Usuwanie zestawów stron” na stronie 501](#)
- [“Tworzenie kopii zapasowych i odtwarzanie kolejek za pomocą komendy CSQUTIL” na stronie 501](#)

Sekcja [Zestawy stron](#) zawiera opis zestawów stron, klas pamięci masowej, buforów i pul buforów, a także niektóre zagadnienia dotyczące wydajności, które mają zastosowanie.

Jak zmienić kwalifikator wysokiego poziomu (HLQ) dla zestawów stron

To zadanie zawiera informacje o sposobie zmiany kwalifikatora 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 w przypadku starych zestawów stron, skopiuj istniejący zestaw stron za pomocą REPRO do pustych nowych zestawów stron HLQ.
3. W przypadku zwiększania wielkości zestawów stron należy użyć funkcji FORMAT programu CSQUTIL, aby sformatować strony docelowe, a następnie funkcji COPYPAGE programu CSQUTIL, aby skopiować wszystkie komunikaty ze źródłowego zestawu stron do docelowego zestawu stron.

Więcej informacji na ten temat zawiera sekcja [Formatowanie zestawów stron \(FORMAT\)](#) oraz sekcja [Rozwijanie zestawu stron \(COPYPAGE\)](#).

4. Zmień instrukcję DD CSQP00xx w procedurze menedżera kolejek, aby wskazywała na nowe zestawy stron HLQ.

Zrestartuj menedżer kolejek i sprawdź zmiany w zestawach stron.

Dodawanie zestawu stron do menedżera kolejek

W tym opisie założono, że menedżer kolejek jest już uruchomiony. Dodanie zestawu stron może być konieczne na przykład wtedy, gdy menedżer kolejek musi obsłużyć nowe aplikacje przy użyciu nowych kolejek.

Aby dodać nowy zestaw stron, wykonaj następującą procedurę:

1. Zdefiniuj i sformatuj nowy zestaw stron. Jako podstawy można użyć przykładowego kodu JCL z pliku thlqual.SCSQPROC(CSQ4PAGE). Więcej informacji na ten temat zawiera sekcja [Formatowanie zestawów stron \(FORMAT\)](#).

Należy uważać, aby nie sformatować żadnych używanych zestawów stron, chyba że jest to zamierzone. Jeśli tak, należy użyć opcji FORCE funkcji programu narzędziowego FORMAT.

2. Użyj komendy DEFINE PSID z opcją DSN, aby powiązać zestaw stron z pulą buforów.
3. Dodaj odpowiednie definicje klas pamięci masowej dla zestawu stron, wydając komendy DEFINE STGCLASS.
4. Opcjonalnie, aby udokumentować sposób konfiguracji menedżera kolejek:
 - a. Dodaj nowy zestaw stron do procedury uruchomionego zadania dla menedżera kolejek.
 - b. Dodaj definicję nowego zestawu stron do zestawu danych inicjowania CSQINP1 .
 - c. Dodaj definicję nowej klasy pamięci masowej do elementu zestawu danych inicjowania CSQ4INJR .

Szczegółowe informacje na temat komend DEFINE PSID i DEFINE STGCLASS można znaleźć w sekcji [DEFINE PSID](#) i w sekcji [DEFINE STGCLASS](#).

Co zrobić, gdy jeden z zestawów stron zostanie zapełniony

Informacje o wykorzystaniu zestawów stron można uzyskać za pomocą komendy DISPLAY USAGE systemu IBM MQ . Na przykład komenda:

```
DISPLAY USAGE PSID(03)
```

Wyświetla bieżący stan zestawu stron 03. Informuje o liczbie wolnych stron w tym zestawie stron.

Jeśli zdefiniowano dodatkowe obszary dla zestawów stron, są one dynamicznie rozwijane przy każdym zapełnieniu. Ostatecznie używane są wszystkie obszary dodatkowe lub nie jest dostępne żadne dodatkowe miejsce na dysku. W takim przypadku aplikacja otrzymuje kod powrotu MQRC_STORAGE_MEDIUM_FULL.

Jeśli aplikacja otrzyma kod powrotu MQRC_STORAGE_MEDIUM_FULL z wywołania MQI, jest to wyraźne wskazanie, że w zestawie stron nie ma wystarczającej ilości wolnego miejsca. Jeśli problem nie ustąpi lub prawdopodobnie się powtórzy, należy coś zrobić, aby go rozwiązać.

Ten problem można rozwiązać na kilka sposobów:

- Równoważenie obciążenia między zestawami stron przez przenoszenie kolejek między zestawami stron.
- Rozwiń zestaw stron. Instrukcje znajdują się w sekcji [“Jak zwiększyć wielkość zestawu stron”](#) na stronie [494](#) .
- Przeedyfniuj zestaw stron tak, aby jego wielkość przekraczała 4 GB do maksymalnej wielkości 64 GB. Instrukcje znajdują się w sekcji [Definiowanie zestawu stron o wielkości większej niż 4 GB](#) .

Jak zrównoważyć obciążenie zestawów stron

Równoważenie obciążenia w zestawach stron oznacza przenoszenie komunikatów powiązanych z jedną lub większą liczbą kolejek z jednego zestawu stron do innego, mniej używanego zestawu stron. Tej techniki należy użyć, jeśli rozwijanie zestawu stron jest niepraktyczne.

Aby zidentyfikować kolejki korzystające z zestawu stron, należy użyć odpowiednich komend 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 których klas 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 poniższą procedurę dla każdej z tych kolejek:

1. Upewnij się, że kolejka, którą chcesz przenieść, nie jest używana przez żadne aplikacje (wartości IPPROCS i OPPROCS z komendy DISPLAY QSTATUS wynoszą zero) i że nie zawiera ona niezatwierdzonych komunikatów (wartość UNCOM z komendy DISPLAY QSTATUS wynosi NO).

Uwaga: Jedynym sposobem zapewnienia, że ten stan będzie kontynuowany, jest tymczasowa zmiana autoryzacji zabezpieczeń kolejki. Więcej informacji na ten temat zawiera sekcja [Profile zabezpieczeń kolejki](#).

Jeśli nie można tego zrobić, późniejsze etapy tej procedury mogą zakończyć się niepowodzeniem, jeśli aplikacje zaczną korzystać z kolejki pomimo kroków zapobiegawczych, takich jak ustawienie PUT (DISABLED). Jednak ta procedura nigdy nie może utracić komunikatów.

2. Aby uniemożliwić aplikacjom umieszczanie komunikatów w przenoszonym kolejce, należy zmienić definicję kolejki w celu wyłączenia operacji MQPUT. Zmień definicję kolejki na PUT (DISABLED).
3. Za pomocą komendy zdefiniuj kolejkę tymczasową z takimi samymi atrybutami, jak przenoszona kolejka:

```
DEFINE QL(TEMP_QUEUE) LIKE(Queue_To_Move) PUT(ENABLED) GET(ENABLED)
```

Uwaga: Jeśli ta kolejka tymczasowa już istnieje od poprzedniego uruchomienia, usuń ją przed wykonaniem operacji definiowania.

4. Przenieś komunikaty do kolejki tymczasowej za pomocą następującej komendy:

```
MOVE QLOCAL(Queue_To_Move) TOQLOCAL(TEMP_QUEUE)
```

5. Usuń przenoszona kolejkę za pomocą komendy:

```
DELETE QLOCAL(Queue_To_Move)
```

6. Zdefiniuj nową klasę pamięci masowej, która jest odwzorowana na wymagany zestaw stron, na przykład:

```
DEFINE STGCLASS(NEW) PSID(nn)
```

Dodaj nową definicję klasy pamięci masowej do zestawów danych CSQINP2 gotowych do następnego restartu menedżera kolejek.

7. Zmień definicję przenoszonym kolejki, zmieniając atrybut klasy pamięci:

```
DEFINE QL(Queue_To_Move) LIKE(TEMP_QUEUE) STGCLASS(NEW)
```

Po ponownym zdefiniowaniu kolejki jest ona oparta na kolejce tymczasowej utworzonej w kroku "3" na stronie 493.

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 493 nie jest już wymagana. Użyj następującej komendy, aby go usunąć:

```
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, aby zastąpić istniejącą definicję kolejki.

Rysunek 37 na stronie 494 przedstawia wyodrębnienie z zadania równoważenia obciążenia.

```
//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)
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 37. Wyodrębnienie z zadania równoważenia obciążenia dla zestawu stron

Jak zwiększyć wielkość zestawu stron

Początkowo można przydzielić zestaw stron większy niż 4 GB, patrz sekcja [Definiowanie zestawu stron większego niż 4 GB](#).

Zestaw stron można zdefiniować w taki sposób, aby był automatycznie rozwijany po zapełnieniu przez określenie parametru EXPAND (SYSTEM) lub EXPAND (USER). Jeśli zestaw stron został zdefiniowany z opcją EXPAND (NONE), można go rozwinąć na jeden z dwóch sposobów:

- Zmień jego definicję, aby umożliwić automatyczne rozszerzanie. Więcej informacji na ten temat zawiera sekcja [Zmiana zestawu stron w celu umożliwienia automatycznego rozwijania](#).
- Utwórz nowy, większy zestaw stron i skopiuj komunikaty ze starego zestawu stron do nowego. Więcej informacji na ten temat zawiera sekcja [Przenoszenie komunikatów do nowego, większego zestawu stron](#).

Definiowanie zestawu stron o wielkości większej niż 4 GB

System IBM MQ może używać strony o wielkości do 64 GB, pod warunkiem, że zestaw danych jest zdefiniowany dla VSAM z opcją 'extended addressability' (rozszerzona adresowalność). Rozszerzona adresowalność jest atrybutem, który jest nadawany przez klasę danych SMS.

Uwaga: Zestawy stron i zestawy danych dziennika aktywnego mogą rezydować w części rozszerzonej przestrzeni adresowej (EAS) woluminu z rozszerzonym adresem (EAV), a od wersji z/OS V1.12 zestaw danych dziennika archiwalnego może również rezydować w EAS.

W przykładzie przedstawionym w poniższym przykładzie JCL klasa zarządzania 'EXTENDED' jest zdefiniowana jako SMS z opcją 'Extended addressability'. Jeśli istniejący zestaw stron nie jest obecnie zdefiniowany jako posiadający rozszerzoną adresowalność, należy użyć następującej metody, aby przeprowadzić migrację do zestawu danych w rozszerzonym formacie adresowalności.

1. Zatrzymaj menedżer kolejek.
2. Użyj programu Access Method Services, aby zmienić nazwę istniejącego zestawu stron.
3. Zdefiniuj docelowy zestaw stron, taki sam rozmiar jak istniejący zestaw stron, ale z opcją DATACLAS (EXTENDED).

Uwaga: Zestawy danych w formacie rozszerzonym muszą być zarządzane przez SMS. Poniżej przedstawiono mechanizmy żądania rozszerzonego formatu dla zestawów danych VSAM:

- Użycie klasy danych, która ma wartość DSNTYPE równą EXT i podparametru R lub P w celu wskazania wymaganego lub preferowanego.
- Kodowanie DSNTYPE=EXTREQ (wymagany format rozszerzony) lub DSNTYPE=EXTPREF (preferowany format rozszerzony) w instrukcji DD.
- Kodowanie parametru LIKE= w instrukcji DD tak, aby odwoływał się do istniejącego zestawu danych w formacie rozszerzonym.

Więcej informacji na ten temat zawiera sekcja [Ograniczenia dotyczące definiowania zestawów danych w formacie rozszerzonym](#).

4. Użyj funkcji COPYPAGE programu CSQUTIL, aby skopiować wszystkie komunikaty ze źródłowego zestawu stron do docelowego zestawu stron. Więcej szczegółów zawiera sekcja [Rozwijanie zestawu stron \(COPYPAGE\)](#).
5. Zrestartuj menedżer kolejek.
6. Zmień zestaw stron w taki sposób, aby używał rozszerzania systemu, co umożliwi mu dalsze zwiększanie się poza jego bieżącą alokację.

Poniższy kod JCL przedstawia przykładowe komendy usług metod dostępu:

```
//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))
/*
```

Modyfikowanie zestawu stron w celu umożliwienia automatycznego rozwijania

Użyj komendy ALTER PSID z opcjami EXPAND (USER) lub EXPAND (SYSTEM). Ogólne informacje dotyczące rozwijania zestawów stron można znaleźć w sekcji [ALTER PSID \(ALTER PSID\)](#) i [Expanding a page set \(COPYPAGE\)](#) (Rozwijanie zestawu stron-COPYPAGE).

Przenoszenie wiadomości do nowego, większego zestawu stron

Ta technika obejmuje zatrzymanie i restartowanie menedżera kolejek. Spowoduje to usunięcie wszystkich nietrwałych komunikatów, które nie znajdują się we współużytkowanych kolejkach w czasie restartu. Jeśli istnieją nietrwałe komunikaty, które nie mają zostać usunięte, należy użyć funkcji równoważenia obciążenia. Więcej informacji na ten temat zawiera sekcja ["Jak zrównoważyć obciążenie zestawów stron"](#) na stronie 492. W tym opisie zestawu stron, który ma zostać rozwinięty,

jest nazywany *źródłowym* zestawem stron. Nowy, większy zestaw stron jest nazywany *docelowym* zestawem stron.

Wykonaj następujące kroki:

1. Zatrzymaj menedżer kolejek.
2. Zdefiniuj docelowy zestaw stron, upewniając się, że jest on większy niż źródłowy zestaw stron, z większą wartością dodatkowego zakresu.
3. Użyj funkcji `FORMAT` programu `CSQUTIL`, aby sformatować docelowy zestaw stron. Więcej informacji na ten temat zawiera sekcja [Formatowanie zestawów stron \(FORMAT\)](#).
4. Użyj funkcji `COPYPAGE` programu `CSQUTIL`, aby skopiować wszystkie komunikaty ze źródłowego zestawu stron do docelowego zestawu stron. Więcej szczegółów zawiera sekcja [Rozwijanie zestawu stron \(COPYPAGE\)](#).
5. Zrestartuj menedżer kolejek przy użyciu zestawu stron docelowych, wykonując jedną z następujących czynności:
 - Zmień procedurę uruchomionego zadania menedżera kolejek tak, aby odwoływała się do docelowego zestawu stron.
 - Użyj usług metody programu `Access`, aby usunąć źródłowy zestaw stron, a następnie zmień nazwę docelowego zestawu stron, nadając mu taką samą nazwę, jak źródłowemu zestawowi stron.

Uwaga:

Przed usunięciem dowolnego zestawu stron IBM MQ należy upewnić się, że zostały utworzone wymagane kopie zapasowe.

Jak zmniejszyć zestaw stron

Uniemożliwia wszystkim użytkownikom innym niż administrator produktu IBM MQ korzystanie z menedżera kolejek. Na przykład przez zmianę ustawień zabezpieczeń dostępu.

Jeśli istnieje duży zestaw stron, który jest w większości pusty (jak pokazano w komendzie `DISPLAY USAGE`), można zmniejszyć jego wielkość. Procedura ta obejmuje użycie funkcji `COPY`, `FORMAT` i `LOAD` programu narzędziowego `CSQUTIL` (patrz [program narzędziowy IBM MQ](#)). Ta procedura nie działa w przypadku zerowego zestawu stron (0), ponieważ nie jest praktyczne, aby zmniejszyć wielkość tego zestawu stron. Jedynym sposobem jest reinicjowanie menedżera kolejek (patrz sekcja [“Reinicjowanie menedżera kolejek”](#) na stronie 522). Wymaganiem wstępnym tej procedury jest próba usunięcia wszystkich użytkowników z systemu, aby wszystkie jednostki UOWs były kompletne, 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, aby zastąpić duży zestaw stron.
4. Uruchom funkcję `FORMAT TYPE (NEW)` programu `CSQUTIL` dla zestawu stron utworzonego w kroku [“3”](#) na stronie 496.
5. Zrestartuj menedżer kolejek przy użyciu zestawu stron utworzonego w kroku [“3”](#) na stronie 496.
6. Uruchom funkcję `LOAD` programu `CSQUTIL`, aby załadować z powrotem wszystkie komunikaty zapisane w kroku [“2”](#) na stronie 496.
7. Zezwól wszystkim użytkownikom na dostęp do menedżera kolejek.
8. Usuń stary duży zestaw stron.

Jak ponownie wprowadzić zestaw stron

W niektórych scenariuszach przydatne jest ponowne wprowadzenie do menedżera kolejek starego zestawu stron w trybie z połączeniem. Jeśli nie zostanie wykonane konkretne działanie, po włączeniu

starego zestawu stron menedżer kolejek rozpoznaje, że archiwum RBA odtwarzania zestawu stron przechowywane w samym zestawie stron i w rekordach punktu kontrolnego jest stare i automatycznie rozpocznie odtwarzanie nośnika zestawu stron w celu jego uaktualnienia.

Takie odtwarzanie nośników może być wykonywane tylko podczas restartu menedżera kolejek i może zająć dużo czasu, zwłaszcza jeśli konieczne jest odczytanie dzienników archiwalnych znajdujących się na taśmie. Jednak zwykle w tym przypadku zestaw stron był odłączony przez ten czas i dlatego dziennik nie zawiera żadnych informacji dotyczących odtwarzania zestawu stron.

Dostępne są trzy następujące opcje:

Umożliwia wykonanie pełnego odtwarzania nośników.

1. Zatrzymaj menedżer kolejek.
2. Upewnij się, że definicje są dostępne dla zestawu stron zarówno w procedurze uruchomionego zadania dla menedżera kolejek, jak i w zestawie danych inicjowania CSQINP1 .
3. Zrestartuj menedżer kolejek.

Zezwalaj na niszczenie wszystkich komunikatów w zestawie stron.

Ta opcja jest użyteczna, gdy zestaw stron jest wyłączony przez długi czas (na przykład przez kilka miesięcy), a teraz zdecydowano o jego ponownym wykorzystaniu w innym celu.

1. Sformatuj zestaw stron za pomocą funkcji FORMAT programu CSQUTIL z opcją TYPE (NEW).
2. Dodaj definicje zestawu stron zarówno do procedury uruchomionego zadania dla menedżera kolejek, jak i do zestawu danych inicjowania CSQINP1 .
3. Zrestartuj menedżer kolejek.

Użycie opcji TYPE (NEW) do formatowania powoduje skasowanie bieżącej zawartości zestawu stron i nakazuje menedżerowi kolejek zignorowanie wszystkich informacji historycznych w punkcie kontrolnym dotyczących zestawu stron.

Ustaw zestaw stron w trybie z połączeniem, unikając procesu odtwarzania nośników.

Tej techniki należy używać tylko wtedy, gdy istnieje pewność, że zestaw stron był w trybie bez połączenia od momentu czystego zamknięcia menedżera kolejek. Ta opcja jest najbardziej odpowiednia w sytuacji, gdy zestaw stron był odłączony przez krótki czas, zwykle z powodu problemów z działaniem, takich jak tworzenie kopii zapasowej podczas uruchamiania menedżera kolejek.

1. Sformatuj zestaw stron za pomocą funkcji FORMAT programu CSQUTIL z opcją TYPE (REPLACE).
2. Dodaj zestaw stron z powrotem do menedżera kolejek dynamicznie przy użyciu komendy DEFINE PSID z opcją DSN lub zezwól na dodanie zestawu stron podczas restartu menedżera kolejek.

Użycie opcji TYPE (REPLACE) do formatowania powoduje sprawdzenie, czy zestaw stron został poprawnie zamknięty przez menedżer kolejek, i oznacza, że odtwarzanie nośnika nie zostanie wykonane. W treści zestawu stron nie są wprowadzane żadne inne zmiany.

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 498](#)
- [“Tworzenie kopii zapasowych zestawów stron” na stronie 499](#)
- [“Odzyskiwanie zbiorów stron” na stronie 500](#)
- [Usuwanie zestawów stron](#)

Informacje na temat sposobu tworzenia punktu odtwarzania dla zasobów współużytkowanych zawiera sekcja [“Odtwarzanie współużytkowanych kolejek” na stronie 507](#).

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 dzienników z tego punktu.

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. Wydadz komendę ARCHIVE LOG, aby upewnić się, że najnowsze dane dziennika są zapisywane w zestawach danych dziennika.
4. Wydadz 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 499).

Metoda 2: kopia zapasowa rozmytego

Ta metoda nie obejmuje wyłączenia 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 499).
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 499, 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 aplikację.
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 aplikację.
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](#).

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 502
- [“Jak usunąć pulę buforów”](#) na stronie 503

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 statycznych 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 501.

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 503
- [“Zarządzanie współużytkowanymi kolejkami”](#) na stronie 506
- [“Zarządzanie obiektami grupy”](#) na stronie 512
- [“Zarządzanie narzędziem CF”](#) na stronie 512

Zarządzanie grupami współużytkowania kolejek

Istnieje możliwość dodania lub usunięcia menedżera kolejek do grupy współużytkowania kolejek (QSG) oraz zarządzania powiązаныmi tabelami Db2.

Ten temat zawiera sekcje dotyczące następujących zadań:

- [“Konfigurowanie grupy współużytkowania kolejek”](#) na stronie 503
- [“Dodawanie menedżera kolejek do grupy współużytkowania kolejek”](#) na stronie 504
- [“Usuwanie menedżera kolejek z grupy współużytkowania kolejek”](#) na stronie 505
- [“Usuwanie grupy współużytkowania kolejek z tabel Db2”](#) na stronie 506
- [“Sprawdzanie spójności definicji Db2”](#) na stronie 506

Konfigurowanie grupy współużytkowania kolejek

Każda grupa współużytkowania kolejek ma nazwę składającą się maksymalnie z czterech znaków. 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, która używa grupy współużytkowania danych Db2, [skonfiguruj środowisko Db2](#).
2. [Skonfiguruj narzędzie CF](#).
3. Dodaj grupę współużytkowania kolejek do tabel Db2. Użyj funkcji `ADD QSG` programu narzędziowego grupy współużytkowania kolejek (`CSQ5PQSG`). Program ten jest opisany w sekcji [Program narzędziowy grupy współużytkowania kolejek](#). Przykład jest dostępny w pliku `thlqual.SCSQPROC(CSQ45AQS)`.
4. Dodaj menedżer kolejek do grupy współużytkowania kolejek, wykonując kroki opisane w sekcji [“Dodawanie menedżera kolejek do grupy współużytkowania kolejek”](#) na stronie 504

5. Zdefiniuj struktury aplikacji dla IBM MQ , wykonując kroki opisane w sekcji “Dodawanie struktury narzędzia CF” na stronie 512.
6. Jeśli jest to wymagane, przeprowadź migrację niewspółużytkowanych kolejek do kolejek współużytkowanych.
7. W celu zapewnienia dostępności należy utworzyć współużytkowane kanały do i z grupy współużytkowania kolejek.
 - W przypadku połączeń z grupą współużytkowania kolejek:
 - Skonfiguruj gniazdo VIPA lub router sprzętowy, aby rozdzielić obciążenie między dostępne menedżery kolejek w grupie QSG.
 - Zdefiniuj kanał odbiorczy za pomocą komendy QSGDISP (GROUP), aby upewnić się, że definicja kanału jest dostępna we wszystkich menedżerach kolejek w grupie QSG.
 - Uruchom proces nasłuchujący z wartością INDISP (GROUP) w każdym menedżerze kolejek dla połączeń kanału MCA do grupy QSG. Połączenia klienta z QSG powinny nadal łączyć się z programem nasłuchującym uruchomionym za pomocą komendy INDISP (QMGR).
 - Zmień aplikacje tak, aby używały nazwy QSG, a nie konkretnej nazwy menedżera kolejek.
 - Upewnij się, że reguły uwierzytelniania kanału we wszystkich menedżerach kolejek w grupie QSG są takie same, aby umożliwić aplikacjom nawiązanie połączenia z dowolnym menedżerem kolejek w grupie QSG.
 - W przypadku połączeń poza grupą współużytkowania kolejek:
 - Zdefiniuj współużytkowaną kolejkę transmisji.
 - Zdefiniuj kanał wychodzący z QSGDISP (GROUP) i DEFCDISP (SHARED).

W przypadku przekształcania istniejącego kanału w kanał współużytkowany może być konieczne wykonanie komendy RESET CHANNEL przed uruchomieniem kanału, ponieważ zmieniła się kolejka synchronizacji używana przez kanał.

Dodawanie menedżera kolejek do grupy współużytkowania kolejek

Menedżer kolejek można dodać do istniejącej grupy współużytkowania kolejek.

Należy pamiętać, że:

- Grupa współużytkowania kolejek musi istnieć przed dodaniem do niej menedżerów kolejek.
- Menedżer kolejek może być elementem tylko jednej grupy współużytkowania kolejek.

Aby dodać menedżera kolejek do grupy współużytkowania kolejek, wykonaj następujące kroki:

1. Aby nadać odpowiedni dostęp do identyfikatorów użytkowników menedżera kolejek i inicjatora kanału, należy wykonać zadania opisane w sekcji Implementacja kontroli zabezpieczeń ESM dla grupy współużytkowania kolejek .
2. Jeśli grupa współużytkowania kolejek ma struktury CF skonfigurowane do przenoszenia danych do SMDS, wykonaj zadania opisane w sekcji konfigurowanie środowiska SMDS.
3. Zatrzymaj menedżer kolejek.
4. Użyj funkcji ADD QMGR programu narzędziowego grupy współużytkowania kolejek (CSQ5PQSG). Program ten jest opisany w programie narzędziowym grupy współużytkowania kolejek. Przykład znajduje się w pliku thlqual.SCSQPROC(CSQ45AQM).
5. Zmień moduł parametrów systemowych , aby dodać dane grupy współużytkowania kolejek:
 - a. Zmodyfikuj CSQ6SYSP , aby określić parametr QSGDATA. Więcej informacji na ten temat zawiera sekcja Używanie komendy CSQ6SYSP .
 - b. Złóż i powiąż moduł parametrów systemowych. Dla modułu ładującego można użyć innej nazwy.
 - c. Zmień proces uruchamiania, aby używał nowego modułu.

6. Skopiuj i dostosuj przykładowy element `thlqual.SCSQPROC(CSQ4INSS)`, który definiuje wymagane struktury CF i kolejki SYSTEM. Dodaj dostosowany element do definicji danych CSQINP2 w definicji JCL uruchamiania menedżera kolejek.
7. Zrestartuj menedżer kolejek przy użyciu modułu parametrów systemowych grupy współużytkowania kolejek.
8. Opcjonalnie można przeprowadzić migrację do profili zabezpieczeń z przedrostkiem nazwy grupy współużytkowania kolejek zamiast nazwy menedżera kolejek.
9. Jeśli kanały współużytkowane są używane na potrzeby połączeń z systemem QSG, należy utworzyć reguły uwierzytelniania kanału, które odzwierciedlają reguły w innych menedżerach kolejek w systemie QSG, aby umożliwić aplikacjom nawiązywanie połączeń z dowolnym menedżerem kolejek w systemie QSG.
10. Opcjonalnie wykonaj jedną z następujących czynności, aby umożliwić aplikacjom połączonym z menedżerem kolejek w grupie QSG umieszczanie komunikatów w kolejkach udostępnianych przez inne menedżery kolejek w grupie QSG:
 - Włącz kolejkowanie wewnątrz grupy, wydając komendę `ALTER QMGR IGQ (ENABLED)`.
 - Zdefiniuj kolejki transmisji i kanały dla innych menedżerów kolejek w QSG. Definiowanie kolejek transmisji o takiej samej nazwie jak nazwy docelowych menedżerów kolejek pozwala uniknąć konieczności definiowania kolejek zdalnych i aliasów menedżerów kolejek.

Uwaga: Aby dodać menedżer kolejek do istniejącej grupy współużytkowania kolejek zawierającej menedżery kolejek z uruchomionymi wcześniejszymi wersjami produktu IBM MQ, należy najpierw zastosować poprawkę PTF współistnienia dla najwyższej wersji produktu IBM MQ w grupie do każdej wcześniejszej wersji menedżera kolejek w grupie.

Usuwanie menedżera kolejek z grupy współużytkowania kolejek

Menedżer kolejek można usunąć z grupy współużytkowania kolejek tylko wtedy, gdy dzienniki menedżera kolejek nie są wymagane przez inny proces, a wszystkie zestawy SMDS należące do menedżera kolejek są puste.

Więcej informacji na ten temat zawiera sekcja [Usuwanie współużytkowanych zestawów danych komunikatów](#) oraz sekcja [DELETE CFSTRUCT](#).

Dzienniki są potrzebne, jeśli zawierają:

- Najnowsza kopia zapasowa jednej ze struktur aplikacji CF (Coupling Facility) używanych przez grupę współużytkowania kolejek
- Dane wymagane przez przyszły proces odtwarzania, to znaczy, że menedżer kolejek używał struktury odtwarzalnej od czasu opisanego przez wartość ostatniego przedziału czasu wykluczania kopii zapasowej.

Jeśli mają zastosowanie oba te punkty lub SMDS, którego właścicielem jest menedżer kolejek, zawiera komunikaty, nie można usunąć menedżera kolejek. Aby określić, które dzienniki menedżerów kolejek są potrzebne do 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 we współużytkowanych kolejkach.
2. Rozstrzygnij wszystkie wątpliwe jednostki pracy związane z tym menedżerem kolejek.
3. Sprawdź, czy w dowolnym zestawie SMDS należącym 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. Przed wykonaniem dalszych czynności liczba komunikatów zgłoszonych przez komendę `DISPLAY USAGE TYPE (SMDS)` powinna wynosić zero.

5. Zamknij poprawnie menedżer kolejek, używając komendy STOP QMGR MODE (QUIESCE).
6. Poczekaj na interwał co najmniej równoważny z wartością parametru EXCLINT, który zostanie podany w komendzie BACKUP CFSTRUCT w następnym kroku.
7. W innym menedżerze kolejek uruchom tworzenie kopii zapasowej 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 żadnych struktur CF, sprawdzając dane wyjściowe za pomocą komendy DISPLAY CFSTATUS (*) TYPE (BACKUP).
9. Użyj funkcji REMOVE QMGR programu narzędziowego CSQ5PQSG, aby usunąć menedżera kolejek z grupy współużytkowania kolejek. Program ten jest opisany w sekcji [Program narzędziowy grupy współużytkowania kolejek](#). Przykład znajduje się w pliku thlqual.SCSQPROC(CSQ45RQM).
10. Przed zrestartowaniem menedżera kolejek przywróć wartość domyślną parametru systemowego QSGDATA i ponownie utwórz moduł parametrów systemowych. Informacje na temat dostosowywania parametrów systemowych zawiera sekcja [Używanie komendy CSQ6SYSP](#).

Należy zauważyć, 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 dzienniki menedżera kolejek wymagane do odtwarzania nie będą sprawdzane pod kątem spójności. Tę operację należy wykonać tylko w przypadku usuwania grupy współużytkowania kolejek.

Usuwanie grupy współużytkowania kolejek z tabel Db2

Aby usunąć grupę współużytkowania kolejek z tabel Db2, należy użyć funkcji REMOVE QSG programu narzędziowego grupy współużytkowania kolejek (CSQ5PQSG). Program ten jest opisany w sekcji [Program narzędziowy grupy współużytkowania kolejek](#). Przykład znajduje się w pliku thlqual.SCSQPROC(CSQ45RQS).

Grupę współużytkowania kolejek można usunąć ze wspólnych tabel grup współużytkowania danych Db2 tylko po usunięciu wszystkich menedżerów kolejek z grupy współużytkowania kolejek (zgodnie z opisem w sekcji ["Usuwanie menedżera kolejek z grupy współużytkowania kolejek"](#) na stronie 505).

Gdy rekord grupy współużytkowania kolejek jest usuwany z tabeli administracyjnej grupy współużytkowania kolejek, wszystkie obiekty i informacje administracyjne związane z tą grupą współużytkowania kolejek są usuwane z innych tabel produktu IBM MQ Db2. Obejmuje to informacje o współużytkowanej kolejce i obiekcie grupy.

Sprawdzanie spójności definicji Db2

Problemy ze współużytkowanymi kolejkami w grupie współużytkowania kolejek mogą wystąpić, jeśli definicje obiektów Db2 z dowolnej przyczyny staną się niespójne.

Aby sprawdzić spójność definicji obiektów Db2 dla menedżerów kolejek, struktur CF i kolejek współużytkowanych, należy użyć funkcji VERIFY QSG programu narzędziowego grupy współużytkowania kolejek (CSQ5PQSG). Program ten 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 507
- ["Przenoszenie kolejek współużytkowanych"](#) na stronie 507
- ["Migrowanie kolejek niewspółużytkowanych do współużytkowanych kolejek"](#) na stronie 510
- [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 jednocześnie odtworzyć kilka struktur CF.

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 [508](#)), 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 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 przenosowaniu 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 508.

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 508 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 38 na stronie 509 przedstawia przykładowe zadanie przenoszenia kolejki z jednej struktury CF do innej.

```
//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) 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 38. 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 508). [Rysunek 39 na stronie 509](#) 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=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 QL(Queue_TO_MOVE) LIKE(TEMP_QUEUE) CFSTRUCT(NEW) QSGDISP(SHARED)
MOVE QLOCAL(TEMP_QUEUE) TOQLOCAL(Queue_TO_MOVE)
DELETE QL(TEMP_QUEUE)
/*
```

Rysunek 39. Przykładowe zadanie przenoszenia niewspółużytkowanej kolejki do kolejki współużytkowanej

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 508).

[Rysunek 40 na stronie 510](#) daje przykładowe zadanie do wykonania tego zadania.

```

//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) 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 40. 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 39 na stronie 509 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 tak, aby była kolejką współużytkowaną, typem indeksu musi być MSGID.

Jeśli kolejka jest pusta lub nie ma potrzeby przechowywania na nim komunikatów, migracja kolejki jest prostsza. Rysunek 41 na stronie 511 przedstawia przykładowe zadanie, które ma być używane w tych okolicznościach.

```
//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 *
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 41. 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 39](#) na stronie 509 lub [Rysunek 41](#) na stronie 511.
2. W przypadku kolejek o tej samej nazwie co istniejąca kolejka współużytkowana, przenieś komunikaty do kolejki współużytkowanej za pomocą komend przedstawionych w sekcji [Rysunek 42](#) na stronie 511.

```
MOVE QLOCAL(QUEUE_TO_MOVE) QSGDISP(QMGR) TOQLOCAL(QUEUE_TO_MOVE)
DELETE QLOCAL(QUEUE_TO_MOVE) QSGDISP(QMGR)
```

Rysunek 42. 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 zawieszane, 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 umożliwia 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 na 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 512](#)
- [“Usuwanie struktury narzędzia CF” na stronie 512](#)

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 odciążać 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 MQSET.

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.

Odtwarzanie i restartowanie w systemie z/OS

W tym temacie opisano mechanizmy odtwarzania i restartowania używane przez produkt IBM MQ.

restartowanieIBM 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 514](#)
- [“Restartowanie po nieprawidłowym zakończeniu” na stronie 514](#)

- [“Restartowanie, jeśli utracono zestawy stron” na stronie 514](#)
- [“Restartowanie, jeśli utracono zestawy danych dziennika” na stronie 514](#)
- [Restartowanie, jeśli utraciłeś 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 441](#).

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 441](#). 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 528](#).

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 497](#).

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 się upewnić, 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 współużytkowania kolejek w ośrodku zapasowym. (W przypadku alternatywnej procedury odtwarzania serwisu dla grupy współużytkowania kolejek należy zapoznać się z [“Odtwarzanie grupy współużytkowania kolejek w ośrodku alternatywnym”](#) na stronie 519).

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 niego najnowsze zarchiwizowane dane 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 516.
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 516), 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 DSN=MQM2.LOGCOPY1.DS01,COPY1
NEWLOG DSN=MQM2.LOGCOPY1.DS02,COPY1
NEWLOG DSN=MQM2.LOGCOPY1.DS03,COPY1
NEWLOG DSN=MQM2.LOGCOPY1.DS04,COPY1
NEWLOG DSN=MQM2.LOGCOPY2.DS01,COPY2
NEWLOG DSN=MQM2.LOGCOPY2.DS02,COPY2
NEWLOG DSN=MQM2.LOGCOPY2.DS03,COPY2
NEWLOG DSN=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 526. 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.

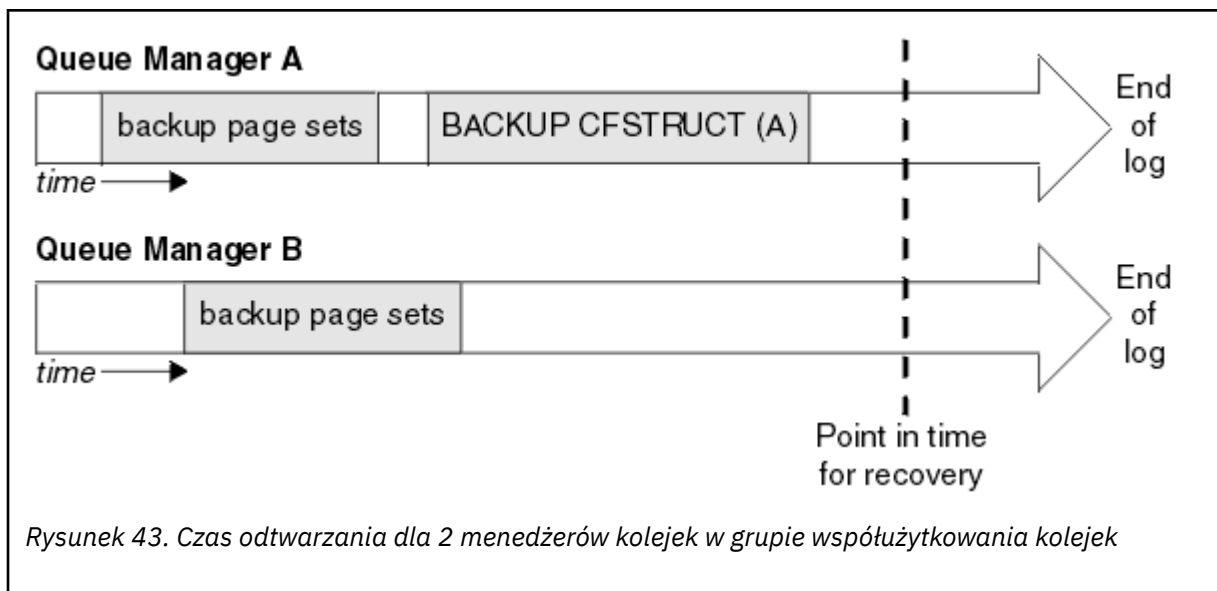
Kopia zapasowa struktury CF jest zapisywana w dzienniku menedżera kolejek, który przetworzył komendę BACKUP CFSTRUCT, dlatego nie ma żadnych dodatkowych zestawów danych, które mają być gromadzone i transportowane do ośrodka alternatywnego.

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 produkcie 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 przełączyć 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 kolejek. Spowoduje to odtworzenie 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 niego najnowsze zarchiwizowane dane 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 obcięty 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 520.
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 520) 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 520.

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 grupie 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, tak więc szybsze jest 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 administratorów równorzędnych, należy sprawdzić, czy elementy BSDS węzłów sieci zawierają te same wartości CRESTART LRSN, co ich 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.

V 9.2.0 IBM MQ 9.1.2, a następnie obsługuje zapis zHyperpodczas zapisu do aktywnych dzienników, których kopia lustrzana jest tworzona przy użyciu Metro Mirror. zHyperZapis może pomóc w zmniejszeniu

wydajności korzystania z funkcji Metro Mirror. Więcej informacji na ten temat zawiera sekcja [Korzystanie z funkcji Metro Mirror z IBM MQ](#).

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 522).

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 522 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 użyć ARM do automatycznego restartowania menedżerów kolejek.

Ta sekcja zawiera informacje na temat następujących tematów:

- [“Co to jest ARM?” na stronie 523](#)
- [“Strategie ARM” na stronie 524](#)
- [“Używanie ARM w sieci IBM MQ” na stronie 526](#)

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 526).

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 526.

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 44](#) na stronie 525 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 44. 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 w fazie zamykania, o ile nie wnioskował o to, aby nie.

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.

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 nastuchujące 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 526](#)
- [“Restartowanie na innym obrazie z/OS z jednostką logiczną 6.2” na stronie 528](#)

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 programu 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), odpowiada poprzez ponowne przydzielanie odpowiednich komunikatów w swojej 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, do której odwołuje się każdy obraz z/OS w obrębie syspleksu. 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 będzie zawierał 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 529](#)
- [“Ręczne odtwarzanie CICS jednostek odzyskiwania” na stronie 529](#)
- [“Ręczne odtwarzanie IMS jednostek odzyskiwania” na stronie 533](#)

- [“Ręczne odzyskiwanie jednostek RRS odzyskiwania” na stronie 535](#)
- [“Odzyskiwanie jednostek odtwarzania w innym menedżerze kolejek w grupie współużytkowania kolejek” na stronie 535](#)

Wyświetlanie połączeń i wątków

Za pomocą komendy `DISPLAY CONN` można uzyskać informacje na temat połączeń z menedżerami kolejek i powiązanymi 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

Ostrzeżenie, że identyfikatory UOW podświetlone znakiem * nie są rozstrzygane automatycznie. Te jednostki UOWs muszą zostać jawnie rozstrzygnięte przez rozproszony komponent kolejki po jego zrestartowaniu.

Rysunek 45 na stronie 530 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 45. 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 kolejki. 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 (C3E2D8C3C7D9F0F940404040404040)
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 zauważyć, 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 odtwarzania, 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  0000000000000000000000000000 CSQ:0001.0
USER2  0000000000000000000000000000 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.

z/OS IBM MQ i 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.

z/OS 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 536](#)
- [“Nawiązanie połączenia z regionu sterującego IMS” na stronie 537](#)
- [“Wyświetlanie w wątpliwych jednostkach odzyskiwania” na stronie 539](#)
- [“Sterowanie połączeniami regionu zależnego IMS” na stronie 541](#)
- [“odłączanie odIMS” na stronie 543](#)
- [“Sterowanie monitorem wyzwalacza IMS” na stronie 543](#)

z/OS 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.

Jeśli w dzienniku konsoli menedżera kolejek zostanie uruchomiona komenda IMS , podobna do tej komendy:

```
MODIFY IMS*,SS*
```

Sprawdź dziennik główny produktu IMS i upewnij się, że produkt IBM MQ ma uprawnienia RACF do wydawania komend MODIFY adaptera IMS .

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. Wydadaj 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(00000000000001F000000007472616E5F6964547565204E6F762020...)
QMURID(000000026232)
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 533.
- 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 produktu 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 odrzucany.

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 przyczyn .

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. Wydadź komendę IMS :

```
/STOP REGION
```

2. Zaktualizuj element SSM.

3. Wydadź 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 wysyła 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
INITQUEUEENAME=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 BMP monitora wyzwalacza. 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** .

Sterowanie mostem IMS

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 545](#)
- [“Sterowanie połączeniami produktu IMS” na stronie 546](#)
- [Sterowanie kolejkami mostu](#)
- [“Resynchronizacja mostu IMS” na stronie 547](#)
- [Praca z nazwami tpipe](#)
- [Usuwanie komunikatów z produktu IMS](#)
- [Usuwanie potoków](#)
- [“IMS Utrata ważności transakcji” na stronie 549](#)

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, IMS lub 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. Zsynchronizowany 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 . Za pomocą komendy IBM MQ RESET TPIPE zainicjuj 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, a także odpowiedzi dostarczanej 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 zostać 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 (AMS), należy użyć komendy z/OS MODIFY.

Na przykład składnia

```
F qmgrAMSM, cmd
```

gdzie *qmgr* jest przedrostkiem nazwy uruchomionego zadania.

Tabela 29 na stronie 549 zawiera opis komend MODIFY, które są akceptowane:

Tabela 29. Komendy MODIFY przestrzeni adresowej Advanced Message Security		
Komenda	Opcja	Opis
WYŚWIETL		Wyświetl informacje o wersji
ODŚWIEŻ	KEYRING strategia ALL	Odśwież certyfikaty kluczy, strategie bezpieczeństwa lub oba te certyfikaty.
KONTROLA SMFAUDIT	Powodzenie FAILURE ALL	Należy określić, czy kontrola SMF jest wymagana, gdy program AMS pomyślnie zabezpiecza lub zabezpiecza komunikaty, gdy program AMS nie chroni ani nie chroni komunikatów.

Tabela 29. Komendy MODIFY przestrzeni adresowej Advanced Message Security (kontynuacja)

Komenda	Opcja	Opis
TYP_MFP	0 - 255	Ustaw typ rekordu SMF, który ma być generowany, gdy program AMS zabezpiecza lub odchroni komunikaty. Aby wyłączyć kontrolę SMF, należy określić typ rekordu o wartości 0.

Uwaga: Aby określić opcję, musi ona być oddzielona przecinkiem. Na przykład:

```
F qmgrAMSM,REFRESH KEYRING
F qmgrAMSM,SMFAUDIT ALL
F qmgrAMSM,SMFTYPE 180
```

REFRESH, komenda

Zmiany wprowadzone za pomocą komendy **REFRESH** są stosowane do aplikacji, które wywołują komendę MQOPEN po zakończeniu działania komendy **REFRESH**. Istniejące aplikacje, które mają otwartą kolejkę, nadal korzystają z opcji, z których aplikacja otworzyła kolejkę. Aby użyć nowych wartości, aplikacja musi zamknąć i ponownie otworzyć kolejkę.

Uruchamianie i zatrzymywanie serwera AMS

Aby uruchomić lub zatrzymać przestrzeń adresową Advanced Message Security, nie trzeba wprowadzać komendy. Przestrzeń adresowa AMS jest uruchamiana automatycznie po uruchomieniu menedżera kolejek, jeśli włączono AMS z parametrem **SPLCAP** o wartości CSQ6SYSPi zatrzymano, gdy menedżer kolejek jest zatrzymany.

administrowanie IBM MQ Internet Pass-Thru

W tej sekcji opisano sposób administrowania produktem IBM MQ Internet Pass-Thru (MQIPT).




Skonfiguruj produkt MQIPT, wprowadzając zmiany w pliku konfiguracyjnym produktu `mqipt.conf` zgodnie z opisem w sekcji [Konfigurowanie produktu IBM MQ Internet Pass-Thru](#). Aby administrować produktem MQIPT, w tym odświeżanie produktu MQIPT w celu wprowadzenia zmian w konfiguracji bez konieczności restartowania produktu MQIPT, należy użyć komendy `mqiptAdmin`. Więcej informacji na temat administrowania produktem MQIPT przy użyciu komendy `mqiptAdmin` zawiera sekcja ["Administrowanie programem MQIPT przy użyciu wiersza komend"](#) na stronie 553.

Uruchamianie i zatrzymywanie serwera MQIPT

Program MQIPT można uruchomić z wiersza komend lub automatycznie podczas uruchamiania systemu. Serwer MQIPT można zatrzymać za pomocą komendy `mqiptAdmin`.

Uruchamianie programu MQIPT z wiersza komend

Produkt MQIPT jest instalowany w katalogu instalacyjnym, na przykład:

-  `C:\MQIPT` w systemach Windows ze skryptami wykonywalnymi w katalogu `C:\MQIPT\bin`
-   `/opt/mqipt` w systemach AIX and Linux ze skryptami wykonywalnymi w katalogu `/opt/mqipt/bin`

Produkt MQIPT używa również katalogu głównego, który zawiera plik konfiguracyjny `mqipt.conf` i wszystkie pliki, które są wyprowadzane przez program MQIPT podczas jego działania. Następujące podkatalogi katalogu głównego MQIPT są tworzone automatycznie podczas pierwszego wywołania funkcji MQIPT:

- Katalog `errors` , w którym zapisywane są wszystkie pliki First Failure Support Technology (FFST) i pliki śledzenia.
- Katalog `logs` , w którym przechowywany jest dziennik połączeń

ID użytkownika, dla którego uruchamiany jest program MQIPT , musi mieć uprawnienia do tworzenia tych katalogów lub katalogi muszą już istnieć, a ID użytkownika musi mieć uprawnienia do tworzenia, odczytywania i zapisywania w nich plików. Ponadto, jeśli używana jest strategia Java security manager , strategia bezpieczeństwa musi nadać wymagane uprawnienia do tych katalogów. Więcej informacji na temat ustawień strategii menedżera zabezpieczeń zawiera sekcja [Java security manager](#).

Jako katalogu osobistego można użyć katalogu instalacyjnego. Jeśli używany jest ten katalog, należy upewnić się, że ID użytkownika, z którego uruchamiany jest produkt MQIPT , ma odpowiednie uprawnienia i że każda strategia menedżera zabezpieczeń jest poprawnie skonfigurowana.

Aby uruchomić program MQIPT, należy użyć komendy `mqiPT` , która znajduje się w katalogu `bin` w katalogu instalacyjnym produktu MQIPT . Na przykład poniższa komenda uruchamia instancję MQIPT , która używa katalogu `C:\mqiptHome` jako katalogu osobistego:

```
mqiPT C:\mqiptHome
```

Więcej informacji na temat komendy `mqiPT` zawiera sekcja [mqiPT \(start MQIPT\)](#).

V 9.2.0 **V 9.2.0** Za pomocą komendy `mqiPT` można określić nazwę, która ma zostać nadana uruchamianej instancji MQIPT . Nazwa instancji MQIPT jest używana do administrowania lokalnymi instancjami serwera MQIPT za pomocą komendy `mqiPTAdmin` bez konieczności używania portu komendy. Jeśli ten parametr nie zostanie podany, jako nazwa instancji MQIPT zostanie użyta nazwa katalogu osobistego MQIPT .

Komunikaty konsoli pokazują status MQIPT. Jeśli wystąpi błąd, należy zapoznać się z sekcją [Rozwiązywanie problemów z produktem IBM MQ Internet Pass-Thru](#). Poniżej przedstawiono przykładowe dane wyjściowe po pomyślnym uruchomieniu komendy MQIPT :

```
5724-H72 (C) Copyright IBM Corp. 2000, 2024. All Rights Reserved
MQCPI001 IBM MQ Internet Pass-Thru V9.2.0.0 starting
MQCPI004 Reading configuration information from mqiPT.conf
MQCPI152 MQIPT name is C:\mqiptHome
MQCPI021 Password checking has been enabled on the command port
MQCPI144 MQ Advanced capabilities not enabled
MQCPI011 The path C:\mqiptHome\logs will be used to store the log files
MQCPI006 Route 1414 is starting and will forward messages to :
MQCPI034 ...examplehost(1414)
MQCPI035 ...using MQ protocols
MQCPI057 ...trace level 5 enabled
MQCPI078 Route 1414 ready for connection requests
```

Automatyczne uruchamianie programu MQIPT

Produkt MQIPT można zainstalować jako usługę systemową uruchamianą automatycznie podczas uruchamiania systemu. Użyj komendy `mqiPTService` , aby zainstalować i zdeinstalować usługę MQIPT .

- **Windows** W systemach Windows komenda `mqiPTService` instaluje MQIPT jako usługę systemu Windows .
- **Linux** **AIX** W systemach AIX and Linux komenda `mqiPTService` instaluje MQIPT jako usługę inicjowania systemu System V, która jest uruchamiana podczas startu systemu. W systemach Linux , które nie obsługują inicjowania systemu System V, należy użyć innej metody, na przykład metody `systemd`, aby zarządzać systemem MQIPT jako usługą.

Po uruchomieniu usługi MQIPT uruchamiane są wszystkie aktywne trasy MQIPT . Po zatrzymaniu usługi wszystkie trasy są natychmiast zamykane.

W systemie można zainstalować tylko jedną usługę systemu MQIPT , nawet jeśli w systemie istnieje więcej niż jedna instalacja produktu MQIPT .

Więcej informacji na temat komendy **mqiPTService** zawiera sekcja [mqiPTService \(zarządzanie usługą MQIPT\)](#).

ZatrzymywanieMQIPT

Serwer MQIPT można zatrzymać za pomocą komendy **mqiPTAdmin** z parametrem **-stop**.

▶ V 9.2.0 ▶ V 9.2.0 Na przykład następująca komenda zatrzymuje instancję bazy danych MQIPT o nazwie **mqiPT1**, która działa lokalnie z tym samym identyfikatorem użytkownika, co komenda **mqiPTAdmin**:

```
mqiPTAdmin -stop -n ipt1
```

Komenda **mqiPTAdmin** łączy się z aktywną instancją produktu MQIPT w celu administrowania przy użyciu jednej z następujących metod:

- ▶ V 9.2.0 ▶ V 9.2.0 przez połączenie z lokalną instancją MQIPT bez użycia portu komend.
- przez nawiązanie połączenia sieciowego z portem komend.

Zdalne zamknięcie systemu musi zostać włączone przez ustawienie właściwości **RemoteShutDown** na wartość **true**, aby komenda **mqiPTAdmin** mogła zostać użyta do zatrzymania serwera MQIPT przez wysłanie komendy do portu komendy.

Więcej informacji na temat administrowania programem MQIPT za pomocą komendy **mqiPTAdmin** zawiera sekcja [“Administrowanie programem MQIPT przy użyciu wiersza komend”](#) na stronie 553.

▶ V 9.2.0 Określanie klucza szyfrowania hasła

W przypadku produktu IBM MQ 9.1.5, jeśli konfiguracja produktu MQIPT zawiera hasła szyfrowane przy użyciu klucza szyfrowania innego niż klucz domyślny, należy podać klucz szyfrowania hasła w pliku, który może być odczytywany przez program MQIPT podczas jego uruchamiania.

Plik kluczy szyfrowania hasła

Hasła zaszyfrowane, które mają być przechowywane i używane przez produkt MQIPT, mogą być szyfrowane przy użyciu klucza szyfrowania udostępnianego przez użytkownika. Jeśli klucz szyfrowania nie zostanie użyty, zostanie użyty domyślny klucz szyfrowania. Nie trzeba określać klucza szyfrowania hasła, jednak jest on bardziej bezpieczny. Jeśli nie zostanie określony własny klucz szyfrowania, zostanie użyty domyślny klucz szyfrowania.

Jeśli zostanie użyty klucz szyfrowania hasła, musi on być zapisany w pliku, do którego można uzyskać dostęp za pomocą komendy **mqiPTPW** używanej do szyfrowania haseł i MQIPT. Jedynymi ograniczeniami dotyczącymi zawartości pliku jest to, że musi on zawierać co najmniej jeden znak i tylko jeden wiersz tekstu.

Uwaga: Należy upewnić się, że w pliku kluczy szyfrowania hasła ustawione są odpowiednie uprawnienia, aby uniemożliwić innym nieautoryzowanym użytkownikom odczytywanie klucza szyfrowania. Tylko użytkownik, który uruchamia komendę **mqiPTPW**, oraz użytkownik, w ramach którego program MQIPT uruchamia uprawnienia do odczytu klucza szyfrowania hasła.

Ten sam klucz szyfrowania hasła jest używany do szyfrowania i deszyfrowania wszystkich zapisanych haseł dla instancji serwera MQIPT. Oznacza to, że dla każdej instalacji produktu MQIPT wymagany jest tylko jeden plik kluczy szyfrowania haseł.

Jeśli klucz szyfrowania hasła dla instalacji produktu MQIPT zostanie zmieniony, wszystkie zaszyfrowane hasła muszą zostać ponownie zaszyfrowane przy użyciu nowego klucza szyfrowania.

UruchamianieMQIPT

Domyślną nazwą pliku kluczy szyfrowania hasła jest **MQIPT_HOME_DIR/mqiPT_cred.key**, gdzie **MQIPT_HOME_DIR** to katalog, w którym zapisany jest plik konfiguracyjny **mqiPT.conf**. Jeśli planowane

jest uruchomienie produktu MQIPT jako usługi, która jest uruchamiana automatycznie, należy utworzyć plik kluczy szyfrowania hasła z nazwą domyślną.

Jeśli plik kluczy szyfrowania hasła został utworzony z nazwą inną niż domyślna, nazwa pliku musi być podana w programie MQIPT po jego uruchomieniu. Nazwa pliku kluczy szyfrowania hasła może być określona przy użyciu dowolnej z poniższych metod, w kolejności:

1. Parametr **-sf** w komendzie **mqi**pt użytej do uruchomienia MQIPT.
2. Zmienna środowiskowa MQS_MQIPTCRED_KEYFILE .
3. Właściwość `com.ibm.mq.ipc.cred.keyfile` Java .

Jeśli nie zostanie podana nazwa pliku kluczy szyfrowania hasła, zostanie użyta domyślna nazwa pliku, jeśli plik istnieje. Jeśli domyślny plik kluczy szyfrowania hasła nie istnieje, używany jest domyślny klucz szyfrowania hasła.

Administrowanie programem MQIPT przy użyciu wiersza komend


Aby administrować programem MQIPT, można użyć komendy **mqi**ptAdmin w wierszu komend.

Za pomocą komendy **mqi**ptAdmin można wykonywać następujące funkcje administracyjne:

- Wyświetla listę aktywnych instancji lokalnych produktu MQIPT.
- Odśwież instancję produktu MQIPT po wprowadzeniu zmian w pliku konfiguracyjnym.
- Zatrzymaj instancję produktu MQIPT.

Komenda `mqi`ptAdmin znajduje się w podkatalogu `bin` w katalogu instalacyjnym MQIPT .

Komenda **mqi**ptAdmin łączy się z aktywną instancją produktu MQIPT , aby administrować przy użyciu jednej z następujących metod:

- przez nawiąże połączenie sieciowe z portem komend.
-  łącząc się z lokalną instancją produktu MQIPT bez używania portu komend.

Komenda **mqi**ptAdmin jest kompatybilna z poprzednimi wersjami produktu MQIPT, ale nie można użyć tej komendy do administrowania wersjami produktu MQIPT , które są wersją wyższą niż wersja komendy **mqi**ptAdmin . W środowisku, w którym znajdują się różne wersje produktu MQIPT, należy użyć najnowszej wersji komendy **mqi**ptAdmin .

Więcej informacji na temat składni komendy **mqi**ptAdmin można znaleźć w sekcji [mqi](#)ptAdmin (administrowanie programem MQIPT).

Administrowanie lokalne bez portu komend



Z poziomu produktu IBM MQ 9.2.0 lokalne instancje produktu MQIPT mogą być administrowane bez użycia portu komend. Administrowanie lokalne umożliwia administrowanie produktem MQIPT przy użyciu komendy **mqi**ptAdmin tylko wtedy, gdy jest ona uruchamiana w tym samym systemie, co instancja produktu MQIPT , która ma być administrować.

Aby produkt **mqi**ptAdmin mógł administrować lokalną instancją produktu MQIPT bez używania portu komend, instancja MQIPT musi być uruchomiona w tym samym systemie i pod tym samym identyfikatorem użytkownika co produkt **mqi**ptAdmin. Alternatywnie w systemie AIX and Linux **mqi**ptAdmin można uruchomić jako root.

Administrowanie lokalne jest domyślnie włączone. Aby wyłączyć administrowanie lokalne, należy użyć właściwości konfiguracyjnej **LocalAdmin** . Więcej informacji na temat właściwości **LocalAdmin** zawiera sekcja [LocalAdmin](#).

Aby administrować lokalnymi instancjami produktu MQIPT, należy nadać każdej instancji nazwę. Nazwę instancji programu MQIPT można przypisać za pomocą parametru **-n** przy uruchamianiu komendy MQIPT za pomocą komendy **mqi**pt . Jeśli podczas uruchamiania produktu MQIPT nie zostanie podana

nazwa, nazwa katalogu głównego będzie używana jako nazwa instancji produktu MQIPT . Na przykład następująca komenda uruchamia program MQIPT i przypisuje nazwę `ipt1` do instancji:

```
mqipt /opt/mqipt1 -n ipt1
```

Gdy instancja ma nazwę, można administrować tą instancją, podając jej nazwę w komendzie **mqiptAdmin** z parametrem **-n** . Na przykład następująca komenda zatrzymuje lokalną instancję produktu MQIPT o nazwie `ipt1`:

```
mqiptAdmin -stop -n ipt1
```

Za pomocą komendy **mqiptAdmin** z parametrem **-list** można wyświetlić listę wszystkich lokalnych aktywnych instancji produktu MQIPT , które są autoryzowane do administrowania komendą **mqiptAdmin** bez użycia portu komend. Na przykład poniższa komenda wyświetla listę wszystkich lokalnych aktywnych instancji produktu MQIPT , które użytkownik, który uruchomił komendę **mqiptAdmin** , ma uprawnienia do administrowania:

```
mqiptAdmin -list
```

Administrowanie przy użyciu portu komend



W programie IBM MQ 9.2.0 można skonfigurować produkt MQIPT z jednym portem komend, który jest niezabezpieczonym, a jednym z portów komend, które są zabezpieczone protokołem TLS. Tych portów komend można użyć do administrowania produktem MQIPT jako dowolnego użytkownika, który znajduje się w tym samym systemie, co instancja produktu MQIPT , która ma być administrować, lub z systemu zdalnego.

Poprzednie wersje produktu MQIPT akceptowano tylko komendy administracyjne wydane dla niezabezpieczonego portu komend.

Uwaga: Połączenia z niezabezpieczonym portem komend nie są szyfrowane, dlatego dane wysłane przez sieć do niezabezpieczonego portu komend, w tym hasło dostępu do produktu MQIPT , mogą być widoczne dla innych użytkowników w sieci.

Aby program MQIPT nasłuchiwać w porcie komend dla komend wydawanych przez komendę **mqiptAdmin** , należy określić wartość dla właściwości **CommandPort** lub **SSLCommandPort** w sekcji globalnej pliku konfiguracyjnego `mqipt.conf` .

Przed włączeniem portów komend produktu MQIPT należy zapoznać się z uwagami dotyczącymi bezpieczeństwa w sekcji [Inne zagadnienia dotyczące zabezpieczeń](#) . Rozważ włączenie uwierzytelniania dla komend odebranych przez porty komend. Więcej informacji na temat uwierzytelniania portu komend zawiera sekcja [“Uwierzytelnianie portu komend”](#) na stronie 558.

Aby administrować instancją produktu MQIPT przy użyciu portu komend, należy podać adres sieciowy hosta, na którym działa produkt MQIPT , a także numer portu komendy, jako parametry komendy **mqiptAdmin** . Aby na przykład odświeżyć instancję produktu MQIPT uruchomioną na serwerze `mqipt.example.com`, a port komendy niezabezpieczony został skonfigurowany do nasłuchiwania na porcie 1890, należy wydać następującą komendę:

```
mqiptAdmin -refresh -r mqipt.example.com:1890
```

Jeśli nazwa hosta i numer portu nie zostaną określone, program **mqiptAdmin** podejmie próbę nawiązania połączenia z serwerem `localhost`(port 1881).

Więcej informacji na temat administrowania produktem MQIPT przy użyciu portu komend TLS zawiera sekcja [“Administrowanie programem MQIPT przy użyciu portu komend TLS”](#) na stronie 555.

komend TLS

W produkcie IBM MQ 9.2.0 można skonfigurować produkt MQIPT w taki sposób, aby używany był port komend TLS do nasłuchiwania komend administracyjnych wydawanych przez komendę **mqiptAdmin**. Korzystanie z portu komend TLS chroni poufne dane, takie jak hasło dostępu MQIPT w sieci między **mqiptAdmin** i MQIPT. Ta procedura służy do konfigurowania portu komend TLS i administrowania produktem MQIPT za pomocą portu komend TLS.

O tym zadaniu

Port komendy TLS musi być skonfigurowany z certyfikatem serwera przechowywanego w pierścieniu kluczy PKCS #12 lub w sprzęcie szyfrującym, który obsługuje interfejs tokenu PKCS #11 Cryptographic Token Interface. Certyfikat serwera portu komend jest wysyłany do komendy **mqiptAdmin** podczas uzgadniania TLS. W przypadku tego zadania założono, że użytkownik zażądał nowego certyfikatu serwera z zaufanego ośrodka certyfikacji (CA) i że certyfikat jest zwracany do użytkownika w pliku. Komenda **mqiptAdmin** sprawdza poprawność certyfikatu portu komendy przy użyciu certyfikatu ośrodka CA, który podpisał certyfikat serwera. Certyfikat ośrodka CA musi być zapisany w pliku kluczy PKCS #12, do którego można uzyskać dostęp za pomocą komendy **mqiptAdmin**.

Uwierzytelnianie certyfikatu klienta nie jest obsługiwane przez port komendy TLS. Informacje na temat włączania uwierzytelniania dla komend administracyjnych wydanych dla portu komend zawiera sekcja [“Uwierzytelnianie portu komend”](#) na stronie 558.

W tej procedurze opisano sposób zarządzania pierścieniami kluczy i certyfikatami cyfrowymi, które są wymagane do korzystania z portu komend TLS przy użyciu interfejsu wiersza komend (CLI) programu **mqiptKeycmd** (iKeyman). Komendy interfejsu CLI można użyć za pomocą komendy **mqiptKeycmd**. Więcej informacji na temat innych komend, które mogą być używane do zarządzania kluczowymi pierścieniami i certyfikatami cyfrowymi, można znaleźć w sekcji [mqiptKeyman](#) i [mqiptKeycmd](#).

Procedura

1. Wykonaj poniższe kroki, aby skonfigurować port komend TLS dla instancji serwera MQIPT.
 - a) Utwórz plik kluczy PKCS #12, który ma być używany przez port komendy TLS. Pierścień kluczy jest używany do przechowywania certyfikatu serwera portów komend TLS.

Aby utworzyć plik kluczy przy użyciu interfejsu CLI, wprowadź następującą komendę:

```
mqiptKeycmd -keydb -create -db filename -pw password -type pkcs12
```

gdzie *nazwa_pliku* jest nazwą pliku kluczy, który ma zostać utworzony, a *hasło* jest hasłem pliku kluczy.

- b) Utwórz żądanie certyfikatu dla certyfikatu serwera portów komend podpisanego przez ośrodek CA. Aby utworzyć żądanie certyfikatu przy użyciu interfejsu CLI iKeyman, wprowadź następującą komendę:

```
mqiptKeycmd -certreq -create -db filename -pw password  
-label label -size key_size -sig_alg algorithm  
-dn distinguished_name -file certreq_filename -type pkcs12
```

gdzie:

-db *nazwa_pliku*

Określa nazwę pliku kluczy.

-pw *hasło*

Określa hasło do pliku kluczy.

-label *etykieta*

Określa etykiety certyfikatu.

-size wielkość_klucza

Określa wielkość klucza.

-sig_alg algorytm

Określa algorytm podpisu asymetrycznego używany do tworzenia pary kluczy pozycji.

-dn nazwa_wyróżniająca

Określa nazwę wyróżniającą X.500 ujętą w podwójny cudzysłów.

-file nazwa_pliku_certreq_cert

Określa nazwę pliku dla żądania certyfikatu.

- c) Wyślij plik żądania certyfikatu utworzony w kroku [“1.b”](#) na stronie 555 do swojego ośrodka CA, który ma zostać podpisany.

- d) Po wysłaniu przez ośrodek CA podpisanego certyfikatu, należy odebrać podpisany certyfikat do pliku kluczy.

Aby odebrać podpisany certyfikat do pliku kluczy przy użyciu interfejsu CLI, wprowadź następującą komendę:

```
mqiPTKeycmd -cert -receive -file cert_filename -db filename  
-pw password -type pkcs12
```

gdzie *nazwa_pliku_certyfikatu* to nazwa pliku zawierającego certyfikat, *nazwa_pliku* jest nazwą pliku kluczy, a *hasło* jest hasłem pliku kluczy.

- e) Szyfruj hasło pliku kluczy za pomocą komendy **mqiPTPW**.

Wprowadź następującą komendę:

```
mqiPTPW -sf encryption_key_file
```

gdzie *plik_kluczy_szyfrowania* to nazwa pliku zawierającego klucz szyfrowania hasła dla instalacji produktu MQIPT. Nie ma potrzeby określania parametru **-sf**, jeśli podczas instalacji produktu MQIPT jest używany domyślny klucz szyfrowania hasła. Wpisz hasło do pliku kluczy w celu zaszyfrowania po wyświetleniu zachęty.

Więcej informacji na temat komendy **mqiPTPW** zawiera sekcja [Encrypting a key ring password](#) (Szyfrowanie hasła pliku kluczy).

- f) Zmodyfikuj plik konfiguracyjny `mqiPT.conf` i podaj następujące właściwości, aby skonfigurować port komendy TLS:

i) Ustaw wartość właściwości **SSLCommandPort** na numer portu komendy TLS.

ii) Ustaw wartość właściwości **SSLCommandPortKeyRing** na nazwę pliku pliku kluczy utworzonego w kroku [“1.a”](#) na stronie 555.

iii) Ustaw wartość parametru **SSLCommandPortKeyRingPW** na dane wyjściowe łańcucha komendy **mqiPTPW** w kroku [“1.e”](#) na stronie 556.

iv) Ustaw wartość właściwości **SSLCommandPortSiteLabel** na nazwę etykiety certyfikatu portu komendy TLS, określonego podczas tworzenia żądania certyfikatu w kroku [“1.b”](#) na stronie 555.

v) Jeśli chcesz ograniczyć połączenia przychodzące do portu komendy TLS do tych z określonego interfejsu sieciowego, ustaw wartość właściwości **SSLCommandPortListenerAddress** na adres sieciowy należący do jednego z interfejsów sieciowych w systemie, w którym działa produkt MQIPT. Na przykład, aby ograniczyć połączenia przychodzące do portu komendy TLS do portów tylko z komputera lokalnego, należy ustawić wartość właściwości **SSLCommandPortListenerAddress** na `localhost`.

- g) Uruchom lub odśwież MQIPT, aby włączyć port komendy TLS.

Program MQIPT wysyła komunikaty konsoli, takie jak następujące informacje, aby wyświetlić konfigurację portu komendy TLS, która jest w stanie:

```
MQCPI155 Listening for control commands on port 1882 on local address * using TLS  
MQCPI139 .....secure socket protocols <NULL>  
MQCPI031 .....cipher suites <NULL>
```

```
MQCPI032 .....key ring file c:\\iptHome\\ssl\\commandport.p12
MQCPI072 .....and certificate label mqiptadmin
```

2. W systemie, w którym do administrowania produktem MQIPTużywana jest komenda **mqiptAdmin**, wykonaj następujące kroki, aby umożliwić **mqiptAdmin** nawiązanie połączenia z portem komend TLS.

- a) Utwórz pierścień kluczy PKCS #12, który ma być używany jako magazyn zaufanych certyfikatów za pomocą komendy **mqiptAdmin**.

Aby utworzyć plik kluczy przy użyciu interfejsu CLI, wprowadź następującą komendę:

```
mqiptKeycmd -keydb -create -db filename -pw password -type pkcs12
```

gdzie *nazwa_pliku* jest nazwą pliku kluczy, który ma zostać utworzony, a *hasło* jest hasłem pliku kluczy.

- b) Zaimportuj certyfikat CA ośrodka CA, który podpisał certyfikat portu komendy TLS, do pliku kluczy utworzonego w kroku [“2.a” na stronie 557](#).

Aby zaimportować certyfikat ośrodka CA za pomocą interfejsu CLI iKeyman, wprowadź następującą komendę:

```
mqiptKeycmd -cert -add -db filename -pw password -type pkcs12
-label certlabel -file cert_filename
```

gdzie:

Nazwa pliku

Określa nazwę pliku kluczy

Hasło

Określa hasło do pliku kluczy.

certlabel

Określa etykietę, która ma zostać nadana certyfikatowi CA.

cert_filename

Określa nazwę pliku zawierającego certyfikat ośrodka CA.

- c) Szyfruj hasło pliku kluczy za pomocą komendy **mqiptPW**.

Wprowadź następującą komendę:

```
mqiptPW -sf encryption_key_file
```

gdzie *plik_kluczy_szyfrowania* to nazwa pliku zawierającego klucz szyfrowania hasła. Plik kluczy szyfrowania hasła może być inny niż plik używany do szyfrowania haseł w konfiguracji produktu MQIPT. Domyślny klucz szyfrowania hasła jest używany, jeśli nie zostanie określony plik kluczy szyfrowania z parametrem **-sf**. Wpisz hasło do pliku kluczy w celu zaszyfrowania po wyświetleniu zachęty.

Więcej informacji na temat komendy **mqiptPW** zawiera sekcja [Encrypting a key ring password](#)(Szyfrowanie hasła pliku kluczy).

- d) Utwórz plik właściwości, który ma być używany przez komendę **mqiptAdmin**, i określ następujące właściwości:

```
SSLClientCAKeyRing=key_ring_file_name
SSLClientCAKeyRingPW=key_ring_password
PasswordProtectionKeyFile=encryption_key_file
```

gdzie:

nazwa_pliku_kluczy

Jest to nazwa pliku kluczy utworzonego w kroku [“2.a” na stronie 557](#).

hasło_kluczy_klucza

to dane wyjściowe zaszyfrowanego hasła są zapisywane przez komendę **mqiptPW** w kroku [“2.c” na stronie 557](#).

encryption_key_file

jest nazwą pliku, który zawiera klucz szyfrowania hasła. Właściwość

PasswordProtectionKeyFile należy określić tylko wtedy, gdy plik kluczy szyfrowania był używany do szyfrowania hasła pliku kluczy w kroku "2.c" na stronie 557.

- e) Wydadź komendę **mqiPTAdmin**, aby administrować produktem MQIPT, określając parametr **-s** w celu wskazania, że połączenie TLS jest wymagane, oraz parametr **-p**, aby określić nazwę pliku właściwości, który został utworzony w kroku "2.d" na stronie 557.

Na przykład można wprowadzić następującą komendę, aby odświeżyć instancję produktu MQIPT, wysyłając komendę odświeżania do portu komend TLS:

```
mqiPTAdmin -refresh -r hostname:port -s -p properties_file
```

Komenda **mqiPTAdmin** wysyła komunikat, taki jak następujący po tym, aby potwierdzić, że połączenie z serwerem MQIPT jest chronione protokołem TLS:

```
MQCAI109 The connection to MQIPT is secured with TLSv1.2.
```

Co dalej

Aby włączyć uwierzytelnianie dla komend odebranych przez port komendy TLS, należy wykonać kroki opisane w sekcji "Uwierzytelnianie portu komend" na stronie 558.

Uwierzytelnianie portu komend

Produkt MQIPT można skonfigurować do uwierzytelniania komend odebranych przez niezabezpieczony port komend i port komendy TLS przy użyciu hasła. Ta procedura służy do włączania uwierzytelniania portu komend.

O tym zadaniu

Komenda **mqiPTAdmin** wyświetla zachętę do wprowadzenia hasła, gdy komenda łączy się z portem komend instancji serwera MQIPT, w której włączono uwierzytelnianie portu komend. Program MQIPT sprawdza poprawność hasła wprowadzonego w komendzie **mqiPTAdmin** w odniesieniu do hasła dostępu określonego w konfiguracji produktu MQIPT.

Właściwości ustawione na potrzeby uwierzytelniania portu komend mają zastosowanie zarówno do portu komend TLS, jak i do niezabezpieczonego portu komend.

Procedura

1. Zszyfruj hasło dostępu MQIPT za pomocą komendy **mqiPTPW**.

Wprowadź następującą komendę:

```
mqiPTPW -sf encryption_key_file
```

gdzie *plik_kluczy_szyfrowania* to nazwa pliku zawierającego klucz szyfrowania hasła dla instalacji produktu MQIPT. Nie ma potrzeby określania parametru **-sf**, jeśli podczas instalacji produktu MQIPT jest używany domyślny klucz szyfrowania hasła. Wpisz hasło dostępu, aby zaszyfrować po wyświetleniu zapytania.

Więcej informacji na temat szyfrowania hasel w konfiguracji produktu MQIPT zawiera sekcja [Encrypting zapisanego hasła](#).

2. Zmodyfikuj plik konfiguracyjny produktu `mqiPT.conf` i określ następujące właściwości:

```
AccessPW=encrypted_password  
RemoteCommandAuthentication=auth_setting
```

gdzie:

zaszyfrowane_hasło

to dane wyjściowe zaszyfrowanego hasła są zapisywane przez komendę **mqiPTPW** w kroku “1” na stronie 558.

Ustawienie autoryzacji

jest wymaganiem uwierzytelniania. Uwierzytelnianie portu komend jest włączone, jeśli ta właściwość jest ustawiona na jedną z następujących wartości:

opcjonalne

Hasło nie jest wymagane, ale jeśli zostało podane hasło, musi ono być poprawne. Ta opcja może być użyteczna podczas migracji, np.

wymagane

Dla każdej komendy otrzymanej przez port komendy należy podać poprawne hasło.

Więcej informacji na temat tych właściwości można znaleźć w sekcji Właściwości globalne produktu MQIPT.

3. Uruchom lub odśwież MQIPT , aby wprowadzone zmiany zostały wprowadzone w życie.

Program MQIPT wysyła komunikat wskazujący, czy włączone jest uwierzytelnianie portu komend. Jeśli na przykład produkt MQIPT jest skonfigurowany tak, aby wymagał wprowadzenia poprawnego hasła przy każdym uruchomieniu komendy **mqiPTAdmin** , zostanie wyświetlony następujący komunikat:

```
MQCPI021 Password checking has been enabled on the command port
```

Tworzenie kopii zapasowych

Istnieje wiele plików MQIPT , które należy utworzyć w ramach zwykłych procedur tworzenia kopii zapasowych.

Utwórz kopię zapasową następujących plików regularnie:

- Plik konfiguracyjny, **mqiPT.conf**
- Pliki kluczy SSL/TLS określone za pomocą następujących właściwości w produkcie **mqiPT.conf**:
 - **SSLClientKeyRing**
 - **SSLClientCAKeyRing**
 - **SSLServerKeyRing**
 - **SSLServerCAKeyRing**
 - **V9.2.0 V9.2.0 SSLCommandPortKeyRing**
- Pliki haseł kluczy SSL/TLS określone za pomocą następujących właściwości w produkcie **mqiPT.conf**:
 - **SSLClientKeyRingPW**
 - **SSLClientCAKeyRingPW**
 - **SSLServerKeyRingPW**
 - **SSLServerCAKeyRingPW**
- **V9.2.0** Plik kluczy szyfrowania hasła, jeśli konfiguracja produktu MQIPT zawiera hasła zaszyfrowane za pomocą klucza szyfrowania innego niż klucz domyślny.
- Plik strategii określony przez produkt **SecurityManagerPolicy**, jeśli ta właściwość została ustawiona.
- Pliki wyjścia zabezpieczeń i pliki wyjścia certyfikatów określone przez następujące właściwości w programie **mqiPT.conf**:
 - **SecurityExitName**
 - **SSLExitName**
- Pliki dzienników połączeń znajdują się w podkatalogu **Log** katalogu głównego produktu MQIPT , jeśli są one potrzebne do celów kontroli.

Dostrajanie wydajności

Wydajność każdej trasy MQIPT można dostroić, korzystając z kombinacji puli wątków i specyfikacji limitu czasu bezczynności.

Wątki połączenia

Każda trasa MQIPT jest przypisywany do puli roboczej współbieżnie działających wątków, które obsługują przychodzące żądania komunikacji. Podczas inicjowania tworzona jest pula wątków (o wielkości określonej w atrybucie trasy `MinConnectionThreads` trasy), a wątek jest przypisany do obsługi pierwszego przychodzącego żądania. Po nadejściu tego żądania przypisany jest inny wątek, gotowy do następnego przychodzącego żądania. Gdy wszystkie wątki są przypisane do pracy, tworzony jest nowy wątek, dodawany do puli roboczej i przypisany do pracy.

W ten sposób pula rośnie aż do osiągnięcia maksymalnej liczby wątków (określonych w `MaxConnectionThreads`). Wątki są zwalniane z powrotem do puli po zakończeniu konwersacji lub upływie podany okres czasu bezczynności. Gdy zostanie osiągnięta maksymalna liczba wątków roboczych, następnę żądanie przychodzące oczekuje na zwolnienie wątku z powrotem do puli roboczej.

Można skrócić czas oczekiwania żądań przez zwiększenie liczby dostępnych wątków. Należy jednak zrównoważyć ten wzrost przy użyciu dostępnych zasobów systemowych.

Idle timeout (Limit czasu bezczynności)

Domyślnie wątki robocze nie są przerywane z powodu braku aktywności. Gdy wątek został przypisany do konwersacji, pozostaje on przypisany do tej konwersacji do czasu, aż zostanie zamknięty normalnie, trasa jest dezaktywowana lub MQIPT jest wyłączany. Opcjonalnie można określić limit czasu bezczynności (w minutach) we właściwości `IdleTimeout`, tak aby wątki, które były nieaktywne przez określony czas, zostały poddane recyklingowi. Wątki są poddawane recyklingowi w celu ich użycia, umieszczając je z powrotem w puli roboczej.

Jeśli działanie IBM MQ jest przerywane, należy ustawić interwał pulsu na wartość mniejszą niż wartość limitu czasu MQIPT, aby wątki nie były stale poddawane recyklingowi.

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 (<https://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: