

9.2

Przegląd techniczny produktu IBM MQ

IBM

Uwaga

Przed skorzystaniem z niniejszych informacji oraz produktu, którego one dotyczą, należy zapoznać się z informacjami zamieszczonymi w sekcji [“Uwagi” na stronie 319](#).

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

| | |
|--|----------|
| Przegląd techniczny..... | 5 |
| Wprowadzenie do kolejkowania komunikatów..... | 5 |
| Główne cechy i korzyści związane z kolejkowaniem komunikatów..... | 7 |
| Terminologia kolejkowania komunikatów..... | 9 |
| Komunikaty i kolejki..... | 12 |
| Obiekty IBM MQ..... | 14 |
| Typy obiektów..... | 15 |
| Nazywanie obiektów IBM MQ..... | 36 |
| Atrybuty obiektu..... | 42 |
| Grupy współużytkowania kolejek..... | 42 |
| Domyślne obiekty systemowe..... | 43 |
| Rozproszone kolejkowanie i klastry..... | 43 |
| Rozproszone komponenty kolejkowania..... | 47 |
| Komponenty klastra..... | 57 |
| Przesyłanie komunikatów publikowania/subskrypcji..... | 63 |
| Komponenty publikowania/subskrybowania..... | 64 |
| Przykład konfiguracji publikowania/subskrypcji pojedynczego menedżera kolejek..... | 92 |
| Rozproszone sieci publikowania/subskrypcji..... | 92 |
| IBM MQ Rozsyłanie grupowe..... | 111 |
| Wstępne koncepcje rozsyłania grupowego..... | 111 |
| MQ TelemetryPrzegląd..... | 112 |
| MQ Telemetry - wprowadzenie..... | 114 |
| Przypadki użycia telemetrii..... | 115 |
| Podłączanie urządzeń telemetrycznych do menedżera kolejek..... | 122 |
| Protokoły połączeń telemetrycznych..... | 123 |
| Usługa telemetryczna (MQXR)..... | 123 |
| Kanały pomiarowe..... | 123 |
| IBM MQ Telemetry Transport :NONE..... | 124 |
| MQTT klienci..... | 124 |
| Wysyłanie komunikatu do klienta MQTT..... | 124 |
| Wysyłanie wiadomości do aplikacji IBM MQ z klienta MQTT..... | 134 |
| Aplikacje publikowania/subskrypcji produktu MQTT..... | 135 |
| Aplikacje telemetryczne..... | 135 |
| Integracja produktu MQ Telemetry z menedżerami kolejek..... | 136 |
| MQTT -bezstanowe i stanowe sesje..... | 139 |
| Gdy klient MQTT nie jest połączony..... | 139 |
| Luźne sprzężenie między klientami MQTT a aplikacjami produktu IBM MQ..... | 139 |
| Zabezpieczenia produktu MQ Telemetry..... | 140 |
| MQ Telemetry globalizacja..... | 141 |
| Wydajność i skalowalność produktu MQ Telemetry..... | 141 |
| Urządzenia obsługiwane przez produkt MQ Telemetry..... | 144 |
| Zabezpieczenia w produkcie IBM MQ..... | 144 |
| Obsługa protokołu TLS klienta zarządzanego przez program IBM MQ.NET..... | 145 |
| Klienci i serwery..... | 146 |
| Przegląd produktu IBM MQ MQI clients..... | 147 |
| Zarządzanie transakcjami i obsługa..... | 153 |
| Rozszerzanie obiektów menedżera kolejek..... | 155 |
| Interfejsy językowe programu IBM MQ Java..... | 156 |
| IBM MQ classes for JMS..... | 157 |
| IBM MQ Dostawca komunikatów..... | 164 |
| IBM MQ for z/OS pojęcia..... | 165 |
| Menedżer kolejek w systemie z/OS..... | 166 |

| | |
|---|------------|
| Inicjator kanału w systemie z/OS..... | 167 |
| Terminy i zadania związane z zarządzaniem IBM MQ for z/OS..... | 169 |
| Współużytkowane kolejki i grupy współużytkowania kolejek..... | 172 |
| Kolejkowanie wewnątrz grupy..... | 221 |
| Zarządzanie pamięcią masową w systemie z/OS..... | 234 |
| logowanieIBM MQ for z/OS..... | 239 |
| Definiowanie systemu w systemie z/OS..... | 250 |
| Odtwarzanie i restartowanie w systemie z/OS..... | 262 |
| Pojęcia dotyczące zabezpieczeń w produkcie IBM MQ for z/OS..... | 280 |
| Dostępność w systemie z/OS..... | 287 |
| Monitorowanie i statystyka w systemie IBM MQ for z/OS..... | 290 |
| Dyspozycja jednostki odtwarzania w systemie z/OS..... | 291 |
| IBM MQ i inne produkty z/OS..... | 294 |
| IBM MQ i CICS..... | 295 |
| IBM MQ i IMS..... | 296 |
| Adaptery IBM MQ i z/OS Batch, TSO i RRS..... | 300 |
| IBM MQ for z/OS i WebSphere Application Server..... | 301 |
| Managed File Transfer..... | 302 |
| W jaki sposób produkt MFT działa z produktem IBM MQ?..... | 305 |
| MFT przegląd topologii..... | 305 |
| Przegląd produktu MFTREST API..... | 307 |
| IBM MQ Internet Pass-Thru..... | 307 |
| Użycie produktu MQIPT..... | 308 |
| Jak działa MQIPT..... | 310 |
| Możliwe konfiguracje produktu MQIPT..... | 311 |
| Kompatybilne konfiguracje..... | 313 |
| Obsługiwane konfiguracje kanałów..... | 315 |
| Warunki zakończenia kanału i niepowodzenia..... | 316 |
| Bezpieczeństwo komunikatów..... | 316 |
| Menedżery kolejek o wielu instancjach i wysoka dostępność..... | 316 |
| Uwagi..... | 319 |
| Informacje dotyczące interfejsu programistycznego..... | 320 |
| Znaki towarowe..... | 321 |

IBM MQ Przegląd techniczny

Produkt IBM MQ służy do łączenia aplikacji i zarządzania dystrybucją informacji w organizacji.

Program IBM MQ umożliwia programom komunikowanie się między różnymi komponentami (procesorami, systemami operacyjnymi, podsystemami i protokołami komunikacyjnymi) za pomocą spójnego interfejsu programistycznego aplikacji. Aplikacje zaprojektowane i napisane przy użyciu tego interfejsu są nazywane aplikacjami kolejkowania komunikatów.

W poniższych podtematach można znaleźć informacje na temat kolejkowania komunikatów i innych funkcji udostępnianych przez produkt IBM MQ.

Pojęcia pokrewne

[IBM MQ - wprowadzenie](#)

Zadania pokrewne

[Planowanie architektury produktu IBM MQ](#)

Odsyłacze pokrewne

[“Główne cechy i korzyści związane z kolejkowaniem komunikatów” na stronie 7](#)

Informacje te wyróżniają niektóre funkcje i korzyści wynikające z kolejkowania komunikatów. Opisano w nim takie funkcje, jak bezpieczeństwo i integralność danych w kolejkowaniu komunikatów.

[Gdzie znaleźć wymagania dotyczące produktu i informacje o wsparciu](#)

Wprowadzenie do kolejkowania komunikatów

Produkty IBM MQ umożliwiają programom komunikowanie się między różnymi komponentami (procesorami, systemami operacyjnymi, podsystemami i protokołami komunikacyjnymi) za pomocą spójnego interfejsu programistycznego aplikacji.

Aplikacje zaprojektowane i napisane przy użyciu tego interfejsu są nazywane aplikacjami *kolejkowania komunikatów*, ponieważ używają stylu *messaging* i *queuing*:

- Przesyłanie komunikatów oznacza, że programy komunikują się, wysyłając wszystkie inne dane w komunikatach, zamiast kontaktować się bezpośrednio ze sobą.
- Kolejkowanie oznacza, że komunikaty są umieszczane w kolejkach w pamięci masowej, co umożliwia uruchamianie programów niezależnie od siebie, przy różnych prędkościach i godzinach, w różnych lokalizacjach i bez logicznego połączenia między nimi.

Kolejkowanie komunikatów jest używane w przetwarzaniu danych przez wiele lat. Najczęściej używa się go dziś w mailach elektronicznych. W przypadku braku kolejkowania wysłanie komunikatu elektronicznego na duże odległości wymaga, aby każdy węzeł na trasie był dostępny do przesyłania komunikatów, a adresaci byli zalogowani i świadomi faktu, że użytkownik próbuje wysłać je do wiadomości. W systemie kolejkowania komunikaty są zapisywane w węzłach pośrednich do czasu, aż system będzie gotowy do ich przekazania. W ich ostatecznym miejscu przeznaczenia są one przechowywane w elektronicznej skrzynce pocztowej do momentu, gdy adresat będzie gotowy do ich odczytania.

Nawet tak, wiele złożonych transakcji biznesowych jest przetwarzanych dzisiaj bez kolejkowania. W dużej sieci system może utrzymywać wiele tysięcy połączeń w stanie gotowym do użycia. Jeśli jedna część systemu cierpi na problem, wiele części systemu staje się bezużyteczne.

Można myśleć o kolejkowaniu komunikatów jako poczty elektronicznej dla programów. W środowisku kolejkowania komunikatów każdy program, który składa się z części pakietu aplikacji, wykonuje poprawnie zdefiniowaną, samodzielną funkcję w odpowiedzi na konkretne żądanie. Aby można było komunikować się z innym programem, program musi umieścić komunikat w predefiniowanej kolejce. Drugi program pobiera komunikat z kolejki i przetwarza żądania i informacje zawarte w komunikacie. Kolejkowanie komunikatów jest to styl komunikacji program-program.

Kolejkowanie to mechanizm, za pomocą którego komunikaty są wstrzymane do momentu, gdy aplikacja będzie gotowa do ich przetwarzania. Kolejkowanie umożliwia:

- Komunikacja między programami (które mogą być uruchamiane w różnych środowiskach) bez konieczności pisania kodu komunikacyjnego.
- Wybierz kolejność, w jakiej program przetwarza komunikaty.
- Równoważenie obciążenia systemu przez zorganizowanie więcej niż jednego programu w celu obsługi kolejki, gdy liczba komunikatów przekroczy próg.
- Zwiększenie dostępności aplikacji poprzez zorganizowanie alternatywnego systemu do obsługi kolejek, jeśli system podstawowy jest niedostępny.

Co to jest kolejka komunikatów?

Kolejka komunikatów, znana po prostu jako kolejka, jest nazwanym miejscem docelowym, do którego mogą być wysyłane komunikaty. Komunikaty gromadzą się w kolejkach do czasu ich pobrania przez programy, które służą do obsługi tych kolejek.

Kolejki znajdują się w menedżerze kolejek i są zarządzane przez menedżera kolejek (patrz sekcja [“Terminologia kolejkowania komunikatów”](#) na stronie 9). Fizyczny charakter kolejki zależy od systemu operacyjnego, na którym jest uruchomiony menedżer kolejek. Kolejka może być niestabilnym obszarem buforu w pamięci komputera lub zestawem danych na urządzeniu pamięci trwałej (takim jak dysk). Fizyczne zarządzanie kolejkami jest odpowiedzialne za menedżera kolejek i nie jest widoczne dla uczestniczących aplikacji.

Programy uzyskują dostęp do kolejek tylko za pośrednictwem usług zewnętrznych menedżera kolejek. Mogą otworzyć kolejkę, umieścić na niej komunikaty, pobrać z niej komunikaty i zamknąć kolejkę. Mogą także ustawiać i pytać o atrybuty kolejek.

Różne style kolejkowania komunikatów

Punkt-punkt

Jeden komunikat jest umieszczany w kolejce, a jedna aplikacja otrzymuje ten komunikat.

W przypadku przesyłania komunikatów w trybie punkt z punktem aplikacja wysyłająca musi znać informacje na temat aplikacji odbierającej, zanim będzie mogła wysłać komunikat do tej aplikacji. Na przykład aplikacja wysyłająca może potrzebować znać nazwę kolejki, do której mają być wysyłane informacje, a także może określić nazwę menedżera kolejek.

Publikowanie/subskrypcja

Kopia każdego komunikatu publikowanego przez aplikację publikacją jest dostarczana do każdej zainteresowanej aplikacji. Może być wiele, jeden lub brak zainteresowanych aplikacji. W publikacji/subskrypcji zainteresowana aplikacja jest znana jako subskrybent, a komunikaty są umieszczane w kolejce w kolejce identyfikowanej przez subskrypcję.

Przesyłanie komunikatów w trybie publikowania/subskrypcji pozwala na oddzielenie dostawcy informacji od konsumentów tych informacji. Aplikacja wysyłająca i odbierająca nie muszą znać więcej informacji o sobie, aby informacje były wysyłane i odbierane. Więcej informacji na ten temat zawiera sekcja [“Przesyłanie komunikatów publikowania/subskrypcji”](#) na stronie 63.

Korzyści płynące z kolejkowania komunikatów do projektanta aplikacji i programisty

Program IBM MQ umożliwia programom użytkowym używanie *kolejkowania komunikatów* do uczestniczenia w przetwarzaniu sterowanym komunikatami. Programy użytkowe mogą komunikować się na różnych platformach za pomocą odpowiednich produktów oprogramowania kolejkowania komunikatów. Na przykład aplikacje produktu z/OS mogą komunikować się za pośrednictwem produktu IBM MQ for z/OS. Aplikacje są ekranowane z mechaniki bazowej komunikacji. Niektóre z innych korzyści związanych z kolejkowaniem komunikatów to:

- Aplikacje można projektować przy użyciu małych programów, które mogą być współużytkowane przez wiele aplikacji.
- Nowe aplikacje można szybko zbudować, korzystając z tych bloków.
- Zmiany w sposobie pracy menedżerów kolejek nie mają wpływu na aplikacje napisane w celu używania technik kolejkowania komunikatów.
- Nie ma potrzeby korzystania z żadnych protokołów komunikacyjnych. Menedżer kolejek zajmuje się wszystkimi aspektami komunikacji dla użytkownika.
- Programy, które odbierają komunikaty, nie muszą być uruchomione w momencie wysłania komunikatów do nich. Komunikaty są zachowywane w kolejkach.

Projektanci mogą zmniejszyć koszt swoich aplikacji, ponieważ programowanie jest szybsze, mniej programistów jest potrzebnych, a wymagania dotyczące umiejętności programowania są niższe niż w przypadku aplikacji, które nie korzystają z kolejkowania komunikatów.

Produkt IBM MQ implementuje wspólny aplikacyjny interfejs programistyczny, znany jako *interfejs kolejki komunikatów* (lub MQI), niezależnie od tego, gdzie są uruchamiane aplikacje. Ułatwia to uruchamianie programów aplikacji z jednej platformy do drugiej.

Szczegółowe informacje na temat interfejsu MQI zawiera sekcja [Przegląd interfejsu kolejki komunikatów](#).

Główne cechy i korzyści związane z kolejkowaniem komunikatów

Informacje te wyróżniają niektóre funkcje i korzyści wynikające z kolejkowania komunikatów. Opisano w nim takie funkcje, jak bezpieczeństwo i integralność danych w kolejkowaniu komunikatów.

Główne funkcje aplikacji, które wykorzystują techniki kolejkowania komunikatów, to:

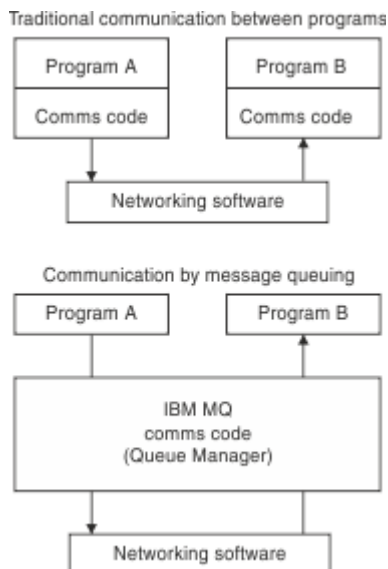
- Nie ma bezpośrednich połączeń między programami.
- Komunikacja między programami może być niezależna od czasu.
- Prace mogą być prowadzone przez małe, samodzielne programy.
- Komunikacja może być sterowana zdarzeniami.
- Aplikacje mogą przypisywać priorytet do komunikatu.
- Bezpieczeństwo.
- Integralność danych.
- Obsługa odtwarzania.

Brak bezpośrednich połączeń między programami

Kolejkowanie komunikatów to technika komunikacji pośredniej między programami. Może być używany w dowolnej aplikacji, w której programy komunikują się ze sobą. Komunikacja odbywa się przez jeden program umieszczający komunikaty w kolejce (którego właścicielem jest menedżer kolejek), a inny program pobierający komunikaty z kolejki.

Programy mogą uzyskać komunikaty, które zostały umieszczone w kolejce przez inne programy. Inne programy mogą być połączone z tym samym menedżerem kolejek, co program odbierający, lub do innego menedżera kolejek. Ten inny menedżer kolejek może znajdować się w innym systemie, innym systemie komputerowym, a nawet w innym przedsiębiorstwie lub przedsiębiorstwie.

Nie istnieją żadne połączenia fizyczne między programami komunikującymi się za pomocą kolejek komunikatów. Program wysyła komunikaty do kolejki, której właścicielem jest menedżer kolejek, a inny program pobiera komunikaty z kolejki (patrz sekcja [Rysunek 1 na stronie 8](#)).



Rysunek 1. Kolejowanie komunikatów w porównaniu z tradycyjną komunikacją

Tak jak w przypadku poczty elektronicznej, poszczególne komunikaty, które są częścią podróży transakcyjnej przez sieć w sklepie-i-do przodupodstawy. Jeśli połączenie między węzłami nie powiedzie się, komunikat zostanie zachowany do momentu odtworzenia odsyłaacza lub operator lub program przekierowuje komunikat.

Mechanizm, za pomocą którego komunikat przenosi się z kolejki do kolejki, jest ukryty przed programami. W związku z tym programy są prostsze.

Komunikacja niezależna od czasu

Programy żądające innych czynności do wykonania nie muszą czekać na odpowiedź na żądanie. Mogą wykonywać inne prace i przetwarzać odpowiedź po jej nadejściu lub w późniejszym czasie. Podczas pisania aplikacji do przesyłania komunikatów nie jest wymagana znajomość (lub jej dotyczy), gdy program wysyła komunikat, lub gdy cel jest w stanie odebrać komunikat. Komunikat nie został utracony. Jest on zachowywany przez menedżer kolejek do czasu, aż obiekt docelowy będzie gotowy do przetworzenia. Komunikat pozostaje w kolejce, dopóki nie zostanie usunięty przez program. Oznacza to, że programy aplikacji wysyłające i odbierające są oddzielone od siebie. Nadawca może kontynuować przetwarzanie bez czekania na odbiorcę, aby potwierdzić otrzymanie komunikatu. Aplikacja docelowa nie musi być uruchomiona, gdy komunikat jest wysyłany. Może on pobrać komunikat po jego uruchomieniu.

Małe programy

Kolejkowanie komunikatów umożliwia korzystanie z zalet korzystania z małych, samodzielnych programów. Zamiast jednego, dużego programu wykonującego wszystkie części zadania sekwencyjnie, można rozłożyć zadanie na kilka mniejszych, niezależnych programów. Program wysyłający żądanie wysyła komunikaty do każdego z oddzielnych programów, prosząc je o wykonanie swojej funkcji; po zakończeniu każdego programu wyniki są wysyłane z powrotem w postaci jednego lub większej liczby komunikatów.

Przetwarzanie sterowane komunikatami

Gdy komunikaty docierają do kolejki, mogą automatycznie uruchamiać aplikację przy użyciu opcji *wyzwalanie*. Jeśli jest to konieczne, aplikacje mogą zostać zatrzymane po przetworzeniu komunikatu (lub komunikatów).

Przetwarzanie sterowane zdarzeniami

Programy mogą być kontrolowane w zależności od stanu kolejek. Można na przykład zaaranżować uruchomienie programu od razu po nadejściu komunikatu do kolejki lub określić, że program nie zostanie uruchomiony, dopóki nie będzie, na przykład, 10 komunikatów powyżej określonego priorytetu w kolejce lub 10 komunikatów o dowolnym priorytecie w kolejce.

Priorytet komunikatu

Program może przypisać priorytet do komunikatu, gdy powoduje umieszczenie komunikatu w kolejce. Określa to położenie w kolejce, w której dodawany jest nowy komunikat.

Programy mogą pobrać komunikaty z kolejki w kolejności, w jakiej komunikaty znajdują się w kolejce, lub przez uzyskanie konkretnego komunikatu. (Program może chcieć uzyskać konkretną wiadomość, jeśli szuka odpowiedzi na żądanie wysłane wcześniej.)

Zabezpieczenia

Udostępniane są zabezpieczenia, w tym uwierzytelnianie aplikacji podczas korzystania z menedżera kolejek, sprawdzanie autoryzacji podczas korzystania z zasobów, takich jak kolejka w menedżerze kolejek, oraz szyfrowanie danych komunikatów podczas podróży przez sieć, a także w przypadku, gdy rezyduje on w kolejkach. Więcej informacji na temat zabezpieczeń zawiera sekcja [Przegląd zabezpieczeń](#).

Integralność danych

Integralność danych jest zapewniana przez jednostki pracy. Synchronizacja początku i końca jednostek pracy jest w pełni obsługiwana jako opcja dla każdej operacji MQGET lub MQPUT, co pozwala na zatwierdzenie wyników jednostki pracy lub wycofanie zmian. Obsługa punktów synchronizacji działa wewnętrznie lub zewnętrznie na IBM MQ w zależności od formy koordynacji punktów synchronizacji wybranej dla aplikacji.

Obsługa odtwarzania


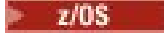

Aby odtwarzanie było możliwe, rejestrowane są wszystkie trwałe aktualizacje produktu IBM MQ . Jeśli odtwarzanie jest konieczne, odtwarzane są wszystkie komunikaty trwałe, wszystkie transakcje w trakcie realizacji są wycofywane, a wszystkie zatwierdzania i wycofywania punktów synchronizacji są obsługiwane w normalny sposób menedżera punktów synchronizacji w elemencie sterującym. Więcej informacji na temat trwałych komunikatów zawiera sekcja [Trwałość komunikatów](#).

Uwaga: Podczas rozważania klientów i serwerów IBM MQ nie trzeba zmieniać aplikacji serwera w celu obsługi dodatkowych IBM MQ MQI clients na nowych platformach. Podobnie, IBM MQ MQI client może, bez zmian, funkcjonować z dodatkowymi typami serwerów.

Terminologia kolejkowania komunikatów

Te informacje dają wgląd w niektóre terminy używane w kolejkowaniu komunikatów.

Są to obiekty następujące:

- [Wiadomość](#)
- [deskryptor komunikatu](#)
- [Kolejka](#)
- [menedżer kolejek](#)
- [Kanały](#)
- [Agent kanału komunikatów](#)
- [Klaster](#)
-  [Kolejka współużytkowana](#)
-  [Grupa współużytkowania kolejek](#)
-  [Kolejkowanie wewnątrz grupy](#)
- [IBM MQ MQI client](#)
- [Punkt-punkt](#)
- [Publikowanie/subskrypcja](#)
- [Temat](#)
- [Subskrypcja](#)

Komunikat

W kolejkowaniu komunikatów jest to zbiór danych wysyłanych przez jeden program i przeznaczony do innego programu. Patrz [Komunikaty produktu IBM MQ](#). Więcej informacji na temat typów komunikatów zawiera sekcja [Typy komunikatów](#).

deskryptor komunikatu

Komunikat IBM MQ składa się z informacji sterujących i danych aplikacji.

Informacje sterujące są definiowane w strukturze deskryptora komunikatu (MQMD) i zawierają takie elementy jak:

- Typ komunikatu
- Identyfikator komunikatu
- Priorytet dostarczenia wiadomości

Struktura i treść danych aplikacji są określane przez uczestniczące programy, a nie przez program IBM MQ.

Kolejka

Nazwane miejsce docelowe, do którego mogą być wysyłane komunikaty. Komunikaty gromadzą się w kolejkach do czasu ich pobrania przez programy, które służą do obsługi tych kolejek.

Menedżer kolejek

Menedżer kolejek jest to program systemowy, który udostępnia aplikacjom usługi kolejkowania.

Udostępnia on aplikacyjny interfejs programistyczny, który umożliwia programom umieszczanie komunikatów w kolejkach oraz pobieranie komunikatów z kolejek. Menedżer kolejek udostępnia dodatkowe funkcje, dzięki czemu administratorzy mogą tworzyć nowe kolejki, zmieniać właściwości istniejących kolejek i sterować działaniem menedżera kolejek.

Aby usługi kolejkowania komunikatów produktu IBM MQ były dostępne w systemie, musi być uruchomiony menedżer kolejek. W pojedynczym systemie może być uruchomiony więcej niż jeden menedżer kolejek (na przykład w celu oddzielenia systemu testowego od *aktywnego* systemu). Do aplikacji każdy menedżer kolejek jest identyfikowany za pomocą *uchwyty połączenia (Hconn)*.

Wiele różnych aplikacji może jednocześnie korzystać z usług menedżera kolejek, a aplikacje te mogą być całkowicie niepowiązane. Aby program mógł korzystać z usług menedżera kolejek, musi nawiązać połączenie z tym menedżerem kolejek.

Aby aplikacje wysyłały komunikaty do aplikacji połączonych z innymi menedżerami kolejek, menedżery kolejek muszą być w stanie komunikować się między sobą. Produkt IBM MQ implementuje protokół *store-and-forward* w celu zapewnienia bezpiecznego dostarczania komunikatów między tymi aplikacjami.

Kanały

Kanały to obiekty, które udostępniają ścieżkę komunikacji z jednego menedżera kolejek do innego. Kanały służą do przenoszenia komunikatów z jednego menedżera kolejek do innego, a także do zabezpieczenia aplikacji z bazowych protokołów komunikacyjnych. Menedżery kolejek mogą istnieć w tym samym systemie lub w różnych systemach na tej samej platformie lub na różnych platformach. Wysyłane komunikaty mogą pochodzić z wielu lokalizacji:

- Programy użytkowe napisane przez użytkownika, które przesyłają dane z jednego węzła do drugiego.
- Aplikacje administracyjne napisane przez użytkownika, które korzystają z komend PCF lub MQAI.
- IBM MQ Explorer.
- Menedżery kolejek, które wysyłają komunikaty zdarzeń instrumentacji do innego menedżera kolejek.
- Menedżery kolejek, które wysyłają zdalne komendy administracyjne do innego menedżera kolejek. Na przykład za pomocą komend MQSC lub administrative REST API.

Agent kanału komunikatów

Agent kanału komunikatów to jeden koniec kanału. Para agentów kanałów komunikatów, jeden nadawca i jeden odbierający, tworzą kanał i przenoszą komunikaty z jednego menedżera kolejek do drugiego.

Klaster

Klaster jest siecią menedżerów kolejek, które są logicznie powiązane w pewien sposób.

W przypadku sieci IBM MQ korzystającej z rozproszonego kolejkowania bez grupowania, każdy menedżer kolejek jest niezależny. Jeśli jeden menedżer kolejek musi wysyłać komunikaty do innego menedżera kolejek, musi on zdefiniować kolejkę transmisji i kanał do zdalnego menedżera kolejek.

Istnieją dwa różne przyczyny korzystania z klastrów: aby zmniejszyć administrowanie systemem oraz zwiększyć dostępność i równoważenie obciążenia.

Gdy tylko ustanowi się najmniejszy klaster, można skorzystać z uproszczonej administracji systemowej. Menedżery kolejek, które są częścią klastra, wymagają mniejszej liczby definicji, a więc ryzyko wystąpienia błędu w definicjach jest zmniejszone.

Więcej informacji na temat łączenia w klastry można znaleźć w sekcji [Klastry](#).

Kolejka współużytkowana

Kolejka współużytkowana jest typem kolejki lokalnej z komunikatami, do których może uzyskać dostęp co najmniej jeden menedżer kolejek, który znajduje się w syspleksie. Wartość ta nie jest taka sama, jak kolejka współużytkowana przez więcej niż jedną aplikację, przy użyciu tego samego menedżera kolejek. Ma to zastosowanie tylko do produktu IBM MQ for z/OS.

Grupa współużytkowania kolejek

Menedżery kolejek, które mogą uzyskiwać dostęp do tego samego zestawu kolejek współużytkowanych, tworzą grupę o nazwie *grupa współużytkowania kolejek* (QSG). Komunikują się ze sobą za pomocą narzędzia CF, w którym przechowywane są współużytkowane kolejki. Ma to zastosowanie tylko do produktu IBM MQ for z/OS. Więcej informacji na temat grup współużytkowania kolejek zawiera sekcja [“Współużytkowane kolejki i grupy współużytkowania kolejek”](#) na stronie 172 .





Kolejowanie wewnątrz grupy

Menedżery kolejek w grupie współużytkowania kolejek mogą komunikować się przy użyciu normalnych kanałów lub można użyć techniki o nazwie *queuing wewnątrzgrupowy* (IGQ), która umożliwia szybkie przesyłanie komunikatów bez definiowania kanałów. Ma to zastosowanie tylko do produktu IBM MQ for z/OS.

IBM MQ MQI client

IBM MQ MQI *klienty* są niezależnie instalowanymi komponentami produktu IBM MQ. Klient MQI umożliwia uruchamianie aplikacji produktu IBM MQ za pomocą protokołu komunikacyjnego, interakcję z co najmniej jednym serwerem Message Queue Interface (MQI) na innych platformach oraz nawiązanie połączenia z ich menedżerami kolejek.

Szczegółowe informacje na temat instalowania i używania komponentów produktu IBM MQ MQI client można znaleźć w następujących tematach:

-  [Instalowanie klienta IBM MQ w systemie AIX](#)
-  [Instalowanie klienta IBM MQ w systemie Linux®](#)
-  [Instalowanie klienta IBM MQ w systemie Windows](#)
-  [Instalowanie klienta IBM MQ w systemie IBM i](#)

i [Konfigurowanie połączeń między serwerem a klientem](#).

Przesyłanie komunikatów w modelu punkt-punkt

W przesyłaniu komunikatów w trybie punkt z punktem każdy komunikat przemieszcza się z jednej aplikacji produkującej do jednej aplikacji konsumowania. Komunikaty są przesyłane przez aplikację produkującą umieszczając komunikaty w kolejce, a aplikacja konsumująca pobiera je z tej kolejki.

Przesyłanie komunikatów publikowania/subskrypcji

W przesyłaniu wiadomości w trybie publikowania/subskrypcji kopia każdego komunikatu publikowanego przez aplikację publikowania jest dostarczana do każdej zainteresowanej aplikacji. Może być wiele, jedna lub żadna z zainteresowanych aplikacji. W publikacji/subskrypcji zainteresowana aplikacja jest znana jako subskrybent, a komunikaty są umieszczane w kolejce w kolejce identyfikowanej przez subskrypcję. Więcej informacji na ten temat zawiera sekcja [“Przesyłanie komunikatów publikowania/subskrypcji” na stronie 63.](#)

Temat

Temat to łańcuch znaków służący do opisywania tematyki informacji, które zostaną opublikowane w ramach komunikatu publikowania/subskrypcji.

Tematy są kluczowym czynnikiem zapewniającym pomyślne dostarczanie komunikatów w ramach systemu publikowania/subskrypcji. Publikator nie wprowadza konkretnego adresu docelowego w przypadku każdego komunikatu, lecz przypisuje temat do komunikatu. Menedżer kolejek dopasowuje temat do listy subskrybentów tego tematu i dostarcza ten komunikat do każdego z nich.

Subskrypcja

Aplikacja publikowania/subskrypcji może rejestrować zainteresowanie komunikatami na temat konkretnych tematów. Gdy aplikacja wykonuje tę aplikację, jest ona znana jako subskrybent, a subskrypcja terminu definiuje, w jaki sposób zgodne komunikaty są kolejgowane do przetwarzania.

Subskrypcja zawiera informacje na temat tożsamości subskrybenta oraz informacje o tożsamości kolejki docelowej, do której mają zostać umieszczone publikacje. Zawiera również informacje o tym, jak publikacja ma być umieszczona w kolejce docelowej.

Komunikaty i kolejki

Komunikaty i kolejki są podstawowymi komponentami systemu kolejgowania komunikatów.

Co to jest wiadomość?

Komunikat to łańcuch bajtów, który ma znaczenie dla aplikacji, które używają tego komunikatu. Komunikaty są używane do przesyłania informacji z jednego programu użytkowego do innego (lub między różnymi częściami tej samej aplikacji). Aplikacje mogą być uruchomione na tej samej platformie lub na różnych platformach.

Komunikat IBM MQ składa się z następujących elementów:

- *Dane aplikacji.* Treść i struktura danych aplikacji są definiowane przez programy użytkowe, które używają tego programu.
- *Deskryptor komunikatu.* Deskryptor komunikatu identyfikuje komunikat i zawiera dodatkowe informacje sterujące, takie jak typ komunikatu i priorytet przypisany do komunikatu przez aplikację wysyłającą.

Format deskryptora komunikatu jest definiowany przez produkt IBM MQ. Pełny opis deskryptora komunikatu znajduje się w sekcji [MQMD-deskryptor komunikatu.](#)

- *Właściwości komunikatu.* Meta-dane na temat komunikatu. Treść właściwości komunikatu jest definiowana przez programy użytkowe, które ich używają. Więcej informacji na ten temat zawiera sekcja [Właściwości komunikatu.](#)

Długości komunikatów

Domyślna maksymalna długość komunikatu to 4 MB, chociaż można ją zwiększyć do maksymalnej długości 100 MB (gdzie 1 MB jest równe 1 048 576 bajtów). W praktyce długość komunikatu może być ograniczona przez:

- Maksymalna długość komunikatu zdefiniowana dla kolejki odbiorczej
- Maksymalna długość komunikatu zdefiniowana dla menedżera kolejek
- Maksymalna długość komunikatu zdefiniowana przez kolejkę
- Maksymalna długość komunikatu zdefiniowana przez aplikację wysyłającą lub odbierającą
- Ilość pamięci dostępnej dla komunikatu

Wysłanie wszystkich informacji wymaganych przez aplikację może zająć kilka komunikatów.

Jak aplikacje wysyłają i odbierają komunikaty?

Aplikacje wysyłają i odbierają komunikaty przy użyciu **wywołań MQI**.

Na przykład, aby umieścić komunikat w kolejce, aplikacja:

1. Otwiera wymaganą kolejkę, wywołując wywołanie MQI MQOPEN
2. Wysyła wywołanie MQI MQPUT w celu umieszczenia komunikatu w kolejce.

Inna aplikacja może pobrać komunikat z tej samej kolejki, wydając wywołanie MQI MQGET .

Więcej informacji na temat wywołań MQI zawiera sekcja [Wywołania MQI](#).

Co to jest kolejka?

Kolejka to struktura danych używana do przechowywania komunikatów.

Każda kolejka jest własnością *menedżera kolejek*. Menedżer kolejek obsługuje należące do niego kolejki i zapisuje wszystkie odebrane komunikaty w odpowiednich kolejkach. Komunikaty mogą być umieszczane w kolejce przez programy użytkowe lub przez menedżera kolejek w ramach normalnej pracy.

Kolejki predefiniowane i kolejki dynamiczne

Kolejki mogą charakteryzować się sposobem ich tworzenia:

- **Kolejki predefiniowane** są tworzone przez administratora przy użyciu odpowiednich komend MQSC lub PCF. Predefiniowane kolejki są trwałe; istnieją niezależnie od aplikacji, które ich używają i przetrwają IBM MQ restarty.
- **Kolejki dynamiczne** są tworzone, gdy aplikacja Wysyła żądanie MQOPEN określające nazwę *kolejki modelowej*. Utworzona kolejka jest oparta na *definicji kolejki szablonów*, która jest nazywana kolejką modelową. Kolejkę modelową można utworzyć przy użyciu komendy MQSC DEFINE QMODEL. Atrybuty kolejki modelowej (na przykład maksymalna liczba komunikatów, które mogą być w niej zapisane) są dziedziczone przez dowolną kolejkę dynamiczną, która została z niego utworzona.

Kolejki modelowe mają atrybut, który określa, czy kolejka dynamiczna ma być trwała czy tymczasowa. Trwałe kolejki przetrwają aplikacje i restarty menedżera kolejek; kolejki tymczasowe są tracone przy restarcie.

Pobieranie komunikatów z kolejek

Odpowiednio autoryzowane aplikacje mogą pobierać komunikaty z kolejki zgodnie z następującymi algorytmami pobierania:

- First-in-first-out (FIFO).
- Priorytet komunikatu zgodnie z definicją w deskrypcji komunikatu. Komunikaty, które mają ten sam priorytet, są pobierane w oparciu o FIFO.

- Żądanie programu dla konkretnego komunikatu.

Żądanie MQGET z aplikacji określa używaną metodę.

Obiekty IBM MQ

Menedżery kolejek definiują właściwości obiektów produktu IBM MQ . Wartości tych właściwości wpływają na sposób, w jaki produkt IBM MQ przetwarza te obiekty. Użytkownik tworzy i zarządza obiektami za pomocą komend i interfejsów produktu IBM MQ . Z poziomu aplikacji do sterowania obiektami służy interfejs MQI (Message Queue Interface). Obiekty są identyfikowane przez program IBM MQ *deskryptor obiektu* (MQOD), gdy jest on adresowany z programu.




Administrowanie obiektami obejmuje następujące zadania:

- Uruchamianie i zatrzymywanie menedżerów kolejek.
- Tworzenie obiektów, w szczególności kolejek, dla aplikacji.
- Wyświetlanie lub zmienianie atrybutów obiektów.
- Usuwanie obiektów.
- Praca z kanałami w celu utworzenia ścieżek komunikacyjnych do menedżerów kolejek w innych systemach (zdalnych).
- Tworzenie *klastrów* menedżerów kolejek w celu uproszczenia ogólnego procesu administracyjnego oraz zrównoważenia obciążenia.

Z wyjątkiem kolejek dynamicznych obiekty muszą być zdefiniowane w menedżerze kolejek, zanim będzie można z nimi pracować.


Jeśli do wykonania operacji administrowania obiektami używana jest komenda IBM MQ , menedżer kolejek sprawdza, czy użytkownik posiada wymagany poziom uprawnień do wykonania operacji. Podobnie, jeśli aplikacja korzysta z wywołania MQOPEN w celu otwarcia obiektu, menedżer kolejek sprawdza, czy aplikacja ma wymagany poziom uprawnień, zanim zezwoli na dostęp do tego obiektu. Kontrole są przeprowadzane na podstawie nazwy otwieranego obiektu.

Obiekty można definiować i zarządzać za pomocą następujących metod:

- Komendy PCF opisane w sekcji [Programmable command formats reference](#) (Skorowidz formatów komend programowalnych) i [Automating administration tasks](#) (Automatyczne
- Komendy MQSC opisane w sekcji [Komendy MQSC](#)
-  IBM MQ for z/OS operacje i panele sterowania opisane w sekcji [Operacyjny IBM MQ for z/OS](#)
-   The IBM MQ Explorer (Windows and Linux for Intel systems only). Więcej informacji na ten temat zawiera sekcja [Wprowadzenie do programu MQ Explorer](#).

Obiekty można również zarządzać za pomocą następujących metod:

- Komendy sterujące, które są wpisywane za pomocą klawiatury. Więcej informacji zawiera sekcja [Administrowanie za pomocą komend sterujących](#).
- Program IBM MQ Administration Interface (MQAI) wywołuje w programie. Patrz [Interfejs administracyjny produktu IBM MQ \(MQAI\)](#).

 W przypadku sekwencji komend IBM MQ for z/OS , które są używane regularnie, można pisać programy administracyjne, które tworzą komunikaty zawierające komendy i które umieszczają te komunikaty w systemowej kolejce wejściowej komend. Menedżer kolejek przetwarza komunikaty w tej kolejce w taki sam sposób, w jaki przetwarza komendy wprowadzone z poziomu wiersza komend lub z paneli operacji i sterowania. Ta technika jest opisana w sekcji [Pisanie programów do administrowania produktem IBM MQ](#) i demonstruje w przykładowej aplikacji programu Mail Manager dostarczonej z produktem IBM MQ for z/OS. Opis tego przykładu można znaleźć w sekcji [Przykładowe programy dla produktu IBM MQ for z/OS](#) .

IBM i W przypadku sekwencji komend IBM MQ for IBM i , które są używane regularnie, można pisać programy CL. Więcej informacji na ten temat zawiera sekcja [Zarządzanie programem IBM MQ for IBM i za pomocą komend CL](#).

ALW W przypadku sekwencji komend produktu IBM MQ w systemie AIX, Linux, and Windows można użyć narzędzia MQSC do uruchamiania serii komend przechowywanych w pliku. Więcej informacji na ten temat zawiera sekcja [Administrowanie za pomocą komend MQSC](#).

Typy obiektów

Wiele zadań administracyjnych obejmuje manipulowanie różnymi typami obiektów IBM MQ *obiektów*.

Informacje na temat nazewnictwa obiektów IBM MQ zawiera sekcja [“Nazywanie obiektów IBM MQ”](#) na stronie 36.

Więcej informacji na temat obiektów domyślnych utworzonych w menedżerze kolejek zawiera sekcja [“Domyślne obiekty systemowe”](#) na stronie 43.

Informacje na temat różnych typów obiektów produktu IBM MQ można znaleźć w następujących podtematach:

Pojęcia pokrewne

[“Wprowadzenie do kolejkowania komunikatów”](#) na stronie 5

Produkty IBM MQ umożliwiają programom komunikowanie się między różnymi komponentami (procesorami, systemami operacyjnymi, podsystemami i protokołami komunikacyjnymi) za pomocą spójnego interfejsu programistycznego aplikacji.

[“Atrybuty obiektu”](#) na stronie 42

Właściwości obiektu są definiowane przez jego atrybuty. Niektóre z nich można określić, inne mogą być tylko wyświetlane.

Odsyłacze pokrewne

[Komendy MQSC](#)

Kolejki

Wprowadzenie do kolejek i atrybutów kolejki produktu IBM MQ .

IBM MQ *kolejka* jest nazwanym obiektem, w którym aplikacje mogą umieszczać komunikaty i z których aplikacje mogą uzyskać komunikaty.

Komunikaty są zapisywane w kolejce, tak więc jeśli aplikacja oczekuje na odpowiedź na jej komunikat, to podczas oczekiwania na tę odpowiedź jest ona bezpłatna. Aplikacje uzyskują dostęp do kolejki przy użyciu interfejsu MQI (Message Queue Interface), który jest opisany w sekcji [Przegląd interfejsu kolejki komunikatów](#).

Zanim komunikat może zostać umieszczony w kolejce, kolejka musi być już utworzona. Właścicielem kolejki jest menedżer kolejek, a menedżer kolejek może być właścicielem wielu kolejek. Jednak każda kolejka musi mieć nazwę unikalną w obrębie tego menedżera kolejek.

Kolejka jest obsługiwana przez menedżer kolejek. W większości przypadków każda kolejka jest fizycznie zarządzana przez menedżer kolejek, ale nie jest to widoczne dla aplikacji. Kolejki współużytkowane IBM MQ for z/OS mogą być zarządzane przez dowolny menedżer kolejek w grupie współużytkowania kolejek.

Aby utworzyć kolejkę, można użyć komend IBM MQ (MQSC), PCF lub interfejsów specyficznych dla platformy. Na przykład panele kontrolne i panele kontrolne produktu IBM MQ for z/OS są specyficzne dla platformy.

W aplikacji można dynamicznie tworzyć kolejki lokalne dla zadań tymczasowych *dynamicznie* . Na przykład: można utworzyć kolejki *reply-to* (które nie są potrzebne po zakończeniu aplikacji). Więcej informacji na ten temat zawiera sekcja [“Dynamiczne i modelowe kolejki”](#) na stronie 20.

Przed użyciem kolejki należy otworzyć kolejkę, określając elementy, które mają być z nią związane. Na przykład można otworzyć kolejkę dla:

- Przeglądanie tylko komunikatów (nie pobieraj ich)
- Pobieranie komunikatów (i współużytkowanie dostępu z innymi programami lub z dostępem wyłącznym)
- Umieszczanie komunikatów w kolejce
- Uzyskiwanie informacji o atrybutach kolejki
- Ustawianie atrybutów kolejki

Pełną listę opcji, które można określić podczas otwierania kolejki, można znaleźć w sekcji [MQOPEN-Open object](#)(MQOPEN-Obiekt otwarty).

Atrybuty kolejek

Niektóre atrybuty kolejki są określone, gdy kolejka jest zdefiniowana i nie można jej zmienić później (na przykład typ kolejki). Inne atrybuty kolejek mogą być grupowane w te, które można zmieniać:

- Przez menedżer kolejek podczas przetwarzania kolejki (na przykład bieżąca głębokość kolejki).
- Tylko przez komendy (na przykład opis tekstowy kolejki)
- Wg aplikacji, za pomocą wywołania MQSET (na przykład, czy operacje put są dozwolone w kolejce)

Za pomocą wywołania MQINQ można znaleźć wartości wszystkich atrybutów.

Atrybuty, które są wspólne dla więcej niż jednego typu kolejki:

Nazwa QName

Nazwa kolejki.

QTYPE

Typ kolejki.

QDesc

Opis tekstowy kolejki.

InhibitGet

Określa, czy programy mają uprawnienia do pobierania komunikatów z kolejki. Nie można jednak nigdy pobrać komunikatów z kolejek zdalnych.

InhibitPut

Określa, czy programy mogą umieszczać komunikaty w kolejce.

DefPriority

Domyślny priorytet komunikatów umieszczanych w kolejce.

DefPersistence

Domyślna trwałość komunikatów umieszczonych w kolejce

Zasięg

Określa, czy pozycja dla tej kolejki istnieje również w usłudze nazw.

 Atrybut **Scope** nie jest obsługiwany w systemie z/OS

Pełny opis tych atrybutów znajduje się w sekcji [Atrybuty dla kolejek](#).

Pojęcia pokrewne

 Kolejki współużytkowane

Kolejka współużytkowana jest typem kolejki lokalnej. Dostęp do komunikatów w tej kolejce może być uzyskiwany przez co najmniej jeden menedżer kolejek, który znajduje się w syspleksie.

“Kolejki klastra” na stronie 59

Kolejka klastra to kolejka udostępniana przez menedżer kolejek klastra innym menedżerom kolejek w klastrze.

“Kolejki niedostarczane” na stronie 50

Kolejka niedostarczonych komunikatów (lub kolejka niedostarczonych komunikatów) jest kolejką, do której wysyłane są komunikaty, jeśli nie mogą być kierowane do ich poprawnego miejsca docelowego. Każdy menedżer kolejek zwykle ma kolejkę niedostarczonych komunikatów.

Zadania pokrewne

[Tworzenie odwołania do aplikacji](#)

Odsyłacze pokrewne

[Komendy MQSC](#)

“Porównanie między kolejkami współużytkowanymi i kolejkami klastra” na stronie 59

Informacje te zostały zaprojektowane w celu ułatwienia porównywania kolejek współużytkowanych i kolejek klastra oraz podejmowania decyzji, które mogą być bardziej odpowiednie dla danego systemu.

Kolejki lokalne

Kolejki przesyłania, inicjowania, martwo-litery, komendy, domyślnej, kanału i zdarzeń są typami kolejek lokalnych.

Kolejka jest znana w programie jako *lokalna*, jeśli jej właścicielem jest menedżer kolejek, z którym połączony jest program. Komunikaty można pobierać z kolejek lokalnych oraz umieszczać je w nich.

Obiekt definicji kolejki przechowuje informacje o definicji kolejki, a także komunikaty fizyczne umieszczone w kolejce.

Każdy menedżer kolejek może mieć kilka kolejek lokalnych, których używa do specjalnych celów:

Kolejki transmisji

Gdy aplikacja wysyła komunikat do kolejki zdalnej, lokalny menedżer kolejek zapisuje ten komunikat w specjalnej kolejce lokalnej, nazywanych *kolejką transmisji*. Aplikacje mogą umieszczać komunikaty bezpośrednio w kolejce transmisji lub pośrednio przez definicję kolejki zdalnej.

Gdy menedżer kolejek wysyła komunikaty do zdalnego menedżera kolejek, identyfikuje kolejkę transmisji przy użyciu następującej sekwencji:

1. Kolejka transmisji określona w atrybucie XMITQ lokalnej definicji kolejki zdalnej.
2. Kolejka transmisji o tej samej nazwie, co zdalny menedżer kolejek. Ta wartość jest wartością domyślną dla kolejki XMITQ lokalnej definicji kolejki zdalnej.
3. Kolejka transmisji określona w atrybucie DEFXMITQ lokalnego menedżera kolejek.

Agent kanału komunikatów jest programem kanału, który jest powiązany z kolejką transmisji i dostarcza komunikat do następnego miejsca docelowego. Następnym miejscem docelowym jest menedżer kolejek, z którym połączony jest kanał komunikatów. Nie musi on być tym samym menedżerem kolejek, co końcowe miejsce docelowe komunikatu. Gdy komunikat zostanie dostarczony do następnego miejsca docelowego, zostanie on usunięty z kolejki transmisji. Komunikat może być musiał przejść przez wiele menedżerów kolejek w trakcie swojej podróży do miejsca docelowego. Należy zdefiniować kolejkę transmisji dla każdego menedżera kolejek na trasie, a każdy z tych komunikatów wstrzymuje oczekiwanie na przesłanie do następnego miejsca docelowego. Normalna kolejka transmisji przechowuje komunikaty dla następnego miejsca docelowego, mimo że komunikaty mogą mieć inne miejsca docelowe. Kolejka transmisji klastra przechowuje komunikaty dla wielu miejsc docelowych. `CORRELID` każdego komunikatu identyfikuje kanał, w którym komunikat jest umieszczany w celu przesłania go do następnego miejsca docelowego.

W menedżerze kolejek można zdefiniować kilka kolejek transmisji. Można zdefiniować kilka kolejek transmisji dla tego samego miejsca docelowego, przy czym każdy z nich jest używany dla innej klasy usług. Na przykład można utworzyć różne kolejki transmisji dla małych komunikatów i dużych komunikatów, które mają być używane do tego samego miejsca docelowego. Następnie można przesyłać komunikaty przy użyciu różnych kanałów komunikatów, tak aby duże komunikaty nie przechowywały mniejszych komunikatów. Wszystkie komunikaty do kolejek klastra lub tematy klastra są umieszczane w pojedynczej kolejce transmisji klastra `SYSTEM.CLUSTER.TRANSMIT.QUEUE` domyślnie. Za pomocą opcji można zmienić wartość domyślną i oddzielić ruch komunikatów do różnych menedżerów kolejek klastra w różnych kolejkach transmisji

klastra. Jeśli atrybut menedżera kolejek DEFCLXQ zostanie ustawiony na wartość CHANNEL, każdy kanał nadawczy klastra utworzy osobną kolejkę transmisji klastra. Alternatywnie można ręcznie zdefiniować kolejki transmisji klastra dla kanałów nadawczych klastra, które mają być używane.

Kolejki transmisji mogą wyzwoliwać agenta kanału komunikatów w celu wysyłania komunikatów. Patrz sekcja [Uruchamianie aplikacji IBM MQ za pomocą wyzwalaczy](#).

z/OS Jeśli w systemie IBM MQ for z/OS używana jest kolejki wewnątrz grupy, kolejka transmisji jest serwisowana przez *wewnątrzgrupowy agent kolejki*. Współużytkowana kolejka transmisji jest używana podczas korzystania z kolejki wewnątrz grupy w systemie IBM MQ for z/OS.

Kolejki inicjuj

Kolejka inicjuj jest kolejką lokalną, w której menedżer kolejek umieszcza komunikat wyzwalacza, gdy zdarzenie wyzwalające występuje w kolejce aplikacji.

Zdarzenie wyzwalające jest to zdarzenie, które ma na celu uruchomienie programu w celu rozpoczęcia przetwarzania kolejki. Na przykład zdarzenie może zawierać więcej niż 10 komunikatów. Więcej informacji na temat sposobu wyzwalania można znaleźć w sekcji [Uruchamianie aplikacji IBM MQ za pomocą wyzwalaczy](#).

Kolejka niedostarczana (niedostarczone komunikaty)

Kolejka niedostarczonych komunikatów (niedostarczonych komunikatów) jest kolejką lokalną, w której menedżer kolejek umieszcza komunikaty, których nie może dostarczyć.

Gdy menedżer kolejek umieszcza komunikat w kolejce niedostarczonych komunikatów, dodaje nagłówek do komunikatu. Informacje nagłówkowe zawierają przyczynę umieszczenia komunikatu w kolejce niedostarczonych komunikatów przez menedżera kolejek. Zawiera on również miejsce docelowe oryginalnego komunikatu, datę i godzinę umieszczenia komunikatu w kolejce niedostarczonych komunikatów.

Aplikacje mogą również używać kolejki dla komunikatów, których nie można dostarczyć. Więcej informacji na ten temat zawiera sekcja [Korzystanie z kolejki niedostarczonych komunikatów \(niedostarczone komunikaty\)](#).

Systemowa kolejka komend

Kolejka komend systemowych to kolejka, do której odpowiednio autoryzowane aplikacje mogą wysłać komendy IBM MQ. Kolejki te otrzymują komendy PCF, MQSC i CL, które są obsługiwane na danej platformie, w gotowości menedżera kolejek do ich działania.

z/OS W systemie IBM MQ for z/OS kolejka jest nazywana SYSTEM.COMMAND.INPUT; na innych platformach jest ona nazywana SYSTEM.ADMIN.COMMAND.QUEUE. Akceptowane komendy różnią się w zależności od platformy. Szczegółowe informacje można znaleźć w sekcji [Skorowidz formatów komend programowalnych](#).

Systemowe kolejki domyślne

Kolejki domyślne systemowe zawierają początkowe definicje kolejek systemu. Podczas tworzenia definicji kolejki menedżer kolejek kopiuje definicję z odpowiedniej domyślnej kolejki systemowej. Tworzenie definicji kolejki różni się od tworzenia kolejki dynamicznej. Definicja kolejki dynamicznej jest oparta na kolejce modelowej wybieranej jako szablon dla kolejki dynamicznej.

kolejki zdarzeń

Kolejki zdarzeń zawierają komunikaty o zdarzeniach. Komunikaty te są zgłaszane przez menedżer kolejek lub kanał.

Kolejki zdalne

W przypadku programu kolejka jest *zdalna*, jeśli jest własnością innego menedżera kolejek do tego, do którego program jest połączony.


Jeśli nawiązano połączenie komunikacyjne, program może wysłać komunikat do kolejki zdalnej. Program nigdy nie może uzyskać komunikatu z kolejki zdalnej.

Obiekt definicji kolejki, utworzony podczas definiowania kolejki zdalnej, przechowuje tylko informacje niezbędne dla lokalnego menedżera kolejek w celu zlokalizowania kolejki, do której ma zostać wyświetlony komunikat. Ten obiekt jest znany jako *lokalna definicja kolejki zdalnej*. Wszystkie atrybuty kolejki zdalnej są przechowywane przez menedżera kolejek, który jest jego właścicielem, ponieważ jest to kolejka lokalna dla tego menedżera kolejek.

Podczas otwierania kolejki zdalnej w celu zidentyfikowania kolejki należy określić jedną z następujących wartości:

- Nazwa lokalnej definicji, która definiuje kolejkę zdalną. Z punktu widzenia aplikacji jest to takie samo, jak otwieranie kolejki lokalnej. Aplikacja nie musi wiedzieć, czy kolejka jest lokalna, czy zdalna.

Aby utworzyć lokalną definicję kolejki zdalnej na wszystkich platformach z wyjątkiem IBM i, należy użyć komendy `DEFINE QREMOTE`.

 W systemie IBM i należy użyć komendy `CRTMQMQ`.

- Nazwa zdalnego menedżera kolejek i nazwa kolejki, która jest znana temu menedżerowi kolejek zdalnych.

Oprócz wspólnych atrybutów opisanych w sekcji [“Atrybuty kolejek”](#) na stronie 16 lokalne definicje kolejek zdalnych mają trzy atrybuty. Te trzy atrybuty są następujące:

RemoteQName

Nazwa, o której wie menedżer kolejek będący właścicielem kolejki.

Nazwa RemoteQMgr

Nazwa menedżera kolejek będącego właścicielem.

XmitQName

Nazwa lokalnej kolejki transmisji, która jest używana podczas przekazywania komunikatów do innych menedżerów kolejek.

Więcej informacji na temat tych atrybutów znajduje się w sekcji [Atrybuty dla kolejek](#).


Jeśli wywołanie `MQINQ` jest używane dla lokalnej definicji kolejki zdalnej, menedżer kolejek zwraca tylko atrybuty definicji lokalnej, to znaczy nazwę kolejki zdalnej, nazwę zdalnego menedżera kolejek i nazwę kolejki transmisji, a nie atrybuty pasującej kolejki lokalnej w systemie zdalnym.

Patrz także: [Kolejki transmisji](#).

Kolejki aliasowe

Kolejka aliasowa to obiekt IBM MQ, którego można użyć w celu uzyskania dostępu do innej kolejki lub tematu. Oznacza to, że więcej niż jeden program może pracować z tą samą kolejką, uzyskując dostęp do niej przy użyciu różnych nazw.

Kolejka powstała w wyniku rozstrzygnięcia nazwy aliasu, znanej jako kolejka podstawowa, może być dowolnym z następujących typów kolejek, co jest obsługiwane przez platformę:

- Kolejka lokalna
- Lokalna definicja kolejki zdalnej.
-  Kolejka współużytkowana, która jest typem kolejki lokalnej, która jest dostępna tylko w systemie IBM MQ for z/OS.
- Predefiniowana kolejka
- Kolejka dynamiczna

Nazwa aliasu może również zostać przetłumaczona na dany temat. Jeśli aplikacja aktualnie umieszcza komunikaty w kolejce, można ją utworzyć w celu opublikowania w temacie przez podanie aliasu nazwy kolejki dla tematu. Nie jest konieczna zmiana kodu aplikacji.

Uwaga: Alias nie może zostać bezpośrednio przetłumaczony na inny alias w tym samym menedżerze kolejek.

Przykładem użycia kolejek aliasowych jest podanie przez administratora systemu różnych uprawnień dostępu do nazwy kolejki podstawowej (czyli kolejki, do której alias jest tłumaczący), oraz do nazwy kolejki aliasowej. Oznacza to, że program lub użytkownik może mieć uprawnienia do korzystania z kolejki aliasowej, ale nie do kolejki podstawowej.

Alternatywnie można ustawić autoryzację do zablokowania operacji put dla nazwy aliasu, ale zezwalaj na ich działanie dla kolejki podstawowej.

W niektórych aplikacjach użycie kolejek aliasowych oznacza, że administratorzy systemu mogą łatwo zmienić definicję obiektu kolejki aliasowej bez konieczności zmiany aplikacji.

Program IBM MQ sprawdza autoryzację przy użyciu aliasu, gdy programy próbują używać tej nazwy. Nie sprawdza, czy program jest uprawniony do uzyskania dostępu do nazwy, do której alias jest tłumaczona. Z tego powodu program może mieć uprawnienia dostępu do nazwy kolejki aliasowej, ale nie do przetłumaczanej nazwy kolejki.

Oprócz ogólnych atrybutów kolejki opisanych w [“Kolejki”](#) na stronie 15, kolejki aliasowe mają atrybut **BaseQName**. Jest to nazwa kolejki podstawowej, do której jest tłumaczona nazwa aliasu. Więcej informacji na temat pełniejszego opisu tego atrybutu można znaleźć w sekcji [BaseQName \(MQCHAR48\)](#).

Atrybuty *InhibitGet* i **InhibitPut** (patrz [“Kolejki”](#) na stronie 15) Kolejki aliasowe należą do aliasu. Na przykład, jeśli nazwa kolejki aliasowej ALIAS1 jest tłumaczona na podstawową nazwę kolejki podstawowej, to zahamowanie na ALIAS1 ma wpływ tylko na ALIAS1, a wartość BASE nie jest zablokowana. Jednak zahamowanie w BASE również ma wpływ na ALIAS1.

Atrybuty *DefPriority* i **DefPersistence** należą także do aliasu. Dlatego na przykład można przypisać różne domyślne priorytety do różnych aliasów tej samej kolejki podstawowej. Można również zmienić te priorytety bez konieczności zmiany aplikacji, które korzystają z aliasów.


Dynamiczne i modelowe kolejki

Informacje te udostępniają informacje na temat kolejek dynamicznych, właściwości tymczasowych i trwałych kolejek dynamicznych, użycia kolejek dynamicznych, niektórych zagadnień związanych z używaniem kolejek dynamicznych i kolejek modelowych.

Gdy program użytkowy zgłosi wywołanie MQOPEN w celu otwarcia kolejki modelowej, menedżer kolejek dynamicznie tworzy instancję kolejki lokalnej o tych samych atrybutach, co kolejka modelowa. W zależności od wartości pola *DefinitionType* w kolejce modelowej menedżer kolejek tworzy tymczasową lub trwałą kolejkę dynamiczną (patrz sekcja [Tworzenie kolejek dynamicznych](#)).

Właściwości tymczasowych kolejek dynamicznych

Tymczasowe kolejki dynamiczne mają następujące właściwości:

-  Nie mogą być kolejkami współużytkowanymi, dostępnymi z menedżerów kolejek w grupie współużytkowania kolejek.
Należy pamiętać, że grupy współużytkowania kolejek są dostępne tylko w produkcie IBM MQ for z/OS.
- Zawierają tylko komunikaty nietrwale.
- Są one nienaprawialne.
- Są one usuwane po uruchomieniu menedżera kolejek.
- Są one usuwane, gdy aplikacja, która wywołała wywołanie MQOPEN, które utworzyło kolejkę, zamknie kolejkę lub kończy działanie.
 - Jeśli w kolejce znajdują się zatwierdzone komunikaty, zostaną one usunięte.

- Jeśli w tej chwili wywołania MQGET, MQPUT lub MQPUT1 nie zostały zatwierdzone, kolejka jest oznaczana jako logicznie usuwana i jest fizycznie usuwana (po zatwierdzeniu tych wywołań) jako część przetwarzania zamknięcia lub gdy aplikacja kończy działanie.
- Jeśli kolejka jest używana w tym czasie (przez utworzenie lub inną aplikację), kolejka jest oznaczana jako logicznie usuwana i jest fizycznie usuwana tylko po zamknięciu przez ostatnią aplikację używając tej kolejki.
- Próby uzyskania dostępu do logicznie usuniętej kolejki (innej niż zamknięcie jej) kończą się niepowodzeniem z kodem przyczyny MQRC_Q_DELETED.
- Wywołania MQCO_NONE, MQCO_DELETE i MQCO_DELETE_PURGE są traktowane jako MQCO_NONE, jeśli zostały określone w wywołaniu MQCLOSE dla odpowiedniego wywołania MQOPEN, które utworzyło kolejkę.

Właściwości trwałych kolejek dynamicznych

State kolejki dynamiczne mają następujące właściwości:

- Zawierają komunikaty trwałe lub nietrwałe.
- Są one odtwarzalne w przypadku awarii systemu.
- Są one usuwane, gdy aplikacja (niekoniecznie ta, która wywołała wywołanie MQOPEN, które utworzyło kolejkę) pomyślnie zamknie kolejkę za pomocą opcji MQCO_DELETE lub MQCO_DELETE_PURGE.
 - Żądanie zamknięcia z opcją MQCO_DELETE nie powiedzie się, jeśli w kolejce nadal znajdują się komunikaty (zatwierdzone lub niezatwierdzone). Żądanie zamknięcia z opcją MQCO_DELETE_PURGE powiedzie się nawet wtedy, gdy w kolejce znajdują się zatwierdzone komunikaty (komunikaty są usuwane jako część końca), ale nie powiedzie się, jeśli istnieją niezatwierdzone wywołania MQGET, MQPUT lub MQPUT1 oczekujące na kolejkę.
 - Jeśli żądanie usunięcia zakończy się pomyślnie, ale kolejka ma być używana (przez utworzenie lub inną aplikację), kolejka jest oznaczona jako logicznie usunięta i jest fizycznie usuwana tylko po zamknięciu przez ostatnią aplikację używając tej kolejki.
- Nie są one usuwane, jeśli zostały zamknięte przez aplikację, która nie jest autoryzowana do usunięcia kolejki, chyba że aplikacja zamykające wygenerował wywołanie MQOPEN, które utworzyło kolejkę. Sprawdzanie autoryzacji jest wykonywane na podstawie identyfikatora użytkownika (lub alternatywnego identyfikatora użytkownika, jeśli określono wartość MQOO_ALTERNATE_USER_AUTHORITY), który został użyty do sprawdzenia poprawności odpowiedniego wywołania MQOPEN.
- Można je usunąć w taki sam sposób, jak w przypadku normalnej kolejki.

Użycie kolejek dynamicznych

Kolejek dynamicznych można używać w następujących celach:

- Aplikacje, które nie wymagają przechowywania kolejek po zakończeniu działania aplikacji.
- Aplikacje, które wymagają odpowiedzi na komunikaty, które mają być przetwarzane przez inną aplikację. Takie aplikacje mogą dynamicznie tworzyć kolejkę zwrotną, otwierając kolejkę modelową. Na przykład aplikacja kliencka może:
 1. Utwórz kolejkę dynamiczną.
 2. Podaj jego nazwę w polu **ReplyToQ** struktury deskryptora komunikatu w komunikacie żądania.
 3. Umieść żądanie w kolejce, która jest przetwarzana przez serwer.

Serwer może następnie umieścić komunikat odpowiedzi w kolejce odpowiedzi. Na koniec, klient może przetworzyć odpowiedź i zamknąć kolejkę odpowiedzi z opcją usuwania.

Uwagi dotyczące używania kolejek dynamicznych

Podczas korzystania z kolejek dynamicznych należy wziąć pod uwagę następujące kwestie:

- W modelu klient-serwer każdy klient musi utworzyć i użyć własnej dynamicznej kolejki odpowiedzi. Jeśli dynamiczna kolejka zwrotna jest współużytkowana przez więcej niż jednego klienta, usunięcie

kolejki odpowiedzi może być opóźnione z powodu niezatwierdzonego działania pozostałego w kolejce lub dlatego, że kolejka jest używana przez innego klienta. Dodatkowo kolejka może być oznaczona jako logicznie usunięta i niedostępna dla kolejnych żądań API (innych niż MQCLOSE).

- Jeśli środowisko aplikacji wymaga, aby kolejki dynamiczne były współużytkowane między aplikacjami, należy upewnić się, że kolejka jest zamknięta (z opcją usuwania) tylko wtedy, gdy wszystkie działania względem kolejki zostały zatwierdzone. Powinien to być ostatni użytkownik. Gwarantuje to, że usunięcie kolejki nie jest opóźnione i minimalizuje okres, w którym kolejka jest niedostępna, ponieważ została oznaczona jako logicznie usunięta.

Kolejki modelowe

Kolejka modelowa jest szablonem definicji kolejki, która jest używana podczas tworzenia kolejki dynamicznej.

Kolejkę lokalną można utworzyć dynamicznie z poziomu programu IBM MQ, nadając nazwę kolejki modelowej, która ma być używana jako szablon dla atrybutów kolejki. W tym momencie można zmienić niektóre atrybuty nowej kolejki. Nie można jednak zmienić **DefinitionType**. Jeśli na przykład wymagana jest kolejka trwała, należy wybrać kolejkę modelową z typem definicji ustawionym na wartość trwała. Niektóre aplikacje konwersacyjne mogą używać kolejek dynamicznych do przechowywania odpowiedzi na zapytania, ponieważ prawdopodobnie nie muszą utrzymywać tych kolejek po przetworzeniu odpowiedzi.

Nazwę kolejki modelowej określa się w *deskrytorze obiektu* (MQOD) wywołania MQOPEN. Za pomocą atrybutów kolejki modelowej menedżer kolejek dynamicznie tworzy kolejkę lokalną dla użytkownika.

Dla kolejki dynamicznej można podać nazwę (w całości) lub rdzeń nazwy (na przykład ABC) i pozwolić menedżerowi kolejek dodać do tego unikalną część lub pozwolić menedżerowi kolejek przypisać pełną unikalną nazwę. Jeśli menedżer kolejek przypisuje nazwę, umieszcza ją w strukturze MQOD.

Nie można wywołać wywołania MQPUT1 bezpośrednio do kolejki modelowej, ale można wydać komendę MQPUT1 do kolejki dynamicznej, która została utworzona przez otwarcie kolejki modelowej.

Komendy MQSET i MQINQ nie można wydać dla kolejki modelowej. Otwarcie kolejki modelowej za pomocą komendy MQOO_INQUIRE lub MQOO_SET powoduje, że kolejne wywołania MQINQ i MQSET są wykonywane względem dynamicznie utworzonej kolejki.

Atrybuty kolejki modelowej są podzbiorem elementów kolejki lokalnej. Więcej informacji na temat pełniejszego opisu zawiera sekcja [Atrybuty kolejek](#).

Sposoby definiowania kolejek

Kolejkę można utworzyć za pomocą komend IBM MQ (MQSC), PCF lub za pomocą interfejsów specyficznych dla platformy.

Kolejki do programu IBM MQ można zdefiniować za pomocą komendy MQSC [DEFINE](#) lub komendy PCF [Create Queue](#). Komendy określają typ kolejki i jej atrybuty. Na przykład obiekt kolejki lokalnej ma atrybuty określające, co się dzieje, gdy aplikacje odwołują się do tej kolejki w wywołaniach MQI. Przykładami atrybutów są:

- Określa, czy aplikacje mogą pobierać komunikaty z kolejki (funkcja GET jest włączona)
- Określa, czy aplikacje mogą umieszczać komunikaty w kolejce (włączone PUT)
- Określa, czy dostęp do kolejki jest wyłączony dla jednej aplikacji, czy jest współużytkowana przez aplikacje
- Maksymalna liczba komunikatów, które mogą być przechowywane w kolejce w tym samym czasie (maksymalna głębokość kolejki)
- Maksymalna długość komunikatów, które można umieścić w kolejce

Istnieją również różne interfejsy specyficzne dla platformy, których można używać do definiowania kolejek. Odsyłacze do dodatkowych informacji można znaleźć w sekcji *Informacje pokrewne*.

Pojęcia pokrewne

[Administrowanie za pomocą komend MQSC](#)

Zadania pokrewne

[Automatyzacja administrowania za pomocą komend PCF](#)

[Tworzenie i konfigurowanie menedżerów kolejek i obiektów za pomocą programu MQ Explorer](#)

[Zarządzanie produktem IBM MQ for IBM i za pomocą komend CL](#)

[Wysyłanie komend do programu IBM MQ for z/OS](#)

Kolejki używane do konkretnych celów przez produkt IBM MQ

Produkt IBM MQ używa niektórych kolejek lokalnych do celów szczegółowych związanych z jego działaniem.

Użytkownik musi zdefiniować te kolejki przed użyciem ich przez program IBM MQ .

Kolejki inicjuj

Kolejki inicjuj są kolejkami, które są używane w wyzwalaniu. Menedżer kolejek umieszcza komunikat wyzwalacza w kolejce inicjuj, gdy wystąpi zdarzenie wyzwalające. Zdarzenie wyzwalające jest logiczną kombinacją warunków, które są wykrywane przez menedżer kolejek. Na przykład zdarzenie wyzwalające może być generowane, gdy liczba komunikatów w kolejce osiągnie predefiniowaną głębokość. To zdarzenie powoduje, że menedżer kolejek umieści komunikat wyzwalacza w określonej kolejce inicjuj. Ten komunikat wyzwalacza jest pobierany przez *monitor wyzwalacza*, specjalną aplikację, która monitoruje kolejkę inicjującą. Następnie monitor wyzwalacza uruchamia program użytkowy określony w komunikacie wyzwalacza.

Jeśli menedżer kolejek ma używać wyzwalania, dla tego menedżera kolejek musi być zdefiniowana co najmniej jedna kolejka inicjująca. Więcej informacji na ten temat zawiera sekcja [Zarządzanie obiektami dla wyzwalania, runmqtrmi Uruchamianie aplikacji produktu IBM MQ za pomocą wyzwalaczy](#) .

Kolejki transmisji

Kolejki transmisji są kolejkami, które tymczasowo przechowują komunikaty przeznaczone dla zdalnego menedżera kolejek. Należy zdefiniować co najmniej jedną kolejkę transmisji dla każdego zdalnego menedżera kolejek, do którego lokalny menedżer kolejek ma wysyłać komunikaty bezpośrednio. Te kolejki są również używane w administrowaniu zdalnym. Patrz sekcja [Zdalne administrowanie z lokalnego menedżera kolejek](#). Informacje na temat korzystania z kolejek transmisji w kolejkach rozproszonych znajdują się w sekcji [Techniki kolejkowania rozproszonego IBM MQ](#).

Każdy menedżer kolejek może mieć domyślną kolejkę transmisji. Jeśli menedżer kolejek, który nie jest częścią klastra, umieszcza komunikat w kolejce zdalnej, domyślnym działaniem jest użycie domyślnej kolejki transmisji. Jeśli istnieje kolejka transmisji o takiej samej nazwie, jak nazwa docelowego menedżera kolejek, komunikat jest umieszczany w tej kolejce transmisji. Jeśli istnieje definicja aliasu menedżera kolejek, w której parametr **RQMNAME** jest zgodny z docelowym menedżerem kolejek, a parametr **XMITQ** jest określony, komunikat jest umieszczany w kolejce transmisji nazwanej przez **XMITQ**. Jeśli nie ma parametru **XMITQ** , komunikat jest umieszczany w kolejce lokalnej o nazwie określonej w komunikacie.

Kolejki transmisji klastra

Każdy menedżer kolejek w klastrze ma kolejkę transmisji klastra o nazwie SYSTEM . CLUSTER . TRANSMIT . QUEUEi modelową kolejkę transmisji klastra, SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE. Definicje tych kolejek są tworzone domyślnie podczas definiowania menedżera kolejek. Jeśli atrybut menedżera kolejek **DEFCLXQ** jest ustawiony na wartość CHANNEL, automatycznie tworzona jest stała dynamiczna kolejka transmisji klastra dla każdego utworzonego kanału nadawczego klastra. Kolejki są nazywane SYSTEM . CLUSTER . TRANSMIT . ChannelName. Kolejki transmisji klastra można również zdefiniować ręcznie.

Menedżer kolejek, który jest częścią klastra, wysyła komunikaty w jednej z tych kolejek do innych menedżerów kolejek, które znajdują się w tym samym klastrze.

W trakcie rozstrzygnięcia nazw kolejka transmisji klastra ma pierwszeństwo przed domyślną kolejką transmisji, a konkretna kolejka transmisji klastra ma pierwszeństwo przed SYSTEM . CLUSTER . TRANSMIT . QUEUE.

Kolejki niedostarczane

Kolejka niedostarczonych komunikatów (niedostarczonych komunikatów) jest kolejką, w której zapisywane są komunikaty, których nie można kierować do właściwych miejsc docelowych. Komunikat nie może być kierowany, gdy na przykład kolejka docelowa jest pełna. Podana kolejka niedostarczonych komunikatów nosi nazwę `SYSTEM.DEAD.LETTER.QUEUE`.

Dla kolejkowania rozproszonego zdefiniuj kolejkę niedostarczonych komunikatów w każdym z zaangażowanych menedżerów kolejek.

Kolejki komend

Kolejka komend, `SYSTEM.ADMIN.COMMAND.QUEUE`, jest kolejką lokalną, do której odpowiednio autoryzowane aplikacje mogą wysyłać komendy MQSC do przetwarzania. Te komendy są następnie pobierane przez komponent IBM MQ o nazwie serwer komend. Serwer komend sprawdza poprawność komend, przekazuje poprawne dane do przetwarzania przez menedżer kolejek i zwraca wszystkie odpowiedzi do odpowiedniej kolejki odpowiedzi.

Kolejka komend jest tworzona automatycznie dla każdego menedżera kolejek po utworzeniu tego menedżera kolejek.

Kolejki odpowiedzi

Gdy aplikacja wysyła komunikat żądania, aplikacja, która odbiera komunikat, może odesłać komunikat odpowiedzi do aplikacji wysyłającej. Ten komunikat jest umieszczany w kolejce, nazywanych kolejką zwrotną, która jest zwykle kolejką lokalną dla aplikacji wysyłającej. Nazwa kolejki odpowiedzi jest określona przez aplikację wysyłającą jako część deskryptora komunikatu.

kolejki zdarzeń

Zdarzenia instrumentacji mogą być używane do monitorowania menedżerów kolejek niezależnie od aplikacji MQI.

Gdy wystąpi zdarzenie instrumentacji, menedżer kolejek umieszcza komunikat zdarzenia w kolejce zdarzeń. Ten komunikat może następnie zostać odczytany przez aplikację monitoringową, która może informować administratora lub inicjować pewne działania zaradcze, jeśli zdarzenie wskazuje na problem.

Uwaga: Zdarzenia wyzwajające są różne od zdarzeń instrumentacji. Zdarzenia wyzwajające nie są powodowane przez te same warunki i nie generują komunikatów zdarzeń.

Więcej informacji na temat zdarzeń instrumentacji zawiera sekcja [Zdarzenia instrumentacji](#).

IBM MQ Menedżery kolejek

Wprowadzenie do *menedżerów kolejek* i usług kolejkowania, które udostępniają aplikacjom.

Program musi mieć połączenie z menedżerem kolejek, zanim będzie mógł korzystać z usług tego menedżera kolejek. Program może nawiązać to połączenie jawnie (za pomocą wywołania `MQCONN` lub `MQCONNX`) lub połączenie może zostać nawiązane niejawnie (zależy to od platformy i środowiska, w którym działa program).

Menedżery kolejek udostępniają aplikacjom usługi kolejkowania i zarządzają kolejkami, które należą do nich. Menedżer kolejek zapewnia następujące działania:

- Atrybuty obiektu są zmieniane zgodnie z otrzymaną komendami.
- Zdarzenia specjalne, takie jak zdarzenia wyzwajające lub zdarzenia instrumentacji, są generowane, gdy spełnione są odpowiednie warunki.
- Komunikaty są umieszczane w poprawnej kolejce, zgodnie z żądaniem aplikacji wywołując wywołanie `MQPUT`. Jeśli nie można tego dokonać, aplikacja jest informowana o tym, że nie można jej wykonać, a także podano odpowiedni kod przyczyny.

Każda kolejka należy do pojedynczego menedżera kolejek i jest ona *kolejką lokalną* tego menedżera kolejek. Menedżer kolejek, z którym połączona jest aplikacja, jest *lokalnym menedżerem kolejek* dla tej aplikacji. W przypadku aplikacji kolejki należące do jego lokalnego menedżera kolejek są kolejkami lokalnymi.


Kolejka zdalna to kolejka, która należy do innego menedżera kolejek. *Menedżer kolejek zdalnych* jest dowolnym menedżerem kolejek innym niż lokalny menedżer kolejek. Zdalny menedżer kolejek może istnieć na komputerze zdalnym w sieci lub może istnieć na tym samym komputerze, co lokalny menedżer kolejek. Produkt IBM MQ obsługuje wiele menedżerów kolejek na tym samym komputerze.

Obiekt menedżera kolejek może być używany w niektórych wywołaniach MQI. Na przykład można uzyskać informacje na temat atrybutów obiektu menedżera kolejek przy użyciu wywołania MQI MQINQ.


Atrybuty menedżerów kolejek

Powiązany z każdym menedżerem kolejek jest zestaw atrybutów (lub właściwości), które definiują jego parametry. Niektóre atrybuty menedżera kolejek są naprawiane podczas jego tworzenia. Inne atrybuty można zmienić za pomocą komend IBM MQ. Użytkownik może zapytać o wartości wszystkich atrybutów, z wyjątkiem tych, które są używane do szyfrowania TLS (Transport Layer Security), za pomocą wywołania MQINQ.

Do stałych atrybutów należą:

- Nazwa menedżera kolejek
- Platforma, na której działa menedżer kolejek (na przykład Windows)
- Poziom komend sterujących systemem obsługiwanych przez menedżer kolejek
- Maksymalny priorytet, który można przypisać do komunikatów przetwarzanych przez menedżer kolejek
- Nazwa kolejki, do której programy mogą wysyłać komendy IBM MQ
- Maksymalna długość komunikatów, które może przetworzyć menedżer kolejek  (naprawionej tylko w produkcie IBM MQ for z/OS)
- Określa, czy menedżer kolejek obsługuje synchronizowanie, gdy programy są wstawiane i otrzymują komunikaty.

Atrybuty *zmienialne* obejmują:

- Opis tekstowy menedżera kolejek
- Identyfikator zestawu znaków używany przez menedżer kolejek w przypadku łańcuchów znaków podczas przetwarzania wywołań MQI
- Przedział czasu używany przez menedżer kolejek w celu ograniczenia liczby komunikatów wyzwacza.
-  Przedział czasu używany przez menedżer kolejek w celu określenia, jak często kolejki mają być skanowane pod kątem komunikatów, które utraciły ważność (tylko system IBM MQ for z/OS)
- Nazwa kolejki niedostarczonych komunikatów menedżera kolejek (niedostarczonych komunikatów).
- Nazwa domyślnej kolejki transmisji menedżera kolejek
- Maksymalna liczba otwartych uchwytów dla dowolnego połączenia
- Włączanie i wyłączenie różnych kategorii raportowania zdarzeń
- Maksymalna liczba niezatwierdzonych komunikatów w jednostce pracy

Menedżery kolejek i zarządzanie obciążeniem


Istnieje możliwość skonfigurowania klastra menedżerów kolejek, który ma więcej niż jedną definicję dla tej samej kolejki (na przykład menedżery kolejek w klastrze mogą być klonami wzajemnie). Komunikaty dla konkretnej kolejki mogą być obsługiwane przez dowolny menedżer kolejek, który udostępnia instancję kolejki. Algorytm zarządzania obciążeniem decyduje o tym, który menedżer kolejek obsługuje tę wiadomość, a następnie rozprzestrzenia obciążenie między menedżerami kolejek. W celu uzyskania dalszych informacji należy zapoznać się z informacjami w sekcji [Algorytm zarządzania obciążeniem klastra](#).

Definicje procesów

Obiekty definicji procesów umożliwiają uruchomienie aplikacji bez konieczności interwencji operatora przez zdefiniowanie atrybutów aplikacji do użycia przez menedżera kolejek.

Obiekt definicji procesu definiuje aplikację, która jest uruchamiana w odpowiedzi na zdarzenie wyzwalające w menedżerze kolejek produktu IBM MQ. Atrybuty definicji procesu zawierają identyfikator aplikacji, typ aplikacji oraz dane specyficzne dla aplikacji. Więcej informacji na ten temat zawiera sekcja *Kolejki inicjujące* w podręczniku [“Kolejki używane do konkretnych celów przez produkt IBM MQ”](#) na stronie 23.

Aby umożliwić uruchamianie aplikacji bez konieczności interwencji operatora, zgodnie z opisem w sekcji *Uruchamianie aplikacji IBM MQ przy użyciu wyzwalaczy*, atrybuty aplikacji muszą być znane menedżerowi kolejek. Atrybuty te są zdefiniowane w *obiekcie definicji procesu*.

Atrybut **ProcessName** jest ustalany podczas tworzenia obiektu. Można jednak zmienić inne atrybuty za pomocą komend IBM MQ .  Alternatywnie w systemie z/OS można używać operacji IBM MQ for z/OS i paneli sterujących.

Istnieje możliwość uzyskania informacji o wartościach wszystkich atrybutów za pomocą komendy [MQINQ-Inquire object attributes](#)(Atrybuty obiektu MQINQ).

Odsyłacze pokrewne


[Atrybuty definicji procesów](#)

Listy nazw

Lista nazw to obiekt IBM MQ, który zawiera listę nazw klastrów, nazw kolejek lub nazw obiektów informacji uwierzytelniających. W klastrze może być używany do identyfikowania listy klastrów, dla których menedżer kolejek przechowuje repozytoria.

Lista nazw to obiekt IBM MQ, który zawiera listę innych obiektów IBM MQ. Zazwyczaj listy nazw są używane przez aplikacje, na przykład monitory wyzwalacza, w przypadku których służą do identyfikowania grupy kolejek. Zaletą użycia listy nazw jest to, że jest ona obsługiwana niezależnie od aplikacji. Można ją zaktualizować bez zatrzymywania żadnej z aplikacji, które go używają. Ponadto, jeśli jedna aplikacja nie powiedzie się, lista nazw nie zostanie naruszona, a inne aplikacje będą mogły kontynuować korzystanie z niej.

Listy nazw są również używane z klastrami menedżerów kolejek w celu zachowania listy klastrów, do których odnosi się więcej niż jeden obiekt IBM MQ.

Za pomocą komend MQSC można definiować i modyfikować listy nazw.  Alternatywnie w systemie z/OS można używać operacji IBM MQ for z/OS i paneli sterujących

Programy mogą korzystać z interfejsu MQI, aby dowiedzieć się, które kolejki znajdują się na tych listach nazw. Odpowiedzialność projektanta aplikacji i administratora systemu jest odpowiedzialna za organizację list nazw.

Odsyłacze pokrewne

[Atrybuty dla list nazw](#)

[DEFINIUJ LISTĘ NAZW](#)

Obiekty informacji uwierzytelniających

Obiekt informacji uwierzytelniającej udostępnia definicje wymagane do sprawdzania odwołań certyfikatów.

Obiekt informacji uwierzytelniających menedżera kolejek stanowi część wsparcia IBM MQ dla protokołu TLS (Transport Layer Security). Udostępnia on definicje wymagane do sprawdzania unieważnionych certyfikatów. Ośrodki certyfikacji unieważnią certyfikaty, które nie mogą być już zaufane.

Aby zdefiniować obiekt informacji uwierzytelniających, można użyć komendy MQSC **DEFINE AUTHINFO**. Więcej informacji na temat atrybutów obiektów informacji uwierzytelniających zawiera sekcja **DEFINE AUTHINFO**.

Z obiektem informacji uwierzytelniającej można używać następujących komend sterujących IBM MQ :

- **setmqaut** (uprawnienie do nadawania lub odbierania uprawnień)
- **dspmqaut** (wyświetlanie autoryzacji obiektu)
- **dmpmqaut** (autoryzacje zrzutu)
- **rcrmqobj** (ponowne tworzenie obiektu)
- **rcdmqimg** (obraz nośnika rekordu)
- **dspmqfls** (wyświetlanie nazw plików)

Przegląd protokołu TLS oraz korzystanie z obiektów informacji uwierzytelniających zawiera sekcja [Protokoły zabezpieczeń TLS w produkcie IBM MQ](#).

Pojęcia pokrewne

[Pojęcia związane z protokołem TLS \(Transport Layer Security\)](#)

Obiekty informacji o komunikacji

Funkcja rozsyłania grupowego produktu IBM MQ umożliwia niezawodne rozsyłanie grupowe komunikatów z zachowaniem małych opóźnień i wysokiego zwielokrotnienia. Obiekt informacji o komunikacji (COMMINFO) jest wymagany dla transmisji w trybie rozsyłania grupowego.

Obiekt COMMINFO to obiekt IBM MQ, który zawiera atrybuty powiązane z transmisją rozsyłania grupowego. Więcej informacji na temat tych atrybutów zawiera sekcja [DEFINE COMMINFO](#). Więcej informacji na temat tworzenia obiektu COMMINFO zawiera sekcja [Pierwsze kroki z rozsyłaniem grupowym](#).

Pojęcia pokrewne

[“IBM MQ Rozsyłanie grupowe” na stronie 111](#)

Funkcja rozsyłania grupowego w produkcie IBM MQ umożliwia niezawodne rozsyłanie grupowe komunikatów z zachowaniem małych opóźnień i wysokiego zwielokrotnienia.

Kanały

Kanał jest logicznym łączem komunikacyjnym, używanym przez rozproszone menedżery kolejek, między serwerem IBM MQ MQI client i serwerem IBM MQ, lub między dwoma serwerami IBM MQ.

Kanały to obiekty, które udostępniają ścieżkę komunikacji z jednego menedżera kolejek do innego. Kanały służą do przenoszenia komunikatów z jednego menedżera kolejek do innego, a także do zabezpieczenia aplikacji z bazowych protokołów komunikacyjnych. Menedżery kolejek mogą istnieć w tym samym systemie lub w różnych systemach na tej samej platformie lub na różnych platformach. Wysyłane komunikaty mogą pochodzić z wielu lokalizacji:

- Programy użytkowe napisane przez użytkownika, które przesyłają dane z jednego węzła do drugiego.
- Aplikacje administracyjne napisane przez użytkownika, które korzystają z komend PCF lub MQAI.
- IBM MQ Explorer.
- Menedżery kolejek, które wysyłają komunikaty zdarzeń instrumentacji do innego menedżera kolejek.
- Menedżery kolejek, które wysyłają zdalne komendy administracyjne do innego menedżera kolejek. Na przykład za pomocą komend MQSC lub administrative REST API.

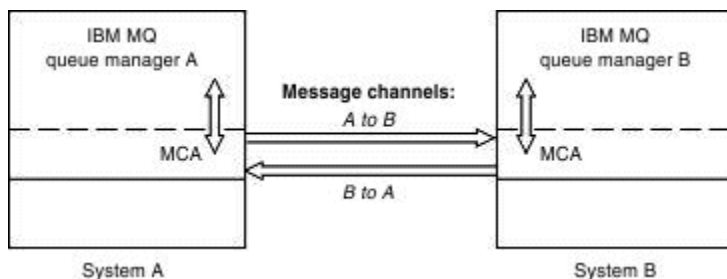
Kanał ma dwie definicje: po jednej na każdym końcu połączenia. Aby menedżery kolejek mogły komunikować się ze sobą, należy zdefiniować jeden obiekt kanału w menedżerze kolejek, który ma wysyłać komunikaty, a inny, uzupełniający, w menedżerze kolejek, który ma je odbierać. Ta sama *nazwa kanału* musi być używana na każdym końcu połączenia, a używany *typ kanału* musi być zgodny.

Istnieją trzy kategorie kanałów w produkcie IBM MQ, które mają różne typy kanałów w następujących kategoriach:

- Kanały komunikatów, które są jednokierunkowe, i przesyłają komunikaty z jednego menedżera kolejek do innego.
- Kanały MQI, które są dwukierunkowe i przesyłają wywołania MQI z IBM MQ MQI client do menedżera kolejek, a odpowiedzi z menedżera kolejek do klienta IBM MQ .
- Kanały AMQP, które są dwukierunkowe i łączą klienta AMQP z menedżerem kolejek na komputerze serwera. Produkt IBM MQ używa kanałów AMQP do przesyłania wywołań i odpowiedzi AMQP między aplikacjami AMQP i menedżerami kolejek

Kanały komunikatów

Celem kanału komunikatów jest przesyłanie komunikatów z jednego menedżera kolejek do innego. Kanały komunikatów nie są wymagane przez środowisko serwera klienta.



Rysunek 2. Kanały komunikatów między dwoma menedżerami kolejek

Kanał komunikatów jest łączem jednokierunkowy. Jeśli zdalny menedżer kolejek ma odpowiadać na komunikaty wysyłane przez lokalny menedżer kolejek, należy skonfigurować drugi kanał w celu wysyłania odpowiedzi z powrotem do lokalnego menedżera kolejek.

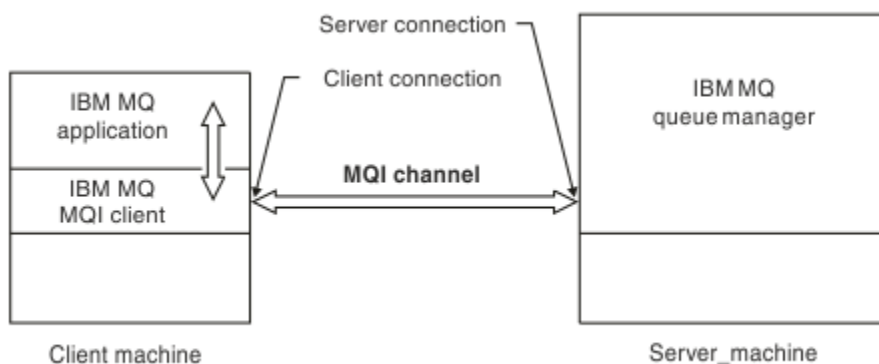
Kanał komunikatów łączy dwa menedżery kolejek przy użyciu *agentów kanałów komunikatów* (MCAs). Na każdym końcu kanału znajduje się agent kanału komunikatów. Można zezwolić agentowi MCA na przesyłanie komunikatów przy użyciu wielu wątków. Ten proces jest znany jako *pipelining*. Pipelining umożliwia agentowi MCA bardziej efektywne przesyłanie komunikatów, poprawiając wydajność kanałów. Więcej informacji na temat pipelining zawiera sekcja [Atrybuty kanałów](#).

Więcej informacji na temat kanałów zawiera sekcja [Kanał-wywołania wyjścia i struktury danych](#), a także sekcja ["Rozproszone komponenty kolejkowania"](#) na stronie 47.

Kanały MQI

Kanał MQI łączy serwer IBM MQ MQI client z menedżerem kolejek na komputerze serwera i jest ustanawiany podczas wydawania wywołania MQCONN lub MQCONNX z aplikacji IBM MQ MQI client .

Jest to łącze dwukierunkowe i jest używane tylko do przesyłania wywołań i odpowiedzi MQI, w tym wywołania MQPUT , które zawierają dane komunikatów, oraz wywołania MQGET , które powodują zwrócenie danych komunikatu. Istnieją różne sposoby tworzenia definicji kanału i korzystania z nich (patrz sekcja [Definiowanie kanałów MQI](#)).



Rysunek 3. Połączenie z klientem i połączenie z serwerem w kanale MQI

z/OS Kanał MQI może być używany do łączenia klienta z pojedynczym menedżerem kolejek lub do menedżera kolejek, który jest częścią grupy współużytkownia kolejek (patrz sekcja [łączenie klienta z grupą współużytkownia kolejek](#)).

Istnieją dwa typy kanałów dla definicji kanałów MQI. Definiują one dwukierunkowy kanał MQI.

Kanał połączenia klienckiego

Ten typ jest przeznaczony dla IBM MQ MQI client.

Kanał połączenia serwera

Ten typ jest przeznaczony dla serwera, na którym działa menedżer kolejek, z którym aplikacja IBM MQ działająca w środowisku IBM MQ MQI client ma komunikować się.

Kanały protokołu AMQP

Multi

Istnieje tylko jeden typ kanału AMQP.

Do nawiązywania połączenia między aplikacją przesyłania komunikatów AMQP i menedżerem kolejek używany jest kanał, umożliwiając aplikacji wymianę komunikatów z aplikacjami produktu IBM MQ. Kanał AMQP umożliwia utworzenie aplikacji przy użyciu produktu MQ Light, a następnie wdrożenie jej jako aplikacji korporacyjnej, dzięki czemu mogą być wykorzystywane funkcje korporacyjne udostępniane przez produkt IBM MQ.

Definicje kanałów

Opisy poszczególnych typów kanałów znajdują się w sekcji [“Definicje kanałów”](#) na stronie 30 .

Pojęcia pokrewne

[“Rozproszone kolejkowanie i klastry”](#) na stronie 43

Kolejkowanie rozproszone oznacza wysyłanie komunikatów z jednego menedżera kolejek do innego. Odbierający menedżer kolejek może znajdować się na tej samej lub innej maszynie; w pobliżu lub po drugiej stronie świata. Może być uruchomiony na tej samej platformie co lokalny menedżer kolejek lub może znajdować się na dowolnej z platform obsługiwanych przez produkt IBM MQ. Można ręcznie zdefiniować wszystkie połączenia w rozproszonym środowisku kolejkowania lub utworzyć klastrer, a następnie IBM MQ zdefiniować wiele szczegółów połączenia dla użytkownika.

[Interfejs kolejki komunikatów-przegląd](#)

Zadania pokrewne

[Administrowanie zdalnymi obiektami IBM MQ](#)

[Zatrzymywanie kanałów MQI](#)

Odsyłacze pokrewne

[Wywołania wyjścia kanału i struktury danych](#)

Definicje kanałów

Tabele opisujące różne typy kanałów komunikatów i kanałów MQI, które są używane przez produkt IBM MQ.

W kontekście kanałów komunikatów słowo „kanał” często jest używane jako synonim definicji kanału. Najczęściej można zorientować się na podstawie kontekstu, czy omawiany jest kanał kompletny, który ma dwa zakończenia, czy też definicja kanału, która ma tylko jedno zakończenie.

Kanały komunikatów

Definicje kanałów komunikatów mogą być następującego typu:

| Typ definicji kanału komunikatów | Opis |
|---|--|
| Nadawca | Kanał nadawczy jest to kanał komunikatów, za pomocą którego menedżer kolejek wysyła komunikaty do innych menedżerów kolejek. Aby wysłać komunikaty przy użyciu kanału nadawcy, należy także utworzyć na innym menedżerze kolejek kanał odbiorcy, który ma taką samą nazwę, co kanał nadawcy. Jeśli został zaimplementowany mechanizm wywołania zwrotnego, można używać kanału nadawczego z kanałem requestera. |
| Serwer | Kanał serwera jest to kanał komunikatów, za pomocą którego menedżer kolejek wysyła komunikaty do innych menedżerów kolejek. Aby wysłać komunikaty przy użyciu kanału serwera, należy także utworzyć na innym menedżerze kolejek kanał odbiorcy, który ma taką samą nazwę, co kanał serwera. Poza tym możliwe jest używanie kanałów serwera z kanałami requestera. W takim przypadku definicja kanału requestera na pozostałym końcu kanału żąda uruchomienia definicji kanału serwera. Serwer wysyła komunikaty do requestera. Serwer może także inicjować komunikację pod warunkiem, że zna nazwę połączenia kanału partnera. |
| Odbiornik | Kanał odbiorczy jest to kanał komunikatów, za pomocą którego menedżer kolejek odbiera komunikaty z innych menedżerów kolejek. Aby odbierać komunikaty przy użyciu kanału odbiorcy, należy także utworzyć na innym menedżerze kolejek kanał nadawcy lub kanał serwera, który ma taką samą nazwę, co kanał odbiorcy. |

| Typ definicji kanału komunikatów | Opis |
|----------------------------------|---|
| Requester | <p>Kanał requestera to kanał komunikatów używany przez menedżer kolejek w celu odbierania komunikatów od innych menedżerów kolejek. Kanał requestera może utworzyć żądanie uruchomienia kierowane do zdefiniowanego kanału partnera na zdalnym elemencie końcowym połączenia. Jeśli kanałem partnera jest kanał serwera, kanał serwera zatwierdza żądanie uruchomienia i rozpoczyna wysyłanie komunikatów, z kolejki przesyłania określonej w definicji kanału serwera do kanału requestera. Jeśli kanał partnera jest kanałem nadawczym, kanał nadawczy akceptuje żądanie uruchomienia, ale później zamyka połączenie z requesterem. Następnie, kanał nadawczy uruchamia się, negocjuje sesję z kanałem requestera partnera i rozpoczyna wysyłanie komunikatów z kolejki przesyłania określonej w definicji kanału nadawczego. Ten drugi przypadek zasadniczo zapewnia mechanizm wywołania zwrotnego, ponieważ kanał requestera tworzy żądanie wywołania zwrotnego kierowane do kanału nadawczego.</p> |
| Nadawca klastra | <p>Definicja kanału wysyłającego klastry (CLUSDR) definiuje wysyłający koniec kanału, z którego menedżer kolejek klastra może wysłać informacje klastra do jednego z pełnych repozytoriów. Kanał wysyłający klastry jest używany w celu powiadomienia repozytorium o wszelkich zmianach w statusie menedżera kolejek, na przykład o dodaniu lub usunięciu kolejki. Jest on również używany w celu przesyłania komunikatów. Menedżery pełnego repozytorium mają kanały wysyłające klastry, które wskazują sobie nawzajem. Używają ich do przekazywania między sobą informacji o zmianach w statusie klastra. Nie ma znaczenia, które pełne repozytorium wskazuje na definicję kanału CLUSDR menedżera kolejek. Po nawiązaniu kontaktu początkowego kolejne obiekty menedżera kolejek klastra są definiowane automatycznie zgodnie z wymaganiami tak, aby menedżer kolejek mógł wysłać informacje o klastrze do każdego pełnego repozytorium oraz komunikaty do każdego menedżera kolejek.</p> |
| Odbiorca klastra | <p>Definicja kanału odbierającego klastry (CLUSRCVR) definiuje odbierający koniec kanału, na którym menedżer kolejek klastra może odbierać komunikaty od innych menedżerów kolejek w klastrze. Kanałem odbierającym klastry mogą być również przesyłane informacje dotyczące klastra przeznaczone dla repozytorium. Definiując kanał odbierający klastry, menedżer kolejek wskazuje innym menedżerom kolejek klastra, że jest dostępny, aby otrzymywać komunikaty. Przynajmniej jeden kanał odbierający klastry jest potrzebny dla każdego menedżera kolejek klastra.</p> |

Należy zdefiniować obydwie końce każdego kanału, aby każdy koniec kanału miał definicję kanału. Obydwie końce kanału muszą być typami zgodnymi.

Możliwe są następujące kombinacje definicji kanałów:

- Kanał nadawcy-Kanał odbiorcy
- Kanał serwera-Kanał odbiorcy
- Kanał requestera-Kanał serwera
- Kanał requestera-Kanał nadawcy (wywołanie zwrotne)
- Kanał wysyłający klastry-Kanał odbierający klastry

Agenci kanału komunikatów

Każda utworzona definicja kanału należy do konkretnego menedżera kolejek. Menedżer kolejek może mieć kilka kanałów tego samego lub różnego typu. Na każdym końcu kanału znajduje się program, którym jest agent kanału komunikatów (MCA). Na jednym końcu kanału wywołujący agent MCA pobiera komunikaty z kolejki transmisji i wysyła je za pośrednictwem kanału. Na drugim końcu kanału odpowiadający agent MCA odbiera komunikaty i dostarcza je do zdalnego menedżera kolejek.

Program wywołujący MCA może zostać przypisany do kanału nadawcy, kanału serwera lub kanału requestera. Program odpowiadający MCA może zostać przypisany do dowolnego typu kanału komunikatów.

Produkt IBM MQ obsługuje następujące kombinacje typów kanałów na dwóch końcach połączenia:

| Program wywołujący | | Kierunek przepływu informacji | Program odpowiadający | |
|--------------------|-------------------------|--|-------------------------|------------|
| Typ kanału | Wymagane nastuchiwanie? | | Wymagane nastuchiwanie? | Typ kanału |
| Nadawca | Nie | Program wywołujący do programu odpowiadającego | Tak | Odbiornik |
| Serwer | Nie | Program wywołujący do programu odpowiadającego | Tak | Odbiornik |
| Serwer | Nie | Program wywołujący do programu odpowiadającego | Tak | Requester |
| Requester | Nie | Program odpowiadający do programu wywołującego | Tak | Serwer |
| Requester | Tak | Program odpowiadający do programu wywołującego | Tak | Nadawca |

Kanały MQI

Kanały MQI mogą być następującego typu:




| Typ kanału MQI | Opis |
|--------------------------|---|
| Kanał połączenia serwera | Kanał połączenia serwera jest dwukierunkowym kanałem MQI, który jest używany w celu nawiązania połączenia między klientem IBM MQ i serwerem IBM MQ. Kanał połączenia serwera to koniec serwera kanału. |
| Połączenie klienta | Kanał połączenia klienta jest dwukierunkowym kanałem MQI, który jest używany w celu nawiązania połączenia między klientem IBM MQ i serwerem IBM MQ. Program IBM MQ Explorer także używa połączeń klienta, aby nawiązać połączenie ze zdalnymi menedżerami kolejek. Kanał połączenia klienta to koniec klienta kanału. Podczas tworzenia kanału połączenia klienta tworzony jest plik na komputerze udostępniającym menedżera kolejek. Plik połączenia klienta należy następnie skopiować na komputer klienta produktu IBM MQ. |

Komunikacja

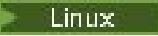
Program IBM MQ MQI clients używa kanałów MQI do komunikowania się z serwerem.

Definicja kanału musi zostać utworzona zarówno na serwerze IBM MQ MQI client , jak i na końcu połączenia serwera. Sposób tworzenia definicji kanałów jest opisany w sekcji [Definiowanie kanałów MQI](#).

Możliwe protokoły transmisji są przedstawione w poniższej tabeli:

| Tabela 1. Protokoły transmisji dla kanałów MQI | | | | |
|---|------------------|--------|---------|-----|
| Platforma klienta | LU 6.2 | TCP/IP | NetBIOS | SPX |
|  IBM i | | Tak | | |
|  Systemy AIX and Linux | Tak ¹ | Tak | | |
|  Windows | Tak | Tak | Tak | Tak |

Uwaga:

1.  Produkt LU6.2 nie jest obsługiwany na następujących platformach:

- Linux (platforma POWER)
- Linux (platformax86-64)
- Linux (platformazSeries s390x)

Protokoły transmisji-kombinacja platform IBM MQ MQI client i serwerów przedstawia możliwe kombinacje platform IBM MQ MQI client i serwerów, które korzystają z tych protokołów transmisji.

Aplikacja IBM MQ na serwerze IBM MQ MQI client może używać wszystkich wywołań MQI w taki sam sposób, jak w przypadku, gdy menedżer kolejek jest lokalny. Produkt **MQCONN** lub produkt **MQCONNX** powiąże aplikację IBM MQ z wybranym menedżerem kolejek, tworząc *uchwyt połączenia*. Inne wywołania używające tego uchwytu połączenia są następnie przetwarzane przez połączonego menedżera kolejek. Komunikacja IBM MQ MQI client wymaga aktywnego połączenia między klientem a serwerem, w przeciwieństwie do komunikacji między menedżerami kolejek, która jest niezależna od połączenia i jest niezależna od czasu.

Protokół transmisji jest określany przy użyciu definicji kanału i nie ma wpływu na aplikację. Na przykład aplikacja Windows może łączyć się z jednym menedżerem kolejek za pośrednictwem protokołu TCP/IP i z innym menedżerem kolejek za pośrednictwem protokołu NetBIOS.

Zagadnienia związane z wydajnością

Używany protokół transmisji może mieć wpływ na wydajność systemu klienta i serwera IBM MQ . W przypadku połączenia modemowego za pomocą powolnej linii telefonicznej zalecane może być użycie kompresji kanału IBM MQ .

Kanały połączenia klienckiego

Kanały połączeń klienta to obiekty, które udostępniają ścieżkę komunikacji z IBM MQ MQI client do menedżera kolejek.

Kanały połączenia klienckiego są używane w rozproszonej kolejkowaniu w celu przenoszenia komunikatów między menedżerem kolejek a klientem. Chronią aplikacje od podstawowych protokołów komunikacyjnych. Klient może istnieć na tej samej lub innej platformie z menedżera kolejek.

Zadania pokrewne

[Konfigurowanie połączeń między serwerem a klientem](#)

Klasy pamięci masowej

Klasa pamięci odwzorowuje jedną lub więcej kolejek do zbioru stron.

Oznacza to, że komunikaty dla tej kolejki są przechowywane (z zastrzeżeniem buforowania) w tym zestawie stron.

Klasy pamięci masowej są obsługiwane tylko w systemie IBM MQ for z/OS.

Więcej informacji na temat klas pamięci masowej zawiera publikacja [Planowanie w systemie z/OS](#).


Procesy następujące

Nastuchiwanie to procesy, które akceptują żądania sieciowe od innych menedżerów kolejek lub aplikacji klienckich, a także uruchamiają powiązane kanały.

Procesy nastuchiwania mogą być uruchamiane za pomocą komendy sterującej **runmq1sr** .

Obiekty nastuchiwania to obiekty programu IBM MQ , które umożliwiają zarządzanie uruchamiają i zatrzymują procesy nastuchiwania z zakresu menedżera kolejek. Definiując atrybuty obiektu nastuchiwania, należy wykonać następujące czynności:

- Skonfiguruj proces nastuchiwania.
- Określ, czy proces nastuchiwania zostanie automatycznie uruchomiony, a następnie zatrzyma się po uruchomieniu i zatrzymaniu menedżera kolejek.

Ważne:  Obiekty nastuchiwania nie są obsługiwane w systemie IBM MQ for z/OS. Więcej informacji na temat implementowania nastuchiwania przez produkt IBM MQ for z/OS przy użyciu inicjatora kanału zawiera sekcja [“Inicjator kanału w systemie z/OS”](#) na stronie 167.

Odsyłacze pokrewne

[runmq1sr](#) (uruchamianie programu nastuchującego)

Usługi

Obiekty *Usługa* to sposób definiowania programów, które mają być uruchamiane podczas uruchamiania lub zatrzymywania menedżera kolejek.


Programy mogą być jednym z następujących typów:

Serwery

Serwer jest obiektem usługi, który ma parametr SERVTYPE określony jako SERVER. Obiekt usługi serwera to definicja programu, który zostanie wykonany po uruchomieniu określonego menedżera kolejek. Tylko jedna instancja procesu serwera może być wykonywana współbieżnie. W trakcie działania status procesu serwera można monitorować za pomocą komendy MQSC DISPLAY SVSTATUS. Zwykle obiekty usług serwera są definicjami programów, takich jak procedury obsługi niewysłanych wiadomości lub monitory wyzwalacza, jednak programy, które mogą być uruchamiane, nie są ograniczone do tych, które są dostarczane z produktem IBM MQ. Dodatkowo obiekt usługi serwera można zdefiniować w taki sposób, aby zawierał komendę, która zostanie uruchomiona, gdy określony menedżer kolejek zostanie zamknięty w celu zakończenia programu.

Komendy

Komenda jest obiektem usługi, który ma parametr SERVTYPE określony jako COMMAND. Obiekt usługi komendy jest definicją programu, który zostanie wykonany, gdy określony menedżer kolejek zostanie uruchomiony lub zatrzymany. Wiele instancji procesu komendy może być wykonywanych współbieżnie. Obiekty usług komend różnią się od obiektów usług serwera, gdy program jest wykonywany, a menedżer kolejek nie będzie monitorować tego programu. Zwykle obiekty usług komend są definicjami programów, które są krótsze, i wykonują określone zadania, takie jak uruchamianie jednego lub wielu innych zadań.

Ważne:  Obiekty usług nie są obsługiwane w systemie IBM MQ for z/OS.

Pojęcia pokrewne

[Praca z usługami](#)

Obiekty tematu

Obiekt tematu to obiekt IBM MQ, który umożliwia przypisywanie konkretnych atrybutów, które nie są domyślne, do tematów.

Temat jest definiowany przez publikowanie aplikacji lub subskrybujący określony *łańcuch tematu*. Łańcuch tematu może określać hierarchię tematów, oddzielając je znakiem ukośnika (/). Może to być wizualizowane przez *drzewo tematów*. Na przykład, jeśli aplikacja publikuje w łańcuchach tematów /Sport/American Football i /Sport/Soccer, zostanie utworzone drzewo tematów, które ma węzeł nadrzędny Sport z dwoma elementami podrzędnymi, American Football i Soccer.

Tematy dziedziczą ich atrybuty z pierwszego nadrzędnego węzła administracyjnego znalezione w ich drzewie tematów. Jeśli w danym drzewie tematów nie ma węzłów tematów administracyjnych, wszystkie tematy dziedziczą ich atrybuty z podstawowego obiektu tematu, SYSTEM.BASE.TOPIC.

Obiekt tematu można utworzyć w dowolnym węźle w drzewie tematów, określając łańcuch tematu tego węzła w atrybucie TOPICSTR obiektu tematu. Można również zdefiniować inne atrybuty dla węzła tematu administracyjnego. Więcej informacji na temat tych atrybutów można znaleźć w sekcji Komendy MQSC lub Automatyzacja administrowania za pomocą komend PCF. Każdy obiekt tematu domyślnie dziedziczy jego atrybuty z najbliższego nadrzędnego węzła tematu administracyjnego.

Obiekty tematów mogą być również używane w celu ukrycia pełnego drzewa tematów od twórców aplikacji. Jeśli obiekt tematu o nazwie FOOTBALL.US został utworzony dla tematu /Sport/American Football, aplikacja może publikować lub subskrybować obiekt o nazwie FOOTBALL.US, a nie łańcuch /Sport/American Football z tym samym wynikiem.

Jeśli w łańcuchu tematu w obiekcie tematu zostanie wprowadzony znak #, +, / lub *, to znak będzie traktowany jak normalny znak w łańcuchu i jest traktowany jako część łańcucha tematu powiązanego z obiektem tematu.

Więcej informacji na temat obiektów tematów zawiera sekcja [“Przesyłanie komunikatów publikowania/subskrypcji”](#) na stronie 63.

Nazywanie obiektów IBM MQ


Konwencja nazewnictwa przyjęta dla obiektów IBM MQ jest zależna od obiektu. Nazwy komputerów i identyfikatory użytkowników używane z produktem IBM MQ również podlegają pewnym ograniczeniom nazewnictwa.

Każda instancja menedżera kolejek jest znana pod nazwą. Ta nazwa musi być unikalna w ramach sieci połączonych ze sobą menedżerów kolejek, tak aby jeden menedżer kolejek mógł jednoznacznie zidentyfikować docelowy menedżer kolejek, do którego wysyłany jest dany komunikat.

W przypadku innych typów obiektów każdy obiekt ma powiązaną nazwę i może być przywołany przez tę nazwę. Te nazwy muszą być unikalne w obrębie jednego menedżera kolejek i typu obiektu. Na przykład można mieć kolejkę i proces o tej samej nazwie, ale nie można mieć dwóch kolejek o tej samej nazwie.

W programie IBM MQ nazwy mogą mieć maksymalnie 48 znaków, z wyjątkiem *kanatów*, których długość nie może być więcej niż 20 znaków. Więcej informacji na temat nazewnictwa obiektów IBM MQ zawiera sekcja [“Reguły nazewnictwa obiektów IBM MQ” na stronie 36](#).

Nazwy komputerów i identyfikatory użytkowników używane z produktem IBM MQ są również objęte pewnymi ograniczeniami nazewnictwa:

- Upewnij się, że nazwa komputera nie zawiera żadnych spacji. Produkt IBM MQ nie obsługuje nazw maszyn, które zawierają spacje. Jeśli produkt IBM MQ zostanie zainstalowany na takim komputerze, nie będzie można utworzyć żadnych menedżerów kolejek.
- W przypadku autoryzacji IBM MQ nazwy identyfikatorów użytkowników i grup nie mogą być dłuższe niż 20 znaków (spacje nie są dozwolone).
-  Serwer IBM MQ for Windows nie obsługuje połączenia IBM MQ MQI client, jeśli klient jest uruchomiony z identyfikatorem użytkownika, który zawiera znak @, na przykład abc@d.

Pojęcia pokrewne

“IBM MQ nazwy plików” na stronie 39

Każdy menedżer kolejek produktu IBM MQ, kolejka, definicja procesu, lista nazw, kanał, kanał połączenia klienta, obiekt nastuchiwania, usługa i obiekt informacji uwierzytelniającej są reprezentowane przez plik. Ponieważ nazwy obiektów nie zawsze są poprawnymi nazwami plików, menedżer kolejek przekształca nazwę obiektu w poprawną nazwę pliku, jeśli jest to konieczne.

Odsyłacze pokrewne

“Reguły nazewnictwa obiektów IBM MQ” na stronie 36

Nazwy obiektów IBM MQ mają maksymalną długość i są rozróżniane wielkości liter. Nie wszystkie znaki są obsługiwane dla każdego typu obiektu, a wiele obiektów ma reguły dotyczące unikalności nazw.

Reguły nazewnictwa obiektów IBM MQ

Nazwy obiektów IBM MQ mają maksymalną długość i są rozróżniane wielkości liter. Nie wszystkie znaki są obsługiwane dla każdego typu obiektu, a wiele obiektów ma reguły dotyczące unikalności nazw.

Istnieje wiele różnych typów obiektów IBM MQ, a obiekty z każdego typu mogą mieć taką samą nazwę, ponieważ istnieją w oddzielnych przestrzeniach nazw obiektów: na przykład kolejka lokalna i kanał nadawczy mogą mieć taką samą nazwę. Jednak obiekt nie może mieć tej samej nazwy co inny obiekt w tej samej przestrzeni nazw. Na przykład kolejka lokalna nie może mieć tej samej nazwy co kolejka modelowa, a kanał wysyłający nie może mieć tej samej nazwy co kanał odbierający.

W oddzielnych przestrzeniach nazw obiektów istnieją następujące obiekty IBM MQ :

- Informacje uwierzytelniające
- Kanał
- Kanał klienta
- Program nastuchujący
- Lista nazw


- Proces
- Kolejka
- Usługa
- Klasa pamięci masowej
- Subskrypcja
- Temat

Długość znaków w nazwach obiektów

Ogólnie nazwy obiektów IBM MQ mogą mieć długość do 48 znaków. Ta reguła ma zastosowanie do następujących obiektów:

- Informacje uwierzytelniające
- Klaster
- Program nasłuchujący
- Lista nazw
- Definicja procesu
- Kolejka
- Menedżer kolejek
- Usługa
- Subskrypcja
- Temat

Istnieją ograniczenia:

1.  W systemach z/OS menedżery kolejek muszą składać się maksymalnie z 4 znaków i muszą zawierać tylko wielkie litery i cyfry.
2. Maksymalna długość nazw obiektów kanału i nazw kanałów połączenia klienta wynosi 20 znaków. Więcej informacji na temat kanałów znajduje się w sekcji [Definiowanie kanałów](#).
3. Łańcuchy tematów mogą być maksymalnie 10240 bajtów. W nazwach wszystkich obiektów produktu IBM MQ rozróżniana jest wielkość liter.
4. Nazwy subskrypcji mogą być maksymalnie 10240 bajtów i mogą zawierać spacje.
5. Maksymalna długość nazw klas pamięci masowej wynosi 8 znaków.
6. Maksymalna długość nazw struktury CF wynosi 12 znaków.

Znaki w nazwach obiektów

Poprawne znaki dla nazw obiektów IBM MQ to:

| Znaków | Ograniczenia |
|--------------------|--------------|
| Wielkie litery A-Z | • Brak |

| Znaków | Ograniczenia |
|-------------------|---|
| Małe litery a-z | <ul style="list-style-type: none"> • W skryptach MQSC nazwy zawierające małe litery muszą być ujęte w pojedyncze znaki cudzysłowu. Zapobiega to składowaniu małych liter przy użyciu wielkich liter. • Systemy używające kodu EBCDIC Katakana nie mogą używać małych liter a-z w nazwach obiektów. • z/OS Podczas używania małych liter w systemach z/OS mogą występować ograniczenia, na przykład nazwy menedżerów kolejek nie mogą zawierać małych liter. • IBM i W systemach IBM i , jeśli używane są komendy CL, nazwy zawierające małe litery muszą być ujęte w znaki pojedynczego cudzysłowu. Zapobiega to składowaniu małych liter przy użyciu wielkich liter. |
| Cyfry 0-9 | <ul style="list-style-type: none"> • Brak |
| Kropka (.) | <ul style="list-style-type: none"> • Brak |
| Podkreślenie (_) | <ul style="list-style-type: none"> • Multi Brak • z/OS Unikaj używania nazw z czołowymi lub kończącymi znakami podkreślenia, ponieważ nie mogą one być obsługiwane przez panele kontrolne i panele kontrolne produktu IBM MQ for z/OS . |
| Prawy ukośnik (/) | <ul style="list-style-type: none"> • Windows W systemach Windows pierwszy znak nazwy menedżera kolejek nie może być ukośnikiem. • IBM i W systemach IBM i , jeśli używane są komendy CL, nazwy zawierające ukośnik muszą być ujęte w znaki pojedynczego cudzysłowu. • z/OS Brak |
| Znak procentu (%) | <ul style="list-style-type: none"> • ALW Brak • z/OS Jeśli produkt RACF jest używany jako zewnętrzny menedżer zabezpieczeń dla produktu IBM MQ for z/OS, nie należy używać wartości% w nazwach obiektów, ponieważ nazwy te nie są uwzględniane w kontrolach zabezpieczeń, gdy używane są profile ogólne produktu RACF . • IBM i W systemach IBM i , jeśli używane są komendy CL, nazwy zawierające znak procentu muszą być ujęte w znaki pojedynczego cudzysłowu. |

Istnieją również ogólne reguły dotyczące znaków w nazwach obiektów:

1. Czołowe lub wewnętrzne odstępy nie są dozwolone.
2. Znaki w języku narodowym nie są dozwolone.

3. Każda nazwa, która jest mniejsza od pełnej długości pola, może być dopełniona do prawej strony z odstępami. Wszystkie nazwy skrócone zwracane przez menedżera kolejek są zawsze dopełniane do prawej strony odstępami.


Nazwy kolejek

Nazwa kolejki składa się z dwóch części:

- Nazwa menedżera kolejek
- Lokalna nazwa kolejki, która jest znana temu menedżerowi kolejek

Każda część nazwy kolejki ma długość 48 znaków.

Aby odwołać się do kolejki lokalnej, można pominąć nazwę menedżera kolejek (zastępując ją odstępami lub używając wiodącego znaku o kodzie zero). Jednak wszystkie nazwy kolejek zwracane do programu przez program IBM MQ zawierają nazwę menedżera kolejek.


 Kolejka współużytkowana, dostępna dla dowolnego menedżera kolejek w grupie współużytkowania kolejek, nie może mieć takiej samej nazwy, jak żadna inna kolejka lokalna, która nie jest współużytkowana w tej samej grupie współużytkowania kolejki. To ograniczenie pozwala uniknąć sytuacji, w której aplikacja omyłkowo otwierała kolejkę współużytkowaną, gdy miała ona otwierać kolejkę lokalną, lub odwrotnie. Kolejki współużytkowane i grupy współużytkowania kolejek są dostępne tylko w systemie IBM MQ for z/OS.

Aby odwołać się do kolejki zdalnej, program musi zawierać nazwę menedżera kolejek w pełnej nazwie kolejki lub musi istnieć lokalna definicja kolejki zdalnej.

Jeśli aplikacja używa nazwy kolejki, może to być nazwa kolejki lokalnej (lub alias do jednej) lub nazwa lokalnej definicji kolejki zdalnej, ale aplikacja nie musi wiedzieć, która, chyba że musi pobrać komunikat z kolejki (gdy kolejka musi być lokalna). Gdy aplikacja otwiera obiekt kolejki, wywołanie MQOPEN wykonuje funkcję rozstrzygania nazw w celu określenia, w której kolejce mają być wykonywane kolejne operacje. Wynika to z tego, że aplikacja nie ma wbudowanej zależności od konkretnych kolejek, które są definiowane w określonych miejscach w sieci menedżerów kolejek. Oznacza to, że jeśli administrator systemu odlokuje kolejki w sieci i zmienia ich definicje, aplikacje, które korzystają z tych kolejek, nie muszą być zmieniane.

Zastrzeżone nazwy obiektów

Nazwy obiektów, które rozpoczynają się od SYSTEM . , są zarezerwowane dla obiektów zdefiniowanych przez menedżera kolejek. Za pomocą komend **Alter**, **Define** **Replace** można zmienić te definicje obiektów w taki sposób, aby odpowiadał on instalacji. Nazwy zdefiniowane dla produktu IBM MQ są wyświetlane w całości w polu [Nazwy kolejek](#).

 W systemie IBM MQ for z/OS nazwa struktury aplikacji CF (Coupling Facility) CSQSYSAPPL jest zastrzeżona.

Pojęcia pokrewne

[Nazwa instalacji w systemie AIX, Linux, and Windows](#)

IBM MQ nazwy plików

Każdy menedżer kolejek produktu IBM MQ , kolejka, definicja procesu, lista nazw, kanał, kanał połączenia klienta, obiekt nastuchiwania, usługa i obiekt informacji uwierzytelniającej są reprezentowane przez plik. Ponieważ nazwy obiektów nie zawsze są poprawnymi nazwami plików, menedżer kolejek przekształca nazwę obiektu w poprawną nazwę pliku, jeśli jest to konieczne.

Ścieżka domyślna do katalogu menedżera kolejek jest następująca:

- Przedrostek, który jest zdefiniowany w informacjach konfiguracyjnych produktu IBM MQ :

- **Linux** **AIX** W systemie AIX and Linux domyślnym przedrostkiem jest /var/mqm. Konfiguracja ta jest skonfigurowana w sekcji DefaultPrefix pliku konfiguracyjnego mqs.ini .
- **Windows** W 32-bitowych systemach Windows domyślnym przedrostkiem jest C:\Program Files (x86)\IBM\WebSphere MQ. W 64-bitowym systemie Windows domyślnym przedrostkiem jest C:\Program Files\IBM\MQ. W przypadku instalacji 32-bitowej i 64-bitowej katalogi danych są instalowane w produkcie C:\ProgramData\IBM\MQ. Konfiguracja ta jest skonfigurowana w sekcji DefaultPrefix pliku konfiguracyjnego mqs.ini .

Jeśli jest to możliwe, przedrostek można zmienić przy użyciu strony właściwości produktu IBM MQ w eksploratorze produktu IBM MQ , w przeciwnym razie należy ręcznie edytować plik konfiguracyjny produktu mqs.ini .

- Nazwa menedżera kolejek jest przekształcana w poprawną nazwę katalogu. Na przykład menedżer kolejek:

```
queue.manager
```

będą reprezentowane jako:

```
queue!manager
```

Ten proces jest określany jako *przekształcenie nazwy*.

W programie IBM MQ można nadać menedżerowi kolejek nazwę zawierającą maksymalnie 48 znaków.

Na przykład można nazwać menedżer kolejek:

```
QUEUE.MANAGER.ACCOUNTING.SERVICES
```

Jednak każdy menedżer kolejek jest reprezentowany przez plik i istnieją ograniczenia dotyczące maksymalnej długości nazwy pliku oraz znaków, które mogą być używane w nazwie. W wyniku tego nazwy plików reprezentujących obiekty są automatycznie przekształcane w celu spełnienia wymagań systemu plików.

Reguły dotyczące transformacji nazwy menedżera kolejek są następujące:

1. Transformuj pojedyncze znaki:
 - Od. do!
 - Od/do &
2. Jeśli nazwa jest nadal niepoprawna:
 - a. Obetnij go do ośmiu znaków
 - b. Dodaj trzyznakowy przyrostek liczbowy

Na przykład, przyjmując domyślny przedrostek i menedżer kolejek o nazwie queue.manager:

- **Windows** W systemie Windows z systemem NTFS lub FAT32 nazwa menedżera kolejek staje się następująca:

```
C:\Program Files\IBM\MQ\mqgrs\queue!manager
```

- **Windows** W systemie Windows z FAT nazwa menedżera kolejek staje się następująca:

```
C:\Program Files\IBM\MQ\mqgrs\queue!ma
```


- Linux
AIX
 W systemie AIX and Linux nazwa menedżera kolejek staje się następująca:

```
/var/mqm/qmgrs/queue!manager
```

Algorytm transformacji rozróżnia nazwy różniące się tylko wielkością liter w systemach plików, które nie rozróżniają wielkości liter.

Transformacja nazwy obiektu

Nazwy obiektów nie muszą być poprawnymi nazwami systemu plików. Może być konieczne przekształcenie nazw obiektów. Używana metoda różni się od używanej w nazwach menedżerów kolejek, ponieważ na każdym komputerze istnieje tylko kilka nazw menedżerów kolejek, ale dla każdego menedżera kolejek może istnieć duża liczba innych obiektów. Kolejki, definicje procesów, listy nazw, kanały, kanały połączeń klientów, obiekty nastuchiwania, usługi i obiekty informacji uwierzytelniających są reprezentowane w systemie plików.

Gdy nowa nazwa jest generowana przez proces transformacji, nie ma prostej relacji z oryginalną nazwą obiektu. Komendy **dspmqls** można użyć do przekształcenia rzeczywistych i przekształconych nazw obiektów.

Odsyłacze pokrewne

dspmqls ([wyświetlanie nazw plików](#))

IBM i Nazwy obiektów w systemie IBM i

Menedżer kolejek ma powiązaną bibliotekę menedżera kolejek, która ma unikalną nazwę. Nazwy i nazwy obiektów menedżera kolejek mogą wymagać transformacji w celu spełnienia wymagań produktu IBM i Integrated File System (IFS).

Po utworzeniu menedżera kolejek produkt IBM MQ przypisuje z nim bibliotekę menedżera kolejek. Ta biblioteka menedżera kolejek ma unikalną nazwę, nie więcej niż 10 znaków, w dużej mierze opartą na nazwie menedżera kolejek zdefiniowanej przez użytkownika. Zarówno menedżer kolejek, jak i biblioteka menedżera kolejek są umieszczane w katalogu, który jest również oparty na nazwie menedżera kolejek z przedrostkiem `/QIBM/UserData/mqm`. Poniżej przedstawiono przykład menedżera kolejek, biblioteki menedżera kolejek i katalogu:

| | |
|------------------------------------|--|
| Nazwa menedżera kolejek | Pomarańczowy |
| Nazwa biblioteki menedżera kolejek | QMORANGE |
| Katalog | <code>/QIBM/UserData/mqm/ORANGE</code> |

Wszystkie nazwy menedżerów kolejek oraz nazwy bibliotek menedżerów kolejek są zapisywane w sekcjach w pliku `/QIBM/UserData/mqm/mqs.ini`.

Katalogi i pliki produktu IBM MQ IFS

IBM i Integrated File System (IFS) jest intensywnie używany przez produkt IBM MQ do przechowywania danych. Więcej informacji na temat systemu plików IFS zawiera publikacja *Integrated File System Wprowadzenie*.

Każdy obiekt IBM MQ, na przykład kanał lub menedżer kolejek, jest reprezentowany przez plik. Ponieważ nazwy obiektów nie zawsze są poprawnymi nazwami plików, menedżer kolejek przekształca nazwę obiektu w poprawną nazwę pliku, jeśli jest to konieczne.

Ścieżka do katalogu menedżera kolejek jest tworzona z następujących elementów:

- Przedrostek, który jest zdefiniowany w pliku konfiguracyjnym menedżera kolejek `qm.ini`. Domyślnym przedrostkiem jest `/QIBM/UserData/mqm`.

- Literał qmgrs.
- Zakodowana nazwa menedżera kolejek, która jest nazwą menedżera kolejek przekształconą w poprawną nazwę katalogu. Na przykład menedżer kolejek `queue/manager` jest reprezentowany przez produkt `queue&manager`.

Ten proces jest określany jako przekształcenie nazwy.

Transformacja nazwy menedżera kolejek IFS

W programie IBM MQ można nadać menedżerowi kolejek nazwę zawierającą maksymalnie 48 znaków.

Na przykład można nazwać menedżer kolejek `QUEUE/MANAGER/ACCOUNTING/SERVICES`. W ten sam sposób, w jaki tworzona jest biblioteka dla każdego menedżera kolejek, każdy menedżer kolejek jest również reprezentowany przez plik. Ze względu na wariantowe punkty kodowe w kodzie EBCDIC istnieją ograniczenia dotyczące znaków, które mogą być używane w nazwie. Oznacza to, że nazwy plików IFS reprezentujące obiekty są automatycznie przekształcane w celu spełnienia wymagań systemu plików.

Korzystając z przykładu menedżera kolejek o nazwie `queue/manager`, transformując znak `/` na `&i` zakładając domyślny przedrostek, nazwa menedżera kolejek w programie IBM MQ for IBM i staje się `/QIBM/UserData/mqm/qmgrs/queue&manager`.

Transformacja nazwy obiektu

Nazwy obiektów nie muszą być poprawnymi nazwami systemów plików, więc nazwy obiektów mogą wymagać transformacji. Używana metoda jest inna niż w przypadku nazw menedżerów kolejek, ponieważ chociaż dla każdego komputera istnieje tylko kilka nazw menedżerów kolejek, dla każdego menedżera kolejek może istnieć duża liczba innych obiektów. W systemie plików reprezentowane są tylko definicje procesów, kolejki i listy nazw. Te uwagi nie mają wpływu na kanały.

Gdy nowa nazwa jest generowana przez proces transformacji, nie ma prostej relacji z oryginalną nazwą obiektu. W celu wyświetlenia przekształconych nazw obiektów IBM MQ można użyć komendy `DSPMQOBJN`.

Atrybuty obiektu

Właściwości obiektu są definiowane przez jego atrybuty. Niektóre z nich można określić, inne mogą być tylko wyświetlane.

Na przykład maksymalna długość komunikatu, jaką może pomieścić kolejka, jest definiowana przez jej atrybut **MaxMsgLength**. Atrybut ten można określić podczas tworzenia kolejki. Atrybut **DefinitionType** określa sposób, w jaki kolejka została utworzona. Ten atrybut może być wyświetlany tylko.

W programie IBM MQ istnieją dwa sposoby odwołania się do atrybutu:

- Można użyć nazwy PCF, na przykład **MaxMsgLength**.
- Używana jest nazwa komendy MQSC, na przykład `MAXMSGL`.

Grupy współużytkowania kolejek

Menedżery kolejek, które mogą uzyskiwać dostęp do tego samego zestawu kolejek współużytkowanych, tworzą grupę o nazwie *grupa współużytkowania kolejek* (QSG) i komunikują się ze sobą przy użyciu narzędzia CF, w którym przechowywane są współużytkowane kolejki.

Kolejka współużytkowana jest typem kolejki lokalnej z komunikatami, do których może uzyskać dostęp co najmniej jeden menedżer kolejek, który znajduje się w grupie współużytkowania kolejek. Wartość ta nie jest taka sama, jak kolejka współużytkowana przez więcej niż jedną aplikację, przy użyciu tego samego menedżera kolejek.

Nazwy grup współużytkujących kolejki składają się z maksymalnie czterech znaków. Nazwa taka musi być unikalna w danej sieci i nie może być identyczna z nazwą menedżera kolejek.

Grupy współużytkowania kolejek nie są ściśle obiektami, ale są tu wymienione dla wygody.

Ważne: Kolejki współużytkowane i grupy współużytkowania kolejek są obsługiwane tylko w systemie IBM MQ for z/OS.

Pojęcia pokrewne

“Współużytkowane kolejki i grupy współużytkowania kolejek” na stronie 172

W celu zaimplementowania wysokiej dostępności zasobów produktu IBM MQ można używać współużytkowanych kolejek i grup współużytkowania kolejek. Kolejki współużytkowane i grupy współużytkowania kolejek są unikalne dla programu IBM MQ for z/OS na platformie z/OS .

Domyślne obiekty systemowe

Domyślne obiekty systemowe to zestaw definicji obiektów, które są tworzone automatycznie za każdym razem, gdy tworzony jest menedżer kolejek.

Istnieje możliwość skopiowania i zmodyfikowania dowolnej z tych definicji obiektów do użycia w aplikacjach podczas instalacji.

Domyślne nazwy obiektów mają system macierzysty, na przykład domyślną kolejką lokalną jest SYSTEM.DEFAULT.LOCAL.QUEUE, a domyślnym kanałem odbiorczym jest SYSTEM.DEF.RECEIVER. Nie można zmieniać nazw tych obiektów; wymagane są obiekty domyślne tych nazw.

Po zdefiniowaniu obiektu wszystkie atrybuty, które nie zostały jawnie określone, są kopiowane z odpowiedniego obiektu domyślnego. Na przykład, jeśli definiowana jest kolejka lokalna, te atrybuty, które nie zostaną określone, są pobierane z domyślnej kolejki SYSTEM.DEFAULT.LOCAL.QUEUE.

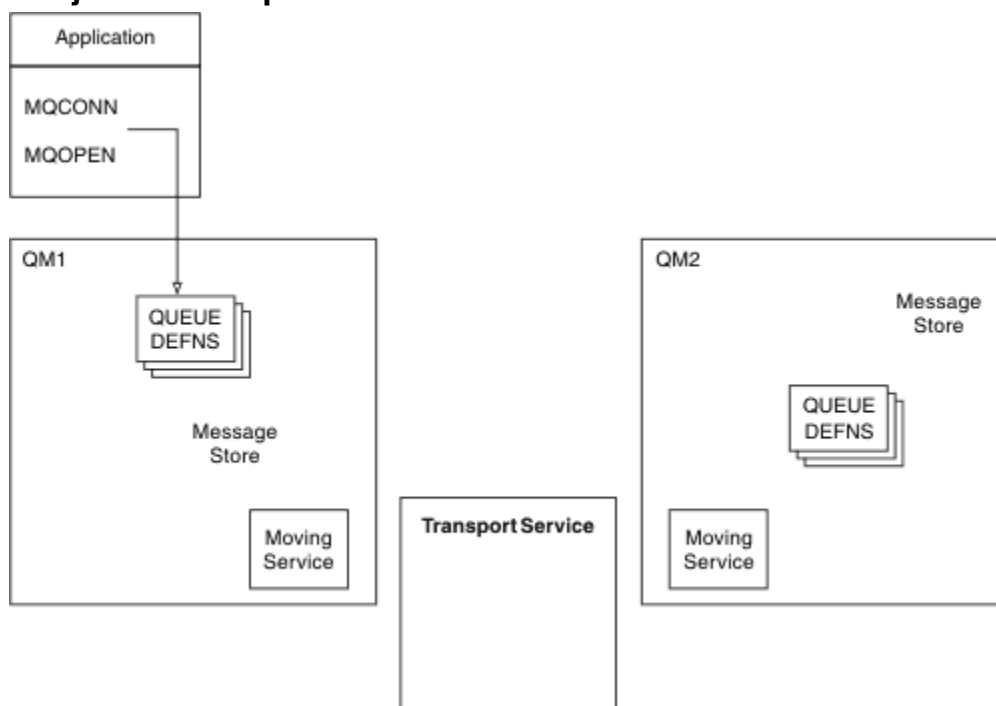
Odsyłacze pokrewne

[Obiekty systemowe i domyślne](#)

Rozproszone kolejkowanie i klastry

Kolejkowanie rozproszone oznacza wysyłanie komunikatów z jednego menedżera kolejek do innego. Odbierający menedżer kolejek może znajdować się na tej samej lub innej maszynie; w pobliżu lub po drugiej stronie świata. Może być uruchomiony na tej samej platformie co lokalny menedżer kolejek lub może znajdować się na dowolnej z platform obsługiwanych przez produkt IBM MQ. Można ręcznie zdefiniować wszystkie połączenia w rozproszonym środowisku kolejkowania lub utworzyć klastr, a następnie IBM MQ zdefiniować wiele szczegółów połączenia dla użytkownika.

kolejkowanie rozproszone



Rysunek 4. Przegląd komponentów kolejkowania rozproszonego

Na poprzednim rysunku:

- Aplikacja używa wywołania MQCONN do łączenia się z menedżerem kolejek. Aplikacja używa następnie wywołania MQOPEN w celu otwarcia kolejki, tak aby mogła ona umieszczać komunikaty w kolejce.
- Każdy menedżer kolejek ma definicję dla każdej z jej kolejek. Może zawierać definicje *kolejek lokalnych* (które są udostępniane przez tego menedżera kolejek) oraz definicje *kolejek zdalnych* (obsługiwane przez inne menedżery kolejek).
- Jeśli komunikaty są przeznaczone dla kolejki zdalnej, lokalny menedżer kolejek przechowuje je w *kolejce transmisji*, która utrzymuje je w składnicy komunikatów, aż do momentu przekazania ich do zdalnego menedżera kolejek.
- Każdy menedżer kolejek zawiera oprogramowanie komunikacyjne zwane *usługą przenoszenia*, które jest używane przez menedżer kolejek do komunikowania się z innymi menedżerami kolejek.
- *Usługa transportowa* jest niezależna od menedżera kolejek i może być dowolną z następujących opcji (w zależności od platformy):
 - System Network Architecture Advanced Program-to Program Communication (SNA APPC)
 - protokół TCP/IP (Transmission Control Protocol/Internet Protocol)
 - Podstawowy system wejścia/wyjścia sieci (NetBIOS)
 - Sekwencyjna wymiana pakietów (SPX)

Komponenty wymagane do wysłania komunikatu

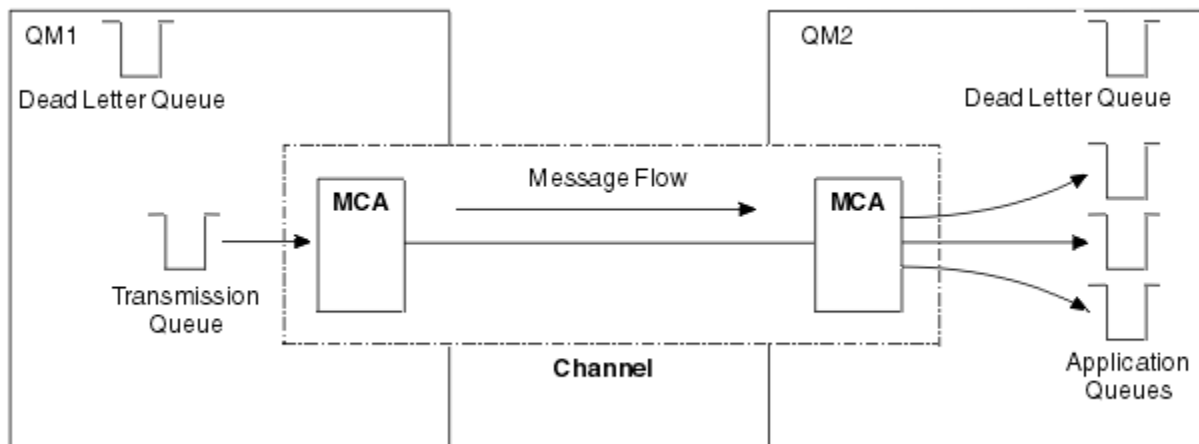
Jeśli komunikat ma zostać wysłany do zdalnego menedżera kolejek, menedżer kolejek lokalnych wymaga definicji dla *kolejki transmisji* i *kanatu*. Kanał jest jednokierunkową łączem komunikacyjnym między dwoma menedżerami kolejek. Może on przenosić komunikaty przeznaczone dla dowolnej liczby kolejek w zdalnym menedżerze kolejek.

Każdy koniec kanału ma osobną definicję, definiując ją, na przykład, jako koniec wysyłający lub odbierający. Prosty kanał składa się z definicji kanału *nadawcy* w lokalnym menedżerze kolejek oraz definicji kanału *odbiornika* w zdalnym menedżerze kolejek. Te dwie definicje muszą mieć taką samą nazwę, a razem tworzą jeden kanał.

Oprogramowanie, które obsługuje wysyłanie i odbieranie komunikatów, jest nazywane *agentem kanału komunikatów* (Message Channel Agent-MCA). Na każdym końcu kanału znajduje się *agent kanału komunikatów* (MCA).

Każdy menedżer kolejek powinien mieć *kolejkę niedostarczonych komunikatów* (znaną również jako *niedostarczona kolejka komunikatów*). Komunikaty są umieszczane w tej kolejce, jeśli nie mogą zostać dostarczone do miejsca docelowego.

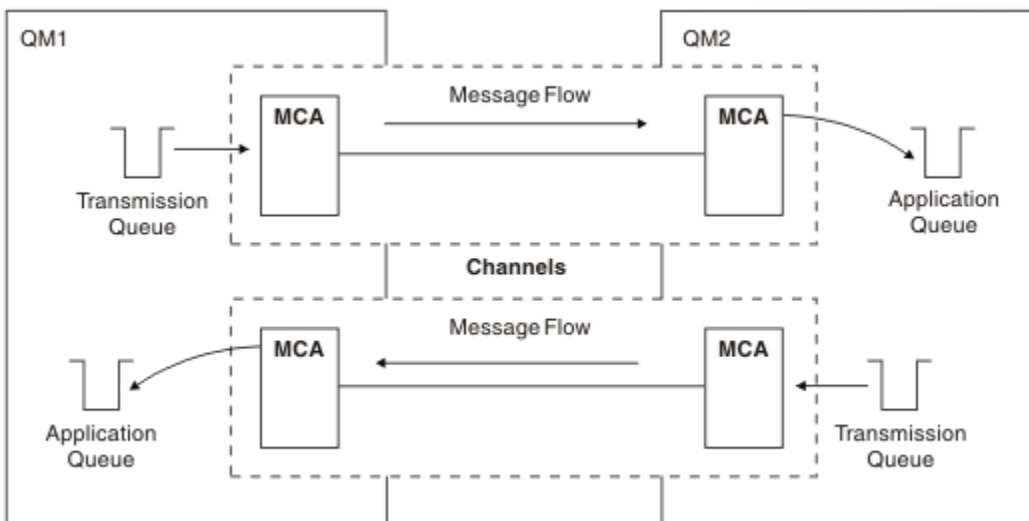
Na poniższym rysunku przedstawiono relacje między menedżerami kolejek, kolejkami transmisji, kanałami i kontekstami MCA:



Rysunek 5. Wysyłanie komunikatów

Komponenty wymagane do zwrócenia komunikatu

Jeśli aplikacja wymaga, aby komunikaty były zwracane ze zdalnego menedżera kolejek, należy zdefiniować inny kanał w celu uruchomienia w odwrotnym kierunku między menedżerami kolejek, jak pokazano na poniższym rysunku:



Rysunek 6. Wysyłanie wiadomości w obu kierunkach

Klastry

Zamiast ręcznego definiowania wszystkich połączeń w rozproszonym środowisku kolejkowania można grupować zestaw menedżerów kolejek w klastrze. W takim przypadku menedżery kolejek mogą udostępniać kolejki, które udostępniają innym menedżerom kolejek w klastrze, bez konieczności tworzenia jawnych definicji kanałów, definicji kolejek zdalnych lub kolejek transmisji dla każdego miejsca

docelowego. Każdy menedżer kolejek w klastrze ma pojedynczą kolejkę transmisji, która przesyła komunikaty do dowolnego innego menedżera kolejek w klastrze. Dla każdego menedżera kolejek należy zdefiniować tylko jeden kanał odbierający klastry i jeden kanał nadawczy klastra. W tym celu wszystkie dodatkowe kanały są automatycznie zarządzane przez klastr.

Klient IBM MQ może łączyć się z menedżerem kolejek, który jest częścią klastra, tak jak może on łączyć się z dowolnym innym menedżerem kolejek. Podobnie jak w przypadku ręcznego konfigurowalnego kolejkowania rozproszonego, można użyć wywołania MQPUT w celu umieszczenia komunikatu w kolejce w dowolnym menedżerze kolejek. Wywołanie MQGET służy do pobierania komunikatów z kolejki lokalnej.

Menedżery kolejek na platformach, które obsługują klastry, nie muszą być częścią klastra. Można kontynuować ręczną konfigurację rozproszonego kolejkowania, a także, lub zamiast, za pomocą klastrów.

Korzyści z używania klastrów

Technologia klastrowa zapewnia dwa kluczowe korzyści:

- Klastry upraszczają administrowanie sieciami IBM MQ, które zwykle wymagają skonfigurowania wielu definicji obiektów dla kanałów, kolejek nadawanych i kolejek zdalnych. Ta sytuacja jest szczególnie prawdziwa w dużych, potencjalnie zmieniających się sieciach, w których wiele menedżerów kolejek musi być połączonych ze sobą wzajemnie. Architektura ta jest szczególnie trudna do skonfigurowania i aktywnego zachowania.
- Klastry mogą być używane do dystrybuowania obciążenia ruchu komunikatów między kolejkami i menedżerami kolejek w klastrze. Taka dystrybucja umożliwi rozproszenie obciążenia komunikatów pojedynczej kolejki w równorzędnych instancjach tej kolejki znajdujących się w wielu menedżerach kolejek. Ta dystrybucja obciążenia może być używana do uzyskania większej odporności na awarie systemu oraz do zwiększenia wydajności skalowania szczególnie aktywnych przepływów komunikatów w systemie. W takim środowisku każda z instancji kolejek rozproszonych korzystała z aplikacji przetwarzając komunikaty. Więcej informacji na ten temat zawiera sekcja [Używanie klastrów do zarządzania obciążeniem](#).

W jaki sposób komunikaty są kierowane w klastrze

Klastr może być używany jako sieć menedżerów kolejek utrzymywanych przez administratora systemów sumiennych. Za każdym razem, gdy definiowana jest kolejka klastra, administrator systemu automatycznie tworzy odpowiednie definicje kolejek zdalnych zgodnie z potrzebami w innych menedżerach kolejek.

Nie ma potrzeby tworzenia definicji kolejek transmisji, ponieważ program IBM MQ udostępnia kolejkę transmisji dla każdego menedżera kolejek w klastrze. Ta pojedyncza kolejka transmisji może być używana do przenoszenia komunikatów do dowolnego innego menedżera kolejek w klastrze. Korzystanie z pojedynczej kolejki transmisji nie jest ograniczone. Menedżer kolejek może używać wielu kolejek transmisji do rozdzielania komunikatów wysyłanych do każdego menedżera kolejek w klastrze. Zwykle menedżer kolejek korzysta z jednej kolejki transmisji klastra. Atrybut menedżera kolejek DEFCLXQ można zmienić, tak aby menedżer kolejek używał innej kolejki transmisji klastra dla każdego menedżera kolejek w klastrze. Kolejki transmisji klastra można również zdefiniować ręcznie.

Wszystkie menedżery kolejek, które łączą się z klastrem, wyrażają zgodę na pracę w ten sposób. Wysyłają informacje o sobie i o kolejkach, które są gospodarzem, a także otrzymują informacje o innych elementach klastra.

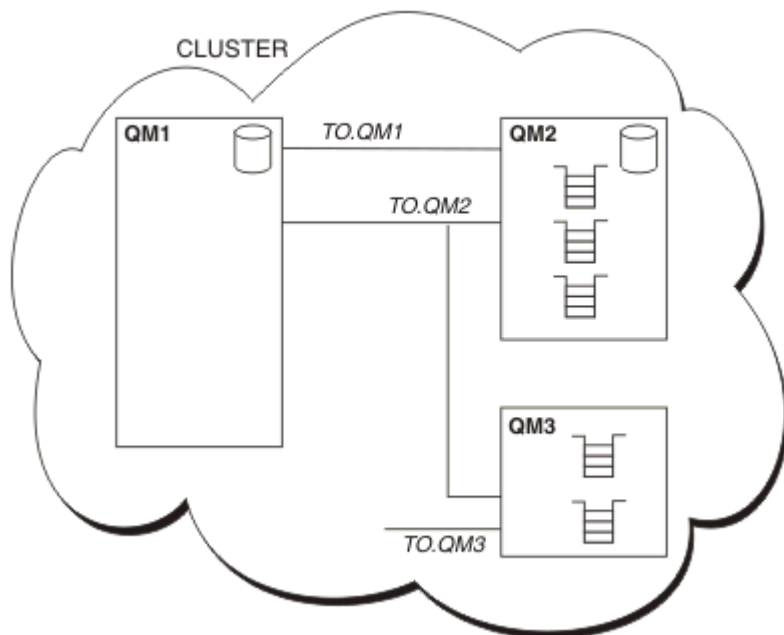
Aby upewnić się, że żadne informacje nie zostaną utracone, gdy menedżer kolejek stanie się niedostępny, należy określić dwa menedżery kolejek w klastrze, aby działały jako *pełne repozytoria*. Te menedżery kolejek przechowują pełny zestaw informacji o wszystkich menedżerach kolejek i kolejkach w klastrze. Wszystkie pozostałe menedżery kolejek w klastrze przechowują tylko informacje o tych menedżerach kolejek i kolejkach, z którymi wymieniają komunikaty. Te menedżery kolejek są nazywane *częściowymi repozytoriami*. Więcej informacji na ten temat zawiera sekcja [“Repozytorium klastra”](#) na stronie 57.

Aby stać się częścią klastra, menedżer kolejek musi mieć dwa kanały; kanał wysyłający klastry i kanał odbierający klastry:

- Kanał nadawczy klastra jest kanałem komunikacyjnym, takim jak kanał nadawczy. Konieczne jest ręczne utworzenie jednego kanału nadawczego klastra w menedżerze kolejek w celu połączenia go z pełnym repozytorium, które jest już elementem klastra.
- Kanał odbiorczy klastra jest kanałem komunikacyjnym, takim jak kanał odbiorczy. Należy ręcznie utworzyć jeden kanał odbiorczy klastra. Kanał działa jako mechanizm dla menedżera kolejek w celu odbierania komunikacji klastra.

Wszystkie inne kanały wymagane do komunikacji między tym menedżerem kolejek i innymi elementami klastra są tworzone automatycznie.

Na poniższym rysunku przedstawiono komponenty klastra o nazwie CLUSTER:



Rysunek 7. Klaster menedżerów kolejek

- Parametr CLUSTER zawiera trzy menedżery kolejek: QM1, QM2 i QM3.
- QM1 i QM2 - pełne repozytoria informacji o menedżerach kolejek i kolejkach w klastrze.
- QM2 i QM3 są hostem niektórych kolejek klastra, czyli kolejek, które są dostępne dla dowolnego innego menedżera kolejek w klastrze.
- Każdy menedżer kolejek ma kanał odbierający klastry o nazwie TO.qmgr, na którym może odbierać komunikaty.
- Każdy menedżer kolejek ma również kanał wysyłający klastry, na którym może wysyłać informacje do jednego z menedżerów kolejek repozytorium.
- QM1 i QM3 wysyłają do repozytorium o wartości QM2, a QM2 wysyła do repozytorium w katalogu QM1.

Rozproszone komponenty kolejkowania

Komponenty kolejkowania rozproszonego to kanały komunikatów, agenty kanałów komunikatów, kolejki transmisji, inicjatory kanału i programy nastuchujące oraz programy obsługi wyjścia kanału. Definicja każdego końca kanału komunikatów może być jednym z kilku typów.

Kanały komunikatów to kanały, które przenoszą komunikaty z jednego menedżera kolejek do innego. Nie należy mylić kanałów komunikatów z kanałami MQI. Istnieją dwa typy kanału MQI, połączenia z serwerem (SVRCONN) i połączenia klienckiego (CLNTCONN). Więcej informacji na ten temat zawiera sekcja [Kanały](#).

Definicja każdego końca kanału komunikatów może być jednym z następujących typów:

- Nadawca (SDR)
- Odbiornik (RCVR)
- Serwer (SVR)
- Żądający (RQSTR)
- Nadawca klastra (CLUSDR)
- Odbiornik klastra (CLUSRCVR)

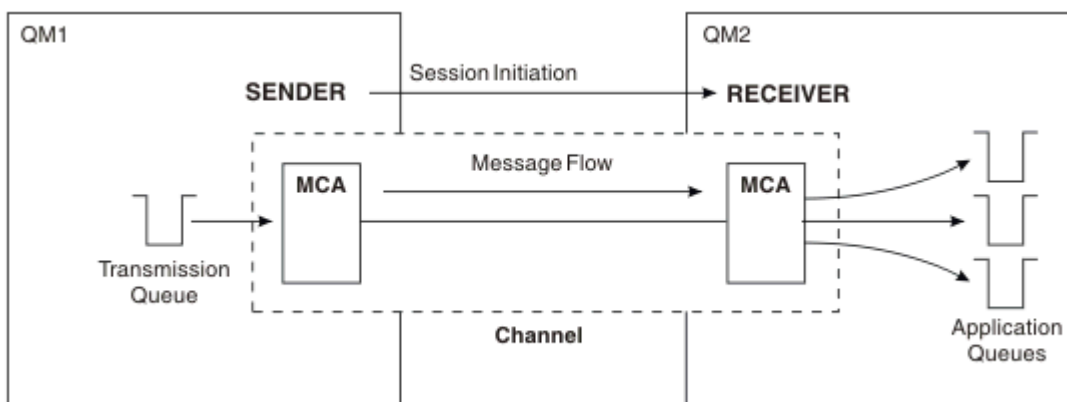
Kanał komunikatów jest definiowany przy użyciu jednego z tych typów zdefiniowanych na jednym końcu, a typ zgodny na drugim końcu. Możliwe kombinacje to:

- Kanał nadawcy-Kanał odbiorcy
- Kanał requestera-Kanał serwera
- Kanał requestera-Kanał nadawcy (wywołanie zwrotne)
- Kanał serwera-Kanał odbiorcy
- Kanał wysyłający klastry-odbiorca klastra

Szczegółowe instrukcje dotyczące tworzenia kanału odbiorczego nadawcy są zawarte w sekcji [Definiowanie kanałów](#). Przykłady parametrów wymaganych do skonfigurowania kanałów nadawczych można znaleźć w sekcji [Przykładowe informacje o konfiguracji](#), które mają zastosowanie do używanej platformy. Parametry wymagane do zdefiniowania kanału dowolnego typu znajdują się w temacie [DEFINE CHANNEL\(DEFINIOWANIE KANAŁU\)](#).

Kanał nadawcy - kanał odbiorcy

Nadawca w jednym systemie uruchamia kanał w taki sposób, aby mógł wysyłać komunikaty do innego systemu. Nadawca żąda, aby odbiorca na drugim końcu kanału został uruchomiony. Nadawca wysyła komunikaty z kolejki transmisji do odbiorcy. Odbiornik umieszcza komunikaty w kolejce docelowej. Rysunek 8 na stronie 48 ilustruje to.

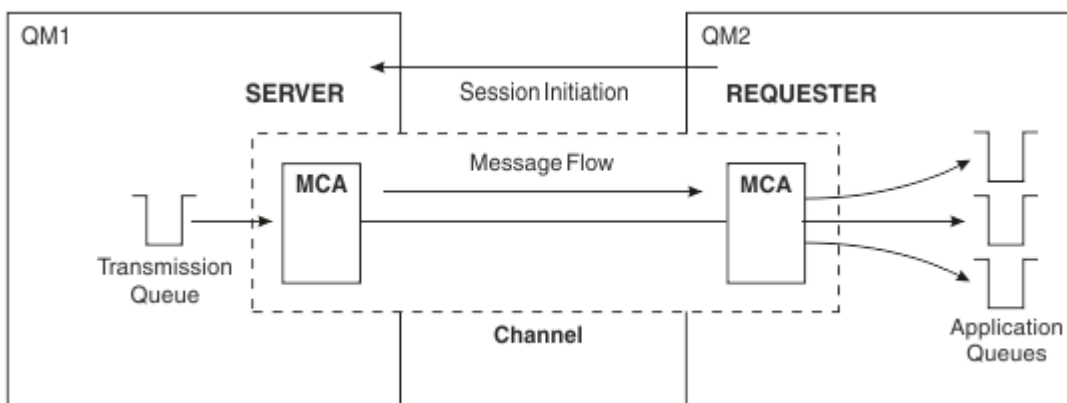


Rysunek 8. Kanał odbiorczy nadawcy

Kanał requestera - kanał serwera

Requester w jednym systemie uruchamia kanał w taki sposób, aby mógł odbierać komunikaty z innego systemu. Requester żąda od serwera na drugim końcu kanału, który ma zostać uruchomiony. Serwer wysyła komunikaty do requestera z kolejki transmisji zdefiniowanej w definicji kanału.

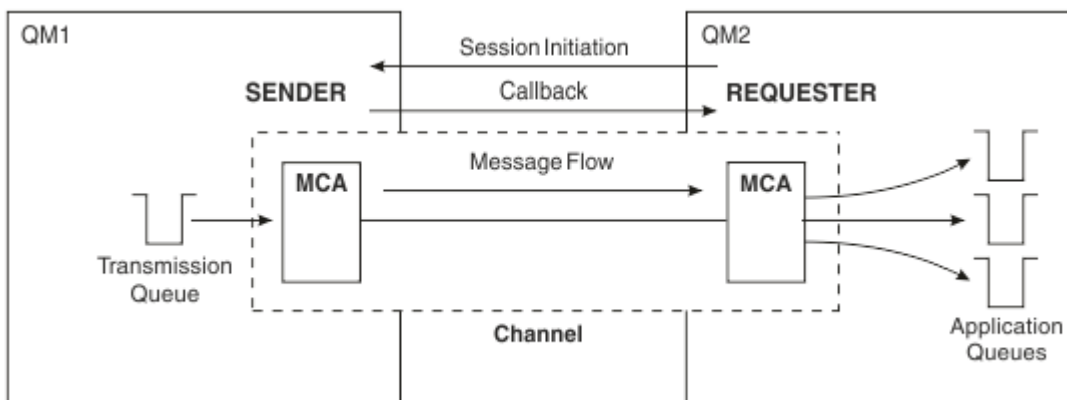
Kanał serwera może również inicjować komunikację i wysyłać komunikaty do requestera. Dotyczy to tylko serwerów *w pełni kwalifikowanych*, czyli kanałów serwera, których nazwa połączenia jest określona przez partnera określonego w definicji kanału. W pełni wykwalifikowany serwer może być uruchomiony przez requester lub może zainicjować komunikację z requestersiem.



Rysunek 9. Kanał requestera-serwer

Kanał requestera - kanał nadawcy

Requester uruchamia kanał, a nadawca kończy połączenie. Następnie nadawca restartuje komunikację zgodnie z informacjami w jego definicji kanału (nazywanym się *wywołaniem zwrotnym*). Wysyła on komunikaty z kolejki transmisji do requestera.



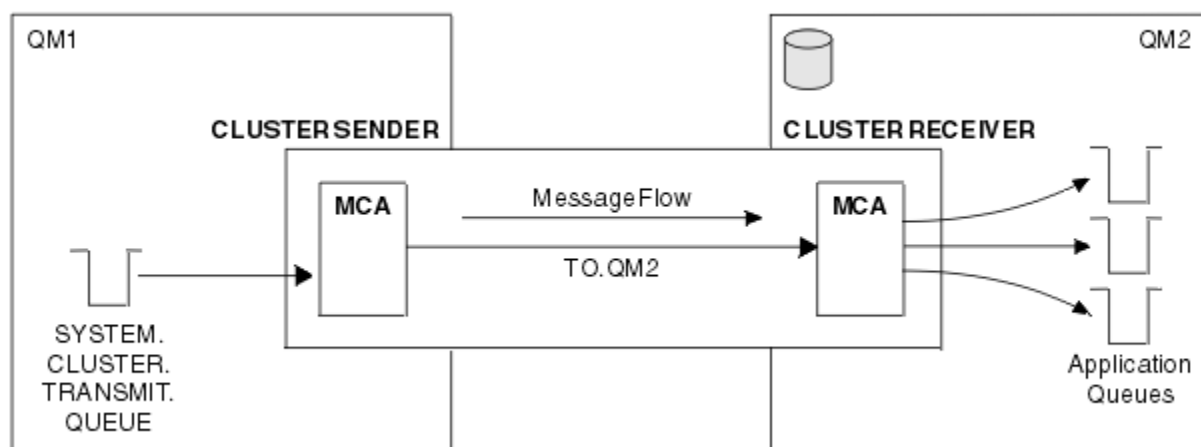
Rysunek 10. Kanał wysyłający requestera

Kanały odbiorcze serwera

Jest to typ nadawca-nadawca, ale dotyczy tylko serwerów *w pełni kwalifikowanych*, czyli kanałów serwera, które mają nazwę połączenia partnera określonego w definicji kanału. Uruchamianie kanału musi być zainicjowane na końcu łącza serwera. Ilustracja przedstawiana jest na ilustracji w podręczniku Rysunek 8 na stronie 48.

Kanały wysyłające klastry

W klastrze każdy menedżer kolejek ma kanał nadawczy klastra, na którym może wysyłać informacje o klastrze do jednego z menedżerów kolejek pełnego repozytorium. Menedżery kolejek mogą również wysyłać komunikaty do innych menedżerów kolejek w kanałach nadawczych klastra.



Rysunek 11. Kanał wysyłający klastry

Kanały odbierające klastry

W klastrze każdy menedżer kolejek ma kanał odbiorczy klastra, na którym może odbierać komunikaty i informacje o klastrze. Ilustracja przedstawiana jest na ilustracji w podręczniku [Rysunek 11 na stronie 50](#).

Kolejki niedostarczane

Kolejka niedostarczonych komunikatów (lub kolejka niedostarczonych komunikatów) jest kolejką, do której wysyłane są komunikaty, jeśli nie mogą być kierowane do ich poprawnego miejsca docelowego. Każdy menedżer kolejek zwykle ma kolejkę niedostarczonych komunikatów.

Kolejka niedostarczonych komunikatów (DLQ), nazywana czasem *kolejką niedostarczonych komunikatów*, jest kolejką wstrzymana dla komunikatów, których nie można dostarczyć do kolejek docelowych, na przykład dlatego, że kolejka nie istnieje, lub ponieważ jest ona pełna. Kolejki niedostarczane są również używane w wysyłającym końcu kanału, w przypadku błędów konwersji danych. Każdy menedżer kolejek w sieci 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.

Komunikaty mogą być umieszczane w kolejce DLQ przez menedżery kolejek, agenty kanałów komunikatów (MCAs) i aplikacje. Wszystkie komunikaty w DLQ muszą być poprzedzane strukturą *dead-letter header*, MQDLH. Pole *Przyczyna* struktury MQDLH zawiera kod przyczyny, który określa, dlaczego komunikat znajduje się w kolejce DLQ.

Zwykle należy zdefiniować kolejkę niedostarczonych komunikatów dla każdego menedżera kolejek. Jeśli nie, a agent MCA nie może umieścić komunikatu, zostanie on umieszczony w kolejce transmisji, a kanał zostanie zatrzymany. Ponadto, jeśli szybkie, nietrwałe komunikaty (patrz *Szybkie, nietrwałe komunikaty*) nie można dostarczyć, a w systemie docelowym nie istnieje żadna kolejka niedostarczonych komunikatów, komunikaty te są odrzucane.

Jednak użycie kolejek niedostarczonych komunikatów może mieć wpływ na kolejność, w jakiej komunikaty są dostarczane i dlatego użytkownik może nie korzystać z nich.

Definicje kolejek zdalnych

Definicje kolejek zdalnych są definicjami dla kolejek, których właścicielem jest inny menedżer kolejek.

Podczas gdy aplikacje mogą pobierać komunikaty tylko z kolejek lokalnych, mogą umieszczać komunikaty w kolejkach lokalnych lub kolejkach zdalnych. Dlatego też, jak również definicja dla każdej z kolejek lokalnych, menedżer kolejek może mieć *definicje kolejek zdalnych*. Zaletą definicji kolejek zdalnych jest umożliwienie aplikacji umieszczenia komunikatu w kolejce zdalnej bez konieczności określania nazwy kolejki zdalnej lub zdalnego menedżera kolejek lub nazwy kolejki transmisji. Definicje kolejek zdalnych dają niezależność lokalizacji.

Istnieją inne zastosowania dla definicji kolejek zdalnych, które zostały opisane w dalszej części.

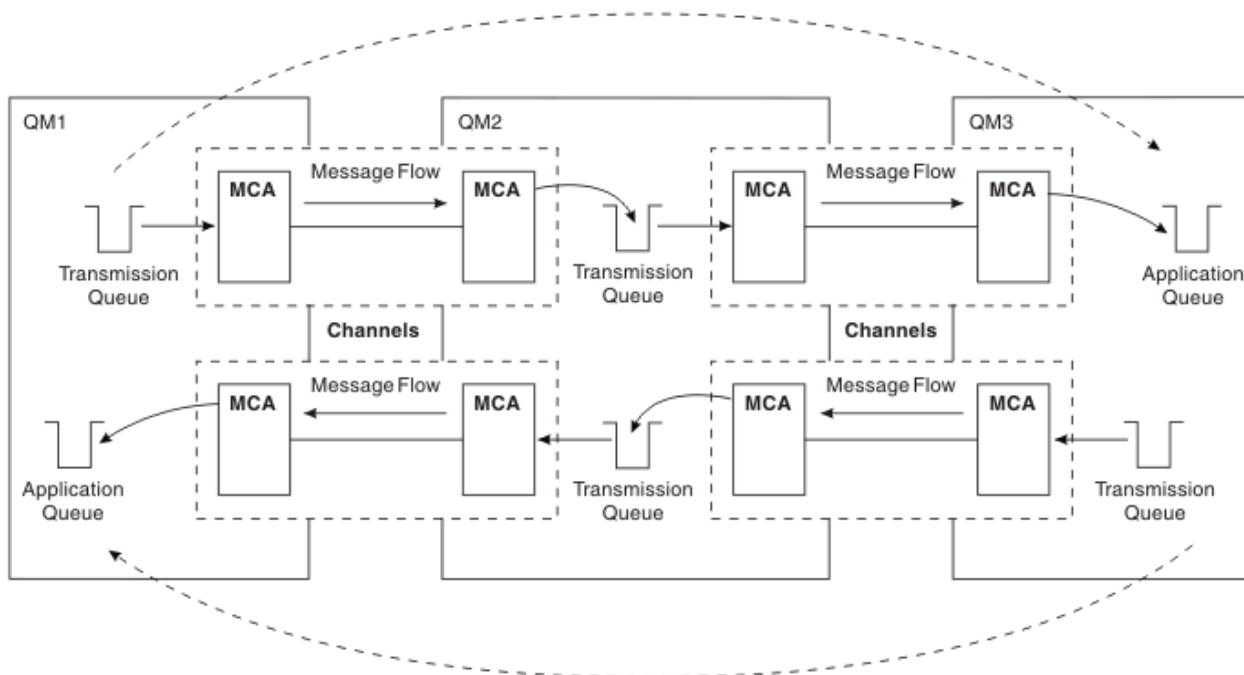
Jak dostać się do zdalnego menedżera kolejek

Między każdym źródłowym a docelowym menedżerem kolejek może nie być zawsze jeden kanał. Istnieje wiele innych sposobów łączenia między tymi dwoma kanałami, w tym między kanałami, które współużytkuje kanały, przy użyciu różnych kanałów i technologii klastrowej.

Wielo-hopping

Jeśli między źródłowym menedżerem kolejek a docelowym menedżerem kolejek nie ma bezpośredniego połączenia komunikacyjnego, możliwe jest przejście przez jeden lub większą liczbę *menedżerów kolejek pośrednich* w drodze do docelowego menedżera kolejek. Jest on znany jako *wieloprzeskokowy*.

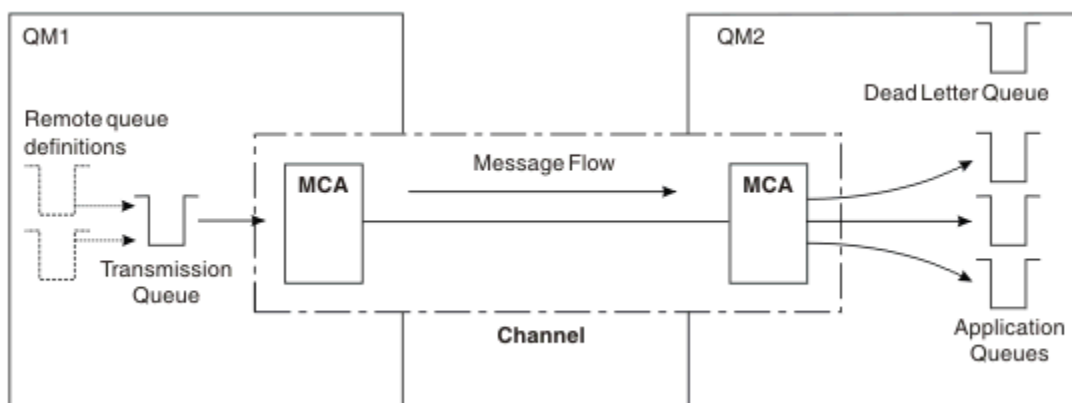
Należy zdefiniować kanały między wszystkimi menedżerami kolejek i kolejkami transmisji w pośrednim menedżerze kolejek. Pokazano to na ilustracji [Rysunek 12 na stronie 51](#).



Rysunek 12. Przekazywanie przez menedżery kolejek pośrednich

Współużytkowanie kanałów

Projektantem aplikacji jest możliwość wymuszenia przez aplikację określenia nazwy zdalnego menedżera kolejek wraz z nazwą kolejki lub utworzenie *definicji kolejki zdalnej* dla każdej kolejki zdalnej. Ta definicja zawiera nazwę zdalnego menedżera kolejek, nazwę kolejki oraz nazwę kolejki transmisji. W każdym przypadku wszystkie komunikaty ze wszystkich aplikacji adresowania kolejek w tym samym miejscu zdalnym mają swoje komunikaty wysłane przez tę samą kolejkę transmisji. Pokazano to na ilustracji [Rysunek 13 na stronie 52](#).



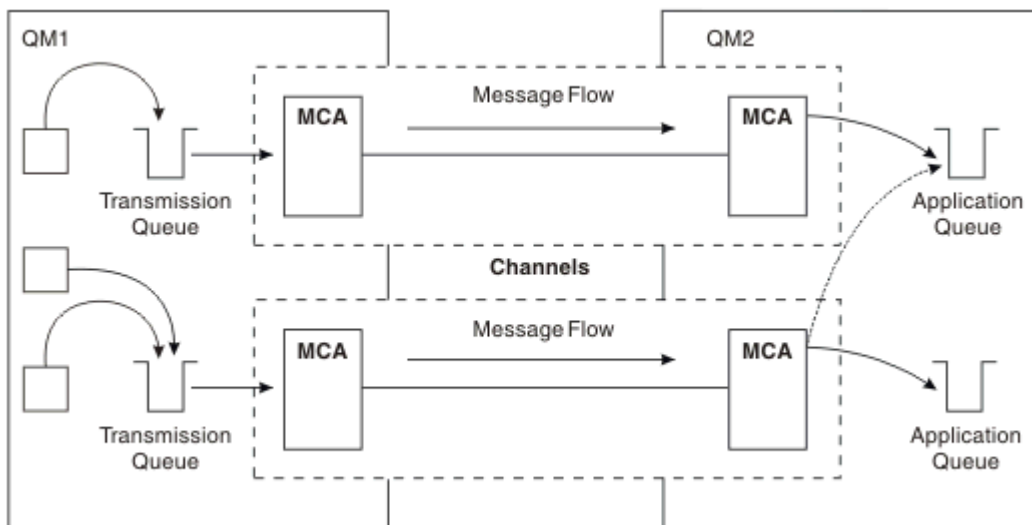
Rysunek 13. Współużytkowanie kolejki transmisji

Rysunek 13 na stronie 52 ilustruje, że komunikaty z wielu aplikacji do wielu kolejek zdalnych mogą korzystać z tego samego kanału.

Korzystanie z różnych kanałów

Jeśli istnieją komunikaty o różnych typach do wysłania między dwoma menedżerami kolejek, można zdefiniować więcej niż jeden kanał między dwoma menedżerami kolejek. Są czasy, kiedy potrzebne są alternatywne kanały, być może dla celów bezpieczeństwa, lub do handlu z prędkością dostawy przed sheer masowego ruchu komunikatów.

Aby skonfigurować drugi kanał, należy zdefiniować inny kanał i inną kolejkę transmisji, a następnie utworzyć definicję kolejki zdalnej określającą położenie i nazwę kolejki transmisji. Aplikacje mogą następnie używać dowolnego kanału, ale komunikaty są nadal dostarczane do tych samych kolejek docelowych. Pokazano to na ilustracji Rysunek 14 na stronie 52.



Rysunek 14. Korzystanie z wielu kanałów

Jeśli w celu określenia kolejki transmisji używane są definicje kolejek zdalnych, aplikacje muszą **nie** określać położenia (czyli docelowego menedżera kolejek). Jeśli tak się stanie, menedżer kolejek nie używa definicji kolejek zdalnych. Definicje kolejek zdalnych dają niezależność lokalizacji. Aplikacje mogą umieszczać komunikaty w kolejce *logicznej* bez wiedzy o miejscu, w którym znajduje się kolejka, i można zmienić kolejkę *fizyczną* bez konieczności zmiany aplikacji.

Korzystanie z grupowania

Każdy menedżer kolejek w klastrze definiuje kanał odbierający klastry. Gdy inny menedżer kolejek chce wysłać komunikat do tego menedżera kolejek, definiuje on odpowiedni kanał nadawczy klastra automatycznie. Na przykład, jeśli w klastrze istnieje więcej niż jedna instancja kolejki, kanał nadawczy klastra może być zdefiniowany dla dowolnego z menedżerów kolejek, które są hostem kolejki. Produkt IBM MQ korzysta z algorytmu zarządzania obciążeniem, który korzysta z procedury cykliczną w celu wybrania dostępnego menedżera kolejek w celu skierowania komunikatu do. Więcej informacji na ten temat zawiera sekcja [Klastry](#).

Informacje adresowe

Gdy aplikacja umieszcza komunikaty przeznaczone dla zdalnego menedżera kolejek, lokalny menedżer kolejek dodaje do nich nagłówki transmisji przed umieszczeniem ich w kolejce transmisji. Ten nagłówek zawiera nazwę kolejki docelowej i menedżera kolejek, tj. *informacje adresowe*.

W środowisku pojedynczego menedżera kolejek adres kolejki docelowej jest ustanawiany, gdy aplikacja otwiera kolejkę na potrzeby umieszczania komunikatów w kolejce. Ponieważ kolejka docelowa znajduje się w tym samym menedżerze kolejek, nie ma potrzeby uzyskania informacji adresowych.

W rozproszonym środowisku kolejkowania menedżer kolejek musi znać nie tylko nazwę kolejki docelowej, ale także położenie tej kolejki (to znaczy nazwę menedżera kolejek) oraz trasę do tej zdalnej lokalizacji (czyli kolejki transmisji). Te informacje adresowania są zawarte w nagłówku transmisji. Kanał odbierający usuwa nagłówki transmisji i używa informacji znajdujących się w nim w celu znalezienia kolejki docelowej.

Jeśli używana jest zdalna definicja kolejki, można uniknąć konieczności podania przez aplikację nazwy docelowego menedżera kolejek. Ta definicja określa nazwę kolejki zdalnej, nazwę zdalnego menedżera kolejek, do którego kierowane są komunikaty, oraz nazwę kolejki transmisji używanej do transportu komunikatów.

Co to są aliasy?

Alias są używane w celu zapewnienia jakości usługi dla komunikatów. Alias menedżera kolejek umożliwia administratorowi systemu zmianę nazwy docelowego menedżera kolejek bez konieczności zmiany aplikacji. Umożliwia również administratorowi systemu zmianę trasy do docelowego menedżera kolejek lub skonfigurowanie trasy, która obejmuje przejście przez wiele innych menedżerów kolejek (multihopping). Alias kolejki odpowiedzi zapewnia jakość usługi na potrzeby odpowiedzi.

Alias menedżera kolejek i aliasy kolejki odpowiedzi są tworzone przy użyciu definicji kolejki zdalnej, która ma pustą nazwę RNAME. Definicje te nie definiują rzeczywistych kolejek; są one używane przez menedżer kolejek do rozstrzygnięcia nazw kolejek fizycznych, nazw menedżerów kolejek i kolejek transmisji.

Definicje aliasów charakteryzują się pustą nazwą RNAME.

Rozstrzygnięcie nazwy kolejki

Rozstrzygnięcie nazw kolejek odbywa się w każdym menedżerze kolejek za każdym razem, gdy kolejka jest otwierana. Jego celem jest identyfikacja docelowej kolejki, docelowego menedżera kolejek (który może być lokalny) oraz trasy do tego menedżera kolejek (który może mieć wartość null). Przetłumaczona nazwa składa się z trzech części: nazwa menedżera kolejek, nazwa kolejki oraz, jeśli menedżer kolejek jest zdalny, kolejka transmisji.


Jeśli istnieje definicja kolejki zdalnej, nie są przywoływane żadne definicje aliasów. Nazwa kolejki podana przez aplikację jest tłumaczona na nazwę kolejki docelowej, zdalnego menedżera kolejek oraz kolejkę transmisji określoną w definicji kolejki zdalnej. Więcej szczegółowych informacji na temat rozstrzygnięcia nazw kolejek zawiera sekcja [Rozdzielczość nazwy kolejki](#).

Jeśli nie istnieje definicja kolejki zdalnej i określona została nazwa menedżera kolejek lub została ona rozstrzygnięta przez usługę nazw, menedżer kolejek sprawdza, czy istnieje definicja aliasu menedżera kolejek, która jest zgodna z podaną nazwą menedżera kolejek. Jeśli istnieje, informacje w nim zawarte są używane do rozstrzygnięcia nazwy menedżera kolejek na nazwę docelowego menedżera kolejek. Definicja

aliasu menedżera kolejek może być również używana do określenia kolejki transmisji do docelowego menedżera kolejek.

Jeśli rozstrzygnięta nazwa kolejki nie jest kolejką lokalną, to zarówno nazwa menedżera kolejek, jak i nazwa kolejki są uwzględniane w nagłówku przesyłania każdego komunikatu umieszczonego przez aplikację w kolejce transmisji.

Używana kolejka transmisji zwykle ma taką samą nazwę jak rozstrzygnięty menedżer kolejek, o ile nie została zmieniona przez definicję kolejki zdalnej lub definicję aliasu menedżera kolejek. Jeśli taka kolejka transmisji nie została zdefiniowana, ale została zdefiniowana domyślna kolejka transmisji, to jest ona używana.

 Nazwy menedżerów kolejek działających w systemie z/OS są ograniczone do czterech znaków.

Definicje aliasów menedżera kolejek

Definicje aliasów menedżera kolejek mają zastosowanie, gdy aplikacja, która otwiera kolejkę w celu umieszczenia komunikatu, określa nazwę kolejki **oraz** nazwę menedżera kolejek.

Definicje aliasów menedżera kolejek mają trzy zastosowania:

- Podczas wysyłania komunikatów przeodzworowywanie nazwy menedżera kolejek
- Podczas wysyłania komunikatów, zmieniania lub określania kolejki transmisji
- Podczas odbierania komunikatów określenie, czy lokalny menedżer kolejek jest docelowym miejscem docelowym dla tych komunikatów.

Komunikaty wychodzące-zmiana odwzorowania nazwy menedżera kolejek

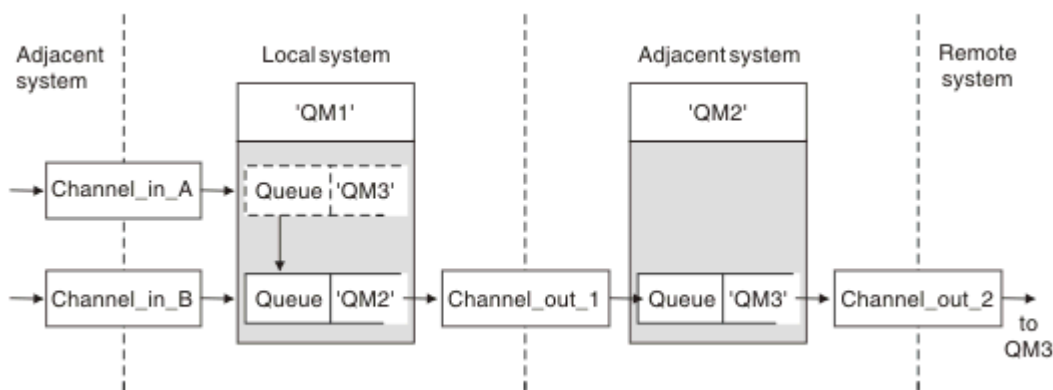
Definicje aliasów menedżera kolejek mogą być używane do zmiany odwzorowania nazwy menedżera kolejek określonej w wywołaniu MQOPEN. Na przykład wywołanie MQOPEN określa nazwę kolejki produktu THISQ i nazwę menedżera kolejek YOURQM. W lokalnym menedżerze kolejek istnieje definicja aliasu menedżera kolejek, taka jak w następującym przykładzie:

```
DEFINE QREMOTE (YOURQM) QMNAME(REALQM)
```

Oznacza to, że rzeczywisty menedżer kolejek, który ma być używany, gdy aplikacja umieszcza komunikaty w menedżerze kolejek YOURQM, to REALQM. Jeśli lokalny menedżer kolejek ma wartość REALQM, umieszcza komunikaty w kolejce THISQ, która jest kolejką lokalną. Jeśli lokalny menedżer kolejek nie jest nazywany REALQM, kieruje on komunikat do kolejki transmisji o nazwie REALQM. Menedżer kolejek zmienia nagłówek transmisji na wartość REALQM zamiast YOURQM.

Komunikaty wychodzące-modyfikowanie lub określanie kolejki transmisji

Rysunek 15 na stronie 55 przedstawia scenariusz, w którym komunikaty pojawiają się w menedżerze kolejek QM1 z nagłówkami transmisji przedstawiających nazwy kolejek w menedżerze kolejek QM3. W tym scenariuszu produkt QM3 jest osiągalny przez wiele operacji, które można uzyskać za pomocą programu QM2.



Rysunek 15. Alias menedżera kolejek

Wszystkie komunikaty dla produktu QM3 są przechwytywane w produkcie QM1 z aliasem menedżera kolejek. Alias menedżera kolejek ma nazwę QM3 i zawiera definicję QM3 poprzez kolejkę transmisji QM2. Definicja wygląda podobnie jak w poniższym przykładzie:

```
DEFINE QREMOTE (QM3) RNAME(' ') RQMNAME(QM3) XMITQ(QM2)
```

Menedżer kolejek umieszcza komunikaty w kolejce transmisji QM2, ale nie zmienia nagłówka kolejki transmisji, ponieważ nazwa docelowego menedżera kolejek QM3 nie zmienia się.

Wszystkie komunikaty przychodzące do programu QM1 i zawierające nagłówek transmisji zawierający nazwę kolejki w QM2 są również umieszczane w kolejce transmisji QM2. W ten sposób komunikaty z różnymi miejscami docelowymi są gromadzone do wspólnej kolejki transmisji do odpowiedniego przylegającego systemu, do celów przekazywania danych do ich miejsc docelowych.

Komunikaty przychodzące-określanie miejsca docelowego

Odbierający agent MCA otwiera kolejkę przywoływaną w nagłówku transmisji. Jeśli istnieje definicja aliasu menedżera kolejek o takiej samej nazwie, jak nazwa przywoływanego menedżera kolejek, nazwa menedżera kolejek otrzymana w nagłówku transmisji jest zastępowana nazwą RQMNAME z tej definicji.

Ten proces ma dwa zastosowania:

- Kierowanie komunikatów do innego menedżera kolejek
- Zmiana nazwy menedżera kolejek tak, aby była taka sama, jak nazwa lokalnego menedżera kolejek

Definicje aliasów kolejki odpowiedzi

Definicja aliasu kolejki odpowiedzi określa alternatywne nazwy dla informacji o odpowiedzi w deskrytorze komunikatu. Zaletą tego jest to, że można zmienić nazwę kolejki lub menedżera kolejek bez konieczności modyfikowania aplikacji.

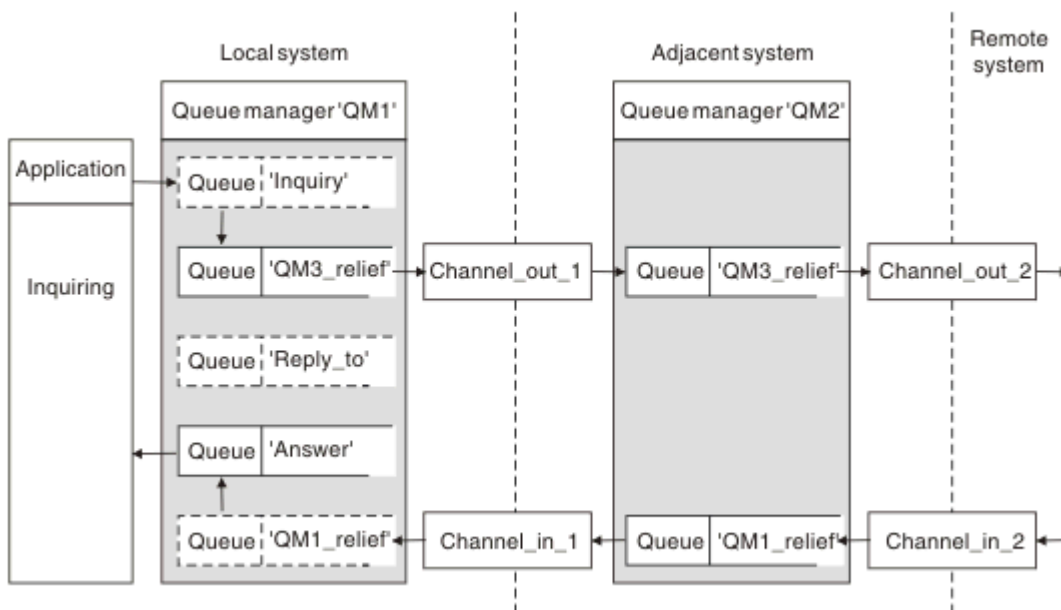
Rozstrzygnięcie nazwy kolejki

Gdy aplikacja odpowiada na komunikat, używa danych w *deskrytorze komunikatu* komunikatu, który otrzymał, aby znaleźć nazwę kolejki, na którą ma zostać wysłana odpowiedź. Aplikacja wysyłający wskazuje, gdzie są wysyłane odpowiedzi i dołącza te informacje do swoich komunikatów. Ta koncepcja musi być koordynowana jako część projektu aplikacji.

Rozstrzygnięcie nazw kolejek odbywa się na wysyłającym końcu aplikacji, zanim komunikat zostanie umieszczony w kolejce. W związku z tym rozstrzygnięcie nazwy kolejki występuje przed interakcją ze zdalną aplikacją, do której wysyłany jest komunikat. Jest to jedyna sytuacja, w której rozstrzygnięcie nazw odbywa się w czasie, gdy kolejka nie jest otwierana.

Rozstrzygnięcie nazw kolejek przy użyciu aliasu menedżera kolejek

Zwykle aplikacja określa kolejkę zwrotną i pozostawia pustą nazwę menedżera kolejek odpowiedzi. Menedżer kolejek kończy swoją własną nazwę w czasie umieszczania. Ta metoda działa dobrze, z wyjątkiem sytuacji, gdy istnieje potrzeba użycia alternatywnego kanału dla odpowiedzi, na przykład kanału, który używa kolejki transmisji QM1_relief zamiast domyślnego kanału powrotu, który korzysta z kolejki transmisji QM1. W tej sytuacji nazwy menedżerów kolejek określone w nagłówkach kolejki transmisji nie są zgodne z nazwami menedżera kolejek "real", ale są ponownie określone przy użyciu definicji aliasów menedżera kolejek. W celu zwrócenia odpowiedzi na alternatywne trasy konieczne jest odwzorowanie danych kolejki zwrotnej, a także użycie definicji aliasów kolejki odpowiedzi.



Rysunek 16. Alias kolejki odpowiedzi używany do zmiany położenia odpowiedzi

W przykładzie w programie Rysunek 16 na stronie 56:

1. Aplikacja umieszcza komunikat przy użyciu wywołania MQPUT i określa następujące informacje w deskrytorze komunikatu:

```
ReplyToQ='Reply_to'  
ReplyToQMgr=' '
```

Wartość ReplyTomenedżera kolejek musi być pusta, aby można było użyć aliasu kolejki odpowiedzi.

2. Użytkownik tworzy definicję aliasu kolejki odpowiedzi o nazwie Reply_to, która zawiera nazwę Answer, oraz nazwę menedżera kolejek QM1_relief.

```
DEFINE QREMOTE ('Reply_to') RNAME ('Answer')  
RQMNAME ('QM1_relief')
```

3. Komunikaty są wysyłane z deskrytorem komunikatów przedstawiających ReplyToQ='Answer' i ReplyToQMgr='QM1_relief'.
4. Specyfikacja aplikacji musi zawierać informacje, które odpowiedzi mają być znalezione w kolejce Answer, a nie w Reply_to.

Aby przygotować się do odpowiedzi, należy utworzyć równoległy kanał zwrotny, definiując:

- W QM2: kolejka transmisji o nazwie QM1_relief

```
DEFINE QLOCAL ('QM1_relief') USAGE(XMITQ)
```

- W QM1, alias menedżera kolejek QM1_relief

```
DEFINE QREMOTE ('QM1_relief') RNAME() RQMNAME(QM1)
```

Ten alias menedżera kolejek kończy łańcuch równoległych kanałów zwrotu i przechwytuje komunikaty dla QM1.

Jeśli uważasz, że warto to zrobić w przyszłości, upewnij się, że aplikacje korzystają z aliasu od początku. W tej chwili jest to normalny alias kolejki do kolejki odpowiedzi, ale później może zostać zmieniony na alias menedżera kolejek.

Nazwa kolejki odpowiedzi

W przypadku kolejek należy zachować ostrożność. Przyczyna, dla której aplikacja umieszcza nazwę kolejki odpowiedzi w komunikacie, polega na tym, że może ona określać kolejkę, do której wysyłane są jej odpowiedzi. Podczas tworzenia definicji aliasu kolejki odpowiedzi z tą nazwą nie można mieć rzeczywistej kolejki odpowiedzi (czyli lokalnej definicji kolejki) o tej samej nazwie. Dlatego definicja aliasu kolejki odpowiedzi musi zawierać nową nazwę kolejki, a także nazwę menedżera kolejek, a specyfikacja aplikacji musi zawierać informacje o tym, że odpowiedzi znajdują się w tej innej kolejce.

Aplikacje muszą teraz pobierać komunikaty z innej kolejki niż ta, która została nazwana jako kolejka zwrotna, gdy wstawiono oryginalny komunikat.

Komponenty klastra

Klaster składa się z menedżerów kolejek, repozytoriów klastra, kanałów klastra i kolejek klastra.

Informacje na temat poszczególnych komponentów klastra znajdują się w następujących podtematach:

Pojęcia pokrewne

[Porównanie grupowania i rozproszonego kolejkowania](#)

Zadania pokrewne

[Konfigurowanie klastra menedżera kolejek](#)

[Konfigurowanie nowego klastra](#)


Repozytorium klastra

Repozytorium to zbiór informacji o menedżerach kolejek, które są elementami klastra.

Informacje o repozytorium obejmują nazwy menedżerów kolejek, ich położenia, kanały, kolejki, które są hostem, a także inne informacje. Informacje te są zapisywane w postaci komunikatów w kolejce o nazwie

SYSTEM.CLUSTER.REPOSITORY.QUEUE. Ta kolejka jest jednym z obiektów domyślnych. 

W systemie Multiplatforms jest on definiowany podczas tworzenia menedżera kolejek produktu IBM

MQ.  W produkcie IBM MQ for z/OS jest on zdefiniowany jako część dostosowania menedżera kolejek.

Pełne repozytorium i częściowe repozytorium

Zwykle dwa menedżery kolejek w klastrze przechowują pełne repozytorium. Pozostałe menedżery kolejek przechowują częściowe repozytorium.

Menedżer kolejek, który udostępnia pełny zestaw informacji o każdym menedżerze kolejek w klastrze, ma pełne repozytorium. Inne menedżery kolejek w klastrze zawierają częściowe repozytoria zawierające podzbiór informacji w pełnych repozytoriach.

Częściowe repozytorium zawiera informacje tylko o tych menedżerach kolejek, z którymi menedżer kolejek musi wymieniać komunikaty. Menedżery kolejek żądają aktualizacji potrzebnych informacji, aby w razie zmian menedżer kolejek pełnego repozytorium wystąpił do nowych informacji. Przez wiele czasu częściowe repozytorium zawiera wszystkie informacje, które musi wykonać menedżer kolejek w klastrze. Jeśli menedżer kolejek potrzebuje informacji dodatkowych, wysyła zapytania o pełne repozytorium i aktualizuje swoje częściowe repozytorium. Menedżery kolejek używają kolejki `SYSTEM.CLUSTER.COMMAND.QUEUE` do żądania i odbierania aktualizacji do repozytoriów.


Podczas migrowania menedżerów kolejek, które należą do klastra, należy przeprowadzić migrację pełnych repozytoriów przed częściowymi repozytoriami. Jest to spowodowane tym, że starsze repozytorium nie może przechowywać nowszych atrybutów wprowadzonych w nowszej wersji. Toleruje je, ale ich nie przechowuje.

Menedżer kolejek klastra

Menedżer kolejek klastra to menedżer kolejek, który jest elementem klastra.

Menedżer kolejek może należeć do kilku klastrów. Każdy menedżer kolejek klastra musi mieć nazwę, która jest unikalna w obrębie wszystkich klastrów, których jest elementem.

Menedżer kolejek klastra może udostępniać kolejki, które reklamuje się do innych menedżerów kolejek w klastrze. Nie musi to jednak robić. Zamiast tego mogą one zasilać komunikaty do kolejek udostępnianych w innym miejscu w klastrze i odbierać tylko odpowiedzi, które są bezpośrednio do niego kierowane.

 W produkcie IBM MQ for z/OS menedżer kolejek klastra może być elementem grupy współużytkowania kolejek. W tym przypadku współużytkuje on definicje kolejek z innymi menedżerami kolejek w tej samej grupie współużytkowania kolejek.

Menedżery kolejek klastra są autonomiczne. Mają pełną kontrolę nad kolejkami i kanałami, które definiują. Ich definicje nie mogą być modyfikowane przez inne menedżery kolejek (inne niż menedżery kolejek w tej samej grupie współużytkowania kolejek). Menedżery kolejek repozytorium nie sterują definicjami w innych menedżerach kolejek w klastrze. Posiadają pełny zestaw wszystkich definicji, do użycia w razie potrzeby. Klaster jest stowarzyszeniem menedżerów kolejek.

Po utworzeniu lub zmianie definicji w menedżerze kolejek klastra informacje te są wysyłane do menedżera kolejek pełnego repozytorium. Inne repozytoria w klastrze są aktualizowane w późniejszym czasie.

Menedżer kolejek pełnego repozytorium

Menedżer kolejek pełnego repozytorium jest menedżerem kolejek klastra, który zawiera pełną reprezentację zasobów klastra. Aby zapewnić dostępność, należy skonfigurować co najmniej dwa menedżery kolejek pełnego repozytorium w każdym klastrze. Menedżery kolejek pełnego repozytorium odbierają informacje wysyłane przez inne menedżery kolejek w klastrze i aktualizują ich repozytoria. Wysyłają komunikaty do siebie nawzajem, aby mieć pewność, że są na bieżąco z nowymi informacjami o klastrze.

Menedżery kolejek i repozytoria

Każdy klaster ma co najmniej jeden (najlepiej dwa) menedżery kolejek, w których znajdują się pełne repozytoria informacji o menedżerach kolejek, kolejkach i kanałach w klastrze. Te repozytoria zawierają również żądania od innych menedżerów kolejek w klastrze w celu aktualizacji informacji.

Pozostałe menedżery kolejek przechowują częściowe repozytorium zawierające informacje na temat podzbioru kolejek i menedżerów kolejek, z którymi muszą się komunikować. Menedżery kolejek budują swoje częściowe repozytoria, wykonując zapytania, gdy najpierw muszą uzyskać dostęp do innej kolejki lub innego menedżera kolejek. Żądają one, aby były one powiadamiane o wszelkich nowych informacjach dotyczących tej kolejki lub menedżera kolejek.

Każdy menedżer kolejek przechowuje swoje informacje o repozytorium w komunikatach w kolejce o nazwie SYSTEM.CLUSTER.REPOSITORY.QUEUE. Menedżery kolejek wymieniają informacje o repozytorium w komunikatach w kolejce o nazwie SYSTEM.CLUSTER.COMMAND.QUEUE.

Każdy menedżer kolejek, który dołącza do klastra, definiuje nadawcę klastra CLUSSDR, kanał do jednego z repozytoriów. Od razu dowiaduje się, które inne menedżery kolejek w klastrze posiadają pełne repozytoria. Od tego czasu menedżer kolejek może zażądać informacji z dowolnego z repozytoriów. Gdy menedżer kolejek wysyła informacje do wybranego repozytorium, wysyła również informacje do jednego innego repozytorium (jeśli istnieje).


Pełne repozytorium jest aktualizowane, gdy menedżer kolejek udostępniający je otrzymuje nowe informacje od jednego z powiązanych z nim menedżerów kolejek. Nowe informacje są również wysyłane do innego repozytorium w celu zmniejszenia ryzyka opóźnienia w przypadku, gdy menedżer kolejek repozytorium nie jest w stanie serwisowania. Ponieważ wszystkie informacje są wysyłane dwa razy, repozytoria muszą usunąć duplikaty. Każdy element informacji zawiera numer kolejny, który jest używany przez repozytoria do identyfikowania duplikatów. Wszystkie repozytoria są przechowywane razem ze sobą przez wymianę komunikatów.

Kolejki klastra

Kolejka klastra to kolejka udostępniana przez menedżer kolejek klastra innym menedżerom kolejek w klastrze.

Definicja kolejki klastra jest ogłaszana w innych menedżerach kolejek w klastrze. Inne menedżery kolejek w klastrze mogą umieszczać komunikaty w kolejce klastra bez konieczności stosowania odpowiadającej jej definicji kolejki zdalnej. Kolejka klastra może zostać ogłoszona w więcej niż jednym klastrze przy użyciu listy nazw klastra.

Po ogłoszeniu kolejki każdy menedżer kolejek w klastrze może umieszczać w niej komunikaty. Aby umieścić komunikat, menedżer kolejek musi ustalić, w którym repozytorium pełnym znajduje się kolejka. Następnie do komunikatu w kolejce transmisji klastra dodawane są niektóre informacje o kierowaniu.

 Kolejka klastra może być kolejką współużytkowaną przez członków grupy współużytkowania kolejki w programie IBM MQ for z/OS.

Zadania pokrewne

[Definiowanie kolejek klastra](#)

Porównanie między kolejkami współużytkowanymi i kolejkami klastra

Informacje te zostały zaprojektowane w celu ułatwienia porównywania kolejek współużytkowanych i kolejek klastra oraz podejmowania decyzji, które mogą być bardziej odpowiednie dla danego systemu.

Koszty inicjatora kanału

W kolejkach klastra komunikaty są wysyłane przez kanały, co pozwala na dodawanie kosztów inicjatora kanału oprócz kosztów aplikacji. W sieci są koszty, ponieważ kanały dostają i umieszczają komunikaty. Te koszty nie są obecne w kolejkach współużytkowanych, co oznacza, że podczas przenoszenia komunikatów między menedżerami kolejek w grupie współużytkowania kolejek są używane mniej mocy przetwarzania niż kolejki klastra.

Dostępność komunikatów

W przypadku umieszczania w kolejce kolejki klastra wysyłają komunikat do jednego z menedżerów kolejek z aktywnymi kanałami połączonymi z menedżerem kolejek. W menedżerze kolejek zdalnych, jeśli aplikacje używane do przetwarzania komunikatów nie działają, komunikaty nie są przetwarzane i oczekują na uruchomienie aplikacji. Podobnie, jeśli menedżer kolejek jest wyłączony, wszystkie komunikaty w menedżerze kolejek nie są udostępniane do momentu zrestartowania menedżera kolejek. Te instancje pokazują niższy poziom dostępności komunikatów niż w przypadku korzystania z kolejek współużytkowanych.

W przypadku korzystania z kolejek współużytkowanych każda aplikacja w grupie współużytkowania kolejek może uzyskać wysłane komunikaty. Jeśli jeden menedżer kolejek zostanie zamknięty w grupie współużytkowania kolejek, komunikaty będą dostępne dla innych menedżerów kolejek, co zapewnia wyższą dostępność komunikatów niż w przypadku korzystania z kolejek klastra.

pojemność

Mechanizm sprzęgający jest droższy od dysku, dlatego koszt przechowywania 1 000 000 komunikatów w kolejce lokalnej jest niższy niż w przypadku narzędzia CF z wystarczającą ilością miejsca do przechowywania tej samej liczby komunikatów.

Wysyłanie do innych menedżerów kolejek

Komunikaty kolejki współużytkowanej są dostępne tylko w grupie współużytkowania kolejek. Aby użyć menedżera kolejek poza grupą współużytkowania kolejek, należy użyć kanałów. W celu równoważenia obciążenia można użyć grupowania między wieloma zdalnymi rozproszonymi menedżerami kolejek.

Równoważenie obciążenia

Można użyć technologii klastrowej w celu nadania wagi, do której kanały i menedżery kolejek są proporcjonalne do wysłanych komunikatów. Na przykład można wysłać 60% komunikatów do jednego menedżera kolejek, a 40% komunikatów do innego menedżera kolejek. Ta instancja nie zależy od możliwości przetwarzania pracy przez zdalny menedżer kolejek. System z pierwszym menedżerem kolejek może być przeciążony, a system z drugim menedżerem kolejek może być beczynny, ale większość komunikatów nadal znajduje się w pierwszym menedżerze kolejek.

W przypadku współużytkowanych kolejek dwa systemy CICS mogą uzyskać komunikaty. Jeśli jeden system jest przeciążony, drugi system przejmuje większość obciążenia.

Kanały klastra

W każdym pełnym repozytorium ręcznie definiowany jest kanał odbierający klastry oraz zestaw kanałów nadawczych klastra w celu nawiązania połączenia z każdym innym pełnym repozytorium w klastrze. Po dodaniu częściowego repozytorium można ręcznie zdefiniować kanał odbiorczy klastra oraz pojedynczy kanał nadawczy klastra, który łączy się z jednym z pełnych repozytoriów. Kolejne kanały nadawcze klastra są definiowane automatycznie przez klastrer w razie potrzeby. Automatycznie zdefiniowane kanały nadawcze klastra przyjmują swoje atrybuty z odpowiedniej definicji kanału odbierającego klastry w odbierającym menedżerze kolejek.

Kanał odbierający klastry: CLUSRCVR

Definicja kanału CLUSRCVR definiuje koniec kanału, na którym menedżer kolejek klastra może odbierać komunikaty od innych menedżerów kolejek w klastrze.

Dla każdego menedżera kolejek klastra należy zdefiniować co najmniej jeden kanał CLUSRCVR. Definiując kanał CLUSRCVR, menedżer kolejek wyświetla inne menedżery kolejek klastra, które są dostępne do odbierania komunikatów.

Definicja kanału CLUSRCVR umożliwia również innym menedżerom kolejek automatyczne definiowanie odpowiednich definicji kanału nadawczego klastra. Zapoznaj się z sekcją “Automatycznie zdefiniowane kanały nadawcze klastra” na stronie 61 w tym artykule.

Kanał wysyłający klastry: CLUSSDR

Użytkownik ręcznie definiuje kanał CLUSSDR z każdego pełnego menedżera kolejek repozytorium do każdego innego menedżera kolejek pełnego repozytorium w klastrze. Wszystkie aktualizacje wymieniane przez pełny przepływ repozytoriów są wykonywane wyłącznie na tych kanałach. Samodzielnie definiując te kanały, można jawnie sterować siecią pełnych repozytoriów.

Po dodaniu menedżera kolejek częściowego repozytorium do klastra należy ręcznie zdefiniować pojedynczy kanał CLUSSDR w celu połączenia się z jednym z pełnych repozytoriów. Niewiele ma znaczenia, które repozytorium pełne zostanie wybrane, ponieważ po początkowym kontakcie dalsze obiekty menedżera kolejek klastra dla menedżera kolejek, w tym kanały CLUSSDR, są definiowane automatycznie, jeśli jest to konieczne. Dzięki temu menedżer kolejek może wysyłać informacje o klastrze do dowolnego pełnego repozytorium i wysyłać komunikaty do dowolnego menedżera kolejek w klastrze.

Jak wyjaśniono w sekcji tego artykułu, automatycznie zdefiniowane kanały nadawcze są oparte na konfiguracji kanału odbiorczego klastra. Z tego względu wszystkie właściwości kanałów ustawione w kanałach klastra powinny być ustawione tak, aby były zgodne z CLUSSDR i kanałami odbiorczymi klastra lub ustawione tylko w kanałach odbiorczych klastra.

Należy ręcznie zdefiniować kanały CLUSSDR z powodów opisanych wcześniej. Oznacza to, że początkowo można połączyć częściowe repozytorium z pełnym repozytorium lub połączyć dwa pełne repozytoria razem. Ręczne konfigurowanie kanału CLUSSDR, który łączy się z częściowym repozytorium lub z menedżerem kolejek, który nie znajduje się w klastrze, powoduje komunikaty o błędach, takie jak AMQ9427 i AMQ9428, które mają zostać wydane. Mimo że czasami może to być nieuniknione jako sytuacja tymczasowa, na przykład podczas modyfikowania położenia pełnego repozytorium, należy jak najszybciej usunąć definicję ręczną.

Automatycznie zdefiniowane kanały nadawcze klastra

Zwykle w przypadku dodawania menedżera kolejek repozytorium częściowego do klastra ręcznie należy zdefiniować tylko dwa kanały klastra w menedżerze kolejek:

- Nadawca klastra (CLUSSDR) Kanał do pełnego menedżera kolejek repozytorium dla klastra.
- Odbiornik klastra (CLUSRCVR) kanał.

Zdefiniowany przez użytkownika kanał CLUSSDR umożliwia menedżerowi kolejek nawiązanie początkowego kontaktu z klastrzem. Po początkowym kontakcie dalsze kanały CLUSSDR są definiowane automatycznie przez klastr w razie potrzeby.

Automatycznie zdefiniowany kanał CLUSSDR pobiera swoje atrybuty z odpowiedniej definicji kanału CLUSRCVR w odbierającym menedżerze kolejek. Nawet jeśli istnieje ręcznie zdefiniowany kanał CLUSSDR, używane są atrybuty z automatycznie zdefiniowanego kanału CLUSSDR. Załóżmy na przykład, że zdefiniowano kanał CLUSRCVR bez podawania numeru portu w parametrze **CONNNAME**, a następnie ręcznie zdefiniuj kanał CLUSSDR, który określa numer portu. Gdy automatycznie zdefiniowany kanał CLUSSDR zastąpi ręcznie zdefiniowany jeden, numer portu (pobierana z kanału CLUSRCVR) staje się pusty. Używany jest domyślny numer portu, a kanał kończy się niepowodzeniem.

Jeśli istnieją różnice w konfiguracji między ręcznie zdefiniowanym kanałem CLUSSDR i odpowiednią definicją kanału CLUSRCVR, niektóre różnice są stosowane natychmiast (na przykład parametry równoważenia obciążenia), a niektóre są aktywne tylko po restarcie kanału (na przykład w konfiguracji TLS).

Aby uniknąć nieporozumień, w miarę możliwości należy przestrzegać następujących wytycznych:

- Tylko ręcznie zdefiniuj kanały CLUSSDR, aby wskazywać na pełne repozytoria.
- W przypadku ręcznego zdefiniowania kanałów CLUSSDR należy skonfigurować je tak, aby identycznie dopasować odpowiednią definicję kanału CLUSRCVR do odbierającego menedżera kolejek.

Patrz także: [Praca z automatycznie zdefiniowanymi kanałami](#).

Pojęcia pokrewne

[Praca z automatycznie zdefiniowanymi kanałami](#)

[Praca z kolejkami transmisji klastra i kanałami nadawczym klastra](#)

Zadania pokrewne

[Konfigurowanie nowego klastra](#)

[Dodawanie menedżera kolejek do klastra](#)

Tematy klastra

Tematy klastra są tematami administracyjnymi ze zdefiniowanym atrybutem **cluster**. Informacje o tematach klastra są przekazywane do wszystkich elementów klastra i łączone z tematami lokalnymi w celu utworzenia fragmentów obszaru tematu, który obejmuje wiele menedżerów kolejek. Umożliwia to dostarczanie komunikatów opublikowanych w temacie w menedżerze kolejek do subskrypcji innych menedżerów kolejek w klastrze.

Po zdefiniowaniu tematu klastra w menedżerze kolejek definicja tematu klastra jest wysyłana do menedżerów kolejek pełnych repozytoriów. Następnie pełne repozytoria propagują definicję tematu klastra do wszystkich menedżerów kolejek w klastrze, udostępniając ten sam temat klastra publikatorom i subskrybentom w każdym menedżerze kolejek w klastrze. Menedżer kolejek, w którym został utworzony temat klastra jest nazywany hostem tematu klastra. Temat klastra może być używany przez dowolny menedżer kolejek w klastrze, ale wszystkie modyfikacje tematu klastra muszą być wprowadzane w menedżerze kolejek, w którym zdefiniowano temat (na hoście). Wówczas modyfikacja jest propagowana do wszystkich elementów klastra przez pełne repozytoria.

Informacje na temat konfigurowania tematów klastra do korzystania z *kierowania bezpośredniego* lub *kierowania hostami tematów* oraz dotyczące klastrowego dziedziczenia tematów i subskrypcji ze znakami wieloznacznymi zawiera sekcja [Definiowanie tematów klastra](#).

Więcej informacji na temat komend, których można używać do wyświetlania tematów klastra, można znaleźć w sekcji informacji pokrewnych.

Pojęcia pokrewne

[Praca z tematami administracyjnymi](#)

[Praca z subskrypcjami](#)



Odsyłacze pokrewne

[WYŚWIETL TEMAT](#)

[WYŚWIETL STATUS TPSTATUS](#)

[WYŚWIETL SUB](#)

Domyślne obiekty klastra

 W przypadku wielu platform domyślne obiekty klastra są włączane do zestawu domyślnych obiektów tworzonych automatycznie podczas definiowania menedżera kolejek.  W produkcji z/OS domyślne definicje obiektów klastra można znaleźć w przykładach dostosowania.

Uwaga: Domyślne definicje kanałów można zmienić w taki sam sposób, jak w przypadku dowolnej innej definicji kanału, uruchamiając komendy MQSC lub PCF. Nie zmieniaj domyślnych definicji kolejek, z wyjątkiem SYSTEM.CLUSTER.HISTORY.QUEUE.

SYSTEM.CLUSTER.COMMAND.QUEUE

Każdy menedżer kolejek w klastrze ma kolejkę lokalną o nazwie SYSTEM.CLUSTER.COMMAND.QUEUE, która jest używana do przesyłania komunikatów do pełnego repozytorium. Komunikat zawiera wszystkie nowe lub zmienione informacje o menedżerze kolejek lub wszystkie żądania dotyczące informacji o innych menedżerach kolejek. SYSTEM.CLUSTER.COMMAND.QUEUE jest zwykle puste.

SYSTEM.CLUSTER.HISTORY.QUEUE

Każdy menedżer kolejek w klastrze ma kolejkę lokalną o nazwie SYSTEM.CLUSTER.HISTORY.QUEUE. Produkt SYSTEM.CLUSTER.HISTORY.QUEUE służy do przechowywania historii informacji o stanie klastra w celach serwisowych.

W domyślnych ustawieniach obiektu wartość SYSTEM.CLUSTER.HISTORY.QUEUE jest ustawiona na PUT (ENABLED). Aby zablokować gromadzenie danych historycznych, należy zmienić ustawienie na PUT (DISABLED).

SYSTEM.CLUSTER.REPOSITORY.QUEUE

Każdy menedżer kolejek w klastrze ma kolejkę lokalną o nazwie SYSTEM.CLUSTER.REPOSITORY.QUEUE. Ta kolejka jest używana do przechowywania wszystkich pełnych informacji o repozytorium. Ta kolejka nie jest normalnie pusta.

SYSTEM.CLUSTER.TRANSMIT.QUEUE

Dla każdego menedżera kolejek istnieje definicja kolejki lokalnej o nazwie SYSTEM.CLUSTER.TRANSMIT.QUEUE. SYSTEM.CLUSTER.TRANSMIT.QUEUE jest domyślną kolejką transmisji dla wszystkich komunikatów do wszystkich kolejek i menedżerów kolejek, które znajdują się w klastrach. Domyślną kolejkę transmisji dla każdego kanału nadawczego klastra można zmienić na SYSTEM.CLUSTER.TRANSMIT.ChannelName, zmieniając atrybut menedżera kolejek DEFCLXQ. Nie można usunąć SYSTEM.CLUSTER.TRANSMIT.QUEUE. Jest on również używany do definiowania autoryzacji sprawdzających, czy używana jest domyślna kolejka transmisji SYSTEM.CLUSTER.TRANSMIT.QUEUE czy SYSTEM.CLUSTER.TRANSMIT.ChannelName.

SYSTEM.DEF.CLUSRCVR

Każdy klaster ma domyślną definicję kanału CLUSRCVR o nazwie SYSTEM.DEF.CLUSRCVR. Produkt SYSTEM.DEF.CLUSRCVR jest używany do dostarczania wartości domyślnych dla wszystkich atrybutów, które nie są określane podczas tworzenia kanału odbiorczego klastra w menedżerze kolejek w klastrze.

SYSTEM.DEF.CLUSSDR

Każdy klaster ma domyślną definicję kanału CLUSSDR o nazwie SYSTEM.DEF.CLUSSDR. Produkt SYSTEM.DEF.CLUSSDR jest używany do dostarczania wartości domyślnych dla wszystkich atrybutów, które nie są określane podczas tworzenia kanału nadawczego klastra w menedżerze kolejek w klastrze.

Pojęcia pokrewne

[Praca z domyślnymi obiektami klastra](#)

Przesyłanie komunikatów publikowania/subskrypcji

Przesyłanie komunikatów w trybie publikowania/subskrypcji pozwala na oddzielenie dostawcy informacji od konsumentów tych informacji. Aplikacja wysyłający i odbierający nie muszą znać żadnych informacji o sobie, aby informacje były wysyłane i odbierane.

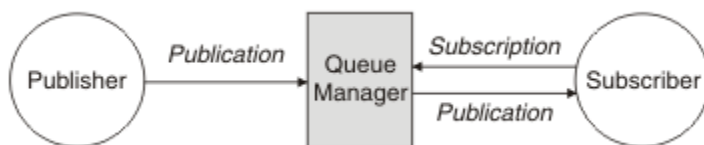
Zanim aplikacja typu punkt z punktem IBM MQ może wysłać komunikat do innej aplikacji, musi on wiedzieć coś o tej aplikacji. Na przykład, musi znać nazwę kolejki, do której mają być wysyłane informacje, a także nazwę menedżera kolejek.

Program IBM MQ publikowania/subskrybowania usuwa potrzebę, aby aplikacja wiedziała cokolwiek o aplikacji docelowej. Cała aplikacja wysyłający musi to zrobić:

- Umieść komunikat IBM MQ zawierający informacje, które mają być używane przez aplikację.
- Przypisz komunikat do tematu, który oznacza temat informacji.
- Pozwól IBM MQ zająć się dystrybucją tych informacji.

Podobnie, aplikacja docelowa nie musi wiedzieć nic o źródle otrzymywanych informacji.

Na poniższym rysunku przedstawiono najprostszy system publikowania/subskrypcji. Istnieje jeden publikator, jeden menedżer kolejek i jeden subskrybent. Subskrybent jest dokonany przez subskrybenta w menedżerze kolejek. Publikacja jest wysyłana z publikatora do menedżera kolejek, a następnie publikacja jest przekazywana przez menedżera kolejek do subskrybenta.



Rysunek 17. Prosta konfiguracja publikowania/subskrypcji

Typowy system publikowania/subskrypcji ma więcej niż jeden publikator i więcej niż jeden subskrybent na wielu różnych tematach, a często ma więcej niż jeden menedżer kolejek. Aplikacja może być zarówno publikatorem, jak i subskrybentem.

Kolejną istotną różnicą między przesyłaniem komunikatów w trybie publikowania/subskrypcji a punktem punkt z punktem jest to, że komunikat wysyłany do kolejki punkt z punktem jest przetwarzany tylko przez pojedynczą aplikację użytkownika. Komunikat opublikowany w temacie publikowania/subskrypcji, w którym więcej niż jeden subskrybent zarejestrował zainteresowanie, jest przetwarzany przez każdego zainteresowanego subskrybenta.

Komponenty publikowania/subskrybowania

Publikowanie/subskrypcja to mechanizm, za pomocą którego subskrybenci mogą otrzymywać informacje, w formie komunikatów, od wydawców. Interakcje między publikatorami i subskrybentami są kontrolowane przez menedżery kolejek przy użyciu standardowych udogodnień produktu IBM MQ .

Typowy system publikowania/subskrypcji ma więcej niż jeden publikator i więcej niż jeden subskrybent na wielu różnych tematach, a często ma więcej niż jeden menedżer kolejek. Aplikacja może być zarówno publikatorem, jak i subskrybentem.

Dostawca informacji jest nazywany *publikatorem*. Publikatory dostarczają informacji o temacie, bez potrzeby poznawania tych informacji o aplikacjach, które są zainteresowane tą informacją. Publikatory generują te informacje w postaci komunikatów, nazywanych *publikacjami* , które mają być publikowane i definiujące temat tych komunikatów.

Konsument informacji jest nazywany *subskrybentem*. Subskrybenci tworzą *subskrypcje* , które opisują temat, w którym subskrybent jest zainteresowany. W związku z tym subskrypcja określa, które publikacje są przekazywane do subskrybenta. Subskrybenci mogą wykonywać wiele subskrypcji i mogą otrzymywać informacje od wielu różnych publikatorów.

Opublikowane informacje są wysyłane w komunikacie IBM MQ , a temat informacji jest identyfikowany przez *temat*. Publikator określa temat, w którym publikuje informacje, a subskrybent określa tematy, na temat których chce otrzymywać publikacje. Subskrybent wysyła informacje o tylko tych tematach, do których się subskrybuje.

Jest to istnienie tematów, które pozwalają na oddzielenie dostawców i konsumentów informacji w przesyłaniu komunikatów w trybie publikowania/subskrypcji, usuwając potrzebę uwzględnienia określonego miejsca docelowego w każdym komunikacie, który jest wymagany w przesyłaniu komunikatów w trybie punkt z punktem.

Interakcje między publikatorami i subskrybentami są kontrolowane przez menedżer kolejek. Menedżer kolejek odbiera komunikaty od publikatorów i subskrypcje z subskrybentów (do zakresu tematów). Zadanie menedżera kolejek polega na skierowaniu opublikowanych komunikatów do subskrybentów, które zarejestrowały zainteresowanie tematem komunikatów.

Standardowe urządzenia IBM MQ są używane do dystrybuowania komunikatów, dzięki czemu aplikacje mogą korzystać ze wszystkich funkcji dostępnych dla istniejących aplikacji produktu IBM MQ . Oznacza to, że można używać trwałych komunikatów w celu uzyskania gwarantowanego dostarczenia tylko jednorazowo oraz że komunikaty mogą być częścią transakcyjnej jednostki pracy, aby zapewnić, że komunikaty są dostarczane do subskrybenta tylko wtedy, gdy są zatwierdzone przez publikatora.

Wydawcy i publikacje

W produkcie IBM MQ publikowanie/subskrypcja publikatora to aplikacja, która udostępnia informacje o określonym temacie dla menedżera kolejek w postaci standardowego komunikatu produktu IBM MQ o nazwie publikowania/publikacji. Publikator może publikować informacje o więcej niż jednym temacie.

Publikatory używają komendy MQPUT w celu umieszczenia komunikatu w poprzednio otwartym temacie. Ten komunikat jest publikacją. Następnie lokalny menedżer kolejek kieruje publikacją do wszystkich subskrybentów, którzy mają subskrypcje tematu publikacji. Opublikowany komunikat może być używany przez więcej niż jednego subskrybenta.

Oprócz dystrybucji publikacji do wszystkich lokalnych subskrybentów, które mają odpowiednie subskrypcje, menedżer kolejek może również dystrybuować publikację do wszystkich innych połączonych z nim menedżerów kolejek, bezpośrednio lub za pośrednictwem sieci menedżerów kolejek, które mają subskrybentów tematu.

W sieci publikowania/subskrybowania w produkcie IBM MQ aplikacja publikowania może być również subskrybentem.

Publikacje w punkcie synchronizacji

Publikatorzy mogą wywołać wywołania MQPUT lub MQPUT1 w punkcie synchronizacji, aby uwzględnić wszystkie komunikaty dostarczone do subskrybentów w jednostce pracy. Jeśli określono opcję MQPMO_RETAIN albo opcje dostarczania tematu NPMSGDLV lub PMSGDLV z wartościami ALL lub ALLDUR, menedżer kolejek używa wewnętrznych wywołań MQPUT lub MQPUT1 w punkcie synchronizacji, w zasięgu wywołania MQPUT lub MQPUT1 publikatora.

Informacje o stanie i zdarzeniach

Publikacje można podzielić na kategorie jako publikacje państwowe, takie jak bieżąca cena akcji lub publikacje o zdarzeniach, takie jak handel tym zapasem.

Publikacje o stanie

Publikacje województwo zawierają informacje o aktualnym stanie czegoś, takich jak cena akcji lub aktualny wynik w meczu piłki nożnej. W momencie wystąpienia zdarzenia (na przykład zmiany ceny akcji lub zmiany wyniku meczu) poprzednia informacja o stanie nie jest już dłużej wymagana, ponieważ zostaje zastąpiona przez nową informację.

Subskrybent będzie chciał otrzymywać bieżącą wersję informacji o stanie podczas uruchamiania i wysłać nowe informacje za każdym razem, gdy zmieni się stan.

Jeśli publikacja zawiera informacje o stanie, często jest ona publikowana jako publikacja zachowana. Nowy subskrybent zwykle chce natychmiast wyświetlić informacje o stanie; subskrybent nie chce czekać na zdarzenie, które powoduje ponowne opublikowanie informacji. Subskrybenci będą automatycznie otrzymywać zachowaną publikację tematu w momencie subskrybowania, chyba że subskrybent korzysta z opcji MQSO_PUBLICATIONS_ON_REQUEST lub MQSO_NEW_PUBLICATIONS_ONLY.

Publikacje dotyczące zdarzeń

Publikacje dotyczące zdarzeń zawierają informacje na temat poszczególnych zdarzeń, takich jak handel zasobami lub punktacja danego celu. Każde zdarzenie jest niezależne od innych zdarzeń.

Subskrybent będzie chciał otrzymywać informacje o zdarzeniach w taki sposób, w jaki się one zdarzają.

Zachowane publikacje

Domyślnie po wystaniu publikacji do wszystkich zainteresowanych subskrybentów jest ona usuwana. Publikator może jednak określić, że kopia publikacji jest zachowywana w taki sposób, aby mogła być wysłana do przyszłych subskrybentów, którzy zarejestrują zainteresowanie tematem.

Usuwanie publikacji po ich wystaniu do wszystkich zainteresowanych subskrybentów jest odpowiednie dla informacji o zdarzeniach, ale nie zawsze nadaje się do informacji o stanie. Zachowując komunikat, nowi subskrybenci nie muszą czekać na ponowne opublikowanie informacji, zanim zostaną wyświetlone początkowe informacje o stanie. Na przykład, subskrybent z abonamentem w cenie akcji otrzyma od razu aktualną cenę, bez czekania na zmianę ceny akcji (i w związku z tym zostać ponownie opublikowane).

Menedżer kolejek może zachować tylko jedną publikację dla każdego tematu, tak więc istniejąca zachowana publikacja tematu zostanie usunięta po nadejściu nowej zachowanej publikacji do menedżera kolejek. Jednak usunięcie istniejącej publikacji może nie odbywać się synchronicznie wraz z nadejściem nowej zachowanej publikacji. W związku z tym, o ile to możliwe, nie ma więcej niż jednego publikatora wysyłającego zachowane publikacje na dowolny temat.

Subskrybenci mogą określić, że nie chcą otrzymywać zachowanych publikacji przy użyciu opcji subskrypcji MQSO_NEW_PUBLICATIONS_ONLY. Dotychczasowi subskrybenci mogą poprosić o powielenie kopii zachowanych publikacji, które mają zostać do nich wysłane.

Czasami użytkownik może nie chcieć zachować publikacji, nawet w przypadku informacji o stanie:

- Jeśli wszystkie subskrypcje tematu zostaną wprowadzone przed dokonaniem publikacji na ten temat, a użytkownik nie będzie oczekiwał lub nie zezwoli na nowe subskrypcje, nie ma potrzeby zatrzymywania publikacji, ponieważ są one dostarczane do kompletnego zestawu subskrybentów po raz pierwszy, gdy są one publikowane.
- Jeśli publikacje występują często, na przykład co sekundę, nowy subskrybent (lub subskrybent odzyskiwania po awarii) otrzymuje bieżący stan niemal natychmiast po ich początkowej subskrypcji, więc nie ma potrzeby przechowywania tych publikacji.
- Jeśli publikacje są duże, może być konieczne zakończenie dużej ilości miejsca w pamięci masowej w celu zapisania zachowanej publikacji dla każdego tematu. W środowisku z wieloma menedżerami kolejek zachowane publikacje są przechowywane przez wszystkie menedżery kolejek w sieci, które mają zgodną subskrypcję.

Decydując, czy mają być używane zachowane publikacje, należy rozważyć, w jaki sposób aplikacje subskrybujące odzyskują działanie po awarii. Jeśli publikator nie korzysta z zachowanych publikacji, aplikacja subskrybenta może wymagać lokalnego przechowywania bieżącego stanu.

Aby upewnić się, że publikacja została zachowana, należy użyć opcji put-message w tabeli MQPMO_RETAIN. Jeśli ta opcja jest używana i nie można zachować publikacji, komunikat nie zostanie opublikowany, a wywołanie zakończy się niepowodzeniem z opcją MQRC_PUT_NOT_ZACHOWANE.

Jeśli komunikat jest zachowaną publikacją, jest to oznaczane za pomocą właściwości komunikatu MQIsRetained. Utrwalanie komunikatu jest tak samo, jak było ono pierwotnie opublikowane.

Pojęcia pokrewne

Uwagi dotyczące projektowania zachowanych publikacji w klastrach publikowania/subskrypcji

Publikacje w punkcie synchronizacji

W produkcie IBM MQ publish/subscribe punkt synchronizacji może być używany przez publikatorów lub wewnętrznie przez menedżer kolejek.

Publikatory używają punktu synchronizacji, gdy emitują wywołania MQPUT/MQPUT1 z opcją MQPMO_SYNCPOINT. Wszystkie komunikaty dostarczane do subskrybentów są liczone w celu osiągnięcia maksymalnej liczby niezatwierdzonych komunikatów w jednostce work.The Atrybut menedżera kolejek MAXUMSGS określa ten limit. Jeśli limit zostanie osiągnięty, publikator otrzymuje kod przyczyny 2024 (07E8) (RC2024): MQRC_SYNCPOINT_LIMIT_REACHED.

Gdy publikator wysyła wywołania MQPUT/MQPUT1 za pomocą wywołania MQPMO_NO_SYNCPOINT z opcją MQPMO_RETAIN albo z opcjami dostarczania tematu NPMMSGDLV/PMSGDLV z wartościami ALL lub ALLDUR, menedżer kolejek używa wewnętrznych punktów synchronizacji w celu zagwarantowania, że komunikaty są dostarczane w żądany sposób. Publikator może otrzymać kod przyczyny 2024 (07E8) (RC2024): MQRC_SYNCPOINT_LIMIT_REACHED, jeśli limit został osiągnięty w zasięgu wywołania obiektu publikatora MQPUT/MQPUT1.

Subskrybenty i subskrypcje

Subskrybent w produkcie IBM MQ publikowania/subskrypcji jest aplikacją, która żąda informacji na temat konkretnego tematu z menedżera kolejek w sieci publikowania/subskrypcji. Subskrybent może odbierać komunikaty, dotyczące tych samych lub różnych tematów, z więcej niż jednego publikatora.

Subskrypcje można tworzyć ręcznie przy użyciu komendy MQSC lub aplikacji. Subskrypcje te są wysyłane do lokalnego menedżera kolejek i zawierają informacje o publikacjach, które subskrybent chce otrzymywać:

- Temat, w którym jest zainteresowany subskrybentem, może być rozwiązany w wielu tematach, jeśli używane są znaki wieloznaczne.
- Opcjonalny łańcuch wyboru, który ma być stosowany do opublikowanych komunikatów.

- Uchwyt do kolejki (nazywany *kolejką subskrybenta*), na którym powinny zostać umieszczone wybrane publikacje, a także opcjonalny parametr CorrelId.

Lokalny menedżer kolejek przechowuje informacje o subskrypcji, a po odebraniu publikacji skanuje informacje w celu określenia, czy istnieje subskrypcja zgodna z tematem publikacji i tańcuchem wyboru. Dla każdej zgodnej subskrypcji menedżer kolejek kieruje publikację do kolejki subskrybenta subskrybenta. Informacje, które są przechowywane przez menedżera kolejek na temat subskrypcji, można wyświetlić za pomocą komend DIS SUB i DIS SBSTATUS.

Subskrypcja jest usuwana tylko wtedy, gdy wystąpi jedno z następujących zdarzeń:

- Subskrybent nie subskrybuje przy użyciu wywołania MQCLOSE (jeśli subskrypcja została wykonana w sposób inny niż duraliński).
- Subskrypcja utraci ważność.
- Subskrypcja została usunięta przez administratora systemu za pomocą komendy DELETE SUB.
- Aplikacja subskrybenta kończy działanie (jeśli subskrypcja została wykonana w sposób inny niż duraliński).
- Menedżer kolejek został zatrzymany lub zrestartowany (jeśli subskrypcja została wykonana w sposób inny niż durably).

Podczas pobierania komunikatów należy użyć odpowiednich opcji w wywołaniu MQGET. Jeśli aplikacja przetwarza tylko komunikaty dla jednej subskrypcji, co najmniej, należy użyć produktu `get-by-correlid`, co zostało wykazane w przykładowym programie `C amqssbxa.c` oraz w niezarządzanym subskrybencie MQ. Produkt **CorrelId** do użycia jest zwracany z tabeli MQSUB w tabeli **MQSD.SubCorrelId**.

Pojęcia pokrewne

Klonowane i współużytkowane subskrypcje

Odsyłacze pokrewne

Przykłady definiowania właściwości sharedSubscription

Kolejki zarządzane i publikowania/subskrybowanie

Podczas tworzenia subskrypcji można wybrać opcję użycia kolejkowania zarządzanego. Jeśli podczas tworzenia subskrypcji używana jest kolejka subskrypcji zarządzana kolejką subskrypcji, zostanie ona utworzona automatycznie. Kolejki zarządzane są automatycznie uginane zgodnie z trwałością subskrypcji. Korzystanie z kolejek zarządzanych oznacza, że nie trzeba się martwić tworzeniem kolejek, aby otrzymywać publikacje, a wszystkie niewykorzystane publikacje są usuwane z kolejek subskrybenta automatycznie, jeśli połączenie nietrwałe subskrypcji jest zamknięte.

Jeśli aplikacja nie musi używać określonej kolejki jako jej kolejki subskrybenta, miejsce docelowe dla publikacji, które otrzymuje, może korzystać z *subskrypcji zarządzanych* przy użyciu opcji subskrypcji MQSO_MANAGED. Jeśli zostanie utworzona subskrypcja zarządzana, menedżer kolejek zwróci do subskrybenta uchwyt obiektu dla kolejki subskrybenta, która jest tworzona przez menedżer kolejek w miejscu, w którym będą odbierane publikacje. Wynika to z faktu, że *zarządzane subskrypcje* to jedna z nich, w której produkt IBM MQ obsługuje subskrypcję. Zostanie zwrócony uchwyt obiektu kolejki umożliwiający przeglądanie, pobieranie lub sprawdzanie w kolejce (nie jest możliwe ustawienie lub ustawienie atrybutów kolejki zarządzanej, chyba że użytkownik jawnie nadano dostęp do tymczasowych kolejek dynamicznych).

Trwałość subskrypcji określa, czy kolejka zarządzana pozostaje w sytuacji, gdy połączenie aplikacji subskrybującej z menedżerem kolejek jest zerwane.

Subskrypcje zarządzane są szczególnie przydatne, gdy są używane z subskrypcjami nietrwałymi, ponieważ po zakończeniu połączenia aplikacji niewykorzystane komunikaty pozostałyby w kolejce subskrybenta w nieskończoność zajęte przez miejsce w menedżerze kolejek. Jeśli używana jest zarządzana subskrypcja, kolejka zarządzana będzie tymczasową kolejką dynamiczną i jako taka zostanie usunięta wraz z nieużyтыми komunikatami, gdy połączenie zostanie zerwane z jednego z następujących powodów:

- Komenda MQCLOSE z opcją MQCO_REMOVE_SUB jest używana, a zarządzany Hobj jest zamknięty.

- Połączenie zostało utracone do aplikacji przy użyciu nietrwałej subskrypcji (MQSO_NON_DURABLE).
- Subskrypcja została usunięta, ponieważ wygasła, a zarządzany Hobj jest zamknięty.

Zarządzane subskrypcje mogą być również używane z subskrypcjami trwałymi, ale możliwe jest pozostawienie niewykorzystanych komunikatów w kolejce subskrybenta, aby można było je pobrać po ponownym otwarciu połączenia. Z tego powodu kolejki zarządzane dla trwałych subskrypcji przyjmują postać trwałej kolejki dynamicznej i pozostaną w sytuacji, gdy połączenie aplikacji subskrybującej z menedżerem kolejek zostanie przerwane.

Istnieje możliwość ustawienia utraty ważności subskrypcji, jeśli ma być używana trwała dynamiczna kolejka zarządzana, dzięki czemu mimo że kolejka nadal będzie istnieć po zerowaniu połączenia, nie będzie ona nadal istnieć w nieskończoność.

W przypadku usunięcia zarządzanej kolejki zostanie wyświetlony komunikat o błędzie.

Tworzone kolejki zarządzane są nazwane liczbami na końcu (datowniki), dzięki czemu każda z nich jest unikalna.

Trwałość subskrypcji

Subskrypcje mogą być skonfigurowane jako trwałe lub nietrwałe. Trwałość subskrypcji określa, co dzieje się z subskrypcjami, gdy aplikacje subskrybujące rozłączają się z menedżerem kolejek.

Subskrypcje stałe

Trwałe subskrypcje nadal istnieją, gdy połączenie aplikacji subskrybującej z menedżerem kolejek jest zamknięte. Jeśli subskrypcja jest trwała, po odłączeniu aplikacji subskrybującej subskrypcja pozostaje na miejscu i może być używana przez aplikację subskrybującą podczas ponownego nawiązywania połączenia z żądaniem subskrypcji przy użyciu programu **SubName**, który został zwrócony podczas tworzenia subskrypcji.

W przypadku subskrypcji w sposób dualisowy nazwa subskrypcji (**SubName**) jest wymagana. Nazwy subskrypcji muszą być unikalne w obrębie menedżera kolejek, aby można było ich używać do identyfikowania subskrypcji. Ten sposób identyfikacji jest niezbędny podczas określania subskrypcji, która ma zostać wznowiona, jeśli połączenie z subskrypcją zostało celowo zamknięte (przy użyciu opcji MQCO_KEEP_SUB) lub zostało odłączone od menedżera kolejek. Istniejącą subskrypcję można wznowić za pomocą wywołania MQSUB z opcją MQSO_RESUME. Nazwy subskrypcji są również wyświetlane w przypadku użycia komendy DISPLAY SBSTATUS z parametrem SUBTYPE ALL lub ADMIN.

Jeśli aplikacja nie wymaga już trwałej subskrypcji, można ją usunąć za pomocą wywołania funkcji MQCLOSE z opcją MQCO_REMOVE_SUB lub można ją usunąć ręcznie za pomocą komendy MQSC DELETE SUB.

Atrybut tematu **DURSUB** umożliwia określenie, czy można tworzyć trwałe subskrypcje tematu.

Po powrocie z wywołania MQSUB z opcją MQSO_RESUME utrata ważności subskrypcji jest ustawiana na pierwotny termin ważności subskrypcji, a nie na pozostały czas ważności.

Menedżer kolejek kontynuuje wysyłanie publikacji w celu spełnienia subskrypcji trwałej, nawet jeśli aplikacja subskrybenta nie jest połączona. Prowadzi to do budowania komunikatów w kolejce subskrybenta. Najprostszym sposobem uniknięcia tego problemu jest użycie nietrwałej subskrypcji tam, gdzie jest to konieczne. Jeśli jednak konieczne jest użycie trwałych subskrypcji, można uniknąć budowania komunikatów, jeśli subskrybent subskrybuje przy użyciu opcji Zachowane publikacje. Subskrybent może następnie sterować odbieraniem publikacji za pomocą wywołania MQSUBRQ.

Subskrypcje nietrwałe

Subskrypcje nietrwałe istnieją tylko wtedy, gdy połączenie aplikacji subskrybującej z menedżerem kolejek pozostaje otwarte. Subskrypcja zostaje usunięta, gdy aplikacja subskrybująca odłącza się od menedżera kolejek celowo lub ze względu na utratę połączenia. Po zamknięciu połączenia informacje o subskrypcji są usuwane z menedżera kolejek i nie są już wyświetlane, jeśli subskrypcje są wyświetlane za pomocą komendy DISPLAY SBSTATUS. W kolejce subskrybenta nie są umieszczane żadne komunikaty.

Informacje o tym, co dzieje się z niewykorzystywaną publikacjami w kolejce subskrybentów dla subskrypcji nietrwących, są określone w następujący sposób.

- Jeśli aplikacja subskrybująca używa zarządzanego miejsca docelowego, wszystkie publikacje, które nie zostały wykorzystane, są automatycznie usuwane.
- Jeśli aplikacja subskrybująca udostępnia uchwyt do własnej kolejki subskrybenta podczas subskrybowania, niewykorzystane komunikaty nie są automatycznie usuwane. Aplikacja jest odpowiedzialna za usunięcie zawartości kolejki, jeśli jest to właściwe. Jeśli kolejka jest współużytkowana przez więcej niż jeden subskrybent lub inne aplikacje typu punkt z punktem, może nie być właściwe całkowite skasowanie kolejki.

Chociaż nie jest to wymagane w przypadku subskrypcji nietrwących, menedżer kolejek używa nazwy subskrypcji, jeśli została podana. Nazwy subskrypcji muszą być unikalne w obrębie menedżera kolejek, aby można było ich używać do identyfikowania subskrypcji.

Pojęcia pokrewne

Subskrypcje sklonowane i współużytkowane

Zadania pokrewne

Korzystanie ze współużytkowanych subskrypcji JMS 2.0

Odsyłacze pokrewne

Przykłady definiowania właściwości sharedSubscription

Łańcuchy wyboru

Łańcuch wyboru jest wyrażeniem, które jest stosowane do publikacji w celu określenia, czy jest ona zgodna z subskrypcją. Łańcuchy wyboru mogą zawierać znaki wieloznaczne.

Po zasubskrybowaniu, oprócz określania tematu, można określić łańcuch wyboru w celu wybrania publikacji zgodnie z ich właściwościami komunikatu.

Łańcuch wyboru jest wartościowany w odniesieniu do komunikatu umieszczonego przez publikatora przed jego zmodyfikowaniem w celu dostarczenia do każdego subskrybenta. Należy zachować ostrożność przy korzystaniu z pól w łańcuchu wyboru, które mogą być modyfikowane w ramach operacji publikowania. Na przykład pola MQMD UserIdentifier, MsgId CorrelId.

Łańcuchy wyboru nie powinny odwoływać się do żadnego z pól właściwości komunikatu dodanych przez menedżer kolejek jako część operacji publikowania (patrz sekcja Właściwości komunikatu publikowania/subskrypcji), z wyjątkiem właściwości komunikatu MQTopicString, która zawiera łańcuch tematu dla publikacji.

Pojęcia pokrewne

Reguły i ograniczenia dotyczące łańcuchów wyboru

Tematy

Temat jest tematem informacji, które są publikowane w komunikacie publikowania/subskrypcji.

Komunikaty w systemach typu punkt z punktem są wysyłane do konkretnego adresu docelowego. Komunikaty w systemach publikowania/subskrybowania oparte na tematyce są wysyłane do subskrybentów na podstawie tematu, który opisuje treść wiadomości. W systemach opartych na treści komunikaty są wysyłane do subskrybentów w zależności od zawartości samego komunikatu.

System publikowania/subskrypcji produktu IBM MQ jest systemem publikowania/subskrypcji opartym na tematyce. Publikator tworzy komunikat i publikuje go za pomocą łańcucha tematu, który najlepiej pasuje do tematu publikacji. W celu otrzymywania publikacji subskrybent tworzy subskrypcję przy użyciu łańcucha tematu zgodnego z wzorcem w celu wybrania tematów publikacji. Menedżer kolejek dostarcza publikacje do subskrybentów, którzy mają subskrypcje zgodne z tematem publikacji, a także są uprawnieni do otrzymywania publikacji. W artykule “Łańcuchy tematów” na stronie 70 opisano składnię łańcuchów tematów, które identyfikują temat publikacji. Subskrybenci tworzą również łańcuchy tematów w celu wybrania tematów do odebrania. Łańcuchy tematów tworzone przez subskrybentów mogą zawierać jeden z dwóch alternatywnych schematów znaków wieloznacznych w celu dopasowania

wzorca do łańcuchów tematów w publikacjach. Dopasowywanie wzorca jest opisane w sekcji “Schematy znaków wieloznacznych” na stronie 71.

W przypadku publikowania/subskrybowania na podstawie tematu publikatory lub administratorzy są odpowiedzialni za klasyfikowanie tematów do tematów. Zwykle tematy są zorganizowane hierarchicznie, w drzewa tematów, przy użyciu znaku ' / ' w celu utworzenia podtematów w łańcuchu tematu. Przykłady drzew tematów można znaleźć w sekcji “Drzewa tematów” na stronie 77 . Tematy są węzłami w drzewie tematów. Tematy mogą być węzłów-liści bez dalszych podtematów lub węzłów pośrednich z podtematami.

Równoległe z organizowaniem przedmiotów w hierarchicznym drzewie tematów można powiązać tematy z obiektami tematu administracyjnego. Atrybuty można przypisywać do tematu, na przykład tego, czy temat jest dystrybuowany w klastrze, poprzez powiązanie go z obiektem tematu administracyjnego. Powiązanie jest wykonywane przez nazwanie tematu przy użyciu atrybutu TOPICSTR obiektu tematu administracyjnego. Jeśli obiekt tematu administracyjnego nie zostanie jawnie powiązany z tematem, wówczas temat dziedziczy atrybuty jego najbliższego przodka w drzewie tematów, które *mają* powiązane z obiektem tematu administracyjnego. Jeśli w ogóle nie zdefiniowano żadnych tematów nadrzędnych, dziedziczy on z SYSTEM.BASE.TOPIC. Obiekty tematów administracyjnych są opisane w sekcji “Administracyjne obiekty tematów” na stronie 78.

Uwaga: Nawet jeśli odziedziczysz wszystkie atrybuty tematu z SYSTEM.BASE.TOPIC, aby zdefiniować temat główny dla tematów, które bezpośrednio dziedziczą z SYSTEM.BASE.TOPIC. Na przykład w obszarze tematów stanów USA, USA/Alabama USA/Alaskaitd. USA jest tematem głównym. Głównym celem tematu głównego jest utworzenie dyskretnych, niepokrywających się obszarów tematów w celu uniknięcia publikacji zgodnych z niepoprawnymi subskrypcjami. Oznacza to również, że można zmienić atrybuty tematu głównego, aby mieć wpływ na cały obszar tematu. Na przykład można ustawić nazwę atrybutu **CLUSTER** .

W przypadku odniesienia do tematu jako publikatora lub subskrybenta istnieje możliwość wyboru łańcucha tematu lub odwołania się do obiektu tematu. Można również wykonać obie czynności. W takim przypadku łańcuch tematu, który podasz, definiuje podtemat obiektu tematu. Menedżer kolejek identyfikuje temat, dołączając łańcuch tematu do przedrostka łańcucha tematu określonego w obiekcie tematu, wstawiając dodatkowy ' / ' między dwoma łańcuchami tematów, na przykład *łańcuch tematu/ łańcuch obiektu*. “Łączenie łańcuchów tematów” na stronie 76 opisuje to dalej. Wynikowy łańcuch tematu jest używany do identyfikowania tematu i powiązania go z obiektem tematu administracyjnego. Obiekt tematu administracyjnego nie musi być tym samym obiektem tematu, co obiekt tematu odpowiadający tematowi wzorcowi.

W publikowania/subskrybowaniu na podstawie treści można zdefiniować komunikaty, które mają być odbierane, udostępniając łańcuchy wyboru, które wyszukają treść każdego komunikatu. Produkt IBM MQ udostępnia pośrednią formę treści publikowania/subskrypcji opartą na treści przy użyciu selektorów komunikatów, które skanują właściwości komunikatu, a nie pełną treść komunikatu. Patrz sekcja Selektory. Archetypowe użycie selektorów komunikatów polega na zasubskrybowaniu tematu, a następnie zakwalifikowaniu do wyboru przy użyciu właściwości liczbowej. Selektor umożliwia określenie, że użytkownik jest zainteresowany wartościami tylko w określonym zakresie. Coś, czego nie można zrobić za pomocą znaków wieloznacznych opartych na znaku lub w temacie. Jeśli konieczne jest filtrowanie w oparciu o pełną treść komunikatu, należy użyć produktu IBM Integration Bus.

Łańcuchy tematów

Informacje o etykietach publikowane jako temat przy użyciu łańcucha tematu. Zasubskrybuj grupy tematów za pomocą łańcuchów tematu ze znakami wieloznacznymi na podstawie znaku lub tematu.

Tematy

Łańcuch tematu jest łańcuchem znaków, który identyfikuje temat komunikatu publikowania/subskrypcji. Podczas tworzenia łańcucha tematu można używać dowolnych znaków, które są podobne do tych, które są używane.

Łańcuch tematu



Trzy znaki mają specjalne znaczenie w publikacji/subskrypcji produktu IBM WebSphere MQ 7. Są one dozwolone w dowolnym miejscu łańcucha tematu, ale należy używać ich z zachowaniem ostrożności. Korzystanie ze znaków specjalnych jest wyjaśnione w [“Schemat wieloznaczny oparty na temacie”](#) na stronie 72.

Ukośnik (/)

Separator poziomu tematu. Aby utworzyć strukturę tematu w drzewie tematów, należy użyć znaku '/'.

Należy unikać pustych poziomów tematów, '//'(jeśli można). Odpowiadają one węzłom w hierarchii tematów bez łańcucha tematu. Początkowe lub końcowe '/' w łańcuchu tematu odpowiada wiodącym lub kończącym pustym węzłom i należy go unikać.

Znak krzyżyka (#)

Używany w połączeniu z produktem '/' do konstruowania wielopoziomowego znaku wieloznacznego w subskrypcjach. Należy zachować ostrożność przy użyciu produktu '#' przylegającego do produktu '/' w łańcuchach tematów używanych do tworzenia nazw publikowanych tematów. W programie [“Przykłady łańcuchów tematów”](#) na stronie 71 jest wyświetlane sensowne użycie produktu '#'.

Łańcuchy '.../#/...', '#/...' i '.../#' mają specjalne znaczenie w łańcuchach tematów subskrypcji. Łańcuchy są zgodne z wszystkimi tematami na co najmniej jednym poziomie w hierarchii tematów. Dlatego też, jeśli utworzono temat z jedną z tych sekwencji, nie można go zasubskrybować, bez subskrybowania wszystkich tematów na wielu poziomach w hierarchii tematów.

Znak plus (+)

Używany w połączeniu z produktem '/' do konstruowania jednopoziomowego znaku wieloznacznego w subskrypcjach. Należy zachować ostrożność przy użyciu produktu '+' przylegającego do produktu '/' w łańcuchach tematów używanych do tworzenia nazw publikowanych tematów.

Łańcuchy '.../+/...', '+/...' i '.../+' mają specjalne znaczenie w łańcuchach tematów subskrypcji. Łańcuchy są zgodne z wszystkimi tematami na jednym poziomie w hierarchii tematów. Dlatego też, jeśli utworzono temat z jedną z tych sekwencji, nie można go zasubskrybować, bez subskrybowania wszystkich tematów na jednym poziomie w hierarchii tematów.

Przykłady łańcuchów tematów

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

Odsyłacze pokrewne

[TOPIC](#)

Schematy znaków wieloznacznych

W celu zasubskrybowania wielu tematów używane są dwa schematy znaków wieloznacznych. Wybór programu to opcja subskrypcji.

MQSO_WILDCARD_TOPIC

Wybierz tematy, które mają być subskrybowane przy użyciu schematu zastępczego opartego na temacie.

Jest to wartość domyślna, jeśli nie wybrano jawnie żadnego schematu ze znakami wieloznacznymi.

MQSO_WILDCARD_CHAR

Wybierz tematy, aby zasubskrybować korzystanie z systemu znaków wieloznacznych opartego na znakach.

Ustaw albo schemat, podając parametr **wschema** w komendzie DEFINE SUB. Więcej informacji na ten temat zawiera sekcja [DEFINE SUB](#).

Uwaga: Subskrypcje utworzone przed produktem IBM WebSphere MQ 7.0 zawsze korzystają z systemu znaków wieloznacznych opartego na znakach.

Przykłady

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

Schemat wieloznaczny oparty na temacie

Znaki wieloznaczne oparte na temacie umożliwiają subskrybentom subskrybowanie więcej niż jednego tematu w danym momencie.

Znaki wieloznaczne oparte na temacie są potężną funkcją systemu tematu w programie IBM MQ publikowania/subskrypcji. Wielopoziomowe i jednopoziomowe znaki wieloznaczne mogą być używane w przypadku subskrypcji, ale nie mogą być używane w temacie przez publikator komunikatu.

Schemat wieloznaczny oparty na temacie umożliwia wybór publikacji pogrupowanych według poziomu tematu. Można wybrać, dla każdego poziomu w hierarchii tematów, czy łańcuch w subskrypcji dla tego poziomu tematu musi być dokładnie zgodny z łańcuchem publikacji, czy też nie. Na przykład subskrypcja IBM+/Results wybiera wszystkie tematy,

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

Istnieją dwa typy znaków wieloznacznych.

Wieloznaczny znak wieloznaczny

- Wielopoziomowy znak wieloznaczny jest używany w subskrypcjach. W przypadku użycia w publikacji jest ona traktowana jako literał.
- Wielopoziomowy znak wieloznaczny '# ' jest używany do dopasowania dowolnej liczby poziomów w obrębie tematu. Na przykład przy użyciu przykładowego drzewa tematów, jeśli użytkownik subskrybuje produkt 'USA/Alaska/# ', otrzymujesz komunikaty dotyczące tematów 'USA/Alaska ' i 'USA/Alaska/Juneau '.
- Wielopoziomowy znak wieloznaczny może reprezentować zero lub więcej poziomów. Oznacza to, że 'USA/# ' może być również zgodny z pojedynczą wartością 'USA ', gdzie '# ' oznacza zero poziomów. Separator poziomu tematu jest w tym kontekście pozbawiony znaczenia, ponieważ nie ma poziomów do rozdzielenia.
- Wielopoziomowy znak wieloznaczny jest skuteczny tylko wtedy, gdy jest określony w jego własnym lub obok znaku separatora poziomu tematu. Dlatego też '# ' i 'USA/# ' są poprawnymi tematami, w których znak '# ' jest traktowany jako znak wieloznaczny. Jednak mimo że 'USA/# ' jest również poprawnym łańcuchem tematu, znak '# ' nie jest traktowany jako znak wieloznaczny i nie ma żadnego specjalnego znaczenia. Więcej informacji można znaleźć w sekcji [“Jeśli znaki wieloznaczne oparte na temacie nie są dzikie”](#) na stronie 74.

Wieloznaczny znak wieloznaczny

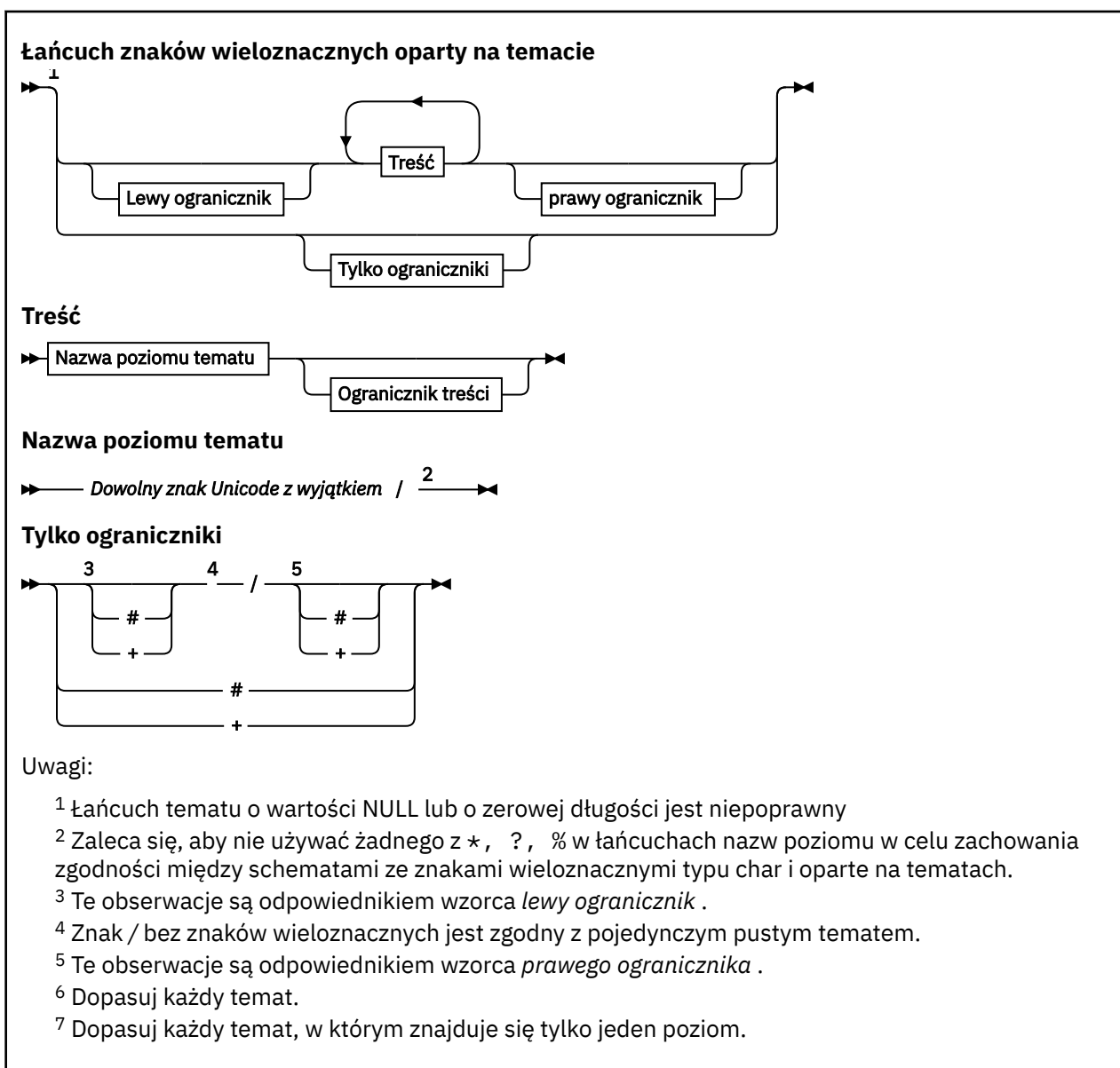
- Pojedynczy znak wieloznaczny jest używany w subskrypcjach. W przypadku użycia w publikacji jest ona traktowana jako literał.
- Jednopoziomowy znak wieloznaczny '+ ' jest zgodny z jednym, a tylko jednym, poziomem tematu. Na przykład 'USA/+ ' jest zgodne z 'USA/Alabama ', ale nie z 'USA/Alabama/Auburn '.

Ponieważ jednopoziomowy znak wieloznaczny jest zgodny tylko z jednym poziomem, 'USA/+' nie jest zgodny z 'USA'.

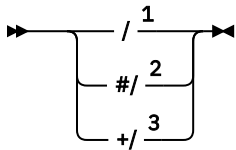
- Jednopoziomowy znak wieloznaczny może być używany na dowolnym poziomie w drzewie tematów, a także w połączeniu z wielopoziomowym znakiem wieloznacznym. Jednopoziomowy znak wieloznaczny musi być określony obok separatora poziomu tematu, z wyjątkiem sytuacji, gdy jest on określony jako własny. Dlatego też '+' i 'USA/+' są poprawnymi tematami, w których znak '+' jest traktowany jako znak wieloznaczny. Jednak mimo że 'USA+' jest również poprawnym łańcuchem tematu, znak '+' nie jest traktowany jako znak wieloznaczny i nie ma żadnego specjalnego znaczenia. Więcej informacji można znaleźć w sekcji [“Jeśli znaki wieloznaczne oparte na temacie nie są dzikie”](#) na stronie 74.

Składnia dla schematu zastępczego opartego na temacie nie zawiera znaków zmiany znaczenia. Czy produkty '#' i '+' są traktowane jako znaki wieloznaczne, czy też nie są zależne od ich kontekstu. Więcej informacji można znaleźć w sekcji [“Jeśli znaki wieloznaczne oparte na temacie nie są dzikie”](#) na stronie 74.

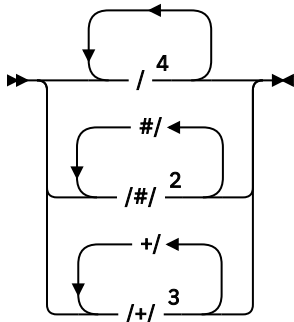
Uwaga: Początek i koniec łańcucha tematu jest traktowany w specjalny sposób. Korzystając z programu '\$' w celu oznaczenia końca łańcucha, '\$#/...' jest wielopoziomowym znakiem wieloznacznym, a '\$/#/...'. Jest pustym węzłem w katalogu głównym, po którym następuje wielopoziomowy znak wieloznaczny.



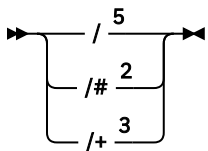
Lewy ogranicznik



Ogranicznik treści



Prawy ogranicznik



Uwagi:

- 1 Łańcuch tematu rozpoczyna się od pustego tematu.
- 2 Pasuje do zera lub większej liczby poziomów. Wiele łańcuchów zgodności wielopoziomowych ma ten sam efekt, co jeden wielopoziomowy łańcuch zgodności.
- 3 Pasuje do dokładnie jednego poziomu.
- 4 // to pusty temat-obiekt tematu bez łańcucha tematu.
- 5 Łańcuch tematu kończy się pustym tematem.

Jeśli znaki wieloznaczne oparte na temacie nie są dzikie

Znaki wieloznaczne '+' i '#/' nie mają specjalnego znaczenia, gdy są one mieszane z innymi znakami (w tym samymi znakami) na poziomie tematu.

Oznacza to, że tematy, które zawierają elementy '+' lub '#/' wraz z innymi znakami na poziomie tematu, mogą być publikowane.

Rozważmy na przykład następujące dwa tematy:

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

W pierwszym przykładzie znaki '+' i '#/' są traktowane jako znaki wieloznaczne i dlatego nie są poprawne w łańcuchu tematu, który ma zostać opublikowany, ale jest poprawny w subskrypcji.

W drugim przykładzie znaki '+' i '#/' nie są traktowane jako znaki wieloznaczne i dlatego łańcuch tematu może być zarówno opublikowany, jak i subskrybowany.

Przykłady

IBM/+/Results

Schemat znaków wieloznacznych oparty na znakach

Schemat znaków wieloznacznych oparty na znakach pozwala wybrać tematy oparte na tradycyjnym dopasowaniu znaków.

Za pomocą łańcucha '*' można wybrać wszystkie tematy na wielu poziomach w hierarchii tematów. Używanie produktu '*' w systemie znaków wieloznacznych znakowych jest równoznaczne z użyciem łańcucha znaków wieloznacznych opartego na temacie '#'

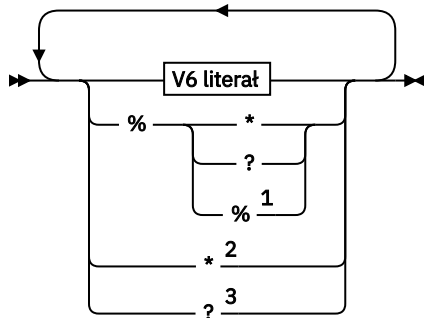
'x*/y' jest odpowiednikiem 'x#/y' w schemacie opartym na temacie i wybiera wszystkie tematy w hierarchii tematów między poziomami 'x' i 'y', gdzie 'x' i 'y' są nazwami tematów, które nie znajdują się w zestawie poziomów zwracanych przez znak wieloznaczny.

'/+/ ' w systemie opartym na temacie nie ma dokładnych odpowiedników w schemacie znakowym. Program 'IBM*/Results' również wybierze opcję 'IBM/Patents/Software/Results'. Tylko wtedy, gdy zbiór nazw tematów na każdym poziomie hierarchii jest unikalny, można zawsze konstruować zapytania z dwoma schematami, które ustępują identycznemu dopasowaniu.

W schemacie opartym na temacie nie ma odpowiedników używanych w ogólnym systemie '*' i '?' w schemacie znakowym. Schemat oparty na temacie nie wykonuje częściowego dopasowania przy użyciu znaków wieloznacznych. Subskrypcja znaków wieloznacznych na podstawie znaku 'IBM/*ware/Results' nie ma odpowiednika opartego na temacie.

Uwaga: Dopasowania z użyciem znaków wieloznacznych znakowych są wolniejsze niż dopasowania przy użyciu subskrypcji opartych na tematach.

Łańcuch znaków wieloznacznych oparty na znakach



V6 literat

►► Dowlony znak Unicode z wyjątkiem *,? i% ◄◄

Uwagi:

- ¹ Oznacza, że znak zmiany znaczenia ma następujący znak, tak aby był traktowany jako literat. Po '%' musi następować następujące wartości: '*', '?' lub '%'. Więcej informacji na ten temat zawiera sekcja "Przykłady łańcuchów tematów" na stronie 71.
- ² Oznacza zgodność z zerem lub większą liczbę znaków w subskrypcji.
- ³ Oznacza zgodność dokładnie jednego znaku w subskrypcji.

Przykłady

IBM*/Results
IBM/*ware/Results

Łączenie łańcuchów tematów

Tworząc subskrypcje lub otwierając tematy, aby można było publikować do nich komunikaty, łańcuch tematu może zostać utworzony przez połączenie dwóch oddzielnych łańcuchów tematów podrzędnych lub "podtematów". Jeden podtemat jest udostępniany przez aplikację lub komendę administracyjną jako łańcuch tematu, a drugi to łańcuch tematu powiązany z obiektem tematu. Można użyć dowolnego podtematu jako łańcucha tematu lub połączyć je w celu utworzenia nowej nazwy tematu.

Na przykład przy definiowaniu subskrypcji za pomocą komendy MQSC **DEFINE SUB** komenda może przyjmować wartość **TOPICSTR** (łańcuch tematu) lub **TOPICOBJ** (obiekt tematu) jako atrybut lub obie te wartości razem. Jeśli podano tylko wartość **TOPICOBJ**, łańcuch tematu powiązany z tym obiektem tematu jest używany jako łańcuch tematu. Jeśli podano tylko wartość **TOPICSTR**, jest ona używana jako łańcuch tematu. Jeśli oba elementy są udostępnione, są one konkatenowane w celu utworzenia pojedynczego łańcucha tematu w postaci **TOPICOBJ / TOPICSTR**, gdzie skonfigurowany łańcuch tematu **TOPICOBJ** jest zawsze pierwszy, a dwie części łańcucha są zawsze oddzielone znakiem "/" .

Podobnie w programie MQI pełna nazwa tematu jest tworzona przez program MQOPEN. Składa się on z dwóch pól używanych w wywołaniach MQI publikowania/subskrybowania, w podanej kolejności:

1. Atrybut **TOPICSTR** obiektu tematu o nazwie podanej w polu **ObjectName** .
2. Parametr **ObjectString** definiujący podtemat udostępniany przez aplikację.

Wynikowy łańcuch tematu jest zwracany w parametrze **ResObjectString** .

Te pola są uważane za obecne, jeśli pierwszy znak każdego pola nie jest znakiem pustym lub pustym, a długość pola jest większa od zera. Jeśli istnieje tylko jedno z tych pól, jest ono używane bez zmian jako nazwa tematu. Jeśli żadna z tych pól nie ma wartości, wywołanie kończy się niepowodzeniem z kodem przyczyny MQRC_UNKNOWN_OBJECT_NAME lub MQRC_TOPIC_STRING_ERROR , jeśli pełna nazwa tematu nie jest poprawna.

Jeśli oba pola są obecne, między dwoma elementami wynikowej połączonej nazwy tematu wstawiany jest znak "/" .

Tabela 2 na stronie 76 przedstawia przykłady konkatenacji łańcuchów tematów:

| TOPICSTR obiektu tematu | Łańcuch tematu udostępniony przez aplikację lub komendę DEFINE SUB | Pełna nazwa tematu | Komentarz |
|-------------------------|--|----------------------|--|
| Piłka nożna/Szkocja | ' ' | Piłka nożna/Szkocja | Parametr TOPICSTR obiektu tematu jest używany samodzielnie. |
| ' ' | Piłka nożna/Szkocja | Piłka nożna/Szkocja | Parametr ObjectString/ TOPICSTR jest używany samodzielnie. |
| Piłka nożna | Oceny | Piłka nożna/Szkocja | Znak "/" jest dodawany w punkcie konkatenacji. |
| Piłka nożna | /Szkocja | Piłka nożna//Szkocja | Między dwoma łańcuchami jest tworzony 'pusty węzeł'. Różni się to od wartości "Football/Scores". |
| Piłka nożna | Oceny | /Football/Scores | Temat rozpoczyna się od 'pustego węzła'. Różni się to od wartości "Football/Scores". |

Znak "/" jest traktowany jako znak specjalny, co zapewnia strukturę pełną nazwę tematu w produkcie "Drzewa tematów" na stronie 77. Znak "/" nie może być używany z jakiegokolwiek innego powodu, ponieważ ma wpływ na strukturę drzewa tematów. Temat "/Football" nie jest taki sam, jak temat "Football".

Uwaga: Jeśli podczas tworzenia subskrypcji używany jest obiekt tematu, wartość łańcucha tematu obiektu tematu jest ustalana w ramach subskrypcji w czasie definiowania. Każda kolejna zmiana obiektu tematu nie ma wpływu na łańcuch tematu, do którego została zdefiniowana subskrypcja.

Znaki wieloznaczne w łańcuchach tematów

Następujące znaki wieloznaczne są znakami specjalnymi:

- znak plus (+)
- znak liczby (#)
- gwiazdka (*)
- znak zapytania (?)

Znaki wieloznaczne mają specjalne znaczenie tylko wtedy, gdy są używane w subskrypcji. Te znaki nie są uznawane za niepoprawne, jeśli są używane w innym miejscu, jednak należy się upewnić, że użytkownik wie, jak są one używane, a użytkownik może nie używać tych znaków w łańcuchach tematów podczas publikowania lub definiowania obiektów tematów.

Jeśli użytkownik opublikuje w łańcuchu tematu z produktem # lub + mieszanym z innymi znakami (w tym ze sobą) na poziomie tematu, łańcuch tematu może zostać zasubskrybowany przy użyciu dowolnego schematu znaków wieloznacznych.

Jeśli publikujesz w łańcuchu tematu z # lub + jako jedyny znak między dwoma znakami /, to łańcuch tematu nie może zostać zasubskrybowany jawnie przez aplikację, używając schematu wieloznacznego MQSO_WILDCARD_TOPIC. Ta sytuacja powoduje, że aplikacja pobierze więcej publikacji niż oczekiwano.

W łańcuchu tematu zdefiniowanego obiektu tematu nie należy używać znaku wieloznacznego. W takim przypadku znak jest traktowany jako literał, gdy obiekt jest używany przez publikator oraz jako znak wieloznaczny w przypadku użycia przez subskrypcję. Może to prowadzić do zamieszania.

Przykładowy fragment kodu

Ten fragment kodu, wyodrębniony z przykładowego programu [Przykład 2: publikator do tematu zmiennej](#), łączy obiekt tematu z łańcuchem tematu o zmiennej temat:

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

Drzewa tematów

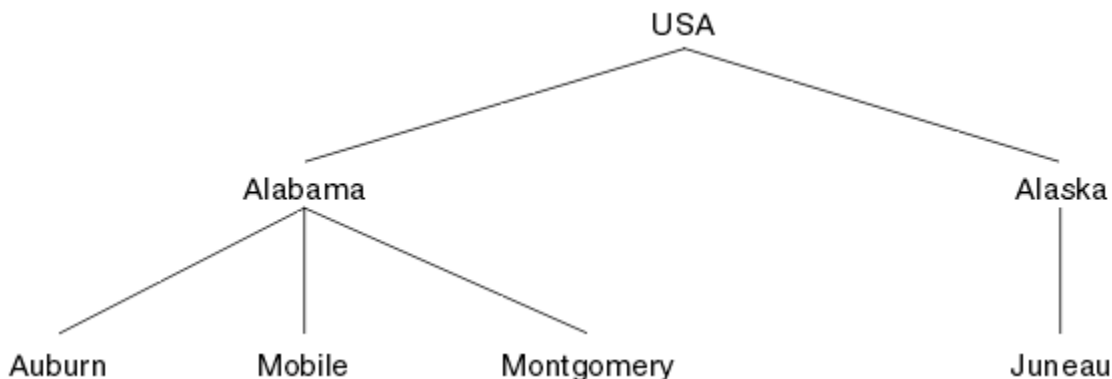
Każdy definiowany temat stanowi element lub węzeł drzewa tematów. Drzewo tematów może być puste, aby rozpocząć od lub zawierać tematy, które zostały wcześniej zdefiniowane przy użyciu komend MQSC lub PCF. Nowy temat można zdefiniować, korzystając z komend tworzenia tematów lub podając temat po raz pierwszy w publikacji lub subskrypcji.

Chociaż w celu zdefiniowania łańcucha tematu tematu można użyć dowolnego łańcucha znaków, zaleca się wybranie łańcucha tematu, który pasuje do hierarchicznej struktury drzewa. Przemyślana konstrukcja stings tematu i drzew tematów może pomóc w następujących operacjach:

- Subskrybowanie wielu tematów.
- Ustanawianie strategii bezpieczeństwa.

Chociaż drzewo tematów może być konstruowane jako płaska, liniowa struktura, lepiej jest zbudować drzewo tematów w strukturze hierarchicznej z jednym lub większą liczbą tematów głównych. Więcej informacji na temat planowania i tematów dotyczących bezpieczeństwa zawiera sekcja [Zabezpieczenia publikowania/subskrypcji](#).

Rysunek 18 na stronie 78 przedstawia przykład drzewa tematów z jednym głównym tematem.



Rysunek 18. Przykład drzewa tematów

Każdy łańcuch znaków na rysunku reprezentuje węzeł w drzewie tematów. Pełny łańcuch tematu jest tworzony przez agregowanie węzłów z jednego lub większej liczby poziomów w drzewie tematów. Poziomy są rozdzielane znakiem "/". Format w pełni określonego łańcucha tematu to: "root/level2/level3".

Poprawne tematy w drzewie tematów przedstawione w sekcji [Rysunek 18 na stronie 78](#) są następujące:

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

Podczas projektowania łańcuchów tematów i drzew tematów należy pamiętać, że menedżer kolejek nie interpretuje ani nie próbuje wyprowadzić znaczenia z samego łańcucha tematu. Po prostu korzysta z łańcucha tematu w celu wysyłania wybranych komunikatów do subskrybentów tego tematu.

Do konstrukcji i treści drzewa tematów mają zastosowanie następujące zasady:

- Liczba poziomów w drzewie tematów nie jest ograniczona.
- Długość nazwy poziomu w drzewie tematów nie jest ograniczona.
- Może istnieć dowolna liczba węzłów "root", to znaczy, że może istnieć dowolna liczba drzew tematów.

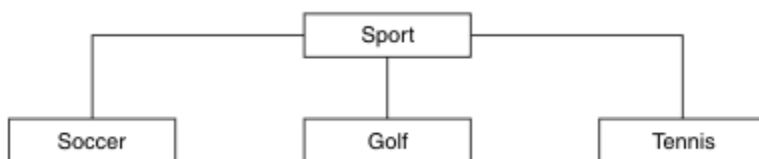
Zadania pokrewne

[Zmniejszanie liczby niechcianych tematów w drzewie tematów](#)

Administracyjne obiekty tematów

Korzystając z obiektu tematu administracyjnego, można przypisać konkretne, inne niż domyślne atrybuty do tematów.

Rysunek 19 na stronie 79 pokazuje, w jaki sposób ogólny temat produktu Sport podzielony na osobne tematy obejmujące różne obiekty sportowe może być wizualizowany jako drzewo tematów:



Rysunek 19. Wizualizacja drzewa tematów

Rysunek 20 na stronie 79 pokazuje, w jaki sposób drzewo tematów może być dalej podzielone, aby oddzielić różne typy informacji o każdym sporcie:



Rysunek 20. Rozszerzone drzewo tematów

Aby utworzyć ilustrowany drzewo tematów, nie trzeba definiować żadnych administracyjnych obiektów tematów. Każdy z węzłów w tym drzewie jest definiowany przez łańcuch tematu utworzony w operacji publikowania lub subskrypcji. Każdy temat w drzewie dziedziczy jego atrybuty od elementu nadrzędnego. Atrybuty są dziedziczone z nadrzędnego obiektu tematu, ponieważ domyślnie wszystkie atrybuty są ustawione na wartość ASPARENT. W tym przykładzie każdy temat ma takie same atrybuty, jak w temacie Sport. Temat Sport nie ma obiektu tematu administracyjnego i dziedziczy jego atrybuty z `SYSTEM.BASE.TOPIC`.

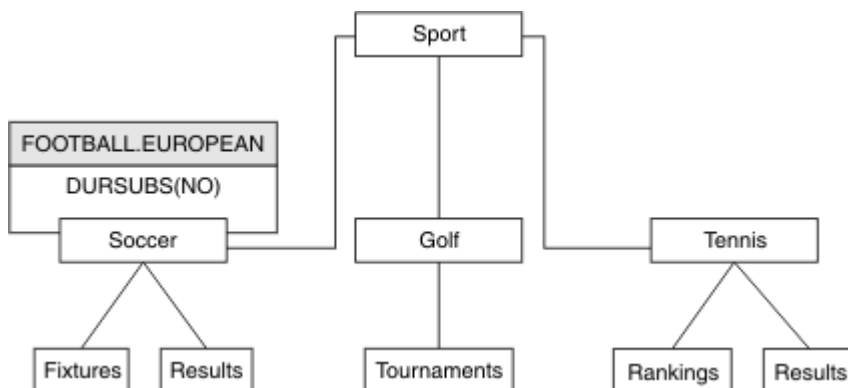
Należy zauważyć, że nie jest dobrą praktyką, aby nadać uprawnienia użytkownikom innym niż mqm w węzle głównym drzewa tematów, który jest `SYSTEM.BASE.TOPIC`, ponieważ uprawnienia są dziedziczone, ale nie mogą być ograniczone. W związku z tym, dając władzom na tym poziomie, dajecie władze całemu drzewowi. Użytkownik powinien nadać uprawnienie na niższym poziomie tematu w hierarchii.

Obiekty tematów administracyjnych mogą być używane do definiowania konkretnych atrybutów dla konkretnych węzłów w drzewie tematów. W poniższym przykładzie definiowany jest obiekt tematu administracyjnego w celu ustawienia właściwości trwałej subskrypcji `DURSUB` tematu piłki nożnej na wartość `NO`:

```

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

Drzewo tematów może teraz być wizualizowane jako:



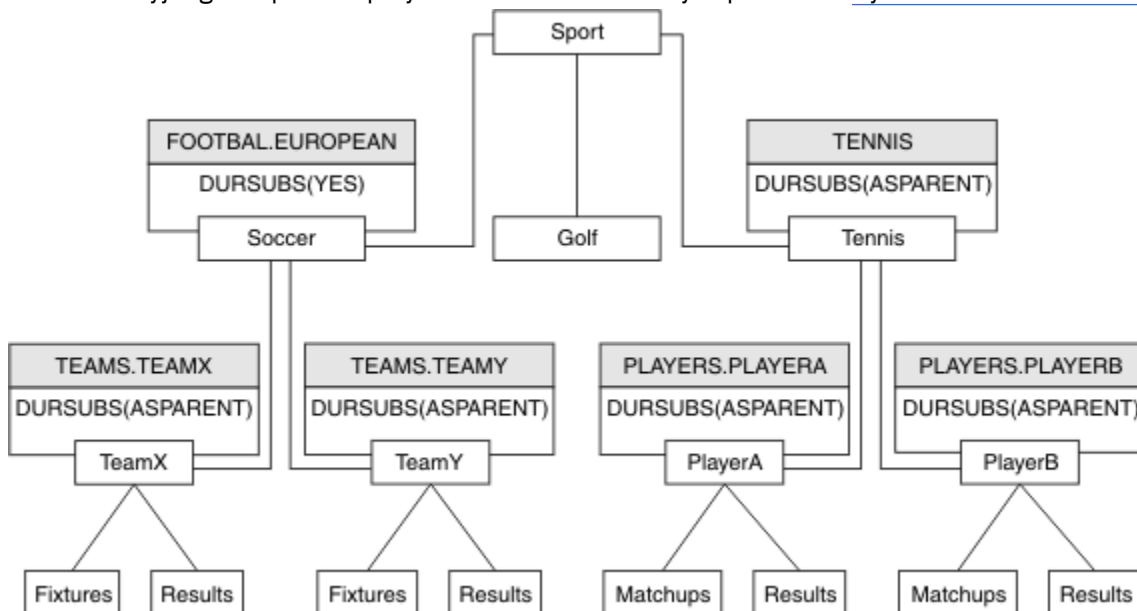
Rysunek 21. Wizualizacja obiektu tematu administracyjnego powiązanego z tematem Sport/Soccer

Wszystkie aplikacje subskrybujące tematy poniżej Soccer w drzewie mogą nadal korzystać z łańcuchów tematów, które były używane przed dodaniem obiektu tematu administracyjnego. Jednak aplikacja może teraz zostać zapisana w celu zasubskrybowania przy użyciu nazwy obiektu FOOTBALL . EUROPEAN, a nie łańcucha /Sport/Soccer. For example, to subscribe to /Sport/Soccer/Results, an application can specify MQSD.ObjectName as FOOTBALL . EUROPEAN and MQSD.ObjectString as Results.

Za pomocą tej funkcji można ukryć część drzewa tematów od twórców aplikacji. Zdefiniuj obiekt tematu administracyjnego w określonym węźle w drzewie tematów, a następnie programiści aplikacji mogą definiować własne tematy jako elementy podrzędne węzła. Programiści muszą wiedzieć o temacie nadrzędnym, ale nie o żadnych innych węzłach w drzewie nadrzędnym.

Dziedziczenie atrybutów

Jeśli drzewo tematów zawiera wiele obiektów tematu administracyjnego, każdy obiekt tematu administracyjnego domyślnie dziedziczy jego atrybuty z najbliższego nadrzędnego tematu administracyjnego. Poprzedni przykład został rozszerzony w produkcie Rysunek 22 na stronie 80:



Rysunek 22. Drzewo tematów z kilkoma obiektami tematu administracyjnego

W tym przykładzie należy użyć dziedziczenia w celu nadania wszystkim tematów podrzędnych produktu /Sport/Soccer właściwości, które subskrypcje nie są trwałe. Zmień atrybut DURSUB produktu FOOTBALL . EUROPEAN na wartość NO.

Ten atrybut można ustawić za pomocą następującej komendy:

ALTER TOPIC(FOOTBALL.EUROPEAN) DURSUB(NO)

Wszystkie obiekty tematów administracyjnych dotyczące tematów podrzędnych produktu Sport/Soccer mają właściwość DURSUB ustawioną na wartość domyślną ASPARENT. Po zmianie wartości właściwości DURSUB produktu FOOTBALL.EUROPEAN na NOtematy podrzędne produktu Sport/Soccer dziedziczą wartość właściwości DURSUB NO. Wszystkie tematy podrzędne produktu Sport/Tennis dziedziczą wartość DURSUB z obiektu SYSTEM.BASE.TOPIC.SYSTEM.BASE.TOPIC ma wartość YES.

Próba wykonania trwałej subskrypcji tematu Sport/Soccer/TeamX/Results nie powiedzie się, jednak próba wykonania trwałej subskrypcji produktu Sport/Tennis/PlayerB/Results zakończy się powodzeniem.

Sterowanie użyciem znaków wieloznacznych przy użyciu właściwości WILDCARD

Użyj właściwości MQSC **Topic** WILDCARD lub równoważnej właściwości PCF Topic WildcardOperation, aby sterować dostarczaniem publikacji do aplikacji subskrybentów, które używają znaków wieloznacznych w nazwach łańcuchów tematów. Właściwość WILDCARD może mieć jedną z dwóch możliwych wartości:

WILDCARD

Zachowanie subskrypcji ze znakami wieloznacznymi w odniesieniu do tego tematu.

PASSTHRU

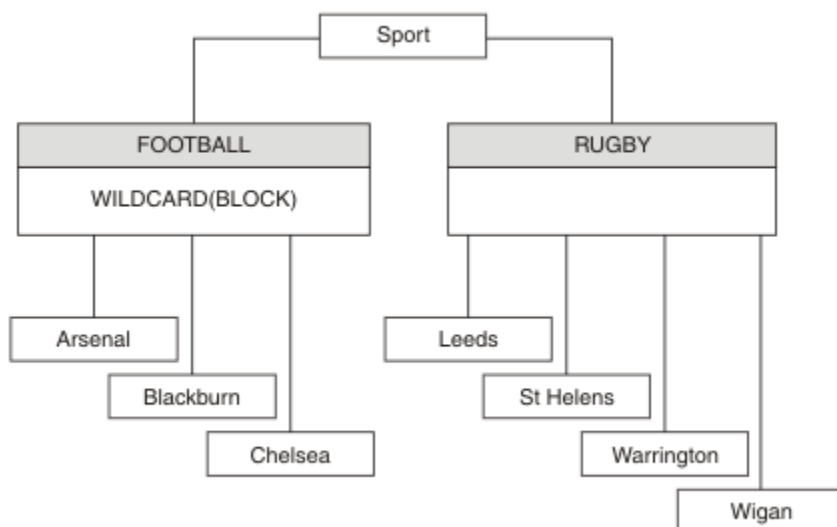
Subskrypcje tematu ze znakami wieloznacznymi, który jest mniej konkretny niż łańcuch tematu w tym obiekcie tematu, otrzymują publikacje zamieszczone w tym temacie i w łańcuchach tematów bardziej konkretnych niż ten temat.

BLOCK

Subskrypcje tematu ze znakami wieloznacznymi, który jest mniej konkretny niż łańcuch tematu w tym obiekcie tematu, nie otrzymują publikacji zamieszczonych w tym temacie i w łańcuchach tematów bardziej konkretnych niż ten temat.

Wartość tego atrybutu jest używana podczas definiowania subskrypcji. Jeśli ten atrybut zostanie zmieniony, modyfikacja nie będzie mieć wpływu na zestaw tematów objętych istniejącymi subskrypcjami. Ten scenariusz dotyczy również sytuacji, w której topologia jest zmieniana podczas tworzenia lub usuwania obiektów tematu. Zestaw tematów zgodnych z subskrypcjami utworzonymi po zmodyfikowaniu atrybutu WILDCARD jest tworzony przy użyciu zmodyfikowanej topologii. Aby wymusić ponowną ocenę zgodnego zestawu tematów pod kątem istniejących subskrypcji, należy zrestartować menedżer kolejek.

W tym przykładzie [“Przykład: tworzenie klastra publikowania/subskrypcji produktu Sport”](#) na stronie 85można wykonać kroki, aby utworzyć strukturę drzewa tematów, która jest wyświetlana w produkcie [Rysunek 23](#) na stronie 82.



Rysunek 23. Drzewo tematów, które korzysta z właściwości WILDCARD , BLOCK

Subskrybent korzystający z łańcucha tematu ze znakami wieloznacznymi # otrzymuje wszystkie publikacje do tematu Sport i poddrzewa Sport/Rugby . Subskrybent nie otrzymuje żadnych publikacji do poddrzewa Sport/Football , ponieważ wartość właściwości WILDCARD w temacie Sport / Football to BLOCK.

PASSTHRU jest ustawieniem domyślnym. Wartość właściwości WILDCARD PASSTHRU można ustawić na węzły w drzewie Sport . Jeśli w węzłach nie ma wartości właściwości WILDCARD BLOCK, ustawienie PASSTHRU nie powoduje zmiany zachowania obserwowanego przez subskrybentów węzłów w drzewie Sports .

W tym przykładzie utwórz subskrypcje, aby sprawdzić, w jaki sposób ustawienie znaku wieloznacznego wpływa na dostarczane publikacje; patrz Rysunek 27 na stronie 87. Aby utworzyć niektóre publikacje, należy uruchomić komendę publikowania w programie Rysunek 30 na stronie 88 .

pub QMA

Rysunek 24. Publikuj w QMA

Wyniki są wyświetlane w programie Tabela 3 na stronie 82. Należy zwrócić uwagę na to, w jaki sposób ustawienie wartości właściwości WILDCARD BLOCK uniemożliwia subskrypcje z użyciem znaków wieloznacznych podczas odbierania publikacji do tematów w zasięgu znaku wieloznacznego.

| Tabela 3. Publikacje otrzymane w systemie QMA | | | |
|---|------------------|--|---|
| Subskrypcja | Łańcuch tematu | Otrzymane publikacje | Uwagi |
| SPORTS | Sports/# | Sports Sports/Rugby Sports/Rugby/Leeds | Wszystkie publikacje dotyczące poddrzewa piłkarskiego zablokowane przez produkt WILDCARD (BLOCK) w systemie Sports/Football |
| SARSENAL | Sports/#/Arsenal | - | Produkt WILDCARD (BLOCK) w systemie Sports/Football uniemożliwia subskrypcję zastępcze w systemie Arsenal |

| Tabela 3. Publikacje otrzymane w systemie QMA (kontynuacja) | | | |
|---|----------------|----------------------|--|
| Subskrypcja | Łańcuch tematu | Otrzymane publikacje | Uwagi |
| SLEEDS | Sports/#/Leeds | Sports/Rugby/Leeds | Wartość domyślna WILDCARD w programie Sports/Rugby nie uniemożliwia subskrypcji ze znakami wieloznacznymi w produkcie Leeds. |

Uwaga:

Załóżmy, że subskrypcja ma znak wieloznaczny, który jest zgodny z obiektem tematu z wartością właściwości WILDCARD BLOCK. Jeśli subskrypcja ma również łańcuch tematu z prawej strony znaku wieloznacznego zgodnego ze zgodnym znakiem wieloznacznym, subskrypcja nigdy nie otrzyma publikacji. Zestaw publikacji, które nie są blokowane, to publikacje na tematy, które są elementami nadrzędnymi zablokowanego znaku wieloznacznego. Publikacje na tematy, które są elementami podrzędnymi tematu z wartością właściwości BLOCK, są blokowane przez znak wieloznaczny. Dlatego łańcuchy tematów subskrypcji, które zawierają temat po prawej stronie znaku wieloznacznego, nigdy nie otrzymują żadnych publikacji do dopasowania.

Ustawienie wartości właściwości WILDCARD na wartość BLOCK nie oznacza, że nie można zasubskrybować łańcucha tematu zawierającego znaki wieloznaczne. Taka subskrypcja jest normalna. Subskrypcja zawiera jawny temat, który jest zgodny z tematem z obiektem tematu, który ma wartość właściwości WILDCARD BLOCK. W przypadku tematów, które są elementami nadrzędnymi lub elementami podrzędnymi tematu z wartością właściwości WILDCARD BLOCK, używane są znaki wieloznaczne. W przykładzie w produkcie [Rysunek 23](#) na stronie 82 subskrypcja, taka jak Sports/Football/#, może odbierać publikacje.

Znaki wieloznaczne i tematy dotyczące klastrów

Definicje tematów klastra są propagowane do każdego menedżera kolejek w klastrze. Subskrypcja tematu klastra w jednym menedżerze kolejek w wyniku klastra powoduje utworzenie subskrypcji proxy w menedżerze kolejek. Subskrypcja proxy jest tworzona w każdym innym menedżerze kolejek w klastrze. Subskrypcje za pomocą łańcuchów tematów zawierających znaki wieloznaczne, w połączeniu z tematami klastra, mogą być trudne do przewidzenia. Zachowanie jest wyjaśnione w poniższym przykładzie.

W klastrze ustawionym na przykład "Przykład: tworzenie klastra publikowania/subskrypcji produktu Sport" na stronie 85, QMB ma ten sam zestaw subskrypcji, co QMA, ale produkt QMB nie otrzymał żadnych publikacji po opublikowaniu publikatora w produkcie QMA, patrz [Rysunek 24](#) na stronie 82. Chociaż tematy Sports/Football i Sports/Rugby są tematami klastra, subskrypcje zdefiniowane w pliku `fullsubs.tst` nie odwołują się do tematu klastra. Żadne subskrypcje proxy nie są propagowane z produktu QMB do produktu QMA. Bez subskrypcji proxy żadne publikacje do produktu QMA nie są przekazywane do produktu QMB.

Niektóre subskrypcje, takie jak Sports/#/Leeds, mogą się wydawać, że odwołują się do tematu klastra, w tym przypadku Sports/Rugby. Subskrypcja Sports/#/Leeds jest rzeczywiście tłumaczona na obiekt tematu SYSTEM.BASE.TOPIC.

Reguła rozstrzygania obiektu tematu, do którego odwołuje się subskrypcja, taka jak Sports/#/Leeds jest następująca. Obetnij łańcuch tematu na pierwszy znak wieloznaczny. Skanowanie w lewo za pomocą łańcucha tematu w poszukiwaniu pierwszego tematu, który ma powiązany obiekt tematu administracyjnego. Obiekt tematu może określać nazwę klastra lub zdefiniować lokalny obiekt tematu. W przykładzie Sports/#/Leeds łańcuch tematu po obcięciu to Sports, który nie ma obiektu tematu, a więc Sports/#/Leeds dziedziczy z SYSTEM.BASE.TOPIC, który jest lokalnym obiektem tematu.

Aby zobaczyć, w jaki sposób subskrybujące tematy w klastrze mogą zmienić sposób działania propagacji znaków wieloznacznych, należy uruchomić skrypt wsadowy `upsubs.bat`. Skrypt kasuje kolejki subskrypcji i dodaje subskrypcje tematów klastra w pliku `fullsubs.tst`. Uruchom ponownie plik `puba.bat`, aby utworzyć zadanie wsadowe publikacji; patrz [Rysunek 24](#) na stronie 82.

Program Tabela 4 na stronie 84 pokazuje wynik dodawania dwóch nowych subskrypcji do tego samego menedżera kolejek, w którym publikowane były publikacje. Wynik jest taki, jak oczekiwano, nowe subskrypcje otrzymują jedną publikację, a liczba publikacji odebranych przez inne subskrypcje nie ulega zmianie. Nieoczekiwane wyniki są wyświetlane w drugim menedżerze kolejek klastra. Patrz Tabela 5 na stronie 84.

| <i>Tabela 4. Publikacje otrzymane w systemie QMA</i> | | | |
|--|-----------------------------|--|--|
| Subskrypcja | Łańcuch tematu | Otrzymane publikacje | Uwagi |
| SPORTS | Sports/# | Sports Sports/Rugby Sports/Rugby/Leeds | Wszystkie publikacje dotyczące poddrzewa piłkarskiego zablokowane przez produkt WILDCARD (BLOCK) w systemie Sports/Football |
| SARSENAL | Sports/#/Arsenal | - | Produkt WILDCARD (BLOCK) w systemie Sports/Football uniemożliwia subskrypcję zastępcze w systemie Arsenal |
| SLEEDS | Sports/#/Leeds | Sports/Rugby/Leeds | Wartość domyślna WILDCARD w programie Sports/Rugby nie uniemożliwia subskrypcji ze znakami wieloznacznymi w produkcie Leeds. |
| FARSENAL | Sports/Football/ Arsenal | Sports/Football/ Arsenal | Produkt Arsenal otrzymuje publikację, ponieważ subskrypcja nie ma znaku wieloznacznego. |
| FLEEDS | Sports/Rugby/Leeds | Sports/Rugby/Leeds | Produkt Leeds otrzyma publikację w dowolnym zdarzeniu. |

W programie Tabela 5 na stronie 84 wyświetlane są wyniki dodawania dwóch nowych subskrypcji w produkcie QMB i publikowania w produkcie QMA. Należy pamiętać, że produkt QMB nie otrzymał publikacji bez tych dwóch nowych subskrypcji. Zgodnie z oczekiwaniami dwie nowe subskrypcje otrzymują publikacje, ponieważ Sports/FootBall i Sports/Rugby są tematami klastra. Produkt QMB przekazał subskrypcje proxy dla produktów Sports/Football/Arsenal i Sports/Rugby/Leeds do produktu QMA, które następnie wysyłały publikacje do produktu QMB.

Nieoczekiwany wynik polega na tym, że dwie subskrypcje Sports/# i Sports/#/Leeds, które wcześniej nie otrzymały publikacji, otrzymują teraz publikacje. Jest to spowodowane tym, że publikacje Sports/Football/Arsenal i Sports/Rugby/Leeds przekazane do produktu QMB dla innych subskrypcji są teraz dostępne dla każdego subskrybenta dołączonego do produktu QMB. W związku z tym subskrypcje tematów lokalnych Sports/# i Sports/#/Leeds otrzymują publikację Sports/Rugby/Leeds. Produkt Sports/#/Arsenal nie otrzymuje publikacji, ponieważ wartość właściwości WILDCARD ustawiona na wartość BLOCK jest ustawiona na wartość właściwości Sport/Football.

| <i>Tabela 5. Publikacje otrzymane w systemie QMB</i> | | | |
|--|-----------------------|-----------------------------|---|
| Subskrypcja | Łańcuch tematu | Otrzymane publikacje | Uwagi |
| SPORTS | Sports/# | Sports/Rugby/ Leeds | Wszystkie publikacje dotyczące poddrzewa piłkarskiego zablokowanego przez produkt WILDCARD (BLOCK) w systemie Sports/Football |

Tabela 5. Publikacje otrzymane w systemie QMB (kontynuacja)

| Subskrypcja | Łańcuch tematu | Otrzymane publikacje | Uwagi |
|-------------|-------------------------|-------------------------|--|
| SARSENAL | Sports/#/Arsenal | - | Produkt WILDCARD (BLOCK) w systemie Sports/Football uniemożliwia subskrypcję zastępcze w systemie Arsenal |
| SLEEDS | Sports/#/Leeds | Sports/Rugby/Leeds | Wartość domyślna WILDCARD w środowisku sportowym/Rugby nie uniemożliwia subskrypcji ze znakami wieloznacznymi w produkcie Leeds. |
| FARSENAL | Sports/Football/Arsenal | Sports/Football/Arsenal | Produkt Arsenal otrzymuje publikację, ponieważ subskrypcja nie ma znaku wieloznacznego. |
| FLEEDS | Sports/Rugby/Leeds | Sports/Rugby/Leeds | Produkt Leeds otrzyma publikację w dowolnym zdarzeniu. |

W większości aplikacji jest ona niepożądana, aby jedna subskrypcja miała wpływ na zachowanie innej subskrypcji. Ważne użycie właściwości WILDCARD z wartością BLOCK polega na tym, że subskrypcje tego samego łańcucha tematu zawierające znaki wieloznaczne są jednakowo zachowane. Niezależnie od tego, czy subskrypcja znajduje się w tym samym menedżerze kolejek, co publikator, czy też inna, wyniki subskrypcji są takie same.

Znaki wieloznaczne i strumienie

W przypadku nowej aplikacji napisanej do interfejsu API publikowania/subskrypcji efekt jest taki, że subskrypcja produktu * nie odbiera żadnych publikacji. Aby otrzymywać wszystkie publikacje dotyczące sportu, należy zasubskrybować produkty Sports/*lub Sports/#, podobnie jak w przypadku publikacji Business .

Zachowanie istniejącej w kolejce aplikacji publikowania/subskrypcji nie zmienia się, gdy broker publikowania/subskrybowania jest migrowany do wersji IBM WebSphere MQ 7 i nowszych. Właściwość **StreamName** w komendach **Publish**, **Register Publisher** lub **Subscriber** jest odwzorowana na nazwę tematu, do którego migrowany jest strumień.

Znaki wieloznaczne i punkty subskrypcji

W przypadku nowej aplikacji napisanej w interfejsie API publikowania/subskrypcji efekt migracji jest taki, że subskrypcja produktu * nie odbiera żadnych publikacji. Aby otrzymywać wszystkie publikacje dotyczące sportu, należy zasubskrybować produkty Sports/*lub Sports/#, podobnie jak w przypadku publikacji Business .

Zachowanie istniejącej w kolejce aplikacji publikowania/subskrypcji nie zmienia się, gdy broker publikowania/subskrybowania jest migrowany do wersji IBM WebSphere MQ 7 i nowszych. Właściwość **SubPoint** w komendach **Publish**, **Register Publisher** lub **Subscriber** jest odwzorowana na nazwę tematu, do którego migrowana została subskrypcja.

Przykład: tworzenie klastra publikowania/subskrypcji produktu Sport

Kroki, które należy wykonać, tworzą klastr CL1z czterema menedżerami kolejek: dwoma pełnymi repozytoriami, CL1A i CL1B oraz dwoma częściowymi repozytoriami, QMA i QMB. Pełne repozytoria są używane do przechowywania tylko definicji klastrów. QMA jest oznaczony jako host tematu klastra. Trwałe subskrypcje są definiowane na serwerze QMA i QMB.

Uwaga: Przykład jest kodowany dla produktu Windows. Należy ponownie utworzyć plik Create qmgrs.bat i create pub.bat , aby skonfigurować i przetestować przykład na innych platformach.

1. Utwórz pliki skryptowe.
 - a. Utwórz plik topics.tst
 - b. Utwórz element wildsubs.tst
 - c. Utwórz fullsubs.tst
 - d. Tworzenie pliku qmgrs.bat
 - e. utwórz plik pub.bat
2. Uruchom komendę Create qmgrs.bat , aby utworzyć konfigurację.

```
qmgrs
```

Utwórz tematy w programie Rysunek 23 na stronie 82. Skrypt na rysunku 5 tworzy tematy klastra Sports/Football i Sports/Rugby.

Uwaga: Opcja REPLACE nie zastępuje właściwości TOPICSTR tematu. TOPICSTR to właściwość, która jest użytecznie udostępniona w przykładzie w celu przetestowania różnych drzew tematów. Aby zmienić tematy, należy najpierw usunąć ten temat.

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

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

Rysunek 25. Usuwanie i tworzenie tematów: topics.tst

Uwaga: Usuń tematy, ponieważ produkt REPLACE nie zastępuje łańcuchów tematów.

Utwórz subskrypcje z użyciem znaków wieloznacznych. Znaki wieloznaczne odpowiadające tematyce z obiektami tematu w programie Rysunek 23 na stronie 82. Utwórz kolejkę dla każdej subskrypcji. Kolejki są kasowane, a subskrypcje są usuwane po uruchomieniu skryptu lub ponownym uruchomieniu.

Uwaga: Opcja REPLACE nie zastępuje właściwości TOPICOBJ ani TOPICSTR subskrypcji. TOPICOBJ lub TOPICSTR to właściwości, które są przydatne na przykład w celu przetestowania różnych subskrypcji. Aby je zmienić, najpierw usuń subskrypcję.

```

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

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

```

Rysunek 26. Utwórz subskrypcje ze znakami wieloznacznymi: wildsubs.tst

Utwórz subskrypcje, które odwołują się do obiektów tematu klastra.

Uwaga:

Separator, /, jest automatycznie wstawiany między łańcuchem tematu, do którego odwołuje się TOPICOBJ, a łańcuchem tematu zdefiniowanym przez parametr TOPICSTR.

Definicja DEFINE SUB(FARSENAL) TOPICSTR('Sports/Football/Arsenal') DEST(QFARSENAL) tworzy tę samą subskrypcję. TOPICOBJ jest używany jako szybki sposób odniesienia do już zdefiniowanego łańcucha tematu. Jeśli subskrypcja została utworzona, nie odnosi się już do obiektu tematu.

```

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

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

```

Rysunek 27. Usuń i utwórz subskrypcje: fullsubs.tst

Utwórz klaster z dwoma repozytoriami. Utwórz dwa częściowe repozytoria dla publikowania i subskrybowania. Uruchom skrypt ponownie, aby usunąć wszystko i rozpocząć ponownie. Skrypt tworzy również hierarchię tematów oraz początkowe subskrypcje ze znakami wieloznacznymi.

Uwaga:

Na innych platformach napisz podobny skrypt lub wpisz wszystkie komendy. Użycie skryptu pozwala na szybkie usuwanie wszystkiego i uruchamianie za pomocą identycznej konfiguracji.

```

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

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

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

```

Rysunek 28. Tworzenie menedżerów kolejek: qmgrs.bat

Zaktualizuj konfigurację, dodając subskrypcje tematów klastra.

```

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

```

Rysunek 29. Aktualizacja subskrypcji: upsubs.bat

Uruchom program pub.bat z menedżerem kolejek jako parametrem, aby opublikować komunikaty zawierające łańcuch tematu publikacji. Program Pub.bat korzysta z przykładowego programu **amqspub**.

```

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

```

Rysunek 30. Opublikuj: pub.bat

Strumienie i tematy

W kolejce publikowania/subskrypcji istnieje pojęcie strumienia publikacji, który nie istnieje w zintegrowanym modelu publikowania/subskrybowania. Strumienie publikowania/subskrybowania w kolejce umożliwiają rozdzielenie przepływu informacji na różne tematy. Począwszy od wersji IBM WebSphere MQ 7.0, strumień jest implementowany jako temat najwyższego poziomu, który może być odwzorowany na inny identyfikator tematu administracyjnie.

Strumień domyślny SYSTEM.BROKER.DEFAULT.STREAM jest automatycznie skonfigurowany dla wszystkich brokerów i menedżerów kolejek w sieci, a do korzystania z strumienia domyślnego nie jest wymagana żadna dodatkowa konfiguracja. Pomyśl o strumieniu domyślnym jako nienazwanym domyślnym obszarze tematu. Tematy opublikowane w strumieniu domyślnym są natychmiast dostępne dla wszystkich połączonych menedżerów kolejek począwszy od wersji IBM WebSphere MQ 7.0, a włączone w kolejce publikowanie/subskrypcja jest włączone. Nazwane strumienie są jak oddzielne,

nazwane, przestrzenie tematów. Nazwany strumień musi być zdefiniowany w każdym brokerze, w którym jest używany.

Jeśli publikatory i subskrybenci znajdują się w różnych menedżerach kolejek, to po nawiązaniu połączenia brokerów w tej samej hierarchii brokera nie są wymagane żadne dalsze konfigurowanie publikacji ani subskrypcje przepływu między nimi. Ta sama interoperacyjność działa także w odwrotnej kolejności.

potoki nazwane

Projektant rozwiązań, współpracując z umieszczonym w kolejce modelem programowania publikowania/subskrybowania, może zdecydować o umieszczeniu wszystkich publikacji sportowych w nazwanym strumieniu o nazwie Sport. W produkcie IBM WebSphere MQ 6.0 strumień jest często replikowany automatycznie na innych brokerach, które korzystają z kolejki modelowej, SYSTEM.BROKER.MODEL.STREAM. Jednak w przypadku strumienia, który ma być dostępny dla menedżera kolejek, który jest uruchamiany w systemie IBM WebSphere MQ 7.0, a następnie włączone w kolejce publikowania/subskrybowania, strumień musi zostać dodany ręcznie.

Kolejkowane aplikacje publikowania/subskrybowania, które subskrybują Soccer/Results w strumieniu Sport, działają bez zmian. Zintegrowane aplikacje publikowania/subskrybowania, które subskrybują temat Sport przy użyciu produktu MQSUB, a także dostarczając łańcuch tematu Soccer/Results, otrzymują te same publikacje.

Czynność dodawania strumienia jest opisana w temacie [Dodawanie strumienia](#). Może być konieczne ręczne dodanie strumieni z dwóch powodów.

1. W dalszym ciągu rozwijasz kolejkowane aplikacje publikowania/subskrypcji, które działają w menedżerach kolejek w nowszej wersji, a nie migrują aplikacje do zintegrowanego interfejsu MQI publikowania/subskrypcji.
2. Domyślne odwzorowanie strumieni na tematy prowadzi do "kolizji" w obszarze tematów, a publikacje w strumieniu mają ten sam łańcuch tematu co publikacje z innego miejsca.

Uprawnienia

Domyślnie w katalogu głównym drzewa tematów znajduje się wiele obiektów tematu: SYSTEM.BASE.TOPIC, SYSTEM.BROKER.DEFAULT.STREAM i SYSTEM.BROKER.DEFAULT.SUBPOINT. Uprawnienia (na przykład w celu publikowania lub subskrybowania) są określane przez uprawnienia na serwerze SYSTEM.BASE.TOPIC. Wszystkie uprawnienia na serwerze SYSTEM.BROKER.DEFAULT.STREAM lub SYSTEM.BROKER.DEFAULT.SUBPOINT są ignorowane. Jeśli albo SYSTEM.BROKER.DEFAULT.STREAM, albo SYSTEM.BROKER.DEFAULT.SUBPOINT zostaną usunięte i utworzone ponownie z niepustym łańcuchem tematu, to uprawnienia zdefiniowane w tych obiektach są używane w taki sam sposób, jak normalny obiekt tematu.

Odzworowanie między strumieniami i tematami

Umieszczony w kolejce strumień publikowania/subskrypcji jest mimiany w programie IBM WebSphere MQ 7.0, tworząc kolejkę i nadając mu taką samą nazwę jak strumień. Czasami kolejka jest nazywana kolejką strumieniową, ponieważ jest ona wyświetlana w kolejkach aplikacji publikowania/subskrypcji. Kolejka jest identyfikowana do mechanizmu publikowania/subskrybowania, dodając ją do specjalnej listy nazw o nazwie SYSTEM.QPUBSUB.QUEUE.NAMELIST. Można dodać tyle strumieni, ile potrzeba, dodając dodatkowe kolejki specjalne do listy nazw. W końcu należy dodać tematy o tych samych nazwach, co strumienie, oraz te same łańcuchy tematów, co nazwa strumienia, co umożliwi publikowanie i subskrybowanie tematów.

Jednak w wyjątkowych okolicznościach można podać tematy odpowiadające strumieniom wszystkich łańcuchów tematów, które zostaną wybrane podczas definiowania tematów. Celem łańcucha tematu jest nadanie tematu unikalnej nazwy w obszarze tematu. Zwykle nazwa strumienia doskonale służy temu celowi. Czasami koliduje się nazwa strumienia i istniejąca nazwa tematu. Aby rozwiązać ten problem, należy wybrać inny łańcuch tematu dla tematu powiązanego ze strumieniem. Wybierz dowolny łańcuch tematu, upewniając się, że jest on unikalny.

Łańcuch tematu zdefiniowany w definicji tematu jest poprzedzony w normalny sposób łańcuchem tematu udostępnionym przez publikatorów i subskrybentów przy użyciu wywołań MQI produktu MQOPEN lub MQSUB. Aplikacje odwołując się do tematów za pomocą obiektów tematów nie mają wpływu na wybór łańcucha tematu przedrostka-dłatego można wybrać dowolny łańcuch tematu, który przechowuje publikacje unikalne w obszarze tematu.

Zmiana odwzorowania różnych strumieni na różne tematy polega na tym, że przedrostki używane dla łańcuchów tematów są unikalne, aby oddzielić jeden zestaw tematów całkowicie od innych. Należy zdefiniować konwencję nazewnictwa tematów uniwersalnych, która jest sztywno przylegająca do odwzorowania do pracy.

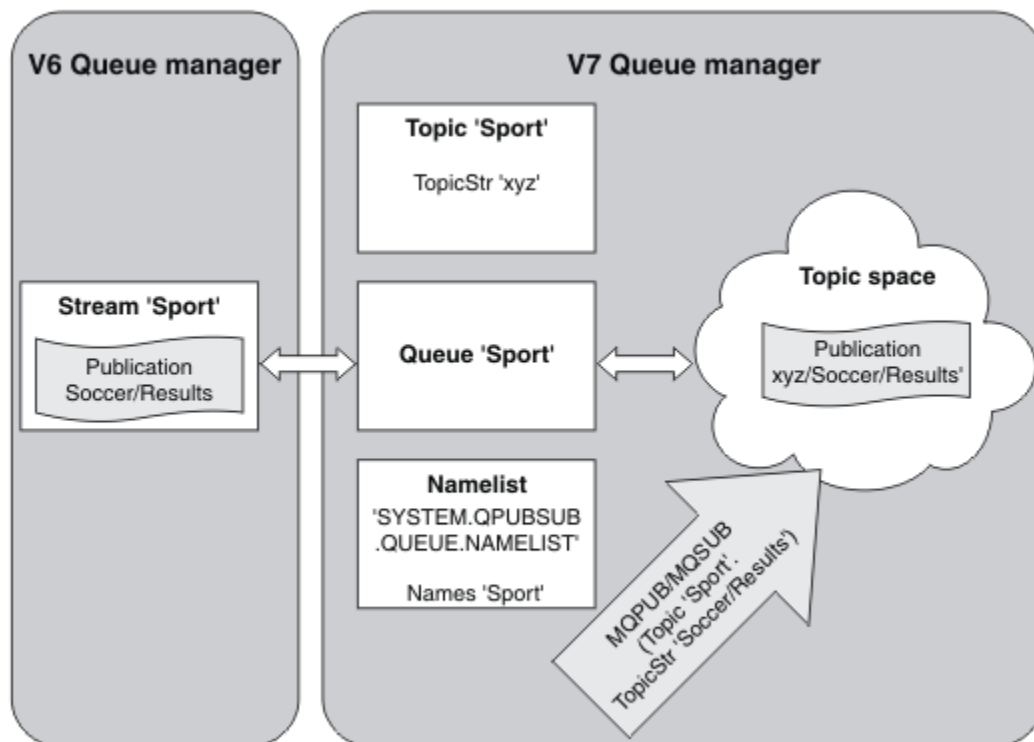
W przypadku kolizji łańcuchów tematów w produkcie IBM WebSphere MQ 7.0 mogą być używane strumienie do oddzielania obszarów tematów.

Począwszy od wersji IBM WebSphere MQ 7.0, można użyć mechanizmu wstępnego do zmiany odwzorowania łańcucha tematu w inne miejsce w obszarze tematu.

Uwaga: Po usunięciu strumienia należy najpierw usunąć wszystkie subskrypcje w strumieniu. To działanie jest najważniejsze, jeśli którekolwiek z subskrypcji pochodzą od innych brokerów w hierarchii brokera.

Przykład

W produkcie Rysunek 31 na stronie 90 temat 'Sport' zawiera łańcuch tematu 'xyz', co powoduje, że publikacje pochodzące ze strumienia 'Sport' są poprzedzane przedrostkiem 'xyz' w obszarze tematu menedżera kolejek w wersji 7. Publikowanie lub subskrybowanie w wersji 7 tematu 'Sport' prefiksów 'xyz' do łańcucha tematu. Jeśli publikacja jest przepływowa do subskrybenta IBM WebSphere MQ 6, przedrostek 'xyz' jest usuwany z publikacji i jest umieszczany w strumieniu 'Sport'. I odwrotnie, gdy publikacja przepływa z programu IBM WebSphere MQ 6 do produktu IBM WebSphere MQ 7, z strumienia 'Sport' do tematu 'Sport', przedrostek 'xyz' jest dodawany do łańcucha tematu.



Rysunek 31. Strumienie IBM WebSphere MQ 6, które współistnieją z tematami produktu IBM WebSphere MQ 7

Punkty subskrypcji i tematy

Nazwane punkty subskrypcji są emulowane przez tematy i obiekty tematów.

Aby ręcznie dodać punkty subskrypcji, należy zapoznać się z sekcji [Dodawanie punktu subskrypcji](#).

Punkty subskrypcji w produkcie IBM MQ

Produkt IBM MQ odwzorowuje punkty subskrypcji na różne obszary tematów w drzewie tematów produktu IBM MQ. Tematy w komunikatach komend bez punktu subskrypcji są odwzorowywane bez zmian w katalogu głównym drzewa tematów produktu IBM MQ i dziedziczą właściwości z produktu SYSTEM.BASE.TOPIC.

Komunikaty komend z punktem subskrypcji są przetwarzane przy użyciu listy obiektów tematów w produkcie SYSTEM.QPUBSUB.SUBPOINT.NAMELIST. Nazwa punktu subskrypcji w komunikacie komendy jest porównana z łańcuchem tematu dla każdego z obiektów tematu na liście. Jeśli zostanie znalezione dopasowanie, nazwa punktu subskrypcji zostanie wstępnie zakończona, jako węzeł tematu, do łańcucha tematu. Temat dziedziczy jego właściwości z powiązanego obiektu tematu znalezionego w produkcie SYSTEM.QPUBSUB.SUBPOINT.NAMELIST.

Efektorem korzystania z punktów subskrypcji jest utworzenie osobnego obszaru tematu dla każdego punktu subskrypcji. Obszar tematu jest zakorzeniony w temacie o tej samej nazwie co punkt subskrypcji. Tematy w każdym obszarze tematu dziedziczą ich właściwości z obiektu tematu o takiej samej nazwie, co punkt subskrypcji.

Wszystkie właściwości, które nie są ustawione w pasującym obiekcie tematu, są dziedziczone, w normalny sposób, od SYSTEM.BASE.TOPIC.

Istniejące w kolejce aplikacje publikowania/subskrypcji, przy użyciu nagłówków komunikatów produktu MQRFH2, kontynuują pracę, ustawiając właściwość **SubPoint** w komunikatach komend produktu Publish lub Register subscriber. Punkt subskrypcji jest łączony z łańcuchem tematu w komunikacie komendy, a otrzymany temat został przetworzony tak, jak każdy inny.

Produkt IBM WebSphere MQ 7.0 lub nowszy nie ma wpływu na aplikacje w punktach subskrypcji. Jeśli aplikacja korzysta z tematu, który dziedziczy informacje z jednego z pasujących obiektów tematów, aplikacja współpracuje z kolejną aplikacją przy użyciu zgodnego punktu subskrypcji.

Przykład

Istniejący WebSphere Message Broker (znany teraz pod nazwą IBM Integration Bus) Aplikacja publikowania/subskrypcji, która została zmigrowana do produktu IBM MQ, utworzyła dwa obiekty tematów, GBP i USD, z odpowiednimi łańcuchami tematów 'GBP' i 'USD'.

Istniejące publikatory w temacie NYSE/IBM/SPOT, poddane migracji w celu uruchomienia na serwerze IBM MQ, które korzystają z punktu subskrypcji USD, tworzą publikacje na temat USD/NYSE/IBM/SPOT. Podobnie istniejący subskrybenci produktu NYSE/IBM/SPOT, korzystając z punktu subskrypcji USD, tworzą subskrypcje na serwerze USD/NYSE/IBM/SPOT.

Zasubskrybuj cenę obszaru dolara w programie IBM WebSphere MQ 7.0 lub w późniejszym czasie, wywołując program publikowania/subskrypcji, wywołując program MQSUB. Utwórz subskrypcję przy użyciu obiektu tematu USD i łańcucha tematu 'NYSE/IBM/SPOT', jak pokazano w fragmencie kodu 'C'.

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

1. Ustaw atrybut CLUSTER obiektów tematów USD i GBP na hoście tematu klastra.
2. Usuń wszystkie kopie obiektów tematów USD i GBP w innych menedżerach kolejek w klastrze.
3. Upewnij się, że produkt USD i GBP są zdefiniowane w produkcie SYSTEM.QPUBSUB.SUBPOINT.NAMELIST w każdym menedżerze kolejek w klastrze.

Przykład konfiguracji publikowania/subskrypcji pojedynczego menedżera kolejek

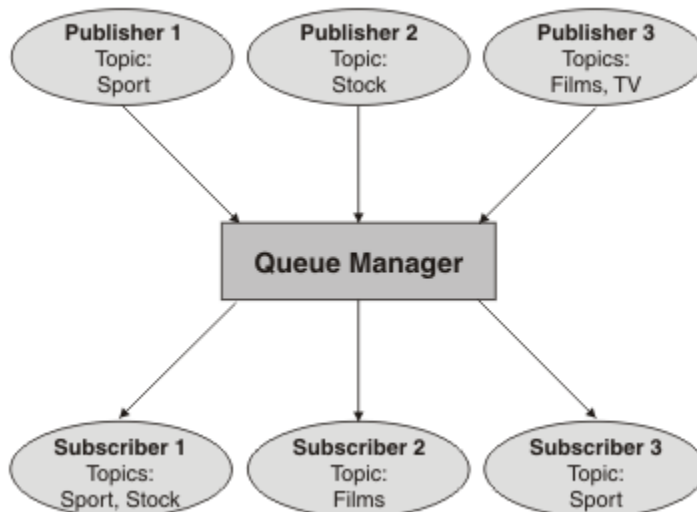
Rysunek 32 na stronie 92 przedstawia podstawową konfigurację publikowania/subskrybowania pojedynczego menedżera kolejek. W przykładzie przedstawiono konfigurację usługi informacyjnej, w której informacje są dostępne od publikatorów na temat kilku tematów:

- Wydawca 1 publikuje informacje o wynikach sportowych za pomocą tematu Sport
- Wydawca 2 publikuje informacje o cenach akcji z wykorzystaniem tematu Stock
- Wydawca 3 publikuje informacje o przeglądach filmowych z wykorzystaniem tematu Films, oraz o listingu telewizyjnym z wykorzystaniem tematu TV

Trzech subskrybentów zarejestrowało zainteresowanie w różnych tematach, dlatego menedżer kolejek wysyła im informacje, które są zainteresowane:

- Subskrybent 1 otrzymuje wyniki sportowe i ceny akcji
- Subskrybent 2 otrzymuje recenzje filmowe
- Abonent 3 otrzymuje wyniki sportowe

Żaden z abonentów nie zarejestrował zainteresowania listem telewizyjnym, dlatego nie są one dystrybuowane.



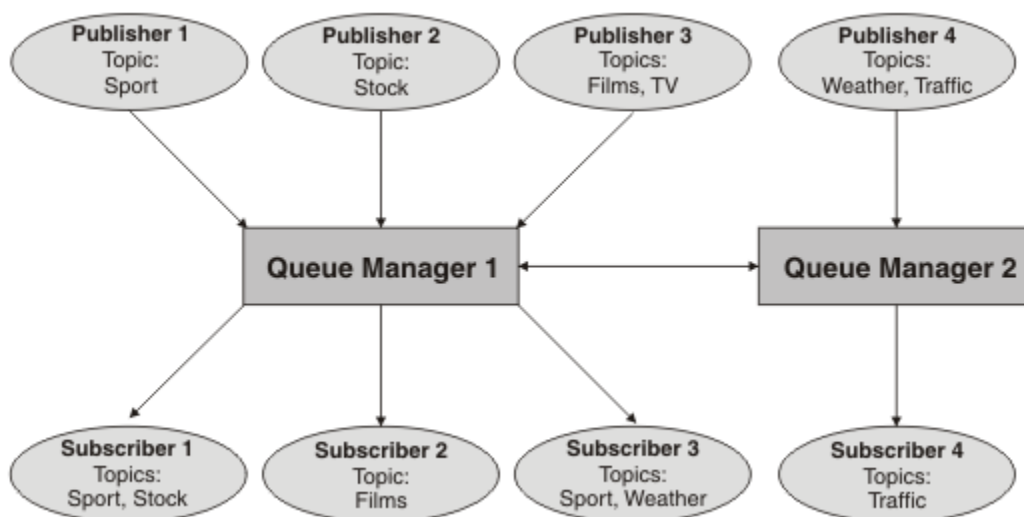
Rysunek 32. Przykład publikowania/subskrypcji pojedynczego menedżera kolejek

Rozproszone sieci publikowania/subskrypcji

Każdy menedżer kolejek jest zgodny z komunikatami opublikowanymi w temacie z lokalnie utworzonymi subskrypcjami, które subskrybują ten temat. Sieć menedżerów kolejek można skonfigurować w taki sposób, aby komunikaty publikowane przez aplikację połączoną z jednym menedżerem kolejek były dostarczane do zgodnych subskrypcji utworzonych w innych menedżerach kolejek w sieci. Wymaga to dodatkowej konfiguracji przy użyciu prostych kanałów między menedżerami kolejek.

Konfiguracja rozproszonej konfiguracji publikowania/subskrypcji to zestaw połączonych ze sobą menedżerów kolejek. Wszystkie menedżery kolejek mogą znajdować się w tym samym systemie fizycznym lub mogą być dystrybuowane przez kilka systemów fizycznych. Po połączeniu menedżerów kolejek subskrybenci mogą subskrybować jeden menedżer kolejek i odbierać komunikaty, które zostały początkowo opublikowane w innym menedżerze kolejek. Poniższy rysunek ilustruje dodanie drugiego menedżera kolejek do konfiguracji opisanej w sekcji [“Przykład konfiguracji publikowania/subskrypcji pojedynczego menedżera kolejek”](#) na stronie 92.

- Menedżer kolejek 2 jest używany przez publikatora 4 do publikowania informacji o prognozie pogody, przy użyciu tematu Weather, oraz informacji o warunkach ruchu na głównych drogach, przy użyciu tematu Traffic.
- Subskrybent 4 korzysta również z tego menedżera kolejek i subskrybuje informacje o warunkach ruchu w ruchu tematu.
- Subskrybent 3 subskrybuje również informacje o warunkach pogodowych, nawet jeśli korzysta z innego menedżera kolejek od wydawcy. Jest to możliwe, ponieważ menedżery kolejek są ze sobą powiązane.

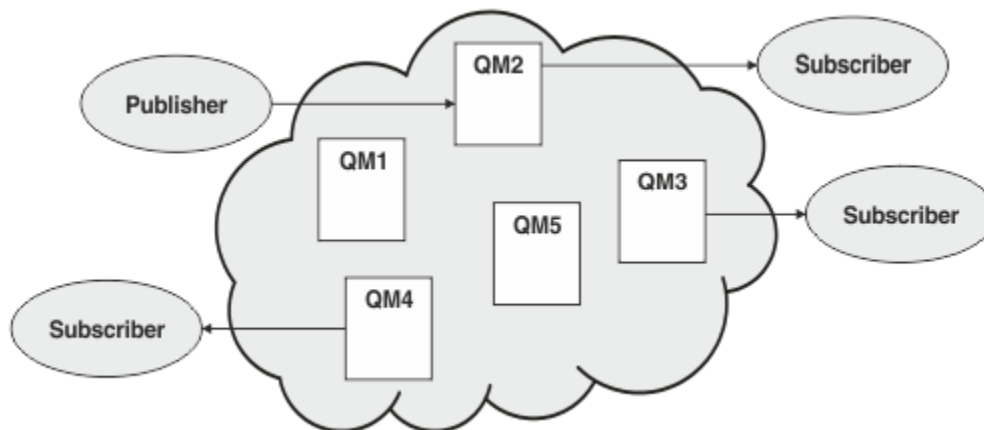


Rysunek 33. Przykład publikowania/subskrypcji z dwoma menedżerami kolejek

Menedżery kolejek można łączyć ręcznie w hierarchii nadrzędnej i podrzędnej albo można utworzyć klaster publikowania/subskrybowania, a IBM MQ zdefiniować wiele szczegółów połączenia dla użytkownika. Można również użyć obu topologii w połączeniu, na przykład łącząc kilka klastrów w hierarchię.

Przegląd klastrów publikowania/subskrypcji

Klaster publikowania/subskrypcji jest standardowym klastrem z jednym lub większą liczbą obiektów tematów dodanych do klastra. Po zdefiniowaniu obiektu tematu administracyjnego w dowolnym menedżerze kolejek w klastrze i udostępnieniem tego obiektu tematu w klastrze poprzez określenie nazwy klastra, publikatory i subskrybenty tematu mogą łączyć się z dowolnymi menedżerami kolejek w klastrze, a publikowane komunikaty są kierowane do subskrybentów przez kanały klastra między menedżerami kolejek.



Rysunek 34. klaster publikowania/subskrypcji

Istnieją dwa sposoby konfigurowania sposobu kierowania komunikatów publikowania/subskrypcji w klastrze:

- kierowanie bezpośrednie
- kierowanie hostami tematów

Podczas konfigurowania bezpośredniego tematu klastra kierowanego komunikaty opublikowane w jednym menedźerze kolejek są wysyłane bezpośrednio z tego menedżera kolejek do każdej subskrypcji w dowolnym innym menedźerze kolejek w klastrze. Może to zapewnić najbardziej bezpośrednią ścieżkę do publikacji, ale powoduje, że wszystkie menedżery kolejek w klastrze będą rozpoznawać wszystkie inne menedżery kolejek, a każdy potencjalnie może mieć ustanowione kanały klastra.

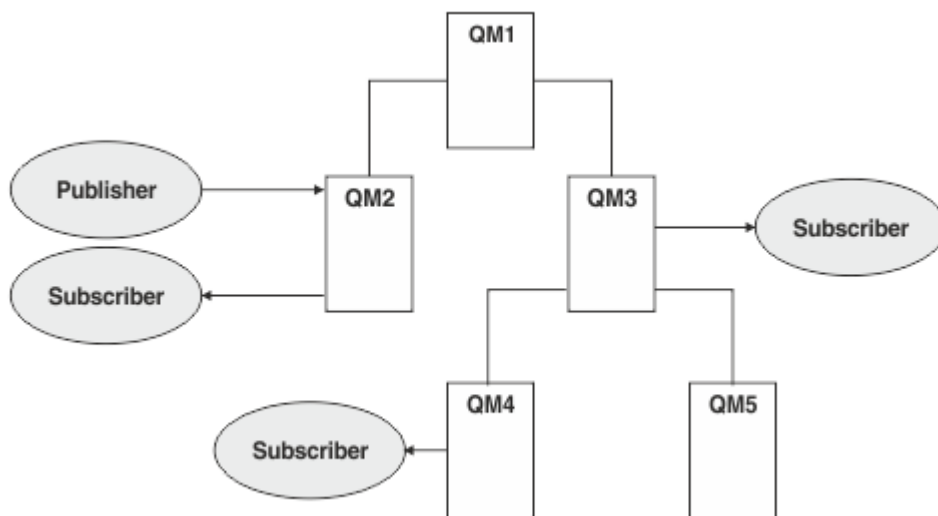
W przypadku korzystania z routingu hosta tematów komunikaty publikowane w jednym menedźerze kolejek są wysyłane z tego miejsca do menedżera kolejek, który udostępnia definicję administrowanego obiektu tematu. Ten *menedżer kolejek hosta tematu* kieruje komunikaty do każdej subskrypcji we wszystkich pozostałych menedżerach kolejek w klastrze. Jeśli publikatory lub subskrybenci nie znajdują się w menedżerach kolejek hostów tematów, to powoduje to dłuższą trasę dla publikacji. Jednak korzyścią jest to, że tylko menedżerowie kolejek hostów tematów będą świadomi wszystkich innych menedżerów kolejek w klastrze i potencjalnie mogą mieć nawiązane kanały klastrowe.

Więcej informacji na ten temat zawiera sekcja [“Klastry publikowania/subskrypcji”](#) na stronie 95.

Przegląd hierarchii publikowania/subskrybowania

Hierarchia publikowania/subskrypcji jest zestawem menedżerów kolejek połączonych kanałami w strukturę hierarchiczną. Każdy menedżer kolejek identyfikuje swój *nadrzędny* menedżer kolejek zgodnie z opisem w sekcji [Podłączanie menedżera kolejek do hierarchii publikowania/subskrypcji](#).

Publikatory i subskrybenci tematu mogą łączyć się z dowolnym menedżerem kolejek w hierarchii, a przepływ komunikatów między nimi jest używany przy użyciu hierarchicznego połączenia menedżera kolejek.



Rysunek 35. Hierarchia publikowania/subskrypcji

Na poprzednim rysunku publikacje dostarczone do subskrybentów QM3 i QM4 zostały przekierowane z QM2 do QM1, a następnie na QM3 i na końcu QM4.

Hierarchie dają bezpośrednią kontrolę nad relacjami między każdym menedżerem kolejek w hierarchii. Umożliwia to precyzyjną kontrolę nad przepływem komunikatów od publikatorów do subskrybentów i jest szczególnie przydatne podczas kierowania między sieciami menedżera kolejek z ograniczonymi połączeniami. Należy zwrócić szczególną uwagę na dostępność i możliwość każdego menedżera kolejek, przez który komunikat jest kierowany w drodze od publikatora do subskrybentów.

Więcej informacji na ten temat zawiera sekcja [“Hierarchie publikowania/subskrypcji”](#) na stronie 98.

Dystrybucja publikacji między menedżerami kolejek

Oprócz opcji routingu dostępne są dwa podejścia do dystrybucji publikacji w sieci menedżerów kolejek:

- Publikacje można wysyłać tylko z jednego menedżera kolejek do menedżerów kolejek, które obecnie są subskrybowane dla tej publikacji.
- Wysyłaj każdą publikację do wszystkich menedżerów kolejek i pozwól im dopasować je do swoich subskrypcji.

Poprzednie wyniki w komunikatach publikacji są wysyłane tylko tam, gdzie jest to konieczne, ale wymagają poziomu wiedzy o subskrypcji, który ma być współużytkowany przez menedżery kolejek. Ten ostatni nie wymaga współużytkowania wiedzy o subskrypcji, ale może spowodować wysyłanie zbędnych komunikatów publikowania między menedżerami kolejek.

Domyślnie produkt IBM MQ korzysta z poprzedniej metody, w której publikacje są wysyłane tylko do menedżerów kolejek, które mają subskrypcje dla nich. Wiedza o subskrypcji jest propagowana między menedżerami kolejek w postaci *subskrypcji proxy*. Zależy ona od dystrybucji i czasu życia subskrypcji oraz częstotliwości publikacji, która jest najbardziej efektywna w obsłudze w topologii rozproszonej publikowania/subskrypcji. Więcej informacji zawiera sekcja [Wydajność subskrypcji w sieciach publikowania/subskrypcji](#).

Pojęcia pokrewne

[“Drzewa tematów”](#) na stronie 77

Każdy definiowany temat stanowi element lub węzeł drzewa tematów. Drzewo tematów może być puste, aby rozpocząć od lub zawierać tematy, które zostały wcześniej zdefiniowane przy użyciu komend MQSC lub PCF. Nowy temat można zdefiniować, korzystając z komend tworzenia tematów lub podając temat po raz pierwszy w publikacji lub subskrypcji.

Zadania pokrewne

[Projektowanie klastrów publikowania/subskrypcji](#)

Odsyłacze pokrewne

[Scenariusze hierarchii publikowania/subskrypcji](#)

Klastry publikowania/subskrypcji

Klaster publikowania/subskrypcji jest standardowym klastrem połączonych ze sobą menedżerów kolejek, na których publikacje są automatycznie przenoszone z aplikacji publikowania do subskrypcji istniejących w dowolnym menedżerze kolejek w klastrze. Istnieją dwie opcje kierowania publikacji w ramach klastra publikowania/subskrybowania: *kierowanie bezpośrednio* i *kierowanie hostami tematów*. Wybór trasy zależy od wielkości i oczekiwanych wzorców działań w klastrze.

Klaster używany do przesyłania komunikatów w trybie publikowania/subskrypcji nie różni się od standardowego klastra IBM MQ. W związku z tym menedżery kolejek w klastrze publikowania/subskrybowania mogą istnieć na fizycznie oddzielnych komputerach, a każda para menedżerów kolejek jest automatycznie łączona przez kanały klastra, gdy jest to konieczne. Więcej informacji na ten temat zawiera sekcja [Klastry](#).

Aby skonfigurować standardowy klaster menedżerów kolejek na potrzeby przesyłania komunikatów publikacji/subskrypcji, należy zdefiniować co najmniej jeden administrowany obiekt tematu w menedżerze kolejek klastra. Aby wykonać temat w klastrze, należy skonfigurować właściwość **CLUSTER** z nazwą klastra. W takim przypadku każdy temat używany przez publikator lub subskrybent w tym punkcie lub poniżej w drzewie tematów jest współużytkowany przez wszystkie menedżery kolejek w klastrze, a komunikaty publikowane w gałęzi klastra w drzewie tematów są automatycznie kierowane do subskrypcji w innych menedżerach kolejek w klastrze.

Tylko jedna kopia każdego komunikatu jest wysyłana między menedżerem kolejek publikatora i każdym innym menedżerem kolejek, niezależnie od liczby subskrybentów dla komunikatu w docelowym menedżerze kolejek. Po przybyciu do menedżera kolejek z jedną lub większą liczbą subskrypcji komunikat jest duplikowany we wszystkich subskrypcjach.

Każdy menedżer kolejek dołączany do klastra automatycznie staje się świadomy tematów klastrowych, a publikatory i subskrybenty tego menedżera kolejek automatycznie uczestniczą w klastrze.

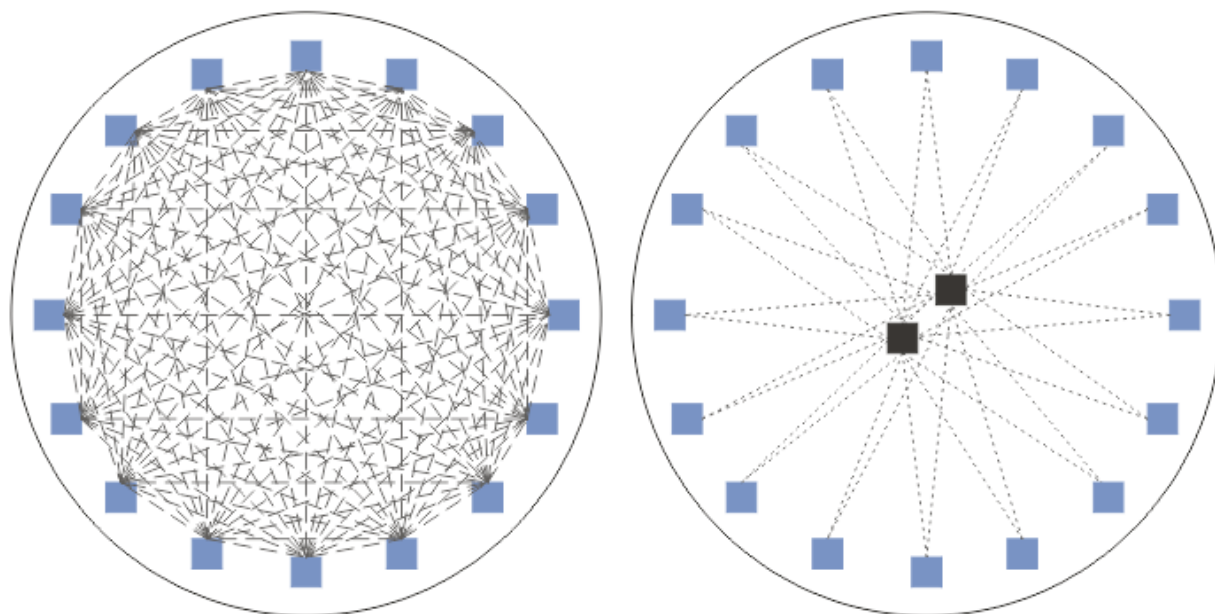
Nieklastrowe działanie publikowania/subskrybowania może również odbywać się w klastrze publikowania/subskrybowania, pracując z łańcuchami tematów, które nie są objęte grupowym obiektem tematu.

Istnieją dwie opcje kierowania publikacji w ramach klastra publikowania/subskrybowania: *kierowanie bezpośrednio* i *kierowanie hostami tematów*. Aby wybrać metodę kierowania komunikatów do użycia w ramach klastra, należy ustawić właściwość **CLROUTE** w administrowanym obiekcie tematu na jedną z następujących wartości:

- **DIRECT**
- **TOPICHOST**

Domyślnie kierowanie tematami odbywa się w systemie **DIRECT**. Była to jedyna opcja w produktach starszych niż IBM MQ 8.0. Po skonfigurowaniu bezpośredniego kierowanego tematu klastra w menedżerze kolejek wszystkie menedżery kolejek w klastrze będą powiadomione o obecności wszystkich innych menedżerów kolejek w klastrze. Podczas wykonywania operacji publikowania i subskrypcji każdy menedżer kolejek może nawiązać bezpośrednie połączenie z dowolnym innym menedżerem kolejek w klastrze.

W produkcie IBM MQ 8.0 można zamiast tego skonfigurować kierowanie tematami za pomocą opcji **TOPICHOST**. Jeśli używane jest kierowanie hostami tematów, wszystkie menedżery kolejek w klastrze będą powiadomione o menedżerach kolejek klastra, które udostępniają definicje kierowanych tematów (czyli o menedżerach kolejek, w których zdefiniowano obiekt tematu). Podczas wykonywania operacji publikowania i subskrypcji menedżery kolejek w klastrze nawiązują połączenie tylko z tymi menedżerami kolejek hostów tematów, a nie bezpośrednio ze sobą. Menedżery kolejek hostów tematów są odpowiedzialne za kierowanie publikacji z menedżerów kolejek, na których publikacje są publikowane, do menedżerów kolejek ze zgodnymi subskrypcjami.



Rysunek 36. Kierowanie bezpośrednio i kierowanie hostami tematów

Przegląd kierowania bezpośredniego

Jeśli administrowany obiekt tematu jest skonfigurowany pod kątem kierowania bezpośredniego, obiekt tematu musi być zdefiniowany tylko na jednym z menedżerów kolejek w klastrze, aby wszystkie menedżery kolejek mogły się z nim zapoznać. Wybór menedżera kolejek, w którym zdefiniowany jest

temat, nie ma wpływu na działanie przesyłania komunikatów w trybie publikowania/subskrypcji dla tego tematu.

Każdy komunikat przepływa bezpośrednio z menedżera kolejek publikatora do każdej subskrypcji w innych menedżerach kolejek w klastrze, a nie przechodząc przez żadne pośrednie menedżery kolejek.

Domyślnie komunikaty są wysyłane tylko do innych menedżerów kolejek w klastrze, w których istnieje jedna lub większa liczba subskrypcji.

- Polega to na tym, że każdy menedżer kolejek jest bezpośrednio informowany o wszystkich innych menedżerach kolejek w klastrze wszystkich tematów, które mają obecnie jedną lub więcej subskrypcji. Powoduje to, że wszystkie menedżery kolejek w klastrze są świadome wszystkich subskrybowanych tematów, a także każdy menedżer kolejek udostępniający subskrypcję nawiązuje kanał do każdego innego menedżera kolejek. Jest to niezależne od tego, czy każdy menedżer kolejek ma wydawcę.
- Znajomość każdego pojedynczego subskrybowanego tematu we wszystkich menedżerach kolejek może zostać usunięta poprzez zmianę na model wysyłania wszystkich publikacji do wszystkich menedżerów kolejek w klastrze, bez względu na to, czy mają one subskrypcje. Zmniejsza to ruch wiedzy o subskrypcji, ale prawdopodobnie zwiększy ruch publikacji i liczbę kanałów ustalanych przez każdego menedżera kolejek. Więcej informacji zawiera sekcja [Wydajność subskrypcji w sieciach publikowania/subskrypcji](#).

Przepływy komunikatów publikowania/subskrypcji korzystające z tematów w klastrze według bezpośredniego kierowania mogą obejmować wiele klastrów publikowania/subskrypcji, dodając jeden menedżer kolejek z każdego klastra do hierarchii publikowania/subskrypcji. Więcej informacji na ten temat zawiera sekcja [Łączenie obszarów tematów z wieloma klastrami](#).

Bardziej szczegółowe eksploracja kierowania bezpośredniego znajduje się w temacie [Routing bezpośredni w klastrach publikowania/subskrypcji](#).

Przegląd kierowania hostami tematów

Jeśli administrowany obiekt tematu jest skonfigurowany dla routingu hosta tematu, publikacje z menedżera kolejek w klastrze są kierowane przez menedżer kolejek, w którym jest skonfigurowany obiekt tematu ("host tematu"), a stamtąd do menedżerów kolejek, w których istnieją subskrypcje.

- Polega to na każdym menedżerze kolejek, który informuje wszystkie hosty tematów o każdym temacie, w którym obecnie znajduje się jedna lub większa liczba subskrypcji. Dowolny menedżer kolejek udostępniający subskrypcję ustanawia kanał dla każdego hosta tematu dla tematu, którego dotyczy subskrypcja.
- Menedżery kolejek, które nie są udostępniane w temacie, nie są informowane o innych menedżerach kolejek, które nie są dostępne w temacie, w klastrze w celu publikowania/subskrybowania, a kanały nie są ustanawiane między nimi w tym celu.
- Jeśli aplikacja publikująca jest połączona z menedżerem kolejek udostępniającym temat, opublikowane komunikaty są kierowane bezpośrednio do menedżerów kolejek, w których utworzono zgodne subskrypcje, bez konieczności dodatkowego "przeskoku". Podobnie, jeśli zgodne subskrypcje są tworzone w jedynym menedżerze kolejek udostępniającym temat, komunikaty publikowane w tym temacie są kierowane bezpośrednio do tego menedżera kolejek, bez konieczności dodatkowego przeskoku.
- Subskrypcje w tym samym menedżerze kolejek co publikator są spełnione bez kierowania publikacji do hostów w obiekcie tematu.

Podobnie jak w przypadku kolejek klastrowych, wiele menedżerów kolejek może skonfigurować ten sam obiekt tematu administracyjnego. Zapewnia to wyższą dostępność routingu komunikatów oraz skalowanie poziome poprzez równoważenie obciążenia. W przypadku obiektów tematów kierowanych przez hosty tematów, gdy wiele menedżerów kolejek konfiguruje ten sam nazwany temat dla tej samej gałęzi drzewa tematów, każdy host tematu jest świadomy subskrybowanych tematów przez każdego menedżera kolejek udostępniającego subskrypcję.

- Po opublikowaniu komunikatu jest on wysyłany do jednego z menedżerów kolejek hosta tematu w celu przekazania do subskrypcji menedżerów kolejek udostępnianych przez subskrypcję. Wybór menedżera

kolejek hostów tematów jest zgodny z tymi samymi domyślnymi regułami równoważenia obciążenia, co w przypadku klastrowych kolejek typu punkt z punktem.

- Jeśli menedżer kolejek publikowania nie może nawiązać kontaktu z co najmniej jednym menedżerem kolejek hostów tematów, komunikaty są kierowane do pozostałych dostępnych tematów, w których znajdują się menedżery kolejek udostępniające temat.

Każda publikacja tematu w rozgątkowaniu kierowanym w drzewie tematów jest przekazywana do jednego z hostów tematów, nawet jeśli nie ma żadnych subskrypcji dla tego tematu w dowolnym miejscu w klastrze. Domyślnie komunikaty są wysyłane z tego miejsca tylko do innych menedżerów kolejek w klastrze, które są hostem jednej lub większej liczby subskrypcji.

- Polega to na tym, że każdy menedżer kolejek hostów tematów jest informowany o wszystkich subskrybowanych łańcuchach tematów w każdym menedżerze kolejek w klastrze.
- Znajomość poszczególnych subskrybowanych tematów może zostać usunięta poprzez zmianę na model wysyłania wszystkich publikacji kierowanych do hosta tematu na wszystkich menedżerach kolejek w klastrze, niezależnie od tego, czy mają one subskrypcje. Zmniejsza to ruch wiedzy o subskrypcji, ale może zwiększyć ruch w publikacjach i potencjalnie liczbę kanałów nawiązanych z każdym tematem udostępniającego menedżera kolejek. Więcej informacji zawiera sekcja [Wydajność subskrypcji w sieciach publikowania/subskrypcji](#).

Przeptywy komunikatów publikowania/subskrypcji korzystające z tematów klastrowych kierowanych przez hosty tematów **nie mogą** obejmować wielu klastrów publikowania/subskrypcji przy użyciu hierarchii publikowania/subskrypcji.

Więcej szczegółowych informacji na temat kierowania hostami tematów znajduje się w temacie [Routing hostów tematów w klastrach publikowania/subskrypcji](#).

Hierarchie publikowania/subskrypcji

Hierarchię publikowania/subskrypcji budujesz, łącząc menedżery kolejek ze sobą przy użyciu kanałów, a następnie definiując relację nadrzędny-nadrzędny między parami menedżerów kolejek. Komunikat przepływa od publikatora do subskrypcji za pośrednictwem bezpośrednich relacji w hierarchii. Należy zauważyć, że może to oznaczać wiele "przeskoków", aby się tam dostać.

Tylko jedna kopia komunikatu jest wysyłana między dowolną parą menedżerów kolejek, niezależnie od liczby subskrybentów wiadomości w docelowym menedżerze kolejek. Po przybyciu do menedżera kolejek z jedną lub większą liczbą subskrypcji komunikat jest duplikowany we wszystkich subskrypcjach.

Domyślnie komunikaty są wysyłane tylko do innych menedżerów kolejek w hierarchii, które znajdują się na trasie do subskrypcji w innym menedżerze kolejek:

- Zależy to od każdego menedżera kolejek informującego o wszystkich bezpośrednich relacjach wszystkich tematów, które mają obecnie jedną lub więcej subskrypcji, albo w tym menedżerze kolejek, albo w jednym z jego innych relacji. Powoduje to, że wszystkie menedżery kolejek w hierarchii mają świadomość wszystkich subskrybowanych tematów.
- To zachowanie można zmienić tak, aby zawsze wysyłały publikacje do wszystkich menedżerów kolejek w hierarchii, niezależnie od istniejących subskrypcji. Powoduje to usunięcie konieczności propagowania informacji o subskrypcji w hierarchii, ale może spowodować zwiększenie ruchu publikacyjnego.

Tworząc klaster, należy zadbać o to, aby nie tworzyć pętli powodujących, że komunikaty będą cyklem na zawsze w sieci. Takie pętle nie mogą być tworzone w hierarchii.

Każdy menedżer kolejek musi mieć unikalną nazwę menedżera kolejek.

Przeptywy komunikatów publikowania/subskrypcji mogą obejmować wiele klastrów publikowania/subskrybowania. W tym celu należy dodać jednego menedżera kolejek z każdego klastra do hierarchii publikowania/subskrypcji.

Bardziej szczegółową eksplorację zawiera sekcja [Routing w hierarchiach publikowania/subskrypcji](#).

Subskrypcje proxy w sieci publikowania/subskrypcji

Subskrypcja proxy to subskrypcja utworzona przez jeden menedżer kolejek i dotycząca tematów publikowanych w innym menedżerze kolejek. Subskrypcja proxy przepływa między menedżerami kolejek dla każdego pojedynczego łańcucha tematów, który subskrybuje subskrypcja. Nie można utworzyć subskrypcji proxy jawnie. Menedżer kolejek robi to w imieniu bieżącego użytkownika.

Menedżery kolejek można łączyć ze sobą w klaster publikowania/subskrybowania lub w hierarchię publikowania/subskrypcji. Przepływ subskrypcji proxy między połączonymi menedżerami kolejek. Subskrypcje proxy powodują, że publikacje dotyczące tematu utworzonego przez publikatora połączonego z jednym menedżerem kolejek mogą być odbierane przez subskrybentów tego tematu połączonych z innymi menedżerami kolejek. Patrz [“Rozproszone sieci publikowania/subskrypcji”](#) na stronie 92.

W topologiach publikowania/subskrypcji z wieloma tysiącami subskrypcji dla poszczególnych łańcuchów tematów lub w przypadku, gdy istnienie tych subskrypcji może być szybko zmieniane, należy rozważyć narzut propagacji subskrypcji proxy. Oprócz automatycznego agregowania opisanego w dalszej części tego tematu można wprowadzić ręczne zmiany konfiguracji, które dodatkowo ograniczają przepływ subskrypcji proxy i publikacji między połączonymi menedżerami kolejek, a także zmniejszają opóźnienie oczekiwania na propagację subskrypcji proxy do wszystkich połączonych menedżerów kolejek. Więcej informacji zawiera sekcja [Wydajność subskrypcji w sieciach publikowania/subskrypcji](#).

Subskrypcje proxy nie zawierają żadnych selektorów używanych przez subskrypcje lokalne, a łańcuchy tematów subskrypcji zawierające znaki wieloznaczne mogą być uproszczone. Może to spowodować, że publikacje będą zgodne z subskrypcjami proxy, w przypadku których rzeczywiste subskrypcje nie są dostępne, co powoduje dodatkowy przepływ publikacji między menedżerami kolejek. Menedżer kolejek udostępniający subskrypcje odfiltrowuje takie rozbieżności, aby dodatkowe publikacje nie zostały zwrócone do subskrypcji.

Agregacja subskrypcji proxy

Subskrypcje proxy są agregowane przy użyciu zduplikowanego systemu eliminowania. W przypadku konkretnego rozstrzygniętego łańcucha tematu subskrypcja proxy jest wysyłana w pierwszej subskrypcji lokalnej lub w ramach subskrypcji proxy. Kolejne subskrypcje tego samego łańcucha tematu korzystają z tej istniejącej subskrypcji proxy.

Subskrypcja proxy została anulowana po anulowaniu ostatniej subskrypcji lokalnej lub odebranej subskrypcji proxy.

Agregacja publikacji

Jeśli w menedżerze kolejek istnieje więcej niż jedna subskrypcja tego samego łańcucha tematu, tylko pojedyncza kopia każdej publikacji zgodnej z tym łańcuchem tematu jest wysyłana z innych menedżerów kolejek w topologii publikowania/subskrypcji. Po przybyciu komunikatu, lokalny menedżer kolejek dostarcza kopię komunikatu do każdej zgodnej subskrypcji.

Możliwe jest, aby jedna subskrypcja proxy była zgodna z łańcuchem tematu pojedynczej publikacji, gdy subskrypcje proxy zawierają znaki wieloznaczne. Jeśli komunikat jest publikowany w menedżerze kolejek, który jest zgodny z co najmniej dwoma subskrypcjami proxy utworzonymi przez jeden połączony menedżer kolejek, tylko jedna kopia tej publikacji jest przekazywana do zdalnego menedżera kolejek w celu spełnienia wielu subskrypcji proxy.

Zadania pokrewne

[Wykrywanie pętli w rozproszonej sieci publikowania/subskrypcji](#)

Znaki wieloznaczne w subskrypcjach proxy

W subskrypcjach mogą być używane znaki wieloznaczne w łańcuchach tematów w celu dopasowania do wielu łańcuchów tematów w publikacjach.

Istnieją dwa schematy znaków wieloznacznych, które mogą być używane przez subskrypcję: *oparte na tematach* i *oparte na znakach*. Patrz [“Schematy znaków wieloznacznych”](#) na stronie 71.

W produkcie IBM WebSphere MQ 7.0 i nowszych wersjach wszystkie subskrypcje proxy dla subskrypcji ze znakami wieloznacznymi są przekształcane w celu użycia znaków wieloznacznych opartych na tematach. Jeśli zostanie znaleziony znak wieloznacznym, zostanie on zastąpiony znakiem # , z powrotem do najbliższego / . Na przykład produkt /aaa/bbb/c*d jest przekształcany w produkt /aaa/bbb/#. Wyniki konwersji w menedżerach kolejek zdalnych wysyłają nieco więcej publikacji, niż zostały jawnie zasubskrybowane. Dodatkowe publikacje są filtrowane przez lokalny menedżer kolejek, gdy dostarcza publikacje do lokalnych subskrybentów.

Sterowanie użyciem znaków wieloznacznych przy użyciu właściwości WILDCARD

Użyj właściwości MQSC **Topic WILDCARD** lub równoważnej właściwości PCF Topic `WildcardOperation` , aby sterować dostarczaniem publikacji do aplikacji subskrybentów, które używają znaków wieloznacznych w nazwach łańcuchów tematów. Właściwość WILDCARD może mieć jedną z dwóch możliwych wartości:

WILDCARD

Zachowanie subskrypcji ze znakami wieloznacznymi w odniesieniu do tego tematu.

PASSTHRU

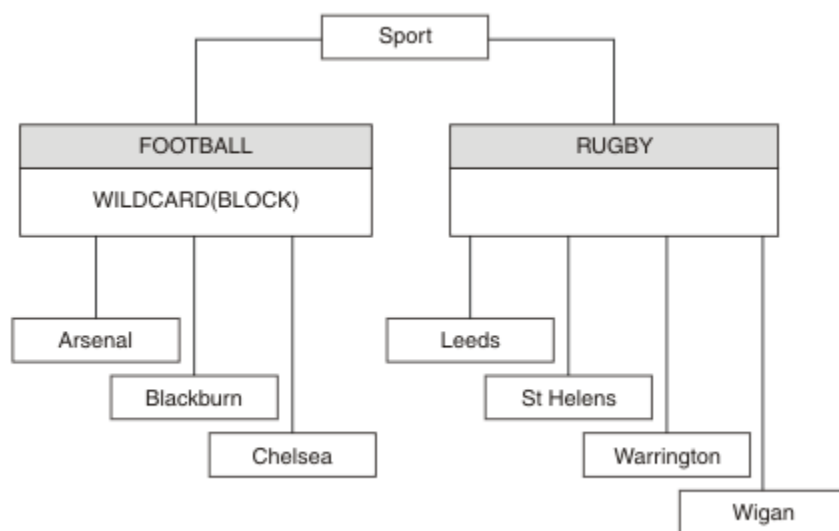
Subskrypcje tematu ze znakami wieloznacznymi, który jest mniej konkretny niż łańcuch tematu w tym obiekcie tematu, otrzymują publikacje zamieszczone w tym temacie i w łańcuchach tematów bardziej konkretnych niż ten temat.

BLOCK

Subskrypcje tematu ze znakami wieloznacznymi, który jest mniej konkretny niż łańcuch tematu w tym obiekcie tematu, nie otrzymują publikacji zamieszczonych w tym temacie i w łańcuchach tematów bardziej konkretnych niż ten temat.

Wartość tego atrybutu jest używana podczas definiowania subskrypcji. Jeśli ten atrybut zostanie zmieniony, modyfikacja nie będzie mieć wpływu na zestaw tematów objętych istniejącymi subskrypcjami. Ten scenariusz dotyczy również sytuacji, w której topologia jest zmieniana podczas tworzenia lub usuwania obiektów tematu. Zestaw tematów zgodnych z subskrypcjami utworzonymi po zmodyfikowaniu atrybutu WILDCARD jest tworzony przy użyciu zmodyfikowanej topologii. Aby wymusić ponowną ocenę zgodnego zestawu tematów pod kątem istniejących subskrypcji, należy zrestartować menedżer kolejek.

W tym przykładzie “Przykład: tworzenie klastra publikowania/subskrypcji produktu Sport” na stronie 85 można wykonać kroki, aby utworzyć strukturę drzewa tematów, która jest wyświetlana w produkcie Rysunek 23 na stronie 82.



Rysunek 37. Drzewo tematów, które korzysta z właściwości WILDCARD , BLOCK

Subskrybent korzystający z łańcucha tematu ze znakami wieloznacznymi # otrzymuje wszystkie publikacje do tematu Sport i poddrzewa Sport/Rugby . Subskrybent nie otrzymuje żadnych publikacji do poddrzewa Sport/Football , ponieważ wartość właściwości WILDCARD w temacie Sport/Football to BLOCK.

PASSTHRU jest ustawieniem domyślnym. Wartość właściwości WILDCARD PASSTHRU można ustawić na węzły w drzewie Sport . Jeśli w węzłach nie ma wartości właściwości WILDCARD BLOCK, ustawienie PASSTHRU nie powoduje zmiany zachowania obserwowanego przez subskrybentów węzłów w drzewie Sports .

W tym przykładzie utwórz subskrypcje, aby sprawdzić, w jaki sposób ustawienie znaku wieloznacznego wpływa na dostarczane publikacje; patrz Rysunek 27 na stronie 87. Aby utworzyć niektóre publikacje, należy uruchomić komendę publikowania w programie Rysunek 30 na stronie 88 .

pub QMA

Rysunek 38. Publikuj w QMA

Wyniki są wyświetlane w programie Tabela 3 na stronie 82. Należy zwrócić uwagę na to, w jaki sposób ustawienie wartości właściwości WILDCARD BLOCK uniemożliwia subskrypcje z użyciem znaków wieloznacznych podczas odbierania publikacji do tematów w zasięgu znaku wieloznacznego.

| Tabela 6. Publikacje otrzymane w systemie QMA | | | |
|---|------------------|--|--|
| Subskrypcja | Łańcuch tematu | Otrzymane publikacje | Uwagi |
| SPORTS | Sports/# | Sports Sports/Rugby Sports/Rugby/Leeds | Wszystkie publikacje dotyczące poddrzewa piłkarskiego zablokowane przez produkt WILDCARD (BLOCK) w systemie Sports/Football |
| SARSENAL | Sports/#/Arsenal | - | Produkt WILDCARD (BLOCK) w systemie Sports/Football uniemożliwia subskrypcję zastępcze w systemie Arsenal |
| SLEEDS | Sports/#/Leeds | Sports/Rugby/Leeds | Wartość domyślna WILDCARD w programie Sports/Rugby nie uniemożliwia subskrypcji ze znakami wieloznacznymi w produkcie Leeds. |

Uwaga:

Założmy, że subskrypcja ma znak wieloznaczny, który jest zgodny z obiektem tematu z wartością właściwości WILDCARD BLOCK. Jeśli subskrypcja ma również łańcuch tematu z prawej strony znaku wieloznacznego zgodnego ze zgodnym znakiem wieloznacznym, subskrypcja nigdy nie otrzyma publikacji. Zestaw publikacji, które nie są blokowane, to publikacje na tematy, które są elementami nadrzędnymi zablokowanego znaku wieloznacznego. Publikacje na tematy, które są elementami podrzędnymi tematu z wartością właściwości BLOCK , są blokowane przez znak wieloznaczny. Dlatego łańcuchy tematów subskrypcji, które zawierają temat po prawej stronie znaku wieloznacznego, nigdy nie otrzymują żadnych publikacji do dopasowania.

Ustawienie wartości właściwości WILDCARD na wartość BLOCK nie oznacza, że nie można zasubskrybować łańcucha tematu zawierającego znaki wieloznaczne. Taka subskrypcja jest normalna. Subskrypcja zawiera jawny temat, który jest zgodny z tematem z obiektem tematu, który ma wartość właściwości WILDCARD BLOCK. W przypadku tematów, które są elementami nadrzędnymi lub elementami podrzędnymi tematu z wartością właściwości WILDCARD BLOCK, używane są znaki wieloznaczne. W przykładzie w produkcie Rysunek 23 na stronie 82 subskrypcja, taka jak Sports/Football/# , może odbierać publikacje.

Znaki wieloznaczne i tematy dotyczące klastrów

Definicje tematów klastra są propagowane do każdego menedżera kolejek w klastrze. Subskrypcja tematu klastra w jednym menedżerze kolejek w wyniku klastra powoduje utworzenie subskrypcji proxy w menedżerze kolejek. Subskrypcja proxy jest tworzona w każdym innym menedżerze kolejek w klastrze. Subskrypcje za pomocą łańcuchów tematów zawierających znaki wieloznaczne, w połączeniu z tematami klastra, mogą być trudne do przewidzenia. Zachowanie jest wyjaśnione w poniższym przykładzie.

W klastrze ustawionym na przykład “Przykład: tworzenie klastra publikowania/subskrypcji produktu Sport” na stronie 85, QMB ma ten sam zestaw subskrypcji, co QMA, ale produkt QMB nie otrzymał żadnych publikacji po opublikowaniu publikatora w produkcie QMA, patrz Rysunek 24 na stronie 82. Chociaż tematy Sports/Football i Sports/Rugby są tematami klastra, subskrypcje zdefiniowane w pliku fullsubs.tst nie odwołują się do tematu klastra. Żadne subskrypcje proxy nie są propagowane z produktu QMB do produktu QMA. Bez subskrypcji proxy żadne publikacje do produktu QMA nie są przekazywane do produktu QMB.

Niektóre subskrypcje, takie jak Sports/#/Leeds, mogą się wydawać, że odwołują się do tematu klastra, w tym przypadku Sports/Rugby. Subskrypcja Sports/#/Leeds jest rzeczywiście tłumaczona na obiekt tematu SYSTEM.BASE.TOPIC.

Reguła rozstrzygania obiektu tematu, do którego odwołuje się subskrypcja, taka jak Sports/#/Leeds jest następująca. Obetnij łańcuch tematu na pierwszy znak wieloznaczny. Skanowanie w lewo za pomocą łańcucha tematu w poszukiwaniu pierwszego tematu, który ma powiązany obiekt tematu administracyjnego. Obiekt tematu może określać nazwę klastra lub zdefiniować lokalny obiekt tematu. W przykładzie Sports/#/Leeds łańcuch tematu po obcięciu to Sports, który nie ma obiektu tematu, a więc Sports/#/Leeds dziedziczy z SYSTEM.BASE.TOPIC, który jest lokalnym obiektem tematu.

Aby zobaczyć, w jaki sposób subskrybujące tematy w klastrze mogą zmienić sposób działania propagacji znaków wieloznacznych, należy uruchomić skrypt wsadowy [upsubs.bat](#). Skrypt kasuje kolejki subskrypcji i dodaje subskrypcje tematów klastra w pliku fullsubs.tst. Uruchom ponownie plik [puba.bat](#), aby utworzyć zadanie wsadowe publikacji; patrz Rysunek 24 na stronie 82.

Program Tabela 4 na stronie 84 pokazuje wynik dodawania dwóch nowych subskrypcji do tego samego menedżera kolejek, w którym publikowane były publikacje. Wynik jest taki, jak oczekiwano, nowe subskrypcje otrzymują jedną publikację, a liczba publikacji odebranych przez inne subskrypcje nie ulega zmianie. Nieoczekiwane wyniki są wyświetlane w drugim menedżerze kolejek klastra. Patrz Tabela 5 na stronie 84.

| Tabela 7. Publikacje otrzymane w systemie QMA | | | |
|---|-----------------------------|--|--|
| Subskrypcja | Łańcuch tematu | Otrzymane publikacje | Uwagi |
| SPORTS | Sports/# | Sports Sports/Rugby Sports/Rugby/Leeds | Wszystkie publikacje dotyczące poddrzewa piłkarskiego zablokowane przez produkt WILDCARD (BLOCK) w systemie Sports/Football |
| SARSENAL | Sports/#/Arsenal | - | Produkt WILDCARD (BLOCK) w systemie Sports/Football uniemożliwia subskrypcję zastępcze w systemie Arsenal |
| SLEEDS | Sports/#/Leeds | Sports/Rugby/Leeds | Wartość domyślna WILDCARD w programie Sports/Rugby nie uniemożliwia subskrypcji ze znakami wieloznacznymi w produkcie Leeds. |
| FARSENAL | Sports/Football/ Arsenal | Sports/Football/ Arsenal | Produkt Arsenal otrzymuje publikację, ponieważ subskrypcja nie ma znaku wieloznacznego. |

| <i>Tabela 7. Publikacje otrzymane w systemie QMA (kontynuacja)</i> | | | |
|--|-----------------------|-----------------------------|--|
| Subskrypcja | Łańcuch tematu | Otrzymane publikacje | Uwagi |
| FLEEDS | Sports/Rugby/Leeds | Sports/Rugby/Leeds | Produkt Leeds otrzyma publikację w dowolnym zdarzeniu. |

W programie Tabela 5 na stronie 84 wyświetlane są wyniki dodawania dwóch nowych subskrypcji w produkcie QMB i publikowania w produkcie QMA. Należy pamiętać, że produkt QMB nie otrzymał publikacji bez tych dwóch nowych subskrypcji. Zgodnie z oczekiwaniami dwie nowe subskrypcje otrzymują publikacje, ponieważ Sports/FootBall i Sports/Rugby są tematami klastra. Produkt QMB przekazał subskrypcje proxy dla produktów Sports/Football/Arsenal i Sports/Rugby/Leeds do produktu QMA, które następnie wysyłały publikacje do produktu QMB.

Nieoczekiwany wynik polega na tym, że dwie subskrypcje Sports/# i Sports/#/Leeds, które wcześniej nie otrzymały publikacji, otrzymują teraz publikacje. Jest to spowodowane tym, że publikacje Sports/Football/Arsenal i Sports/Rugby/Leeds przekazane do produktu QMB dla innych subskrypcji są teraz dostępne dla każdego subskrybenta dołączonego do produktu QMB. W związku z tym subskrypcje tematów lokalnych Sports/# i Sports/#/Leeds otrzymują publikację Sports/Rugby/Leeds. Produkt Sports/#/Arsenal nie otrzymuje publikacji, ponieważ wartość właściwości WILDCARD ustawiona na wartość BLOCK jest ustawiona na wartość właściwości Sport/Football.

| <i>Tabela 8. Publikacje otrzymane w systemie QMB</i> | | | |
|--|-------------------------|-----------------------------|--|
| Subskrypcja | Łańcuch tematu | Otrzymane publikacje | Uwagi |
| SPORTS | Sports/# | Sports/Rugby/Leeds | Wszystkie publikacje dotyczące poddrzewa piłkarskiego zablokowanego przez produkt WILDCARD (BLOCK) w systemie Sports/Football |
| SARSENAL | Sports/#/Arsenal | - | Produkt WILDCARD (BLOCK) w systemie Sports/Football uniemożliwia subskrypcję zastępcze w systemie Arsenal |
| SLEEDS | Sports/#/Leeds | Sports/Rugby/Leeds | Wartość domyślna WILDCARD w środowisku sportowym/Rugby nie uniemożliwia subskrypcji ze znakami wieloznacznymi w produkcie Leeds. |
| FARSENAL | Sports/Football/Arsenal | Sports/Football/Arsenal | Produkt Arsenal otrzymuje publikację, ponieważ subskrypcja nie ma znaku wieloznacznego. |
| FLEEDS | Sports/Rugby/Leeds | Sports/Rugby/Leeds | Produkt Leeds otrzyma publikację w dowolnym zdarzeniu. |

W większości aplikacji jest ona niepożądana, aby jedna subskrypcja miała wpływ na zachowanie innej subskrypcji. Ważne użycie właściwości WILDCARD z wartością BLOCK polega na tym, że subskrypcje tego samego łańcucha tematu zawierające znaki wieloznaczne są jednakowo zachowane. Niezależnie od tego, czy subskrypcja znajduje się w tym samym menedżerze kolejek, co publikator, czy też inna, wyniki subskrypcji są takie same.

Znaki wieloznaczne i strumienie

W przypadku nowej aplikacji napisanej do interfejsu API publikowania/subskrypcji efekt jest taki, że subskrypcja produktu * nie odbiera żadnych publikacji. Aby otrzymywać wszystkie publikacje dotyczące

sportu, należy zasubskrybować produkty `Sports/*` lub `Sports/#`, podobnie jak w przypadku publikacji `Business`.

Zachowanie istniejącej w kolejce aplikacji publikowania/subskrypcji nie zmienia się, gdy broker publikowania/subskrybowania jest migrowany do wersji IBM WebSphere MQ 7 i nowszych. Właściwość **StreamName** w komendach **Publish**, **Register Publisher** lub **Subscriber** jest odwzorowana na nazwę tematu, do którego migrowany jest strumień.

Znaki wieloznaczne i punkty subskrypcji

W przypadku nowej aplikacji napisanej w interfejsie API publikowania/subskrypcji efekt migracji jest taki, że subskrypcja produktu `*` nie odbiera żadnych publikacji. Aby otrzymywać wszystkie publikacje dotyczące sportu, należy zasubskrybować produkty `Sports/*` lub `Sports/#`, podobnie jak w przypadku publikacji `Business`.

Zachowanie istniejącej w kolejce aplikacji publikowania/subskrypcji nie zmienia się, gdy broker publikowania/subskrybowania jest migrowany do wersji IBM WebSphere MQ 7 i nowszych. Właściwość **SubPoint** w komendach **Publish**, **Register Publisher** lub **Subscriber** jest odwzorowana na nazwę tematu, do którego migrowana została subskrypcja.

Przykład: tworzenie klastra publikowania/subskrypcji produktu Sport

Kroki, które należy wykonać, tworzą klaster CL1z czterema menedżerami kolejek: dwoma pełnymi repozytoriami, CL1A i CL1B oraz dwoma częściowymi repozytoriami, QMA i QMB. Pełne repozytoria są używane do przechowywania tylko definicji klastrów. QMA jest oznaczony jako host tematu klastra. Trwałe subskrypcje są definiowane na serwerze QMA i QMB.

Uwaga: Przykład jest kodowany dla produktu Windows. Należy ponownie utworzyć plik [Create qmgrs.bat](#) i [create pub.bat](#), aby skonfigurować i przetestować przykład na innych platformach.

1. Utwórz pliki skryptowe.
 - a. [Utwórz plik topics.tst](#)
 - b. [Utwórz element wildsubs.tst](#)
 - c. [Utwórz fullsubs.tst](#)
 - d. [Tworzenie pliku qmgrs.bat](#)
 - e. [utwórz plik pub.bat](#)
2. Uruchom komendę [Create qmgrs.bat](#), aby utworzyć konfigurację.

```
qmgrs
```

Utwórz tematy w programie [Rysunek 23 na stronie 82](#). Skrypt na rysunku 5 tworzy tematy klastra `Sports/Football` i `Sports/Rugby`.

Uwaga: Opcja REPLACE nie zastępuje właściwości TOPICSTR tematu. TOPICSTR to właściwość, która jest użytecznie udostępniona w przykładzie w celu przetestowania różnych drzew tematów. Aby zmienić tematy, należy najpierw usunąć ten temat.


```

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

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

```

Rysunek 39. Usuwanie i tworzenie tematów: topics.tst

Uwaga: Usuń tematy, ponieważ produkt REPLACE nie zastępuje łańcuchów tematów.

Utwórz subskrypcje z użyciem znaków wieloznacznych. Znaki wieloznaczne odpowiadające tematyce z obiektami tematu w programie [Rysunek 23 na stronie 82](#). Utwórz kolejkę dla każdej subskrypcji. Kolejki są kasowane, a subskrypcje są usuwane po uruchomieniu skryptu lub ponownym uruchomieniu.

Uwaga: Opcja REPLACE nie zastępuje właściwości TOPICOBJ ani TOPICSTR subskrypcji. TOPICOBJ lub TOPICSTR to właściwości, które są przydatne na przykład w celu przetestowania różnych subskrypcji. Aby je zmienić, najpierw usuń subskrypcję.

```

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

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

```

Rysunek 40. Utwórz subskrypcje ze znakami wieloznacznymi: wildsubs.tst

Utwórz subskrypcje, które odwołują się do obiektów tematu klastra.

Uwaga:

Separator, /, jest automatycznie wstawiany między łańcuchem tematu, do którego odwołuje się TOPICOBJ, a łańcuchem tematu zdefiniowanym przez parametr TOPICSTR.

Definicja DEFINE SUB(FARSENAL) TOPICSTR('Sports/Football/Arsenal') DEST(QFARSENAL) tworzy tę samą subskrypcję. TOPICOBJ jest używany jako szybki sposób odniesienia do już zdefiniowanego łańcucha tematu. Jeśli subskrypcja została utworzona, nie odnosi się już do obiektu tematu.

```

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

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

```

Rysunek 41. Usuń i utwórz subskrypcje: fullsubs.tst

Utwórz klaster z dwoma repozytoriami. Utwórz dwa częściowe repozytoria dla publikowania i subskrybowania. Uruchom skrypt ponownie, aby usunąć wszystko i rozpocząć ponownie. Skrypt tworzy również hierarchię tematów oraz początkowe subskrypcje ze znakami wieloznacznymi.

Uwaga:

Na innych platformach napisz podobny skrypt lub wpisz wszystkie komendy. Użycie skryptu pozwala na szybkie usuwanie wszystkiego i uruchamianie za pomocą identycznej konfiguracji.

```

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

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

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

```

Rysunek 42. Tworzenie menedżerów kolejek: qmgrs.bat

Zaktualizuj konfigurację, dodając subskrypcje tematów klastra.

```

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

```

Rysunek 43. Aktualizacja subskrypcji: upsubs.bat

Uruchom program pub.bat z menedżerem kolejek jako parametrem, aby opublikować komunikaty zawierające łańcuch tematu publikacji. Program Pub.bat korzysta z przykładowego programu **amqspub**.

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

Rysunek 44. Opublikuj: pub.bat

Pojęcia pokrewne

Subskrypcje wieloznaczne i zachowane publikacje

Zasięg publikacji

Podczas konfigurowania klastra lub hierarchii publikowania/subskrybowania zakres publikacji dodatkowo określa, czy menedżery kolejek przesyłają publikację do menedżerów kolejek zdalnych. Za pomocą atrybutu tematu **PUBSCOPE** można administrować zakresem publikacji.

Jeśli publikacja nie jest przekazywana do zdalnych menedżerów kolejek, publikowanie jest odbierane tylko subskrybentom lokalnym.

Jeśli używany jest klaster publikowania/subskrypcji, zakres publikacji jest kontrolowany przede wszystkim przez definicję obiektów tematu w klastrze w pewnych punktach drzewa tematów. Zasięg publikacji musi być ustawiony w taki sposób, aby umożliwić przepływ publikacji do innych menedżerów kolejek w klastrze. Zasięg publikowania należy ograniczyć tylko w przypadku tematu klastrowego, gdy wymagana jest kontrola szczegółowego zakresu publikacji dotyczących konkretnych tematów w niektórych menedżerach kolejek.

Jeśli używana jest hierarchia publikowania/subskrypcji, zakres publikacji jest sterowany głównie przez ten atrybut w połączeniu z atrybutem Zasięg subskrypcji.

Atrybut **PUBSCOPE** służy do określania zasięgu publikacji wykonanych w konkretnym temacie. Atrybut można ustawić na jedną z następujących wartości:

QMGR

Publikacja jest dostarczana tylko do lokalnych subskrybentów. Publikacje te są nazywane *publikacjami lokalnymi*. Publikacje lokalne nie są przekazywane do zdalnych menedżerów kolejek i dlatego nie są odbierane przez subskrybentów połączonych ze zdalnymi menedżerami kolejek.

ALL

Publikacja jest dostarczana do lokalnych subskrybentów i subskrybentów połączonych ze zdalnymi menedżerami kolejek w klastrze publikowania/subskrypcji lub hierarchii. Te publikacje są nazywane *publikacjami globalnymi*.

ASPARENT

Użyj ustawienia **PUBSCOPE** tematu nadrzędnego w drzewie tematów.

Publikatory mogą również określać, czy publikacja jest lokalna, czy globalna, przy użyciu opcji put komunikatu produktu MQPMO_SCOPE_QMGR. Jeśli ta opcja jest używana, przestania ona wszystkie zachowania ustawione za pomocą atrybutu tematu **PUBSCOPE**.

Pojęcia pokrewne

“Administracyjne obiekty tematów” na stronie 78

Korzystając z obiektu tematu administracyjnego, można przypisać konkretne, inne niż domyślne atrybuty do tematów.

Zadania pokrewne

Konfigurowanie rozproszonych sieci publikowania/subskrypcji

Zasięg subskrypcji

Zasięg subskrypcji określa, czy subskrypcja jednego menedżera kolejek odbiera publikacje, które są publikowane w innym menedżerze kolejek w klastrze lub w hierarchii publikowania/subskrypcji, czy tylko publikacje pochodzące od lokalnych publikatorów.

Ograniczenie zasięgu subskrypcji do menedżera kolejek powoduje, że subskrypcje proxy nie są przekazywane do innych menedżerów kolejek w topologii publikowania/subskrypcji. Powoduje to zmniejszenie ruchu przesyłania komunikatów w trybie publikowania/subskrypcji menedżera kolejek między kolejkami.

Jeśli używany jest klaster publikowania/subskrypcji, zasięg subskrypcji jest kontrolowany przede wszystkim przez definicję obiektów tematu w klastrze w pewnych punktach drzewa tematów. Zasięg subskrypcji musi być ustawiony w taki sposób, aby możliwe było przepływanie subskrypcji proxy do innych menedżerów kolejek w klastrze. Zasięg subskrypcji dla tematu klastrowego należy ograniczyć tylko wtedy, gdy wymagana jest kontrola szczegółowego tematu dotyczącego konkretnych tematów w niektórych menedżerach kolejek.

Jeśli używana jest hierarchia publikowania/subskrypcji, zasięg subskrypcji jest kontrolowany głównie przez ten atrybut w połączeniu z atrybutem Zasięg publikacji.

Atrybut tematu **SUBSCOPE** jest używany do określania zasięgu subskrypcji, które zostały wykonane w konkretnym temacie. Atrybut można ustawić na jedną z następujących wartości:

QMGR

Subskrypcja odbiera tylko publikacje lokalne, a subskrypcje proxy nie są propagowane do menedżerów kolejek zdalnych.

ALL

Subskrypcja proxy jest propagowana do menedżerów kolejek zdalnych w klastrze lub hierarchii publikowania/subskrypcji, a subskrybent otrzymuje publikacje lokalne i zdalne.

ASPARENT

Użyj ustawienia **SUBSCOPE** tematu nadrzędnego w drzewie tematów.

Jeśli zasięg subskrypcji dla tematu jest ustawiony na wartość WSZYSTKIE, bezpośrednio lub rozstrzygnięte za pomocą elementu ASPARENT, poszczególne subskrypcje tego tematu mogą ograniczyć zasięg do QMGR, określając wartość MQSO_SCOPE_QMGR podczas tworzenia subskrypcji. Subskrypcja tematu, który ma zasięg QMGR, nie może rozszerzyć zasięgu na ALL.

Pojęcia pokrewne

“Administracyjne obiekty tematów” na stronie 78

Korzystając z obiektu tematu administracyjnego, można przypisać konkretne, inne niż domyślne atrybuty do tematów.

Zadania pokrewne

Konfigurowanie rozproszonych sieci publikowania/subskrypcji

Obszary tematów

Obszar tematu to zestaw tematów, które można subskrybować i publikować. Menedżer kolejek w rozproszonej topologii publikowania/subskrypcji zawiera obszar tematu, który potencjalnie obejmuje tematy, które zostały zasubskrybowane i opublikowane w połączonych menedżerach kolejek w tej topologii.

Uwaga: Przegląd tematów w menedżerze kolejek, takich jak administracyjne obiekty tematów, łańcuchy tematów i drzewa tematów, zawiera sekcja “Tematy” na stronie 69. Dalsze odwołania do *tematów* w bieżącym artykule odnoszą się do *łańcuchów tematów*, o ile nie określono inaczej.

Tematy są początkowo tworzone w jeden z następujących sposobów:

- pod względem administracyjnym, podczas definiowania obiektu tematu lub trwałej subskrypcji.
- dynamicznie, gdy aplikacja dynamicznie tworzy publikację lub subskrypcję na nowy temat.

Tematy są propagowane do innych menedżerów kolejek zarówno za pośrednictwem subskrypcji proxy, jak i przez tworzenie obiektów tematu klastra administracyjnego. Subskrypcje proxy powodują, że publikacje są przekazywane z menedżera kolejek, z którym jest połączony publikator, do menedżerów kolejek subskrybentów.

Subskrypcje proxy są propagowane między wszystkimi menedżerami kolejek, które są połączone ze sobą przez relacje nadrzędny-podrzędny w hierarchii menedżera kolejek. W wyniku tego można

zasubskrybować jeden menedżer kolejek do tematu zdefiniowanego w dowolnym innym menedżerze kolejek w hierarchii. Jeśli istnieje połączona ścieżka między menedżerami kolejek, nie ma znaczenia, w jaki sposób menedżery kolejek są połączone.

Subskrypcje proxy są również propagowane w przypadku subskrypcji tematów klastra w klastrze publikowania/subskrypcji. Temat klastra to temat, który jest dołączony do obiektu tematu, który zawiera atrybut **CLUSTER**, lub dziedziczny atrybut z jego elementu nadrzędnego. Tematy, które nie są tematami klastra, są znane jako tematy lokalne i nie są replikowane do klastra. Żadne subskrypcje proxy nie są propagowane do klastra z subskrypcji do tematów lokalnych.

Aby podsumować, subskrypcje proxy są tworzone dla subskrybentów w dwóch okolicznościach.

1. Menedżer kolejek jest elementem hierarchii, a subskrypcja proxy jest przekazywana do elementu nadrzędnego i podrzędnych menedżera kolejek.
2. Menedżer kolejek jest elementem klastra, a łańcuch tematu subskrypcji jest tłumaczone na temat powiązany z obiektem tematu klastra. Jeśli temat jest *bezpośrednim kierowanym* tematem klastra, subskrypcje proxy są przekazywane do wszystkich elementów klastra. Jeśli temat jest tematem klastra *kierowanego przez hosta tematu*, subskrypcje proxy są przekazywane tylko do menedżerów kolejek w klastrze, które zdefiniowały klastrowy obiekt tematu. Więcej informacji na ten temat zawiera sekcja [“Klustry publikowania/subskrypcji”](#) na stronie 95.

Jeśli menedżer kolejek jest elementem klastra i hierarchii, subskrypcje proxy są propagowane przez oba mechanizmy bez dostarczania zduplikowanych publikacji do subskrybenta.

Obszary tematów z trzema topologiami publikowania/subskrypcji są opisane na poniższej liście:

- [“Przypadek 1. Klustry publikowania/subskrypcji”](#) na stronie 109.
- [“Przypadek 2. Hierarchie publikowania/subskrypcji w wersji 7 lub nowszej”](#) na stronie 110.
- [“Przypadek 3. Hierarchie publikowania/subskrypcji i strumienie w produkcie IBM WebSphere MQ 6”](#) na stronie 110.

W oddzielnych tematach opisane poniżej zadania konfiguracyjne opisują sposób łączenia obszarów tematów.

- [Tworzenie pojedynczego obszaru tematu w klastrze publikowania/subskrypcji.](#)
- [Dodawanie menedżera kolejek produktu IBM WebSphere MQ 7 lub nowszego do istniejących obszarów tematów produktu IBM WebSphere MQ 6.](#)
- [Łączenie obszarów tematów z wieloma klastrami.](#)
- [Łączenie i izolowanie obszarów tematów w wielu klastrach.](#)
- [Publikowanie i subskrybowanie obszarów tematów w wielu klastrach.](#)

Przypadek 1. Klustry publikowania/subskrypcji

W tym przykładzie założono, że menedżer kolejek nie jest połączony z hierarchią publikowania/subskrypcji.

Jeśli menedżer kolejek jest elementem klastra publikowania/subskrybowania, jego obszar tematu jest składany z tematów lokalnych i tematów klastra. Tematy lokalne są powiązane z obiektami tematów bez atrybutu **CLUSTER**. Jeśli menedżer kolejek ma definicje lokalnego obiektu tematu, jego obszar tematu różni się od innego menedżera kolejek w klastrze, który ma również własne obiekty tematu zdefiniowane lokalnie.

W klastrze publikowania/subskrybowania nie można zasubskrybować tematu zdefiniowanego w innym menedżerze kolejek, chyba że temat, który subskrybuje ten temat, jest tłumaczona na obiekt tematu klastra.

Jeśli te same nazwane definicje obiektu tematu klastra są wymagane w wielu menedżerach kolejek, na przykład w przypadku korzystania z *routingu hosta tematów*, ważne jest, aby wszystkie definicje były zgodne, jeśli jest to konieczne. Więcej informacji na ten temat zawiera sekcja [Tworzenie pojedynczego obszaru tematu w klastrze publikowania/subskrypcji](#).

Lokalna definicja obiektu tematu, bez względu na to, czy definicja jest dla tematu klastra, czy tematu lokalnego, ma pierwszeństwo przed tym samym obiektem tematu zdefiniowanym w innym miejscu w klastrze. Używany jest temat lokalnie zdefiniowany, nawet jeśli obiekt zdefiniowany w innym miejscu jest bardziej aktualny.

Ważne jest, aby obiekt tematu klastra był powiązany z tym samym łańcuchem tematu we wszystkich miejscach w klastrze. Nie można modyfikować łańcucha tematu, z którym powiązany jest obiekt tematu. Aby powiązać ten sam obiekt tematu z innym łańcuchem tematu, należy usunąć obiekt tematu i utworzyć go ponownie przy użyciu nowego łańcucha tematu. Jeśli temat jest zgrupowany, efektem jest usunięcie kopii obiektu tematu zapisanego na innych elementach klastra, a następnie utworzenie kopii nowego obiektu tematu w całym klastrze. Kopie obiektu tematu odnoszą się do tego samego łańcucha tematu.

Możliwe jest przypadkowe utworzenie dwóch definicji tego samego nazwanego obiektu tematu w różnych menedżerach kolejek w klastrze, z różnymi łańcuchami tematów. Może to powodować mylące zachowanie, ponieważ wiele definicji tego samego obiektu tematu z różnymi łańcuchami tematu może generować różne wyniki w zależności od sposobu i miejsca, do którego dany temat jest przywoływany. Więcej informacji na temat tego ważnego punktu zawiera sekcja [Wiele definicji tematów klastra o tej samej nazwie](#).

Przypadek 2. Hierarchie publikowania/subskrypcji w wersji 7 lub nowszej

W tym przykładzie założono, że menedżer kolejek nie jest elementem klastra publikowania/subskrypcji.

W produkcie IBM WebSphere MQ 7 lub nowszym, jeśli menedżer kolejek jest elementem hierarchii publikowania/subskrybowania, jego obszar tematu obejmuje wszystkie tematy zdefiniowane lokalnie i w połączonych menedżerach kolejek. Obszar tematu dla wszystkich menedżerów kolejek w hierarchii jest taki sam. Nie ma podziału tematów na tematy lokalne i tematy globalne.

Ustaw opcje **PUBSCOPE** i **SUBSCOPE** na wartość QMGR, aby zapobiec publikacji w temacie przepływaącym od publikatora do subskrybenta połączonego z różnymi menedżerami kolejek w hierarchii.

Założmy, że definiujesz obiekt tematu A1abama z łańcuchem tematu USA/A1abama w menedżerze kolejek QMA. Wynik jest następujący:

1. Obszar tematu w QMA zawiera teraz obiekt tematu A1abama i łańcuch tematu USA/A1abama.
2. Aplikacja lub administrator może utworzyć subskrypcję w QMA przy użyciu nazwy obiektu tematu A1abama.
3. Aplikacja może utworzyć subskrypcję dla dowolnego tematu, w tym produktu USA/A1abama, w dowolnym menedżerze kolejek w hierarchii. Jeśli wartość QMA nie została zdefiniowana lokalnie, temat USA/A1abama jest tłumaczona na obiekt tematu SYSTEM.BASE.TOPIC.

Przypadek 3. Hierarchie publikowania/subskrypcji i strumienie w produkcie IBM WebSphere MQ 6

Przed IBM WebSphere MQ 7obszar tematu został podzielony na osobne strumienie, które obejmowały strumień domyślny, który był obecny we wszystkich menedżerach kolejek. Publikacje nie mogą przepływać między różnymi strumieniami. Jeśli używane są nazwane strumienie, obszary tematów w różnych menedżerach kolejek mogą być różne. Tematy są podzielone na tematy w strumieniu domyślnym, a tematy w różnych nazwanych strumieniach.

Uwaga: Każdy nazwany strumień tworzy osobny obszar tematu. Aby utworzyć topologię połączonej, każdy nazwany strumień musi istnieć w połączonych menedżerach kolejek. Założmy, że strumień X jest zdefiniowany w systemach QMA i QMC, ale nie na QMB. Jeśli QMA jest elementem nadrzędnym QMB, a QMB jest elementem nadrzędnym QMC, żadne tematy w strumieniu X nie mogą przepływać między QMA i QMC.

Ustawienie zarówno opcji **PUBSCOPE**, jak i opcji **SUBSCOPE** na wartość QMGR lub ALL wymaga, aby publikator i subskrybent w temacie wymieniali tylko publikacje dotyczące lokalnej konsumpcji lub wymieniali tylko publikacje na potrzeby globalnej konsumpcji.

Strumienie produktu IBM WebSphere MQ 7nie są dostępne przy użyciu interfejsu API publikowania/subskrypcji. Jeśli użytkownik korzysta z kolejki publikowania/subskrypcji w menedżerze kolejek produktu IBM WebSphere MQ 7, strumienie są odwzorowywane na różne obiekty tematów, które mogą symulować efekt strumieni. Strumień jest symulowany przez utworzenie obiektu tematu, który jest głównym tematem

dla wszystkich tematów w strumieniu. Menedżer kolejek odwzorowuje publikacje i subskrypcje między strumieniem a odpowiadając im głównym tematem każdego drzewa.

Pojęcia pokrewne

“Zasięg publikacji” na stronie 107

Podczas konfigurowania klastra lub hierarchii publikowania/subskrybowania zakres publikacji dodatkowo określa, czy menedżery kolejek przesyłają publikację do menedżerów kolejek zdalnych. Za pomocą atrybutu tematu **PUBSCOPE** można administrować zakresem publikacji.

“Zasięg subskrypcji” na stronie 107

Zasięg subskrypcji określa, czy subskrypcja jednego menedżera kolejek odbiera publikacje, które są publikowane w innym menedżerze kolejek w klastrze lub w hierarchii publikowania/subskrypcji, czy tylko publikacje pochodzące od lokalnych publikatorów.

Zadania pokrewne

Konfigurowanie rozproszonych sieci publikowania/subskrypcji

IBM MQ Rozsyłanie grupowe

Funkcja rozsyłania grupowego w produkcie IBM MQ umożliwia niezawodne rozsyłanie grupowe komunikatów z zachowaniem małych opóźnień i wysokiego zwielokrotnienia.

Rozsyłanie grupowe jest efektywną formą przesyłania komunikatów w trybie publikowania/subskrypcji, ponieważ może być skalowane do dużej liczby subskrybentów bez szkodliwych efektów w działaniu. Produkt IBM MQ umożliwia niezawodne przesyłanie komunikatów oparte na rozsyłaniu grupowym przez korzystanie z potwierdzeń, potwierdzeń negatywnych i numerów kolejnych przy zachowaniu małych opóźnień i wysokiego zwielokrotnienia.

Funkcja rozsyłania grupowego z bezstronnym dostarczaniem produktu IBM MQ umożliwia prawie równoczesne dostarczenie przy zapewnieniu, że żaden z odbiorców nie ma przewagi. Ponieważ funkcja rozsyłania grupowego produktu IBM MQ używa sieci do dostarczania komunikatów, mechanizm publikowania/subskrypcji nie jest potrzebny do zwielokrotnienia danych. Po odwzorowaniu tematu na adres grupy nie ma potrzeby obsługi menedżera kolejek, ponieważ publikatorzy i subskrybenci mogą działać w trybie równorzędym. Pozwala to na zmniejszenie obciążenia serwerów menedżera kolejek, przez co serwer menedżera kolejek przestaje być potencjalnym punktem awarii.

Wstępne koncepcje rozsyłania grupowego

IBM MQ Rozsyłanie grupowe może być łatwo zintegrowane z istniejącymi systemami i aplikacjami za pomocą obiektu Informacje o komunikacji (COMMINFO). Dwa pola obiektu TOPIC umożliwiają szybką konfigurację istniejących obiektów TOPIC w celu obsługi lub ignorowania ruchu rozsyłania grupowego.

Obiekty wymagane do rozsyłania grupowego

Poniższe informacje stanowią krótki przegląd dwóch obiektów wymaganych dla programu IBM MQ Multicast:

COMMINFO, obiekt

Obiekt COMMINFO zawiera atrybuty powiązane z transmisją rozsyłania grupowego. Więcej informacji na temat parametrów obiektu COMMINFO zawiera sekcja [DEFINE COMMINFO](#).

Jedynym polem COMMINFO, które MUSI być ustawione, jest nazwa obiektu COMMINFO. Nazwa ta jest następnie używana do identyfikowania obiektu COMMINFO w temacie. Pole **GRPADDR** obiektu COMMINFO musi być zaznaczone, aby upewnić się, że wartość jest poprawnym adresem grupowym rozsyłania grupowego.

Obiekt tematu

Temat jest tematem informacji, które są publikowane w komunikacie publikowania/subskrypcji, a temat jest definiowany przez utworzenie obiektu TOPIC. Więcej informacji na temat parametrów obiektu TOPIC zawiera sekcja [DEFINE TOPIC\(DEFINIOWANIE TEMATU\)](#).

Istniejące tematy mogą być używane z rozsyłaniem grupowym przez zmianę wartości następujących parametrów obiektu TOPIC: **COMMINFO** i **MCAST**.

- **COMMINFO** Ten parametr określa nazwę obiektu informacji o komunikacji rozsyłania grupowego.
- **MCAST** Ten parametr określa, czy rozsyłanie grupowe jest dozwolone na tej pozycji w drzewie tematów. Domyślnie parametr **MCAST** jest ustawiony na wartość **ASPARENT**, co oznacza, że atrybut rozsyłania tematu jest dziedziczony z elementu nadrzędnego. Ustawienie opcji **MCAST** na wartość **ENABLED** umożliwia ruch grupowy przy użyciu rozsyłania grupowego na tym węźle.

Sieci i tematy rozsyłania grupowego

Poniższe informacje stanowią przegląd czynności związanych z subskrypcjami o różnych typach subskrypcji i definicji tematów. W poniższych przykładach założono, że parametr **COMMINFO** obiektu TOPIC jest ustawiony na nazwę poprawnego obiektu COMMINFO:

Włączony zestaw tematów do rozsyłania grupowego

Jeśli parametr łańcucha tematu **MCAST** jest ustawiony na wartość **ENABLED**, subskrypcje z klientów z obsługą rozsyłania grupowego są dozwolone, a subskrypcja rozsyłania grupowego jest dokonywana, chyba że:

- Jest to trwała subskrypcja z klienta obsługującego rozsyłanie grupowe.
- Jest to subskrypcja niezarządzana przez klienta z obsługą rozsyłania grupowego.
- Jest to subskrypcja z klienta, który nie obsługuje rozsyłania grupowego.

W takich przypadkach następuje subskrypcja bez rozsyłania grupowego, a subskrypcje są obniżane do normalnego publikowania/subskrybowania.

Temat ustawiony na rozsyłanie grupowe wyłączony

Jeśli parametr łańcucha tematu **MCAST** jest ustawiony na wartość **DISABLED**, zawsze jest dokonywana subskrypcja bez rozsyłania grupowego i subskrypcje są obniżane do normalnego publikowania/subskrybowania.

Tylko temat ustawiony na rozsyłanie grupowe

Jeśli parametr łańcucha tematu **MCAST** jest ustawiony na wartość **ONLY**, subskrypcje z klientów z obsługą rozsyłania grupowego są dozwolone, a subskrypcja rozsyłania grupowego jest dokonywana, chyba że:

- Jest to subskrypcja trwała: subskrypcje trwałe są odrzucane z kodem przyczyny 2436 (0984) (RC2436): MQRC_DURABILITY_NOT_ALLOWED
- Jest to subskrypcja niezarządzana: subskrypcje bez zarządzania są odrzucane z kodem przyczyny 2046 (07FE) (RC2046): MQRC_OPTIONS_ERROR
- Jest to subskrypcja z klienta nieobsługującego rozsyłania grupowego: subskrypcje te są odrzucane z kodem przyczyny 2560 (0A00) (RC2560): MQRC_MULTICAST_ONLY
- Jest to subskrypcja z aplikacji powiązanej lokalnie: subskrypcje te są odrzucane z kodem przyczyny 2560 (0A00) (RC2560): MQRC_MULTICAST_ONLY

Windows

Linux

AIX

MQ TelemetryPrzeгляд

Produkt MQ Telemetry składa się z usługi telemetrycznej (MQXR), która jest częścią menedżera kolejek, klientów telemetrycznych, które można samodzielnie napisać lub pobrać za darmo, a także interfejsy administracyjne wiersza komend i eksploratora. Telemetria odnosi się do gromadzenia danych z wielu różnych urządzeń zdalnych i administrowania nimi. Za pomocą programu MQ Telemetry można zintegrować gromadzenie danych i sterowanie urządzeniami za pomocą aplikacji WWW.

W produkcie IBM WebSphere MQ 7.1 produkt MQ Telemetry jest komponentem produktu IBM MQ. Aktualizacja dla tych wersji jest zasadniczo instalowana w nowszej wersji produktu IBM MQ.

Jednak począwszy od IBM MQ 8.0, pakiet Client Software Development Kit nie jest już dostarczany jako część produktu. Podobne przykładowe aplikacje są nadal swobodnie dostępne w środowisku Eclipse Paho i MQTT.org. Patrz [IBM MQ Telemetry Transport przykładowe programy](#).

Ponieważ produkt MQ Telemetry jest komponentem produktu IBM WebSphere MQ 7.1 i jego nowszych wersji, produkt MQ Telemetry może zostać zainstalowany z produktem głównym lub zainstalowany po zainstalowaniu produktu głównego. Informacje na temat migracji znajdują się w sekcji [Migrowanie produktu MQ Telemetry w systemie Windows](#) i [Migrowanie produktu MQ Telemetry w systemie Linux](#).

Produkt MQ Telemetry zawiera następujące komponenty:

Kanały pomiarowe

Kanały telemetryczne służą do zarządzania połączeniem klientów MQTT z produktem IBM MQ . Kanały telemetryczne używają nowych obiektów IBM MQ , takich jak SYSTEM . MQTT . TRANSMIT . QUEUE, do interakcji z produktem IBM MQ.

Usługa telemetryczna (MQXR)

Klienci MQTT korzystają z usługi telemetrycznej SYSTEM . MQXR . SERVICE w celu nawiązania połączenia z kanałami telemetrycznymi.

IBM MQ Explorer obsługa MQ Telemetry

MQ Telemetry można administrować za pomocą IBM MQ Explorer.

Documentation

Dokumentacja produktu MQ Telemetry jest zawarta w standardowej dokumentacji produktu IBM MQ z produktu IBM WebSphere MQ 7.1 . Dokumentacja pakietu SDK dla klientów Java i C jest dostępna w dokumentacji produktu, a także w dokumentacji produktu Javadoc i HTML.

Pojęcia dotyczące telemetrii

Zbierasz informacje ze środowiska wokół siebie, aby zdecydować, co robić. Jako konsument, sprawdzasz co masz w sklepie, przed podjęciem decyzji o tym, co jedzenie kupić. Chcesz wiedzieć, jak długo trwa podróż, jeśli odejdziesz teraz, przed dokonując rezerwacji połączenia. Sprawdzasz swoje objawy, przed podjęciem decyzji o wizycie u lekarza. Sprawdzasz, kiedy autobus przyjedzie, przed podjęciem decyzji o tym, czy czekać. Informacje na temat tych decyzji pochodzą bezpośrednio z liczników i urządzeń, od słowa pisanego na papierze lub z ekranu, a od Ciebie. Gdzie kiedykolwiek jesteś, a kiedy kiedykolwiek potrzebujesz, zbierasz informacje, zbierasz je razem, analizujesz i działasz na nim.

Jeśli źródła informacji są szeroko rozpraszane lub niedostępne, staje się to trudne i kosztowne, aby zebrać najdokładniejsze informacje. Jeśli jest wiele zmian, które chcesz wprowadzić, lub trudno dokonać zmian, to zmiany nie są wprowadzane, lub są wykonane, gdy są mniej skuteczne.

Co jeśli koszty zbierania informacji z i kontroli, szeroko rozpraszanych urządzeń są znacznie zredukowane poprzez połączenie urządzeń z technologią cyfrową do internetu? Informacje mogą być analizowane przy użyciu zasobów internetu i przedsiębiorstwa. Masz więcej możliwości, aby podejmować świadome decyzje i działać na nich.

Tendencje technologiczne, a także naciski środowiskowe i gospodarcze, kierują te zmiany, aby się wydarzyć:

1. Koszt podłączenia i sterowania czujnikami i siłownikami zmniejsza się, ze względu na standaryzację i podłączenie do niskokosztowych procesorów cyfrowych.
2. Internet, i technologie internetowe, są coraz częściej wykorzystywane do łączenia urządzeń. W niektórych krajach telefony komórkowe przekraczają komputery osobiste w liczbie połączeń do aplikacji internetowych. Inne urządzenia są z pewnością śledzone.
3. Internet, a technologie internetowe, znacznie ułatwiają aplikację do pobierania danych. Łatwy dostęp do danych prowadzi do korzystania z analizy danych w celu włączenia danych z czujników do informacji przydatnych w wielu innych rozwiązaniach.
4. Inteligentne wykorzystanie zasobów jest często szybszym i tańszym sposobem ograniczenia emisji dwutlenku węgla i kosztów. Alternatywy: znalezienie nowych zasobów, lub opracowanie nowych technologii w celu wykorzystania istniejących zasobów, może być rozwiązaniem długoterminowym. W krótkim okresie rozwój nowych technologii, lub znalezienie nowych zasobów, jest często bardziej ryzykowny, wolniejszy i bardziej kosztowny, niż udoskonalanie istniejących rozwiązań.

Przykład

Przykład pokazuje, w jaki sposób te trendy tworzą nowe możliwości w inteligentnej interakcji z otoczeniem.

Międzynarodowa Konwencja o Bezpieczeństwie Życia na Morzu (SOLAS) wymaga, aby system automatycznego rozpoznawania statków (AIS) był wdrażany na wielu statkach. Jest on wymagany na statkach handlowych powyżej 300 ton i statków pasażerskich. AIS to przede wszystkim system zapobiegania kolizjom dla żeglugi przybrzeżnej. Jest on wykorzystywany przez władze morskie do monitorowania i kontroli wód przybrzeżnych.

Entuzjaści na całym świecie wdrażają niskokosztowe stacje śledzenia AIS i umieszczają informacje o żegludze przybrzeżnej w internecie. Inni entuzjaści piszą aplikacje, które łączą w sobie informacje z AIS z innymi informacjami z internetu. Wyniki są umieszczane na stronach internetowych, a publikowane za pomocą Twittera i SMS.

W jednej aplikacji informacje pochodzące z stacji AIS w pobliżu Southampton są łączone z prawem własności statku i informacjami geograficznymi. Aplikacja przekazuje informacje na żywo o przyjazdach promowych i wyjazdach do serwisu Twitter. Zwykli komentatorzy korzystający z promów pomiędzy Southampton a Wyspą Wight subskrybują nowości za pomocą Twittera lub SMS-ów. Jeśli pasażer pokazuje swój prom biegnie późno, dojeżdżający mogą opóźnić swój wyjazd i złapać prom, gdy doki później niż jego planowy czas przylotu.

Więcej przykładów można znaleźć w sekcji [“Przypadki użycia telemetrii”](#) na stronie 115.

Zadania pokrewne

[Instalowanie produktu MQ Telemetry](#)

[administrowanieMQ Telemetry](#)

[Migrowanie produktu MQ Telemetry w systemie Windows](#)

[Migrowanie produktu MQ Telemetry w systemie Linux](#)

[Tworzenie aplikacji dla składnika MQ Telemetry](#)

[MQ Telemetry rozwiązywanie problemów](#)

Odsyłacze pokrewne

[Informacje uzupełniające dotyczące produktu MQ Telemetry](#)

Windows

Linux

AIX

MQ Telemetry - wprowadzenie

Ludzie, firmy i rządy coraz częściej chcą używać MQ Telemetry do bardziej inteligentnej interakcji z otoczeniem, w którym żyjemy i pracujemy. Program MQ Telemetry łączy wszystkie rodzaje urządzeń z Internetem i z przedsiębiorstwem, a także zmniejsza koszty tworzenia aplikacji dla inteligentnych urządzeń.

Co to jest MQ Telemetry?

- Jest to składnik produktu IBM MQ , który rozszerza uniwersalne szkielety przesyłania komunikatów udostępniane przez produkt IBM MQ na szeroką gamę zdalnych czujników, urządzeń uruchamiających i urządzeń telemetrycznych. Produkt MQ Telemetry rozszerza produkt IBM MQ , tak aby mógł łączyć inteligentne aplikacje korporacyjne, usługi i decydentów z sieciami instrumentowanych urządzeń.
- Podstawowe części produktu MQ Telemetry są następujące:

Usługa MQ Telemetry (MQXR).

Ta usługa działa wewnątrz serwera IBM MQ i używa protokołu IBM MQ Telemetry Transport (MQTT) do komunikowania się z urządzeniami telemetrycznych.

Aplikacje produktu MQTT , które są zapisywane.

Te aplikacje sterują informacjami, które są przenoszone między urządzeniami telemetrycznych i menedżerem kolejek produktu IBM MQ , a także wszelkimi działaniami podejmowana w odpowiedzi na te informacje. Aby ułatwić tworzenie tych aplikacji, należy użyć bibliotek klienta MQTT .

Co to może dla mnie zrobić?

- MQTT to otwarty transport przesyłania komunikatów, który umożliwia tworzenie implementacji MQTT dla wielu różnych urządzeń.
- Klienci MQTT mogą działać na małych urządzeniach, które mają ograniczone zasoby.
- Produkt MQTT działa wydajnie w sieciach, w których przepustowość jest niska, gdzie koszt wysyłania danych jest kosztowny lub który może być niestabilny.
- Dostarczanie komunikatów jest zapewnione i oddzielone od aplikacji.
- Programiści aplikacji nie muszą posiadać wiedzy na temat programowania komunikacji.
- Komunikaty mogą być wymieniane z innymi aplikacjami przesyłania komunikatów. Może to być inna aplikacja telemetryczna lub aplikacja MQI, JMS lub korporacyjna aplikacja do przesyłania komunikatów.

Jak go używać?

- Udostępniono przykładowe skrypty, które działają z przykładową aplikacją kliencką IBM MQ Telemetry Transport v3 (mqttv3app.jar). Patrz [IBM MQ Telemetry Transport przykładowe programy](#).
- IBM MQ Explorer i powiązane z nim narzędzia umożliwiają administrowanie funkcją telemetryczną produktu IBM MQ.
- Biblioteki klienta ułatwiają tworzenie aplikacji produktu MQTT, które łączą się z menedżerem kolejek, i które korzystają z przesyłania komunikatów w trybie publikowania/subskrypcji.
- Dystrybuuj bibliotekę aplikacji i klienta do urządzenia, w którym aplikacja ma być uruchamiana.

Jak to działa?

- MQTT jest protokołem publikowania subskrypcji. Aplikacja kliencka MQTT może publikować komunikaty na serwerze MQTT lub zasubskrybować komunikaty wysyłane przez aplikacje, które łączą się z serwerem MQTT.
- Aplikacje klienckie produktu MQTT korzystają z bibliotek klienta, które implementują transport komunikatów produktu MQTT.
- Podstawowa aplikacja kliencka MQTT działa podobnie jak standardowy klient MQ, ale może działać na znacznie szerszej gamie platform i sieci.
- Usługa MQ Telemetry (MQXR) zmienia menedżera kolejek produktu IBM MQ na serwer MQTT.
- Gdy menedżer kolejek produktu IBM MQ działa jako serwer MQTT, inne aplikacje, które łączą się z menedżerem kolejek, mogą subskrybować i odbierać komunikaty z klienta MQTT.
- Menedżer kolejek działa jako router, który dystrybuuje komunikaty z aplikacji publikujących do subskrybowania aplikacji.
- Komunikaty mogą być dystrybuowane między różnymi typami aplikacji klienckich. Na przykład między klientami Telemetry i klientami JMS.

Uwaga: MQ Telemetry zastępuje węzły SCADA wycofane w wersji 7 produktu WebSphere Message Broker (obecnie nazywanego IBM Integration Bus) i działa w systemach Windows, Linux i AIX.

Windows

Linux

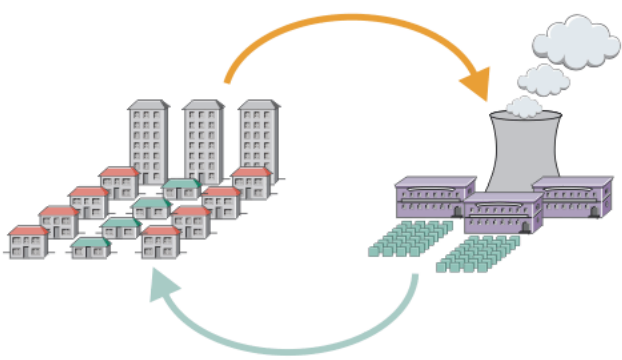
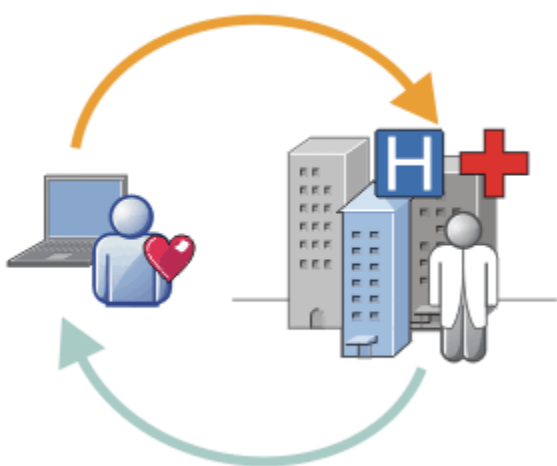
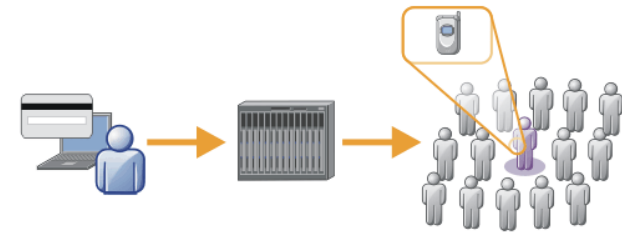
AIX

Przypadki użycia telemetry

Telemetria to automatyczne czujniki, pomiar danych i sterowanie urządzeniami zdalnymi. Nacisk kładziony jest na przesyłanie danych z urządzeń do centralnego punktu kontrolnego. Telemetria obejmuje również wysyłanie informacji o konfiguracji i sterowaniu do urządzeń.

Produkt MQ Telemetry łączy małe urządzenia za pomocą konsoli MQTT protokołu łączy urządzenia z innymi aplikacjami za pomocą programu IBM MQ. MQ Telemetry mosty przepaść między urządzeniami a Internetem, co ułatwia budowanie "inteligentnych rozwiązań". Inteligentne rozwiązania odblokowują bogactwo informacji dostępnych w internecie, a także w aplikacjach korporacyjnych, dla aplikacji monitorujących i sterujących urządzeniami.

Poniższe diagramy demonstrują niektóre typowe zastosowania produktu MQ Telemetry:

| Telemetria: Smart Electricity | |
|---|--|
|  | <ul style="list-style-type: none">• Komunikat MQTT zawierający dane o wykorzystaniu energii wysłane do dostawcy usług.• Produkt MQ Telemetry wysyła komendy sterujące oparte na analizie danych o wykorzystaniu energii.• Więcej informacji na ten temat można znaleźć w następującym przypadku użycia: “Przypadek użycia telemetrii: Monitorowanie energii domowej i sterowanie” na stronie 118 |
| Telemetria: Inteligentne usługi zdrowotne | |
| <ul style="list-style-type: none">• Produkt MQ Telemetry wysyła dane dotyczące poprawności do szpitala & Doctor.• Alerty komunikatów produktu MQTT lub informacje zwrotne mogą być wysłane na podstawie analizy danych poprawności.• Więcej informacji na ten temat można znaleźć w następującym przypadku użycia: “Przypadek użycia telemetrii: Monitorowanie pacjentów domowych” na stronie 116 |  |
| Telemetria: Jeden w tłumie | |
|  | <ul style="list-style-type: none">• Transakcja z kartą prostą jest wysyłana na serwer banku.• Produkt MQ Telemetry identyfikuje jedną osobę z tysięcy, ostrzegając klienta o tym, że używana jest ich karta.• Produkt MQ Telemetry może korzystać z najprostszego wprowadzania informacji, a także znaleźć tę osobę. |

Przypadki użycia opisane w podtematach są rysowane na podstawie rzeczywistych przykładów. Ilustrują one niektóre sposoby korzystania z telemetrii, a także niektóre z typowych problemów, które muszą rozwiązać technologia telemetryczna.

Windows Linux AIX **Przypadek użycia telemetrii: Monitorowanie pacjentów domowych**

We współpracy między IBM a świadczeniodawcą w systemie opieki nad pacjentem kardiochirskim, defibrylator kardiowerterowy wszczepiony jest ze szpitalem. Dane dotyczące pacjenta i wszczepionego urządzenia są przesyłane z wykorzystaniem telemetrii RF do urządzenia MQTT w domu pacjenta.

Zwykle transfer odbywa się nocnie do nadajnika znajdującego się przy łóżku. Nadajnik przesyła dane bezpiecznie przez system telefoniczny do szpitala, gdzie dane są analizowane.

System redukuje liczbę wizyt pacjenta musi wykonać lekarz. Wykrywa on, kiedy pacjent lub urządzenie wymaga uwagi, a w przypadku wystąpienia nagłego zagrożenia alarmuje on dyżurnego lekarza.

Współpraca między produktem IBM a dostawcą usług medycznych ma cechy wspólne dla wielu przypadków użycia telemetrii:

Invisibility

Urządzenie nie wymaga interwencji użytkownika innego niż zasilanie, linia telefoniczna i przebycie w pobliżu urządzenia przez część dzienną. Jego działanie jest niezawodne i proste w obsłudze.

Aby usunąć konieczność skonfigurowania urządzenia przez pacjenta, dostawca urządzenia wstępnie skonfiguruje urządzenie. Pacjent musi go podłączyć tylko do siebie. Wyeliminowanie konfiguracji przez pacjenta upraszcza pracę urządzenia i zmniejsza szanse na nieśluszenie skonfigurowane urządzenie.

Klient MQTT jest osadzony jako część urządzenia. Programista urządzeń osadza implementację klienta MQTT w urządzeniu, a programista lub dostawca konfiguruje klienta MQTT jako część konfiguracji wstępnej.

Klient MQTT jest dostarczany jako plik JAR Java SE, który programista zawiera w swojej aplikacji Java. W środowiskach innych niż Java, takich jak ten, programista urządzeń może zaimplementować klienta w innym języku, korzystając z opublikowanych formatów i protokołów MQTT. Programista może również użyć jednego z klientów C dostarczanych jako współużytkowane biblioteki dla platform Windows, Linux i ARM.

Nieparzysta łączność

Komunikacja pomiędzy defibrylatorem a szpitalem ma nierówne cechy sieciowe. Dwie różne sieci służą do rozwiązywania różnych problemów związanych z gromadzeniem danych od pacjenta, a także przesyłaniem danych do szpitala. Między patentem a urządzeniem MQTT używana jest sieć krótkofalowa o małej mocy. Nadajnik łączy się ze szpitalem za pomocą połączenia VPN TCP/IP w linii telefonicznej o niskiej przepustowości.

Znalezienie sposobu na podłączenie każdego urządzenia bezpośrednio do sieci Internet Protocol jest często niepraktyczne. Korzystanie z dwóch sieci, połączonych za pomocą koncentratora, stanowi wspólne rozwiązanie. Urządzenie MQTT jest prostym serwerem koncentrującym, przechowując informacje od pacjenta i przekazując go do szpitala.

Zabezpieczenia

Lekarz musi mieć możliwość zaufania do autentyczności danych pacjenta, a pacjent chce, aby prywatność ich danych była respektowana.

W niektórych sytuacjach wystarczy zaszyfrować połączenie, używając sieci VPN lub TLS. W innych sytuacjach pożądane jest zachowanie bezpieczeństwa danych nawet po jego zapisaniu.

Czasami urządzenie telemetryczne nie jest bezpieczne. Może to być na przykład we wspólnym mieszkaniu. Użytkownik urządzenia musi być uwierzytelniony, aby upewnić się, że dane pochodzą od właściwego pacjenta. Samo urządzenie może zostać uwierzytelnione na serwerze za pomocą protokołu TLS, a serwer uwierzytelniony na urządzeniu.

Kanał pomiarowy między urządzeniem i menedżerem kolejek obsługuje usługę JAAS w celu uwierzytelniania użytkowników i TLS w celu szyfrowania komunikacji i uwierzytelniania urządzeń. Dostęp do publikacji jest kontrolowany przez menedżera uprawnień do obiektów w programie IBM MQ.

Identyfikator używany do uwierzytelniania użytkownika może być odwzorowany na inny identyfikator, taki jak wspólna tożsamość pacjenta. Wspólny identyfikator upraszcza konfigurowanie autoryzacji do tematów publikacji w produkcie IBM MQ.

Łączność

Połączenie między urządzeniem MQTT a szpitalem korzysta z połączenia modemowego i działa z pasmem o niskiej przepustowości 300 baud.

Aby działać efektywnie przy 300 baud, program MQTT protocol dodaje tylko kilka dodatkowych bajtów do komunikatu, oprócz nagłówków TCP/IP.

Produkt MQTT protocol udostępnia przesyłanie komunikatów *fire i forget* o pojedynczej transmisji, które utrzymuje niskie opóźnienia. Możliwe jest również użycie wielu transmisji w celu zagwarantowania *co najmniej raz i dokładnie raz* dostarczenia, jeśli gwarantowane dostarczenie jest ważniejsze niż czas odpowiedzi. W celu zagwarantowania dostarczenia wiadomości są przechowywane w urządzeniu do czasu ich pomyślnego dostarczenia. Jeśli urządzenie jest połączone bezprzewodowo, gwarantowana dostawa jest szczególnie przydatna.

Skalowalność

Urządzenia telemetryczne są zazwyczaj wdrażane w dużych ilościach, od dziesiątek tysięcy do milionów.

Podłączenie wielu urządzeń do systemu stawia duże zapotrzebowanie na rozwiązanie. Istnieją wymagania biznesowe, takie jak koszt urządzeń i ich oprogramowania, a także wymagania administracyjne związane z zarządzaniem licencjami, urządzeniami i użytkownikami. Wymagania techniczne obejmują obciążenie sieci, a także na serwerach.

Podczas otwierania połączeń używany jest więcej zasobów serwera niż utrzymywanie otwartych połączeń. Jednak w przypadku użycia takich jak ta, która używa linii telefonicznych, koszt połączeń oznacza, że połączenia nie są otwarte, nie dłużej niż jest to wymagane. Transfery danych w dużej mierze mają charakter nietoperzowy. Połączenia mogą być zaplanowane przez całą noc, aby uniknąć nagłego szczytu połączeń w porze nocnej.

W przypadku klienta skalowalność klientów jest wspomagana przez wymaganą minimalną konfigurację klienta. Klient MQTT jest osadzony w urządzeniu. Nie ma wymogu, aby konfiguracja lub krok akceptacji licencji klienta MQTT zostały wbudowane w wdrażanie urządzeń dla pacjentów.

Na serwerze program MQ Telemetry ma początkowy cel 50 000 otwartych połączeń na menedżera kolejek.

Połączenia są zarządzane za pomocą konsoli IBM MQ Explorer. Program IBM MQ Explorer filtruje połączenia, które mają być wyświetlane do zarządzalnej liczby. Używając odpowiednio wybranego schematu przydzielania identyfikatorów do klientów, można filtrować połączenia w oparciu o geografię lub alfabetycznie według nazwy pacjenta.

Windows Linux AIX **Przypadek użycia telemetrii: Monitorowanie energii domowej i sterowanie**

Inteligentne liczniki zbierają więcej szczegółów na temat zużycia energii niż tradycyjne liczniki.

Inteligentne liczniki są często sprzężone z lokalną siecią telemetryczną do monitorowania i sterowania poszczególnymi urządzeniami w domu. Niektóre z nich są również połączone zdalnie w celu monitorowania i kontroli na odległość.

Połączenie zdalne może być skonfigurowane przez pojedyncze, przez narzędzie zasilania lub przez centralny punkt kontrolny. Zdalny punkt kontrolny może odczytywać wykorzystanie zasilania i udostępniać dane o wykorzystaniu. Może on dostarczać danych, które mają wpływ na takie wykorzystanie, jak ciągłe ustalanie cen i informacje o pogodzie. Może to ograniczyć obciążenie, aby poprawić ogólną efektywność wytwarzania energii.

Inteligentne liczniki zaczynają się szeroko wdrażać. Na przykład rząd brytyjski jest w konsultacji na temat rozmieszczenia inteligentnych liczników do każdego domu brytyjskiego do 2020 roku.

Przypadki użycia pomiarów w domu mają pewną liczbę cech wspólnych:

Invisibility

O ile użytkownik nie chce być zaangażowany w oszczędzanie energii za pomocą miernika, licznik nie może wymagać interwencji użytkownika. Nie może on ograniczać niezawodności dostaw energii do poszczególnych urządzeń.

Klient MQTT może być osadzony w oprogramowaniu wdrożonym z licznikiem i nie wymaga oddzielnej instalacji ani konfiguracji.

Nieparzysta łączność

Komunikacja pomiędzy urządzeniami a inteligentnym miernikiem wymaga różnych standardów łączności niż między licznikiem a zdalnym punktem połączenia.

Połączenie między inteligentnym miernikiem a urządzeniami musi być wysoce dostępne i zgodne z normami sieciowymi dla sieci domowej.

Sieć zdalna prawdopodobnie będzie korzystała z różnych połączeń fizycznych. Niektóre z nich, takie jak komórkowe, mają wysokie koszty transmisji i mogą być sporadyczne. Specyfikacja MQTT v3 jest przeznaczona dla połączeń zdalnych i połączeń między adapterami lokalnymi a inteligentnym licznikiem.

Połączenie między gniazdkami zasilaczy i aplikacjami, a miernikiem, korzysta z sieci domowej, takiej jak Zigbee. MQTT dla sieci czujników (MQTT-S), jest przeznaczony do pracy z protokołami sieciowymi firmy Zigbee i innymi sieciami o niskiej przepustowości. Produkt MQ Telemetry nie obsługuje bezpośrednio produktu MQTT-S. Wymaga on bramy do połączenia serwera MQTT-S z serwerem MQTT v3.

Podobnie jak w przypadku monitorowania pacjentów domowych, rozwiązania w zakresie monitorowania i kontroli energii domowej wymagają wielu sieci, połączonych za pomocą inteligentnego licznika jako koncentratora.

Zabezpieczenia

Z inteligentnymi miernikami jest powiązana pewna liczba problemów związanych z bezpieczeństwem. Do tych zagadnień należą: nieodrzucając transakcji, autoryzacja wszystkich zainicjowanych działań kontrolnych oraz prywatność danych dotyczących zużycia energii.

Aby zapewnić prywatność, dane przesyłane między licznikiem a zdalnym punktem sterowania przez program MQTT mogą być szyfrowane za pomocą protokołu TLS. Aby zapewnić autoryzację działań sterujących, połączenie MQTT między licznikiem a zdalnym punktem kontrolnym może być wzajemnie uwierzytelnianie za pomocą protokołu TLS.

Łączność

Fizyczny charakter sieci zdalnej może się znacznie różnić. Może on korzystać z istniejącego połączenia szerokopasmowego lub korzystać z sieci mobilnej o wysokich kosztach połączeń, a także o sporadycznych dyspozycyjności. W przypadku wysokich kosztów, sporadycznych połączeń MQTT jest to wydajny i niezawodny protokół; patrz ["Przypadek użycia telemetrii: Monitorowanie pacjentów domowych"](#) na stronie 116.

Skalowalność

W końcu elektrownie, czyli centralne punkty kontrolne, planują wdrożyć dziesiątki milionów inteligentnych liczników. Początkowo liczba liczników na wdrożenie jest w dziesiątkach do setek tysięcy. Liczba ta jest porównywalna z początkowym celem produktu MQTT, który wynosi 50 000 otwartych połączeń klientów na menedżera kolejek.

Krytycznym aspektem architektury w zakresie monitorowania i kontroli energii domowej jest wykorzystanie inteligentnego miernika jako koncentratora sieciowego. Każdy adapter urządzenia jest oddzielnym czujnikiem. Łącząc je z lokalnym serwerem koncentrującym za pomocą MQTT, koncentrator może skoncentrować przepływy danych na pojedynczej sesji TCP/IP z centralnym punktem kontrolnym, a także przechowywać komunikaty przez krótki czas w celu pokonania wyłączeń sesji.

Połączenia zdalne muszą być otwarte w domowych przypadkach użycia energii z dwóch powodów. Najpierw, ponieważ otwieranie połączeń zajmuje dużo czasu w porównaniu z wysyłaniem żądań. Czas otwarcia wielu połączeń w celu wysłania żądań "load-limitation" w krótkich odstępach czasu jest zbyt długi. Po drugie, aby otrzymać żądania ograniczenia obciążenia od firmy power, połączenie musi być najpierw otwarte przez klienta. W przypadku produktu MQTT połączenia są zawsze inicjowane przez

klienta i w celu odbierania żądań ograniczenia obciążenia od jednostki zasilaczy, połączenie musi być otwarte.

Jeśli szybkość otwierania połączeń jest krytyczna lub serwer inicjuje żądania krytyczne w czasie, zwykle rozwiązaniem jest utrzymywanie wielu otwartych połączeń.

Windows

Linux

AIX

Przypadki użycia telemetrii: Identyfikacja

radiowa (RFID)

RFID to użycie wbudowanego znacznika RFID w celu identyfikacji i śledzenia obiektu bezprzewodowo. Znaczniki RFID mogą być odczytywane do zakresu kilku metrów, a poza linią wzroku czytnik RFID. Znaczniki pasywne są aktywowane przez czytnik RFID. Aktywne znaczniki przesyłają bez zewnętrznej aktywacji. Aktywne znaczniki muszą mieć źródło zasilania. Znaczniki pasywne mogą zawierać źródło zasilania, aby zwiększyć ich zakres.

RFID jest używane w wielu aplikacjach, a typy przypadków użycia różnią się znacznie. Przypadki użycia RFID, a także monitorowanie pacjentów domowych oraz monitorowanie energii domowej i przypadków użycia kontroli, mają pewne podobieństwa i różnice.

Invisibility

W wielu przypadkach użycia czytnik RFID jest wdrażany w dużych ilościach i musi pracować bez interwencji użytkownika. Czytnik zawiera wbudowany klient MQTT do komunikowania się z centralnym punktem kontrolnym.

Na przykład w magazynie dystrybucyjnym czytnik korzysta z czujnika ruchu w celu wykrycia palety. Aktywuje on znaczniki RFID elementów na paletce i wysyła dane i żądania do aplikacji centralnych. Dane są używane do aktualizowania lokalizacji zapasów. Żądania sterują tym, co dzieje się z paletą obok, np. przenosi ją do konkretnej zatoki. Linie lotnicze i systemy bagażu lotniskowego korzystają z RFID w ten sposób.

W niektórych przypadkach użycia RFID czytnik ma standardowe środowisko obliczeniowe, takie jak Java Platform, Micro Edition (Java ME). W takich przypadkach klient MQTT może zostać wdrożony w odrębnym kroku konfiguracji po zakończeniu produkcji.

Nieparzysta łączność

Czytniki RFID mogą być oddzielone od lokalnego urządzenia sterującego, które zawiera klienta MQTT, lub każdy czytnik może osadzić klienta MQTT. Zwykle czynniki geograficzne lub komunikacyjne wskazują na wybór topologii.

Zabezpieczenia

Prywatność i autentyczność to problemy związane z bezpieczeństwem w przywiązaniu znaczników RFID. Znaczniki RFID są nieufne i mogą być ściśle monitorowane, spoofed lub manipulowane przy użyciu.

Rozwiązanie problemów związanych z bezpieczeństwem RFID zwiększa szansę wdrożenia nowych rozwiązań RFID. Mimo że narażenie bezpieczeństwa jest w znaczniku RFID, a lokalny czytnik, przy użyciu centralnego przetwarzania informacji sugeruje podejścia do przeciwdziałania innym zagrożeniom. Na przykład manipulowanie oznakami może być wykrywane przez dynamicznie korelujące poziomy zapasów w odniesieniu do dostaw i wysyłek.

Łączność

Aplikacje RFID zwykle zajmują się zarówno przechowaniem danych wsadowych, jak i przesyłaniem informacji zebranych od czytników RFID i natychmiastowych zapytań. W przypadku użycia magazynu dystrybucyjnego, czytnik RFID jest podłączony przez cały czas. Po przeczytaniu znacznika jest on publikowany razem z informacjami o czytniku. Aplikacja hurtowni publikuje odpowiedź z powrotem do programu czytającego.

W aplikacji opracowywania danych sieć jest zwykle niezawodna, a natychmiastowe żądania mogą używać komunikatów *fire i forget* w celu zapewnienia niskiej wydajności. Wsadowe dane sklepu i przesyłania danych mogą być używane w przesyłaniu komunikatów produktu *dokładnie raz* w celu zminimalizowania kosztów administracyjnych związanych z lotyzującymi danymi.

Skalowalność

Jeśli aplikacja RFID wymaga natychmiastowych odpowiedzi, w kolejności drugiej lub drugiej, czytniki RFID muszą pozostać połączone.

Windows

Linux

AIX

Przypadki użycia telemetrii: sensowanie

Środowiska

Sensowanie środowiska wykorzystuje telemetrię do zbierania informacji o poziomach wód rzeki i jakości, zanieczyszczeń atmosferycznych, i innych danych środowiskowych.

Czujniki często znajdują się w odległych miejscach, bez dostępu do przewodowej komunikacji. Przepustowość bezprzewodowa jest kosztowna, a niezawodność może być niska. Zwykle pewna liczba czujników otoczenia w małym obszarze geograficznym jest połączona z lokalnym urządzeniem monitoringowym w bezpiecznym miejscu. Połączenia lokalne mogą być połączone lub bezprzewodowe.

Invisibility

Urządzenia czujnika mogą być mniej dostępne, niższe z zasilaniem i rozmieszczone w większej liczbie, niż centralne urządzenie monitorujące. Czujniki są czasami "głupie", a lokalne urządzenie monitorujące zawiera adaptery do transformowania i przechowywania danych czujników. Urządzenie monitorujące prawdopodobnie będzie wyposażone w komputer ogólnego przeznaczenia, który obsługuje Java Platform, Standard Edition (Java SE) lub Java Platform, Micro Edition (Java ME). Podczas konfigurowania klienta MQTT niewidoczność jest mało prawdopodobna, aby była wymagana.

Nieparzysta łączność

Możliwości czujników, a także koszty i przepustowość połączenia zdalnego, zazwyczaj skutkują w lokalnym serwerze koncentrującym monitorowanie połączonym z serwerem centralnym.

Zabezpieczenia

Jeśli rozwiązanie nie jest używane w przypadku użycia wojskowego lub obronnego, zabezpieczenie nie jest istotnym wymogiem.

Łączność

Wiele zastosowań nie wymaga stałego monitorowania lub natychmiastowej dostępności danych. Dane o wyjątkach, takie jak alarm na poziomie powodziowym, muszą być natychmiast przekazywane. Dane czujników są agregowane na lokalnym monitorze w celu zmniejszenia kosztów połączenia i komunikacji, a następnie są przesyłane za pomocą zaplanowanych połączeń. Dane wyjątku są przekazywane natychmiast po wykryciu ich w monitorze.

Skalowalność

Czujniki koncentrują się wokół węzłów lokalnych, a dane czujników są agregowane w pakiety, które są przesyłane zgodnie z harmonogramem. Oba te czynniki zmniejszają obciążenie na serwerze centralnym, który byłby narzucony za pomocą podłączonych bezpośrednio czujników.

Windows

Linux

AIX

Przypadki użycia telemetrii: Aplikacje mobilne

Aplikacje mobilne to aplikacje, które działają na urządzeniach bezprzewodowych. Urządzenia są ogólnymi platformami aplikacji lub urządzeniami niestandardowymi.

Ogólne platformy obejmują urządzenia kieszonkowe, takie jak telefony i asystenty danych osobowych, oraz urządzenia przenośne, takie jak komputery notebook. Urządzenia niestandardowe używają sprzętu specjalnego przeznaczenia dopasowanego do konkretnych aplikacji. Urządzenie do rejestrowania "signed-for" dostarczania paczek jest przykładem niestandardowego urządzenia mobilnego. Aplikacje na niestandardowych urządzeniach mobilnych są często budowane na ogólnej platformie programowej.

Invisibility

Wdrażanie niestandardowych aplikacji mobilnych jest zarządzane i może obejmować konfigurację aplikacji klienckiej produktu MQTT. Podczas konfigurowania klienta MQTT niewidoczność jest mało prawdopodobna, aby była wymagana.

Nieparzysta łączność

W przeciwieństwie do topologii lokalnej koncentratora w poprzednich przypadkach użycia, klienci mobilne łączą się zdalnie. Warstwa aplikacji klienckiej łączy się bezpośrednio z aplikacją na centralnym serwerze koncentrującym.

Zabezpieczenia

Przy niewielkim zabezpieczeniu fizycznym urządzenie przenośne i użytkownik mobilny muszą być uwierzytelniane. Protokół TLS jest używany do potwierdzania tożsamości urządzenia, a JAAS do uwierzytelniania użytkownika.

Łączność

Jeśli aplikacja mobilna zależy od zasięgu bezprzewodowego, musi być w stanie działać w trybie bez połączenia i skutecznie radzić sobie z przerwaniem połączenia. W tym środowisku celem jest pozostania połączona, ale aplikacja musi być w stanie zapisywać i przekazywać komunikaty. Często komunikaty są zamówieniami lub potwierdzeniami dostarczenia, a także mają ważną wartość biznesową. Muszą one być przechowywane i przekazywane rzetelnie.

Skalowalność

Skalowalność nie jest poważną kwestią. Liczba klientów aplikacji prawdopodobnie nie może przekroczyć liczby tysięcy, czyli dziesiątki tysięcy, w niestandardowych przypadkach użycia aplikacji dla urządzeń mobilnych.

Podłączanie urządzeń telemetrycznych do menedżera kolejek

Urządzenia telemetryczne łączą się z menedżerem kolejek przy użyciu klienta MQTT v3. Klient MQTT v3 używa protokołu TCP/IP do połączenia się z programem nastuchującym TCP/IP o nazwie usługi telemetrycznej (MQXR).

Po połączeniu urządzenia telemetrycznego z menedżerem kolejek klient MQTT inicjuje połączenie TCP/IP przy użyciu metody `MqttClient.connect`. Podobnie jak klienci IBM MQ, klient MQTT musi być połączony z menedżerem kolejek w celu wysyłania i odbierania komunikatów. Połączenie jest nawiązywanych na serwerze przy użyciu programu nastuchującego TCP/IP, zainstalowanego razem z programem MQ Telemetry, nazywanym usługą telemetryczną (MQXR). Każdy menedżer kolejek uruchamia maksymalnie jedną usługę telemetryczną (MQXR).

Usługa telemetryczna (MQXR) używa zdalnego adresu gniazda ustawionego przez każdego klienta w metodzie `MqttClient.connect` w celu przydzielenia połączenia do kanału telemetrycznego. Adres gniazda to kombinacja nazwy hosta TCP/IP i numeru portu. Wiele klientów używających tego samego zdalnego adresu gniazda jest połączonych z tym samym kanałem telemetrycznym przez usługę telemetryczną (MQXR).

Jeśli na serwerze znajduje się wiele menedżerów kolejek, rozdziel kanały telemetryczne między menedżerami kolejek. Przydziel zdalne adresy gniazd między menedżerami kolejek. Zdefiniuj każdy kanał telemetryczny z unikalnym adresem zdalnego gniazda. Dwa kanały telemetryczne nie mogą używać tego samego adresu gniazda.

Jeśli ten sam zdalny adres gniazda jest skonfigurowany dla kanałów pomiarowych w wielu menedżerach kolejek, pierwszy kanał pomiarowy do połączenia, wygrywa. Kolejne kanały łączące się na tym samym adresie nie powiodą się.

Jeśli na serwerze znajduje się wiele adapterów sieciowych, rozdziel adresy zdalnych gniazd między kanałami telemetrycznymi. Przydzielanie adresów gniazd jest całkowicie dowolne, o ile każdy konkretny adres gniazda jest skonfigurowany tylko na jednym kanale telemetrycznym.

Skonfiguruj produkt IBM MQ do łączenia klientów MQTT przy użyciu kreatorów udostępnionych w suplemencie MQ Telemetry dla produktu IBM MQ Explorer. Alternatywnie postępuj zgodnie z instrukcjami w sekcji [Konfigurowanie menedżera kolejek na potrzeby teledystrybucji w systemach Linux i AIX](#) oraz [Konfigurowanie menedżera kolejek dla teledystrybucji w produkcie Windows](#) w celu ręcznego skonfigurowania teledystrybucji.

Odsyłacze pokrewne

[Właściwości MQXR](#)

Windows

Linux

AIX

Protokoły połączeń telemetrycznych

Produkt MQ Telemetry obsługuje protokoły TCP/IP IPv4 i IPv6 oraz TLS.

Windows

Linux

AIX

Usługa telemetryczna (MQXR)

Usługa telemetryczna (MQXR) jest programem nasłuchującym TCP/IP, który jest zarządzany jako usługa IBM MQ. Utwórz usługę za pomocą kreatora IBM MQ Explorer lub za pomocą komendy **runmqsc**.

Usługa MQ Telemetry (MQXR) nosi nazwę SYSTEM.MQXR.SERVICE.

Przykładowy kreator konfiguracji telemetrycznej, udostępniony w funkcji MQ Telemetry dla produktu IBM MQ Explorer, tworzy usługę telemetryczną i przykładowy kanał pomiarowy. Patrz sekcja [Sprawdzenie poprawności instalacji produktu MQ Telemetry przy użyciu produktu IBM MQ Explorer](#).

Utwórz przykładową konfigurację z poziomu wiersza komend. Patrz sekcja [Weryfikowanie instalacji produktu MQ Telemetry przy użyciu wiersza komend](#).

Usługa telemetryczna (MQXR) jest uruchamiana i zatrzymywana automatycznie wraz z menedżerem kolejek. Należy sterować usługą przy użyciu folderu usług w produkcie IBM MQ Explorer. Aby wyświetlić usługę, należy kliknąć tę ikonę, aby zatrzymać program IBM MQ Explorer filtrujący obiekty SYSTEM z ekranu.

Aby uzyskać przykład ręcznego tworzenia usługi, należy zapoznać się z:

- [Linux](#) [AIX](#) [Tworzenie SYSTEM.MQXR.SERVICE w systemie Linux](#).
- [Windows](#) [Tworzenie SYSTEM.MQXR.SERVICE w systemie Windows](#).

V 9.2.4 Począwszy od wersji IBM MQ 9.2.4, [Tworzenie SYSTEM.MQXR.SERVICE w systemie Linux](#) i [Tworzenie SYSTEM.MQXR.SERVICE w systemie Windows](#) są aktualizowane w celu określenia domyślnego klucza, który będzie wymagał szyfrowania frazy hasła dla kanałów MQTT TLS. Więcej informacji na ten temat zawiera sekcja [Szyfrowanie haseł dla kanałów TLS MQTT](#).

Windows

Linux

AIX

Kanały pomiarowe

Utwórz kanały telemetryczne, aby utworzyć połączenia z różnymi właściwościami, takimi jak Java Usługa uwierzytelniania i autoryzacji (JAAS) (JAAS) lub uwierzytelnianie TLS, lub w celu zarządzania grupami klientów.

Utwórz kanały telemetryczne przy użyciu kreatora **New Telemetry Channel**, który jest dostarczany w funkcji MQ Telemetry dla produktu IBM MQ Explorer. Skonfiguruj kanał przy użyciu kreatora, aby akceptować połączenia z klientów MQTT na określonym porcie TCP/IP. Od wersji IBM WebSphere MQ 7.1 można skonfigurować program MQ Telemetry za pomocą programu wiersza komend **runmqsc**.

Tworzenie wielu kanałów telemetrycznych, na różnych portach, w celu ułatwienia zarządzania dużą liczbą połączeń klientów poprzez podział klientów na grupy. Każdy kanał pomiarowy ma inną nazwę.

Można skonfigurować kanały telemetryczne z różnymi atrybutami zabezpieczeń w celu utworzenia różnych typów połączeń. Utwórz wiele kanałów, aby akceptować połączenia klientów na różnych adresach TCP/IP. Użyj protokołu TLS do szyfrowania komunikatów i uwierzytelniania kanału pomiarowego i klienta. Patrz sekcja [Konfiguracja TLS w klientach MQTT i kanałach telemetrycznych](#). Podaj ID użytkownika, aby uprościć autoryzowanie dostępu do obiektów produktu IBM MQ. Określ konfigurację JAAS, aby uwierzytelnić użytkownika produktu MQTT za pomocą usługi JAAS. Patrz sekcja [Identyfikacja, autoryzacja i uwierzytelnianie klienta MQTT](#).

IBM MQ Telemetry Transport :NONE.

Protokół IBM MQ Telemetry Transport (MQTT) v3 jest przeznaczony do wymiany komunikatów między małymi urządzeniami w niskiej przepustowości, lub kosztownych połączeń, a także niezawodnego wysyłania komunikatów. Używa protokołu TCP/IP.

Publikacja MQTT protocol jest opublikowana; patrz [IBM MQ Telemetry Transport format i protokół](#). Wersja 3 protokołu używa publikowania/subskrybowania i obsługuje trzy rodzaje usług: *fire and forget, at least once i dokładnie jeden raz*.

Niewielka wielkość nagłówków protokołów i ładunku komunikatu tablicy bajtowej powoduje, że komunikaty są małe. Nagłówki składają się z 2-bajtowego stałego nagłówka, a do 12 bajtów dodatkowych nagłówków zmiennych. Protokół używa 12-bajtowych nagłówków zmiennych do zasubskrybowania i łączenia, a tylko 2 bajtowe nagłówki zmiennych dla większości publikacji.

Dzięki trzem jakości usług można wyhandlować między niewielkim opóźnieniem i niezawodnością; patrz [Jakość usług świadczonych przez klienta MQTT](#). Opcja *Ogień i zapomnij* nie używa pamięci trwałej urządzenia, a tylko jedna transmisja do wystania lub odebrania publikacji. *Co najmniej raz i dokładnie jeden raz* wymaga trwałej pamięci masowej na urządzeniu w celu zachowania stanu protokołu i zapisania komunikatu, dopóki nie zostanie potwierdzone.

MQTT klienty

Aplikacja kliencka MQTT jest odpowiedzialna za gromadzenie informacji z urządzenia telemetrycznego, nawiązywaniem połączenia z serwerem oraz publikowanie informacji na serwerze. Może również zasubskrybować tematy, odbierać publikacje i sterować urządzeniem telemetrycznym.

W przeciwieństwie do aplikacji klienckich IBM MQ aplikacje klienckie MQTT nie są aplikacjami IBM MQ . Nie określają one menedżera kolejek, z którym ma zostać nawiązane połączenie. Nie są one ograniczone do korzystania z konkretnych interfejsów programistycznych produktu IBM MQ . Zamiast tego klienty MQTT implementują protokół MQTT 3 . Użytkownik może napisać własną bibliotekę kliencką, aby interfejs był używany w języku MQTT protocol w języku programowania, a także na platformie wybranej przez użytkownika. Patrz [IBM MQ Telemetry Transport format i protokół](#).

Aby uprościć pisanie aplikacji klienckich MQTT , należy użyć bibliotek klienta C, Javai JavaScript , które hermetyzują MQTT protocol w celu uzyskania wielu platform. W przypadku włączenia tych bibliotek do aplikacji MQTT , w pełni funkcjonalny klient produktu MQTT może być tak krótki, jak 15 wierszy kodu. Biblioteki klienta produktu MQTT są swobodnie dostępne w środowisku Eclipse Paho oraz pod adresem MQTT.org. Patrz [IBM MQ Telemetry Transport przykładowe programy](#).

Aplikacja kliencka MQTT jest zawsze odpowiedzialna za zainicjowanie połączenia z kanałem telemetrycznym. Po nawiązaniu połączenia aplikacja kliencka MQTT lub aplikacja IBM MQ może rozpocząć wymianę komunikatów.

Aplikacje klienckie MQTT i aplikacje produktu IBM MQ publikują i subskrybują ten sam zestaw tematów. Aplikacja IBM MQ może również wysłać komunikat bezpośrednio do aplikacji klienckiej MQTT bez pierwszej aplikacji klienckiej, która tworzy subskrypcję. Patrz sekcja [Konfigurowanie rozproszonego kolejkowania w celu wysyłania komunikatów do klientów MQTT](#).

Aplikacje klienckie produktu MQTT są połączone z produktem IBM MQ za pomocą kanału telemetrycznego. Kanał pomiarowy działa jako most między różnymi typami komunikatów używanych przez produkty MQTT i IBM MQ. Tworzy on publikacje i subskrypcje w menedżerze kolejek w imieniu aplikacji klienckiej MQTT . Kanał pomiarowy wysyła publikacje, które są zgodne z subskrypcjami aplikacji klienckiej MQTT z menedżera kolejek do aplikacji klienckiej MQTT .

Wysyłanie komunikatu do klienta MQTT

Aplikacje produktu IBM MQ mogą wysyłać komunikaty klientów produktu MQTT v3 przez publikowanie do subskrypcji utworzonych przez klienty lub bezpośrednio przez wysyłanie komunikatów. Klienty MQTT mogą wysyłać do siebie komunikaty, publikując tematy subskrybowane przez innych klientów.

Klient MQTT subskrybuje publikację, którą otrzymuje od IBM MQ .

Wykonaj zadanie “Publikowanie komunikatu w programie narzędziowym klienta MQTT z programu IBM MQ Explorer” na stronie 126 , aby wysłać publikację z programu IBM MQ do klienta MQTT .

Standardowym sposobem odbierania komunikatów przez klient MQTT v3 jest utworzenie subskrypcji tematu lub zestawu tematów. W przykładowym fragmencie kodu, Rysunek 45 na stronie 125, klient MQTT subskrybuje użycie łańcucha tematu "MQTT Examples". Aplikacja IBM MQ C, Rysunek 46 na stronie 126, publikuje w temacie za pomocą łańcucha tematu "MQTT Examples". W fragmencie kodu Rysunek 47 na stronie 126 klient MQTT otrzymuje publikację w metodzie wywołania zwrotnego `messageArrived`.

Więcej informacji na temat sposobu konfigurowania produktu IBM MQ w celu wysyłania publikacji w odpowiedzi na subskrypcje klientów MQTT zawiera sekcja [Publikowanie komunikatu w odpowiedzi na subskrypcję klienta MQTT](#).

Aplikacja IBM MQ wysyła komunikat bezpośrednio do klienta MQTT

Wykonaj zadanie “Wysyłanie wiadomości do klienta MQTT za pomocą IBM MQ Explorer” na stronie 131 , aby wysłać wiadomość bezpośrednio z programu IBM MQ do klienta MQTT .

Komunikat wysłany w ten sposób do klienta MQTT jest nazywany komunikatem niezamówionym. Klienci MQTT v3 odbierają niezamówione komunikaty jako publikacje z zestawem nazw tematów. Usługa telemetryczna (MQXR) ustawia nazwę tematu na nazwę kolejki zdalnej.

Więcej informacji na temat sposobu konfigurowania produktu IBM MQ do wysyłania komunikatów bezpośrednio do klientów MQTT zawiera sekcja [Wysyłanie komunikatu bezpośrednio do klienta](#).

Klient MQTT publikuje komunikat

Klient MQTT v3 może opublikować komunikat otrzymany przez inny klient MQTT v3 , ale nie może wysłać niezamówionego komunikatu. Fragment kodu Rysunek 48 na stronie 126 przedstawia sposób, w jaki klient MQTT v3 zapisany w produkcie Javapublikuje komunikat.

Typowy wzorzec wysyłania komunikatu do jednego konkretnego klienta MQTT v3 jest przeznaczony dla każdego klienta w celu utworzenia subskrypcji dla własnego `ClientIdentifier`. Wykonaj zadanie “Publikowanie komunikatu w konkretnym kliencie MQTT v3” na stronie 132 , aby opublikować komunikat z jednego klienta MQTT do innego klienta MQTT przy użyciu parametru `ClientIdentifier` jako łańcucha tematu.

Przykładowe fragmenty kodu

Fragment kodu w programie Rysunek 45 na stronie 125 pokazuje, w jaki sposób klient MQTT napisany w produkcie Java tworzy subskrypcję. Potrzebna jest również metoda zwrotna, `messageArrived` , aby otrzymywać publikacje dotyczące subskrypcji.

```
String    clientId = String.format("%-23.23s",
                                System.getProperty("user.name") + "_" +
                                (UUID.randomUUID().toString()).trim()).replace('-', '_');
MqttClient client = new MqttClient("localhost", clientId);
String topicString = "MQTT Examples";
int      QoS = 1;
client.subscribe(topicString, QoS);
```

Rysunek 45. Subskrybent klienta produktu MQTT v3

Fragment kodu w programie Rysunek 46 na stronie 126 pokazuje, w jaki sposób aplikacja IBM MQ napisana w języku C wysyła publikację. Fragment kodu jest wyodrębniany z zadania, [Utwórz publikator w temacie o zmiennej](#)

```

/* Define and set variables to defaults */
/* Omitted lines declaring variables */
char * topicName = ""
char * topicString = "MQTT Examples"
char * publication = "Hello world!";
do {
    MQCONN(qMgrName, &Hconn, &CompCode, &Reason);
    if (CompCode != MQCC_OK) break;
    td.ObjectType = MQOT_TOPIC; /* Object is a topic */
    td.Version = MQOD_VERSION_4; /* Descriptor needs to be V4 */
    strncpy(td.ObjectName, topicName, MQ_TOPIC_NAME_LENGTH);
    td.ObjectString.VSPtr = topicString;
    td.ObjectString.VSLength = (MQLONG)strlen(topicString);
    MQOPEN(Hconn, &td, MQOO_OUTPUT | MQOO_FAIL_IF QUIESCING, &Hobj, &CompCode, &Reason);
    if (CompCode != MQCC_OK) break;
    pmo.Options = MQPMO_FAIL_IF QUIESCING | MQPMO_RETAIN;
    MQPUT(Hconn, Hobj, &md, &pmo, (MQLONG)strlen(publication)+1, publication, &CompCode, &Reason);
    if (CompCode != MQCC_OK) break;
    MQCLOSE(Hconn, &Hobj, MQCO_NONE, &CompCode, &Reason);
    if (CompCode != MQCC_OK) break;
    MQDISC(&Hconn, &CompCode, &Reason);
} while (0);

```

Rysunek 46. IBM MQ publikator

When the publication arrives, the MQTT client calls the `messageArrived` method of the MQTT application client `MqttCallback` class.

```

public class Callback implements MqttCallback {
    public void messageArrived(MqttTopic topic, MqttMessage message) {
        try {
            System.out.println("Message arrived: \"" + message.toString()
                + "\" on topic \"" + topic.toString() + "\"");
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
// ... Other callback methods

```

Rysunek 47. Metoda `messageArrived`

Rysunek 48 na stronie 126 przedstawia MQTT v3 publikując komunikat do subskrypcji utworzonej w produkcie Rysunek 45 na stronie 125.

```

String address = "localhost";
String clientId = String.format("%-23.23s",
    System.getProperty("user.name") + "-" +
    (UUID.randomUUID().toString()).trim()).replace('-', '_');
MqttClient client = new MqttClient(address, clientId);
String topicString = "MQTT Examples";
MqttTopic topic = client.getTopic(Example.topicString);
String publication = "Hello world";
MqttMessage message = new MqttMessage(publication.getBytes());
MqttDeliveryToken token = topic.publish(message);

```

Rysunek 48. Publikator klienta produktu MQTT v3

Windows Linux AIX **Publikowanie komunikatu w programie narzędziowym klienta MQTT z programu IBM MQ Explorer**

Wykonaj kroki opisane w tym zadaniu, aby opublikować komunikat przy użyciu programu IBM MQ Explorer i zasubskrybuj go za pomocą programu narzędziowego klienta MQTT. Dodatkowe zadanie

przedstawia sposób konfigurowania aliasu menedżera kolejek zamiast ustawiania domyślnej kolejki transmisji na wartość `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Zanim rozpoczniesz

W ramach zadania założono, że użytkownik zna IBM MQ i IBM MQ Explorer oraz że opcja IBM MQ i MQ Telemetry są zainstalowane.

Użytkownik tworzący zasoby menedżera kolejek dla tego zadania musi mieć wystarczające uprawnienia do wykonania tego zadania. W celach demonstracyjnych przyjmuje się, że ID użytkownika IBM MQ Explorer jest członkiem grupy `mqm`.

O tym zadaniu

W ramach zadania tworzony jest temat w produkcie IBM MQ i zasubskrybowanie tematu przy użyciu programu narzędziowego klienta MQTT. Po opublikowaniu w temacie za pomocą IBM MQ Explorer klient MQTT otrzymuje publikację.

Procedura

Wykonaj jedną z następujących czynności:

- Produkt MQ Telemetry został zainstalowany, ale jeszcze nie został on uruchomiony. Wykonaj zadanie: [“Zadanie uruchomienia bez zdefiniowanego jeszcze usługi telemetrycznej \(MQXR\)”](#) na stronie 127.
- Wcześniej uruchomiono telemetry IBM MQ, ale chcesz użyć nowego menedżera kolejek, aby wykonać demonstrację. Wykonaj zadanie: [“Zadanie uruchomienia bez zdefiniowanego jeszcze usługi telemetrycznej \(MQXR\)”](#) na stronie 127.
- Użytkownik chce wykonać to zadanie przy użyciu istniejącego menedżera kolejek, który nie ma zdefiniowanych zasobów pomiarowych. Nie ma potrzeby uruchamiania kreatora **Definiowanie przykładowej konfiguracji**.
 - a. Aby skonfigurować telemetry, wykonaj jedną z następujących czynności:
 - [Konfigurowanie menedżera kolejek dla telemetry w systemach Linux i AIX](#)
 - [Konfigurowanie menedżera kolejek dla telemetry w systemie Windows](#)
 - b. Wykonaj zadanie: [“Uruchom zadanie z działającą usługą telemetryczną \(MQXR\)”](#) na stronie 128
- Aby wykonać to zadanie przy użyciu istniejącego menedżera kolejek, dla którego zdefiniowano już zasoby telemetryczne, należy wykonać następujące czynności: [“Uruchom zadanie z działającą usługą telemetryczną \(MQXR\)”](#) na stronie 128.

Co dalej

Wykonaj [“Wysyłanie wiadomości do klienta MQTT za pomocą IBM MQ Explorer”](#) na stronie 131, aby wysłać komunikat bezpośrednio do programu narzędziowego klienta.

Zadanie uruchomienia bez zdefiniowanego jeszcze usługi telemetrycznej (MQXR)

Utwórz menedżer kolejek i uruchom opcję **Definiowanie przykładowej konfiguracji**, aby zdefiniować przykładowe zasoby telemetryczne dla menedżera kolejek. Opublikuj komunikat przy użyciu programu IBM MQ Explorer i zasubskrybuj go za pomocą programu narzędziowego klienta MQTT.

O tym zadaniu

Podczas konfigurowania przykładowych zasobów telemetrycznych przy użyciu opcji **Definiowanie przykładowej konfiguracji** kreator ustawia uprawnienia identyfikatora użytkownika gościa. Należy dokładnie rozważyć, czy identyfikator użytkownika gościa ma być autoryzowany w ten sposób. `guest` w systemach Windowsi `nobody` w systemie Linux mają nadane uprawnienia do publikowania i subskrybowania katalogu głównego drzewa tematów oraz do umieszczania komunikatów w produkcie `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Kreator ustawia także domyślną kolejkę transmisji na SYSTEM.MQTT.TRANSMIT.QUEUE, co może kolidować z aplikacjami działającymi w istniejącym menedżerze kolejek. Jest to możliwe, ale pracochłonne, aby skonfigurować telemetrię i nie korzystać z domyślnej kolejki transmisji; należy wykonać następujące czynności: “Korzystanie z aliasu menedżera kolejek” na stronie 130. W tym zadaniu należy utworzyć menedżer kolejek, aby uniknąć możliwości ingerowania w żadną istniejącą domyślną kolejkę transmisji.

Procedura

1. Za pomocą programu IBM MQ Explorer utwórz i uruchom nowy menedżer kolejek.
 - a) Kliknij prawym przyciskiem myszy folder Queue Managers > **Nowy > Menedżer kolejek** Wpisz nazwę menedżera kolejek > **Zakończ.**
Utwórz nazwę menedżera kolejek, na przykład MQTTQMGR.
2. Utwórz i uruchom usługę telemetryczną (MQXR) i utwórz przykładowy kanał telemetryczny.
 - a) Otwórz folder Queue Managers\QmgrName\Telemetry.
 - b) Kliknij opcję **Zdefiniuj przykładową konfigurację ... > Zakończ.**
Pozostaw zaznaczone pole wyboru **Uruchom program narzędziowy klienta MQTT.**
3. Utwórz subskrypcję dla produktu MQTT Example, korzystając z programu narzędziowego klienta MQTT.
 - a) Kliknij przycisk **Połącz.**
Historia klienta rejestruje zdarzenie Connected.
 - b) Wpisz MQTT Example w polu **Subskrypcja \ Temat > Subskrybuj.**
Historia klienta rejestruje zdarzenie Subscribed.
4. Utwórz MQTTExampleTopic w IBM MQ.
 - a) Kliknij prawym przyciskiem myszy folder Queue Managers\QmgrName\Topics w **EksploatörzeMQ > Nowy > Temat.**
 - b) Wpisz MQTTExampleTopic jako **Nazwa > Dalej.**
 - c) Wpisz MQTT Example jako **Łańcuch tematu > Zakończ.**
 - d) Kliknij przycisk **OK**, aby zamknąć okno potwierdzenia.
5. Opublikuj Hello World! w temacie MQTT Example, używając IBM MQ Explorer.
 - a) Kliknij folder Queue Managers\QmgrName\Topics w IBM MQ Explorer.
 - b) Kliknij prawym przyciskiem myszy opcję MQTTExampleTopic > **Testuj publikację ...**
 - c) Wpisz Hello World! w polu **Dane komunikatu > Publikuj komunikat > Przetłącz się do okna programu narzędziowego klienta produktu MQTT.**
Historia klienta rejestruje zdarzenie Received.

Uruchom zadanie z działającą usługą telemetryczną (MQXR)

Utwórz kanał pomiarowy i temat. Autoryzuj użytkownika do korzystania z tematu i kolejki przesyłania danych telemetrycznych. Opublikuj komunikat przy użyciu programu IBM MQ Explorer i zasubskrybuj go za pomocą programu narzędziowego klienta MQTT.

Zanim rozpoczniesz

W tej wersji zadania jest zdefiniowany i uruchomiony menedżer kolejek QmgrName. Usługa telemetryczna (MQXR) jest zdefiniowana i uruchomiona. Usługa telemetryczna (MQXR) mogła zostać utworzona ręcznie lub przez uruchomienie kreatora **Definiowanie przykładowej konfiguracji.**

O tym zadaniu

W ramach tej czynności można skonfigurować istniejący menedżer kolejek w celu wystania publikacji do programu narzędziowego klienta MQTT.

Krok "1" na stronie 129 zadania ustawia domyślną kolejkę transmisji na SYSTEM.MQTT.TRANSMIT.QUEUE, co może kolidować z aplikacjami działającymi w istniejącym menedżerze kolejek. Jest to możliwe, ale pracochłonne, aby skonfigurować telemetrię i nie korzystać z domyślnej kolejki transmisji; należy wykonać następujące czynności: "Korzystanie z aliasu menedżera kolejek" na stronie 130.

Procedura

1. Ustaw SYSTEM.MQTT.TRANSMIT.QUEUE jako domyślną kolejkę wyjściową.
 - a) Kliknij prawym przyciskiem myszy opcję Queue Managers*QmgrName* folder > **Właściwości ...**
 - b) Kliknij opcję **Komunikacja** w nawigatorze.
 - c) Kliknij przycisk **Wybierz ...** > Wybierz opcję SYSTEM.MQTT.TRANSMIT.QUEUE > **OK** > **OK**.
2. Utwórz kanał pomiarowy MQTTExampleChannel, aby połączyć program narzędziowy klienta MQTT z serwerem IBM MQ, a następnie uruchom program narzędziowy klienta MQTT.
 - a) Kliknij prawym przyciskiem myszy folder Queue Managers*QmgrName* \Telemetry\Channels w **EksploratorzeMQ** > **Nowy** > **Kanał pomiarowy**
 - b) Wpisz MQTTExampleChannel w polu **Nazwa kanału** > **Dalej** > **Dalej**.
 - c) Zmień wartość w polu **Stały identyfikator użytkownika** na panelu autoryzacji klienta na identyfikator użytkownika, który ma zostać opublikowany, i zasubskrybuj MQTTExample > **Dalej**.
 - d) Pozostaw zaznaczone pole wyboru **Uruchom program narzędziowy klienta** > **Zakończ**.
3. Utwórz subskrypcję dla produktu MQTT Example, korzystając z programu narzędziowego klienta MQTT.
 - a) Kliknij przycisk **Połącz**.

Historia klienta rejestruje zdarzenie Connected.
 - b) Wpisz MQTT Example w polu **Subskrypcja \ Temat** > **Subskrybuj**.

Historia klienta rejestruje zdarzenie Subscribed.
4. Utwórz MQTTExampleTopic w IBM MQ.
 - a) Kliknij prawym przyciskiem myszy folder Queue Managers*QmgrName*\Topics w **EksploratorzeMQ** > **Nowy** > **Temat**.
 - b) Wpisz MQTTExampleTopic jako **Nazwa** > **Dalej**.
 - c) Wpisz MQTT Example jako **Łańcuch tematu** > **Zakończ**.
 - d) Kliknij przycisk **OK**, aby zamknąć okno potwierdzenia.
5. Aby opublikować i zasubskrybować temat MQTTExample, aby użytkownik nie był w grupie mqm, wykonaj następujące czynności:
 - a) Autoryzuj użytkownika do publikowania i subskrybowania tematu MQTTExampleTopic:

```
setmqaut -m qMgrName -t topic -n MQTTExampleTopic -p User ID -all +pub +sub
```
 - b) Autoryzuj użytkownika w celu umieszczenia komunikatu na serwerze SYSTEM.MQTT.TRANSMIT.QUEUE:

```
setmqaut -m qMgrName -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p User ID -all +put
```
6. Opublikuj Hello World! w temacie MQTT Example, używając IBM MQ Explorer.
 - a) Kliknij folder Queue Managers*QmgrName*\Topics w IBM MQ Explorer.
 - b) Kliknij prawym przyciskiem myszy opcję MQTTExampleTopic > **Testuj publikację ...**
 - c) Wpisz Hello World! w polu **Dane komunikatu** > **Publikuj komunikat** > Przełącz się do okna programu narzędziowego klienta produktu MQTT.

Historia klienta rejestruje zdarzenie Received.

Korzystanie z aliasu menedżera kolejek

Opublikuj komunikat w programie narzędziowym klienta MQTT przy użyciu programu IBM MQ Explorer bez ustawiania domyślnej kolejki transmisji na wartość `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Zadanie jest kontynuacją poprzedniego zadania i korzysta z aliasu menedżera kolejek w celu uniknięcia ustawienia domyślnej kolejki transmisji na `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Zanim rozpoczniesz

Wykonaj zadanie “Zadanie uruchomienia bez zdefiniowanego jeszcze usługi telemetrycznej (MQXR)” na stronie 127 lub zadanie “Uruchom zadanie z działającą usługą telemetryczną (MQXR)” na stronie 128.

O tym zadaniu

Gdy klient MQTT tworzy subskrypcję, produkt IBM MQ wysyła swoją odpowiedź za pomocą `ClientIdentifier`, jako nazwy zdalnego menedżera kolejek. W ramach tego zadania używany jest parametr `ClientIdentifier`, `MyClient`.

Jeśli nie istnieje kolejka transmisji lub alias menedżera kolejek o nazwie `MyClient`, odpowiedź jest umieszczana w domyślnej kolejce transmisji. Ustawienie domyślnej kolejki transmisji na wartość `SYSTEM.MQTT.TRANSMIT.QUEUE` powoduje, że klient MQTT otrzyma odpowiedź.

Można uniknąć ustawienia domyślnej kolejki transmisji na `SYSTEM.MQTT.TRANSMIT.QUEUE` przy użyciu aliasów menedżera kolejek. Należy skonfigurować alias menedżera kolejek dla każdego elementu `ClientIdentifier`. Zwykle jest zbyt wiele klientów, aby można było używać aliasów menedżera kolejek. Często `ClientIdentifier` jest nieprzewidywalny, co uniemożliwia skonfigurowanie telemetrii w ten sposób.

Mimo to w niektórych sytuacjach może być konieczne skonfigurowanie domyślnej kolejki transmisji na inną niż `SYSTEM.MQTT.TRANSMIT.QUEUE`. Kroki opisane w sekcji Procedura konfiguruje alias menedżera kolejek zamiast ustawiania domyślnej kolejki transmisji na wartość `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Procedura

1. Usuń `SYSTEM.MQTT.TRANSMIT.QUEUE` jako domyślną kolejkę wyjściową.
 - a) Kliknij prawym przyciskiem myszy opcję `Queue Managers\QmgrName folder > Właściwości ...`
 - b) Kliknij opcję **Komunikacja** w nawigatorze.
 - c) Usuń element `SYSTEM.MQTT.TRANSMIT.QUEUE` z pola **Domyślna kolejka transmisji > OK**.
2. Upewnij się, że nie można już utworzyć subskrypcji za pomocą programu narzędziowego klienta MQTT :
 - a) Kliknij przycisk **Połącz**.
Historia klienta rejestruje zdarzenie `Connected` .
 - b) Wpisz `MQTT Example` w polu **Subskrypcja \ Temat > Subskrybuj**.
Historia klienta rejestruje zdarzenie `Subscribe failed iConnection lost` .
3. Utwórz alias menedżera kolejek dla parametru `ClientIdentifier`, `MyClient`.
 - a) Kliknij prawym przyciskiem myszy folder `Queue Managers\QmgrName\Queues > Nowy > Definicja kolejki zdalnej`.
 - b) Nazwij definicję, `MyClient > Dalej`.
 - c) Wpisz `MyClient` w polu **Menedżer kolejek zdalnych** .
 - d) Wpisz `SYSTEM.MQTT.TRANSMIT.QUEUE` w polu **Kolejka transmisji > Zakończ**.
4. Ponownie połącz program narzędziowy klienta MQTT .
 - a) Sprawdź, czy w polu **Identyfikator klienta** ustawiona jest wartość `MyClient`.
 - b) **Połączenie**

Historia klienta rejestruje zdarzenie Connected .

5. Utwórz subskrypcję dla produktu MQTT Example , korzystając z programu narzędziowego klienta MQTT .

- a) Kliknij przycisk **Połącz**.

Historia klienta rejestruje zdarzenie Connected .

- b) Wpisz MQTT Example w polu **Subskrypcja \ Temat > Subskrybuj**.

Historia klienta rejestruje zdarzenie Subscribed .

6. Opublikuj Hello World! w temacie MQTT Example , używając IBM MQ Explorer.

- a) Kliknij folder Queue Managers \ QmgrName \ Topics w IBM MQ Explorer.

- b) Kliknij prawym przyciskiem myszy opcję MQTTExampleTopic > **Testuj publikację ...**

- c) Wpisz Hello World! w polu **Dane komunikatu > Publikuj komunikat > Przełącz się do okna programu narzędziowego klienta produktu MQTT** .

Historia klienta rejestruje zdarzenie Received .

Windows Linux AIX Wysłanie wiadomości do klienta MQTT za pomocą IBM MQ Explorer

Wyślij komunikat do programu narzędziowego klienta MQTT , umieszczając komunikat w kolejce produktu IBM MQ przy użyciu produktu IBM MQ Explorer. W ramach zadania przedstawiono sposób konfigurowania definicji kolejki zdalnej w celu wysłania komunikatu bezpośrednio do klienta MQTT .

Zanim rozpoczniesz

Wykonaj zadanie [“Publikowanie komunikatu w programie narzędziowym klienta MQTT z programu IBM MQ Explorer”](#) na stronie 126. Pozostaw połączony program narzędziowy klienta MQTT .

O tym zadaniu

Zadanie demonstruje wysłanie komunikatu do klienta MQTT przy użyciu kolejki, a nie do publikowania w temacie. W kliencie nie jest tworzona subskrypcja. Krok [“2”](#) na stronie 131 zadania pokazuje, że poprzednia subskrypcja została usunięta.

Procedura

1. Usuń wszystkie istniejące subskrypcje, rozłączając i ponownie łącząc program narzędziowy klienta MQTT .

Subskrypcja jest odrzucana, ponieważ jeśli nie zostanie zmieniona wartość domyślna, program narzędziowy klienta MQTT połączy się z czystą sesją. Patrz sekcja [Czyszczenie sesji](#).

Aby ułatwić wykonanie zadania, należy wpisać własny identyfikator ClientIdentifier, a nie wygenerowany przez program narzędziowy klienta MQTT wygenerowany obiekt ClientIdentifier .

- a) Kliknij przycisk **Rozłącz** , aby odłączyć program narzędziowy klienta MQTT od kanału pomiarowego.

Historia klienta rejestruje zdarzenie produktu Disconnected .

- b) Zmień wartość parametru **Client Identifier** na MyClient.

- c) Kliknij przycisk **Połącz**.

Historia klienta rejestruje zdarzenie produktu Connected .

2. Sprawdź, czy program narzędziowy klienta MQTT nie otrzymuje już publikacji dla MQTTExampleTopic.

- a) Kliknij folder Queue Managers \ QmgrName \ Topics w IBM MQ Explorer.

- b) Kliknij prawym przyciskiem myszy opcję MQTTExampleTopic > **Testuj publikację ...**

c) Wpisz Hello World! w polu **Dane komunikatu** > **Publikuj komunikat** > Przełącz się do okna programu narzędziowego klienta produktu MQTT .

Żadne zdarzenie nie jest rejestrowane w **historii klienta**.

3. Utwórz definicję kolejki zdalnej dla klienta.

Ustaw parametr ClientIdentifier, MyClient, jako nazwę zdalnego menedżera kolejek w definicji kolejki zdalnej. Użyj dowolnej nazwy, która ma być nazwą kolejki zdalnej. Nazwa kolejki zdalnej jest przekazywana do klienta MQTT jako nazwa tematu.

a) Kliknij prawym przyciskiem myszy folder Queue Managers\QmgrName\Queues > **Nowy** > **Definicja kolejki zdalnej**.

b) Nazwij definicję, MyClientRemoteQueue > **Dalej**.

c) Wpisz MQTTExampleQueue w polu **Kolejka zdalna** .

d) Wpisz MyClient w polu **Menedżer kolejek zdalnych** .

e) Wpisz SYSTEM.MQTT.TRANSMIT.QUEUE w polu **Kolejka transmisji** > **Zakończ**.

4. Umieść komunikat testowy na serwerze MyClientRemoteQueue.

a) Kliknij prawym przyciskiem myszy opcję **MyClientRemoteQueue** > **Umieść komunikat testowy ...**

b) Wpisz Hello queue! w polu Dane komunikatu > **Umieść komunikat** > **Zamknij** .

Historia klienta rejestruje zdarzenie Received .

5. Usuń SYSTEM.MQTT.TRANSMIT.QUEUE jako domyślną kolejkę wyjściową.

a) Kliknij prawym przyciskiem myszy opcję Queue Managers\QmgrName folder > **Właściwości ...**

b) Kliknij opcję **Komunikacja** w nawigаторze.

c) Usuń element SYSTEM.MQTT.TRANSMIT.QUEUE z pola **Domyślna kolejka transmisji** > **OK**.

6. Powtórz krok "4" na stronie 132.

MyClientRemoteQueue jest definicją kolejki zdalnej, która jawnie określa nazwę kolejki transmisji. Aby wysłać komunikat do programu MyClient, nie jest konieczne definiowanie domyślnej kolejki transmisji.

Co dalej

Jeśli domyślna kolejka transmisji nie jest już ustawiona na wartość SYSTEM.MQTT.TRANSMIT.QUEUE, program narzędziowy klienta MQTT nie może utworzyć nowej subskrypcji, jeśli nie zostanie zdefiniowany alias menedżera kolejek dla parametru ClientIdentifier MyClient. Odtwórz domyślną kolejkę transmisji do programu SYSTEM.MQTT.TRANSMIT.QUEUE.

Windows Linux AIX Publikowanie komunikatu w konkretnym kliencie MQTT v3

Opublikuj komunikat z jednego klienta MQTT v3 na inny, używając ClientIdentifier jako nazwy tematu i IBM MQ jako brokera publikowania/subskrypcji.

Zanim rozpoczniesz

Wykonaj zadanie "Publikowanie komunikatu w programie narzędziowym klienta MQTT z programu IBM MQ Explorer" na stronie 126. Pozostaw połączony program narzędziowy klienta MQTT .

O tym zadaniu

Zadanie demonstruje dwie rzeczy:

1. Subskrybowanie tematu w jednym kliencie MQTT i odbieranie publikacji z innego klienta MQTT .
2. Konfigurowanie subskrypcji "punkt z punktem" za pomocą parametru ClientIdentifier jako łańcucha tematu.

Procedura

1. Usuń wszystkie istniejące subskrypcje, rozłączając i ponownie łącząc program narzędziowy klienta MQTT .

Subskrypcja jest odrzucana, ponieważ jeśli nie zostanie zmieniona wartość domyślna, program narzędziowy klienta MQTT połączy się z czystą sesją. Patrz sekcja [Czyszczenie sesji](#).

Aby ułatwić wykonanie zadania, należy wpisać własny identyfikator `ClientIdentifier`, a nie wygenerowany przez program narzędziowy klienta MQTT wygenerowany obiekt `ClientIdentifier` .

- a) Kliknij przycisk **Rozłącz** , aby odłączyć program narzędziowy klienta MQTT od kanału pomiarowego.

Historia klienta rejestruje zdarzenie produktu `Disconnected` .

- b) Zmień wartość parametru **Client Identifier** na `MyClient`.
- c) Kliknij przycisk **Połącz**.

Historia klienta rejestruje zdarzenie produktu `Connected` .

2. Utwórz subskrypcję tematu, `MyClient`

`MyClient` jest to `ClientIdentifier` tego klienta.

- a) Wpisz `MyClient` w polu **Subskrypcja \ Temat > Subskrybuj**.

Historia klienta rejestruje zdarzenie `Subscribed` .

3. Uruchom inny program narzędziowy klienta MQTT .

- a) Otwórz folder `Queue Managers\QmgrName\Telemetry\channels` .

- b) Kliknij prawym przyciskiem myszy kanał **PlainText > Uruchom program narzędziowy klienta MQTT ...**

- c) Kliknij przycisk **Połącz**.

Historia klienta rejestruje zdarzenie produktu `Connected` .

4. Opublikuj `Hello MyClient!` w temacie `MyClient`.

- a) Skopiuj temat subskrypcji `MyClient` z programu narzędziowego klienta MQTT działającego z identyfikatorem `ClientIdentifier MyClient`.

- b) Wklej plik `MyClient` do pola **Publication \ Topic** (Publikacja) w każdej instancji programu narzędziowego klienta produktu MQTT .

- c) Wpisz `Hello MyClient!` w polu **Publikacja \ komunikatu** .

- d) W obu instancjach kliknij opcję **Publikuj** .

Wyniki

Historia klienta w programie narzędziowym klienta MQTT z atrybutem `ClientIdentifier`, `MyClient`, rejestruje dwa zdarzenia **Odebrane** i jedno zdarzenie **Opublikowane** . Inna instancja programu narzędziowego klienta MQTT rejestruje jedno zdarzenie **Opublikowane** .

Jeśli zostanie wyświetlona tylko jedna **odebrana** zdarzenie, sprawdź, czy możliwe jest następujące przyczyny:

1. Czy domyślna kolejka transmisji dla menedżera kolejek jest ustawiona na `SYSTEM.MQTT.TRANSMIT.QUEUE` ?
2. Czy w trakcie wykonywania innych ćwiczeń utworzono aliasy menedżera kolejek lub definicje kolejek zdalnych odwołujące się do produktu `MyClient` ? W przypadku wystąpienia problemu z konfiguracją należy usunąć wszelkie zasoby, które odwołują się do produktu `MyClient`, takie jak aliasy menedżera kolejek lub kolejki transmisji. Odłącz programy narzędziowe klienta, zatrzymaj i zrestartuj usługę telemetryczną (MQXR).

z klienta MQTT

Aplikacja IBM MQ może odbierać komunikat z klienta MQTT v3 poprzez subskrybowanie tematu. Klient MQTT łączy się z produktem IBM MQ za pomocą kanału pomiarowego, a następnie wysyła komunikat do aplikacji IBM MQ poprzez opublikowanie go w tym samym temacie.

Wykonaj zadanie [“Publikowanie komunikatu w produkcie IBM MQ z poziomu klienta produktu MQTT” na stronie 134](#), aby dowiedzieć się, jak wysłać publikację z klienta produktu MQTT do subskrypcji zdefiniowanej w produkcie IBM MQ.

Jeśli temat jest zgrupowany lub dystrybuowany przy użyciu hierarchii publikowania/subskrypcji, subskrypcja może znajdować się w innym menedżerze kolejek, do menedżera kolejek, z którym połączony jest klient MQTT.

z poziomu klienta produktu MQTT

Utworzenie subskrypcji tematu za pomocą produktu IBM MQ Explorer i opublikowanie go w tym temacie przy użyciu programu narzędziowego klienta MQTT.

Zanim rozpoczniesz

Wykonaj zadanie [“Publikowanie komunikatu w programie narzędziowym klienta MQTT z programu IBM MQ Explorer” na stronie 126](#). Pozostaw połączony program narzędziowy klienta MQTT.

O tym zadaniu

Zadanie demonstruje publikowanie komunikatu przy użyciu klienta MQTT i odbieranie publikacji przy użyciu niezarządzanej trwałej subskrypcji utworzonej przy użyciu produktu IBM MQ Explorer.

Procedura

1. Utwórz trwałą subskrypcję łańcucha tematu MQTT Example.
Wykonaj następujące kroki, aby utworzyć kolejkę i subskrypcję przy użyciu produktu IBM MQ Explorer.
 - a) Kliknij prawym przyciskiem myszy folder Queue Managers*QmgrName*\Queues w oknie IBM MQ Explorer > **Nowy > Kolejka lokalna**
 - b) Wpisz MQTTExampleQueue jako nazwę kolejki > **Zakończ.**
 - c) Kliknij prawym przyciskiem myszy folder Queue Managers*QmgrName*\Subscriptions w oknie IBM MQ Explorer > **Nowy > Subskrypcja**
 - d) Wpisz MQTTExampleSubscription jako nazwę kolejki > **Dalej.**
 - e) Kliknij przycisk **Wybierz ...** > MQTTExampleTopic > **OK.**

You have already created the topic, MQTTExampleTopic in step “4” na stronie 128 of [“Publikowanie komunikatu w programie narzędziowym klienta MQTT z programu IBM MQ Explorer” na stronie 126.](#)
 - f) Wpisz MQTTExampleQueue jako nazwę miejsca docelowego > **Zakończ.**
2. W ramach opcjonalnego kroku ustaw kolejkę do użycia przez innego użytkownika, bez uprawnień mqm.

W przypadku konfigurowania konfiguracji dla użytkowników posiadających uprawnienia mniejsze niż mqm, należy nadać uprawnienie put i get do produktu MQTTExampleQueue. Dostęp do tematu i do kolejki transmisji został skonfigurowany w produkcie [“Publikowanie komunikatu w programie narzędziowym klienta MQTT z programu IBM MQ Explorer” na stronie 126.](#)

 - a) Autoryzuj użytkownika do umieszczania i pobierania do kolejki MQTTExampleQueue:

```
setmqaut -m qMgrName -t queue -n MQTTExampleQueue -p User ID -all +put +get
```

3. Opublikuj Hello IBM MQ! w temacie MQTT Example przy użyciu programu narzędziowego klienta MQTT .

Jeśli nie zostało pozostawione połączenie z programem narzędziowym klienta MQTT , kliknij prawym przyciskiem myszy kanał **PlainText > Uruchom program narzędziowy klienta MQTT ... > Połącz**.

- a) Wpisz MQTT Example w polu **Publikacja \ Temat** .
 - b) Wpisz Hello IBM MQ! w polu **Publikacja \ Komunikat > Publikuj**.
4. Otwórz folder Queue Managers\QmgrName\Queues i znajdź MQTTExampleQueue.
Pole **Bieżące zapełnienie kolejki** to 1
 5. Kliknij prawym przyciskiem myszy opcję MQTTExampleQueue > **Przeglądaj komunikaty ...** i zapoznaj się z publikacją.

Windows

Linux

AIX

Aplikacje publikowania/subskrypcji produktu

MQTT

Aby napisać aplikacje produktu MQTT , należy użyć publikowania/subskrybowania opartego na temacie.

Gdy klient MQTT jest połączony, publikacje przepływa albo w obu kierunkach, między klientem a serwerem. Publikacje są wysyłane z klienta, gdy informacje są publikowane na kliencie. Publikacje są odbierane przez klienta po opublikowaniu komunikatu w temacie, który jest zgodny z subskrypcją utworzoną przez klienta.

Broker publikowania/subskrybowania produktu IBM MQ zarządza tematami i subskrypcjami utworzonymi przez klienty MQTT . Tematy utworzone przez klienty MQTT współużytkują ten sam obszar tematów, co tematy utworzone przez aplikacje produktu IBM MQ .

Publikacje zgodne z łańcuchem tematu w subskrypcji klienta produktu MQTT są umieszczane w produkcie SYSTEM.MQTT.TRANSMIT.QUEUE z nazwą zdalnego menedżera kolejek ustawionym na wartość ClientIdentifier klienta. Usługa telemetryczna (MQXR) przekazuje publikacje do klienta, który utworzył subskrypcję. Używany jest parametr ClientIdentifier, który został ustawiony jako nazwa zdalnego menedżera kolejek w celu identyfikacji klienta.

Zwykle wartość SYSTEM.MQTT.TRANSMIT.QUEUE musi być zdefiniowana jako domyślna kolejka transmisji. Możliwe jest, ale uciążliwe, skonfigurowanie produktu MQTT , aby nie używało domyślnej kolejki transmisji. Patrz sekcja [Konfigurowanie rozproszonego kolejkowania w celu wysyłania komunikatów do klientów MQTT](#).

Klient MQTT może utworzyć sesję trwałą; patrz [“MQTT -bezstanowe i stanowe sesje”](#) na stronie 139. Subskrypcje utworzone w trwałej sesji są trwałe. Publikacje dostarczane dla klienta z trwałą sesją są zapisywane w produkcie SYSTEM.MQTT.TRANSMIT.QUEUEi przekazywane do klienta po ponownym nawiązaniu połączenia.

Klient MQTT może również publikować i subskrybować zachowane publikacje. Patrz sekcja [Zachowane publikacje i klienty MQTT](#). Subskrybent zachowanego tematu publikacji otrzymuje najnowszą publikację do tematu. Subskrybent otrzymuje zachowaną publikację, gdy tworzy subskrypcję, lub po ponownym nawiązaniu połączenia z wcześniejszą sesją.

Windows

Linux

AIX

Aplikacje telemetryczne

Napisz aplikacje telemetryczne przy użyciu przepływów komunikatów IBM MQ lub IBM Integration Bus .

Użyj JMS, interfejsu MQI lub innych interfejsów programistycznych produktu IBM MQ , aby programować aplikacje telemetryczne w produkcie IBM MQ.

Usługa telemetryczna (MQXR) konwertuje między komunikatami MQTT v3 i komunikatami IBM MQ . Tworzy on subskrypcje i publikacje w imieniu klientów MQTT i przekazuje publikacje do klientów MQTT . Publikacja jest ładunkiem komunikatu MQTT v3 . Ładunek składa się z nagłówek komunikatów i tablicy

bajtów w formacie `jms-bytes`. Serwer telemetryczny odwzorowuje nagłówki między komunikatem MQTT v3 i komunikatem IBM MQ; patrz [“Integracja produktu MQ Telemetry z menedżerami kolejek”](#) na stronie 136.

Węzły `Publication`, `MQInput` i `JMSInput` są używane do wysyłania i odbierania publikacji między klientami IBM Integration Bus i MQTT.

Za pomocą przepływow komunikatów można integrować telemetry z serwisami WWW przy użyciu protokołu HTTP, a także z innymi aplikacjami przy użyciu produktów IBM MQ i WebSphere Adapters.

Windows

Linux

AIX

Integracja produktu MQ Telemetry

z menedżerami kolejek

Klient MQTT jest zintegrowany z produktem IBM MQ jako aplikacja publikowania/subskrypcji. Może on publikować lub subskrybować tematy w produkcie IBM MQ, tworzyć nowe tematy lub korzystać z istniejących tematów. Otrzymuje on publikacje z produktu IBM MQ w wyniku działania klientów MQTT, w tym samego siebie, lub innych aplikacji produktu IBM MQ, które publikują tematy swoich subskrypcji. Reguły są stosowane do decydowania o atrybutach publikacji.

Wiele atrybutów powiązanych z tematami, publikacjami, subskrypcjami i komunikatami, które są udostępniane przez produkt IBM MQ, nie jest obsługiwane. [“Klient MQTT do brokera publikowania/subskrypcji produktu IBM MQ”](#) na stronie 136 i [“IBM MQ do klienta MQTT”](#) na stronie 138 opisują, w jaki sposób są ustawiane atrybuty publikacji. Ustawienia zależą od tego, czy publikacja ma być czy z brokera publikowania/subskrybowania produktu IBM MQ.

W produkcie IBM MQ tematy publikowania/subskrypcji są powiązane z obiektami tematu administracyjnego. Tematy utworzone przez klienty MQTT nie różnią się od siebie. Gdy klient MQTT tworzy łańcuch tematu dla publikacji, broker publikowania/subskrypcji produktu IBM MQ powiąże go z obiektem tematu administracyjnego. Broker odwzorowuje łańcuch tematu w publikacji na najbliższy element nadrzędny obiektu tematu administracyjnego. Odwzorowanie jest takie samo, jak w przypadku aplikacji produktu IBM MQ. Jeśli użytkownik nie utworzył tematu, temat publikowania jest odwzorowywany na `SYSTEM.BASE.TOPIC`. Atrybuty, które są stosowane do publikacji, są uzyskiwane z obiektu tematu.

Gdy aplikacja IBM MQ lub administrator tworzy subskrypcję, nazwa subskrypcji jest nazwana. Wyświetlanie listy subskrypcji przy użyciu programu IBM MQ Explorer lub komend `runmqsc` lub `PCF`. Zostaną nazwane wszystkie subskrypcje klienta MQTT. Otrzymują one nazwę w postaci: `ClientIdentifier:Topic name`

Klient MQTT do brokera publikowania/subskrypcji produktu IBM MQ

Klient MQTT wysłał publikację do produktu IBM MQ. Usługa telemetryczna (MQXR) przekształca publikację w komunikat IBM MQ. Komunikat IBM MQ zawiera trzy części:

1. `MQMD`
2. `RFH2`
3. Komunikat

Właściwości `MQMD` są ustawiane na wartości domyślne, z wyjątkiem sytuacji, w których jest to zaznaczone w [Tabela 9 na stronie 136](#).

| Pole MQMD | Typ | Wartość |
|-----------|---------|-------------------|
| Format | MQCHAR8 | MQFMT_RF_HEADER_2 |

| <i>Tabela 9. Ustawienia właściwości MQMD (kontynuacja)</i> | | |
|--|------------|---|
| Pole MQMD | Typ | Wartość |
| UserIdentifier | MQCHAR12 | Ustaw na jedną z następujących wartości: MqttClient.ClientIdentifier MqttConnectOptions.UserName Identyfikator użytkownika ustawiony przez administratora IBM MQ dla kanału telemetrycznego. |
| Priority | MQLONG | MQPRI_PRIORITY_AS_Q_DEF (wartość domyślna dla IBM MQ, która różni się od JMS , która ma wartość domyślną 4). |
| Persistence | MQLONG | QoS=0→MQPER_NOT_PERSISTENT QoS=1→MQPER_PERSISTENT QoS=2→MQPER_PERSISTENT |

Nagłówek RFH2 nie zawiera folderu <msd> w celu zdefiniowania typu komunikatu JMS . Usługa telemetryczna (MQXR) tworzy komunikat IBM MQ jako domyślny komunikat JMS . Domyślny typ komunikatu JMS to komunikat jms-bytes . Aplikacja może uzyskać dostęp do dodatkowych informacji nagłówka jako właściwości komunikatu. Patrz sekcja [Właściwości komunikatu](#).

Wartości RFH2 są ustawiane w sposób przedstawiony w sekcji [Tabela 10 na stronie 137](#). Właściwość Format jest ustawiona w stałym nagłówku RFH2 , a pozostałe wartości są ustawiane w folderach RFH2 .

| <i>Tabela 10. Ustawienia właściwości RFH2</i> | | |
|---|-------------------|---|
| Właściwość RFH2 | Typ/folder | Nagłówek |
| Formatowanie | MQCHAR8 | MQFMT_NONE |
| ClientIdentifier | mqtt/clientId | Skopiuj element MqttClient.ClientIdentifier o długości 1...23 bajtów. |
| QoS | mqtt/qos | Skopiuj QoS z przychodzącego komunikatu MQTT . |
| ID komunikatu | mqtt/msgid | Skopiuj ID komunikatu z przychodzącego komunikatu MQTT , jeśli QoS ma wartość 1 lub 2. |
| MQIsRetained | mmps/Ret | Ustaw, jeśli oryginalna publikacja MQTT została wysłana razem z zestawem właściwości RETAIN , a komunikat jest odbierany jako zachowana publikacja. |
| MQTopicString | mmps/Top | Temat, do którego opublikowany został komunikat MQTT . |

Ładunek w publikacji produktu MQTT jest odwzorowywany na treść komunikatu produktu IBM MQ :

| <i>Tabela 11. Sposób, w jaki ładunek w publikacji MQTT jest odwzorowywany na treść komunikatu produktu IBM MQ .</i> | | |
|---|-----------------|--|
| Treść wiadomości | Typ | Treść komunikatu IBM MQ |
| Buforuj | MQBYTE <i>n</i> | Kopia bajtów z przychodzącego komunikatu MQTT . Długość może wynosić zero. |

IBM MQ do klienta MQTT

Klient zasubskrybował temat publikowania. Aplikacja produktu IBM MQ została opublikowana w temacie, co spowodowało wystanie publikacji do subskrybenta produktu MQTT przez brokera publikowania/subskrybowania produktu IBM MQ. Alternatywnie aplikacja IBM MQ wysłała niezamówiony komunikat bezpośrednio do klienta MQTT. Tabela 12 na stronie 138 opisuje sposób, w jaki stałe nagłówki komunikatów są ustawiane w komunikacie wysydanym do klienta MQTT. Wszystkie inne dane znajdujące się w nagłówku komunikatu produktu IBM MQ lub w innych nagłówkach są usuwane. Dane komunikatu w komunikacie IBM MQ są wysyłane jako ładunek komunikatu w komunikacie MQTT, bez zmiany. Komunikat MQTT jest wysyłany do klienta MQTT przez usługę telemetryczną (MQXR).

Tabela 12. Sposób ustawiania stałych nagłówków komunikatów w komunikacie IBM MQ wysydanym do klienta MQTT

| MQTT field (pole) | Typ | Wartość |
|-------------------|-------------------------|---|
| DUP | boolean (boolowskie) | Ustaw, jeśli QoS = 1 lub 2, a komunikat został wysłany do tego klienta w poprzedniej transmisji, a komunikat nie został potwierdzony po określonym czasie. |
| QoS | int | <p>Sposób ustawienia wartości QoS w wychodzącej publikacji z brokera publikowania/subskrypcji w produkcie IBM MQ zależy od daty publikacji przychodzącej. Zależy ona od tego, czy przychodząca publikacja została wysłana z klienta MQTT, czy z aplikacji IBM MQ.</p> <p>MQTT Niższa wartość QoS w przychodzącej publikacji oraz w QoS żądanym przez subskrybenta.</p> <p>IBM MQ Niższa wartość parametru QoS pochodząca z przychodzącej publikacji: MQPER_NOT_PERSISTENT→QoS=0 MQPER_PERSISTENT→QoS=2</p> <p>i QoS żądane przez subskrybenta. Jeśli komunikat jest wysyłany do klienta bez subskrypcji, wartość QoS jest ustawiana domyślnie na 2. Klient może zmienić tę wartość, subskrybując wartość DEFAULT. QoS inną wartością QoS.</p> |
| RETAIN | boolean (boolowskie) | Ustaw, jeśli przychodząca publikacja ma zachowany zestaw właściwości. |

Tabela 13 na stronie 138 opisuje sposób, w jaki nagłówki komunikatów zmiennych są ustawiane w komunikacie MQTT, który jest wysyłany do klienta MQTT.

Tabela 13. Sposób ustawiania właściwości nagłówka zmiennej MQTT w komunikacie MQTT wysydanym do klienta MQTT.

| MQTT field (pole) | Typ | Wartość |
|-------------------|---------|--|
| Topic name | łańcuch | łańcuch tematu, z którym został opublikowany komunikat. |
| Message ID | łańcuch | Ostatnie 2 bajty MQMD.MsgId, właściwość publikacji, gdy jest ona umieszczona w produkcie SYSTEM.MQTT.TRANSMIT.QUEUE. |
| Payload | byte[] | Bezpośrednia kopia bajtów z przychodzącej publikacji do brokera publikowania/subskrypcji. Długość może wynosić zero. |

MQTT -bezstanowe i stanowe sesje

Klienty MQTT mogą tworzyć stanową sesję z menedżerem kolejek. Gdy stanowy klient MQTT rozłącza się, menedżer kolejek utrzymuje subskrypcje utworzone przez klienta oraz komunikaty w trakcie przetwarzania. Gdy klient ponownie nawiąże połączenie, rozwiązuje on komunikat w trakcie lotu. Wysyła on wszystkie komunikaty, które znajdują się w kolejce dostarczania, a następnie odbiera wszystkie komunikaty opublikowane dla jego subskrypcji, podczas gdy została rozłączona.

Gdy klient MQTT łączy się z kanałem pomiarowym, uruchamia nową sesję lub wznawia starą sesję. Nowa sesja nie ma zaległych wiadomości, które nie zostały potwierdzone, nie ma subskrypcji i nie ma publikacji oczekujących na dostarczenie. Gdy klient łączy się, określa, czy rozpocząć od sesji czyszczenia, czy też wznowić istniejącą sesję. Patrz sekcja [Czyszczenie sesji](#).

Jeśli klient wznawia istniejącą sesję, kontynuuje działanie tak, jakby połączenie nie zostało zerwane. Publikacje oczekujące na dostarczenie są wysyłane do klienta, a wszelkie transfery komunikatów, które nie zostały zatwierdzone, są zakończone. Gdy klient w sesji trwałej rozłącza się z usługą telemetryczną (MQXR), wszystkie subskrypcje utworzone przez klienta pozostaną. Publikacje dotyczące subskrypcji są wysyłane do klienta po ponownym nawiązaniu połączenia. Jeśli zostanie ponownie nawiązane połączenie bez wznowienia starej sesji, publikacje zostaną odrzucone przez usługę telemetryczną (MQXR).

Informacje o stanie sesji są zapisywane przez menedżer kolejek w kolejce SYSTEM.MQTT.PERSISTENT.STATE.

Administrator produktu IBM MQ może odłączyć i wyczyścić sesję.

Gdy klient MQTT nie jest połączony

Jeśli klient nie jest połączony z menedżerem kolejek, może nadal odbierać publikacje w jego imieniu. Są one przekazywane do klienta po ponownym nawiązaniu połączenia. Klient może utworzyć "Ostatni testament i testament", który menedżer kolejek publikuje w imieniu klienta, jeśli klient zostanie nieoczekiwanie rozłączony.

Jeśli użytkownik chce być powiadamiany, gdy klient nieoczekiwanie rozłącza się, może zarejestrować ostatnią publikację testamentową i testament; patrz [Publikacja testamentu ostatniego testamentu i testamentu](#). Jest on wysyłany przez usługę telemetryczną (MQXR), jeśli wykryje, że połączenie z klientem zostało zerwane bez żądania przez klienta.

Klient może opublikować zachowaną publikację w dowolnym momencie. Patrz sekcja [Zachowane publikacje i klienty MQTT](#). Nowa subskrypcja tematu może zażądać wysłania wszystkich zachowanych publikacji powiązanych z tematem. Jeśli zostanie utworzona ostatnia wola i testament jako zachowana publikacja, można użyć jej do monitorowania statusu klienta.

Na przykład klient publikuje zachowaną publikację, gdy połączy się z nią, a następnie reklamuje jej dostępność. Jednocześnie tworzy zachowaną ostatnią publikację testamentową i testamentową, która zapowiada niedostępność. Dodatkowo, tuż przed tym, jak zaplanowano rozłączenie, publikuje swoją niedostępność jako zachowaną publikację. Aby dowiedzieć się, czy klient jest dostępny, należy zapisać się na temat zachowanej publikacji. Zawsze otrzymywałabyś jedną z trzech publikacji.

Jeśli klient ma odbierać komunikaty publikowane po odłączeniu, należy ponownie podłączyć klienta do poprzedniej sesji; patrz "MQTT -bezstanowe i stanowe sesje" na stronie 139. Jego subskrypcje są aktywne do momentu ich usunięcia lub do czasu utworzenia przez klienta sesji czyszczenia.

Luźne sprzężenie między klientami MQTT a aplikacjami produktu IBM MQ

Przepływ publikacji między klientami MQTT i aplikacjami produktu IBM MQ jest luźno powiązany. Publikacje mogą pochodzić zarówno z klienta MQTT, jak i z aplikacji IBM MQ, a w żadnym przypadku nie są dostępne żadne zamówienie. Publikatorzy i subskrybenci są luźno sprzężone. Współdziałają one ze sobą pośrednio poprzez publikacje i subskrypcje. Komunikaty można również wysyłać bezpośrednio do klienta MQTT z aplikacji IBM MQ.

Klienty MQTT i aplikacje produktu IBM MQ są luźno powiązane w dwóch sensach:

1. Publikatory i subskrybenci są luźno sprzężone przez powiązanie publikacji i subskrypcji z tematem. Wydawcy i subskrybenci nie mają zazwyczaj informacji o adresie lub tożsamości innego źródła publikacji lub subskrypcji.
2. Klienci produktu MQTT publikują, subskrybują, odbierają publikacje i przetwarzają potwierdzenia dostarczania w oddzielnych wątkach.

Aplikacja kliencka MQTT nie czeka aż do momentu dostarczenia publikacji. Aplikacja przekazuje komunikat do klienta MQTT, a następnie aplikacja kontynuuje działanie we własnym wątku. Element dostarczania jest używany do synchronizowania aplikacji z publikacją publikacji. Patrz sekcja [Znaczniki dostarczania](#).

Po przesłaniu komunikatu do klienta MQTT aplikacja ma wybór oczekiwania na token dostarczania. Klient zamiast czekać może udostępnić metodę wywołania zwrotnego, która jest wywoływana, gdy publikacja jest dostarczana do produktu IBM MQ. Może on również zignorować znacznik dostarczania.

W zależności od jakości usługi powiązanej z komunikatem znacznik dostarczania jest zwracany natychmiast do metody wywołania zwrotnego lub prawdopodobnie po pewnym czasie. Znacznik dostarczania może zostać zwrócony po odłączeniu i ponownym nawiązaniu połączenia przez klienta. Jeśli jakość usługi to *ogień i zapomnij*, element dostarczania jest zwracany natychmiast. W pozostałych dwóch przypadkach znacznik dostarczania jest zwracany tylko wtedy, gdy klient otrzymuje potwierdzenie, że publikacja została wysłana do subskrybentów.

Publikacje wysłane do klienta MQTT w wyniku subskrypcji klienta są dostarczane do metody wywołania zwrotnego `messageArrived`. Program `messageArrived` działa w innym wątku do aplikacji głównej.

Wysyłanie komunikatów bezpośrednio do klienta MQTT

Komunikat można wysłać do konkretnego klienta MQTT na jeden z dwóch sposobów.

1. Aplikacja IBM MQ może wysłać komunikat bezpośrednio do klienta MQTT bez subskrypcji. Patrz sekcja [Wysyłanie komunikatu bezpośrednio do klienta](#).
2. Alternatywnym podejściem jest użycie konwencji nazewnictwa `ClientIdentifier`. Wszyscy subskrybenci programu MQTT tworzą subskrypcje przy użyciu ich unikalnego `ClientIdentifier` jako tematu. Opublikuj do katalogu `ClientIdentifier`. Publikacja jest wysyłana do klienta, który zasubskrybował dany temat `ClientIdentifier`. Za pomocą tej techniki można wysłać publikację do konkretnego subskrybenta produktu MQTT.

Windows

Linux

AIX

Zabezpieczenia produktu MQ Telemetry

Zabezpieczanie urządzeń telemetrycznych może być ważne, ponieważ urządzenia mogą być przenośne, a stosowane w miejscach, których nie można dokładnie kontrolować. Za pomocą sieci VPN można zabezpieczyć połączenie z urządzenia MQTT do usługi telemetrycznej (MQXR). Produkt MQ Telemetry udostępnia dwa inne mechanizmy zabezpieczeń: TLS i JAAS.

Protokół TLS jest używany głównie do szyfrowania komunikacji między urządzeniem i kanałem pomiarowym, a także do uwierzytelniania urządzenia, które łączy się z poprawnym serwerem. Patrz sekcja [Uwierzytelnianie kanału telemetrycznego za pomocą protokołu TLS](#). Za pomocą protokołu TLS można również sprawdzić, czy urządzenie klienckie może łączyć się z serwerem. Patrz sekcja [Uwierzytelnianie klienta MQTT przy użyciu protokołu TLS](#).

JAAS jest głównie używany do sprawdzania, czy użytkownik urządzenia jest uprawniony do używania aplikacji serwera. Patrz sekcja [Uwierzytelnianie klienta MQTT przy użyciu hasła](#). Usługa JAAS może być używana z serwerem LDAP do sprawdzania hasła przy użyciu katalogu pojedynczego logowania.

Protokoły TLS i JAAS mogą być używane w połączeniu w celu zapewnienia dwóch czynników uwierzytelniania. Istnieje możliwość ograniczenia szyfrów używanych przez protokół TLS do szyfrów spełniających standardy FIPS.

Przy co najmniej kilkudziesięciu tysiącach użytkowników nie zawsze jest to praktyczne zapewnienie indywidualnych profili bezpieczeństwa. Nie zawsze jest też praktyczne użycie profili w celu autoryzowania poszczególnych użytkowników do uzyskiwania dostępu do obiektów IBM MQ. Zamiast tego należy

grupować użytkowników w klasy, aby autoryzować publikację i subskrypcję tematów, a także wysyłać publikacje do klientów.

Skonfiguruj każdy kanał telemetryczny, aby odwzorować klienty na wspólne identyfikatory użytkowników klientów. Dla każdego klienta, który łączy się z konkretnym kanałem, należy użyć wspólnego identyfikatora użytkownika. Patrz sekcja [Tożsamość klienta i autoryzacja klientaMQTT](#).

Autoryzowanie grup użytkowników nie stanowi kompromisu dla uwierzytelniania poszczególnych użytkowników. Każdy użytkownik może być uwierzytelniany, na kliencie lub na serwerze, z ich nazwą użytkownika i hasłem, a następnie autoryzowany na serwerze przy użyciu wspólnego identyfikatora użytkownika.

Windows

Linux

AIX

MQ Telemetry globalizacja

Ładunek komunikatu w protokole MQTT v3 jest zakodowany jako tablica bajtów. Ogólnie, aplikacja obsługując tekst tworzy ładunek komunikatu w produkcie UTF-8. Kanał pomiarowy opisuje ładunek komunikatu jako UTF-8, ale nie powoduje konwersji stron kodowych. Łańcuch tematu publikacji musi mieć wartość UTF-8.

Aplikacja jest odpowiedzialna za przekształcanie danych alfabetycznych na poprawną stronę kodową i dane numeryczne na poprawne kodowanie liczb.

Klient MQTT Java ma wygodną metodę `MqttMessage.toString`. Metoda traktuje ładunek komunikatu jako zakodowany w domyślnym zestawie znaków platformy lokalnej, który jest ogólnie UTF-8. Przekształca on ładunek w łańcuch Java. Java ma metodę `String.getBytes`, która przekształca łańcuch w tablicę bajtów zakodowaną przy użyciu domyślnego zestawu znaków platformy lokalnej. Dwa programy MQTT Java wymieniające tekst w ładunku komunikatu, między platformami o tym samym domyślnym zestawie znaków, są tak łatwe i wydajne w produkcie UTF-8.

Jeśli domyślnym zestawem znaków jednej z platform jest inny niż UTF-8, aplikacje muszą ustanowić konwencję wymiany komunikatów. Na przykład: publikator określa konwersję z łańcucha na UTF-8 przy użyciu metody `getBytes("UTF8")`. W celu odebrania tekstu komunikatu, subskrybent przyjmuje, że komunikat jest zakodowany w zestawie znaków UTF-8.

Usługa telemetryczna (MQXR) opisuje kodowanie wszystkich przychodzących publikacji z komunikatów klientów produktu MQTT jako produktu UTF-8. Ustawia wartość `MQMD.CodedCharSetId` do UTF-8i `RFH2.CodedCharSetId` do `MQCCSI_INHERIT`; patrz ["Integracja produktu MQ Telemetry z menedżerami kolejek"](#) na stronie 136. Format publikacji jest ustawiony na wartość `MQFMT_NONE`, dlatego nie może być wykonywana konwersja przez kanały ani przez program MQGET.

Windows

Linux

AIX

Wydajność i skalowalność produktu MQ

Telemetry

Podczas zarządzania dużą liczbą klientów i zwiększania skalowalności produktu MQ Telemetry należy wziąć pod uwagę następujące czynniki.

Planowania mocy obliczeniowej

Więcej informacji na temat raportów dotyczących wydajności dla produktu MQ Telemetry zawiera sekcja [MQ Performance documents](#).

Połączenia

Koszty związane z połączeniami obejmują

- Koszt utworzenia połączenia pod względem wykorzystania procesora i czasu.
- Koszty sieci.
- Pamięć używana podczas utrzymywania otwartego połączenia, ale bez jego używania.

Istnieje dodatkowe obciążenie związane z pozostaniem połączenia klientów. Jeśli połączenie jest otwarte, przepływy TCP/IP i komunikaty produktu MQTT korzystają z sieci w celu sprawdzenia, czy połączenie

nadal istnieje. Dodatkowo, pamięć jest używana na serwerze dla każdego połączenia klienckiego, które jest otwarte.

W przypadku wysyłania wiadomości więcej niż 1 na minutę, należy pozostawić otwarte połączenie, aby uniknąć kosztów inicjowania nowego połączenia. Jeśli wiadomości są wysyłane mniej niż co 10-15 minut, należy rozważyć usunięcie połączenia, aby uniknąć kosztów utrzymania go w otwartym systemie. Połączenie TLS może być otwarte, ale bezczynne, przez dłuższy czas, ponieważ jest bardziej kosztowne do skonfigurowania.

Ponadto należy rozważyć możliwość obsługi klienta. Jeśli na kliencie znajduje się funkcja przechowywania i przekazywania, wówczas można utworzyć wsadowe komunikaty i usunąć połączenie między wysyłaniem zadań wsadowych. Jeśli jednak klient zostanie odłączony, nie będzie możliwe odebranie przez klienta komunikatu z serwera. W związku z tym cel Twojej aplikacji ma wpływ na decyzję.

Jeśli w systemie jest jeden klient wysyłający wiele komunikatów, na przykład przesyłanie plików, nie należy czekać na odpowiedź serwera na komunikat. Zamiast tego należy wysłać wszystkie wiadomości i sprawdzić, czy wszystkie zostały odebrane. Alternatywnie można użyć funkcji [Quality of Service](#) (QoS).

QoS można zmieniać, dostarczając nieistotne komunikaty przy użyciu QoS 0 i ważnych komunikatów przy użyciu QoS 2. Przepustowość komunikatów może być dwukrotnie wyższa od wartości QoS 0 niż w przypadku QoS 2.

Konwencje nazewnictwa

Jeśli projektujesz aplikację dla wielu klientów, zaimplementuj skuteczną konwencję nazewnictwa. Aby odwzorować każdego klienta na poprawny identyfikator `ClientIdentifier`, należy nadać wartość `ClientIdentifier`, która ma znaczenie. Dobra konwencja nazewnictwa ułatwia administratorowi pracę, które klienty są uruchomione. Konwencja nazewnictwa pomaga administratorowi filtrować długą listę klientów w programie IBM MQ Explorer i pomaga w określeniu problemu. Patrz sekcja [Identyfikator klienta](#).

Przepustowość

Długość nazw tematów ma wpływ na liczbę bajtów, które przepłyną przez sieć. W przypadku publikowania lub subskrybowania liczba bajtów w komunikacie może być istotna. W związku z tym należy ograniczyć liczbę znaków w nazwie tematu. Gdy klient MQTT subskrybuje temat IBM MQ, nadaje mu nazwę formularza:

```
ClientIdentifier: TopicName
```

Aby wyświetlić wszystkie subskrypcje dla klienta MQTT, można użyć komendy IBM MQ MQSC **DISPLAY**:

```
DISPLAY SUB(' ClientID1:*')
```

Definiowanie zasobów w produkcie IBM MQ do użytku przez klienty MQTT

Klient MQTT łączy się ze zdalnym menedżerem kolejek produktu IBM MQ. Istnieją dwie podstawowe metody przesyłania komunikatów do klienta MQTT za pomocą aplikacji IBM MQ: ustawienie domyślnej kolejki transmisji na `SYSTEM.MQTT.TRANSMIT.QUEUE` lub użycie aliasów menedżera kolejek. Zdefiniuj domyślną kolejkę transmisji dla menedżera kolejek, jeśli liczba klientów MQTT jest duża. Użycie domyślnego ustawienia kolejki transmisji upraszcza nakład pracy administracji. Patrz sekcja [Konfigurowanie rozproszonego kolejkowania w celu wysyłania komunikatów do klientów MQTT](#).

Poprawa skalowalności poprzez unikanie subskrypcji.

Gdy klient MQTT V3 subskrybuje temat, subskrypcja jest tworzona przez usługę telemetryczną (MQXR) w produkcie IBM MQ. Subskrypcja kieruje publikacjami dla klienta do produktu `SYSTEM.MQTT.TRANSMIT.QUEUE`. Nazwa zdalnego menedżera kolejek w nagłówku przesyłania każdej

publikacji jest ustawiana na wartość `ClientIdentifier` klienta MQTT, który dokonał subskrypcji. Jeśli istnieje wiele klientów, każda z nich ma własne subskrypcje, co powoduje, że w całym klastrze lub hierarchii IBM MQ publikowania/subskrypcji utrzymywana jest wiele subskrypcji proxy. Informacje o niekorzystaniu z publikowania/subskrypcji, ale z użyciem rozwiązania opartego na punktach zamiast tego można znaleźć w sekcji [Wysyłanie komunikatu bezpośrednio do klienta](#).

Zarządzanie dużą liczbą klientów

Aby obsłużyć wiele jednocześnie połączonych klientów, zwiększ ilość pamięci dostępnej dla usługi telemetrycznej (MQXR), ustawiając parametry maszyny JVM **-Xms** i **-Xmx**. Wykonaj następujące kroki:

1. Znajdź plik `java.properties` w katalogu konfiguracji usługi telemetrycznej. Patrz sekcja [Katalog konfiguracyjny usługi Telemetry \(MQXR\) w produkcie Windows](#) lub [Katalog konfiguracyjny usługi telemetrycznej w produkcie Linux](#).
2. Należy postępować zgodnie ze wskazówkami zawartymi w pliku; sarta o wielkości 1 GB jest wystarczająca dla 50 000 jednocześnie połączonych klientów.

```
# Heap sizing options - uncomment the following lines to set the heap to 1G
#-Xmx1024m
#-Xms1024m
```

3. Dodaj inne argumenty wiersza komend, aby przejść do maszyny JVM, na której działa usługa telemetryczna (MQXR) w pliku `java.properties`. Patrz sekcja [Przekazywanie parametrów maszyny JVM do usługi telemetrycznej \(MQXR\)](#).

Aby zwiększyć liczbę otwartych deskryptorów plików w systemie Linux, należy dodać następujące wiersze do programu `/etc/security/limits.conf`/i zalogować się ponownie.

```
@mqm soft nofile 65000
@mqm hard nofile 65000
```

Każde gniazdo wymaga jednego deskryptora pliku. Usługa telemetryczna wymaga dodatkowych deskryptorów plików, dlatego liczba ta musi być większa od wymaganej liczby otwartych gniazd.

Menedżer kolejek używa uchwytu obiektu dla każdej nietrwałej subskrypcji. Aby można było obsługiwać wiele aktywnych, nietrwałych subskrypcji, należy zwiększyć maksymalną liczbę aktywnych uchwytów w menedżerze kolejek, na przykład:

```
echo ALTER QMGR MAXHANDS(99999999) | runmqsc qMgrName
```

Rysunek 49. Zmień maksymalną liczbę uchwytów w systemie Windows

```
echo "ALTER QMGR MAXHANDS(99999999)" | runmqsc qMgrName
```

Rysunek 50. Zmień maksymalną liczbę uchwytów w systemie Linux

Inne zagadnienia

Podczas planowania wymagań systemowych należy wziąć pod uwagę czas, jaki należy podjąć, aby zrestartować system. Planowany czas przestoju może mieć wpływ na liczbę komunikatów, które kolejka jest w kolejce, oczekując na przetworzenie. Skonfiguruj system w taki sposób, aby komunikaty mogły zostać pomyślnie przetworzone w akceptowalnym czasie. Przejrzyj pamięć dyskową, pamięć i moc przetwarzania. W przypadku niektórych aplikacji klienckich komunikaty mogą być odrzucane po ponownym nawiązaniu połączenia przez klienta. Aby usunąć komunikaty, ustaw parametr `CleanSession` w parametrach połączenia klienta. Patrz sekcja [Czyszczenie sesji](#). Alternatywnie można opublikować i zasubskrybować przy użyciu najlepszej jakości usługi 0w kliencie MQTT. Patrz sekcja [Jakość usługi](#). Użyj komunikatów nietrwałych podczas wysyłania komunikatów z produktu IBM MQ. Komunikaty z tymi cechami usługi nie są odtwarzane w przypadku restartu systemu lub połączenia.

Telemetry

Klienty MQTT mogą być uruchamiane na różnych urządzeniach, od czujników i siłowników, do ręcznych urządzeń i systemów pojazdów.

Klienty MQTT są małe i działają na urządzeniach, które są ograniczone przez niewielką ilość pamięci i małą moc przetwarzania. MQTT protocol jest niezawodny i ma małe nagłówki, które odpowiadają sieciom ograniczonym przez niską przepustowość, wysokie koszty i sporadyczne dostępność.

Produkt MQ Telemetry komunikuje się z urządzeniami telemetryczne za pośrednictwem aplikacji klienckich MQTT. Aplikacje te korzystają z następujących zasobów, z których wszystkie implementują protokół MQTT v3 :

- Następujące biblioteki klienta:
 - *MQTT client for Java*, który jest używany do budowania rodzimych aplikacji dla (na przykład) urządzeń z systemem Android, OS X, Linux lub Windows. Aplikacje, które korzystają z tej biblioteki klienta, mogą działać we wszystkich wariantach produktu Java z najmniejszego CLDC (Connected Limited Device Configuration) /MIDP (Mobile Information Device Profile-Mobile Information Device Profile) za pośrednictwem CDC (Connected Device Configuration) /Foundation, J2SE (Java Platform, Standard Edition) i J2EE (Java Platform, Enterprise Edition). Obsługiwana jest również niestandardowa biblioteka klas IBM jclRM. Platforma Java ME jest zwykle używana na małych urządzeniach, takich jak siłowniki, czujniki, telefony komórkowe i inne wbudowane urządzenia. Platforma Java SE jest zwykle instalowana na wyższych urządzeniach wbudowanych, takich jak komputery desktop i serwery.
 - *MQTT client for C*, który jest używany do budowania rodzimych aplikacji dla (na przykład) urządzeń z systemem iOS, OS X, Linux lub Windows. Ta biblioteka klienta udostępnia implementację referencyjną w języku C wraz z wstępnie zbudowanym klientem rodzimym dla systemów Windows i Linux. Implementacja referencyjna języka C umożliwia wyeksportowanie MQTT do szerokiej gamy urządzeń i platform. Niektóre systemy Windows w systemie Intel, w tym systemy Windows 7, RedHat, Ubuntu i niektóre systemy Linux na platformach ARM, takie jak Eurotech Viper, implementują wersje produktu Linux, które uruchamiają klienta C, ale produkt IBM nie obsługuje obsługi platform. Jeśli planowane jest wywołanie centrum wsparcia produktu IBM, należy odtworzyć problemy z klientem na obsługiwanej platformie.
 - *MQTT client for Java*, który jest używany do budowania aplikacji WWW opartych na przeglądarce.

Biblioteki klienta produktu MQTT są swobodnie dostępne w środowisku Eclipse Paho oraz pod adresem MQTT.org. Patrz [IBM MQ Telemetry Transport przykładowe programy](#).

Zabezpieczenia w produkcie IBM MQ

W produkcie IBM MQ istnieje kilka metod zapewniania bezpieczeństwa: interfejs usługi autoryzacji, interfejs użytkownika lub osoba trzecia, wyjścia kanału, zabezpieczenia kanału przy użyciu protokołu TLS (Transport Layer Security), rekordy uwierzytelniania kanału oraz zabezpieczenia komunikatów.

Interfejs usługi autoryzacji

Autoryzacja za pomocą wywołań MQI, komend i dostępu do obiektów jest udostępniana przez **menedżera uprawnień do obiektów** (OAM), który domyślnie jest włączony. Dostęp do obiektów IBM MQ jest kontrolowany przez grupy użytkowników IBM MQ i OAM. Administratorzy mogą używać interfejsu wiersza komend do nadawania lub odbierania autoryzacji zgodnie z wymaganiami.

Więcej informacji na temat tworzenia komponentów usług autoryzacji znajduje się w sekcji [Konfigurowanie zabezpieczeń w systemach AIX, Linux, and Windows](#).

Wyjścia dla kanału napisanego przez użytkownika lub osoby trzeciej

Kanały mogą korzystać z wyjść użytkownika lub wyjść z kanału innej firmy. Więcej informacji na ten temat zawiera sekcja [Programy obsługi wyjścia kanału dla kanałów przesyłania komunikatów](#).

Zabezpieczenia kanału przy użyciu protokołu TLS

Protokół TLS (Transport Layer Security) zapewnia bezpieczeństwo kanałowe w standardzie branżowym, z ochroną przed podsłuchami, manipulowaniem i personifikacją.

Protokół TLS korzysta z klucza publicznego i technik symetrycznych w celu zapewnienia poufności i integralności komunikatów oraz wzajemnego uwierzytelniania.

Obszerny przegląd zabezpieczeń w produkcie IBM MQ, w tym szczegółowe informacje na temat protokołu TLS, zawiera sekcja [Zabezpieczanie](#). Przegląd protokołu TLS, w tym wskaźniki do komend opisanych w tej sekcji, zawiera sekcja [Cryptographic Security protocols: TLS](#) (Protokoły bezpieczeństwa szyfrowania: TLS).

Rekordy uwierzytelniania kanału

Rekordy uwierzytelniania kanału umożliwiają wykonywanie precyzyjnej kontroli nad dostępem przyznanego do łączenia systemów na poziomie kanału. Więcej informacji na ten temat zawiera sekcja [Rekordy uwierzytelniania kanału](#).

Bezpieczeństwo komunikatu

Produkt Advanced Message Security, który jest oddzielnie instalowanym i licencjonowanym komponentem produktu IBM MQ, umożliwia ochronę kryptografii dla komunikatów wysyłanych i odbieranych przy użyciu produktu IBM MQ. Patrz [Advanced Message Security](#).

Zadania pokrewne

[Zabezpieczanie](#)

[Planowanie wymagań dotyczących bezpieczeństwa](#)

Obsługa protokołu TLS klienta zarządzanego przez program IBM MQ.NET

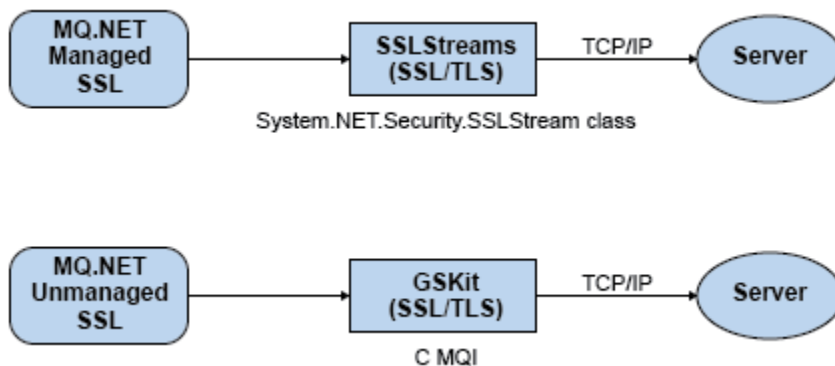
W pełni zarządzany klient IBM MQ.NET zapewnia obsługę protokołu TLS (Transport Layer Security), która jest oparta na pakiecie Microsoft.NET SSLStreams. Różni się to od innych klientów produktu IBM MQ, które są oparte na pakiecie GSKit.

Aplikacje produktu IBM MQ.NET można tworzyć w taki sposób, aby były uruchamiane w trybie zarządzanym lub w trybie niezarządzanym.

- W trybie zarządzanym aplikacje produktu .NET działają w środowisku CLR .NET (Common Language Runtime) bez wywoływania wielu platform, takich jak wywoływanie interfejsu C MQI.
- W trybie niezarządzanym interfejs C MQI jest wywoływany dla bazowych operacji MQI. Interfejs trybu niezarządzanego składa się z klas opakowujących produktu .NET na początku interfejsu C MQI.

Zarządzany klient IBM MQ.NET korzysta z bibliotek środowiska Microsoft.NET Framework w celu zaimplementowania protokołów bezpiecznych gniazd TLS. System.NET.Security.SSLStream z produktu Microsoft jest używana do implementowania zabezpieczeń (TLS) w produkcie IBM MQ.NET.

Niezarządzany tryb klienta IBM MQ.NET obsługuje już funkcję TLS, która jest oparta na języku C MQI (i GSKit). Oznacza to, że operacje TLS są obsługiwane przez interfejs C MQI. W tym przypadku pakiet GSKit implementuje protokoły bezpiecznego gniazda TLS.



Rysunek 51. Porównanie zarządzanych i niez zarządzanych TLS w programie IBM MQ.NET

W poniższej tabeli podsumowano różnice między implementacjami zarządzanymi i niez zarządzanymi:

Tabela 14. Różnice między implementacjami zarządzanymi i niez zarządzanymi

| Tryb | Protocols (Protokoły) | Implementacja | Komentarze |
|---------------------------|-----------------------|---|---|
| Zarządzany SSL IBM MQ.NET | TLS | System.NET.Security.SSLStream Klasa SSLStream działa jako strumień przez połączone gniazdo TCP | TLS 1.0 TLS 1.2 (tylko w środowisku Microsoft.NET Framework v4.5) |
| IBM MQ.NET unmanaged SSL | TLS | GSKIT i C-MQI | Protokoły bezpiecznego gniazda TLS |

Pojęcia pokrewne

Obsługa protokołu SSL (Secure Sockets Layer) i TLS (Transport Layer Security) dla produktu .NET

Klienci i serwery

Wprowadzenie do sposobu, w jaki produkt IBM MQ obsługuje konfiguracje typu klient-serwer dla aplikacji.

Klient MQI produktu IBM MQ to komponent, który umożliwia uruchomienie aplikacji w systemie w celu wywołania wywołań MQI w menedżerze kolejek uruchomionym w innym systemie. Dane wyjściowe z wywołań są wysyłane z powrotem do klienta, który przekazuje je z powrotem do aplikacji.

IBM MQ *serwer* to menedżer kolejek, który udostępnia usługi kolejkowania dla jednego lub wielu klientów. Wszystkie obiekty IBM MQ (na przykład kolejki) istnieją tylko na komputerze z menedżerem kolejek (komputer z serwerem IBM MQ), a nie na kliencie. Serwer IBM MQ może również obsługiwać lokalne aplikacje produktu IBM MQ.

Różnica między serwerem IBM MQ a zwykłym menedżerem kolejek polega na tym, że serwer ma dedykowane łącze komunikacyjne z każdym klientem. Więcej informacji na temat tworzenia kanałów dla klientów i serwerów znajduje się w sekcji [Konfigurowanie kolejkowania rozproszonego](#).

Informacje na temat klientów ogólnie zawiera sekcja [“Przegląd produktu IBM MQ MQI clients”](#) na stronie 147.

Aplikacje produktu IBM MQ w środowisku klient-serwer

W przypadku połączenia z serwerem aplikacje klienckie IBM MQ mogą wydawać większość wywołań MQI w taki sam sposób, jak aplikacje lokalne. Aplikacja kliencka wywołuje wywołanie MQCONN w celu nawiązania połączenia z określonym menedżerem kolejek. Wszystkie dodatkowe wywołania MQI, które określają uchwyt połączenia zwrócony z żądania połączenia, są następnie przetwarzane przez tego menedżera kolejek.

Należy połączyć aplikacje z odpowiednimi bibliotekami klienta. Patrz sekcja [Budowanie aplikacji dla produktu IBM MQ MQI clients](#).

Pojęcia pokrewne

“Zarządzanie transakcjami i obsługa” na stronie 153

Wprowadzenie do zarządzania transakcjami i sposobu obsługi transakcji przez produkt IBM MQ .

“Rozszerzanie obiektów menedżera kolejek” na stronie 155

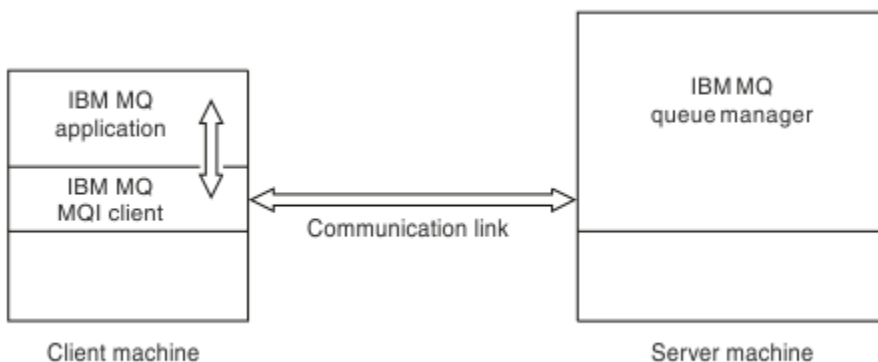
Obiekty menedżera kolejek można rozszerzać, korzystając z wyjść użytkownika, wyjść funkcji API lub usług instalowalnych.

Przegląd produktu IBM MQ MQI clients

IBM MQ MQI client jest komponentem produktu IBM MQ , który może być zainstalowany w systemie, w którym nie jest uruchomiony żaden menedżer kolejek.

Za pomocą IBM MQ MQI client aplikacja działająca w tym samym systemie, w której klient może nawiązać połączenie z menedżerem kolejek uruchomionym w innym systemie. Aplikacja może wydawać wywołania MQI do tego menedżera kolejek. Taka aplikacja jest nazywana aplikacją IBM MQ MQI client , a menedżer kolejek jest nazywany *menedżerem kolejek serwera*.

Aplikacja IBM MQ MQI client i menedżer kolejek serwera komunikują się ze sobą za pomocą *kanatu MQI*. Kanał MQI jest uruchamiany wtedy, gdy aplikacja kliencka wysyła wywołanie **MQCONN** lub **MQCONNX** w celu nawiązania połączenia z menedżerem kolejek i kończy się, gdy aplikacja kliencka zgłasza wywołanie **MQDISC** w celu rozłączenia się z menedżerem kolejek. Parametry wejściowe przepływu wywołania MQI w jednym kierunku na kanale MQI i w przepływie parametrów wyjściowych są w przeciwnym kierunku.



Rysunek 52. Połączenie między klientem a serwerem

Można użyć następujących platform. Kombinacje te zależą od używanego produktu IBM MQ i są opisane w sekcji [“Obsługa platform dla klientów IBM MQ”](#) na stronie 150.

IBM MQ MQI client

AIX and Linux
Windows
IBM i

Serwer IBM MQ

AIX and Linux
Windows
IBM i
z/OS

Interfejs MQI jest dostępny dla aplikacji działających na platformie klienckiej. Kolejki i inne obiekty produktu IBM MQ są przechowywane w menedżerze kolejek, który został zainstalowany na serwerze.

Aplikacja, która ma zostać uruchomiona w środowisku IBM MQ MQI client , musi być najpierw powiązana z odpowiednią biblioteką klienta. Gdy aplikacja wysyła wywołanie MQI, IBM MQ MQI client kieruje

żądanie do menedżera kolejek, gdzie jest przetwarzany, i z miejsca, w którym odpowiedź jest wysyłana z powrotem do IBM MQ MQI client.

Połączenie między aplikacją a IBM MQ MQI client jest ustanawiane dynamicznie w czasie wykonywania.

Aplikacje klienckie można również tworzyć przy użyciu produktów IBM MQ classes for .NET, IBM MQ classes for Java lub IBM MQ classes for Java Message Service (JMS). Klientów Java i JMS można używać na następujących platformach:

-  IBM i
-  AIX
-  Linux
-  Windows

W tym miejscu nie opisano używania produktów Java i JMS. Szczegółowe informacje na temat instalowania, konfigurowania i używania produktów IBM MQ classes for Java i IBM MQ classes for JMS można znaleźć w sekcji [Korzystanie z produktów IBM MQ classes for Java](#) i [Korzystanie z produktu IBM MQ classes for JMS](#).

Pojęcia pokrewne

[“Dlaczego należy używać klientów IBM MQ ?” na stronie 148](#)

Korzystanie z klientów IBM MQ jest wydajnym sposobem implementowania przesyłania komunikatów w produkcie IBM MQ i kolejowania.

[“Jak skonfigurować IBM MQ MQI client” na stronie 150](#)

Wykonaj poniższe instrukcje, aby skonfigurować klienta.

[“Co to jest rozszerzony klient transakcyjny?” na stronie 151](#)

Rozszerzony klient transakcyjny IBM MQ może aktualizować zasoby zarządzane przez innego menedżera zasobów, pod kontrolą zewnętrznego menedżera transakcji.

[“Sposób łączenia się klienta z serwerem” na stronie 152](#)

Klient łączy się z serwerem za pomocą MQCONN lub MQCONNX, a następnie komunikuje się za pośrednictwem kanału.

Dlaczego należy używać klientów IBM MQ ?

Korzystanie z klientów IBM MQ jest wydajnym sposobem implementowania przesyłania komunikatów w produkcie IBM MQ i kolejowania.

Istnieje możliwość zastosowania aplikacji, która używa interfejsu MQI uruchomionego na jednym komputerze, oraz menedżera kolejek działającego na innym komputerze (fizycznym lub wirtualnym). Korzyści z tego związane są następujące:

- Na komputerze klienckim nie ma potrzeby wykonywania pełnej implementacji produktu IBM MQ .
- Wymagania sprzętowe w systemie klienckim są zmniejszone.
- Wymagania dotyczące administrowania systemem są zmniejszone.
- Aplikacja IBM MQ działająca na kliencie może łączyć się z wieloma menedżerami kolejek w różnych systemach.
- Można wykorzystać alternatywne kanały przy użyciu różnych protokołów transmisji.

Odsyłacze pokrewne

[“Jakie aplikacje działają na serwerze IBM MQ MQI client?” na stronie 149](#)

Pełny interfejs MQI jest obsługiwany w środowisku klienta.

[“Obsługa platform dla klientów IBM MQ” na stronie 150](#)

Produkt IBM MQ na wszystkich obsługiwanych platformach serwerów akceptuje połączenia klientów z produktu IBM MQ MQI clients na wielu platformach.

Jakie aplikacje działają na serwerze IBM MQ MQI client?

Pełny interfejs MQI jest obsługiwany w środowisku klienta.

Umożliwia to skonfigurowanie niemal dowolnej aplikacji produktu IBM MQ w taki sposób, aby była uruchamiana w systemie IBM MQ MQI client przez połączenie aplikacji z IBM MQ MQI client z biblioteką MQIC, a nie z biblioteką MQI. Wyjątkami są:

- MQGET z sygnałem
- Aplikacja, która wymaga koordynacji punktów synchronizacji z innymi menedżerami zasobów, musi korzystać z rozszerzonego klienta transakcyjnego.

Jeśli funkcja odczytu z wyprzedzeniem jest włączona, aby zwiększyć wydajność przesyłania komunikatów nietrwałych, nie wszystkie opcje MQGET są dostępne. Tabela zawiera opcje, które są dozwolone oraz informacje o tym, czy można je zmieniać między wywołaniami MQGET.

Tabela 15. Opcje MQGET są dozwolone, gdy opcja odczytu z wyprzedzeniem jest włączona

| Wartości | Dozwolone, gdy funkcja odczytu z wyprzedzeniem jest włączona i może być zmieniana między wywołaniami MQGET | Dozwolone, gdy funkcja odczytu z wyprzedzeniem jest włączona, ale nie może być zmieniana między wywołaniami MQGET ¹ | Opcje MQGET, które nie są dozwolone, gdy funkcja odczytu z wyprzedzeniem jest włączona (²) |
|----------------------|---|---|--|
| Wartości MQGET MD | MsgId ³ CorrelId ³ | Kodowanie CodedCharSetId | |
| Opcje MQGMO MQGET | MQGMO_WAIT MQGMO_NO_WAIT MQGMO_FAIL_IF QUIESCING MQGMO_BROWSE_FIRST ⁴ MQGMO_BROWSE_NEXT ⁴ MQGMO_BROWSE_MESSAGE _UNDER_CURSOR ⁴ | MQGMO_SYNCPOINT_IF_PERSISTENT MQGMO_NO_SYNCPOINT Komunikat MQGMO_ACCEPT_TRUNCATED_MSG MQGMO_CONVERT MQGMO_LOGICAL_ORDER MQGMO_COMPLETE_MSG MQGMO_ALL_MSGS_AVAILABLE MQGMO_ALL_SEGMENTS_AVAILABLE MQGMO_MARK_BROWSE_HANDLE MQGMO_MARK_BROWSE_CO_OP MQGMO_UNMARK_BROWSE_CO_OP MQGMO_UNMARK_BROWSE_HANDLE MQGMO_UNMARKED_BROWSE_MSG MQGMO_PROPERTIES_FORCE_MQRFH2 MQGMO_NO_PROPERTIES MQGMO_PROPERTIES_IN_HANDLE MQGMO_PROPERTIES_COMPATIBILITY | MQGMO_SET_SIGNAL MQGMO_SYNCPOINT, MQGMO_MARK_SKIP _BACKOUT MQGMO_MSG_UNDER _CURSOR ⁴ Blokada MQGMO_LOCK MQGMO_UNLOCK |
| Wartości MQGMO | | MsgHandle | |

1. Jeśli te opcje zostaną zmienione między wywołaniami MQGET, zostanie zwrócony kod przyczyny MQRC_OPTIONS_CHANGED.
2. Jeśli te opcje zostaną podane podczas pierwszego wywołania MQGET, odczyt z wyprzedzeniem zostanie wyłączony. Jeśli te opcje zostaną podane w kolejnym wywołaniu MQGET, zostanie zwrócony kod przyczyny MQRC_OPTIONS_ERROR.
3. Aplikacje klienckie muszą uwzględniać fakt, że jeśli wartości MsgId i CorrelId zostały zmienione między wywołaniami MQGET, komunikaty z poprzednimi wartościami mogły już zostać wysłane do klienta i pozostają w buforze odczytu z wyprzedzeniem na kliencie, dopóki nie zostaną wykorzystane (lub automatycznie usunięte).
4. Pierwsze wywołanie MQGET określa, czy komunikaty mają być przeglądane lub pobierane z kolejki, gdy włączony jest odczyt z wyprzedzeniem. Jeśli w aplikacji zostanie podjęta próba użycia zarówno operacji przeglądania, jak i pobierania, zostanie zwrócony kod przyczyny MQRC_OPTIONS_CHANGED.

5. Opcja MQGMO_MSG_UNDER_CURSOR nie jest dostępna, jeśli włączony jest odczyt z wyprzedzeniem. Komunikaty można przeglądać albo odbierać, gdy włączony jest odczyt z wyprzedzeniem. Nie można jednak jednocześnie korzystać z obu tych funkcji.

Aplikacja działająca na serwerze IBM MQ MQI client może współbieżnie łączyć się z więcej niż jednym menedżerem kolejek lub używać nazwy menedżera kolejek z gwiazdką (*) w wywołaniu MQCONN lub MQCONNX (patrz przykłady w sekcji [Łączenie aplikacji IBM MQ MQI client z menedżerami kolejek](#)).

Obsługa platform dla klientów IBM MQ

Produkt IBM MQ na wszystkich obsługiwanych platformach serwerów akceptuje połączenia klientów z produktu IBM MQ MQI clients na wielu platformach.

Produkt IBM MQ zainstalowany jako *produkt podstawowy i serwer* na wszystkich obsługiwanych platformach serwera może akceptować połączenia z poziomu produktu IBM MQ MQI clients na następujących platformach:





-  IBM i
-  AIX
-  Linux
-  Windows

Połączenia klienckie podlegają różnicom w kodowanym identyfikatorze zestawu znaków (CCSID) i protokole komunikacyjnym.

Jak skonfigurować IBM MQ MQI client

Wykonaj poniższe instrukcje, aby skonfigurować klienta.

Aby skonfigurować IBM MQ MQI client, należy mieć już zainstalowany i działający serwer IBM MQ, z którym będzie się łączyć klient. Kroki związane z konfiguracją klienta są następujące:

1. Sprawdź, czy masz odpowiednią platformę dla partycji IBM MQ MQI client oraz czy sprzęt i oprogramowanie spełniają wymagania. Obsługa platformy jest opisana w podręczniku [“Obsługa platform dla klientów IBM MQ”](#) na stronie 150.
2. Zdecyduj, w jaki sposób ma zostać zainstalowany produkt IBM MQ na stacji roboczej klienta, a następnie postępuj zgodnie z instrukcjami dotyczącymi konkretnej kombinacji platform klienta i serwera. Instalacja została opisana w następujących tematach:
 -  [Instalowanie klienta IBM MQ w systemie AIX](#)
 -  [Instalowanie klienta IBM MQ w systemie Linux](#)
 -  [Instalowanie klienta IBM MQ w systemie Windows](#)
 -  [Instalowanie klienta IBM MQ w systemie IBM i](#)
3. Upewnij się, że łącza komunikacyjne są skonfigurowane i połączone. Konfiguracja łączy komunikacyjnych jest opisana w sekcji [Konfigurowanie połączeń między serwerem a klientem](#).
4. Sprawdź, czy instalacja działa poprawnie. Zapoznaj się z sekcją weryfikacji procedury instalacyjnej dla platformy lub platform, które są używane w przedsiębiorstwie.
5. Po zweryfikowaniu instalacji produktu IBM MQ MQI client należy rozważyć, czy należy zabezpieczyć klienta. Zabezpieczenia klienta są opisane w sekcji [Konfigurowanie zabezpieczeń produktu IBM MQ MQI client](#).
6. Skonfiguruj kanały między serwerem IBM MQ MQI client i serwerem, które są wymagane przez aplikacje produktu IBM MQ, które mają być uruchamiane na kliencie. Konfigurowanie kanałów jest opisane w sekcji [Definiowanie kanałów MQI](#). Jeśli używany jest protokół TLS, należy wziąć pod uwagę dodatkowe zagadnienia. Te zagadnienia zostały opisane w sekcji [Określanie, że kanał MQI używa protokołu TLS](#). W celu skonfigurowania kanałów może być konieczne użycie pliku konfiguracyjnego

IBM MQ MQI client lub zmiennych środowiskowych programu IBM MQ . Zmienne środowiskowe produktu IBM MQ są opisane w sekcji [Korzystanie ze zmiennych środowiskowych produktu IBM MQ](#).

7. Aplikacje produktu IBM MQ są w pełni opisane w sekcji [Projektowanie aplikacji](#).

8. Istnieją pewne różnice w środowisku menedżera kolejek, które należy uwzględnić podczas projektowania, budowania i uruchamiania aplikacji w środowisku produktu IBM MQ MQI client . Więcej informacji na temat tych różnic można znaleźć w:

- [Korzystanie z interfejsu kolejki komunikatów \(MQI\) w aplikacji klienckiej](#)
- [Budowanie aplikacji dla produktu IBM MQ MQI clients](#)
- [Łączenie aplikacji IBM MQ MQI client z menedżerami kolejek](#)
- [Rozwiązywanie problemów z produktem IBM MQ MQI clients](#)

Co to jest rozszerzony klient transakcyjny?

Rozszerzony klient transakcyjny IBM MQ może aktualizować zasoby zarządzane przez innego menedżera zasobów, pod kontrolą zewnętrznego menedżera transakcji.

Jeśli użytkownik nie zna pojęć związanych z zarządzaniem transakcjami, należy zapoznać się z [“Zarządzanie transakcjami i obsługa” na stronie 153](#).

Należy zauważyć, że klient transakcyjny XA jest teraz dostarczany jako część produktu IBM MQ.

Aplikacja kliencka może uczestniczyć w jednostce pracy, która jest zarządzana przez menedżer kolejek, z którym jest połączony. W ramach jednostki pracy aplikacja kliencka może umieszczać komunikaty do kolejek należących do tego menedżera kolejek i je z nich pobrać. Aplikacja kliencka może następnie użyć wywołania **MQCMIT** w celu zatwierdzenia jednostki pracy lub wywołania **MQBACK** w celu utworzenia kopii zapasowej jednostki pracy. Jednak w ramach tej samej jednostki pracy aplikacja kliencka nie może zaktualizować zasobów innego menedżera zasobów, na przykład tabel bazy danych Db2 . Użycie rozszerzonego klienta transakcyjnego IBM MQ powoduje usunięcie tego ograniczenia.


Rozszerzony klient transakcyjny IBM MQ to IBM MQ MQI client z pewną dodatkową funkcją. Za pomocą tej funkcji aplikacja kliencka, w ramach tej samej jednostki pracy, może wykonywać następujące zadania:

- Umieszczanie komunikatów w kolejkach i pobieranie komunikatów z kolejek, których właścicielem jest menedżer kolejek, z którym jest połączony
- Aktualizacja zasobów menedżera zasobów innego niż menedżer kolejek produktu IBM MQ

Ta jednostka pracy musi być zarządzana przez zewnętrznego menedżera transakcji, który działa w tym samym systemie, co aplikacja kliencka. Jednostka pracy nie może być zarządzana przez menedżer kolejek, z którym połączona jest aplikacja kliencka. Oznacza to, że menedżer kolejek może działać tylko jako menedżer zasobów, a nie jako menedżer transakcji. Oznacza to również, że aplikacja kliencka może zatwierdzić lub wycofać jednostkę pracy przy użyciu interfejsu API udostępnionego przez zewnętrznego menedżera transakcji. Dlatego aplikacja kliencka nie może korzystać z wywołań MQI, **MQBEGIN**, **MQCMIT** i **MQBACK**.

Zewnętrzny menedżer transakcji komunikuje się z menedżerem kolejek jako menedżer zasobów przy użyciu tego samego kanału MQI, który jest używany przez aplikację kliencką, która jest połączona z menedżerem kolejek. Jednak w sytuacji odtwarzania po awarii, gdy nie są uruchomione żadne aplikacje, menedżer transakcji może użyć dedykowanego kanału MQI w celu odtworzenia niekompletnych jednostek pracy, w których menedżer kolejek uczestniczyli w czasie awarii.

W tej sekcji IBM MQ MQI client , który nie ma rozszerzonej funkcji transakcyjnej, jest określany jako podstawowy klient IBM MQ . W związku z tym można rozważyć, że rozszerzony klient transakcyjny produktu IBM MQ może składać się z klienta podstawowego IBM MQ z dodaniem rozszerzonej funkcji transakcyjnej.

Uwaga:  Produkt IBM MQ MQI client w systemie IBM i nie obsługuje rozszerzonej funkcji transakcyjnej IBM MQ .

Odsyłacze pokrewne

[“Obsługa platform dla rozszerzonych klientów transakcyjnych” na stronie 152](#)

Rozszerzone klienty transakcyjne są dostępne dla wszystkich platform Multiplatforms obsługujących klienta podstawowego. Klienci nie są dostępne dla produktu z/OS.

Multi **Obsługa platform dla rozszerzonych klientów transakcyjnych**

Rozszerzone klienty transakcyjne są dostępne dla wszystkich platform Multiplatforms obsługujących klienta podstawowego. Klienci nie są dostępne dla produktu z/OS.

Aplikacja kliencka, która korzysta z rozszerzonego klienta transakcyjnego, może łączyć się z menedżerem kolejek tylko w następujących produktach IBM MQ 9.0 :

- **AIX** IBM MQ for AIX
- **IBM i** IBM MQ for IBM i
- **Linux** IBM MQ dla Linux
- **Windows** IBM MQ for Windows

z/OS Mimo że nie istnieją rozszerzone klienty transakcyjne, które działają w systemie z/OS, aplikacja kliencka używająca rozszerzonego klienta transakcyjnego może nawiązać połączenie z menedżerem kolejek, który działa w systemie z/OS.

W przypadku każdej platformy wymagania sprzętowe i programowe dla rozszerzonego klienta transakcyjnego są takie same jak wymagania dla klienta podstawowego IBM MQ . Język programowania jest obsługiwany przez rozszerzony klient transakcyjny, jeśli jest obsługiwany przez klienta podstawowego IBM MQ oraz przez menedżera transakcji, który jest używany.

Więcej informacji na temat menedżerów transakcji zewnętrznych dla wszystkich platform zawiera sekcja [Wymagania systemowe dla produktu IBM MQ](#).

Sposób łączenia się klienta z serwerem

Klient łączy się z serwerem za pomocą MQCONN lub MQCONNX, a następnie komunikuje się za pośrednictwem kanału.

Aplikacja działająca w środowisku klienta IBM MQ musi utrzymywać aktywne połączenie między komputerami klienta i serwera.

Połączenie jest nawiązywać przez aplikację, która wysyła wywołanie MQCONN lub MQCONNX. Klienci i serwery komunikują się za pośrednictwem *kanalów MQI* lub, w przypadku korzystania z współużytkowanych konwersacji, konwersacje współużytkują instancję kanału MQI. Gdy wywołanie powiedzie się, instancja kanału MQI lub konwersacja pozostają połączone do momentu wydania przez aplikację wywołania MQDISC. Ma to miejsce w przypadku każdego menedżera kolejek, z którym aplikacja musi się połączyć.

Pojęcia pokrewne

[“Klient i menedżer kolejek na tym samym komputerze” na stronie 152](#)

Aplikację można również uruchomić w środowisku IBM MQ MQI client , gdy na komputerze jest zainstalowany również menedżer kolejek.

[“Klienci na różnych platformach” na stronie 153](#)

Poniżej przedstawiono inny przykład systemu IBM MQ MQI client i serwera. W tym przykładzie komputer serwera komunikuje się z trzema IBM MQ MQI clients na różnych platformach.

[“Korzystanie z różnych wersji oprogramowania klienta i serwera” na stronie 153](#)

Jeśli używane są wcześniejsze wersje produktów IBM MQ , należy upewnić się, że konwersja kodu z identyfikatora CCSID klienta jest obsługiwana przez serwer.

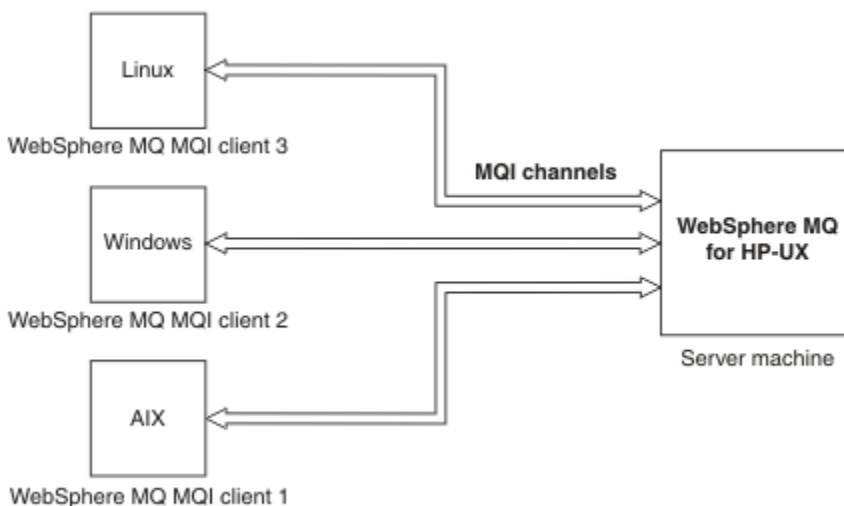
Klient i menedżer kolejek na tym samym komputerze

Aplikację można również uruchomić w środowisku IBM MQ MQI client , gdy na komputerze jest zainstalowany również menedżer kolejek.

W takiej sytuacji użytkownik może wybrać dowiązanie do bibliotek menedżera kolejek lub bibliotek klienta, ale należy pamiętać, że w przypadku połączenia z bibliotekami klienta nadal konieczne jest zdefiniowanie połączeń kanału. Może to być przydatne w fazie projektowania aplikacji. Program można testować na własnym komputerze, bez zależności od innych, i mieć pewność, że nadal będzie działał podczas przenoszenia go do niezależnego środowiska IBM MQ MQI client .

Klienci na różnych platformach

Poniżej przedstawiono inny przykład systemu IBM MQ MQI client i serwera. W tym przykładzie komputer serwera komunikuje się z trzema IBM MQ MQI clients na różnych platformach.



Rysunek 53. Serwer IBM MQ połączony z klientami na różnych platformach

Inne bardziej złożone środowiska są możliwe. Na przykład klient IBM MQ może łączyć się z więcej niż jednym menedżerem kolejek lub dowolną liczbą menedżerów kolejek połączonych w ramach grupy współużytkowania kolejek.

Korzystanie z różnych wersji oprogramowania klienta i serwera

Jeśli używane są wcześniejsze wersje produktów IBM MQ , należy upewnić się, że konwersja kodu z identyfikatora CCSID klienta jest obsługiwana przez serwer.

Klient produktu IBM MQ może nawiązać połączenie ze wszystkimi obsługiwanyymi wersjami menedżera kolejek. W przypadku nawiązywania połączenia z menedżerem kolejek we wcześniejszej wersji nie można używać funkcji i struktur z nowszej wersji produktu w aplikacji IBM MQ na kliencie.

Menedżer kolejek systemu IBM MQ może komunikować się z klientami w różnych wersjach, prowadząc negocjacje na najwyższym wzajemnie obsługiwany poziomie protokołu. Oznacza to, że starsze klienty mogą być używane z nowszymi poziomami menedżera kolejek. Zaleca się, aby zarówno klient, jak i serwer były w wersji produktu IBM MQ , która jest obecnie obsługiwana, w celu ułatwienia diagnozowania problemów i włączenia obsługi przez IBM.

Więcej informacji na ten temat zawiera sekcja dotycząca języków programowania obsługiwanych w sekcji [Tworzenie aplikacji](#).

Zarządzanie transakcjami i obsługa

Wprowadzenie do zarządzania transakcjami i sposobu obsługi transakcji przez produkt IBM MQ .

Menedżer zasobów jest podsystemem komputerowym, który jest właścicielem zasobów, do których można uzyskiwać dostęp i zarządzać nimi za pomocą aplikacji. Poniżej przedstawiono przykłady menedżerów zasobów:

- Menedżer kolejek produktu IBM MQ z zasobami, które są jego kolejkami.
- Baza danych Db2 , z zasobami, które są jej tabelami

Jeśli aplikacja zaktualizuje zasoby jednego lub większej liczby menedżerów zasobów, może istnieć wymaganie biznesowe, aby upewnić się, że niektóre aktualizacje zostały pomyślnie zakończone pomyślnie jako grupa lub żaden z nich nie został zakończony. Przyczyną tego typu wymagań jest pozostawienie niespójny stan danych biznesowych, jeśli niektóre z tych aktualizacji zostały zakończone pomyślnie, ale inne nie.

Aktualizacje zasobów, które są zarządzane w ten sposób, są powtarzane w *jednostce pracy* lub *transakcji*. Program użytkowy może grupować zestaw aktualizacji w jednostce pracy.

Podczas wykonywania jednostki pracy aplikacja wysyła żądania do menedżerów zasobów w celu zaktualizowania ich zasobów. Jednostka pracy kończy się, gdy aplikacja wysyła żądanie zatwierdzenia wszystkich aktualizacji. Dopóki aktualizacje nie zostaną zatwierdzone, żaden z nich nie stanie się widoczny dla innych aplikacji, które uzyskują dostęp do tych samych zasobów. Alternatywnie, jeśli aplikacja zdecyduje, że nie może zakończyć jednostki pracy z jakiegokolwiek powodu, może wydać żądanie, aby wycofać wszystkie aktualizacje, o które zażądano do tego momentu. W takim przypadku żadna z aktualizacji nie stanie się widoczna dla innych aplikacji. Aktualizacje te są zwykle logicznie powiązane i muszą być pomyślane w celu zachowania integralności danych. Jeśli jedna aktualizacja powiedzie się, podczas gdy inna nie powiedzie się, integralność danych zostanie utracona.

Gdy jednostka pracy zakończy się pomyślnie, jest ona *zatwierdzana*. Po zatwierdzeniu wszystkie aktualizacje dokonane w ramach tej jednostki pracy są trwałe i nieodwracalne. Jeśli jednak jednostka pracy nie powiedzie się, wszystkie aktualizacje są zamiast *wycofane*. Ten proces, w którym jednostki pracy są zatwierdzane lub wycofują się z integralności, jest nazywany *koordynacją punktów synchronizacji*.

Punkt w czasie, w którym wszystkie aktualizacje w jednostce pracy są zatwierdzane lub wycofane, jest nazywane *punktem synchronizacji*. Aktualizacja w ramach jednostki pracy jest powtarzana w *ramach elementu sterującego punktu synchronizacji*. Jeśli aplikacja żąda aktualizacji, która jest *poza kontrolą punktu synchronizacji*, menedżer zasobów zatwierdza aktualizację natychmiast, nawet jeśli istnieje jednostka pracy w toku, a aktualizacja nie może zostać wycofana później.

Podsystem komputera, który zarządza jednostkami pracy, jest nazywany *menedżerem transakcji* lub *koordynatorem punktu*.

lokalna jednostka pracy to jedna, w której jedynymi aktualizowanymi zasobami są te, które są aktualizowane przez menedżera kolejek produktu IBM MQ. W tym przypadku koordynacja punktów synchronizacji jest udostępniana przez sam menedżer kolejek przy użyciu procesu zatwierdzania jednofazowego.

Globalne jednostki pracy to jedna z nich, w której aktualizowane są również zasoby należące do innych menedżerów zasobów, takich jak bazy danych zgodne z XA. W tym przypadku należy użyć procedury zatwierdzania dwufazowego, a jednostka pracy może być koordynowana przez sam menedżer kolejek lub zewnętrznie przez inny menedżer transakcji zgodny z interfejsem XA, taki jak IBM TXSeries, lub BEA Tuxedo.

Menedżer transakcji jest odpowiedzialny za zapewnienie, że wszystkie aktualizacje zasobów w jednostce pracy zakończą się pomyślnie, lub żadne z nich nie zostało zakończone. Jest to dla menedżera transakcji, że aplikacja wysyła żądanie zatwierdzenia lub utworzenia kopii zapasowej jednostki pracy. Przykładami menedżerów transakcji są CICS i WebSphere Application Server, chociaż obie te funkcje mają również inną funkcję.

Niektóre menedżery zasobów udostępniają własną funkcję zarządzania transakcjami. Na przykład menedżer kolejek produktu IBM MQ może zarządzać jednostkami pracy obejmując aktualizacje własnych zasobów i aktualizacji tabel produktu Db2. Menedżer kolejek nie potrzebuje oddzielnego menedżera transakcji do wykonania tej funkcji, chociaż można go użyć, jeśli jest to wymaganie dotyczące użytkownika. Jeśli używany jest oddzielny menedżer transakcji, jest on określany jako *zewnętrzny menedżer transakcji*.

Aby menedżer transakcji zewnętrznej mógł zarządzać jednostką pracy, musi istnieć standardowy interfejs między menedżerem transakcji i każdym menedżerem zasobów, który uczestniczy w jednostce pracy. Ten interfejs umożliwia menedżerowi transakcji i menedżerowi zasobów komunikowanie się ze sobą. Jednym z tych interfejsów jest interfejs *XA Interface*, który jest standardowym interfejsem obsługiwany przez wiele menedżerów transakcji i menedżerów zasobów. Interfejs XA jest publikowany przez grupę Open Group w obszarze *Distributed Transaction Processing: The XA Specification* (Przetwarzanie transakcji rozproszonych: Specyfikacja XA).

Jeśli w jednostce pracy uczestniczy więcej niż jeden menedżer zasobów, menedżer transakcji musi użyć protokołu *zatwierdzania dwufazowego*, aby upewnić się, że wszystkie aktualizacje w jednostce pracy zostały pomyślnie zakończone lub żadne z nich nie zostało zakończone, nawet jeśli wystąpi awaria systemu. Gdy aplikacja wysyła żądanie do menedżera transakcji w celu zatwierdzenia jednostki pracy, menedżer transakcji wykonuje następujące czynności:

Faza 1 (Przygotowanie do zatwierdzenia)

Menedżer transakcji zwraca się do każdego menedżera zasobów uczestniczącego w jednostce pracy, aby upewnić się, że wszystkie informacje o planowanych aktualizacjach jego zasobów są w stanie odtwarzalnym. Menedżer zasobów zazwyczaj wykonuje ten proces, zapisując informacje w dzienniku i upewniając się, że informacje te są zapisywane na dysku twardym. Faza 1 kończy się, gdy menedżer transakcji otrzymuje powiadomienie od każdego menedżera zasobów o tym, że informacje o planowanych aktualizacjach zasobów są w stanie odtwarzalnym.

Faza 2 (zatwierdzenie)

Gdy faza 1 jest zakończona, menedżer transakcji podejmuje nieodwołalną decyzję o zatwierdzeniu jednostki pracy. Zwraca się do każdego menedżera zasobów, który uczestniczy w jednostce pracy w celu zatwierdzenia aktualizacji zasobów. Gdy menedżer zasobów odbierze to żądanie, musi zatwierdzić aktualizacje. Nie ma możliwości ich wycofać na tym etapie. Faza 2 kończy się, gdy menedżer transakcji otrzymuje powiadomienie od każdego menedżera zasobów, że zatwierdziło aktualizacje jego zasobów.

Interfejs XA korzysta z protokołu zatwierdzania dwufazowego.

Więcej informacji na ten temat zawiera sekcja [Scenariusze obsługi transakcyjnej](#).

Produkt IBM MQ udostępnia także obsługę serwera Microsoft Transaction Server (COM +). [Korzystanie z serwera transakcyjnego Microsoft \(COM +\)](#) zawiera informacje na temat sposobu konfigurowania produktu IBM MQ w celu korzystania z funkcji wsparcia COM +.

Rozszerzanie obiektów menedżera kolejek

Obiekty menedżera kolejek można rozszerzać, korzystając z wyjść użytkownika, wyjść funkcji API lub usług instalowalnych.

Procedury zewnętrzne

Procedury zewnętrzne udostępniają mechanizm, który umożliwia wstawienie własnego kodu do funkcji menedżera kolejek. Obsługiwane są następujące procedury zewnętrzne:

Wyjścia kanału

Te wyjścia zmieniają sposób działania kanałów. Wyjścia kanału są opisane w sekcji [Programy obsługi wyjścia kanału dla kanałów przesyłania komunikatów](#).

Wyjścia konwersji danych

Te wyjścia umożliwiają tworzenie fragmentów kodu źródłowego, które mogą być umieszczane w programach aplikacji w celu przekształcenia danych z jednego formatu na inny. Wyjścia konwersji danych są opisane w sekcji [Zapisywanie wyjść konwersji danych](#).

Wyjście obciążenia klastra

Funkcja wykonywana przez to wyjście jest zdefiniowana przez dostawcę wyjścia. Informacje o definicji połączenia są podane w pliku [MQ_CLUSTER_WORKLOAD_EXIT-Call description](#).

Wyjścia funkcji API

Wyjścia funkcji API umożliwiają zapisywanie kodu, który zmienia zachowanie wywołań interfejsu API produktu IBM MQ, takich jak MQPUT i MQGET, a następnie wstawiany jest ten kod bezpośrednio przed lub bezpośrednio po tych wywołaniach. Wstawianie jest automatyczne; menedżer kolejek kieruje kod wyjścia w zarejestrowanych punktach. Więcej informacji na temat wyjść funkcji API zawiera sekcja [Korzystanie i zapisywanie wyjść funkcji API](#).

Usługi instalowalne

Możliwe do zainstalowania usługi mają sformalizowane interfejsy (interfejs API) z wieloma punktami wejścia.

Implementacja usługi instalowalnej jest nazywana *komponentem usługi*. Komponentów dostarczanych z produktem IBM MQ można używać do pisania własnych komponentów w celu wykonywania wymaganych funkcji.

Obecnie dostępne są następujące usługi instalacyjne:

Usługa autoryzacji

Usługa autoryzacji pozwala na budowanie własnego narzędzia ochrony.

Domyślnym komponentem usługi, który implementuje usługę, jest menedżer uprawnień do obiektów (Object Authority Manager-OAM). Domyślnie program OAM jest aktywny, a użytkownik nie musi wykonywać żadnych czynności, aby go skonfigurować. Za pomocą interfejsu usługi autoryzacji można tworzyć inne komponenty w celu zastąpienia lub rozszerzenia OAM. Więcej informacji na temat programu OAM zawiera sekcja [Konfigurowanie zabezpieczeń w systemach AIX, Linux, and Windows](#).

usługa nazw

Usługa nazw umożliwia aplikacjom współużytkowanie kolejek przez zidentyfikowanie kolejek zdalnych tak, jakby były kolejkami lokalnymi.

Użytkownik może napisać własny komponent usługi nazw. Można to zrobić, jeśli na przykład ma być używana usługa nazw z produktem IBM MQ. Aby korzystać z usługi nazw, należy posiadać albo komponent, który jest napisany przez użytkownika, albo dostarczony przez innego dostawcę oprogramowania. Domyślnie usługa nazw jest nieaktywna.

Pojęcia pokrewne

[Procedury zewnętrzne, wyjścia funkcji API i usługi instalowalne produktu IBM MQ](#)

Interfejsy językowe programu IBM MQ Java

Produkt IBM MQ udostępnia dwa alternatywne aplikacyjne interfejsy programistyczne (API) do użycia w aplikacjach Java : IBM MQ classes for Java Message Service i IBM MQ classes for Java.

Produkt IBM obsługuje i jest aktywnym uczestnikiem otwartych standardów, a w obszarze przesyłania komunikatów standardem API jest Java Message Service (JMS). W produkcie IBM MQ 8.0 produkt implementuje standard JMS 2.0 , który wprowadza nowy uproszczony interfejs API wraz z funkcjami, takimi jak subskrypcje współużytkowane. Ponadto produkt WebSphere Liberty obsługuje produkt JMS 2.0 zarówno z domyślnym dostawcą przesyłania komunikatów, jak i z produktem IBM MQ.

W produkcie IBM MQ dostępne są dwa alternatywne interfejsy API do użycia w aplikacjach Java :

IBM MQ classes for JMS

IBM MQ classes for Java Message Service (JMS) jest dostawcą JMS dostarczonym z produktem IBM MQ. Java Platform, Enterprise Edition Connector Architecture (JCA) udostępnia standardowy sposób łączenia aplikacji działających w środowisku Java EE z systemem informacyjnym przedsiębiorstwa (Enterprise Information System-EIS), takim jak IBM MQ lub Db2.

IBM MQ classes for Java

Produkt IBM MQ classes for Java umożliwia korzystanie z produktu IBM MQ w środowisku produktu Java . Program IBM MQ classes for Java umożliwia aplikacji Java łączenie się z serwerem IBM MQ jako klient IBM MQ lub bezpośrednie połączenie z menedżerem kolejek produktu IBM MQ .

Uwaga: Produkt IBM MQ classes for Java jest stabilny pod względem funkcjonalnym na poziomie dostarczonym w produkcie IBM MQ 8.0. Istniejące aplikacje, które korzystają z produktu IBM MQ classes for Java , będą nadal w pełni obsługiwane, ale ta funkcja API jest stabilna, dlatego nowe funkcje nie zostaną dodane, a żądania dotyczące udoskonaleń zostały odrzucone. W pełni obsługiwane oznacza, że defekty zostaną naprawione wraz ze zmianami wymaganowymi przez zmiany w wymaganiach systemowych produktu IBM MQ .

W produkcie IBM MQ 8.0 IBM MQ classes for Java i IBM MQ classes for JMS są budowane z produktem Java 7. Środowisko wykonawcze produktu Java 7 obsługuje uruchamianie wcześniejszych wersji plików klas.

Pojęcia pokrewne

[Dlaczego należy używać produktu IBM MQ classes for JMS?](#)

[Dlaczego należy używać produktu IBM MQ classes for Java?](#)

[Model JMS](#)

Zadania pokrewne

[Korzystanie z funkcji produktu JMS 2.0](#)

IBM MQ classes for JMS

IBM MQ classes for JMS jest dostawcą JMS dostarczonym z produktem IBM MQ. Produkt IBM MQ classes for JMS implementuje interfejsy zdefiniowane w pakiecie javax.jms , a także udostępnia dwa zestawy rozszerzeń do interfejsu API produktu JMS . Zarówno aplikacje Java Platform, Standard Edition (Java SE), jak i Java Platform, Enterprise Edition (Java EE) mogą używać produktu IBM MQ classes for JMS.

Specyfikacja JMS definiuje zestaw interfejsów, których aplikacje mogą używać do wykonywania operacji przesyłania komunikatów. W produkcie IBM MQ 8.0 produkt obsługuje wersję JMS 2.0 standardu JMS . Ta implementacja oferuje wszystkie funkcje klasycznego interfejsu API, ale wymaga mniejszej liczby interfejsów i jest prostsza w użyciu. Więcej informacji na ten temat zawiera sekcja [“JMS model”](#) na stronie 158 oraz specyfikacja JMS 2.0 w serwisie [Java.net](#).

Pakiet javax.jms określa szczegółowe informacje na temat interfejsów produktu JMS , a dostawca produktu JMS implementuje te interfejsy dla konkretnego produktu przesyłania komunikatów. IBM MQ classes for JMS jest dostawcą JMS , który implementuje interfejsy JMS dla produktu IBM MQ , a także udostępnia następujące dwa zestawy rozszerzeń interfejsu API produktu JMS :

- Rozszerzenia IBM MQ JMS
- Rozszerzenia IBM JMS

Fabryka połączeń, kolejka lub obiekt tematu utworzone za pomocą interfejsu javax.jms lub zestawu rozszerzeń produktu JMS mogą być adresowane za pomocą dowolnego z tych interfejsów APIs; to znaczy można rzutować na dowolny z interfejsów. Aby zachować przenośność aplikacji na najwyższym poziomie, należy użyć najbardziej ogólnego interfejsu API, który jest odpowiedni dla wymagań użytkownika.

Rozszerzenia IBM MQ JMS

Produkt IBM MQ classes for JMS udostępnia również rozszerzenia interfejsu API produktu JMS . Poprzednie wersje produktu IBM MQ classes for JMS zawierają rozszerzenia implementowane w obiektach MQConnectionFactory, MQQueue i MQTopic. Te obiekty mają właściwości i metody specyficzne dla produktu IBM MQ. Obiektami mogą być obiekty administrowane, a aplikacja może tworzyć obiekty dynamicznie w czasie wykonywania. Produkt IBM MQ classes for JMS obsługuje te rozszerzenia, a użytkownik może nadal używać, bez zmian, żadnych aplikacji, które używają tych rozszerzeń. Rozszerzenia te są znane jako rozszerzenia IBM MQ JMS . Należy zauważyć, że w tym zestawie dokumentacji obiekty, które są tworzone dynamicznie przez aplikację w czasie wykonywania, nie są traktowane jako obiekty administrowane.

Rozszerzenia IBM JMS

Oprócz rozszerzeń IBM MQ JMS produkt IBM MQ classes for JMS udostępnia bardziej ogólny zestaw rozszerzeń do interfejsu API produktu JMS . Rozszerzenia te są znane jako rozszerzenia IBM JMS i mają następujące ogólne cele:

- Zapewnienie większego poziomu spójności dla dostawców IBM JMS
- Aby ułatwić zapis aplikacji pomostowej między dwoma systemami przesyłania komunikatów produktu IBM
- Aby ułatwić sobie port aplikacji z jednego dostawcy IBM JMS do innego

Główne uwagi dotyczące tych rozszerzeń dotyczą tworzenia i konfigurowania fabryk połączeń i miejsc docelowych dynamicznie w czasie wykonywania, ale rozszerzenia udostępniają również funkcje, które nie są bezpośrednio związane z przesyłaniem komunikatów, takie jak funkcja określania problemu.

JMS model

Model JMS definiuje zestaw interfejsów, które mogą być używane przez aplikacje produktu Java do wykonywania operacji przesyłania komunikatów. IBM MQ classes for JMS, jako dostawca JMS, definiuje sposób, w jaki obiekty JMS są powiązane z pojęciami IBM MQ. Specyfikacja JMS oczekuje, że niektóre obiekty JMS będą administrowane obiektami.

Specyfikacja JMS oraz pakiet `javax.jms` definiują zestaw interfejsów, które mogą być używane przez aplikacje produktu Java do wykonywania operacji przesyłania komunikatów.

W produkcie IBM MQ 8.0 produkt obsługuje wersję JMS 2.0 standardu JMS, która wprowadza uproszczony interfejs API, zachowując jednocześnie klasyczny interfejs API z produktu JMS 1.1.

Uproszczony interfejs API

Produkt JMS 2.0 wprowadza uproszczony interfejs API, zachowując jednocześnie specyficzne dla domeny interfejsy niezależne od produktu JMS 1.1. Uproszczona funkcja API redukuje liczbę obiektów potrzebnych do wysyłania i odbierania komunikatów, a także składa się z następujących interfejsów:

ConnectionFactory

Obiekt `ConnectionFactory` jest administrowany obiektem, który jest używany przez klienta JMS do tworzenia połączenia. Ten interfejs jest również używany w klasycznym interfejsie API.

JMSKontekst

Ten obiekt łączy obiekty połączenia i sesji klasycznego interfejsu API. Obiekty kontekstu produktu JMS mogą być tworzone z innych obiektów kontekstu produktu JMS, a bazowe połączenie jest duplikowane.

JMSProducent

Producent produktu JMS jest tworzony przy użyciu kontekstu produktu JMS i służy do wysyłania komunikatów do kolejki lub tematu. Obiekt producenta JMS powoduje utworzenie obiektów, które są wymagane do wysłania komunikatu.

JMSKonsument

Konsument produktu JMS jest tworzony przy użyciu kontekstu produktu JMS i służy do odbierania komunikatów z tematu lub kolejki.

Uproszczony interfejs API ma szereg efektów:

- Obiekt kontekstu JMS zawsze automatycznie uruchamia bazowe połączenie.
- Producenci JMS i konsumenci produktu JMS mogą teraz pracować bezpośrednio z ciałami komunikatów, bez konieczności pobierania całego obiektu komunikatu przy użyciu metody `getBody` komunikatu.
- Właściwości komunikatu można ustawić w obiekcie producenta produktu JMS przy użyciu łańcucha łączenia metod przed wysłaniem treści wiadomości. Producent produktu JMS będzie obsługiwać tworzenie wszystkich obiektów, które są potrzebne do wysłania komunikatu. Za pomocą składnika JMS 2.0 można ustawić właściwości i wysłać komunikat w następujący sposób:

```
context.createProducer().
setProperty("foo", "bar").
setTimeToLive(10000).
setDeliveryMode(NON_PERSISTENT).
setDisableMessageTimestamp(true).
send(dataQueue, body);
```

Produkt JMS 2.0 wprowadza również subskrypcje współużytkowane, w których komunikaty mogą być współużytkowane przez wielu konsumentów. Wszystkie subskrypcje produktu JMS 1.1 są traktowane jako subskrypcje niewspółużytkowane.

Klasyczne API

Poniższa lista zawiera podsumowanie głównych interfejsów produktu JMS w klasycznym interfejsie API:

Miejsce docelowe

Miejsce docelowe to miejsce, w którym aplikacja wysyła komunikaty lub jest to źródło, z którego aplikacja odbiera komunikaty, lub oba te komunikaty.

ConnectionFactory

Obiekt ConnectionFactory hermetyzuje zestaw właściwości konfiguracyjnych dla połączenia. Aplikacja korzysta z fabryki połączeń w celu utworzenia połączenia.

Połączenie

Obiekt połączenia hermetyzuje aktywne połączenie aplikacji z serwerem przesyłania komunikatów. Aplikacja korzysta z połączenia w celu utworzenia sesji.

Sesja

Sesja jest jednowątkowym kontekstem do wysyłania i odbierania komunikatów. Aplikacja korzysta z sesji w celu utworzenia komunikatów, producentów komunikatów i konsumentów komunikatów. Sesja jest transakcyjna lub nie jest transakcyjna.

Komunikat

Obiekt komunikatu hermetyzuje komunikat, który aplikacja wysyła lub odbiera.

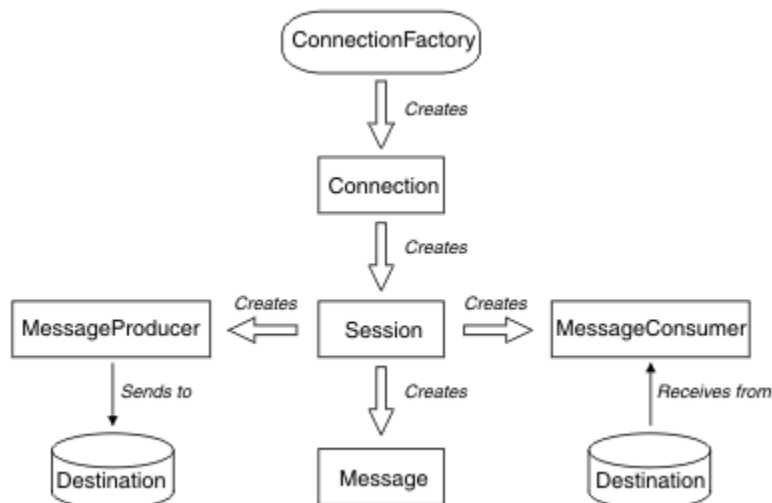
MessageProducer

Aplikacja korzysta z producenta komunikatów w celu wysyłania komunikatów do miejsca docelowego.

MessageConsumer

Aplikacja używa konsumenta komunikatów do odbierania komunikatów wysyłanych do miejsca docelowego.

Rysunek 54 na stronie 159 przedstawia te obiekty i ich relacje.



Rysunek 54. Obiekty JMS i ich relacje

Na diagramie przedstawiono główne interfejsy: ConnectionFactory, Connection, Session, MessageProducer, MessageConsumer, Message i Destination. Aplikacja korzysta z fabryki połączeń w celu utworzenia połączenia i korzysta z połączenia w celu utworzenia sesji. Aplikacja może następnie użyć sesji w celu utworzenia komunikatów, producentów komunikatów i konsumentów komunikatów. Aplikacja używa producenta komunikatów do wysyłania komunikatów do miejsca docelowego i używa konsumenta komunikatów do odbierania komunikatów wysyłanych do miejsca docelowego.

Obiekt docelowy, obiekt ConnectionFactory lub obiekt połączenia może być używany współbieżnie przez różne wątki aplikacji wielowątkowej, ale nie można jednocześnie używać obiektu Session, MessageProducer lub obiektu MessageConsumer przez różne wątki. Najprostszym sposobem zapewnienia, że obiekt Session, MessageProducer lub obiekt MessageConsumer nie jest używany współbieżnie, jest utworzenie osobnego obiektu sesji dla każdego wątku.

Produkt JMS obsługuje dwa style przesyłania komunikatów:

- Przesyłanie komunikatów w modelu punkt-punkt
- Przesyłanie komunikatów publikowania/subskrypcji

Te style przesyłania komunikatów są również nazywane *domenami przesyłania komunikatów*, a użytkownik może łączyć oba style przesyłania komunikatów w aplikacji. W domenie typu punkt z punktem miejsce docelowe jest kolejką, a w domenie publikowania/subskrybowania-miejszem docelowym jest temat.

W przypadku wersji produktu JMS przed JMS 1.1, programowanie dla domeny typu punkt z punktem wykorzystuje jeden zestaw interfejsów i metod, a programowanie dla domeny publikowania/subskrybowania używa innego zestawu. Oba zestawy są podobne, ale oddzielne. W produkcie JMS 1.1 można używać wspólnego zestawu interfejsów i metod obsługujących zarówno domeny przesyłania komunikatów. Wspólne interfejsy udostępniają niezależny widok domeny dla każdej domeny przesyłania komunikatów. Tabela 16 na stronie 160 zawiera listę niezależnych interfejsów domeny JMS i odpowiadających im interfejsów specyficznych dla domeny.

Tabela 16. Domena JMS niezależna i odpowiadające im interfejsy specyficzne dla domeny

| Niezależne interfejsy domeny | Interfejsy specyficzne dla domeny dla domeny punkt z punktem | Interfejsy specyficzne dla domeny dla domeny publikowania/subskrybowania |
|-------------------------------------|---|---|
| ConnectionFactory | Fabryka QueueConnection | Fabryka TopicConnection |
| Połączenie | QueueConnection | TopicConnection |
| Miejsce docelowe | Kolejka | Temat |
| Sesja | QueueSession | TopicSession |
| MessageProducer | QueueSender | TopicPublisher |
| MessageConsumer | QueueReceiver QueueBrowser | TopicSubscriber |

Program JMS 2.0 zachowuje wszystkie interfejsy specyficzne dla domeny, a więc istniejące aplikacje nadal mogą używać tych interfejsów. Jednak w przypadku nowych aplikacji należy rozważyć użycie niezależnych interfejsów domeny produktu JMS 1.1 lub uproszczonego interfejsu API produktu JMS 2.0.

W produkcie IBM MQ classes for JMS obiekty produktu JMS są powiązane z pojęciami produktu IBM MQ w następujący sposób:

- Obiekt połączenia ma właściwości, które są pobierane z właściwości fabryki połączeń, która została użyta do utworzenia połączenia. Te właściwości sterują sposobem łączenia się aplikacji z menedżerem kolejek. Przykładami tych właściwości są nazwa menedżera kolejek oraz, dla aplikacji, która łączy się z menedżerem kolejek w trybie klienta, nazwa hosta lub adres IP systemu, na którym jest uruchomiony menedżer kolejek.
- Obiekt sesji hermetyzuje uchwyt połączenia IBM MQ, który w związku z tym definiuje zasięg transakcyjny sesji.
- Obiekt MessageProducer i obiekt MessageConsumer każdy hermetyzuje uchwyt obiektu IBM MQ.

W przypadku korzystania z produktu IBM MQ classes for JMS stosowane są wszystkie normalne reguły produktu IBM MQ. Należy zwrócić uwagę, że aplikacja może wysłać komunikat do kolejki zdalnej, ale może odebrać komunikat tylko z kolejki, której właścicielem jest menedżer kolejek, z którym połączona jest aplikacja.

Specyfikacja JMS oczekuje, że obiekty ConnectionFactory i Destination mają być administrowane obiektami. Administrator tworzy i obsługuje administrowane obiekty w centralnym repozytorium, a aplikacja JMS pobiera te obiekty za pomocą interfejsu JNDI (Java Naming and Directory Interface).

W programie IBM MQ classes for JMS implementacja interfejsu docelowego jest abstrakcyjną nadklasą kolejki i tematu, a więc instancją miejsca docelowego jest obiekt kolejki lub obiekt tematu. Niezależne interfejsy domeny traktują kolejkę lub temat jako miejsce docelowe. Domena przesyłania komunikatów dla obiektu MessageProducer lub MessageConsumer jest określana na podstawie tego, czy miejscem docelowym jest kolejka, czy temat.

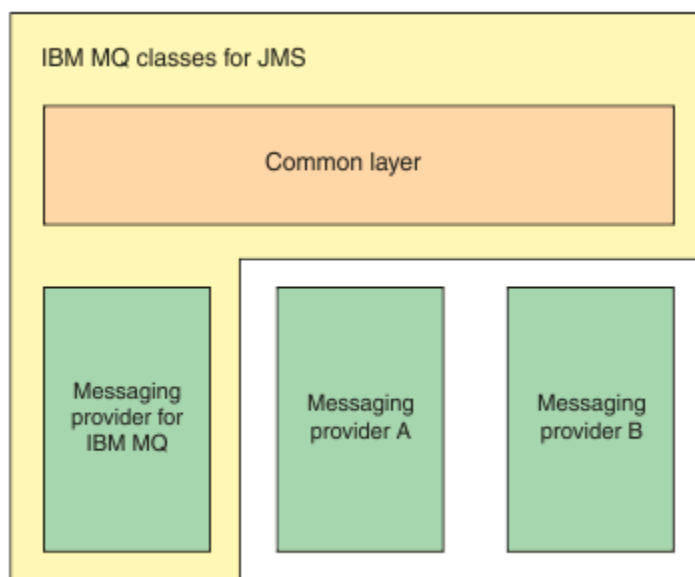
Dlatego w programie IBM MQ classes for JMS obiekty następujących typów mogą być administrowanymi obiektami:

- ConnectionFactory
- Fabryka QueueConnection
- Fabryka TopicConnection
- Kolejka
- Temat
- XAConnectionFactory
- Fabryka XAQueueConnection
- Fabryka XATopicConnection

Architektura produktu IBM MQ classes for JMS

Produkt IBM MQ classes for JMS ma architekturę warstwową. Warstwa najwyższej warstwy kodu jest wspólną warstwą, której może używać dowolny dostawca IBM JMS .

Produkt IBM MQ classes for JMS ma architekturę warstwową, jak pokazano na diagramie [Rysunek 55](#) na stronie 161. Warstwa najwyższej warstwy kodu jest wspólną warstwą, która może być używana przez dowolnego dostawcę IBM JMS . Jeśli aplikacja wywołuje metodę JMS , wszystkie przetwarzanie wywołania, które nie jest specyficzne dla systemu przesyłania komunikatów, jest wykonywane przez wspólną warstwę, która również zapewnia spójną odpowiedź na wywołanie. Każde przetwarzanie wywołania, które jest specyficzne dla systemu przesyłania komunikatów, jest delegowane do niższej warstwy. Na poniższym diagramie dostawca przesyłania komunikatów produktu IBM MQ jest wyświetlany w dolnej warstwie razem z dwoma innymi dostawcami przesyłania komunikatów (dostawca przesyłania komunikatów A i dostawca przesyłania komunikatów B.).



Rysunek 55. Architektura warstwową dla dostawców IBM JMS

Architektura warstwową spełnia następujące cele:

- W celu zwiększenia spójności działania różnych dostawców IBM JMS

- Aby ułatwić zapis aplikacji pomostowej między dwoma systemami przesyłania komunikatów produktu IBM
- Aby ułatwić sobie port aplikacji z jednego dostawcy IBM JMS do innego

Obsługa administrowanych obiektów

Produkt IBM MQ classes for JMS obsługuje użycie administrowanych obiektów.

Przeptyw logiki w aplikacji produktu JMS rozpoczyna się od obiektu ConnectionFactory i obiektów docelowych. Aplikacja korzysta z obiektu ConnectionFactory w celu utworzenia obiektu połączenia, który reprezentuje aktywne połączenie z aplikacji na serwer przesyłania komunikatów. Aplikacja korzysta z obiektu połączenia w celu utworzenia obiektu sesji, który jest pojedynczym kontekstem wielowątkowym służącym do tworzenia i konsumowania komunikatów. Aplikacja może następnie użyć obiektu Session i obiektu docelowego w celu utworzenia obiektu MessageProducer, którego aplikacja używa do wysyłania komunikatów do określonego miejsca docelowego. Miejsce docelowe jest kolejką lub tematem w systemie przesyłania komunikatów i jest hermetyzowane przez obiekt docelowy. Aplikacja może również użyć obiektu Session i obiektu docelowego w celu utworzenia obiektu MessageConsumer, którego aplikacja używa do odbierania komunikatów, które zostały wysłane do określonego miejsca docelowego.

Specyfikacja JMS oczekuje, że obiekty ConnectionFactory i Destination mają być administrowane obiektami. Administrator tworzy i obsługuje administrowane obiekty w centralnym repozytorium, a aplikacja JMS pobiera te obiekty za pomocą Java Naming Directory Interface (JNDI). Repozytorium administrowanych obiektów może być używane z prostego pliku do katalogu LDAP (Lightweight Directory Access Protocol).

Produkt IBM MQ classes for JMS obsługuje użycie administrowanych obiektów. Aplikacja może korzystać ze wszystkich funkcji produktu IBM MQ classes for JMS, które są ujawniane za pośrednictwem produktu IBM MQ, bez konieczności posiadania żadnych informacji specyficznych dla produktu IBM MQ, które zostały zakodowane w samej aplikacji. Ten układ udostępnia aplikację o stopniu niezależności od bazowej konfiguracji produktu IBM MQ. Aby uzyskać tę niezależność, aplikacja może używać interfejsu JNDI do pobierania fabryk połączeń i miejsc docelowych, które są przechowywane jako obiekty administrowane, i używać tylko interfejsów zdefiniowanych w pakiecie javax.jms w celu wykonywania operacji przesyłania komunikatów. Administrator może użyć narzędzia administracyjnego IBM MQ JMS lub programu IBM MQ Explorer, aby utworzyć i konserwować administrowane obiekty w centralnym repozytorium. Jednak serwer aplikacji zwykle udostępnia własne repozytorium dla administrowanych obiektów oraz własnych narzędzi do tworzenia i obsługi obiektów. Aplikacja Java EE może w związku z tym korzystać z produktu JNDI do pobierania administrowanych obiektów z repozytorium serwera aplikacji lub z repozytorium centralnego.

Informacje pokrewne

[Konfigurowanie zasobów JMS](#)

Obsługiwane typy komunikacji na platformach Java EE

Na platformie Java EE produkt IBM MQ classes for JMS obsługuje dwa typy komunikacji między komponentem aplikacji a menedżerem kolejek produktu IBM MQ.

Obsługiwane są następujące dwa typy komunikacji między komponentem aplikacji i menedżerem kolejek produktu IBM MQ:

- Komunikacja wychodząca
- Komunikacja przychodząca

Komunikacja wychodząca

Bezpośrednio przy użyciu interfejsu API produktu JMS komponent aplikacji tworzy połączenie z menedżerem kolejek, a następnie wysyła i odbiera komunikaty.

Komponentem aplikacji może być na przykład klient aplikacji, serwlet, strona serwera Java (JSP), komponent EJB (Enterprise Java Bean) lub komponent bean sterowany komunikatami (message driven

bean-MDB). W tym typie komunikacji kontener serwera aplikacji udostępnia tylko funkcje niskiego poziomu wspierające operacje przesyłania komunikatów, takie jak zestawianie połączeń i zarządzanie wątkami.

Komunikacja przychodząca

W przypadku komunikacji przychodzącej komunikat przychodzący do miejsca docelowego jest dostarczany do komponentu MDB, który następnie przetwarza ten komunikat.

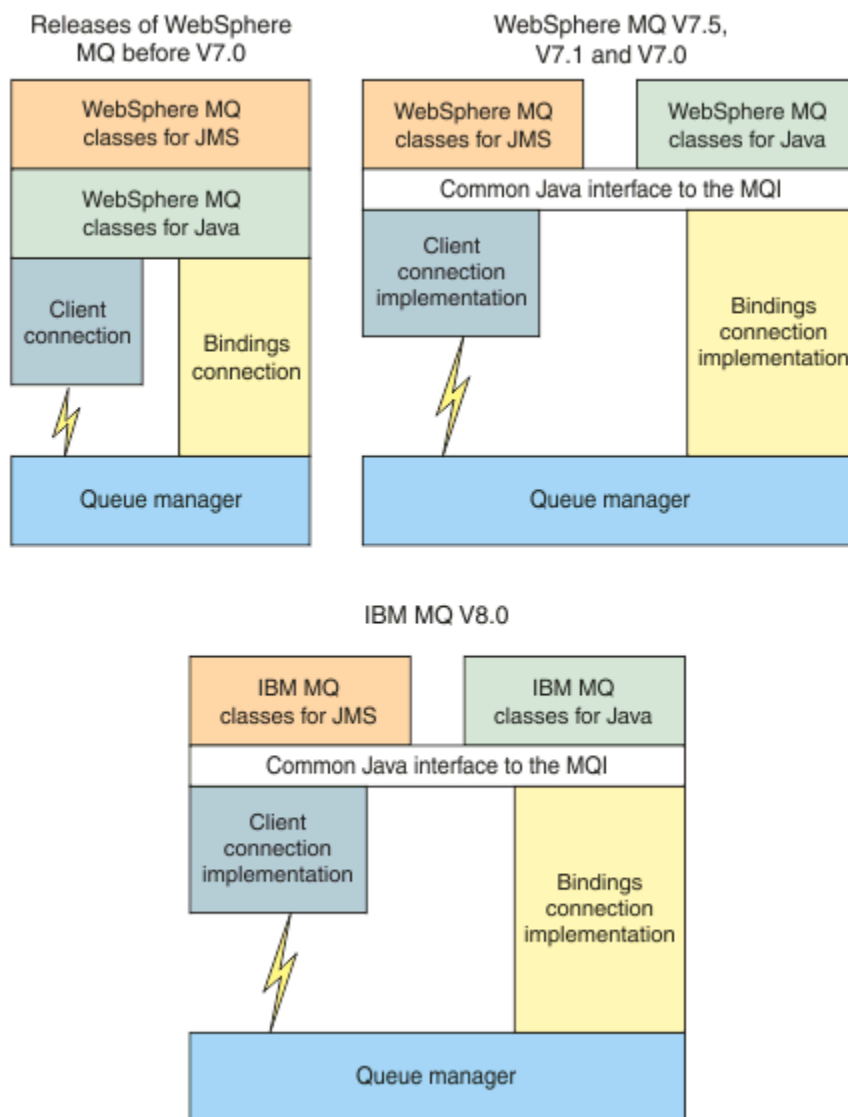
Aplikacje produktu Java EE korzystają z komponentów MDB w celu asynchronicznego przetwarzania komunikatów. Komponent MDB działa jako obiekt nasłuchiwanie komunikatów produktu JMS i jest implementowany przez metodę `onMessage()`, która definiuje sposób przetwarzania komunikatu. Komponent MDB jest wdrażany w kontenerze EJB serwera aplikacji. Dokładny sposób konfigurowania komponentu MDB zależy od używanego serwera aplikacji, ale informacje o konfiguracji muszą określać, który menedżer kolejek ma nawiązać połączenie, jak nawiązać połączenie z menedżerem kolejek, którego miejsce docelowe ma być monitorowane pod kątem komunikatów, a także zachowanie transakcyjne komponentu MDB. Informacje te są następnie używane przez kontener EJB. Gdy komunikat spełniający kryteria wyboru komponentu MDB nadchodzi do określonego miejsca docelowego, kontener EJB używa produktu IBM MQ classes for JMS do pobrania komunikatu z menedżera kolejek, a następnie dostarcza komunikat do komponentu MDB, wywołując jego metodę `onMessage()`.

Relacja z IBM MQ classes for Java

Ponieważ produkty IBM WebSphere MQ 7.0, IBM WebSphere MQ classes for JMS i IBM WebSphere MQ classes for Java są implementowane jako równorzędne. Ta implementacja różni się od wcześniejszych wersji, w których implementacja produktu IBM WebSphere MQ classes for JMS była uzależniana od IBM WebSphere MQ classes for Java.

W przypadku wersji wcześniejszych niż IBM WebSphere MQ 7.0 produkt IBM WebSphere MQ classes for JMS został zaimplementowany niemal w całości jako warstwa kodu w górnej części produktu IBM WebSphere MQ classes for Java. To rozwiązanie spowodowało pewne zamieszanie wśród twórców aplikacji, ponieważ ustawienie pól lub metod wywoływania w klasie `MQEnvironment` może spowodować niepożądane i nieoczekiwane skutki działania kodu napisanego przy użyciu produktu IBM MQ classes for JMS. Ponadto implementacja produktu IBM MQ classes for JMS miała pewne ograniczenia w obszarach, w których interfejs API produktu JMS nie jest naturalnym pasem w górnej części produktu IBM MQ classes for Java, a ograniczenia te doprowadziły do niektórych problemów dotyczących wydajności środowiska wykonawczego.

W produkcie IBM WebSphere MQ 7.0 implementacja produktu IBM MQ classes for JMS nie jest już zależna od produktu IBM MQ classes for Java. IBM MQ classes for Java i IBM MQ classes for JMS to równorzędne elementy, które używają wspólnego interfejsu Java do interfejsu MQI. Takie rozwiązanie pozwala na większą optymalizację wydajności i oznacza, że ustawienie pól lub metod wywoływania w klasie `MQEnvironment` nie ma wpływu na zachowanie kodu w środowisku wykonawczym napisanego przy użyciu produktu IBM MQ classes for JMS. [Rysunek 56 na stronie 164](#) przedstawia relację między IBM MQ classes for JMS a IBM MQ classes for Java w poprzednich wersjach produktu IBM WebSphere MQ classes for JMS i w wersjach wcześniejszych niż IBM WebSphere MQ 7.0 i w jaki sposób ta relacja została zmieniona w przypadku późniejszych wersji.



Rysunek 56. Relacja między IBM MQ classes for JMS i IBM MQ classes for Java

Aby zachować kompatybilność z wersjami przed IBM WebSphere MQ 7.0, klasy obsługi wyjścia kanału napisane w produkcie Java mogą nadal używać interfejsów IBM MQ classes for Java, nawet jeśli klasy wyjścia kanału są wywoływane z produktu IBM MQ classes for JMS. Jednak użycie interfejsów IBM MQ classes for Java oznacza, że aplikacje nadal są zależne od pliku JAR IBM MQ classes for Java, `com.ibm.mq.jar`. Jeśli produkt `com.ibm.mq.jar` nie ma być w ścieżce klasy, zamiast niego można użyć zestawu interfejsów w pakiecie `com.ibm.mq.exits`.

Za pomocą programu IBM WebSphere MQ 7.0 można tworzyć i konfigurować obiekty administrowane JMS za pomocą programu IBM MQ Explorer.

IBM MQ Dostawca komunikatów

Dostawca przesyłania komunikatów produktu IBM MQ ma trzy tryby działania: tryb normalny, tryb normalny z ograniczeniami i tryb migracji.

Dostawca przesyłania komunikatów produktu IBM MQ ma trzy tryby działania:

- Tryb normalny dostawcy przesyłania komunikatów produktu IBM MQ
- Tryb normalny dostawcy przesyłania komunikatów produktu IBM MQ z ograniczeniami

- Tryb migracji dostawcy przesyłania komunikatów produktu IBM MQ

Tryb normalny dostawcy przesyłania komunikatów produktu IBM MQ korzysta ze wszystkich funkcji menedżera kolejek produktu IBM MQ w celu zaimplementowania produktu JMS. Ten tryb jest zoptymalizowany pod kątem użycia funkcjonalności i interfejsu API JMS 2.0.

Tryb normalny dostawcy przesyłania komunikatów produktu IBM MQ z ograniczeniami używa interfejsu API produktu JMS 2.0, ale nie są to nowe funkcje produktu IBM MQ 8.0, takie jak subskrypcje współużytkowane, opóźnione dostarczanie lub wysyłanie asynchroniczne.

Tryb migracji dostawcy przesyłania komunikatów produktu IBM MQ jest oparty na funkcji IBM WebSphere MQ 6.0 i używa tylko funkcji, które były dostępne w menedżerze kolejek produktu IBM WebSphere MQ 6.0 w celu zaimplementowania produktu JMS. Możliwe jest nawiązanie połączenia z menedżerem kolejek, który korzysta z trybu migracji w wersji IBM WebSphere MQ 7.0 lub nowszej, ale nie można użyć żadnego z optymalizowanych IBM WebSphere MQ 7.0. Ten tryb umożliwia nawiązanie połączeń z jedną z następujących wersji menedżera kolejek:

1. IBM WebSphere MQ 7.0 lub nowsza, menedżer kolejek w powiązaniach lub trybie klienta, ale w tym trybie są używane tylko te opcje, które były dostępne dla menedżera kolejek produktu IBM WebSphere MQ 6.0.
2. IBM WebSphere MQ 6.0 lub wcześniejszy menedżer kolejek w trybie klienta.

Aby nawiązać połączenie z produktem IBM Integration Bus przy użyciu produktu IBM MQ Enterprise Transport, należy użyć trybu migracji. Jeśli używany jest produkt IBM MQ Transport w czasie rzeczywistym, tryb migracji jest wybierany automatycznie, ponieważ w obiekcie fabryki połączeń zostały jawnie wybrane właściwości. Połączenie z produktem IBM Integration Bus przy użyciu produktu IBM MQ Enterprise Transport jest zgodne z ogólnymi regułami wyboru trybu, które opisano w sekcji **Konfigurowanie właściwości JMS PROVIDERVERSION**.

Zadania pokrewne

[Konfigurowanie zasobów JMS](#)

z/OS IBM MQ for z/OS pojęcia

Niektóre z pojęć używanych przez produkt IBM MQ for z/OS są unikalne dla platformy z/OS. Na przykład mechanizm rejestrowania, techniki zarządzania pamięcią masową, dyspozycyjność jednostki i grupy współużytkowania kolejek są udostępniane tylko z produktem IBM MQ for z/OS. Ten temat zawiera dodatkowe informacje na temat tych pojęć.

Pojęcia pokrewne

[“Menedżer kolejek w systemie z/OS” na stronie 166](#)

Aby programy użytkowe mogły korzystać z produktu IBM MQ w systemie z/OS, należy zainstalować produkt IBM MQ for z/OS i uruchomić menedżer kolejek. Menedżer kolejek jest właścicielem zestawu zasobów, które są używane przez produkt IBM MQ, i zarządza nim.

[“Inicjator kanału w systemie z/OS” na stronie 167](#)

Inicjator kanału udostępnia zasoby, które umożliwiają kolejkowanie rozproszone IBM MQ. Produkt IBM MQ używa *agentów kanałów komunikatów* (Message Channel Agents) do wysyłania komunikatów z jednego menedżera kolejek do innego.

[“Terminy i zadania związane z zarządzaniem IBM MQ for z/OS” na stronie 169](#)

Ten temat zawiera wprowadzenie do terminologii oraz zadania specyficzne dla produktu IBM MQ for z/OS.

[“Współużytkowane kolejki i grupy współużytkowania kolejek” na stronie 172](#)

W celu zaimplementowania wysokiej dostępności zasobów produktu IBM MQ można używać współużytkowanych kolejek i grup współużytkowania kolejek. Kolejki współużytkowane i grupy współużytkowania kolejek są unikalne dla programu IBM MQ for z/OS na platformie z/OS.

[“Kolejkowanie wewnątrz grupy” na stronie 221](#)

W tej sekcji opisano kolejkowanie wewnątrz grupy, a funkcja IBM MQ for z/OS jest unikalna w stosunku do platformy z/OS. Ta funkcja jest dostępna tylko dla menedżerów kolejek zdefiniowanych w grupie współużytkowania kolejek.

[“Zarządzanie pamięcią masową w systemie z/OS” na stronie 234](#)

Produkt IBM MQ for z/OS wymaga trwałych i tymczasowych struktur danych oraz używa zestawów stron i buforów pamięci do przechowywania tych danych. W tych tematach przedstawiono bardziej szczegółowe informacje na temat używania przez produkt IBM MQ tych zestawów stron i buforów.

[“logowanie IBM MQ for z/OS” na stronie 239](#)

Produkt IBM MQ przechowuje *dzienniki* zmian danych i istotnych zdarzeń w miarę ich występowania. Dzienniki te mogą być używane do odzyskiwania danych do poprzedniego stanu, jeśli jest to wymagane.

[“Odtwarzanie i restartowanie w systemie z/OS” na stronie 262](#)

Odsyłacze w tym temacie umożliwiają znalezienie informacji na temat funkcji produktu IBM MQ for z/OS na potrzeby restartowania i odtwarzania.

[“Pojęcia dotyczące zabezpieczeń w produkcie IBM MQ for z/OS” na stronie 280](#)

W tym temacie opisano znaczenie zabezpieczeń produktu IBM MQ oraz implikacje braku odpowiednich ustawień zabezpieczeń w systemie.

[“Dostępność w systemie z/OS” na stronie 287](#)

Produkt IBM MQ for z/OS ma wiele funkcji w celu zapewnienia wysokiej dostępności. W tej sekcji opisano niektóre zagrożenia związane z dostępnością.

[“Dyspozycja jednostki odtwarzania w systemie z/OS” na stronie 291](#)

Niektóre aplikacje transakcyjne mogą używać grupy, a nie QMGR, jednostki odtwarzania po nawiązaniu połączenia z menedżerem kolejek w grupie współużytkowania kolejek (QSG), określając nazwę QSG podczas nawiązywania połączenia, a nie nazwy menedżera kolejek. Dzięki temu odtwarzanie transakcji może być bardziej elastyczne i stabilne przez usunięcie wymagania ponownego połączenia z tym samym menedżerem kolejek w QSG.

Odsyłacze pokrewne

[“Definiowanie systemu w systemie z/OS” na stronie 250](#)

Produkt IBM MQ for z/OS używa wielu domyślnych definicji obiektów i udostępnia przykładowe JCL w celu utworzenia tych obiektów domyślnych. W tym temacie opisano te obiekty domyślne i przykładowe JCL.

[“Monitorowanie i statystyka w systemie IBM MQ for z/OS” na stronie 290](#)

Produkt IBM MQ for z/OS zawiera zestaw narzędzi do monitorowania menedżera kolejek i zbierania statystyk.

z/OS

Menedżer kolejek w systemie z/OS

Aby programy użytkowe mogły korzystać z produktu IBM MQ w systemie z/OS, należy zainstalować produkt IBM MQ for z/OS i uruchomić menedżer kolejek. Menedżer kolejek jest właścicielem zestawu zasobów, które są używane przez produkt IBM MQ, i zarządza nim.

Menedżer kolejek

Menedżer kolejek to program, który udostępnia usługi przesyłania komunikatów aplikacjom. Aplikacje, które używają interfejsu kolejek komunikatów (Message Queue Interface - MQI) mogą umieszczać komunikaty w kolejkach i pobierać komunikaty z kolejek. Menedżer kolejek sprawdza, czy komunikaty są wysyłane do poprawnych kolejek lub czy są kierowane do innego menedżera kolejek. Menedżer kolejek przetwarza wywołania interfejsu MQI, które są do niego wysyłane oraz komendy, które są wprowadzane do menedżera kolejek (niezależnie od źródła, z którego pochodzą). Menedżer kolejek generuje odpowiednie kody zakończenia dla każdego wywołania lub komendy.

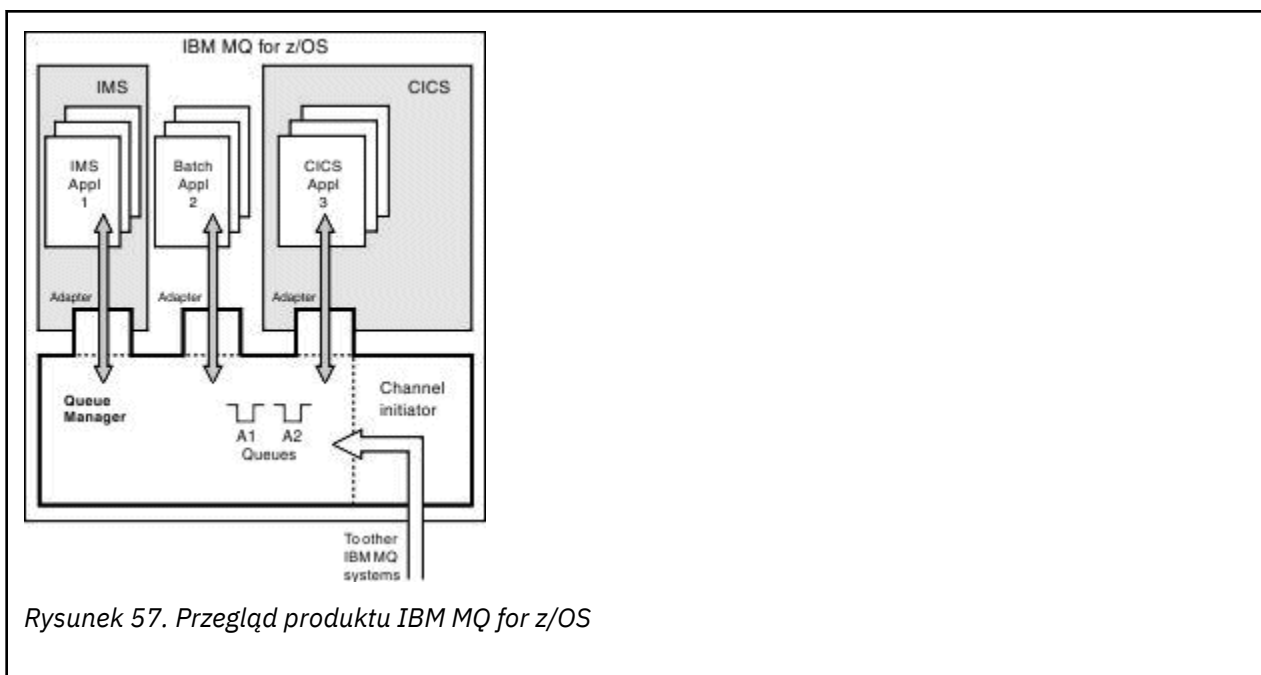
Do zasobów zarządzanych przez menedżera kolejek należą:

- Zestawy stron, w których przechowywane są definicje obiektów i komunikaty produktu IBM MQ
- Dzienniki, które są używane do odtwarzania komunikatów i obiektów w przypadku niepowodzenia menedżera kolejek
- Pamięć masowa procesora
- Połączenia, za pośrednictwem których różne środowiska aplikacji (CICS, IMSi Batch) mogą uzyskiwać dostęp do interfejsu API produktu IBM MQ

- Inicjator kanału IBM MQ , który umożliwia komunikację między programem IBM MQ w systemie z/OS a innymi systemami.

Menedżer kolejek ma nazwę, a aplikacje mogą łączyć się z nią przy użyciu tej nazwy.

Rysunek 57 na stronie 167 przedstawia menedżera kolejek, który przedstawia połączenia z różnymi środowiskami aplikacji i inicjatorem kanału.



Rysunek 57. Przegląd produktu IBM MQ for z/OS

Podsystem menedżera kolejek w systemie z/OS

W systemie z/OS program IBM MQ jest uruchamiany jako podsystem z/OS , który jest uruchamiany w czasie IPL. W podsystemie menedżer kolejek jest uruchamiany przez wykonanie procedury JCL, która określa zestawy danych programu z/OS zawierające informacje o dziennikach, a także definicje obiektów wstrzymania i dane komunikatu (zestawy stron). Podsystem i menedżer kolejek mają taką samą nazwę, co najwyżej cztery znaki. Wszystkie menedżery kolejek w sieci muszą mieć unikalne nazwy, nawet jeśli znajdują się w różnych systemach, syspleksach lub platformach.

z/OS Inicjator kanału w systemie z/OS

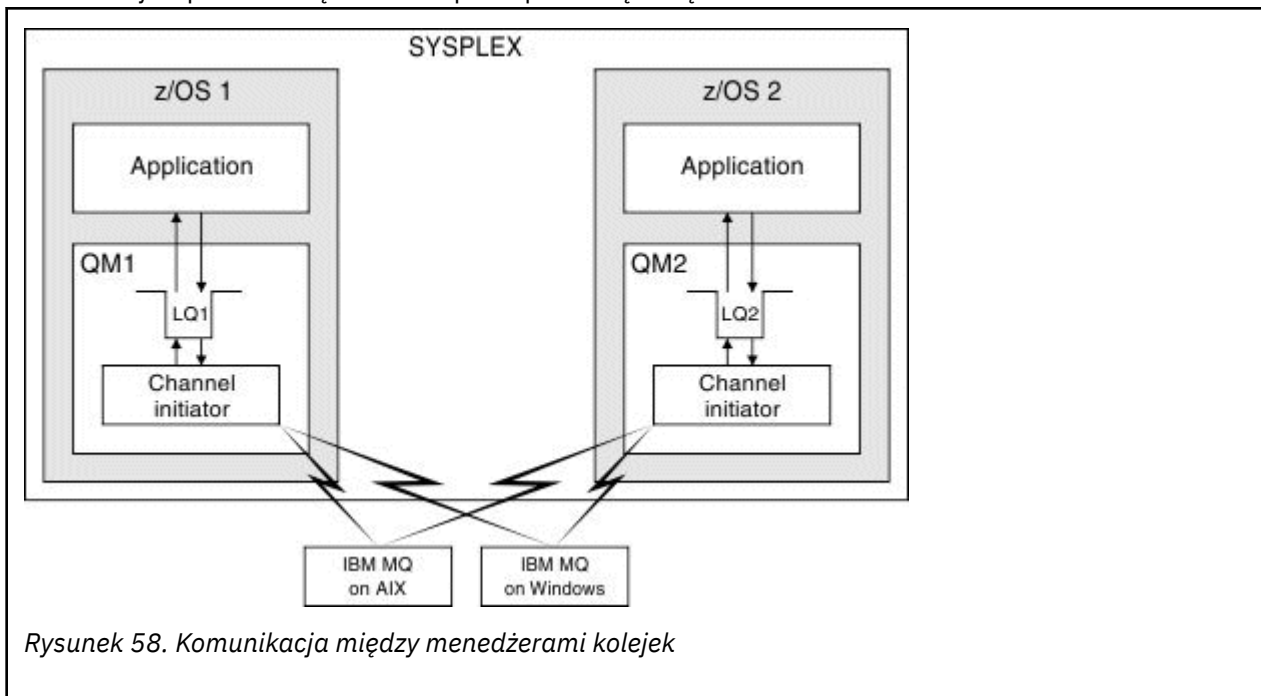
Inicjator kanału udostępnia zasoby, które umożliwiają kolejkowanie rozproszone IBM MQ . Produkt IBM MQ używa *agentów kanałów komunikatów* (Message Channel Agents) do wysyłania komunikatów z jednego menedżera kolejek do innego.

Aby wysłać komunikaty z menedżera kolejek A do menedżera kolejek B, *wysyłanie* MCA w menedżerze kolejek A musi skonfigurować łącze komunikacyjne do menedżera kolejek B. *odbierający* agent MCA musi być uruchomiony w menedżerze kolejek B, aby odbierać komunikaty z łącza komunikacyjnego. Ta jednokierunkowa ścieżka składająca się z wysyłającego agenta MCA, łącza komunikacyjnego i odbierającego agenta MCA jest znana jako *kanał*. Wysyłający agent MCA pobiera komunikaty z kolejki transmisji i wysyła je w dół kanału do odbierającego agenta MCA. Odbierający agent MCA odbiera komunikaty i umieszcza je w kolejkach docelowych.

W programie IBM MQ for z/OS: wysyłanie i odbieranie MCAs wszystkie uruchamiane wewnątrz inicjatora kanału (inicjator kanału jest znany również jako *mover*). Inicjator kanału jest uruchamiany jako przestrzeń adresowa z/OS pod kontrolą menedżera kolejek. Może istnieć tylko jeden inicjator kanału połączony z menedżerem kolejek i jest on uruchamiany wewnątrz tego samego obrazu z/OS co menedżer kolejek. W inicjatorze kanału może działać wiele procesów MCA działających jednocześnie.

W produkcie [Rysunek 58 na stronie 168](#) są wyświetlane dwa menedżery kolejek w ramach syspleksu. Każdy menedżer kolejek ma inicjator kanału i kolejkę lokalną. Komunikaty wysyłane przez menedżery

kolejek w systemach AIX i Windows są umieszczane w kolejce lokalnej, skąd są pobierane przez aplikację. Komunikaty odpowiedzi są zwracane przez podobną trasę.



Rysunek 58. Komunikacja między menedżerami kolejek

Inicjator kanału zawiera również inne procesy, których dotyczy zarządzanie kanałami. Procesy te obejmują:

Procesy nasłuchujące

Te procesy nasłuchują żądań kanału danych przychodzących w podsystemie komunikacyjnym, takim jak TCP, i uruchamiają nazwaną MCA, gdy odebrano żądanie przychodzące.

Nadzorca

Zarządza on przestrzenią adresową inicjatora kanału, na przykład odpowiada za restartowanie kanałów po awarii.

Serwer nazw

Służy do translacji nazw TCP na adresy.

Zadania TLS

Są one używane do wykonywania szyfrowania i deszyfrowania oraz sprawdzania list odwołań certyfikatów.

z/OS Rekordy SMF dla inicjatora kanału

Inicjator kanału (CHINIT) może generować rekordy statystyk SMF i rekordy rozliczeniowe z informacjami o zadaniach i kanałach.

CHINIT może generować rekordy statystyki SMF i rekordy rozliczeniowe, korzystając z następujących typów informacji:

- Zadania: przekaźnik, adapter, serwer nazw domen (DNS) i SSL. Te zadania tworzą tzw. statystyki CHINIT.
- Kanały: udostępnia informacje rozliczeniowe podobne do dostępnych za pomocą komendy DIS CHSTATUS. Jest to nazywane rachunkowość kanałów.

Produkt IBM MQ for Multiplatforms udostępnia podobne informacje, zapisując komunikaty PCF w systemie SYSTEM.ADMIN.STATISTICS.QUEUE. Więcej informacji na temat sposobu rejestrowania informacji statystycznych w systemie IBM MQ for Multiplatforms zawiera sekcja [Dane komunikatów statystyki kanału](#).

Dane statystyczne

Informacje te można wykorzystać w celu znalezienia następujących informacji:

- Określa, czy wymagane jest więcej zadań CHINIT, takich jak liczba obiektów TCB SSL i ilość czasu procesora używanego przez te zadania.
- Średni czas żądań dotyczących tych zadań.
- Najdłuższe żądanie czasu trwania w przedziale czasu i godzina wystąpienia tego wystąpienia dla zadań DNS i SSL. Możesz korelować tę porę dnia z problemami, które mogą Państwo doświadczyć z kanałem.

Dane księgowo

Informacje te mogą być używane do monitorowania wykorzystania kanałów oraz do wyszukiwania następujących informacji:

- Kanały o największej przepustowości.
- Szybkość wysyłania wiadomości oraz szybkość wysyłania danych w MB/s.
- Osiągnięta wielkość zadania wsadowego. Jeśli osiągnięta wielkość zadania wsadowego jest zbliżana do wielkości zadania wsadowego określonego dla kanału, może on być bliski jego limitu na potrzeby wysyłania komunikatów.

Za pomocą komend `START TRACE` i `STOP TRACE` można sterować gromadzeniem danych śledzenia rozliczania i śledzenia statystyk. Można użyć opcji `STATCHL` i `STATACLS` w kanale i menedżerze kolejek w celu określenia, czy kanały generują dane SMF.

z/OS

Terminy i zadania związane z zarządzaniem IBM MQ for z/OS

Ten temat zawiera wprowadzenie do terminologii oraz zadania specyficzne dla produktu IBM MQ for z/OS.

Niektóre z terminów i zadań wymaganych do zarządzania produktem IBM MQ for z/OS są specyficzne dla platformy z/OS. Poniższa lista zawiera niektóre z tych terminów i zadań.

- [Kolejki współużytkowane](#)
- [Zestawy stron i pule buforów](#)
- [Rejestrowanie](#)
- [Dostosowywanie środowiska menedżera kolejek](#)
- [Restartowanie i odtwarzanie](#)
- [Bezpieczeństwo](#)
- [Dostępność](#)
- [Manipulowanie obiektami](#)
- [Monitorowanie i statystyka](#)
- [Środowiska aplikacji](#)

Kolejki współużytkowane

Kolejki mogą być *niewspółużytkowane*, których właścicielem jest tylko jeden menedżer kolejek lub *współużytkowane*, których właścicielem jest *grupa współużytkowania kolejek*. Grupa współużytkowania kolejek składa się z wielu menedżerów kolejek działających w obrębie jednego systemu z/OS sysplex, które mogą współbieżnie uzyskiwać dostęp do tych samych definicji obiektów i komunikatów produktu IBM MQ. W ramach grupy współużytkowania kolejek definicje obiektów współużytkowalnych są przechowywane we współużytkowanej bazie danych Db2. Współużytkowane komunikaty kolejki są przechowywane w jednej lub większej liczbie struktur narzędzia CF (struktury CF). Jeśli dane komunikatu są zbyt duże do zapisania bezpośrednio w strukturze (o więcej niż 63 kB) lub jeśli komunikat jest wystarczająco duży, że reguły zdefiniowane przez instalację wybierają go do odciążenia, informacje dotyczące kontroli komunikatów są nadal zapisywane w pozycji narzędzia CF, ale dane komunikatu są

przenoszone do współużytkowanego zestawu danych komunikatów (SMDS) lub do współużytkowanej bazy danych Db2. Współużytkowane zestawy danych komunikatów, współużytkowana baza danych Db2 oraz struktury narzędzia CF to zasoby, które są wspólnie zarządzane przez wszystkie menedżery kolejek w grupie.

Zestawy stron i pule buforów

Gdy komunikat jest umieszczany w niewspółużytkowanej kolejce, menedżer kolejek zapisuje dane na stronie ustawionej w taki sposób, aby można ją było pobrać, gdy kolejna operacja pobiera komunikat z tej samej kolejki. Jeśli komunikat zostanie usunięty z kolejki, miejsce w zestawie stron, w którym przechowywane są dane, zostanie później zwolnione w celu ponownego wykorzystania. Liczba komunikatów wstrzymanych w kolejce zwiększa się, tak więc ilość miejsca używanego w zestawie stron zwiększa się, a liczba komunikatów w kolejce zmniejsza się, zmniejsza się ilość miejsca używanego w zestawie stron.

Aby zmniejszyć koszt wydajności zapisu danych do zestawów stron i odczytywania danych z zestawów stron, menedżer kolejek buforuje aktualizacje w pamięci masowej procesora. Ilość pamięci masowej używanej do buforowania dostępu do zestawu stron jest sterowana za pomocą obiektów IBM MQ o nazwie *Pule buforów*.

Więcej informacji na temat zestawów stron i pul buforów zawiera sekcja [Zarządzanie pamięcią masową](#).

Rejestrowanie

Wszelkie zmiany obiektów znajdujących się w zestawach stron oraz operacje na trwałych komunikatach są rejestrowane jako rekordy dziennika. Te rekordy dziennika są zapisywane w zestawie danych dziennika o nazwie *dziennik aktywny*. Nazwa i wielkość aktywnego zestawu danych dziennika są przechowywane w zestawie danych nazywanych *zestawem danych programu startowego* (BSDS).

Gdy aktywny zestaw danych dziennika zostanie zapełniony, menedżer kolejek przełącza się do innego zestawu danych dziennika, aby rejestrowanie było kontynuowane, a następnie kopiuje treść pełnego aktywnego zestawu danych dziennika do zestawu danych *dziennika archiwalnego*. Informacje na temat tych działań, w tym nazwa zestawu danych dziennika archiwalnego, są przechowywane w zestawie danych programu startowego. Koncepcyjnie jest to pierścienie aktywnych zestawów danych dziennika, przez które menedżer kolejek jest przełączany. Po zapełnieniu aktywnego dziennika dane dziennika są przenoszone do dziennika archiwalnego, a zestaw danych aktywnego dziennika jest dostępny do ponownego wykorzystania.

Więcej informacji na temat zestawów danych dziennika i programu startowego znajduje się w sekcji ["logowanie IBM MQ for z/OS"](#) na stronie 239.

Dostosowywanie środowiska menedżera kolejek

Po uruchomieniu menedżera kolejek zostanie odczytany zestaw parametrów inicjowania, które sterują sposobem działania menedżera kolejek. Dodatkowo odczytane są zestawy danych zawierające komendy IBM MQ, a komendy, które zawierają, są wykonywane. Zwykle te zestawy danych zawierają definicje obiektów systemowych wymaganych przez produkt IBM MQ do uruchomienia, a użytkownik może dostosować je do zdefiniowania lub zainicjowania obiektów produktu IBM MQ, które są niezbędne dla danego środowiska operacyjnego. Jeśli te zestawy danych zostały odczytane, wszystkie zdefiniowane przez nie obiekty są zapisywane na stronie lub w produkcie Db2.

Więcej informacji na temat parametrów inicjowania i obiektów systemowych zawiera sekcja ["Definiowanie systemu w systemie z/OS"](#) na stronie 250.

Odtwarzanie i restartowanie

W dowolnym momencie operacji IBM MQ mogą wystąpić zmiany w pamięci masowej procesora, które nie zostały jeszcze zapisane w zestawie stron. Te zmiany są zapisywane w zestawie stron, który jest najmniej ostatnio używany przez zadanie w tle w menedżerze kolejek.

Jeśli menedżer kolejek zakończy się nieprawidłowo, faza odtwarzania po restarcie menedżera kolejek może odzyskać utracone zmiany zestawu stron, ponieważ trwałe dane komunikatów są przechowywane w rekordach dziennika. Oznacza to, że program IBM MQ może odtworzyć dane i zmiany obiektu trwałego komunikatu, aż do momentu wystąpienia awarii.

Jeśli menedżer kolejek, który jest elementem grupy współużytkowania kolejek, napotka błąd narzędzia CF, trwałe komunikaty w tej kolejce mogą zostać odtworzone tylko wtedy, gdy utworzono kopię zapasową struktury narzędzia CF.

Więcej informacji na temat odtwarzania i restartowania zawiera sekcja [“Odtwarzanie i restartowanie w systemie z/OS”](#) na stronie 262.

Zabezpieczenia

Można użyć zewnętrznego menedżera zabezpieczeń, takiego jak Security Server (wcześniej znanego pod nazwą RACF) w celu ochrony zasobów, których właścicielem jest produkt IBM MQ, oraz zarządza nimi przed dostępem nieuprawnionych użytkowników. Można również użyć protokołu TLS (Transport Layer Security) w celu zabezpieczenia kanału. Protokół TLS jest dołączony jako część produktu IBM MQ.

Więcej informacji na temat zabezpieczeń produktu IBM MQ zawiera sekcja [“Pojęcia dotyczące zabezpieczeń w produkcie IBM MQ for z/OS”](#) na stronie 280.

Dostępność

Istnieje kilka funkcji produktu IBM MQ, które zostały zaprojektowane w celu zwiększenia dostępności systemu w przypadku awarii menedżera kolejek lub podsystemu komunikacyjnego. Więcej informacji o tych funkcjach zawiera temat [“Dostępność w systemie z/OS”](#) na stronie 287.

Manipulowanie obiektami

Gdy menedżer kolejek jest uruchomiony, można manipulować obiektami produktu IBM MQ za pomocą interfejsu konsoli produktu z/OS lub za pomocą narzędzia administracyjnego, które korzysta z usług ISPF w TSO. Oba mechanizmy umożliwiają definiowanie, modyfikowanie lub usuwanie obiektów produktu IBM MQ. Możliwe jest również sterowanie i wyświetlanie statusu różnych funkcji menedżera kolejek i IBM MQ.

Obiekty IBM MQ można także manipulować przy użyciu Eksploratora IBM MQ, graficznego interfejsu użytkownika, który udostępnia wizualny sposób pracy z kolejkami, menedżerami kolejek i innymi obiektami.

Więcej informacji na temat tych narzędzi znajduje się w sekcji [Komendy wydawane](#).

Monitorowanie i statystyka

Dostępnych jest kilka narzędzi do monitorowania menedżerów kolejek i inicjatorów kanałów. Istnieje również możliwość zbierania statystyk dotyczących oceny wydajności i celów księgowych.

Więcej informacji na temat tych narzędzi znajduje się w sekcji [“Monitorowanie i statystyka w systemie IBM MQ for z/OS”](#) na stronie 290.

Środowiska aplikacji

Gdy menedżer kolejek został uruchomiony, aplikacje mogą się z nim łączyć i uruchamiać za pomocą interfejsu API IBM MQ. Mogą to być aplikacje CICS, IMS, wsadowe lub WebSphere Application Server. Aplikacje produktu IBM MQ mogą także uzyskiwać dostęp do aplikacji w systemach CICS i IMS, które nie są świadome produktu IBM MQ, przy użyciu mostów CICS i IMS.

Więcej informacji na temat tych narzędzi znajduje się w sekcji [“IBM MQ i inne produkty z/OS”](#) na stronie 294.

Informacje na temat pisania aplikacji IBM MQ można znaleźć w następującej dokumentacji:

- [Projektowanie aplikacji](#)
- [Używanie języka C++](#)
- [Korzystanie z produktu IBM MQ classes for Java](#)

z/OS

Współużytkowane kolejki i grupy współużytkowania kolejek

W celu zaimplementowania wysokiej dostępności zasobów produktu IBM MQ można używać współużytkowanych kolejek i grup współużytkowania kolejek. Kolejki współużytkowane i grupy współużytkowania kolejek są unikalne dla programu IBM MQ for z/OS na platformie z/OS .

W tej sekcji opisano atrybuty i korzyści, a także informacje o tym, w jaki sposób kilka menedżerów kolejek może współużytkować te same kolejki i komunikaty w tych kolejkach.

Pojęcia pokrewne

[“Co to jest kolejka współużytkowana?”](#) na stronie 172

Kolejka współużytkowana jest typem kolejki lokalnej. Dostęp do komunikatów w tej kolejce może być uzyskiwany przez co najmniej jeden menedżer kolejek, który znajduje się w syspleksie.

[“Co to jest grupa współużytkowania kolejek?”](#) na stronie 174

Grupa menedżerów kolejek, które mogą uzyskiwać dostęp do tych samych współużytkowanych kolejek, jest nazywana grupą współużytkowania kolejek. Dla każdego podzbioru grupy współużytkowania kolejek udostępniony jest ten sam zbiór kolejek współużytkowanych.

[“Gdzie przechowywane są komunikaty kolejki współużytkowanej?”](#) na stronie 175

Każdy komunikat w kolejce współużytkowanej jest reprezentowany przez pozycję w strukturze listy narzędzia CF (Coupling Facility) systemu z/OS . Jeśli dane komunikatu są zbyt duże, aby zmieścić się w tej samej pozycji, są przenoszone do współużytkowanego zestawu danych komunikatu (SMDS) lub do produktu Db2.

[“Zalety korzystania z kolejek współużytkowanych”](#) na stronie 193

Kolejka współużytkowana pozwala na skalowalność, wysoką dostępność aplikacji IBM MQ i umożliwia zaimplementowanie równoważenia obciążenia.

[“Rozproszone grupy kolejkowania i współużytkowania kolejek”](#) na stronie 215

Rozproszone grupy kolejkowania i współużytkowania kolejek to dwie techniki, których można użyć w celu zwiększenia dostępności systemów aplikacji. W tym temacie znajdują się dalsze informacje na temat tych technik.

[“Wpływanie na dystrybucję obciążenia z kolejkami współużytkowanymi”](#) na stronie 219

W tym temacie opisano czynniki, które mają wpływ na dystrybucję obciążenia z kolejkami współużytkowanymi w grupie współużytkowania kolejek.

Odsyłacze pokrewne

[“Gdzie znaleźć więcej informacji o współużytkowanych kolejkach i grupach współużytkowania kolejek”](#) na stronie 220

Tabela w tym temacie zawiera więcej informacji na temat sposobu, w jaki produkt IBM MQ for z/OS używa współużytkowanych kolejek i grup współużytkowania kolejek.

z/OS

Co to jest kolejka współużytkowana?

Kolejka współużytkowana jest typem kolejki lokalnej. Dostęp do komunikatów w tej kolejce może być uzyskiwany przez co najmniej jeden menedżer kolejek, który znajduje się w syspleksie.

Grupa współużytkowania kolejek

Menedżery kolejek, które mogą uzyskiwać dostęp do tego samego zestawu kolejek współużytkowanych, tworzą grupę o nazwie *grupa współużytkowania kolejek*.

Dowolny menedżer kolejek może uzyskać dostęp do komunikatów

Każdy menedżer kolejek w grupie współużytkowania kolejek może uzyskać dostęp do współużytkowanej kolejki. Oznacza to, że można umieścić komunikat w kolejce współużytkowanej w jednym menedżerze kolejek i uzyskać ten sam komunikat z kolejki z innego menedżera kolejek. Zapewnia to szybki mechanizm komunikacji w ramach grupy współużytkowania kolejek, która nie wymaga aktywności kanałów między menedżerami kolejek.

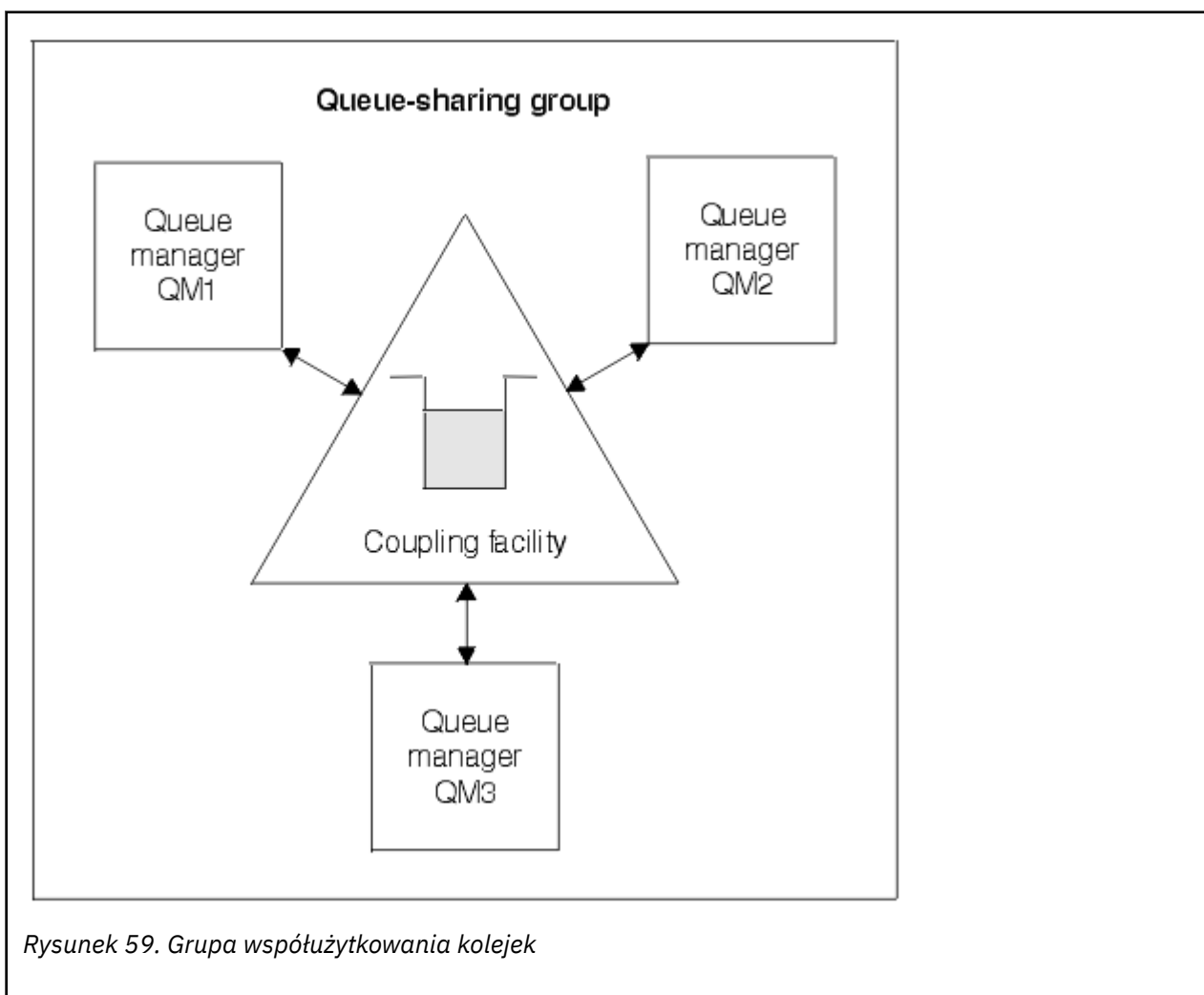
Produkt IBM WebSphere MQ 7.1 i nowszy obsługuje przenoszenie komunikatów do produktu Db2 lub współużytkowanego zestawu danych komunikatów (SMDS). Odciążenie wiadomości o dowolnej wielkości jest konfigurowalne.

We wcześniejszych wersjach produktu IBM MQ duże komunikaty (> 63 kB) mają obiekt zastępczy przechowywany w narzędziu CF (4 kB), a ich dane komunikatu są przechowywane w produkcie Db2.

Program [Rysunek 59 na stronie 173](#) wyświetla trzy menedżery kolejek i narzędzie CF tworzące grupę współużytkowania kolejek. Wszystkie trzy menedżery kolejek mogą uzyskać dostęp do współużytkowanej kolejki w narzędziu CF.

Aplikacja może połączyć się z dowolnym z menedżerów kolejek w grupie współużytkowania kolejek. Ponieważ wszystkie menedżery kolejek w grupie współużytkowania kolejek mogą uzyskiwać dostęp do wszystkich współużytkowanych kolejek, aplikacja nie jest zależna od dostępności konkretnego menedżera kolejek. Menedżer kolejek w grupie współużytkowania kolejek może korzystać z tej kolejki.

Daje to większą dostępność, ponieważ wszystkie inne menedżery kolejek w grupie współużytkowania kolejek mogą kontynuować przetwarzanie kolejki, jeśli jeden z menedżerów kolejek ma problem.



Rysunek 59. Grupa współużytkowania kolejek

Definicja kolejki jest współużytkowana przez wszystkie menedżery kolejek

Definicje kolejek współużytkowanych są przechowywane w tabeli OBJ_B_QUEUE bazy danych Db2 . Z tego powodu konieczne jest zdefiniowanie kolejki tylko raz, a następnie dostęp do niej przez wszystkie menedżery kolejek w grupie współużytkowania kolejek. Oznacza to, że istnieje mniej definicji do wykonania.

Natomiast definicja kolejki niewspółużytkowanej jest przechowywana na stronie zerowej w menedżerze kolejek, który jest właścicielem kolejki (zgodnie z opisem w sekcji [Zestawy stron](#)).

Nie można zdefiniować kolejki współużytkowanej, jeśli kolejka o tej nazwie została już zdefiniowana w zestawach stron definiującego menedżera kolejek. Analogicznie, nie można zdefiniować lokalnej wersji kolejki w zestawach stron menedżera kolejek, jeśli istnieje kolejka współużytkowana o tej samej nazwie.

Co to jest grupa współużytkowania kolejek?

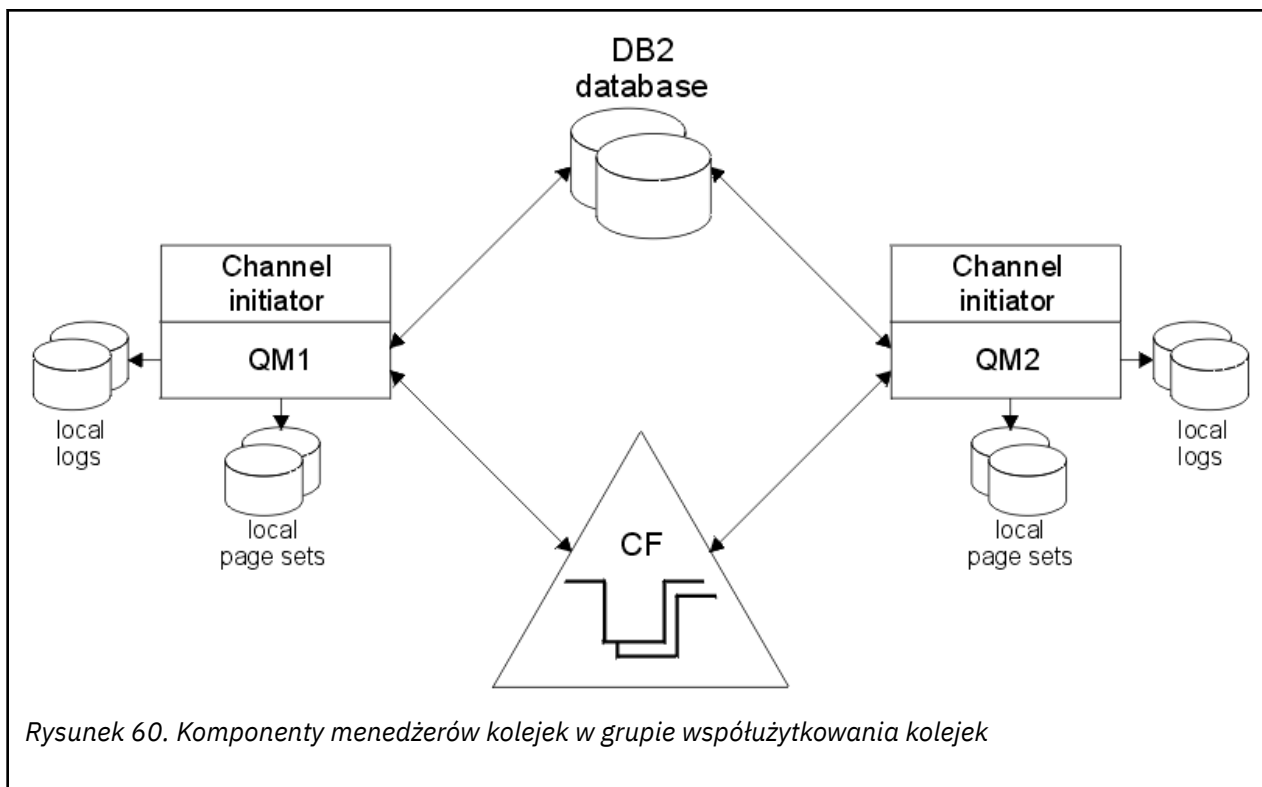
Grupa menedżerów kolejek, które mogą uzyskiwać dostęp do tych samych współużytkowanych kolejek, jest nazywana grupą współużytkowania kolejek. Dla każdego podzbioru grupy współużytkowania kolejek udostępniony jest ten sam zbiór kolejek współużytkowanych.

Nazwy grup współużytkujących kolejki składają się z maksymalnie czterech znaków. Nazwa taka musi być unikalna w danej sieci i nie może być identyczna z nazwą menedżera kolejek.

Rysunek 60 na stronie 175 przedstawia grupę współużytkowania kolejek, która zawiera dwa menedżery kolejek. Każdy menedżer kolejek ma inicjator kanału i własne zestawy stron lokalnych oraz zestawy danych dziennika.

Każdy członek grupy współużytkowania kolejek musi również łączyć się z systemem Db2 . Wszystkie systemy Db2 muszą znajdować się w tej samej grupie współużytkowania danych produktu Db2 , co umożliwia menedżerom kolejek uzyskiwanie dostępu do współużytkowanego repozytorium produktu Db2 używanego do przechowywania definicji współużytkowanych obiektów. Są to definicje dowolnego typu obiektu IBM MQ (na przykład kolejki i kanały), które są definiowane tylko raz, a następnie każdy menedżer kolejek w grupie może ich używać. Są one nazywane definicjami *globalnymi* i są opisane w sekcji [Definicje prywatne i globalne](#).

Więcej niż jedna grupa współużytkowania kolejek może odwoływać się do konkretnej grupy współużytkowania danych. Należy określić nazwę podsystemu Db2 i grupę współużytkowania danych używaną przez menedżer kolejek w parametrach systemowych IBM MQ podczas uruchamiania.



Gdy menedżer kolejek dołączył do grupy współużytkowania kolejek, ma on dostęp do współużytkowanych obiektów zdefiniowanych dla tej grupy. Można również użyć tego menedżera kolejek do zdefiniowania nowych współużytkowanych obiektów w grupie. Jeśli kolejki współużytkowane są zdefiniowane w grupie, można użyć tego menedżera kolejek w celu umieszczenia komunikatów w tych kolejkach współużytkowanych i pobierania komunikatów z tych kolejek. Każdy menedżer kolejek w grupie może pobierać komunikaty przechowywane w kolejce współużytkowanej.

Można wprowadzić komendę MQSC raz, a następnie wykonać ją we wszystkich menedżerach kolejek w grupie współużytkowania kolejek, tak jakby była wprowadzana osobno dla każdego menedżera kolejek. Do tego atrybutu używany jest atrybut *scope*. Ten atrybut jest opisany w sekcji [Kierowanie komend](#) do różnych menedżerów kolejek.

Gdy menedżer kolejek jest uruchamiany jako członek grupy współużytkowania kolejek, musi istnieć możliwość rozróżniania między obiektami produktu IBM MQ zdefiniowanymi prywatnie względem tego menedżera kolejek a obiektami produktu IBM MQ zdefiniowanymi globalnie, które są dostępne dla wszystkich menedżerów kolejek w grupie współużytkowania kolejek. Atrybut *dyspozycja grupy współużytkowania kolejki* jest używany na potrzeby tego atrybutu. Ten atrybut jest opisany w sekcji [Definicje prywatne i globalne](#).

Istnieje możliwość zdefiniowania jednego zestawu profili zabezpieczeń, które sterują dostępem do obiektów IBM MQ w dowolnym miejscu grupy. Oznacza to, że liczba profili, które należy zdefiniować, jest znacznie zmniejszona.

Menedżer kolejek może należeć tylko do jednej grupy współużytkowania kolejek, a wszystkie menedżery kolejek w grupie muszą znajdować się w tym samym sysplexie. Użytkownik określa, która grupa współużytkowania kolejek, do której należy menedżer kolejek, należy do parametrów systemu podczas uruchamiania.

▶ z/OS **Gdzie przechowywane są komunikaty kolejki współużytkowanej?**

Każdy komunikat w kolejce współużytkowanej jest reprezentowany przez pozycję w strukturze listy narzędzia CF (Coupling Facility) systemu z/OS. Jeśli dane komunikatu są zbyt duże, aby zmieścić się

w tej samej pozycji, są przenoszone do współużytkowanego zestawu danych komunikatu (SMDS) lub do produktu Db2.

Jeśli struktura CF została skonfigurowana do używania pamięci SCM (System Class Memory), produkt IBM MQ może użyć tej opcji bez dodatkowej konfiguracji. Patrz [IBM MQ V8 Features and Enhancements, rozdział 8](#).

Ważne: Planowane jest, że IBM z16 będzie ostatnią generacją IBM Z® obsługującej użycie wirtualnej pamięci Flash (znanej również jako pamięć klasy pamięci masowej lub SCM) dla obrazów narzędzia CF. Więcej informacji na ten temat zawierają: [IBM Z i IBM LinuxONE 4Q 2023 Oświadczenia o kierunku](#).

Alternatywnie należy użyć większych struktur lub przenieść komunikaty do SMDS.

Pamięć komunikatów kolejki współużytkowanej

Komunikaty umieszczane w kolejkach współużytkowanych nie są przechowywane w zestawach stron i nie korzystają z pul buforów.

Komunikaty w kolejkach współużytkowanych mają pozycje w strukturach listy w narzędziu CF (Coupling Facility) systemu z/OS. Wiele menedżerów kolejek w tym samym syspleksie może uzyskać dostęp do tych komunikatów przy użyciu struktury listy CF.

Dane komunikatu dla małych komunikatów kolejki współużytkowanej są zwykle uwzględniane w pozycji narzędzia CF. W przypadku większych komunikatów dane komunikatu mogą być przechowywane we współużytkowanym zestawie danych komunikatu (SMDS) lub jako jeden lub więcej dużych obiektów binarnych (BLOB) w tabeli Db2, która jest współużytkowana przez grupę współużytkowania danych Db2. Dane komunikatu przekraczające 63 kB są zawsze przenoszone do SMDS lub Db2. Mniejsze komunikaty mogą być również opcjonalnie przenoszone w ten sam sposób, aby zaoszczędzić miejsce w strukturze narzędzia CF. Więcej informacji na temat zawiera sekcja [“Określanie opcji odciążania dla komunikatów współużytkowanych”](#) na stronie 178.

Komunikaty umieszczane w kolejce współużytkowanej są przywoływane w strukturze narzędzia CF, dopóki nie zostaną pobrane za pomocą wywołania MQGET. Operacje narzędzia CF służą do:

- Wyszukaj następnym możliwym do pobrania komunikat
- Zablokuj niezatwierdzone komunikaty w kolejkach współużytkowanych
- Powiadomianie zainteresowanych menedżerów kolejek o nadejściu zatwierdzonych komunikatów

Operacje MQPUT i MQGET na trwałych komunikatach są zapisywane w dzienniku menedżera kolejek wykonującego tę operację. Minimalizuje to ryzyko utraty danych w przypadku awarii narzędzia CF.

Narzędzie CF

Komunikaty przechowywane w kolejkach współużytkowanych są przywoływane w narzędziu CF. Narzędzie CF znajduje się poza dowolnym obrazem z/OS w syspleksie i jest zwykle skonfigurowane do działania na innym zasilaczu. Narzędzie CF jest zatem odporne na awarie oprogramowania i można je skonfigurować w taki sposób, aby były odporne na awarie sprzętu lub przerwy w zasilaniu. Oznacza to, że komunikaty przechowywane w narzędziu CF mają wysoką dostępność.

Każda struktura listy narzędzia CF używana przez produkt IBM MQ jest dedykowana dla konkretnej grupy współużytkowania kolejek, ale narzędzie CF może przechowywać struktury dla więcej niż jednej grupy współużytkowania kolejek. Menedżery kolejek w różnych grupach współużytkowania kolejek nie mogą współużytkować danych. Maksymalnie 32 menedżery kolejek w grupie współużytkowania kolejek mogą jednocześnie łączyć się ze strukturą listy narzędzia CF.

Struktura listy pojedynczego narzędzia CF może zawierać maksymalnie 512 współużytkowanych kolejek. Łączna ilość danych komunikatu przechowywanych w strukturze jest ograniczona przez pojemność struktury. Jednak w systemie **CFLEVEL (5)** można użyć parametrów odciążania w celu odciążenia danych dla komunikatów o wielkości mniejszej niż 63 kB, co spowoduje zwiększenie liczby komunikatów, które

mogą być przechowywane w strukturze, chociaż każdy komunikat nadal wymaga co najmniej pozycji narzędzia CF powiększonej o co najmniej 512 bajtów danych.

Wielkość struktury listy jest ograniczona przez następujące czynniki:

- Musi leżeć w obrębie jednego narzędzia sprzęgającego.
- Może współużytkować dostępną pamięć masową narzędzia CF z innymi strukturami produktu IBM MQ i innych produktów.

Struktury listy narzędzia CF mogą mieć powiązaną pamięć klasy pamięci. W pewnych sytuacjach ta pamięć klasy pamięci może być przydatna, gdy jest używana z kolejkami współużytkowanymi. Więcej informacji zawiera sekcja [“Użycie pamięci klasy pamięci masowej z kolejkami współużytkowanymi”](#) na stronie 195.

Uwaga: W przypadku użycia IBM MQ można zaszyfrować strukturę narzędzia CF. Więcej informacji na ten temat zawiera sekcja [Szyfrowanie danych struktury narzędzia CF](#).

Planowanie wielkości struktury CF

Jeśli potrzebne są wskazówki dotyczące określania wielkości struktur CF, można użyć narzędzia [MP16: IBM MQ for z/OS Planowanie i strojenie mocy obliczeniowej](#). Można również użyć narzędzia z interfejsem WWW [CFSizer](#), które jest dostarczane przez IBM w celu ułatwienia obsługi wielkości systemów CF.

Obiekt struktury CF

Użycie struktury narzędzia CF przez menedżer kolejek jest określone w obiekcie IBM MQ struktury CF (CFSTRUCT).

Te obiekty struktury są przechowywane w katalogu Db2.

W przypadku używania komend lub definicji systemu z/OS związanych ze strukturą narzędzia CF wymagane są pierwsze cztery znaki nazwy grupy współużytkowania kolejek. Jednak obiekt IBM MQ CFSTRUCT zawsze istnieje w pojedynczej grupie współużytkowania kolejek i dlatego jego nazwa nie zawiera pierwszych czterech znaków nazwy grupy współużytkowania kolejek. Na przykład CFSTRUCT (MYDATA) zdefiniowany w grupie współużytkowania kolejek rozpoczynającej się od SQ03 używa struktury listy narzędzia CF SQ03MYDATA.

Struktury CF mają atrybut CFLEVEL określający ich możliwości funkcjonalne:

- 1, 2-mogą być używane dla nietrwałych komunikatów o wielkości mniejszej niż 63 kB
- 3-może być używany dla trwałych i nietrwałych komunikatów o wielkości mniejszej niż 63 kB
- 4-może być używany dla trwałych i nietrwałych komunikatów o wielkości do 100 MB
- 5-może być używane dla trwałych i nietrwałych komunikatów o wielkości do 100 MB i selektywnie przenoszone do współużytkowanych zestawów danych komunikatów (SMDS) lub Db2.

Tworzenie i odtwarzanie kopii zapasowych narzędzia CF

Za pomocą komendy IBM MQ [BACKUP CFSTRUCT](#) można tworzyć kopie zapasowe struktur listy narzędzia CF. Spowoduje to umieszczenie kopii komunikatów trwałych znajdujących się obecnie w strukturze systemu CF w zestawie danych aktywnego dziennika menedżera kolejek, który tworzy kopię zapasową, i zapisanie rekordu kopii zapasowej w pliku Db2.

Jeśli narzędzie CF ulegnie awarii, można użyć komendy IBM MQ [RECOVER CFSTRUCT](#). W tym celu używany jest rekord kopii zapasowej z systemu Db2 w celu znalezienia i odtworzenia trwałych komunikatów z kopii zapasowej struktury CF. Wszystkie działania od czasu utworzenia ostatniej kopii zapasowej są powtarzane przy użyciu dzienników wszystkich menedżerów kolejek w grupie współużytkowania kolejek, a struktura CF jest odtwarzana aż do momentu wystąpienia awarii.

Więcej szczegółów zawierają komendy [BACKUP CFSTRUCT](#) i [RECOVER CFSTRUCT](#).

Pojęcia pokrewne

“Określanie opcji odciążania dla komunikatów współużytkowanych” na stronie 178

Można wybrać miejsce, w którym przechowywane są dane komunikatu dla współużytkowanej kolejki, w tabeli Db2 lub w zestawie danych współużytkowanych komunikatów (SMDS). Można również wybrać, które komunikaty są odciążone, na podstawie wielkości komunikatu i bieżącego użycia struktury CF (Coupling Facility).

“Zarządzanie środowiskiem współużytkowanego zestawu danych komunikatów (SMDS)” na stronie 180
Jeśli wybrano współużytkowane zestawy danych komunikatów w celu przenoszenia dużych komunikatów, należy również pamiętać o informacjach używanych przez produkt IBM MQ do zarządzania tymi zestawami danych i komendach używanych do pracy z tymi informacjami. W tym temacie opisano sposób zarządzania współużytkowanymi zestawami danych komunikatów.

Określanie opcji odciążania dla komunikatów współużytkowanych

Można wybrać miejsce, w którym przechowywane są dane komunikatu dla współużytkowanej kolejki, w tabeli Db2 lub w zestawie danych współużytkowanych komunikatów (SMDS). Można również wybrać, które komunikaty są odciążone, na podstawie wielkości komunikatu i bieżącego użycia struktury CF (Coupling Facility).

Dane komunikatów dla kolejek współużytkowanych mogą być odciążane z narzędzia CF i zapisywane w tabeli Db2 lub w zarządzanym zestawie danych IBM MQ o nazwie *współużytkowany zestaw danych komunikatów* (SMDS).

W przypadku komunikatów większych niż wielkość pozycji narzędzia CF o wielkości 63 kB przesyłanie danych komunikatu do SMDS może mieć znaczną poprawę wydajności w porównaniu z odciążeniem do produktu Db2.

Każdy komunikat kolejki współużytkowanej jest nadal zarządzany za pomocą pozycji listy w strukturze narzędzia CF, ale gdy dane komunikatu są przenoszone do SMDS, pozycja narzędzia CF zawiera tylko niektóre informacje sterujące i listę odwołań do odpowiednich bloków dyskowych, w których przechowywany jest komunikat. Użycie tego mechanizmu oznacza, że ilość pamięci elementu narzędzia CF wymagana dla każdego komunikatu jest tylko ułamkiem rzeczywistej wielkości komunikatu.

Wybieranie miejsca, w którym są zapisywane komunikaty kolejki współużytkowanej

Wybór pamięci masowej współużytkowanych komunikatów SMDS lub Db2 jest kontrolowany za pomocą parametru **OFFLOAD (SMDS | DB2)** w definicji **CFSTRUCT**. **OFFLOAD (SMDS)** jest wartością domyślną.

Ten parametr wymaga również, aby produkt **CFSTRUCT** był używany w wersji **CFLEVEL (5)** lub nowszej. Tylko menedżery kolejek w systemie IBM WebSphere MQ 7.1 lub wyższym mogą łączyć się ze strukturą systemu CF na tym poziomie.

Strukturę można zmienić tylko na wartość **CFLEVEL (5)**, jeśli wszystkie menedżery kolejek w grupie współużytkowania kolejek są w wersji IBM WebSphere MQ 7.1 lub nowszej.

Parametr **OFFLOAD** jest poprawny tylko z poziomu **CFLEVEL (5)**. Więcej informacji na ten temat zawiera sekcja [DEFINE CFSTRUCT](#).

Produkt **OFFLOAD (DB2)** jest obsługiwany przede wszystkim w celu migracji.

Wybieranie, które komunikaty kolejki współużytkowanej są odciążane

Dane komunikatu są przenoszone do SMDS lub Db2 w zależności od wielkości danych komunikatu oraz bieżącego użycia struktury narzędzia CF. Istnieją trzy reguły, a każda reguła określa zgodną parę parametrów. Te parametry są odpowiednim procentem progu użycia struktury narzędzia CF (**OFFLDnTH**) i limit wielkości komunikatu (**OFFLDnSZ**).

Bieżąca implementacja trzech reguł jest określona przy użyciu następujących par słów kluczowych:

- OFFLD1TH i OFFLD1SZ
- OFFLD2TH i OFFLD2SZ
- OFFLD3TH i OFFLD3SZ

| Para reguł | Wartość domyślna | Opis |
|--------------|---------------------------------|--|
| Para reguł 1 | OFFLD1TH(70) i OFFLD1SZ(32K) | Jeśli struktura narzędzia CF przekracza 70% pełnego odciążania komunikatów o wielkości przekraczającej 32 kB |
| Para reguł 2 | OFFLD2TH(80) i OFFLD2SZ(4K) | Jeśli struktura narzędzia CF jest większa niż 80% pełnego odciążania dla komunikatów przekraczających 4 kB |
| Para reguł 3 | OFFLD3TH(90) i OFFLD3SZ(0K) | Jeśli struktura narzędzia CF przekracza 90% pełnego odciążania danych dla komunikatów przekraczających 0 kB (wszystkie komunikaty) |

Jeśli reguła odciążania ma wartość OFFLD x SZ o wartości 64K , oznacza to, że reguła nie jest w stanie. W tym przypadku komunikaty zostaną przesunięte tylko wtedy, gdy zostanie zastosowane inne reguły przenoszenia lub jeśli komunikat jest większy niż 63.75 kB, a więc zbyt duży, aby mógł on zostać zapisany w strukturze.

Każdy komunikat, który jest odciążony, nadal wymaga 0.75 KB pamięci masowej w narzędziu CF.

Trzy reguły odciążania, które mogą być określone dla każdej struktury, są przeznaczone do użycia w następujący sposób.

- Wydajność

- Jeśli w strukturze aplikacji jest dużo miejsca, dane komunikatu powinny być odciążone tylko wtedy, gdy jest zbyt duże do przechowywania w strukturze, lub jeśli przekracza dolny próg wielkości komunikatu w taki sposób, że wartość wydajności przechowywania go w strukturze nie jest warta ilości miejsca w strukturze, którą będzie potrzebował.
- Jeśli wymagany jest określony próg wielkości komunikatu, jest on określony w sposób konwencjonalny przy użyciu pierwszej reguły odciążania.

- pojemność

- Gdy w strukturze aplikacji znajduje się bardzo mało miejsca, maksymalna ilość danych komunikatu powinna być przesunięta w taki sposób, aby jak najlepiej wykorzystać pozostałą przestrzeń.
- Trzecia reguła przenoszenia jest zwykle używana w celu wskazania, że gdy struktura jest prawie pełna, większość komunikatów powinna być odciążana, więc wpisy w strukturze aplikacji będą zwykle mieć minimalną wielkość (wymagającą około 0.75K bajtów).
- Parametr progu użycia powinien być wybrany na podstawie wielkości struktury aplikacji i maksymalnego przewidywanego zalegania. Na przykład, jeśli maksymalny przewidywany dziennik zaległy to 1M , to ilość pamięci masowej wymaganej dla tej liczby komunikatów wynosi około 0.75G bajtów. Oznacza to na przykład, że jeśli struktura wynosi około 10G bajtów, próg użycia dla odciążania wszystkich komunikatów musi być ustawiony na 92% lub niższy.
- Obszar struktury jest podzielony na elementy i pozycje, a nawet jeśli istnieje wystarczająca ilość miejsca, jeden z nich może zostać uruchomiony przed drugim. System oferuje funkcję AUTOALTER, aby dostosować współczynnik, gdy jest to konieczne, ale nie jest to bardzo wrażliwe, więc ilość miejsca faktycznie dostępna może być nieco mniejsza. W związku z tym można lepiej wykorzystać nie więcej niż 90% maksymalnej powierzchni, tak więc w poprzednim przykładzie próg wykorzystania dla odciążania wszystkich wiadomości byłby lepiej ustawiony na poziomie około 80%.

- Przejście z poduszką:

- Ponieważ ilość miejsca pozostawione w strukturze narzędzia CF zmniejsza się, niepożądana zmiana charakterystyki wydajności powinna być niepożądana. Jest również niepożądana, aby funkcja zarządzania narzędzia CF miała nagłą zmianę wartości progowej w typowym stosunku pozycji do używanych elementów.
- Druga reguła przenoszenia jest używana w sposób konwencjonalny, aby zapewnić pewną pośrednią poduszkę między wydajnością i rozłożonymi w mocy regułą przenoszenia. Można go ustawić w taki sposób, aby spowodować znaczące zwiększenie aktywności przenoszenia, gdy przestrzeń używana w strukturze narzędzia CF przekroczy próg pośredni. Oznacza to, że pozostała przestrzeń jest używana wolniej, a mechanizm sprzęgający automatycznie modyfikuje czas przetwarzania, aby dostosować go do wyższych poziomów wykorzystania.

Jeśli struktura narzędzia CF nie może być rozwinięta, a istnieje potrzeba zapisania co najmniej pewnej z góry określonej liczby komunikatów, trzecia reguła może zostać zmodyfikowana w razie potrzeby w celu zapewnienia, że odciążenie danych dla wszystkich komunikatów rozpoczyna się od odpowiedniego progu w celu zapewnienia, że miejsce jest zarezerwowane dla określonej wcześniej liczby komunikatów.

Na przykład, jeśli wielkość struktury narzędzia CF wynosi 4 GB, a wstępnie określona liczba komunikatów wynosi 1 milion, wymagane są wartości $1\ 000\ 000 * 0.75\ \text{kB}$, czyli 768 MB, 18.75% z 4 GB. W takim przypadku próg odciążania wszystkich komunikatów musi być ustawiony na poziomie 80%, a nie 90%. Daje to parametry OFFLD3TH(80) i OFFLD3SZ(0K). Pozostałe parametry offload również powinny zostać dostosowane.

Jeśli zostanie stwierdzone, że odciążenie bardzo małych komunikatów ma istotny wpływ na wydajność, ale względny wpływ jest mniejszy na większe komunikaty, to progi użycia dla innych reguł mogą zostać zredukowane, aby odciążać większe komunikaty wcześniej, pozostawiając więcej miejsca w strukturze dla małych komunikatów, zanim będą one musiały zostać przesunięte.

Na przykład, jeśli komunikaty przekraczające 32KB występują często, ale wydajność ich odciążania (określona na podstawie statystyk RMF lub wydajności aplikacji) jest bardzo podobna do zachowania ich w narzędziu CF, to próg pierwszej reguły może być ustawiony na 0% w celu odciążania wszystkich takich komunikatów. Daje to parametry OFFLD1TH(0) i OFFLD1SZ(32K). Ponownie trzeba będzie skorygować inne parametry przenoszenia.

Jeśli istnieje wiele komunikatów dotyczących konkretnych wielkości pośrednich, takich jak 16 kB i 6 kB, może być przydatne zmianę opcji wielkości komunikatu dla drugiej reguły w taki sposób, aby większe były odciążone przy dość niskim progu użycia, zachowując znaczną ilość miejsca, ale te mniejsze nadal są zapisywane tylko w narzędziu CF.

Zarządzanie środowiskiem współużytkowanego zestawu danych komunikatów (SMDS)

Jeśli wybrano współużytkowane zestawy danych komunikatów w celu przenoszenia dużych komunikatów, należy również pamiętać o informacjach używanych przez produkt IBM MQ do zarządzania tymi zestawami danych i komendach używanych do pracy z tymi informacjami. W tym temacie opisano sposób zarządzania współużytkowanymi zestawami danych komunikatów.

Obiekty SMDS

Właściwości i status każdego zestawu danych komunikatów współużytkowanych są śledzone we współużytkowanym obiekcie SMDS, który może być aktualizowany za pośrednictwem dowolnego menedżera kolejek w grupie współużytkowania kolejek.

Dla każdego menedżera kolejek istnieje jeden współużytkowany zestaw danych komunikatu, który może uzyskać dostęp do każdej struktury aplikacji CF. Współużytkowany zestaw danych komunikatu jest identyfikowany przez nazwę menedżera kolejek będącego właścicielem określoną za pomocą słowa kluczowego SMDS oraz przez nazwę struktury aplikacji określoną za pomocą słowa kluczowego CFSTRUCT.

Uwaga: Podczas definiowania zestawów danych SMDS dla struktury należy mieć po jednym dla każdego menedżera kolejek.

Obiekt SMDS jest przechowywany w tablicy (z jedną pozycją na menedżera kolejek w grupie), która tworzy rozszerzenie odpowiedniego obiektu CFSTRUCT przechowywanego w programie Db2.

Nie ma komendy DEFINE lub DELETE dla obiektu SMDS, ponieważ jest on tworzony lub usuwany jako część obiektu CFSTRUCT, ale istnieje komenda ALTER umożliwiająca zmianę ustawień dla pojedynczego menedżera kolejek będącego właścicielem.

Więcej informacji na temat komend SMDS zawiera sekcja [“Komendy związane z SMDS”](#) na stronie 192

Informacje SMDSCONN

Możliwe, że współużytkowany zestaw danych komunikatów jest w stanie normalnym, ale co najmniej jeden menedżer kolejek nie może nawiązać z nim połączenia, na przykład z powodu problemu z definicją zabezpieczeń lub z połączeniem z urządzeniem o dostępie bezpośrednim. Dlatego konieczne jest, aby każdy menedżer kolejek śledził status połączenia i informacje o dostępności dla każdego zestawu danych komunikatów współużytkowanych, wskazując na przykład, czy może on obecnie nawiązać z nim połączenie, a jeśli nie, to dlaczego nie.

Informacje SMDSCONN reprezentują połączenie menedżera kolejek ze współużytkowanym zestawem danych komunikatów. Podobnie jak w przypadku samego współużytkowanego zestawu danych komunikatu, jest on identyfikowany przez menedżer kolejek, który jest właścicielem współużytkowanego zestawu danych komunikatu (określonego przez słowo kluczowe SMDS dla samego obiektu współużytkowanego) w połączeniu z nazwą CFSTRUCT.

Nie ma parametru identyfikującego połączony menedżer kolejek, ponieważ komendy adresowane do konkretnego menedżera kolejek mogą odwoływać się tylko do informacji SMDSCONN dla tego samego menedżera kolejek.

Pozycje informacji SMDSCONN są przechowywane w pamięci głównej w menedżerze kolejek będącym właścicielem i są ponownie tworzone po zrestartowaniu menedżera kolejek. Jeśli jednak połączenie z pojedynczego menedżera kolejek zostało jawnie zatrzymane, informacje te są również przechowywane jako flaga w tablicy połączeń w odpowiednim obiekcie CFSTRUCT lub SMDS, tak aby były zachowywane po restarcie menedżera kolejek.

Informacje o statusie i dostępności

Informacje o statusie wskazują stan zasobu lub połączenia (na przykład, czy zasób nie jest jeszcze używany, jest używany normalnie lub wymaga odtwarzania). Jest ona zwykle opisywana za pomocą słowa kluczowego STATUS. Możliwe wartości zależą od typu obiektu.

Informacje o statusie są zwykle aktualizowane automatycznie, na przykład w przypadku wykrycia błędu podczas korzystania z zasobu lub połączenia. Jednak w niektórych przypadkach można również użyć komendy w celu zaktualizowania statusu, aby umożliwić automatyczne określenie poprawnego statusu przez menedżera kolejek w przypadku, gdy nie jest to możliwe.

Informacje o dostępności wskazują, czy zasób lub połączenie mogą być używane i są zwykle określane głównie przez informacje o statusie. W przypadku typów zasobów lub połączeń używanych w obsłudze współużytkowanego zestawu danych komunikatów zaimplementowane są trzy poziomy dostępności:

Dostępny

Oznacza to, że zasób jest dostępny do normalnego użycia. Nie musi to oznaczać, że jest ona obecnie używana (można ją określić na podstawie wartości STATUS). W przypadku zestawu danych, jeśli wymaga on ponownego uruchomienia, pozwala to menedżerowi kolejek, do którego należy, na jego otwarcie, ale inne menedżery kolejek muszą czekać, aż zestaw danych znajdzie się ponownie w stanie ACTIVE.

Niedostępne z powodu błędu

Oznacza to, że zasób został automatycznie zablokowany z powodu błędu i nie powinien być ponownie dostępny, dopóki nie zostanie wykonana pewna forma naprawy lub odtwarzania. Jednakże próby ponownego udostępnienia są dozwolone bez interwencji operatora. Taka próba może być również wyzwalana przez komendę oznaczającą zasób jako włączony lub przez komendę, która zmienia status w taki sposób, aby wskazać, że przetwarzanie odtwarzania zostało zakończone.

Przyczyna, dla której zasób stał się niedostępny, jest zwykle oczywista w stosunku do powiązanej wartości STATUS, ale w niektórych przypadkach mogą istnieć inne powody, dla których zasób jest niedostępny. W takim przypadku podawana jest oddzielna wartość REASON wskazująca przyczynę.

Niedostępne z powodu komendy operatora

Oznacza to, że dostęp do zasobu został jawnie wyłączony przez komendę. Można go udostępnić tylko za pomocą komendy włączonej ponownie.

Dostępność SMDS

W przypadku współużytkowanego obiektu SMDS dostępność jest opisana przez słowo kluczowe ACCESS z możliwymi wartościami ENABLED, SUSPENDED i DISABLED.

Dostępność można zaktualizować przy użyciu komendy **RESET SMDS** dla odpowiedniego obiektu współużytkowanego z dowolnego menedżera kolejek w grupie, aby ustawić parametr ACCESS (ENABLED) lub ACCESS (DISABLED).

Jeśli dostępność była wcześniej określona jako ACCESS (SUSPENDED), zmiana na ACCESS (ENABLED) spowoduje wyzwolenie nowej próby użycia współużytkowanego zestawu danych komunikatów, ale jeśli poprzedni błąd nadal istnieje, dostępność zostanie przywrócona do ACCESS (SUSPENDED).

Dostępność SMDSCONN

W przypadku lokalnej pozycji informacji SMDSCONN dostępność jest opisana przez słowo kluczowe AVAIL z możliwymi wartościami NORMAL, ERROR lub STOPPED. Dostępność można zaktualizować przy użyciu komendy **START SMDSCONN** lub **STOP SMDSCONN** adresowanej do konkretnego menedżera kolejek w celu włączenia lub wyłączenia jego połączenia.

Jeśli wcześniej była dostępna usługa AVAIL (ERROR), zmiana jej na AVAIL (NORMAL) spowoduje nową próbę użycia współużytkowanego zestawu danych komunikatów, ale jeśli poprzedni błąd nadal istnieje, dostępność zostanie przywrócona do AVAIL (ERROR).

Współużytkowany status i dostępność zestawu danych komunikatów

Dostępność każdego współużytkowanego zestawu danych komunikatów jest zarządzana w ramach grupy przy użyciu współużytkowanych informacji o statusie, które można wyświetlić za pomocą komendy **DISPLAY CFSTATUS** z parametrem TYPE (SMDS). Spowoduje to wyświetlenie informacji o statusie dla każdego menedżera kolejek, który aktywował zestaw danych dla każdej struktury. Każdy zestaw danych może być w jednym z następujących stanów:

NIE ZNALEZIONO

Oznacza to, że odpowiedni zestaw danych nie został jeszcze aktywowany. Ten status jest wyświetlany tylko wtedy, gdy określono konkretny menedżer kolejek, ponieważ zestawy danych, które nie zostały aktywowane, są pomijane po wybraniu wszystkich menedżerów kolejek.

NOWOŚĆ

Zestaw danych jest otwierany i inicjowany po raz pierwszy, gotowy do aktywowania.

AKTYWNE

Oznacza to, że zestaw danych jest w pełni dostępny i powinien zostać przydzielony i otwarty przez wszystkie aktywne menedżery kolejek dla struktury.

Niepowodzenie

Oznacza to, że zestaw danych nie jest w ogóle dostępny (z wyjątkiem przetwarzania odtwarzania) i musi być zamknięty i zwalniany przez wszystkie menedżery kolejek.

NIEODZYSKAJ

Oznacza to, że dla tego zestawu danych trwa odzyskiwanie nośników (za pomocą komendy RECOVER CFSTRUCT).

Odtworzono

Oznacza to, że została wydana komenda przetłoczenia zestawu danych zakończonego niepowodzeniem z powrotem do stanu aktywnego, ale wymagane jest dalsze przetwarzanie restartu, które nie jest jeszcze zakończone, dlatego zestaw danych może zostać otwarty tylko przez menedżera kolejek będącego właścicielem na potrzeby przetwarzania restartu.

PUSTE

Zestaw danych nie zawiera komunikatów. Zestaw danych jest wprowadzany w ten stan, jeśli jest zamykany normalnie przez menedżera kolejek będącego właścicielem w momencie, gdy nie zawiera on żadnych komunikatów. Może być również w stanie EMPTY, gdy poprzednia zawartość zestawu danych ma zostać usunięta, ponieważ struktura aplikacji została opróżniona (przy użyciu parametru **RECOVER CFSTRUCT** z typem PURGE lub, tylko w przypadku struktury nienaprawialnej, przez usunięcie poprzedniej instancji struktury). Przy następnym otwarciu zestawu danych przez menedżera kolejek będącego właścicielem mapa obszaru zostanie zresetowana do wartości pustej, a status zostanie zmieniony na AKTYWNY. Ponieważ poprzednia zawartość zestawu danych nie jest już wymagana, zestaw danych w tym stanie można zastąpić nowo przydzielonym zestawem danych, na przykład w celu zmiany przydziału miejsca lub przeniesienia go na inny wolumin.

Dane wyjściowe komendy zawierają datę i godzinę włączenia rejestrowania odtwarzania, jeśli istnieje, oraz datę i godzinę niepowodzenia zestawu danych, jeśli nie jest on aktualnie aktywny.

Współużytkowany zestaw danych komunikatu może zostać przełączony w stan NIEPOWODZENIE za pomocą komendy **RESET SMDS** lub automatycznie po wykryciu dowolnego z następujących typów błędów:

- Zestaw danych nie może zostać przydzielony lub otwarty przez menedżera kolejek będącego właścicielem.
- Sprawdzanie poprawności nagłówka zestawu danych kończy się niepowodzeniem po pomyślnym otwarciu nagłówka przez dowolny menedżer kolejek.
- Trwały błąd we/wy występuje, gdy menedżer kolejek będący właścicielem odczytuje lub zapisuje dane.
- Trwały błąd we/wy występuje, gdy inny menedżer kolejek odczytuje dane z zestawu danych, który pomyślnie zakończył otwarte przetwarzanie i sprawdzanie poprawności.

Gdy zestaw danych jest w stanie NIEPOWODZENIE lub INRECOVER, nie jest on dostępny do normalnego użytku, więc jeśli stan dostępności to ACCESS (ENABLED), jest on zmieniany na ACCESS (SUSPENDED).

Jeśli zestaw danych został przełączony w stan NIEPOWODZENIE, ale nie jest wymagane odtwarzanie nośnika, na przykład dlatego, że dane były nadal poprawne, ale urządzenie pamięci masowej było tymczasowo w trybie bez połączenia, można użyć komendy **RESET SMDS**, aby zażądać zmiany statusu bezpośrednio na ODTWORZONY.

Po przejściu zestawu danych w stan ODTWORZONE, po zakończeniu przetwarzania odtwarzania lub w wyniku działania komendy **RESET SMDS**, zestaw danych jest gotowy do ponownego użycia po zakończeniu przetwarzania restartu. Jeśli był w stanie ACCESS (SUSPENDED), jest on automatycznie przełączany z powrotem do stanu ACCESS (ENABLED), co umożliwia menedżerowi kolejek, do którego należy, wykonanie przetwarzania restartu. Po zakończeniu przetwarzania restartu stan jest zmieniany na AKTYWNY, a wszystkie inne menedżery kolejek mogą ponownie nawiązać połączenie z zestawem danych.

Status i dostępność połączenia zestawu danych komunikatów współużytkowanych

Każdy menedżer kolejek przechowuje informacje o statusie lokalnym i dostępności na potrzeby połączenia z każdym współużytkowanym zestawem danych komunikatów, którego właścicielem jest ten sam menedżer kolejek oraz z innymi menedżerami kolejek w grupie. Te informacje można wyświetlić za pomocą komendy **DISPLAY SMDSCONN**.

Jeśli nie można uzyskać dostępu do współużytkowanego zestawu danych komunikatów w stanie ACTIVE, który należy do innego menedżera kolejek, oznacza to, że połączenie jest niedostępne z jego własnego punktu widzenia.

Jeśli błąd zdecydowanie wskazuje na problem z samym zestawem danych, menedżer kolejek również automatycznie zmienia status współużytkowania, aby wskazać, że zestaw danych jest teraz w stanie NIEPOWODZENIE. Jeśli jednak błąd może być spowodowany przez problem ze środowiskiem, na przykład brak uprawnień do otwarcia zestawu danych, menedżer kolejek wysyła komunikaty o błędach i traktuje zestaw danych jako niedostępny, ale nie modyfikuje statusu współużytkowanego zestawu danych. Jeśli błąd środowiskowy okaże się mimo to problemem z zestawem danych (na przykład został on przydzielony do urządzenia, do którego niektóre menedżery kolejek nie mają dostępu), operator może użyć komendy

RESET SMDS z parametrem STATUS (FAILED), aby umożliwić odtworzenie lub naprawę zestawu danych w razie potrzeby.

Jeśli nie można nawiązać połączenia ze współużytkowanym zestawem danych komunikatu, ale zestaw danych wydaje się poprawny, nowa próba użycia tego zestawu danych może zostać wywołana przez wydanie komendy **START SMDSCONN** dla menedżera kolejek będącego właścicielem.

Jeśli istnieje potrzeba działania w celu tymczasowego przerwania połączenia między konkretnym menedżerem kolejek i zestawem danych, ale sam zestaw danych nie jest uszkodzony, to zestaw danych można zamknąć i przydzielić za pomocą komendy **STOP SMDSCONN**. Jeśli zestaw danych jest używany, menedżer kolejek zamknie go normalnie (choć wszystkie żądania dotyczące danych w tym zestawie danych będą odrzucane z kodem powrotu). Jeśli jest to posiadany zestaw danych, menedżer kolejek zapisze mapę obszaru podczas przetwarzania CLOSE, unikając konieczności restartowania przetwarzania.

Jeśli zestaw danych musi zostać tymczasowo wycofany z użytku ze wszystkich menedżerów kolejek (na przykład w celu przeniesienia go), ale nie jest uszkodzony, zaleca się użycie parametru **STOP SMDSCONN** dla odpowiedniego zestawu danych z opcją CMDSCOPE (*), aby zatrzymać menedżery kolejek używające go jako pierwszego, ponieważ pozwoli to uniknąć konieczności restartowania przetwarzania po przywróceniu zestawu danych do eksploatacji. Dla porównania, jeśli zestaw danych jest oznaczony jako USZKODZONY, informuje to menedżery kolejek, że muszą natychmiast zaprzestać korzystania z tego zestawu, co oznacza, że mapa obszarów nie zostanie zapisana i będzie musiała zostać odbudowana przez ponowne uruchomienie przetwarzania.

Dostęp do wszystkich współużytkowanych zestawów danych komunikatów, które znajdowały się wcześniej w stanie ACCESS (SUSPENDED), zostanie ponowiony, jeśli menedżer kolejek zostanie zrestartowany.

Rejestrowanie odtwarzania współużytkowanego zestawu danych komunikatów

Trwałe komunikaty współużytkowane są rejestrowane na potrzeby odtwarzania nośników. Oznacza to, że komunikaty można odzyskać po awarii struktur narzędzia CF lub współużytkowanych zestawów danych komunikatów, pod warunkiem, że dzienniki odtwarzania są nadal nienaruszone. Trwałe komunikaty mogą być również ponownie tworzone z dzienników odtwarzania w innym ośrodku na potrzeby odtwarzania po awarii.

Gdy dane komunikatu są zapisywane we współużytkowanym zestawie danych komunikatu, każdy blok zapisywany w zestawie danych jest rejestrowany oddzielnie, po którym następuje pozycja komunikatu (wraz z odwzorowaniem danych) zapisana w narzędziu CF. Proces odtwarzania zawsze odtwarza strukturę narzędzia CF, ale nie musi odtwarzać pojedynczych współużytkowanych zestawów danych komunikatów, z wyjątkiem sytuacji, gdy status zestawu danych ma wartość NIEPOWODZENIE lub gdy status ma wartość AKTYWNE, ale rekord nagłówek zestawu danych nie jest już poprawny, co oznacza, że zestaw danych został ponownie utworzony. Zestaw danych nie jest wybierany do odtwarzania, jeśli jego status to ACTIVE, a nagłówek zestawu danych jest nadal poprawny, lub jeśli jego status to EMPTY, co oznacza, że w momencie niepowodzenia nie były w nim przechowywane żadne komunikaty.

Kopie zapasowe współużytkowanych zestawów danych komunikatów

Jeśli komenda BACKUP CFSTRUCT jest używana do tworzenia kopii zapasowej komunikatów współużytkowanych w strukturze aplikacji, wszystkie dane dla komunikatów trwałych zapisanych we współużytkowanych zestawach danych komunikatów są kopiowane w tym samym czasie, tak jak w przypadku trwałych komunikatów współużytkowanych poprzednio przechowywanych w bazie danych.

Odtwarzanie współużytkowanego zestawu danych komunikatów

Jeśli współużytkowany zestaw danych komunikatu jest uszkodzony lub utracony, należy go wprowadzić w stan NIEPOWODZENIE, aby uniemożliwić używanie tego zestawu przez menedżery kolejek do czasu, aż zostanie on naprawiony. Zwykle odbywa się to automatycznie, ale można to również zrobić za pomocą komendy **RESET SMDS** z parametrem STATUS (FAILED).

Jeśli współużytkowany zestaw danych komunikatu zawierał jakiegokolwiek komunikaty trwałe, można je odzyskać za pomocą komendy RECOVER CFSTRUCT. Ta komenda najpierw odtwarza dane komunikatów

trwałych dla tego zestawu danych komunikatów współużytkowanych z ostatniej komendy **BACKUP CFSTRUCT**, a następnie stosuje wszystkie zarejestrowane zmiany od tego czasu. Jeśli od czasu pierwszej aktywacji zestawu danych nie została wykonana żadna komenda **BACKUP CFSTRUCT**, zostanie ona zresetowana do wartości pustej, a następnie zostaną zastosowane wszystkie zmiany od momentu aktywacji.

Jeśli treść **CFSTRUCT** i wszystkie współużytkowane zestawy danych komunikatów są niedostępne, na przykład w sytuacji odtwarzania po awarii, wszystkie mogą zostać odzyskane za pomocą jednej komendy **RECOVER CFSTRUCT**.

Jeśli współużytkowany zestaw danych komunikatów jest uszkodzony, ale odzyskiwanie nie było aktywne dla **CFSTRUCT** lub dziennik zawierający najnowszą kopię zapasową **BACKUP CFSTRUCT** jest niedostępny lub nie można go użyć, to komunikaty przenoszone do tego zestawu danych nie mogą zostać odzyskane. W takim przypadku można użyć komendy **RECOVER CFSTRUCT** z parametrem **TYPE (PURGE)**, aby oznaczyć współużytkowany zestaw danych komunikatu jako pusty, a następnie usunąć wszystkie komunikaty ze struktury, w której znajdują się dane zapisane w tym zestawie danych.

Po wydaniu komendy **RECOVER CFSTRUCT** status zestawu danych komunikatów współużytkowanych jest zmieniany z **NIEPOWODZENIE** na **INRECOVER**. Jeśli odtwarzanie zakończy się pomyślnie, status zostanie automatycznie zmieniony na **ODZYSKANY**, w przeciwnym razie zostanie zmieniony z powrotem na **NIEPOWODZENIE**.

Po zmianie stanu zestawu danych na **ODTWORZONY** informuje on menedżera kolejek będącego właścicielem, że może on teraz podjąć próbę otwarcia zestawu danych i wykonania przetwarzania restartu.

Odtwarzanie współużytkowanego zestawu danych komunikatów i punkty synchronizacji

Proces odtwarzania współużytkowanego zestawu danych komunikatów ponownie stosuje zmiany dla wszystkich pełnych rekordów dziennika do końca dziennika, niezależnie od punktów synchronizacji.

Jeśli zmiany zostały wprowadzone w punkcie synchronizacji, przetwarzanie operacji restartowania lub odtwarzania dla **CFSTRUCT** może spowodować wycofanie niezatwierdzonych żądań, dlatego niektóre z odtworzonych zmian mogą nie być w rzeczywistości używane, ale mimo to nie ma żadnej szkody w ich odzyskiwaniu.

Możliwe jest również, że niezatwierdzony komunikat **MQPUT** został zapisany w strukturze, ale odpowiednie dane mogły nie zostać zapisane w zestawie danych lub w dzienniku (ponieważ zakończenie operacji **we/wy** jest wymuszane tylko na początku przetwarzania punktu synchronizacji). Jest to nieszkodliwe, ponieważ proces restartowania wycofa pozycję komunikatu w strukturze, więc fakt, że odnosi się ona do nieodzyskanych danych, nie ma znaczenia.

Przetwarzanie restartu współużytkowanego zestawu danych komunikatu

Jeśli połączenie menedżera kolejek z **CFSTRUCT** zostanie zakończone normalnie, menedżer kolejek zapisuje mapę wolnego obszaru bloków dla każdego współużytkowanego zestawu danych komunikatu w obszarze punktu kontrolnego w zestawie danych, tuż przed zamknięciem zestawu danych. Odwzorowanie obszaru można następnie wczytać ponownie podczas restartu połączenia, pod warunkiem, że ani **CFSTRUCT**, ani współużytkowany zestaw danych komunikatu nie wymagają przetwarzania odtwarzania przed następnym restartem.

Jeśli jednak menedżer kolejek zostanie zakończony nieprawidłowo lub struktura lub zestaw danych wymagają przetwarzania odtwarzania, w celu dynamicznego odbudowania mapy obszarów po zrestartowaniu połączenia menedżera kolejek ze strukturą wymagane jest dodatkowe przetwarzanie.

Pod warunkiem, że sam zestaw danych nie wymaga odzyskiwania, menedżer kolejek restartuje po prostu skanuje bieżącą zawartość struktury, aby znaleźć odwołania do danych komunikatu, których właścicielem jest bieżący menedżer kolejek, i oznacza odpowiednie bloki danych jako należące do odwzorowania obszaru. Inne menedżery kolejek mogą nadal używać tej struktury i odczytywać dane, których właścicielem jest restartowany menedżer kolejek, podczas odbudowywania mapy obszarów.

Restart współużytkowanego zestawu danych komunikatu po odtworzeniu

Jeśli współużytkowany zestaw danych komunikatu musiał zostać odzyskany z kopii zapasowej, wszystkie nietrwale komunikaty zapisane w zestawie danych zostaną utracone, a jeśli zestaw danych został odzyskany przy użyciu komendy TYPE (PURGE), wszystkie komunikaty zapisane w zestawie danych zostaną utracone. Do czasu zakończenia odtwarzania zestaw danych zostanie oznaczony jako NIEPOWODZENIE lub INRECOVER, więc każda próba odczytania jednego z komunikatów z innego menedżera kolejek powoduje zwrócenie kodu błędu wskazującego, że zestaw danych jest tymczasowo niedostępny.

Po odtworzeniu zestawu danych jego status jest zmieniany na ODZYSKANY, co umożliwia menedżerowi kolejek, który jest właścicielem, otwarcie go w celu zrestartowania, ale zestaw danych pozostaje niedostępny dla innych menedżerów kolejek. Restart menedżera kolejek skanuje strukturę w celu odbudowania mapy obszarów dla pozostałych komunikatów. Skanowanie sprawdza również komunikaty, dla których dane zostały utracone, i usuwa je ze struktury (lub, jeśli to konieczne, oznacza je jako utracone, aby usunąć je później).

Po zakończeniu skanowania restartu status zestawu danych jest automatycznie zmieniany z ODZYSKANE na AKTYWNE, w którym to momencie inne menedżery kolejek mogą ponownie zacząć z niego korzystać.

Informacje o użyciu współużytkowanego zestawu danych komunikatów

Komenda DISPLAY USAGE wyświetla teraz również informacje o wykorzystaniu miejsca we współużytkowanym zestawie danych komunikatów i puli buforów dla wszystkich obecnie otwartych współużytkowanych zestawów danych komunikatów. Ta informacja jest wyświetlana, jeśli podano nową opcję TYPE (SMDS) lub istniejącą opcję TYPE (ALL).

Uwagi dotyczące wydajności i pojemności współużytkowanych danych komunikatów

Monitorowanie użycia zestawu danych

Bieżący procent zapelnienia każdego posiadanego współużytkowanego zestawu danych komunikatów można wyświetlić za pomocą komendy **DISPLAY USAGE** z opcją **TYPE (SMDS)**.

Menedżer kolejek zwykle automatycznie rozszerzy współużytkowany zestaw danych komunikatu, gdy osiągnie 90% zapelnienia, pod warunkiem, że dla definicji SMDS włączona jest opcja **DSEXPAND (YES)**. Ma to zastosowanie, gdy opcja SMDS ma wartość **DSEXPAND (YES)** lub opcja SMDS ma wartość **DSEXPAND (DEFAULT)**, a opcja domyślna CFSTRUCT ma wartość **DSEXPAND (YES)**.

Jeśli próba rozwinięcia nie powiedzie się, ponieważ podczas tworzenia zestawu danych nie określono dodatkowej wielkości przydziału (komunikat IEC070I z kodem przyczyny 203) Menedżer kolejek powtarza żądanie rozszerzenia przy użyciu dodatkowego przydziału przesłonięcia wynoszącego około 20% bieżącej wielkości.

Po rozwinięciu zestawu danych nowe przydziały zestawu danych są formatowane jako część procesu rozszerzania, co może zająć dziesiątki sekund, a nawet minuty w przypadku bardzo dużych przydziałów. Nowy obszar staje się dostępny do użycia po zakończeniu formatowania i zaktualizowaniu katalogu w celu wyświetlenia nowego okresu kontroli wysokiego użycia.

Jeśli nowe komunikaty są tworzone bardzo szybko, istniejący zestaw danych może zostać zapelniony przed zakończeniem przetwarzania rozszerzenia. W takim przypadku każde żądanie, które nie mogło przydzielić obszaru, jest tymczasowo zawieszane do czasu zakończenia próby rozszerzenia i udostępnienia nowego obszaru do użycia. Jeśli rozszerzenie powiodło się, żądanie jest automatycznie ponawiane.

Jeśli próba rozszerzenia nie powiedzie się z powodu braku dostępnego miejsca lub z powodu osiągnięcia maksymalnej liczby przydziałów, zostanie wyświetlony komunikat z informacją o przyczynie niepowodzenia, wówczas opcja nadpisania dla danego zestawu SMDS zostanie automatycznie zmieniona na **DSEXPAND (NO)**, aby zapobiec kolejnym próbom rozszerzenia. W takim przypadku istnieje ryzyko, że zestaw danych zostanie zapelniony. W takim przypadku może być konieczne podjęcie dalszych działań zgodnie z opisem w sekcji Zestaw danych zostanie zapelniony.

Monitorowanie wykorzystania struktury aplikacji

Poziom użycia struktury aplikacji można wyświetlić za pomocą komendy MVS **DISPLAY XCF, STRUCTURE**, określając pełną nazwę struktury aplikacji (w tym przedrostek grupy współużytkownika kolejek). Komunikat odpowiedzi IXC360I przedstawia bieżące użycie elementów i pozycji.

Jeśli użycie struktury przekracza wartość **FULLTHRESHOLD** określoną w strategii CFRM, system wysyła komunikat IXC585E i może wykonać automatyczne **ALTER** działania, jeśli zostały określone, które mogą zmienić współczynnik pozycji do elementu lub zwiększyć wielkość struktury.

Optymalizowanie wielkości puli buforów

Każdy bufor w puli buforów współużytkowanych jest używany do odczytu lub zapisu ciągłego zakresu stron dla jednego komunikatu aż do wielkości bloku logicznego. Jeśli komunikat jest zapisywany w kolejnych blokach, każdy zakres stron w oddzielnym bloku wymaga osobnego buforu.

Bufory zawierające dane komunikatów po operacji zapisu lub odczytu są zachowywane w pamięci masowej i ponownie wykorzystywane przy użyciu schematu pamięci podręcznej LRU (least-recently-used), tak aby żądanie ponownego odczytania tych samych danych wkrótce potem nie było konieczne. Zapewnia to znaczną optymalizację, gdy komunikaty współużytkowane są zapisywane, a następnie odczytywane z powrotem przez aplikacje działające w tym samym systemie. Jeśli komunikaty należące do innego menedżera kolejek są przeglądane w celu dokonania wyboru, a następnie pobierane, pozwala to uniknąć konieczności ponownego odczytania komunikatu z dysku.

Oznacza to, że liczba buforów wymaganych dla każdej struktury aplikacji wynosi jeden dla każdego współbieżnego żądania API, które odczytuje lub zapisuje duże komunikaty dla tej struktury aplikacji oraz pewną liczbę dodatkowych buforów, które zostaną użyte do zapisania ostatnio używanych danych w celu zoptymalizowania kolejnych operacji odczytu.

W przypadku pul buforów współużytkowanych, jeśli liczba buforów jest niewystarczająca, żądania API będą po prostu oczekiwać, jeśli bufor nie jest natychmiast dostępny. Należy jednak unikać tej sytuacji, ponieważ może ona spowodować znaczne obniżenie wydajności.

Statystyki z komendy **DISPLAY USAGE** dla pul buforów współużytkowanych pokazują, czy w bieżącym przedziale czasu statystyk istniały jakiegokolwiek oczekiwania na bufor, a także przedstawiają najniższą liczbę wolnych buforów (lub wartość ujemną wskazującą maksymalną liczbę wątków, które oczekiwały na bufor w dowolnym momencie), liczbę buforów, które zapisały dane, oraz procent czasu, przez jaki żądanie buforu pomyślnie znalazło zapisane dane w łańcuchu LRU ("trafienia LRU") zamiast czytać ("LRU misses")¹.

- Jeśli były jakieś oczekiwania, należy zwiększyć liczbę buforów.
- Jeśli istnieje wiele nieużywanych buforów, liczba buforów może zostać zmniejszona w celu udostępnienia większej ilości pamięci masowej w regionie do innych celów.
- Jeśli istnieje wiele buforów zawierających zapisane dane, ale odsetek odczytów, które były trafieniami w stosunku do tych zapisanych danych, jest bardzo mały, liczba buforów może zostać zmniejszona, jeśli pamięć masowa może być lepiej wykorzystywana do innych celów. Liczba buforów nie powinna być jednak zmniejszana o więcej niż najniższą liczbę wolnych buforów, ponieważ mogłoby to wyzwolić oczekiwania, i powinna być na tyle wysoka, aby najniższa liczba wolnych buforów była zwykle znacznie większa od zera.

Usuwanie współużytkowanych zestawów danych komunikatów

Komenda **DELETE CFSTRUCT** (która jest dozwolona tylko wtedy, gdy wszystkie kolejki współużytkowane w strukturze są puste i zamknięte) nie usuwa samych współużytkowanych zestawów danych komunikatów, ale można je usunąć w zwykły sposób po zakończeniu działania tej komendy. Jeśli ten sam zestaw danych ma być ponownie wykorzystywany jako współużytkowany zestaw danych komunikatu, należy go najpierw ponownie sformatować, aby przywrócić pusty stan.

¹ $(\text{Hits} / (\text{Hits} + \text{Misses})) * 100$

Sytuacje wyjątków dla współużytkowanych zestawów danych komunikatów

Istnieje wiele sytuacji wyjątkowych, które mogą wystąpić podczas normalnego używania, nawet jeśli nie występuje błąd programowy lub sprzętowy.

Zestaw danych staje się pełny

Jeśli zestaw danych zostanie zapełniony, ale nie można go rozwinąć, lub jeśli próba rozwinięcia nie powiedzie się, aplikacje używające odpowiedniego menedżera kolejek do zapisywania dużych komunikatów w odpowiedniej strukturze aplikacji otrzymają błąd 2192, MQRD_STORAGE_MEDIUM_FULL (znany również jako MQRD_PAGESET_FULL).

Zestaw danych może zostać zapełniony z powodu awarii aplikacji, która ma przetwarzać dane, co powoduje gromadzenie się dużej liczby komunikatów. Jeśli tak, dalsze rozszerzanie zestawu danych będzie tylko rozwiązaniem tymczasowym i ważne jest jak najszybsze ponowne działanie aplikacji przetwarzającej.

Jeśli można udostępnić więcej miejsca, można użyć komendy **ALTER SMDS** do ustawienia **DSEXPAND(YES)** lub **DSEXPAND(DEFAULT)** (przy założeniu, że wartość YES została ustawiona lub przyjęta jako wartość domyślna **DSEXPAND** dla definicji CFSTRUCT) w celu wyzwolenia ponowienia. Jeśli jednak przyczyną niepowodzenia było osiągnięcie maksymalnej liczby przydziałów, nowa próba rozwinięcia zostanie odrzucona z komunikatem i ponownie zostanie ustawiona wartość **DSEXPAND(NO)**. W tym przypadku jedynym sposobem na jego dalsze rozszerzenie jest ponowne przydzielenie, co oznacza, że jest on tymczasowo niedostępny, zgodnie z opisem poniżej.

Zestaw danych musi zostać przeniesiony lub ponownie przydzielony

Jeśli zestaw danych wymaga przeniesienia lub rozwinięcia, ale jest używany w inny sposób, może zostać tymczasowo wyjęty z użycia, aby umożliwić jego przeniesienie lub ponowne przydzielenie. Każde żądanie API, które próbuje użyć zestawu danych, gdy jest on niedostępny, otrzyma kod przyczyny MQRD_DATA_SET_NOT_AVAILABLE.

1. Użyj komendy **RESET SMDS**, aby oznaczyć zestaw danych jako **ACCESS(DISABLED)**. Spowoduje to, że zostanie ona normalnie zamknięta i zwolnione przez wszystkie aktualnie podłączone menedżery kolejek.
2. Przenieś lub zmień przydział zestawu danych zgodnie z potrzebami, kopiując starą zawartość do nowo przydzielonego zestawu danych, na przykład za pomocą komendy AMS (Access Method Services) **REPRO**.

Nie należy próbować wstępnie formatować nowego zestawu danych przed skopiowaniem do niego starych danych, ponieważ spowodowałoby to dopisanie skopiowanych danych na końcu sformatowanego zestawu danych.

3. Użyj komendy **RESET SMDS**, aby ponownie oznaczyć zestaw danych jako **ACCESS(ENABLED)**, aby przywrócić go do użycia.

Jeśli stara zawartość jest mniejsza niż wielkość nowego zestawu danych, reszta obszaru zostanie wstępnie sformatowana automatycznie po otwarciu nowego zestawu danych.

Jeśli stara zawartość była większa niż wielkość nowego zestawu danych, menedżer kolejek musi przeskanować komunikaty w strukturze narzędzia CF i odbudować mapę obszaru, aby upewnić się, że żadne aktywne dane nie zostały utracone. Jeśli zostanie znalezione odwołanie do bloku danych, który znajduje się poza nowymi przydziałami, zestaw danych jest oznaczany jako **STATUS(FAILED)** i musi zostać naprawiony przez zastąpienie zestawu danych jedną z poprawnych wielkości i ponowne skopiowanie do niego starego zestawu danych lub użycie funkcji **RECOVER CFSTRUCT** w celu odtworzenia wszystkich trwałych komunikatów.

Struktura narzędzia CF ma mało miejsca

Jeśli w strukturze narzędzia CF kończy się miejsce, co powoduje wyświetlenie komunikatu IXC585E, warto sprawdzić, czy reguły przenoszenia zostały ustawione w celu zapewnienia, że w tym przypadku maksymalna ilość danych jest odciążana. Jeśli nie, reguły przenoszenia mogą być modyfikowane za pomocą komendy **ALTER CFSTRUCT**.

Sytuacje błędów dla współużytkowanych zestawów danych komunikatów

Istnieje wiele problemów, o których należy pamiętać, które mogą być spowodowane tylko przez błędy i nie występują w normalnych sytuacjach operacyjnych.

Nie można otworzyć posiadanego zestawu danych

Jeśli menedżer kolejek, który jest właścicielem współużytkowanego zestawu danych komunikatów, nie może go przydzielić lub otworzyć, albo atrybuty zestawu danych nie są obsługiwane, menedżer kolejek ustawia odpowiednią wartość statusu **SMDSCONN** na **ALLOCFAIL** lub **OPENFAIL** i ustawia dostępność **SMDSCONN** na **AVAIL (ERROR)**. Ustawia również dostępność SMDS na **ACCESS (SUSPENDED)**. Po usunięciu błędu użyj komendy **RESET SMDS**, aby ustawić parametr **ACCESS (ENABLED)** w celu wyzwolenia ponowienia, lub wprowadź komendę **START SMDSCONN** do menedżera kolejek będącego właścicielem.

Nie można otworzyć zestawu danych tylko do odczytu

Jeśli menedżer kolejek nie może przydzielić lub otworzyć współużytkowanego zestawu danych komunikatów, którego właścicielem jest inny menedżer kolejek i który jest oznaczony jako **STATUS (ACTIVE)**, przyjmuje się, że przyczyną jest prawdopodobnie konkretny problem z połączeniem z zestawem danych (reprezentowany przez obiekt **SMDSCONN**), a nie problem z samym zestawem danych.

Oznacza on **SMDSCONN** jako **STATUS (ALLOCFAIL)** lub **STATUS (OPENFAIL)** jako odpowiednie i oznacza dostępność **SMDSCONN** jako **AVAIL (ERROR)**, aby zapobiec dalszym próbom jego użycia.

Jeśli problem można rozwiązać bez wpływu na status samego zestawu danych, należy użyć komendy **START SMDSCONN**, aby wyzwolić ponowienie.

Jeśli okaże się, że problem dotyczy samego zestawu danych, można użyć komendy **RESET SMDS**, aby oznaczyć zestaw danych jako **STATUS (FAILED)**, dopóki nie zostanie on odtworzony. Po odtworzeniu zestawu danych zmiana statusu z powrotem na **STATUS (ACTIVE)** spowoduje powiadomienie innych menedżerów kolejek. Jeśli właściwość **SMDSCONN** jest oznaczona jako **AVAIL (ERROR)**, zostanie ona automatycznie zmieniona z powrotem na **AVAIL (NORMAL)**, aby wyzwolić nową próbę otwarcia zestawu danych.

Nagłówek zestawu danych jest uszkodzony

Jeśli zestaw danych został pomyślnie otwarty, ale format informacji nagłówka jest niepoprawny, menedżer kolejek zamyka i zwalnia zestaw danych i ustawia status na **STATUS (FAILED)**, a dostępność na **ACCESS (SUSPENDED)**. Pozwala to na użycie komendy **RECOVER CFSTRUCT** do odtworzenia zawartości.

Jeśli błąd wystąpił, ponieważ zestaw danych zawierał dane rezydujące z innego użycia i nie został później wstępnie sformatowany, należy wstępnie sformatować zestaw danych i użyć komendy **RESET SMDS**, aby zmienić status na **STATUS (RECOVERED)**.

W przeciwnym razie zestaw danych musi zostać odtworzony.

Zestaw danych jest nieoczekiwanie pusty

Jeśli menedżer kolejek otwiera zestaw danych, który jest oznaczony jako **STATUS (ACTIVE)**, ale stwierdza, że nie został zainicjowany lub jest nowo sformatowany, ale jest poprawny w inny sposób, menedżer kolejek zamyka i zwalnia współużytkowany zestaw danych komunikatów, a następnie ustawia status **STATUS (FAILED)** (Niezainicjowany) i dostępność na wartość **ACCESS (SUSPENDED)** (Dostępność).

Zestaw danych zawiera trwałe błędy we/wy

Jeśli w zestawie danych występują trwałe błędy we/wy po pomyślnym przetworzeniu systemu **OPEN**, prawdopodobnie wymaga to odtworzenia. Menedżer kolejek oznaczy zestaw danych jako **STATUS (FAILED)**, aby wszystkie obecnie połączone menedżery kolejek mogły go zamknąć i zwolnić.

Zestaw danych zawiera naprawialne błędy we/wy

Jeśli występują problemy sprzętowe z zestawem danych, może to spowodować naprawialne błędy we/wy, które nie są odzwierciedlane w menedżerze kolejek, ale powodują znaczne obniżenie wydajności, a także wskazuje na ryzyko wystąpienia trwałych błędów we/wy w niedalekiej przyszłości.

W takim przypadku zestaw danych może zostać wyłaczony w celu odtworzenia za pomocą komendy **RESET SMDS** w celu oznaczenia go jako **STATUS (FAILED)**. Spowoduje to zamknięcie menedżera kolejek i jego ponowne przydzielenie przez wszystkie menedżery kolejek, na przykład można go przenieść do nowego woluminu przed ponownym udostępnieniem.

Jeśli zestaw danych jest w ten sposób niedostępny, odwzorowanie obszaru nie zostanie zapisane, więc przetwarzanie restartu połączenia menedżera kolejek będzie wymagało skanowania struktury narzędzia CF w celu znalezienia komunikatów w zestawie danych i odbudowania odwzorowania obszaru przed ponownym udostępnieniem zestawu danych. Jeśli współużytkowany zestaw danych komunikatu jest nadal użyteczny, można go ustawić jako niedostępny. W tym celu należy użyć komendy **RESET SMDS**, aby oznaczyć zestaw danych **ACCESS (DISABLED)** do czasu, aż będzie gotowy do ponownego udostępnienia.

Zawartość zestawu danych jest niepoprawna

Menedżer kolejek nie może bezpośrednio wykryć, że zestaw danych zawiera niepoprawne dane lub nie jest aktualny, na przykład dlatego, że wolumin zawierający ten zestaw danych musiał zostać odtworzony z kopii zapasowych. Sprawdza jednak integralność, co sprawia, że jest bardzo mało prawdopodobne, aby takie błędy mogły spowodować, że aplikacje będą widzieć niepoprawne dane komunikatów.

Na potrzeby sprawdzania integralności każdy blok komunikatu w zestawie danych jest poprzedzony kopią odpowiedniego identyfikatora pozycji narzędzia CF, w tym unikalnym znacznikiem czasu, który jest sprawdzany przy każdym odczycie bloku komunikatu przed przekazaniem danych komunikatu do programu użytkownika. Jeśli przedrostek bloku komunikatów nie jest zgodny z identyfikatorem pozycji (a pozycja narzędzia CF nie została usunięta w czasie średnim), zakłada się, że blok komunikatów jest uszkodzony i nie nadaje się do użycia.

Jeśli uszkodzony komunikat był trwały, zestaw danych jest oznaczany jako **STATUS (FAILED)**, a zawartość struktury musi zostać odzyskana za pomocą komendy **RECOVER CFSTRUCT**. Jeśli uszkodzony komunikat był nietrwały, nie ma możliwości jego odzyskania, dlatego wysyłany jest komunikat diagnostyczny i usuwana jest odpowiednia pozycja komunikatu narzędzia CF.

Jeśli podczas otwierania zestawu danych nie jest dostępne żadne zapisane odwzorowanie obszaru, jest ono odbudowywane przez skanowanie struktury narzędzia CF w poszukiwaniu odwołań do danych w zestawie danych. Podczas tego skanowania menedżer kolejek wykonuje kilka działań:

1. Menedżer kolejek określa położenie najnowszego komunikatu (jeśli istnieje), który aktualnie pozostaje w zestawie danych.
2. Następnie menedżer kolejek odczytuje ten komunikat z zestawu danych, aby upewnić się, że przedrostek bloku jest zgodny z identyfikatorem pozycji komunikatu.

Te działania gwarantują, że menedżer kolejek wykryje wszystkie przypadki, w których poziom zestawu danych jest niższy, i oznaczy zestaw danych jako NIEPOWODZENIE. To sprawdzenie jest jednak tolerowane w przypadku, gdy zestaw danych został odtworzony z poprzedniej kopii i od tego czasu nie dodano żadnych nowych wiadomości lub wszystkie wiadomości dodane od tej kopii zostały później odczytane i usunięte.

Aby zabezpieczyć się przed danymi niższego poziomu w przypadku normalnego zamknięcia zestawu danych, menedżer kolejek wykonuje kilka działań:

1. Menedżer kolejek zapisuje kopię znacznika czasu mapy obszaru w obiekcie SMDS w programie Db2, gdy zestaw danych jest normalnie zamknięty.
2. Następnie menedżer kolejek sprawdza, czy znacznik czasu mapy obszaru jest taki sam, gdy zestaw danych jest ponownie otwierany.

Jeśli znacznik czasu nie jest zgodny, oznacza to, że mogła zostać użyta kopia zestawu danych o niższym poziomie, dlatego menedżer kolejek ignoruje istniejące odwzorowanie obszaru i odbudowuje je, co powiedzie się tylko wtedy, gdy nie utracono żadnych danych komunikatu.

Uwaga: Te sprawdzenia integralności nie gwarantują wykrycia uszkodzonego lub niekompletnego zestawu danych we wszystkich teoretycznie możliwych przypadkach. Na przykład nie wykryją przypadku, w którym początek bloku komunikatów jest poprawny, ale reszta danych została częściowo nadpisana.

Scenariusze odtwarzania dla współużytkowanych zestawów danych komunikatów

W tej sekcji opisano scenariusze odtwarzania współużytkowanego zestawu danych komunikatów.

Odtwarzanie zestawu danych w przypadku, gdy nie utracono żadnych danych

W niektórych przypadkach poprawna zawartość zestawu danych, który uległ awarii, może zostać odtworzona bez konieczności rzeczywistego odtwarzania. Przykładem może być sytuacja, w której zbiór danych zawiera dane rezydualne z poprzedniego użycia i nie został ponownie wstępnie sformatowany, co można naprawić przez wstępne formatowanie. Innym przypadkiem jest przeniesienie zestawu danych, ale wystąpił błąd w procesie kopiowania danych, który można naprawić, kopiując dane ponownie poprawnie.

W takich przypadkach poprawiony zestaw danych można ponownie udostępnić za pomocą komendy **RESET SMDS** w celu ustawienia wartości **STATUS (RECOVERED)**. Jeśli aktualnie dostępna jest opcja **ACCESS (SUSPENDED)**, zostanie ona automatycznie ustawiona z powrotem na wartość **ACCESS (ENABLED)**.

Gdy menedżer kolejek będący właścicielem zostanie powiadomiony o odzyskaniu zestawu danych, skanuje zawartość struktury, aby zrekonstruować mapę obszaru, a następnie zmienia status na **STATUS (ACTIVE)**. Inne menedżery kolejek mogą następnie ponownie rozpocząć odczytywanie zestawu danych.

Odzyskiwanie zestawu danych z TYPE (NORMAL)

Jeśli treść zestawu danych została utracona, ale struktura aplikacji została zdefiniowana z produktem **RECOVER (YES)** i dostępne są odpowiednie dzienniki odtwarzania, można użyć komendy **RECOVER CFSTRUCT** do odtworzenia wszystkich trwałych komunikatów zapisanych w strukturze, w tym danych komunikatów trwałych, które zostały załadowane do współużytkowanych zestawów danych komunikatów. Ta komenda odtwarza bieżący stan przy użyciu informacji rejestrowanych przez komendę **BACKUP CFSTRUCT** oraz wszystkich zarejestrowanych zmian w komunikatach trwałych od czasu utworzenia kopii zapasowej.

Komenda **RECOVER CFSTRUCT** zawsze odtwarza wszystkie trwałe komunikaty w strukturze narzędzia CF wraz z odciążonymi danymi komunikatów zapisanymi w pliku Db2. W przypadku odciążonych danych przechowywanych we współużytkowanych zestawach danych komunikatów każdy zestaw danych jest wybierany do przetwarzania odtwarzania tylko wtedy, gdy jest już oznaczony jako **STATUS (FAILED)** lub gdy zostanie nieoczekiwanie pusty albo niepoprawny, gdy zostanie otwarty podczas przetwarzania odtwarzania. Każdy współużytkowany zestaw danych komunikatu, który jest oznaczony jako aktywny i który spełnia kryteria sprawdzania poprawności, nie musi być odtwarzany, ponieważ istniejące dane komunikatu są już poprawne, ale nagłówek jest aktualizowany w celu wskazania, że wszystkie zapisane odwzorowania obszaru będą musiały zostać odbudowane po odtworzeniu.

Przetwarzanie odtwarzania jest możliwe tylko wtedy, gdy struktura została oznaczona jako zakończona niepowodzeniem, ponieważ cała zawartość struktury musi zostać zrekonstruowana przez przetwarzanie odtwarzania. Jeśli jednak co najmniej jeden współużytkowany zestaw danych komunikatu został oznaczony jako zakończony niepowodzeniem, komenda **RECOVER CFSTRUCT** automatycznie oznaczy strukturę jako zakończoną niepowodzeniem, jeśli jest to konieczne, aby umożliwić kontynuowanie przetwarzania odtwarzania.

Odtwarzanie można wykonać z dowolnego menedżera kolejek w grupie współużytkowania kolejek, pod warunkiem, że ma on prawo do zapisu w odpowiednich zestawach danych.

Składowane i rejestrowane są tylko komunikaty trwałe, dlatego podczas normalnego przetwarzania odtwarzania zostaną odtworzone wszystkie komunikaty trwałe, ale wszystkie komunikaty nietrwałe w strukturze zostaną utracone.

Po zakończeniu odtwarzania każdy zestaw danych, który został wybrany do odtwarzania, jest automatycznie zmieniany na **STATUS (RECOVERED)**, a jeśli dostępność była **ACCESS (SUSPENDED)**, jest zmieniany na **ACCESS (ENABLED)**. Menedżer kolejek odbudowuje odwzorowanie obszaru dla każdego zestawu danych, skanując komunikaty w narzędziu CF, a następnie oznacza zestaw danych jako **STATUS (ACTIVE)**, aby można go było użyć ponownie.

Odzyskiwanie zestawu danych z TYPE (PURGE)

W przypadku struktury odtwarzalnej, jeśli treść zestawu danych została utracona, ale z jakiegoś powodu odtwarzanie nie jest możliwe, na przykład z powodu braku dostępnych dzienników odtwarzania lub zbyt długiego czasu odtwarzania, można użyć komendy **RECOVER CFSTRUCT** z komendą **TYPE (PURGE)**, aby przywrócić strukturę do stanu używalnego. Spowoduje to przywrócenie struktury do stanu pustego i oznaczenie wszystkich powiązanych zestawów danych jako **STATUS (EMPTY)**.

Usuwanie struktury aplikacji

Jeśli nienaprawialna struktura aplikacji zostanie usunięta za pomocą komendy MVS **SETXCF FORCE** lub w wyniku awarii struktury, przy następnym połączeniu struktury zostanie wyświetlony komunikat CSQE028I informujący, że struktura została zresetowana, a wszystkie istniejące komunikaty zostały usunięte, a wszystkie istniejące zestawy danych również zostaną automatycznie zresetowane do wartości **STATUS (EMPTY)**. Ta czynność powoduje, że struktura nienaprawialna staje się ponownie użyteczna po utracie danych w strukturze lub w dowolnym z powiązanych zestawów danych.

Jeśli odtwarzalna struktura aplikacji zostanie usunięta, będzie traktowana w taki sam sposób, jak w przypadku niepowodzenia struktury.

Odtwarzanie zestawu danych nie powiodło się

Jeśli z jakiegoś powodu program **RECOVER CFSTRUCT** nie może zakończyć działania, na przykład dlatego, że zestaw danych dziennika nie jest już dostępny lub ponieważ menedżer kolejek został przerwany w trakcie odtwarzania, wszystkie zestawy danych, dla których odtwarzanie zostało przynajmniej uruchomione, zostaną oznaczone w nagłówku, aby pokazać, że podjęto próbę częściowego odtwarzania, a zestaw danych pozostanie w stanie **STATUS (FAILED)**.

W takim przypadku można powtórzyć oryginalne żądanie odtwarzania lub odzyskać dane za pomocą komendy **TYPE (PURGE)**, usuwając istniejące dane.

Jeśli zostanie podjęta próba oznaczenia zestawu danych jako **STATUS (RECOVERED)** bez rzeczywistego odtwarzania, przy następnym otwarciu menedżera kolejek menedżer kolejek zobaczy, że nagłówek wskazuje na niepełne odtwarzanie, a następnie ponownie oznaczy go jako **STATUS (FAILED)**.

Odtwarzanie po awarii poza siedzibą przedsiębiorstwa

W przypadku odtwarzania po awarii poza siedzibą przedsiębiorstwa trwałe komunikaty współużytkowane mogą być ponownie tworzone tylko przy użyciu dzienników i obiektów współużytkowanych Db2 zawierających definicje CFSTRUCT i powiązane informacje o statusie SMDS.

Po skonfigurowaniu tabel Db2 zawierających definicje struktury aplikacji i współużytkowane zestawy danych komunikatów mogą być ustawione jako puste. Gdy menedżer kolejek połączy się z nimi i stwierdzi, że są one nieoczekiwanie puste, oznaczy je jako zakończone niepowodzeniem, po czym będzie można użyć jednej komendy **RECOVER CFSTRUCT** do odtworzenia wszystkich trwałych komunikatów dla wszystkich struktur, których to dotyczy.

Komendy związane z SMDS

W tym temacie opisano i zapewnia dostęp do komend związanych ze współużytkowanymi zestawami danych komunikatów.

Wyświetl i zmień opcje **CFSTRUCT** odnoszące się do przenoszenia dużych komunikatów (**OFFLOAD** i reguł przenoszenia) oraz zestawów danych komunikatów współużytkowanych (**DSGROUP**, **DSBLOCK**, **DSBUFS**, **DSEXPAND**):

- [WYŚWIETL CFSTRUCT](#)
- [DEFINE CFSTRUCT](#)
- [ALTER CFSTRUCT](#)
- [USUŃ CFSTRUCT](#)

Wyświetl status **CFSTRUCT** odnoszący się do dużego rozładowania komunikatów (**OFFLDUSE**):

- [WYŚWIETL STATUS CFSTATUS](#)

Wyświetlanie i modyfikowanie opcji nadpisywania zestawu danych (**DSEXPAND** i **DSBUFS**) dla poszczególnych menedżerów kolejek:

- [Wyświetlanie zestawu SMDS](#)
- [ALTER SMDS](#)

Wyświetlanie lub modyfikowanie statusu i dostępności zestawów danych w grupie współużytkowania kolejek:

- [WYŚWIETLANIE TYPU CFSTATUS \(SMDS\)](#)
- [RESETOWANIE SMDS](#)

Wyświetl informacje o wykorzystaniu miejsca w zestawie danych SMDS i informacje o wykorzystaniu buforu dla menedżera kolejek:

- [WYŚWIETLANIE TYPU WYKORZYSTANIA \(SMDS\)](#)

Wyświetlanie lub modyfikowanie statusu i dostępności połączeń (**SMDSCONN**) do zestawów danych z pojedynczego menedżera kolejek:

- [WYŚWIETL SMDSCONN](#)
- [URUCHOM SMDSCONN](#)
- [ZATRZYMAJ SMDSCONN](#)

Tworzenie i odtwarzanie kopii zapasowych współużytkowanych komunikatów, w tym dużych danych komunikatów w SMDS, gdy jest to konieczne:

- [KOPIA ZAPASOWA CFSTRUCT](#)
- [ODZYSKIWANIE CFSTRUCT](#)

Zalety korzystania z kolejek współużytkowanych

Kolejka współużytkowana pozwala na skalowalność, wysoką dostępność aplikacji IBM MQ i umożliwia zaimplementowanie równoważenia obciążenia.

Zalety kolejek współużytkowanych

Architektura współużytkowanej kolejki, w której sklonowane serwery pobierają pracę z jednej współużytkowanej kolejki, ma kilka użytecznych właściwości:

- Jest to skalowalne, dodając nowe instancje aplikacji serwera, a nawet dodając nowy obraz z/OS z menedżerem kolejek (w grupie współużytkowania kolejek) i kopią aplikacji.
- Jest on bardzo dostępny.
- Naturalnie wykonuje *ciągnać* równoważenie obciążenia w oparciu o dostępną moc obliczeniową każdego menedżera kolejek w grupie współużytkowania kolejek.

Korzystanie z kolejek współużytkowanych w celu zapewnienia wysokiej dostępności

Poniższe przykłady ilustrują sposób korzystania z kolejki współużytkowanej w celu zwiększenia dostępności aplikacji.

Należy wziąć pod uwagę scenariusz IBM MQ, w którym aplikacje klienckie działające w sieci chcą składać żądania aplikacji serwera działających w systemie z/OS. Aplikacja kliencka konstruuje komunikat żądania i umieszcza go w kolejce żądań. Następnie klient czeka na odpowiedź z serwera, wysłaną do kolejki zwrotnej o nazwie w deskrypcji komunikatu żądania.

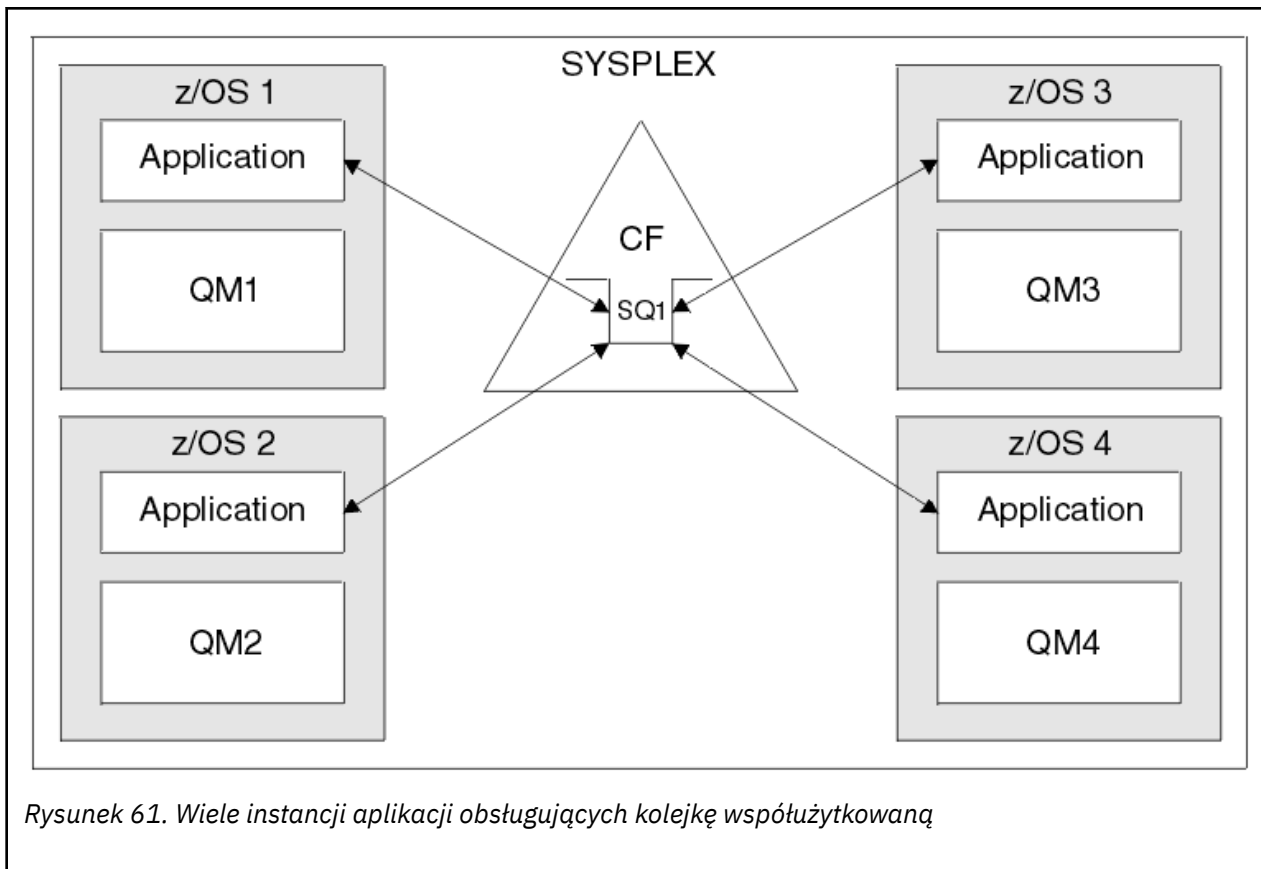
Produkt IBM MQ zarządza transportem komunikatu żądania z komputera klienckiego do kolejki wejściowej serwera w systemie z/OS oraz z odpowiedzi serwera z powrotem do klienta. Definiując kolejkę wejściową serwera jako kolejkę współużytkowaną, wszystkie komunikaty umieszczane w kolejce mogą być pobierane w dowolnym menedżerze kolejek w grupie współużytkowania kolejek. Oznacza to, że menedżer kolejek można skonfigurować na każdym obrazie produktu z/OS w syspleksie, a następnie łącząc je wszystkie z tą samą grupą współużytkowania kolejek, każdy z nich może uzyskać dostęp do komunikatów w kolejce wejściowej serwera.

Komunikaty w kolejce wejściowej serwera są nadal dostępne, nawet jeśli jeden z menedżerów kolejek kończy działanie w sposób nieprawidłowy lub użytkownik musi go zatrzymać z przyczyn administracyjnych. Cały obraz produktu z/OS można przetaczać w tryb bez połączenia, a komunikaty nadal będą dostępne.

Aby skorzystać z tej dostępności komunikatów w kolejce współużytkowanej, należy uruchomić instancję aplikacji serwera na każdym obrazie produktu z/OS w syspleksie, aby zapewnić większą pojemność i dostępność aplikacji serwera, jak to pokazano w sekcji [Rysunek 61](#) na stronie 195.

Jedna instancja aplikacji serwera pobiera komunikat żądania z kolejki współużytkowanej i w oparciu o treść wykonuje przetwarzanie, tworząc wynik, który jest wysyłany z powrotem do klienta jako komunikat IBM MQ. Komunikat odpowiedzi jest przeznaczony dla menedżera kolejek zwrotnych odpowiedzi i odpowiedzi na menedżera kolejek o nazwie w deskrypcji komunikatu żądania.

Istnieje wiele opcji, których można użyć do skonfigurowania ścieżki powrotnej. Więcej informacji o tych opcjach zawiera sekcja [“Rozproszone grupy kolejkowania i współużytkowania kolejek”](#) na stronie 215.



Odtwarzanie równorzędne

Aby jeszcze bardziej zwiększyć dostępność komunikatów w grupie współużytkowania kolejek, program IBM MQ wykrywa, czy inny menedżer kolejek w grupie rozłącza się z obiektem sprzęgającym w sposób nieprawidłowy i wykonuje jednostki pracy dla tego menedżera kolejek, które nadal oczekują, o ile to możliwe. Ta funkcja jest znana jako *odtworzenie równorzędne*.

Przypuśćmy, że menedżer kolejek kończy działanie w sposób nieprawidłowy w punkcie, w którym aplikacja wczytała komunikat z kolejkę w punkcie synchronizacji, ale nie umiała jeszcze umieścić komunikatu odpowiedzi lub zatwierdzić jednostkę pracy. Inny menedżer kolejek w grupie współużytkowania kolejek wykrywa awarię i tworzy kopie zapasowe jednostek pracy wykonywanych w menedżerze kolejek, które się nie powiodły. Oznacza to, że komunikat zapytania jest umieszczany z powrotem w kolejkę zapytań i jest dostępny dla jednej z pozostałych instancji serwera do przetworzenia bez oczekiwania na zrestartowanie menedżera kolejek, który uległ awarii.

Jeśli produkt IBM MQ nie może automatycznie rozstrzygnąć jednostki pracy, można ręcznie rozstrzygnąć współużytkowaną część, aby umożliwić innemu menedżerowi kolejek w grupie współużytkowania kolejek kontynuację przetwarzania tej pracy.

► z/OS **Użycie pamięci klasy pamięci masowej z kolejkami współużytkowanymi**

Użycie pamięci klasy pamięci masowej (SCM) może być korzystne w przypadku używania z kolejkami współużytkowanymi IBM MQ for z/OS .

Ważne: Planowane jest, że IBM z16 będzie ostatnią generacją IBM Z[®] obsługującej użycie wirtualnej pamięci Flash (znanej również jako pamięć klasy pamięci masowej lub SCM) dla obrazów narzędzia CF. Więcej informacji na ten temat zawierają: [IBM Z i IBM LinuxONE 4Q 2023 Oświadczenia o kierunku](#).

Alternatywnie należy użyć większych struktur lub przenieść komunikaty do SMDS.

Komputery z13, zEC12i zBC12 umożliwiają instalowanie kart Flash Express. Karty te zawierają dyski SSD. Po instalacji pamięć masowa flash z kart może zostać przydzielona do jednej lub kilku partycji LPAR, gdzie jest zwykle znana jako SCM.

System SCM znajduje się między rzeczywistą pamięcią masową a urządzeniem pamięci masowej o dostępie bezpośrednim (DASD) zarówno pod względem opóźnienia we/wy, jak i kosztu. Ponieważ system SCM nie ma ruchomych części, charakteryzuje się znacznie mniejszymi opóźnieniami we/wy niż DASD.

SCM jest również znacznie tańsze niż prawdziwej pamięci masowej. W związku z tym można zainstalować dużą ilość pamięci masowej przy stosunkowo niskich kosztach, na przykład para kart Flash Express zawiera 1424 GB użytecznej pamięci masowej.

Te cechy oznaczają, że SCM jest przydatne, gdy duża ilość danych musi być pobrana z rzeczywistej pamięci masowej w krótkim okresie czasu, ponieważ dane te mogą być zapisane w systemie SCM znacznie szybciej niż mogą być zapisane w DASD. Ten konkretny punkt może być bardzo przydatny podczas korzystania ze struktur listy narzędzia CF zawierających kolejki współużytkowane w systemie IBM MQ .

Dlaczego struktury list wypełniają się

Po zdefiniowaniu struktury CF jest ona konfigurowana z atrybutem SIZE opisującym maksymalną wielkość struktury. Ponieważ struktury CF są zawsze trwale rezydentne w pamięci rzeczywistej, suma atrybutów SIZE struktur zdefiniowanych w systemie CF powinna być mniejsza niż ilość pamięci rzeczywistej przydzielonej do systemu CF.

W związku z tym istnieje stałe ciśnienie, które pozwala zachować wartość SIZE dowolnej struktury do wartości minimalnej, tak aby większa liczba struktur mogła zmieścić się w systemie CF. Jednak zapewnienie, że struktury są wystarczająco duże, aby osiągnąć ich cel, może spowodować powstanie presji powodującej konflikt, ponieważ zbyt mała struktura oznacza, że może się wypełnić, zakłócając aplikacje lub podsystemy korzystające z niej.

Istnieje silna potrzeba dokładnego określenia wielkości struktury na podstawie jej oczekiwanego użycia. Zadanie to jest jednak trudne do wykonania, ponieważ obciążenia mogą się zmieniać w czasie, a rozliczanie się z ich fluktuacji nie jest łatwe.

Kolejki współużytkowane systemu IBM MQ używają struktur list systemu CF do przechowywania komunikatów. Produkt IBM MQ wywołuje struktury CF, które zawierają komunikaty i struktury aplikacji.

Struktury aplikacji są przywoływane przy użyciu informacji przechowywanych w obiektach IBM MQ CFSTRUCT. Jeśli komunikat o wielkości mniejszej niż 63 kB jest umieszczany w kolejce współużytkowanej, komunikat jest przechowywany w całości w strukturze aplikacji jako pojedyncza pozycja listy i zero lub więcej elementów listy.

Ponieważ kolejki współużytkowane w systemie IBM MQ używają struktur list, opisane ciśnienia mają również wpływ na kolejki współużytkowane. W takim przypadku maksymalna liczba komunikatów, które mogą być przechowywane w kolejce współużytkowanej, jest funkcją:

- Wielkość komunikatów w kolejce
- Maksymalna wielkość struktury
- Liczba wpisów i elementów dostępnych w strukturze

Ponieważ do 512 współużytkowanych kolejek może korzystać z tej samej struktury i skutecznie konkurować o pozycje i elementy, to jeszcze bardziej komplikuje sprawę.

Kolejki produktu IBM MQ są używane do przesyłania danych między aplikacjami, dlatego typową sytuacją jest aplikacja umieszczająca komunikaty w kolejce, gdy aplikacja partnerska, która powinna otrzymać te komunikaty, nie jest uruchomiona.

W takim przypadku liczba komunikatów w kolejce zwiększa się z upływem czasu, aż wystąpi co najmniej jedna z następujących sytuacji:

- Aplikacja umieszczania zatrzymuje umieszczanie komunikatów.

- Aplikacja pobierająca rozpoczyna pobieranie komunikatów.
- Istniejące komunikaty w kolejce tracą ważność i są usuwane z kolejki.
- Kolejka osiąga maksymalne zapętnienie, w którym to przypadku kod przyczyny MQRC_Q_FULL jest zwracany do aplikacji umieszczającej.
- Struktura zawierająca kolejkę współużytkowaną osiąga maksymalną wielkość lub w systemie CF zawierającym strukturę zabraknie dostępnej pamięci. W obu przypadkach do aplikacji umieszczającej zwracany jest kod przyczyny MQRC_STORAGE_MEDIUM_FULL.

W ostatnich trzech przypadkach kolejka jest pełna. W tym momencie aplikacja odkładania ma problem, ponieważ nie ma dokąd pójść jej komunikat. Aplikacja aplikująca zwykle rozwiązuje ten problem, korzystając z jednego lub kilku z następujących rozwiązań:

- Powtarzające się próby umieszczenia komunikatu, opcjonalnie z opóźnieniem między ponownymi próbami.
- Umieścić komunikaty w innym miejscu, na przykład w bazie danych lub w pliku. Dostęp do komunikatów można uzyskać później i umieścić je w kolejce w normalny sposób.
- Usunąć komunikat, jeśli jest nietrwały.

Jednak w przypadku niektórych klas aplikacji, na przykład z dużą liczbą komunikatów przychodzących lub bez dostępu do systemu plików, rozwiązania te nie są praktyczne. Istnieje realna potrzeba zapewnienia, że kolejki nigdy nie zostaną zapętnione lub jest mało prawdopodobne, aby zostały zapętnione w pierwszej kolejności, co jest szczególnie istotne dla kolejek współużytkowanych.

Reguły SMDS i odciążania

Reguły przenoszenia wprowadzone w produkcie IBM WebSphere MQ 7.1 umożliwiają zmniejszenie prawdopodobieństwa zapętnienia struktury aplikacji.

Z każdą strukturą aplikacji są powiązane trzy reguły, określone za pomocą trzech par słów kluczowych:

- OFFLD1SZ i OFFLD1TH
- OFFLD2SZ i OFFLD2TH
- OFFLD3SZ i OFFLD3TH

Każda reguła określa warunki, które muszą być spełnione, aby dane komunikatu były przenoszone do mechanizmu pamięci masowej powiązanego ze strukturą aplikacji. Obecnie dostępne są dwa typy mechanizmów pamięci masowej:

- Db2
- Grupa liniowych zestawów danych VSAM (Virtual Storage Access Method), które IBM MQ wywołuje współużytkowany zestaw danych komunikatów (SMDS).

Poniższy przykład przedstawia komendę MQSC służącą do tworzenia struktury aplikacji o nazwie LIST1 przy użyciu komendy `DEFINE CFSTRUCT`.

Ta struktura zawiera domyślne reguły przenoszenia i używa SMDS jako mechanizmu przenoszenia. Oznacza to, że jeśli struktura jest zapętniona w 70% (OFFLD1TH), wszystkie komunikaty o wielkości 32 kB lub większe (OFFLD1SZ) są przenoszone do SMDS.


Podobnie, jeśli struktura jest zapętniona w 80% (OFFLD2TH), wszystkie komunikaty o wielkości 4 kB lub większe (OFFLD2SZ) są przenoszone. Gdy struktura jest zapętniona w 90% (OFFLD3TH), wszystkie komunikaty (OFFLD3SZ) są przenoszone.

```
DEFINE CFSTRUCT(LIST1)
CFLEVEL(5)
OFFLOAD(SMDS)
OFFLD1SZ(32K) OFFLD1TH(70)
OFFLD2SZ(4K) OFFLD2TH(80)
OFFLD3SZ(0K) OFFLD3TH(90)
```

Odciążony komunikat jest przechowywany na nośniku odciażającym, a wskaźnik do komunikatu jest przechowywany w strukturze. Podczas gdy reguły przenoszenia zmniejszają prawdopodobieństwo zapelnienia struktury, przez umieszczenie mniejszej ilości danych komunikatu w strukturze po wyczerpaniu pamięci, niektóre dane są nadal zapisywane w strukturze dla każdego komunikatu. Oznacza to, że wskaźnik do odciażonego komunikatu.

Ponadto reguły odciażania mają wpływ na wydajność. Zapisywanie komunikatu do struktury jest stosunkowo szybkie i w dużym stopniu zdominowane przez czas potrzebny na wysłanie żądania zapisu do systemu CF. Faktyczny zapis do struktury jest szybki, dzieje się to z prawdziwą szybkością pamięci masowej.

Zapisywanie komunikatu do SMDS jest znacznie wolniejsze, ponieważ obejmuje zapis do struktury dla wskaźnika komunikatu i zapis danych komunikatu do SMDS. Ta druga operacja zapisu jest wykonywana z szybkością DASD i może spowodować dodatkowe opóźnienie. Jeśli jako mechanizm przenoszenia używany jest parametr Db2, koszt wydajności jest znacznie większy.

 *Sposób działania pamięci klasy pamięci masowej z produktem IBM MQ for z/OS*

Przegląd użycia pamięci klasy pamięci masowej (SCM) z kolejkami współużytkowanymi produktu IBM MQ for z/OS.

Ważne: Planowane jest, że IBM z16 będzie ostatnią generacją IBM Z[®] obsługującej użycie wirtualnej pamięci Flash (znanej również jako pamięć klasy pamięci masowej lub SCM) dla obrazów narzędzia CF. Więcej informacji na ten temat zawierają: [IBM Z i IBM LinuxONE 4Q 2023 Oświadczenia o kierunku](#).

Alternatywnie należy użyć większych struktur lub przenieść komunikaty do SMDS.

Do narzędzia CF, które jest na poziomie CFLEVEL 19 lub wyższym, może być przydzielony system SCM. Struktury zdefiniowane w tym systemie CF można następnie skonfigurować w taki sposób, aby korzystały z systemu SCM w celu zmniejszenia ryzyka zapelnienia struktur (znanego jako pełny warunek struktury). Gdy struktura skonfigurowana do korzystania z systemu SCM zapelnia się po punkcie określonym przez system, system CF rozpoczyna przenoszenie danych ze struktury do systemu SCM, co zwalnia miejsce w strukturze dla nowych danych.

Uwaga: Ponieważ sam komponent SCM może zostać zapelniony, przydzielenie komponentu SCM do struktury zmniejsza tylko prawdopodobieństwo spełnienia pełnego warunku struktury, ale nie całkowicie usuwa możliwości wystąpienia takiego stanu.

Struktura jest skonfigurowana do korzystania z systemu SCM przez określenie zarówno słów kluczowych **SCMAlgorithm**, jak i **SCMMaxSize** w strategii menedżera zasobów narzędzia CF (Coupling Facility Resource Manager-CFRM) zawierającej definicję tej struktury.

Należy zauważyć, że po podaniu tych słów kluczowych i zastosowaniu strategii CFRM struktura musi zostać odbudowana lub zwolona, aby mogły one zostać zastosowane.

Słowo kluczowe SCMAlgorithm

Ponieważ szybkość wejścia/wyjścia systemu SCM jest wolniejsza niż rzeczywista pamięć masowa, system CF używa algorytmu dostosowanego do oczekiwanego użycia struktury w celu zmniejszenia wpływu zapisu lub odczytu z systemu SCM.

Algorytm jest konfigurowany za pomocą słowa kluczowego **SCMAlgorithm** w strategii CFRM dla struktury przy użyciu wartości **KEYPRIORITY1**. Należy pamiętać, że wartości **KEYPRIORITY1** należy używać tylko ze strukturami listy używanymi przez kolejki współużytkowane systemu IBM MQ.

Algorytm **KEYPRIORITY1** działa, zakładając, że większość aplikacji będzie pobierała komunikaty z kolejki współużytkowanej w kolejności priorytetów. Oznacza to, że gdy aplikacja pobiera komunikat, otrzymuje on najstarszy komunikat o najwyższym priorytecie.

Gdy struktura zaczyna zapelniać się po przekroczeniu progu zdefiniowanego przez system (90%), system CF rozpoczyna asynchronicznie migrowanie komunikatów, które są najmniej prawdopodobne, że będą następne. Są to komunikaty o niższych priorytetach, które zostały ostatnio umieszczone w kolejce.

Ta asynchroniczna migracja komunikatów ze struktury do systemu SCM jest nazywana migracją wstępną.

Wstępne przemieszczanie zmniejsza koszty wydajności związane z używaniem systemu SCM, ponieważ zmniejsza ryzyko zablokowania aplikacji podczas występowania synchronicznego wejścia/wyjścia do systemu SCM.

Oprócz wstępnego przemieszczania algorytm *KEYPRIORITY1* także asynchronicznie przywołuje komunikaty z systemu SCM i do struktury, gdy dostępna jest wystarczająca ilość wolnego miejsca. Dla algorytmu *KEYPRIORITY1* oznacza to, że gdy struktura jest mniejsza lub równa 70% pełnej.

Działanie pobierania komunikatów z systemu SCM do struktury jest określane jako "preselekcja".

Preselekcja zmniejsza prawdopodobieństwo, że aplikacja próbuje pobrać komunikat, który został wstępnie przemieszczony do systemu SCM, i musi czekać, aż system CF synchronicznie przywróci komunikat do struktury.

Słowo kluczowe SCMMAXSIZE

Słowo kluczowe **SCMMAXSIZE** definiuje maksymalną ilość SCM, która może być używana przez strukturę. Ponieważ system SCM jest przydzielany do struktury przez system CF, gdy jest to wymagane, można określić parametr **SCMMAXSIZE**, który jest większy niż łączna ilość dostępnego wolnego systemu SCM. Jest to nazywane "nadmiernym zatwierdzeniem".

Ważne: Nigdy nie należy nadzatwierdzać SCM. W takim przypadku aplikacje, które na nim polegają, nie uzyskają oczekiwanego zachowania. Na przykład aplikacje IBM MQ korzystające z kolejek współużytkowanych mogą otrzymać nieoczekiwane kody przyczyny MQRCS_STORAGE_MEDIUM_FULL.

System CF używa różnych struktur danych do śledzenia użycia systemu SCM. Te struktury danych rezydują w pamięci rzeczywistej, która jest przydzielona do systemu CF i w rezultacie zmniejszają ilość pamięci rzeczywistej, która może być używana przez struktury. Pamięć masowa używana przez te struktury danych jest nazywana "przestrzenią rozszerzoną".

Jeśli struktura jest skonfigurowana z systemem SCM, niewielka ilość rzeczywistej pamięci masowej jest przydzielana z systemu CF do struktury zwanej stałym obszarem rozszerzonym. Jest ona przydzielana nawet wtedy, gdy struktura nigdy nie używa żadnego systemu SCM. Ponieważ dane ze struktury są przechowywane w systemie SCM, dodatkowy dynamiczny obszar rozszerzony zostanie przydzielony z zapasowej rzeczywistej pamięci masowej w systemie CF.

Po usunięciu danych z systemu SCM dynamiczny obszar rozszerzony jest zwracany do systemu CF. Obszar rozszerzony, stały lub dynamiczny, nigdy nie jest pobierany z rzeczywistej pamięci masowej, która jest przydzielona do struktury.

Oprócz rozszerzonej pamięci masowej, gdy struktura jest skonfigurowana do korzystania z systemu SCM, zwiększa się ilość pamięci masowej używanej przez tę strukturę. Oznacza to, że struktura listy skonfigurowana z użyciem systemu SCM może zawierać mniej pozycji i elementów niż struktura o tej samej wielkości bez konfigurowania systemu SCM.

Aby zrozumieć wpływ systemu SCM na nowe lub istniejące struktury, należy użyć narzędzia [CFSizer](#).

Należy zauważyć, że po przeniesieniu danych ze struktury do systemu SCM i użyciu dynamicznego, rozszerzonego obszaru, struktura nie może być zmieniana ani ręcznie, ani automatycznie.

Oznacza to, że nie można zwiększyć ani zmniejszyć ilości pamięci masowej przydzielonej dla struktury, nie można zmienić współczynnika pozycji do elementu, który jest używany przez strukturę itd. Aby można było ponownie zmienić strukturę, struktura nie może mieć żadnych danych przechowywanych w systemie SCM i nie może korzystać z dynamicznej rozszerzonej pamięci masowej.

Dlaczego należy używać systemu SCM

Awaryjna pamięć masowa i zwiększona wydajność to dwa przypadki użycia systemu SCM z produktem IBM MQ for z/OS.

Ważne: Planowane jest, że IBM z16 będzie ostatnią generacją IBM Z[®] obsługującej użycie wirtualnej pamięci Flash (znanej również jako pamięć klasy pamięci masowej lub SCM) dla obrazów narzędzia CF. Więcej informacji na ten temat zawierają: [IBM Z i IBM LinuxONE 4Q 2023 Oświadczenia o kierunku](#).

Alternatywnie należy użyć większych struktur lub przenieść komunikaty do SMDS.

W tej sekcji przedstawiono teorię leżącą u podstaw dwóch możliwych scenariuszy. Więcej informacji na temat konfigurowania scenariuszy można znaleźć w następujących sekcjach:

- [“Pamięć awaryjna-podstawowa konfiguracja” na stronie 203](#)
- [“Poprawiona wydajność-podstawowa konfiguracja” na stronie 209](#)

Ważne: Użycie systemu SCM ze strukturami narzędzia CF nie jest zależne od żadnej konkretnej wersji produktu IBM MQ. Jednak scenariusz awaryjnej pamięci masowej działa tylko z systemem IBM WebSphere MQ 7.1 i nowszym, ponieważ wymaga on SMDS i reguł przenoszenia.

Składowanie awaryjne

Aby zmniejszyć prawdopodobieństwo zwrócenia kodu przyczyny MQRD_STORAGE_MEDIUM_FULL do aplikacji IBM MQ podczas rozszerzonego wyłączenia, można użyć funkcji SMDS i przenoszenia komunikatów w połączeniu z systemem SCM.

Przegląd

Pojedyncza kolejka współużytkowana jest skonfigurowana w strukturze aplikacji. Aplikacja umieszcza komunikaty w kolejce współużytkowanej, a aplikacja pobiera komunikaty z kolejki współużytkowanej.

Podczas normalnego działania głębokość kolejki powinna być bliska zeru, ale wymagania biznesowe wskazują, że system musi być w stanie tolerować dwugodzinne wyłączenie aplikacji pobierającej. Oznacza to, że kolejka współużytkowana musi zawierać dwie godziny komunikatów z aplikacji umieszczającej.

Obecnie ten proces jest realizowany przy użyciu domyślnych reguł przenoszenia i SMDS, dzięki czemu wielkość struktury jest minimalizowana, przy jednoczesnym obniżeniu kosztów wydajności związanych z przenoszeniem.

Oczekuje się, że szybkość wysyłania komunikatów do kolejki współużytkowanej podwoi się w perspektywie krótko- i średnioterminowej. Mimo że wymagania, aby system mógł tolerować dwugodzinne wyłączenie, nadal istnieją, w systemie CF nie ma wystarczającej ilości pamięci rzeczywistej, aby podwoić wielkość struktury.

Ponieważ system CF, który zawiera strukturę aplikacji, znajduje się na komputerze z systemem zEC12, istnieje możliwość powiązania wystarczającego systemu SCM ze strukturą w celu zapisania wystarczającej liczby komunikatów, aby można było tolerować dwugodzinne wyłączenie.

Zastanów się, co się dzieje w danym okresie:

1. Początkowo system jest w stanie ustalonym. Zarówno aplikacja umieszczana, jak i pobierana działają normalnie, a głębokość kolejki jest zbliżona do zera lub wynosi zero. W rezultacie struktura aplikacji jest w dużej mierze pusta.
2. W pewnym momencie aplikacja pobierająca ulega nieoczekiwanym awariom i zatrzymuje się. Aplikacja umieszczająca nadal umieszcza komunikaty w kolejce i zaczyna wypełniać strukturę aplikacji.
3. Gdy struktura osiągnie 70% zapętnienia, warunki pierwszej reguły przenoszenia są spełnione, a wszystkie komunikaty o wielkości większej lub równej 32 kB są przenoszone do SMDS.

Przegląd reguł przenoszenia zawiera sekcja [“Reguły SMDS i odciążania” na stronie 197](#).

4. Ponieważ komunikaty są nadal umieszczane w kolejce współużytkowanej, struktura jest nadal wypełniona (z powodu danych komunikatu przechowywanych w strukturze lub w wyniku wskaźników do odciążonych komunikatów przechowywanych w strukturze).

Gdy struktura osiągnie 80% zapętnienia, zaczyna obowiązywać druga reguła przenoszenia, a komunikaty o wielkości 4 kB lub większe są przenoszone do SMDS.

5. Jeśli struktura jest zapętniona w 90%, wszystkie komunikaty są przenoszone do SMDS i tylko wskaźniki komunikatów są umieszczane w strukturze.

W tym czasie algorytm wstępnego przemieszczania rozpoczyna działanie i rozpoczyna przenoszenie danych ze struktury do systemu SCM. Zakładając, że wszystkie komunikaty w kolejce mają ten sam priorytet, najnowsze komunikaty są wstępnie przemieszczane.

Ponieważ wszystkie komunikaty są teraz przenoszone do SMDS, dane przesyłane do systemu SCM nie są rzeczywistymi danymi komunikatu, ale wskaźnikami do komunikatów w SMDS.

W związku z tym liczba komunikatów, które można zapisać w kombinacji struktury oraz SCM i SMDS powiązanych ze strukturą, jest bardzo duża.

Wydajność: Na tym etapie wyłączenia aplikacja może być w pewnym stopniu narażona na spadek wydajności z powodu konieczności zapisu w SMDS. W tym przypadku użycie SCM nie powinno być czynnikiem ograniczającym umieszczanie aplikacji pod względem wydajności. System SCM udostępnia dodatkowe miejsce, aby zapobiec zapełnieniu struktury.

6. W końcu aplikacja uzyskująca dostęp jest ponownie dostępna i wyłączenie jest skończone.

Jednak system SCM jest nadal używany przez strukturę. Aplikacja pobierająca rozpoczyna odczytywanie komunikatów z kolejki, pobierając najpierw najstarsze komunikaty o najwyższym priorytecie.

Ponieważ te komunikaty zostały napisane zanim struktura zaczęła się zapełniać, pochodzą one całkowicie z rzeczywistej części struktury przechowywanej w pamięci.

7. Gdy struktura zaczyna być pusta, spada poniżej wartości progowej, przy której jest aktywne wstępne przemieszczanie, a więc wstępne przemieszczanie zostaje zatrzymane.
8. Użycie struktury zmniejsza się poniżej punktu, w którym reguły przesyłania są stosowane, więc komunikaty nie są już przesyłane do SMDS, chyba że mają więcej niż 63 kB.

W tym momencie algorytm preselekcji rozpoczyna przesyłanie danych z systemu SCM do struktury. Ponieważ aplikacja pobiera komunikaty z kolejki w kolejności oczekiwanej przez algorytm SCM, komunikaty są wprowadzane przed aplikacją pobierającą, która ich potrzebuje.

W rezultacie aplikacja pobierająca nigdy nie musi czekać, aż komunikaty będą wprowadzane synchronicznie z systemu SCM.

9. Ponieważ aplikacja pobierająca kontuuje przesyłanie w dół kolejki, rozpoczyna pobieranie komunikatów, które zostały przeniesione do SMDS.
10. Wreszcie, system jest w stanie stabilnym. W systemie SCM lub SMDS nie są przechowywane żadne komunikaty, a głębokość kolejki jest bliska zeru.

Większa wydajność

W tym scenariuszu opisano korzystanie z systemu SCM w celu zwiększenia liczby komunikatów, które mogą być przechowywane w kolejce współużytkowanej bez ponoszenia kosztów związanych z wydajnością związanych z używaniem SMDS.

Opis

W tym scenariuszu aplikacja umieszczająca i pobierająca komunikuje się za pośrednictwem kolejki współużytkowanej, która jest przechowywana w strukturze aplikacji.

Aplikacja umieszczająca ma tendencję do uruchamiania w porcjach, gdy umieszcza dużą liczbę komunikatów w krótkim czasie. Następnie, w dłuższym okresie czasu, nie generuje żadnych komunikatów.

Aplikacja pobierająca przetwarza sekwencyjnie każdy komunikat i wykonuje złożone przetwarzanie każdego z nich. Oznacza to, że w większości przypadków zapełnienie kolejki wynosi zero, z wyjątkiem sytuacji, gdy aplikacja umieszczająca rozpoczyna działanie, w której zapełnienie kolejki zaczyna rosnąć, ponieważ komunikaty są umieszczane szybciej niż są odbierane.

Zapełnienie kolejki zwiększa się do momentu zatrzymania aplikacji umieszczającej, a aplikacja pobierającej komunikaty ma wystarczająco dużo czasu na przetworzenie wszystkich komunikatów w kolejce.

Uwagi:

1. W tym scenariuszu kluczowym czynnikiem jest wydajność. Komunikaty wysyłane do kolejki są zawsze mniejsze niż 63 kB i dlatego nigdy nie muszą być przesyłane do SMDS.

2. Struktura aplikacji została tak dobrana, aby była wystarczająco duża, aby zawierała wszystkie komunikaty, które zostaną na niej umieszczone przez aplikację umieszczoną w pojedynczym rozdzielaniu.
3. Wszystkie reguły przenoszenia muszą być wyłączone, aby nawet po wypełnieniu struktury komunikaty nie były przenoszone do SMDS. Wynika to z faktu, że koszty związane z wydajnością związane z zapisywaniem komunikatów do SMDS i odczytem komunikatów z SMDS są uważane za niedopuszczalne.

Z biegiem czasu liczba komunikatów wysyłanych przez aplikację umieszczającą w porcji musi wzrosnąć o kilka rzędów wielkości. Ponieważ aplikacja pobierająca musi przetwarzać każdy komunikat sekwencyjnie, liczba komunikatów w kolejce zwiększa się do punktu, w którym struktura wypełnia się.

W tym momencie aplikacja umieszczająca otrzymuje kod przyczyny (MQRC_STORAGE_MEDIUM_FULL) podczas umieszczania komunikatu i operacja umieszczania kończy się niepowodzeniem. Aplikacja umieszczana może chwilowo tolerować okresy tylko wtedy, gdy nie jest w stanie umieścić komunikatów w kolejce. Jeśli okres jest zbyt długi, aplikacja kończy działanie.

Zakładając, że nie masz dostępnego czasu lub umiejętności na przepisanie aplikacji lub aplikacji, ten problem ma trzy możliwe rozwiązania:

1. Zwiększ wielkość struktury aplikacji.
2. Dodaj reguły przenoszenia do struktury aplikacji, aby komunikaty były przenoszone do SMDS w momencie rozpoczęcia wypełniania kolejki.
3. Powiąż system SCM ze strukturą.

Pierwsze rozwiązanie jest szybkie w implementacji, ale w systemie CF nie ma wystarczającej ilości rzeczywistej pamięci masowej.

Drugie rozwiązanie może być również szybkie w implementacji, ale wpływ na wydajność przenoszenia do SMDS jest uważany za zbyt istotny, aby można było użyć tej opcji.

Trzecie rozwiązanie, które wiąże system SCM ze strukturą, zapewnia akceptowalne równoważenie kosztów i wydajności.

Powiązanie systemu SCM ze strukturą powoduje większe użycie rzeczywistej pamięci masowej w systemie CF z powodu rozszerzonej pamięci masowej, która jest używana do wykonywania operacji. Jednak rzeczywista ilość pamięci rzeczywistej będzie mniejsza niż ilość użyta w pierwszej opcji.

Innym czynnikiem jest koszt SCM. Jednak koszt ten jest znacznie tańszy niż rzeczywisty magazyn. Czynniki te łączą się, aby trzecia opcja była tańsza niż pierwsza opcja.

Chociaż trzecia opcja, potencjalnie, może nie działać tak dobrze, jak pierwsza, algorytmy preselekcji i wstępnego przemieszczania używane przez system CF mogą się łączyć, aby różnice w wydajności mogły być akceptowalne lub w niektórych przypadkach nieistotne.

Z pewnością wydajność może być znacznie lepsza niż użycie SMDS do przenoszenia komunikatów.

Zastanów się, co się dzieje w danym okresie:

1. Początkowo aplikacja pobierająca jest aktywna i oczekuje na dostarczenie komunikatów do kolejki współużytkowanej. Aplikacja umieszczana nie jest aktywna, a kolejka współużytkowana jest pusta.
2. W określonym momencie aplikacja umieszczająca komunikaty staje się aktywna i rozpoczyna umieszczanie dużej liczby komunikatów we współużytkowanej kolejce. Aplikacja pobierająca rozpoczyna pobieranie komunikatów, ale wypełnienie kolejki szybko zaczyna rosnąć, ponieważ aplikacja pobierająca jest wolniejsza od aplikacji umieszczającej.

W rezultacie struktura aplikacji zaczyna się wypełniać.

3. Wraz ze wzrostem czasu aplikacja umieszczana jest nadal aktywna. Struktura aplikacji wypełnia się do około 90%.

Jest to moment, w którym algorytm wstępnego przemieszczania SCM rozpoczyna przenoszenie komunikatów ze struktury do systemu SCM, zwalniając miejsce w strukturze.

Ponieważ aplikacja pobierająca pobiera najpierw najstarsze komunikaty o najwyższym priorytecie z kolejki, zawsze pobiera komunikaty ze struktury i nie musi czekać, aż komunikaty zostaną wprowadzone synchronicznie z systemu SCM do struktury.

4. Aplikacja umieszczana jest nadal aktywna i umieszcza komunikaty we współużytkowanej kolejce. Jednak aplikacja nigdy nie otrzymuje kodu przyczyny MQR_STORAGE_MEDIUM_FULL, ponieważ w systemie SCM istnieje wystarczająca ilość miejsca do przechowywania wszystkich komunikatów, które nie mieszczą się w strukturze.
5. W końcu aplikacja wstawiania zatrzymuje się, ponieważ nie ma więcej komunikatów do umieszczenia. Algorytm wstępnego przemieszczania zostaje zatrzymany, ponieważ struktura jest poniżej 90% w użyciu, a aplikacja pobierająca kontuuje przetwarzanie komunikatów w kolejce.
6. Gdy aplikacja pobierająca rozpoczyna zwalnianie miejsca w strukturze, algorytm preselekcji rozpoczyna pobieranie komunikatów z systemu SCM do struktury.

Ponieważ aplikacja pobierająca przetwarza komunikaty w kolejności oczekiwanej przez algorytm preselekcji, aplikacja pobierająca nigdy nie jest blokowana, oczekując na synchroniczne wprowadzenie danych komunikatu z systemu SCM do struktury.
7. Na koniec aplikacja pobierająca przetwarza wszystkie komunikaty w kolejce współużytkowanej i czeka na udostępnienie następnego komunikatu. Struktura i system SCM są puste dla komunikatów.

Pamięć awaryjna-podstawowa konfiguracja

W jaki sposób można skonfigurować podstawowy scenariusz awaryjnej pamięci masowej w systemie IBM MQ.

O tym zadaniu

Ważne: Planowane jest, że IBM z16 będzie ostatnią generacją IBM Z[®] obsługującej użycie wirtualnej pamięci Flash (znanej również jako pamięć klasy pamięci masowej lub SCM) dla obrazów narzędzia CF. Więcej informacji na ten temat zawierają: [IBM Z i IBM LinuxONE 4Q 2023 Oświadczenia o kierunku](#).

Alternatywnie należy użyć większych struktur lub przenieść komunikaty do SMDS.

Aby zmniejszyć prawdopodobieństwo zwrócenia kodu przyczyny MQR_STORAGE_MEDIUM_FULL do aplikacji IBM MQ podczas rozszerzonego wyłączenia, można użyć funkcji SMDS i przenoszenia komunikatów w połączeniu z systemem SCM.

Na przykład w przedsiębiorstwie znajduje się aplikacja, która umieszcza komunikaty w kolejce i aplikacja, która pobiera komunikaty z kolejki. Podczas normalnego działania oczekiwane jest, że głębokość kolejki będzie bliska zeru, ale wymaganie biznesowe wskazuje, że system może tolerować dwugodzinne wyłączenie aplikacji, która pobiera komunikaty.

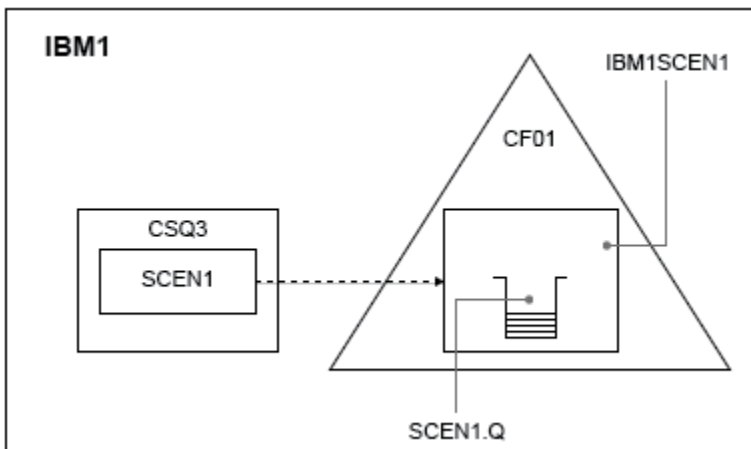
Oznacza to, że używana kolejka współużytkowana musi zawierać dwie godziny komunikatów z aplikacji umieszczającej. Obecnie można to osiągnąć za pomocą domyślnych reguł przenoszenia i SMDS.

Oczekuje się, że szybkość wysyłania komunikatów do kolejki współużytkowanej podwoi się w perspektywie krótko- i średnioterminowej. Pomimo tego, że system nadal musi tolerować dwugodzinne wyłączenie, w systemie CF nie ma wystarczającej ilości pamięci rzeczywistej, aby podwoić wielkość struktury. Ponieważ system CF zawierający strukturę aplikacji rezyduje na komputerze z systemem zEC12, istnieje możliwość powiązania wystarczającego systemu SCM ze strukturą do przechowywania wystarczającej liczby komunikatów, co pozwala na tolerowanie dwugodzinnego wyłączenia.

W tym scenariuszu początkowym używane są:

- Grupa współużytkowania kolejek IBM1, która zawiera pojedynczy menedżer kolejek CSQ3. Oprócz struktury administracyjnej grupa współużytkowania kolejek zdefiniuje pojedynczą strukturę aplikacji SCEN1.
- Narzędzie CF CF01, w którym struktura aplikacji SCEN1 jest przechowywana jako struktura IBM1SCEN1. Maksymalna wielkość tej struktury wynosi 1 GB.
- Pojedyncza kolejka współużytkowana SCEN1. Q używana przez strukturę aplikacji.

Ta konfiguracja została zilustrowana na ilustracji Rysunek 62 na stronie 204.



Rysunek 62. Konfiguracja podstawowa

Ponadto założmy, że menedżer kolejek CSQ3 jest już jedynym elementem grupy współużytkowania kolejek IBM1.

Należy dodać definicję struktury IBM1SCEN1 do strategii menedżera zasobów CF (Coupling Facility) menedżera zasobów CFRM. Dla uproszczenia struktura jest definiowana w taki sposób, aby można ją było utworzyć tylko w pojedynczym narzędziu CF CF01, podając parametr PREFLIST (CF01).



Ostrzeżenie: Aby zapewnić wysoką dostępność w systemie produkcyjnym, należy dołączyć co najmniej dwa systemy CF do pliku PREFLIST dla wszystkich struktur używanych przez produkt IBM MQ.

Procedura

1. Odśwież strategię CFRM za pomocą następującej komendy:

```
SETXCF START,POLICY,TYPE=CFRM,POLNAME=IBM1SCEN1
```

Przykładowa strategia CFRM dla struktury IBM1SCEN1:

```
STRUCTURE  
NAME(IBM1SCEN1)  
SIZE(1024M)  
INITSIZE(512M)  
ALLOWAUTOALT(YES)  
FULLTHRESHOLD(85)  
PREFLIST(CF01)  
ALLOWREALLOCATE(YES)  
DUPLEX(DISABLED)  
ENFORCEORDER(NO)
```

2. Sprawdź, czy struktura została poprawnie utworzona, używając następującej komendy:

```
D XCF,STR,STRNAME=IBM1SCEN1
```

W tym momencie struktura nie została przydzielona (wyświetlana w wierszu STATUS) do grupy współużytkowania kolejek.

3. Skonfiguruj IBM MQ, aby użyć struktury zdefiniowanej w strategii CFRM.

- a. Użyj komendy DEFINE CFSTRUCT z nazwą struktury SCEN1, aby utworzyć obiekt IBM MQ CFSTRUCT:

```
DEFINE CFSTRUCT(SCEN1)  
CFCONLOS(TOLERATE)
```

```
CFLEVEL(5)
DESCR('Structure for SCM scenario 1')
RECOVER(NO)
RECAUTO(YES)
OFFLOAD(DB2)
OFFLD1SZ(64K) OFFLD1TH(70)
OFFLD2SZ(64K) OFFLD2TH(80)
OFFLD3SZ(64K) OFFLD3TH(90)
```

- b. Sprawdź poprawność struktury za pomocą komendy `DISPLAY CFSTRUCT`.
- c. Zdefiniuj współużytkowaną kolejkę `SCEN1.Q`, aby użyć struktury `SCEN1`, używając następującej komendy `MQSC`:

```
DEFINE QLOCAL(SCEN1.Q) QSGDISP(SHARED) CFSTRUCT(SCEN1) MAXDEPTH(999999999)
```

4. Użyj komendy IBM MQ Explorer, aby umieścić pojedynczy komunikat w kolejce `SCEN1.Q` i ponownie go zdjąć.
5. Wydadź następującą komendę, aby sprawdzić, czy struktura jest teraz przydzielona:

```
D XCF,STR,STRNAME=IBM1SCEN1
```

Sprawdź dane wyjściowe komendy, w wierszu `STATUS` wpisz `ALLOCATED`.

Wyniki

Utworzono konfigurację podstawową. Teraz można uzyskać informacje na temat podstawowej wydajności konfiguracji przy użyciu dowolnej wybranej metody.

Co dalej

Dodaj `SMDS` i `SCM` do struktury początkowej

Pojęcia pokrewne

“[Użycie pamięci klasy pamięci masowej z kolejkami współużytkowanymi](#)” na stronie 195

Użycie pamięci klasy pamięci masowej (`SCM`) może być korzystne w przypadku używania z kolejkami współużytkowanymi IBM MQ for z/OS.

 *Dodawanie zestawów `SMDS` i `SCM` do struktury początkowej*

Sposób dodawania zestawów `SMDS` i `SCM` na potrzeby pamięci awaryjnej w systemie IBM MQ.

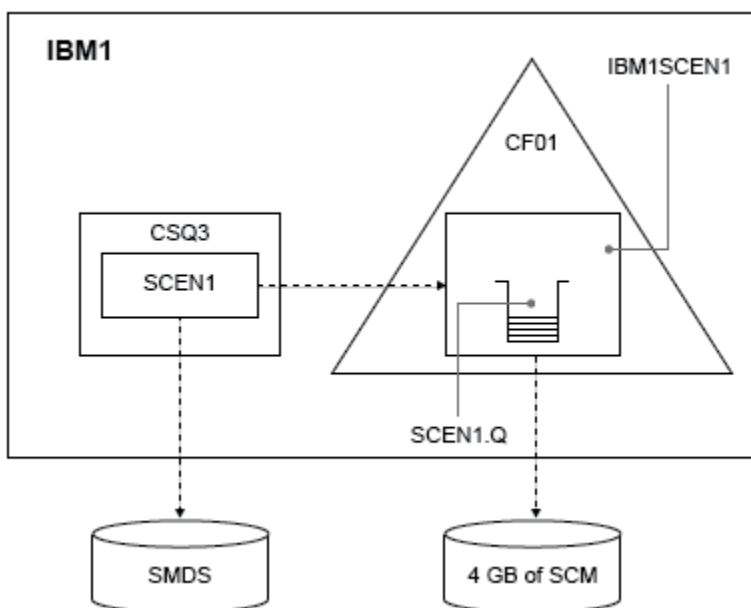
O tym zadaniu

Ważne: Planowane jest, że IBM z16 będzie ostatnią generacją IBM Z® obsługującej użycie wirtualnej pamięci Flash (znanej również jako pamięć klasy pamięci masowej lub `SCM`) dla obrazów narzędzia CF. Więcej informacji na ten temat zawierają: [IBM Z i IBM LinuxONE 4Q 2023 Oświadczenia o kierunku](#).

Alternatywnie należy użyć większych struktur lub przenieść komunikaty do `SMDS`.

Ta część zadania korzysta z konfiguracji podstawowej opisanej w sekcji “[Pamięć awaryjna-podstawowa konfiguracja](#)” na stronie 203. Scenariusz opisuje dodawanie współużytkowanych zestawów danych komunikatów (`SMDS`), a następnie systemu `SCM` do struktury początkowej.

Ta ostateczna konfiguracja została zilustrowana w sekcji [Rysunek 63](#) na stronie 206.



Rysunek 63. Konfiguracja z dodaniem SMDS i SCM dla pamięci masowej awaryjnej

Procedura

1. Utwórz zestaw danych SMDS używany przez strukturę aplikacji SCEN1 , edytując przykładowy kod JCL **CSQ4SMDS** , jak pokazano poniżej:

```
//CSQ4SMDS JOB NOTIFY=&SYSUID
//*
/* Allocate SMDS
/*
//DEFINE EXEC PGM=IDCAMS,REGION=4M
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DEFINE CLUSTER          -
(NAME(CSQSMDS.SCEN1.CSQ3.SMDS) -
MEGABYTES(5000 3000) -
LINEAR                  -
SHAREOPTIONS(2 3) ) -
DATA                    -
(NAME(CSQSMDS.SCEN1.CSQ3.SMDS.DATA) )
/*
/*
/* Format the SMDS
/*
//FORM EXEC PGM=CSQJUFMT,COND=(0,NE),REGION=0M
//STEPLIB DD DSN=MQ800.SCSQANLE,DISP=SHR
// DD DSN=MQ800.SCSQAUTH,DISP=SHR
//SYSUT1 DD DISP=OLD,DSN=CSQSMDS.SCEN1.CSQ3.SMDS
//SYSPRINT DD SYSOUT=*
```

2. Wydadź komendę **ALTER CFSTRUCT** , aby zmienić strukturę aplikacji SCEN1 w celu użycia SMDS do odciążania, implementując domyślne reguły odciążania:

```
ALTER CFSTRUCT(SCEN1) OFFLOAD(SMDS) OFFLD1SZ(32K) OFFLD2SZ(4K) OFFLD3SZ(0K)
DSGROUP('CSQSMDS.SCEN1.*.SMDS') DSBLOCK(1M)
```

Na co zwrócić uwagę:

- Ponieważ SCEN1.Q jest jedyną kolejką współużytkowaną w strukturze aplikacji SCEN1 , wartość **DSBLOCK** została ustawiona na 1M, co jest największą możliwą wartością. To powinno być najbardziej efektywne ustawienie dla naszego scenariusza.

- Ponieważ komunikaty wysyłane przez aplikację umieszczającą mają wielkość 30 kB, przenoszenie do SMDS nie jest uruchamiane do momentu spełnienia drugiej reguły przenoszenia, gdy struktura jest zapelniona w 80%.
- Uruchom ponownie aplikację testową.
Zanotuj zwiększoną ilość pamięci komunikatów w kolejce.
 - Dodaj 4 GB SCM do struktury IBM1SCEN1 , wykonując następującą procedurę:
 - Sprawdź ilość zainstalowanego systemu SCM i przydzielonego do produktu CF01, wprowadzając następującą komendę:

```
D CF,CFNAME=CF01
```

- Sprawdź wartości STORAGE - CLASS MEMORY w sekcji STORAGE CONFIGURATION wyświetlanych danych wyjściowych, aby wyświetlić dostępną pamięć masową.
- Zaktualizuj strategię CFRM, używając słów kluczowych SCMMAXSIZE i SCMALGORITHM w następujący sposób:

```
STRUCTURE
NAME (IBM1SCEN1)
SIZE (1024M)
INITSIZE (512M)
ALLOWAUTOALT (YES)
FULLTHRESHOLD (85)
PREFLIST (CF01)
ALLOWREALLOCATE (YES)
DUPLEX (DISABLED)
ENFORCEORDER (NO)
SCMMAXSIZE (4G)
SCMALGORITHM (KEYPRIORITY1)
```

- Aktywuj strategię CFRM, wprowadzając następującą komendę:

```
SETXCF START,POLICY,TYPE=CFRM,POLNAME=polname
```

- Odbuduj strukturę IBM1SCEN1 .

Tę procedurę należy wykonać, ponieważ struktura została przydzielona podczas wprowadzania wcześniejszych zmian.

Wprowadź następującą komendę, aby odbudować strukturę:

```
SETXCF START,REBUILD,STRNM=IBM1SCEN1
```

Wyniki

System SCM został pomyślnie dodany do konfiguracji.

Co dalej

Optymalizacja wydajności systemu. Więcej informacji zawiera sekcja [“Optymalizowanie wykorzystania pamięci klasy pamięci masowej”](#) na stronie 207.

 *Optymalizowanie wykorzystania pamięci klasy pamięci masowej*

W jaki sposób można usprawnić korzystanie z pamięci klasy pamięci masowej (SCM).

Ważne: Planowane jest, że IBM z16 będzie ostatnią generacją IBM Z[®] obsługującej użycie wirtualnej pamięci Flash (znanej również jako pamięć klasy pamięci masowej lub SCM) dla obrazów narzędzia CF. Więcej informacji na ten temat zawierają: [IBM Z i IBM LinuxONE 4Q 2023 Oświadczenia o kierunku](#).

Alternatywnie należy użyć większych struktur lub przenieść komunikaty do SMDS.

Uruchom następującą komendę:

```
D XCF,STR,STRNAME=IBM1SCEN1
```

Ponieważ struktura była już pełna danymi komunikatu, z powodu poprzednich testów część odbudowywania obejmowała wstępne przemieszczanie niektórych komunikatów ze struktury do systemu SCM. Ten proces został zainicjowany przy użyciu poprzedniej komendy.

Dane wyjściowe tej komendy są generowane, na przykład:

```
ACTIVE STRUCTURE
-----
ALLOCATION TIME: 06/17/2014 09:28:50
CFNAME : CF01
COUPLING FACILITY: 002827.IBM.02.00000000B8D7
PARTITION: 3B CPCID: 00
STORAGE CONFIGURATION ALLOCATED MAXIMUM %
ACTUAL SIZE: 1024 M 1024 M 100
AUGMENTED SPACE: 3 M 142 M 2
STORAGE-CLASS MEMORY: 88 M 4096 M 2
ENTRIES: 120120 1089536 11
ELEMENTS: 240240 15664556 1
SPACE USAGE IN-USE TOTAL %
ENTRIES: 84921 219439 38
ELEMENTS: 2707678 3149050 85
EMCS: 2 282044 0
LOCKS: 1024
SCMHIGHTHRESHOLD : 90
SCMLOWTHRESHOLD : 70
ACTUAL SUBNOTIFYDELAY: 5000
PHYSICAL VERSION: CD5186A0 2BD8B85C
LOGICAL VERSION: CD515C50 CE2ED258
SYSTEM-MANAGED PROCESS LEVEL: 9
XCF GRPNAME : IXCLO053
DISPOSITION : KEEP
ACCESS TIME : NOLIMIT
MAX CONNECTIONS : 32
# CONNECTIONS : 1
CONNECTION NAME ID VERSION SYSNAME JOBNAME ASID STATE
-----
CSQEIBM1CSQ301 01 00010059 SC61 CSQ3MSTR 0091 ACTIVE
```

Należy zwrócić uwagę na następujące informacje z danych wyjściowych komendy:

- Program STORAGE_CLASS MEMORY potwierdza, że do struktury dodano system SCM **MAXIMUM** o wielkości 4096 MB.
- Wartość ALLOCATED określa ilość danych STORAGE-CLASS MEMORY używanych na potrzeby przemieszczania wstępnego. W strukturze jest teraz wolne miejsce, w którym nie było miejsca przed dodaniem komponentu SCM.
- Ilość AUGMENTED SPACE używana do śledzenia użycia SCM.
- Punkt, w którym algorytm wstępnego przemieszczania rozpoczyna przenoszenie danych ze struktury do systemu SCM, jest wtedy, gdy struktura jest zapełniona w 90%. Jest to wskazywane przez niekonfigurowalną właściwość **SCMHIGHTHRESHOLD**.
- Poniżej znajduje się punkt, w którym algorytm pobierania wstępnego rozpoczyna przenoszenie danych z systemu SCM do struktury, gdy struktura jest zapełniona w 70%. Jest to wskazywane przez niekonfigurowalną właściwość **SCMLOWTHRESHOLD**.

Teraz można przetestować różne sposoby optymalizacji użycia systemu SCM. Na co zwrócić uwagę:

- Po użyciu systemu SCM do przechowywania komunikatów nie można zmienić struktury, dopóki wszystkie dane nie zostaną usunięte z systemu SCM.

W tym przypadku oznacza to, że współczynnik pozycji do elementu jest zamrożony w wartości, która znajdowała się w miejscu, gdy system SCM był używany po raz pierwszy. Należy upewnić się, że struktura jest w pożądanym stanie, zanim algorytm wstępnego przemieszczania rozpocznie przenoszenie danych do systemu SCM.

- Czy bieżąca wielkość struktury jest poprawna przed użyciem SCM?

Na przykład, czy zwiększono **INITSIZE** z 512 MB do 1 GB?

Jeśli ta czynność nie zostanie włączona, możliwe jest, że pomimo włączenia automatycznej modyfikacji struktury, algorytm wstępnego przemieszczania rozpocznie przenoszenie danych do systemu SCM,

zanim będzie można rozpocząć tę zmianę. W wyniku tego struktura jest zamrożona przy użyciu 512 MB pamięci rzeczywistej.

- Czy współczynnik pozycji do elementu jest poprawny przed użyciem systemu SCM?

Celem tego scenariusza jest zwiększenie liczby odciążonych wskaźników komunikatów, które mogą być przechowywane w strukturze i systemie SCM jako całości, a także zachowanie jak największej liczby komunikatów w pamięci masowej struktury. Dostęp do tych komunikatów jest szybszy niż dostęp do komunikatów w SMDS.

Dlatego konieczne jest posiadanie struktury rozpoczynającej się współczynnikiem wejścia do elementu, który jest dobry do przechowywania komunikatów, a następnie przejście do współczynnika, który jest dobry do przechowywania wskaźników komunikatów przed pierwszym uruchomieniem algorytmu wstępnego etapu. To przejście można częściowo wykonać, korzystając z reguły przenoszenia IBM MQ .

Zmień reguły przenoszenia, wydając następującą komendę:

```
ALTER CFSTRUCT(SCEN1) OFFLD1SZ(0K)
```

Może być konieczne wykonanie kilku przebiegów w celu zoptymalizowania współczynników wejścia do elementu.

W poniższej tabeli przedstawiono możliwe ulepszenia dotyczące liczby komunikatów umieszczanych w kolejce w różnych fazach scenariusza pamięci awaryjnej.

| Opis testu | Liczba komunikatów | Czas wypełniania kolejki (sekundy) |
|--|--------------------|------------------------------------|
| Konfiguracja podstawowa | 27 850 | 3.2 |
| SMDS z domyślnymi regułami przenoszenia | 205 000 | 158 |
| SCM z domyślnymi regułami przenoszenia | 828 610 | 469 |
| SCM z dopasowanymi regułami przenoszenia | 1,135,775 | 679 |

Ostatni wiersz w tabeli pokazuje, że dopasowywanie reguł przenoszenia miało wymagany efekt.

Należy sprawdzić system, aby sprawdzić, czy można poprawić te dane liczbowe w jakikolwiek sposób. Na przykład może zabraknąć dostępnej pamięci masowej SMDS. Jeśli można przydzielić więcej pamięci SMDS, powinno być możliwe znaczne zwiększenie liczby komunikatów w kolejce.

Poprawiona wydajność-podstawowa konfiguracja

W jaki sposób można skonfigurować podstawowy scenariusz w celu zwiększenia wydajności przy użyciu kolejek współużytkowanych w systemie IBM MQ.

O tym zadaniu

Ważne: Planowane jest, że IBM z16 będzie ostatnią generacją IBM Z[®] obsługującej użycie wirtualnej pamięci Flash (znanej również jako pamięć klasy pamięci masowej lub SCM) dla obrazów narzędzia CF. Więcej informacji na ten temat zawierają: [IBM Z i IBM LinuxONE 4Q 2023 Oświadczenia o kierunku](#).

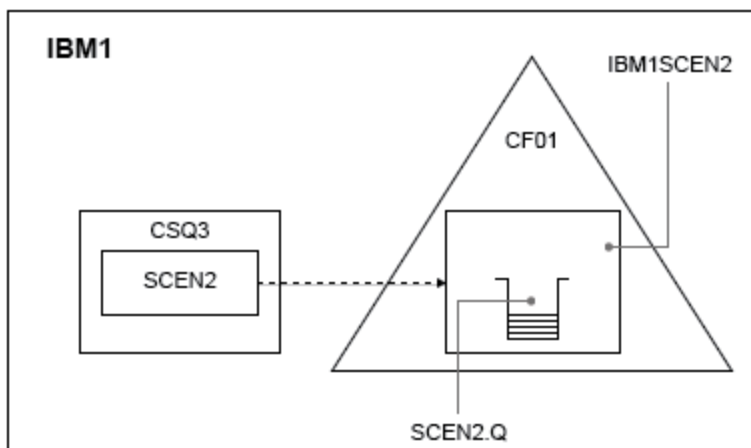
Alternatywnie należy użyć większych struktur lub przenieść komunikaty do SMDS.

W tym scenariuszu opisano użycie systemu SCM w celu zwiększenia liczby komunikatów, które mogą być przechowywane w kolejce współużytkowanej bez ponoszenia kosztów wydajności związanych z używaniem SMDS.

Ten początkowy scenariusz jest bardzo podobny do tego, który jest używany do przechowywania w nagłych wypadkach i używa:

- Grupa współużytkowania kolejek IBM1, która zawiera pojedynczy menedżer kolejek CSQ3. Oprócz struktury administracyjnej grupa współużytkowania kolejek zdefiniuje pojedynczą strukturę aplikacji SCEN2.
- Narzędzie CF CF01, w którym struktura aplikacji SCEN2 jest przechowywana jako struktura IBM1SCEN2. Maksymalna wielkość tej struktury wynosi 2 GB.
- Pojedyncza kolejka współużytkowana, SCEN2.Q, która jest skonfigurowana do korzystania ze struktury aplikacji.

Ta konfiguracja została zilustrowana na ilustracji [Rysunek 64](#) na stronie 210.



Rysunek 64. Konfiguracja podstawowa

Ponadto założmy, że menedżer kolejek CSQ3 jest już jedynym elementem grupy współużytkowania kolejek IBM1.

Należy dodać definicję struktury IBM1SCEN2 do strategii menedżera zasobów CF (Coupling Facility) menedżera zasobów CFRM. Dla uproszczenia struktura jest definiowana w taki sposób, aby można ją było utworzyć tylko w pojedynczym narzędziu CF CF01, podając parametr PREFLIST (CF01).

Przykładowa strategia CFRM dla struktury IBM1SCEN2:

```
STRUCTURE
NAME (IBM1SCEN2)
SIZE (2048M)
INITSIZE (2048M)
ALLOWAUTOALT (YES)
FULLTHRESHOLD (85)
PREFLIST (CF01)
ALLOWREALLOCATE (YES)
DUPLEX (DISABLED)
ENFORCEORDER (NO)
```

Zarówno słowa kluczowe **INITSIZE**, jak i **SIZE** mają wartość 2048M, dlatego nie można zmienić wielkości struktury.

Procedura

1. Odśwież strategię CFRM za pomocą następującej komendy:

```
SETXCF START,POLICY,TYPE=CFRM,POLNAME=IBM1SCEN2
```

2. Sprawdź, czy struktura została poprawnie utworzona, używając następującej komendy:

```
D XCF,STR,STRNAME=IBM1SCEN2
```

Wykonanie powyższej komendy powoduje wyświetlenie następujących danych wyjściowych:

```

RESPONSE=SC61
IXC360I 07.58.51 DISPLAY XCF 581
STRNAME: IBM1SCEN2
STATUS: NOT ALLOCATED
POLICY INFORMATION:
POLICY SIZE : 2048 M
POLICY INITSIZE: 2048 M
POLICY MINSIZE : 1536 M
FULLTHRESHOLD : 85
ALLOWAUTOALT : YES
REBUILD PERCENT: N/A
DUPLEX : DISABLED
ALLOWREALLOCATE: YES
PREFERENCE LIST: CF01
ENFORCEORDER : NO
EXCLUSION LIST IS EMPTY

EVENT MANAGEMENT: MESSAGE-BASED   MANAGER SYSTEM NAME: SC53
MANAGEMENT LEVEL : 01050107

```

W tym momencie struktura nie została przydzielona (wyświetlana w wierszu STATUS) do grupy współużytkowania kolejek.

3. Skonfiguruj IBM MQ , aby użyć struktury zdefiniowanej w strategii CFRM.

- a. Użyj komendy DEFINE CFSTRUCT z nazwą struktury SCEN2 , aby utworzyć obiekt IBM MQ CFSTRUCT.

```

DEFINE CFSTRUCT(SCEN2)
CFCONLOS(TOLERATE)
CFLEVEL(5)
DESCR('Structure for SCM scenario 2')
RECOVER(NO)
RECAUTO(YES)
OFFLOAD(DB2)
OFFLD1SZ(64K) OFFLD1TH(70)
OFFLD2SZ(64K) OFFLD2TH(80)
OFFLD3SZ(64K) OFFLD3TH(90)

```

- b. Sprawdź strukturę za pomocą komendy DISPLAY CFSTRUCT .
- c. Zdefiniuj współużytkowaną kolejkę SCEN2 . Q , aby użyć struktury SCEN2 , używając następującej komendy MQSC:

```

DEFINE QLOCAL(SCEN2.Q) QSGDISP(SHARED) CFSTRUCT(SCEN2) MAXDEPTH(999999999)

```

4. Użyj programu IBM MQ Explorer, aby umieścić pojedynczy komunikat w kolejce SCEN2 . Q i ponownie go zdjąć.
5. Wydadaj następującą komendę, aby sprawdzić, czy struktura jest teraz przydzielona:

```

D XCF,STR,STRNAME=IBM1SCEN2

```

Przejrzyj dane wyjściowe komendy, której część jest wyświetlana, i upewnij się, że w wierszu STATUS jest wyświetlana wartość ALLOCATED.

```

RESPONSE=SC61
IXC360I 08.31.27 DISPLAY XCF 703
STRNAME: IBM1SCEN2
STATUS: ALLOCATED
EVENT MANAGEMENT: MESSAGE-BASED
TYPE: SERIALIZED LIST
POLICY INFORMATION:
POLICY SIZE : 2048 M
POLICY INITSIZE: 2048 M
POLICY MINSIZE : 1536 M
FULLTHRESHOLD : 85
ALLOWAUTOALT : YES
REBUILD PERCENT: N/A
DUPLEX : DISABLED
ALLOWREALLOCATE: YES

```

```
PREFERENCE LIST: CF01
ENFORCEORDER : NO
EXCLUSION LIST IS EMPTY
```

Ponadto należy zwrócić uwagę na wartości pól w sekcji SPACE USAGE:

- Pozycje
- Elementy
- EMCS,
- BLOKADY

Poniżej przedstawiono przykładowe wartości:

| SPACE USAGE | IN-USE | TOTAL | % |
|-------------|---------|---------|----|
| ENTRIES: | 344686 | 345242 | 99 |
| ELEMENTS: | 6548455 | 6548467 | 99 |
| EMCS: | 2 | 780318 | 0 |
| LOCKS: | 1024 | | |

Wyniki

Utworzono konfigurację podstawową. Teraz można uzyskać informacje na temat podstawowej wydajności konfiguracji przy użyciu dowolnej wybranej metody.

Co dalej

Należy przetestować podstawowy scenariusz. Na przykład można użyć następujących trzech aplikacji, uruchamiając aplikacje w podanej kolejności i uruchamiając je współbieżnie.

1. Użyj aplikacji PCF, aby zażądać bieżącej głębokości (**CURDEPTH**) wartość SCEN2 . Q co pięć sekund. Dane wyjściowe mogą być użyte do wykreślenia głębokości kolejki w czasie.
2. Aplikacja pobierająca w pojedynczym wątku wielokrotnie pobiera komunikaty z produktu SCEN2 . Q, korzystając z metody get z nieskończonym oczekiwaniem. Aby zasymulować przetwarzanie usuniętych komunikatów, aplikacja pobierająca zatrzyma się na cztery milisekundy na każde dziesięć usuniętych komunikatów.
3. Pojedyncza aplikacja umieszczania w wątku umieszcza w pliku SCEN2 . Q łącznie milion nietrwałych komunikatów o wielkości 4 kB. Ta aplikacja nie jest wstrzymywana między umieszczaniem poszczególnych komunikatów, więc komunikaty są umieszczane w produkcie SCEN2 . Q szybciej niż może je uzyskać aplikacja pobierająca.

W wyniku tego, gdy aplikacja umieszczana jest uruchomiona, głębokość SCEN2 . Q zwiększa się.

Gdy struktura IBM1SCEN2 jest wypełniona, a aplikacja umieszczająca odbiera kod przyczyny MQRC_STORAGE_MEDIUM_FULL, aplikacja umieszczająca jest uśpiona przez pięć sekund przed próbą umieszczenia następnego komunikatu w kolejce.


Wyniki aplikacji CURDEPTH można wykreślić w czasie. Otrzymujesz jakąś formę wyjścia fal saw-zęba, ponieważ aplikacja wstrzymuje się, aby umożliwić częściowe opróżnianie kolejki.

Przejdź do sekcji [“Dodawanie systemu SCM do struktury początkowej”](#) na stronie 212.

Pojęcia pokrewne

[“Użycie pamięci klasy pamięci masowej z kolejkami współużytkowanymi”](#) na stronie 195

Użycie pamięci klasy pamięci masowej (SCM) może być korzystne w przypadku używania z kolejkami współużytkowanymi IBM MQ for z/OS .

 [Dodawanie systemu SCM do struktury początkowej](#)

Sposób dodawania komponentu SCM w celu zwiększenia wydajności w produkcie IBM MQ.

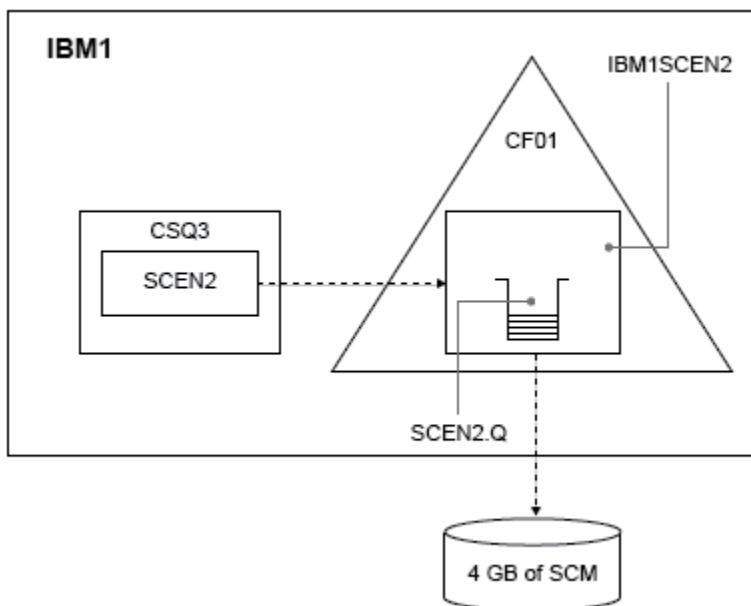
O tym zadaniu

Ważne: Planowane jest, że IBM z16 będzie ostatnią generacją IBM Z® obsługującej użycie wirtualnej pamięci Flash (znanej również jako pamięć klasy pamięci masowej lub SCM) dla obrazów narzędzia CF. Więcej informacji na ten temat zawierają: [IBM Z i IBM LinuxONE 4Q 2023 Oświadczenia o kierunku](#).

Alternatywnie należy użyć większych struktur lub przenieść komunikaty do SMDS.

Ta część zadania korzysta z konfiguracji podstawowej opisanej w sekcji [“Poprawiona wydajność- podstawowa konfiguracja”](#) na stronie 209. Scenariusz opisuje dodawanie systemu SCM do struktury początkowej.

Ta ostateczna konfiguracja została zilustrowana w sekcji [Rysunek 65](#) na stronie 213.



Rysunek 65. Dodawanie konfiguracji SCM w celu zwiększenia wydajności

Procedura

1. Dodaj 4 GB SCM do struktury IBM1SCEN2 , wykonując następującą procedurę:

- Sprawdź ilość zainstalowanego systemu SCM i przydzielonego do produktu CF01, wprowadzając następującą komendę:

```
D CF,CFNAME=CF01
```

- Sprawdź wartości STORAGE - CLASS MEMORY w sekcji STORAGE CONFIGURATION wyświetlanych danych wyjściowych, aby wyświetlić dostępną pamięć masową.
- Zaktualizuj strategię CFRM, używając słów kluczowych SCMMAXSIZE i SCMALGORITHM w następujący sposób:

```
STRUCTURE  
NAME(IBM1SCEN2)  
SIZE(2048M)  
INITSIZE(2048M)  
ALLOWAUTOALT(YES)  
FULLTHRESHOLD(85)  
PREFLIST(CF01)  
ALLOWREALLOCATE(YES)  
DUPLEX(DISABLED)  
ENFORCEORDER(NO)  
SCMMAXSIZE(4G)  
SCMALGORITHM(KEYPRIORITY1)
```

2. Aktywuj strategię CFRM, wprowadzając następującą komendę:

```
SETXCF START ,POLICY ,TYPE=CFRM ,POLNAME=IBM1SCEN2
```

3. Odbuduj strukturę IBM1SCEN2 .

Tę procedurę należy wykonać, ponieważ struktura została przydzielona podczas wprowadzania wcześniejszych zmian.

Wprowadź następującą komendę, aby odbudować strukturę:

```
SETXCF START ,REBUILD ,STRNM=IBM1SCEN2
```

4. Wydaj następującą komendę, aby potwierdzić nową konfigurację struktury:

```
D XCF ,STR ,STRNAME=IBM1SCEN2
```

Przejrzyj dane wyjściowe komendy, z których część jest następująca:

| SPACE USAGE | IN-USE | TOTAL | % |
|-------------|--------|---------|---|
| ENTRIES: | 33 | 342684 | 0 |
| ELEMENTS: | 48 | 6503697 | 0 |
| EMCS: | 2 | 575600 | 0 |
| LOCKS: | | 1024 | |

Wyniki

Oblicz zmianę w użyciu rzeczywistej pamięci masowej przez zwiększenie pamięci masowej kontroli wymaganej do użycia systemu SCM.

- Przed dodaniem SCM do struktury, struktura ma następujące sumy, jak pokazano w [“Poprawiona wydajność-podstawowa konfiguracja”](#) na stronie 209:
 - 345,242 pozycje
 - 6,548,467 elementów
 - 780 318 EMCS
- Po dodaniu systemu SCM do struktury struktura zawiera następujące sumy:
 - 342,684 pozycji
 - 6,503,697 elementów
 - 575600 EMCS

Na podstawie tych danych liczbowych, po dodaniu systemu SCM, struktura jest zmniejszana o:

- 2558 wpisów
- 44 770 elementów
- 204,718 EMCS

Ilość pamięci masowej struktury, która jest używana do zarządzania systemem SCM, jest następująca dla struktury 2 GB z 4 GB przydzielonego systemu SCM:

$$(2558 + 44,770 + 204,718) * 256 = 61.5 \text{ MB}$$

Należy zauważyć, że dodanie większej liczby elementów SCM prawdopodobnie spowoduje tylko marginalne zmniejszenie wielkości struktury, ponieważ ilość pamięci masowej kontroli używanej do śledzenia wzrostów SCM, zarówno jako wielkość struktury, jak i ilość przydzielonych wzrostów SCM.

Co dalej

Powtórz testy opisane w końcowej sekcji sekcji [“Poprawiona wydajność-podstawowa konfiguracja”](#) na stronie 209.

Wyniki skorygowanej aplikacji można wykreślić w czasie. Porównując wykres z wcześniej uzyskanym, otrzymujesz teraz dane wyjściowe bez fali saw-tooth, ponieważ aplikacja nie musi już czekać na częściowe opróżnianie kolejki.

Więcej informacji na ten temat zawiera sekcja [MP16: WebSphere MQ for z/OS -planowanie i strojenie mocy obliczeniowej](#).

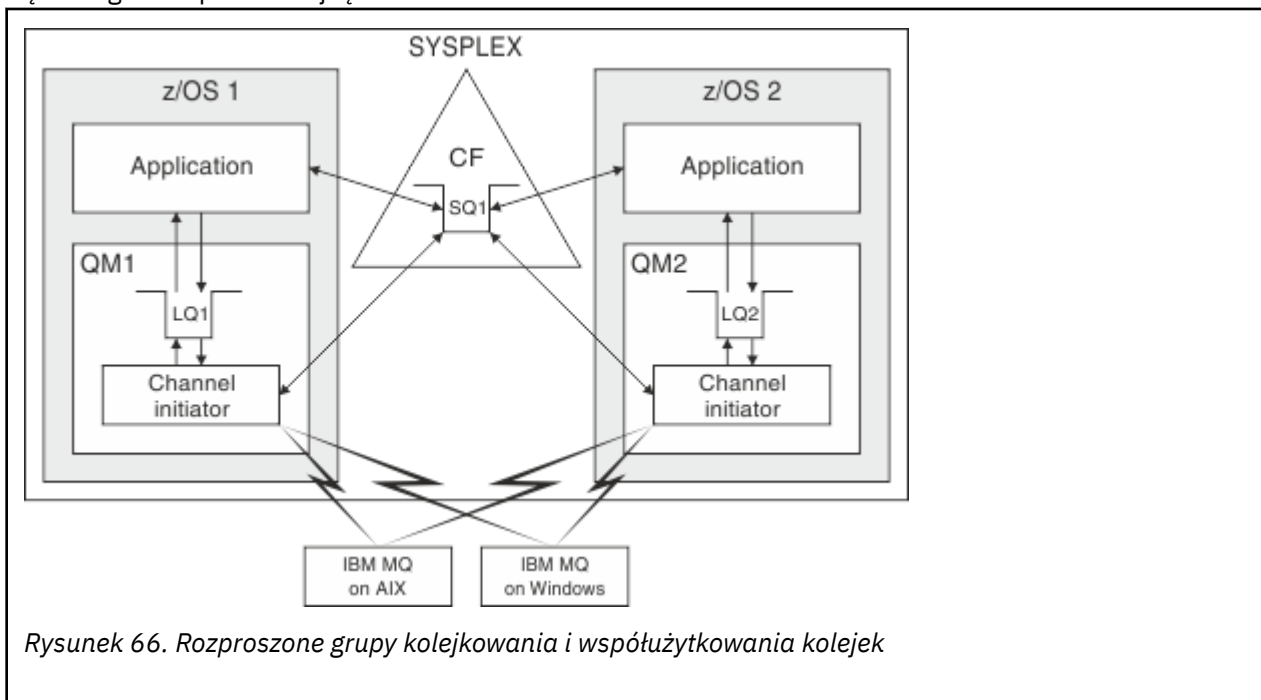
z/OS Rozproszone grupy kolejkowania i współużytkowania kolejek

Rozproszone grupy kolejkowania i współużytkowania kolejek to dwie techniki, których można użyć w celu zwiększenia dostępności systemów aplikacji. W tym temacie znajdują się dalsze informacje na temat tych technik.

Aby uzupełnić wysoką dostępność komunikatów w kolejkach współużytkowanych, komponent kolejkowania rozproszonego produktu IBM MQ udostępnia dodatkowe funkcje, które zapewniają:

- Wyższa dostępność do sieci.
- Zwiększona pojemność dla połączeń sieciowych przychodzących do grupy współużytkowania kolejek.

Rysunek 66 na stronie 215 przedstawia rozproszone grupy kolejkowania i współużytkowania kolejek. Wyświetlane są dwa menedżery kolejek w ramach sysplexu, które należą do tej samej grupy współużytkowania kolejek. Mogą oni uzyskać dostęp do kolejki współużytkowanej SQ1. Menedżery kolejek w sieci (na przykład w systemach AIX i Windows) mogą umieszczać komunikaty w tej kolejce za pośrednictwem inicjatora kanału menedżera kolejek. Sklonowane aplikacje w obu menedżerach kolejek są obsługiwane przez kolejkę.



Rysunek 66. Rozproszone grupy kolejkowania i współużytkowania kolejek

Pojęcia pokrewne

[“Kanały współużytkowane”](#) na stronie 216

Ten temat zawiera informacje o pojęciach związanych z kanałami współużytkowanymi i ich używaniem w produkcie IBM MQ for z/OS.

[“Kolejkowanie wewnątrz grupy”](#) na stronie 218

Kolejkowanie wewnątrz grupy umożliwia przesyłanie komunikatów między menedżerami kolejek w grupie współużytkowania kolejek.

“Klasy i grupy współużytkowania kolejek” na stronie 219

W tym temacie opisano sposób korzystania z grup współużytkowania kolejek z klastrami.

Kanały współużytkowane

Ten temat zawiera informacje o pojęciach związanych z kanałami współużytkowanymi i ich używaniem w produkcie IBM MQ for z/OS.

Wiele produktów sieciowych udostępnia mechanizm ukrywania awarii serwera w sieci lub równoważenia przychodzących żądań sieciowych między różnymi serwerami. Produkty sieciowe udostępniają *port ogólny* dla przychodzących żądań połączeń sieciowych, a żądanie przychodzące można spełnić, łącząc się z jednym z zakwalifikowanych serwerów.

Do tych produktów sieciowych należą:

- Zasoby ogólne VTAM
- Dystrybutor SYSPLEX

Inicjator kanału wykorzystuje te produkty do korzystania z możliwości kolejek współużytkowanych

Istnieją dwa typy kanałów współużytkowanych: *współużytkowany kanał przychodzący* i *współużytkowany kanał wychodzący*.

- [Współużytkowane kanały przychodzące](#)
- [Współużytkowane kanały wychodzące](#)

Aby uzyskać więcej informacji na temat kanałów, patrz

- [Podsumowanie kanału współużytkowanego](#)
- [Status kanału współużytkowanego](#)

Współużytkowane kanały przychodzące

Każdy inicjator kanału w grupie współużytkowania kolejek uruchamia dodatkowe zadanie nasłuchiwanie, które nasłuchuje na *porcie ogólnym*. Ten port ogólny jest udostępniany sieci przez jedną z technologii pomocniczych (VTAM, TCP/IP). Przychodzące żądania przyłączenia sieciowego do portu ogólnego są rozsyłane przez technologię sieciową do dowolnego programu nasłuchującego w grupie współużytkowania kolejek (QSG), który nasłuchuje na porcie ogólnym.

Kanał można uruchomić w inicjatorze kanału, do którego jest kierowane przyłączenie przychodzące, jeśli inicjator kanału ma dostęp do definicji kanału dla kanału o tej nazwie. Definicję kanału można zdefiniować jako prywatną dla menedżera kolejek lub jako składowaną we współużytkowanym repozytorium i dostępną w dowolnym miejscu (definicja globalna). Oznacza to, że można udostępnić definicję kanału w dowolnym inicjatorze kanału w grupie współużytkowania kolejek, definiując ją jako definicję globalną.

Istnieje dodatkowa różnica podczas uruchamiania kanału za pośrednictwem portu ogólnego. Synchronizacja kanału odbywa się z grupą współużytkowania kolejek, a nie z pojedynczym menedżerem kolejek. Załóżmy na przykład, że zdalny menedżer kolejek uruchamia kanał za pośrednictwem portu ogólnego. Przy pierwszym uruchomieniu kanału może on zostać uruchomiony w menedżerze kolejek QM1 i w przepływie komunikatów. Jeśli kanał zostanie zatrzymany i zrestartowany w menedżerze kolejek QM2, informacje o liczbie komunikatów, które przepływały, są nadal poprawne, ponieważ synchronizacja jest wykonywana z grupą współużytkowania kolejek.

Aby umieścić komunikaty w dowolnej kolejce, można użyć kanału danych przychodzących uruchomionego za pośrednictwem portu ogólnego. Zdalny menedżer kolejek nie wie, czy kolejka docelowa jest współużytkowana, czy nie. Jeśli kolejka docelowa jest kolejką współużytkowaną, zdalny menedżer kolejek łączy się za pośrednictwem dowolnego dostępnego inicjatora kanału w sposób z równoważeniem obciążenia, a komunikaty są umieszczane w kolejce współużytkowanej.

Jeśli kolejka docelowa jest kolejką prywatną, komunikaty są umieszczane w kolejce prywatnej, której właścicielem jest menedżer kolejek, z którym jest połączona bieżąca instancja kanału. W tym środowisku,

zwanym *replikowanymi kolejkami lokalnymi*, każdy menedżer kolejek musi mieć zdefiniowany ten sam zestaw kolejek prywatnych.

Konfigurowanie kanałów SVRCONN dla grupy współużytkowania kolejek

Optymalną konfiguracją dla kanałów SVRCONN w grupie współużytkowania kolejek jest skonfigurowanie prywatnych programów nasłuchujących w każdym CHINIT, które używają innego numeru portu od punktu do punktu. Te porty nasłuchiwania są następnie używane jako zasoby zaplecza dla nowego mechanizmu dystrybucji obciążenia, takiego jak Sysplex Distributor korzystający z wirtualnych adresów IP (VIPA). Zewnętrzny adres VIPA jest następnie używany jako adres docelowy dla definicji CLNTCONN w sieci. Kanał SVRCONN można zdefiniować za pomocą komendy QSGDISP (GROUP), więc ta sama definicja jest dostępna dla wszystkich menedżerów kolejek w grupie QSG. Taka konfiguracja pozwala uniknąć korzystania ze współużytkowanego programu nasłuchującego, a tym samym zmniejsza wpływ na wydajność grupy współużytkowania kolejek utrzymującej stan współużytkowanego kanału, który nie jest wymagany dla kanałów klient/serwer.

Współużytkowane kanały wychodzące

Kanał wychodzący jest uważany za kanał współużytkowany, jeśli przyjmuje komunikaty ze współużytkowanej kolejki transmisji. Jeśli jest współużytkowana, przechowuje informacje o synchronizacji na poziomie grupy współużytkowania kolejki. Oznacza to, że kanał może zostać zrestartowany w innej instancji menedżera kolejek i inicjatora kanału w grupie współużytkowania kolejek, jeśli podsystem komunikacyjny, inicjator kanału lub menedżer kolejek ulegnie awarii. Restartowanie uszkodzonych kanałów w ten sposób jest funkcją współużytkowanych kanałów nazywaną *odtworzeniem kanału równorzędnego*.

Równoważenie obciążenia dla współużytkowanych kanałów wychodzących

Wychodzący kanał współużytkowany jest zakwalifikowany do uruchomienia w dowolnym inicjatorze kanału w grupie współużytkowania kolejek, jeśli nie określono, że ma on zostać uruchomiony w konkretnym inicjatorze kanału. Inicjator kanału wybrany przez program IBM MQ jest określany na podstawie następujących kryteriów:

- Czy podsystem komunikacyjny jest obecnie dostępny dla inicjatora kanału?
- Czy połączenie Db2 jest dostępne dla inicjatora kanału?
- Który inicjator kanału ma najniższe bieżące obciążenie? Obciążenie obejmuje kanały, które są aktywne i ponawiane.

Podsumowanie kanału współużytkowanego

Kanały współużytkowane różnią się od kanałów prywatnych w następujący sposób:

Kanał prywatny

Powiązany z pojedynczym inicjatorem kanału.

- Kanał wychodzący używa lokalnej kolejki transmisji.
- Kanał przychodzący został uruchomiony przez port lokalny.
- Informacje o synchronizacji przechowywane w systemie SYSTEM.CHANNEL.SYNCQ .

Kanał współużytkowany

Obciążenie zbalansowane z wysoką dostępnością.

- Kanał wychodzący używa współużytkowanej kolejki transmisji.
- Kanał przychodzący został uruchomiony przez port ogólny.
- Informacje o synchronizacji przechowywane w systemie SYSTEM.QSG.CHANNEL.SYNCQ .

Za pomocą opcji CHLDISP komendy START CHANNEL można określić, czy kanał jest prywatny, czy współużytkowany. Kanał współużytkowany można uruchomić, wyzwalając go w taki sam sposób, jak kanał

prywatny. Jednak po uruchomieniu kanału współużytkowanego produkt IBM MQ wykonuje równoważenie obciążenia i uruchamia kanał w najbardziej odpowiednim inicjatorze kanału w grupie współużytkowania kolejek. (W razie potrzeby można określić, że kanał współużytkowany ma być uruchamiany w konkretnym inicjatorze kanału).

Status kanału współużytkowanego

Inicjatory kanałów w grupie współużytkowania kolejek zachowują tabelę współużytkowanych statusów kanałów w programie Db2. Zapisuje, które kanały są aktywne w których inicjatorach kanałów. Tabela statusu kanału współużytkowanego jest używana w przypadku awarii inicjatora kanału lub systemu komunikacyjnego. Wskazuje, które kanały muszą zostać zrestartowane w innym inicjatorze kanału w grupie współużytkowania kolejek.

Kolejowanie wewnątrz grupy

Kolejkowanie wewnątrz grupy umożliwia przesyłanie komunikatów między menedżerami kolejek w grupie współużytkowania kolejek.

Istnieje możliwość szybkiego przesyłania komunikatów między menedżerami kolejek w grupie współużytkowania kolejek bez definiowania kanałów. W ten sposób używana jest kolejka systemowa o nazwie SYSTEM.QSG.TRANSMIT.QUEUE, która jest współużytkowaną kolejką transmisji. Każdy menedżer kolejek w grupie współużytkowania kolejek uruchamia zadanie o nazwie wewnątrzgrupowy agent kolejkowania, który oczekuje na przybycie komunikatów do tej kolejki, które są przeznaczone dla ich menedżera kolejek. Po wykryciu takiego komunikatu jest on usuwany z kolejki i umieszczany w odpowiedniej kolejce docelowej.

Używane są standardowe reguły rozstrzygania nazw, ale jeśli włączono kolejkowanie wewnątrz grupy (IGQ), a docelowy menedżer kolejek znajduje się w grupie współużytkowania kolejek, to SYSTEM.QSG.TRANSMIT.QUEUE służy do przesyłania komunikatu do poprawnego docelowego menedżera kolejek, a nie do kolejki transmisji i kanału.

Istnieje możliwość włączenia kolejkowania wewnątrz grupy za pomocą atrybutu menedżera kolejek. Kolejowanie wewnątrz grupy przenosi nietrwałe komunikaty poza punkt synchronizacji oraz komunikaty trwałe w punkcie synchronizacji. Jeśli w kolejce znajduje się problem z dostarczeniem komunikatów do kolejki docelowej, kolejkowanie wewnątrz grupy próbuje umieścić je w kolejce niedostarczonych komunikatów. Jeśli kolejka niedostarczonych komunikatów jest pełna lub niezdefiniowana, komunikaty nietrwałe są usuwane, ale komunikaty trwałe są wycofane i zwracane do systemu SYSTEM.QSG.TRANSMIT.QUEUE, a agent IGQ próbuje dostarczyć komunikaty, dopóki nie powiedzie się.

Przechodzący kanał współużytkowany, który odbiera komunikat przeznaczony dla kolejki w innym menedżerze kolejek w grupie współużytkowania kolejek, może używać kolejkowania wewnątrz grupy do *przeskoku* komunikatu do właściwego miejsca docelowego.

Czasami lokalny menedżer kolejek może umieścić komunikat bezpośrednio w kolejce docelowej, jeśli kolejka docelowa jest kolejką współużytkowaną, a nie komunikat, który najpierw jest przesyłany do docelowego menedżera kolejek. Do sterowania tym elementem można użyć atrybutu SQQMNAME menedżera kolejek. Jeśli wartość parametru SQQMNAME zostanie ustawiona na USE, komenda MQOPEN zostanie wykonana w menedżerze kolejek określonym za pomocą nazwy ObjectQMgr. Jeśli jednak kolejka docelowa jest kolejką współużytkowaną, a wartość parametru SQQMNAME zostanie ustawiona na IGNORE, a nazwa ObjectQMgr jest nazwą innego menedżera kolejek w grupie współużytkowania kolejek, to kolejka współużytkowana jest otwierana w lokalnym menedżerze kolejek. Jeśli lokalny menedżer kolejek nie może otworzyć kolejki docelowej lub umieścić komunikat w kolejce, komunikat jest przesyłany do określonej nazwy ObjectQMgrza pomocą interfejsu IGQ lub kanału MQ .

Kolejkowanie wewnątrz grupy (IGQ) obsługuje duże komunikaty, największa wartość to 100 MB *minus* długości nagłówka kolejki transmisji.

Jeśli ta opcja jest używana, użytkownicy muszą mieć taki sam dostęp do kolejek w każdym menedżerze kolejek w grupie współużytkowania kolejki.

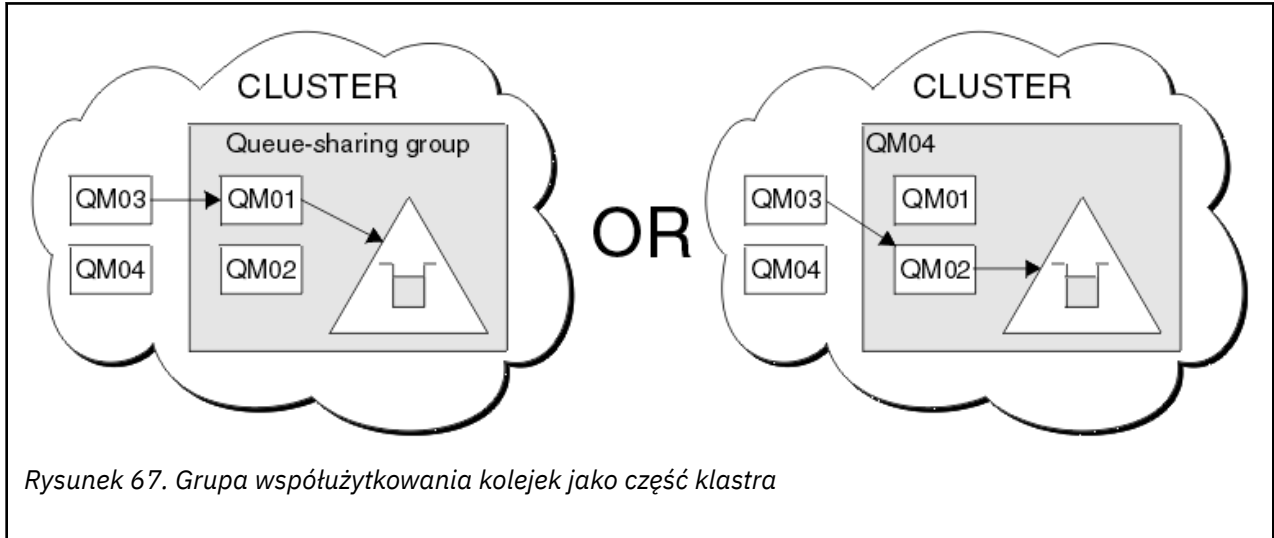
z/OS **Klasy i grupy współużytkowania kolejek**

W tym temacie opisano sposób korzystania z grup współużytkowania kolejek z klastrami.

W pojedynczej definicji można udostępnić współużytkowane kolejki do klastra. W tym celu należy określić nazwę klastra podczas definiowania kolejki współużytkowanej.

Użytkownicy w sieci widzą kolejkę współużytkowaną jako udostępnianą przez każdego menedżera kolejek w ramach grupy współużytkowania kolejek (kolejka współużytkowana nie jest reklamowana jako udostępniana przez grupę współużytkowania kolejek). Klienci mogą uruchamiać sesje z dowolnymi członkami grupy współużytkowania kolejek w celu umieszczania komunikatów w tej samej współużytkowanej kolejce.

Rysunek 67 na stronie 219 pokazuje, w jaki sposób członkowie klastra mogą uzyskać dostęp do współużytkowanej kolejki za pośrednictwem dowolnego członka grupy współużytkowania kolejek.



z/OS **Wpływanie na dystrybucję obciążenia z kolejkami współużytkowanymi**

W tym temacie opisano czynniki, które mają wpływ na dystrybucję obciążenia z kolejkami współużytkowanymi w grupie współużytkowania kolejek.

Produkt IBM MQ nie udostępnia funkcji równoważenia obciążenia dla współużytkowanych kolejek. W *modzie typu pull* można jednak wpływać na dystrybucję obciążenia w grupie współużytkowania kolejek (QSG). Wybór, którego usługi menedżera kolejek jest kolejka (odbiera komunikat zapisany do kolejki współużytkowanej), ma wpływ na dostępną moc przetwarzania każdego menedżera kolejek w grupie współużytkowania kolejek i cele zarządzania obciążeniem zdefiniowane w sysplexie.

Jednak należy docenić, że menedżer kolejek, który wykonuje operację MQPUT komunikatu, może również mieć duży wpływ na decydowanie, który menedżer kolejek otrzyma komunikat.

Menedżer kolejek lokalnych jest bardziej prawdopodobne, aby wykonać operację MQGET

W przypadku aplikacji wykonujących operację MQPUT lokalny menedżer kolejek jest menedżerem kolejek, z którym połączona jest aplikacja.

Następujące uwagi mają wpływ na to, do których usługi menedżera kolejek są wykonywane MQPUT komunikatu, wykonując operację MQGET w imieniu aplikacji pobierających.

Gdy komunikat jest umieszczany w pustej współużytkowanej kolejce, lokalny menedżer kolejek jest zwykle księgowany przed powiadomieniem dowolnego innego menedżera kolejek w grupie współużytkowania kolejki. Jeśli lokalny menedżer kolejek jest w stanie przetworzyć komunikat, otrzymuje powiadomienie o przejściu listy z narzędzia CF (Coupling Facility) przed każdym innym menedżerem

kolejek w QSG. (Powiadomienie o przejściu listy jest powiadomieniem, że kolejka współużytkowana zmieniła stan z pustej na niepustą).

Możliwe scenariusze, w tym przypadku, są następujące:

1. Wywołanie MQPUT nietrwałego komunikatu z punktu synchronizacji i *szybkie umieszczanie w oczekiwaniu na proces pobierający*.

Jeśli istnieje aplikacja z *MQGET with wait* w lokalnym menedżerze kolejek dla kolejki, to wywołanie MQPUT komunikatu jest przekazywane bezpośrednio do buforu aplikacji pobierających i nie jest zapisywane w kolejce. Jest to prawda w przypadku kolejek współużytkowanych i niewspółużytkowanych. Ta funkcja jest często nazywana mechanizmem *fast put to a waiting getter*. W przypadku kolejek współużytkowanych żaden inny menedżer kolejek w QSG nie jest powiadamiany, ponieważ nie ma przejścia z pustego do niepustego kolejki. Oznacza to na przykład, że ten menedżer kolejek może serwisować wszystkie operacje umieszczania z tej aplikacji i zakładając, że żadne inne aplikacje nie umieszczają komunikatów w kolejce, a żaden inny menedżer kolejek w grupie współużytkowania kolejek nie pomaga w przeciąganiu tej kolejki. Jeśli jednak nie ma komendy MQGET z oczekiwaniem na menedżer kolejek lokalnych, a komunikat jest umieszczany w kolejce współużytkowanej, system CF będzie powiadamiał inne menedżery kolejek w grupie współużytkowania kolejki zgodnie z jej regułami dotyczącymi powiadomień o przejściach listy.

2. MQPUT dla komunikatu trwałego lub komunikatu w punkcie synchronizacji.

W takim przypadku, jeśli istnieje aplikacja z *MQGET with wait* w lokalnym menedżerze kolejek, komunikat jest umieszczany w kolejce współużytkowanej, a system CF powiadamia inne menedżery kolejek w grupie współużytkowania kolejek zgodnie z jej regułami dotyczącymi powiadomień o przejściach listy. Jednak lokalny menedżer kolejek nie czeka na powiadomienie o przejściu z systemu CF, ale honoruje najpierw lokalną komendę *MQGET z oczekiwaniem* i zwykle wykonuje operację pobrania tego komunikatu w imieniu aplikacji, zanim jakkolwiek inny menedżer kolejek w grupie współużytkowania kolejek może odpowiedzieć na powiadomienie systemu CF. Jest to zależne od tego, jak zajęty jest lokalny menedżer kolejek. W przeciwnym razie dowolny menedżer kolejek zgłoszony przez system CF w związku z nadejściem komunikatu w pustej kolejce będzie próbował wykonać pierwsze działanie. Pierwszy menedżer kolejek, który ma odpowiedzieć na nowe komunikaty, przetwarza nowy komunikat.

3. Na koniec, jeśli kolejka nie jest przeciągana z komunikatów, a system CF wysłał powiadomienie o zmianie stanu z pustego na niepusty dla kolejki, wszystkie połączone menedżery kolejek będą miały możliwość asystowania podczas przetwarzania kolejki. W tym zdarzeniu obciążenie jest określone jako *oparte na pobierze*.

Ta konstrukcja pozwala na zwiększenie wydajności przy użyciu rozkładu obciążenia opartego wyłącznie na ciągach typu pull. Celem jest skorzystanie z wysokiej dostępności oferowanej przez kolejki znajdujące się w systemie CF, umożliwiając menedżerowi kolejek, w miarę możliwości, wykonanie operacji MQGET bez konieczności odwoływania się do systemu CF i w celu jak najefektywniejszego przetwarzania obciążenia komunikatu.

Możliwe jest przyjęcie alternatywnych metod, w których nacisk na równoważenie obciążenia jest ważniejszy niż poprzednio opisane udoskonalenia wydajności. Na przykład, upewniając się, że żadna z aplikacji pobierających nie jest połączona z tym samym menedżerem kolejek, z którym połączona jest aplikacja umieszczanie. Przy użyciu tego projektu wszystkie komunikaty są umieszczane w kolejce, a wszystkie menedżery kolejek w QSG są powiadamiane, gdy kolejka jest przenoszona z pustej do niepustej, zgodnie z algorytmem CF do obsługi takich przejść. Ponadto mechanizm *fast put to waiting getter* nie ma zastosowania.

Gdzie znaleźć więcej informacji o współużytkowanych kolejkach i grupach współużytkowania kolejek

Tabela w tym temacie zawiera więcej informacji na temat sposobu, w jaki produkt IBM MQ for z/OS używa współużytkowanych kolejek i grup współużytkowania kolejek.

Tabela 18. Gdzie znaleźć więcej informacji o współużytkowanych kolejkach i grupach współużytkowania kolejek

| Temat | Gdzie szukać |
|---|---|
| Odtwarzanie grupy współużytkowania kolejki | “Odtwarzanie i restartowanie w systemie z/OS” na stronie 262 |
| Zabezpieczenia grupy współużytkowania kolejki | “Pojęcia dotyczące zabezpieczeń w produkcie IBM MQ for z/OS” na stronie 280 |
| Definicje obiektów prywatnych i globalnych Kierowanie komend do innej kolejki menedżery | Wykonywanie komend |
| Planowanie narzędzia CF środowisko | Definiowanie zasobów narzędzia CF |
| Planowanie środowiska SMDS | Planowanie środowiska współużytkowanego zestawu danych komunikatów (SMDS) |
| Planowanie Środowisko Db2 | Planowanie środowiska Db2 |
| Konfigurowanie współużytkowanych kolejek Parametry systemu | “Współużytkowane kolejki i grupy współużytkowania kolejek” na stronie 172 |
| Programy narzędziowe Migracja kolejek | Korzystanie z programów narzędziowych IBM MQ |
| Komunikaty konsoli | Komunikaty produktu IBM MQ for z/OS |
| Komendy MQSC | Komendy MQSC |
| IBM MQKlastry | Konfigurowanie klastra menedżera kolejek |
| IBM MQ kolejkowanie rozproszone Nazwy kanałów | Wprowadzenie do zarządzania kolejkami rozproszonymi |
| Pisanie aplikacji | Przegląd projektu aplikacji |
| Wywołanie MQCONN | MQCONN |

z/OS

Kolejkowanie wewnątrz grupy

W tej sekcji opisano kolejkowanie wewnątrz grupy, a funkcja IBM MQ for z/OS jest unikalna w stosunku do platformy z/OS. Ta funkcja jest dostępna tylko dla menedżerów kolejek zdefiniowanych w grupie współużytkowania kolejek.

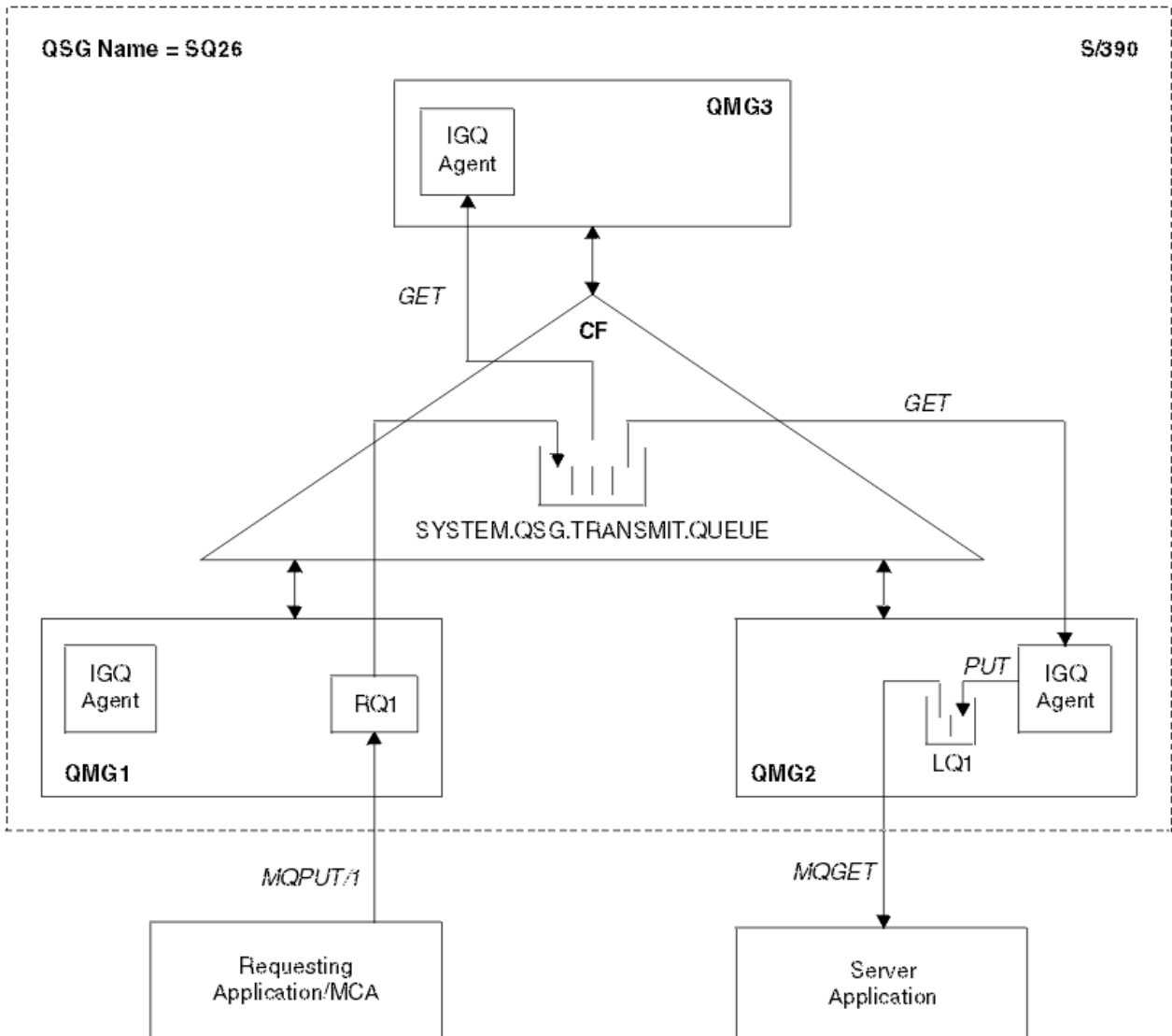
Więcej informacji na temat grup współużytkowania kolejek zawiera sekcja [“Współużytkowane kolejki i grupy współużytkowania kolejek” na stronie 172](#).

z/OS Pojęcia związane z kolejkowaniem wewnątrz grupy

Kolejkowanie wewnątrz grupy może być używane do dostarczania, bardziej wydajnych, małych komunikatów do kolejek znajdujących się w zdalnych menedżerach kolejek w ramach grupy współużytkowania kolejek.

Kolejkowanie wewnątrz grupy (IGQ) może spowodować potencjalnie szybkie i mniej kosztowne małe przesyłanie komunikatów między menedżerami kolejek w grupie współużytkowania kolejek (QSG), bez potrzeby definiowania kanałów między menedżerami kolejek.

Na poniższym diagramie przedstawiono typowy przykład kolejkowania wewnątrz grupy.



Rysunek 68. Przykład kolejkowania wewnątrz grupy

Diagram przedstawia:

- Agenty IGQ działające na trzech menedżerach kolejek (QM1, QM2i QM3), które są zdefiniowane w grupie współużytkowania kolejek o nazwie SQ26.
- Współużytkowana kolejka transmisji SYSTEM.QSG.TRANSMIT.QUEUE , która jest zdefiniowana w narzędziu CF.
- Definicja kolejki zdalnej, która jest zdefiniowana w menedżerze kolejek QM1.
- Kolejka lokalna, która jest zdefiniowana w menedżerze kolejek QM2.

- Żądająca aplikacja (ta aplikacja może być agentem kanału komunikatów (Message Channel Agent-MCA)), która jest połączona z menedżerem kolejek QMG1.
- Aplikacja serwera, która jest połączona z menedżerem kolejek QMG2.
- Komunikat żądania jest umieszczany w systemie SYSTEM.QSG.TRANSMIT.QUEUE.

Kolejkowanie wewnątrz grupy i agent kolejkowania wewnątrz grupy

Agent IGQ jest uruchamiany podczas inicjowania menedżera kolejek. Gdy aplikacje otwierają i umieszczają komunikaty w kolejkach zdalnych, lokalny menedżer kolejek określa, czy kolejkowanie wewnątrz grupy jest używane do przesyłania komunikatów. Jeśli ma być używana kolejkowanie wewnątrz grupy, lokalny menedżer kolejek umieszcza komunikat w systemie SYSTEM.QSG.TRANSMIT.QUEUE. Agent IGQ w docelowym zdalnym menedżerze kolejek pobiera komunikat i umieszcza go w kolejce docelowej.

Terminologia kolejkowania wewnątrz grupy

Wyjaśnienia dotyczące terminologii: kolejkowanie wewnątrz grupy, kolejka transmisji współużytkowanej do użycia przez kolejkowanie wewnątrz grupy oraz agent kolejkowania wewnątrz grupy.

Kolejkowanie wewnątrz grupy

Kolejkowanie wewnątrz grupy może spowodować potencjalnie szybkie i mniej kosztowne przesyłanie komunikatów między menedżerami kolejek w grupie współużytkowania kolejki, bez potrzeby definiowania kanałów.

Współużytkowana kolejka transmisji do użycia przez kolejkowanie wewnątrz grupy

Każda grupa współużytkowania kolejek ma współużytkowaną kolejkę transmisji o nazwie SYSTEM.QSG.TRANSMIT.QUEUE, która jest używana przez kolejkowanie wewnątrz grupy. Jeśli kolejkowanie wewnątrz grupy jest włączone, SYSTEM.QSG.TRANSMIT.QUEUE pojawia się w ścieżce rozstrzygnięcia nazw podczas otwierania kolejek zdalnych. Gdy aplikacje (w tym agenty kanału komunikatów (Message Channel Agents-MCAs)) umieszczają komunikaty w kolejce zdalnej, lokalny menedżer kolejek określa kwalifikowanie komunikatów do szybkiego przesyłania i umieszcza je w systemie SYSTEM.QSG.TRANSMIT.QUEUE.

Wewnątrzgrupowy agent kolejkowania

Agent IGQ jest to zadanie uruchomione przy inicjowaniu menedżera kolejek, które oczekuje na dotarcie odpowiednich komunikatów do SYSTEM.QSG.TRANSMIT.QUEUE. Agent IGQ pobiera odpowiednie komunikaty z tej kolejki i dostarcza je do kolejek docelowych.

Agent IGQ dla każdego menedżera kolejek jest zawsze uruchamiany, ponieważ kolejkowanie wewnątrz grupy jest używane przez sam menedżer kolejek na potrzeby własnego przetwarzania wewnętrznego.

Korzyści wynikające z kolejkowania wewnątrz grupy

Korzyści wynikające z kolejkowania wewnątrz grupy to: zmniejszenie definicji systemu, zmniejszenie administrowania systemem, zwiększenie wydajności, obsługa migracji i dostarczanie komunikatów w przypadku wielu operacji przesyłania między menedżerami kolejek w grupie współużytkowania kolejek.

Korzyści wynikające z kolejkowania wewnątrz grupy są następujące:

Zredukowane definicje systemów

Kolejkowanie wewnątrz grupy usuwa konieczność definiowania kanałów między menedżerami kolejek w grupie współużytkowania kolejek.

Zredukowana administracja systemem

Ponieważ w grupie współużytkowania kolejek nie ma zdefiniowanych kanałów między menedżerami kolejek, nie ma wymagania dotyczącego administrowania kanałami.

Większa wydajność

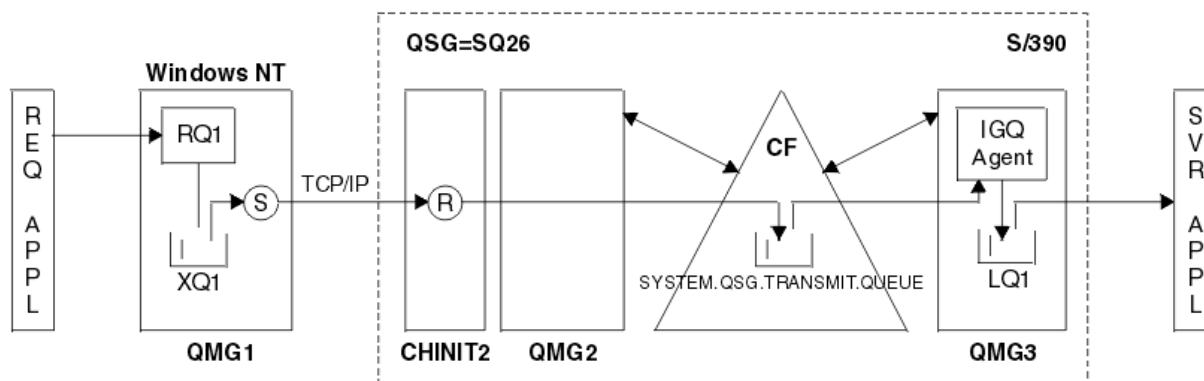
Ponieważ istnieje tylko jeden agent IGQ potrzebny do dostarczenia komunikatu do kolejki docelowej (zamiast dwóch pośrednich agentów wysyłających i odbierających), dostarczanie komunikatów przy użyciu kolejkowania wewnątrz grupy może być mniej kosztowne niż dostarczanie komunikatów za pomocą kanałów. W kolejkowaniu wewnątrzgrupowym istnieje tylko komponent odbierający, ponieważ usunięto zapotrzebowanie na komponent wysyłający. Jest to spowodowane tym, że komunikat jest dostępny dla agenta IGQ w docelowym menedźerze kolejek w celu dostarczenia go do kolejki docelowej po zakończeniu operacji put w lokalnym menedźerze kolejek, a w przypadku komunikatów umieszczonych w zasięgu punktu synchronizacji-zatwierdzone.

Obsługa migracji

Aplikacje zewnętrzne względem grupy współużytkowania kolejek mogą dostarczać komunikaty do kolejki rezydujących w dowolnym menedźerze kolejek w grupie współużytkowania kolejek, a jednocześnie są połączone tylko z określonym menedźerem kolejek w grupie współużytkowania kolejek. Dzieje się tak dlatego, że komunikaty docierające do kanału odbiorczego, przeznaczone dla kolejki w zdalnym menedźerze kolejek, mogą być przesyłane w sposób przezroczysty do kolejki docelowej przy użyciu kolejkowania wewnątrz grupy. Narzędzie to umożliwia wdrażanie aplikacji między grupą współużytkowania kolejek bez konieczności zmiany systemów zewnętrznych względem grupy współużytkowania kolejek.

Typowa konfiguracja jest ilustrowana na poniższym diagramie, w którym:

- Aplikacja żądająca połączona z menedźerem kolejek QMG1 musi wysłać komunikat do kolejki lokalnej w menedźerze kolejek QMG3.
- Menedźer kolejek QMG1 jest połączony tylko z menedźerem kolejek QMG2.
- Menedżery kolejek QMG2 i QMG3, które były wcześniej połączone za pomocą kanałów, są teraz elementami grupy współużytkowania kolejek SQ26.



Rysunek 69. Przykład obsługi migracji

Przebieg operacji jest następujący:

1. Aplikacja żądająca umieszcza komunikat, który jest przeznaczony dla kolejki lokalnej LQ1 w zdalnym menedźerze kolejek QMG3, do definicji kolejki zdalnej RQ1.
2. Menedźer kolejek QMG1, działający na stacji roboczej Windows NT, umieszcza komunikat w kolejce transmisji XQ1.
3. Nadawca MCA (S) na QM1 przesyła komunikat, używając protokołu TCP/IP, do odbiornika MCA (R) na inicjatorze kanału CHINIT2.
4. Odbiornik MCA (R) w inicjatorze kanału CHINIT2 umieszcza komunikat w kolejce transmisji współużytkowanej SYSTEM.QSG.TRANSMIT.QUEUE.
5. Agent IGQ w menedźerze kolejek QMG3 pobiera komunikat z systemu SYSTEM.QSG.TRANSMIT.QUEUE i umieszcza ją w docelowej kolejce lokalnej LQ1.
6. Aplikacja serwera pobiera komunikat z docelowej kolejki lokalnej i przetwarza ją.

Dostarczanie komunikatów podczas wielu operacji wyszukiwania między menedżerami kolejek w grupie współużytkowania kolejek

Poprzedni diagram w sekcji [Obsługa migracji](#) ilustruje również dostarczanie komunikatów podczas wielu operacji między menedżerami kolejek w grupie współużytkowania kolejek. Komunikaty przychodzące do menedżera kolejek w ramach grupy współużytkowania kolejek, ale przeznaczone do kolejki w innym menedżerze kolejek w grupie współużytkowania kolejek, mogą być łatwo przestane do kolejki docelowej w docelowym menedżerze kolejek przy użyciu kolejkowania wewnątrz grupy.

z/OS Ograniczenia kolejkowania wewnątrz grupy

Ograniczenia kolejkowania wewnątrz grupy to: komunikaty zakwalifikowane do transferu z użyciem kolejkowania wewnątrz grupy, liczba agentów kolejkowania wewnątrz grupy na menedżera kolejek oraz uruchamianie i zatrzymywanie agenta kolejkowania wewnątrz grupy.

W tej sekcji opisano ograniczenia kolejkowania wewnątrz grupy.

Komunikaty zakwalifikowane do transferu z użyciem kolejkowania wewnątrz grupy

Ponieważ kolejkowanie wewnątrz grupy korzysta ze współużytkowanej kolejki transmisji zdefiniowanej w narzędziu CF (CF), kolejkowanie wewnątrz grupy jest ograniczone do dostarczania komunikatów o maksymalnej obsługiwanej długości komunikatów dla kolejek współużytkowanych pomniejszonej o długość nagłówka kolejki transmisji (MQXQH).

Liczba wewnątrzgrupowych agentów kolejkowania dla menedżera kolejek

Dla każdego menedżera kolejek w grupie współużytkowania kolejki uruchomiono tylko jeden agent IGQ.

Uruchamianie i zatrzymywanie agenta kolejkowania wewnątrz grupy

Agent IGQ jest uruchamiany podczas inicjowania menedżera kolejek i kończy działanie w trakcie wyłączania menedżera kolejek. Jest on przeznaczony do długotrwałego, samoregeneracji (w przypadku nieprawidłowego zakończenia pracy), zadania. W przypadku wystąpienia błędu w definicji SYSTEM.QSG.TRANSMIT.QUEUE (na przykład, jeśli ta kolejka jest zablokowana), agent IGQ będzie ponawiać próby. Jeśli agent IGQ napotka błąd, który powoduje normalne zakończenie działania agenta, gdy menedżer kolejek jest nadal aktywny, można go zrestartować, wydając komendę ALTER QMGR IGQ (ENABLED). Ta komenda pozwala uniknąć konieczności ponownego przetwarzania menedżera kolejek.

Ustawianie atrybutu IGQ dla menedżera kolejek na WŁĄCZONY lub WYŁĄCZONY

Jeśli atrybut IGQ menedżera kolejek jest ustawiony na wartość ENABLED lub DISABLED, to istniejące uchwytów obiektów mogą zostać unieważnione z kodem przyczyny MQRC_OBJECT_CHANGED. Więcej informacji na ten temat zawiera sekcja [Pierwsze kroki z kolejkowaniem wewnątrz grupy](#).

z/OS Pierwsze kroki z kolejkowaniem wewnątrz grupy

Kolejkowanie wewnątrz grupy można włączyć, wyłączyć i używać w sposób opisany w tym temacie.

Włączanie kolejkowania wewnątrz grupy

Aby włączyć kolejkowanie wewnątrz grupy w menedżerach kolejek, należy wykonać następujące czynności:

- Zdefiniuj współużytkowaną kolejkę transmisji o nazwie SYSTEM.QSG.TRANSMIT.QUEUE. Definicję tej kolejki można znaleźć w pliku thlqual.SCSQPROC(CSQ4INSS), a przykład CSQINP2 dla obiektów SYSTEM dla grup współużytkowania kolejek. Ta kolejka musi być zdefiniowana z poprawnymi atrybutami, jak określono w pliku thlqual.SCSQPROC(CSQ4INSS), aby kolejkowanie wewnątrz grupy działało poprawnie.
- Ponieważ agent IGQ jest zawsze uruchamiany przy inicjowaniu menedżera kolejek, kolejkowanie wewnątrz grupy jest zawsze dostępne na potrzeby przetwarzania komunikatów przychodzących. Agent IGQ przetwarza wszystkie komunikaty umieszczone w systemie SYSTEM.QSG.TRANSMIT.QUEUE. Aby jednak włączyć kolejkowanie wewnątrz grupy dla przetwarzania danych wychodzących, atrybut IGQ menedżera kolejek musi być ustawiony na wartość ENABLED.

Ważne: Jeśli atrybut IGQ menedżera kolejek jest ustawiony na wartość ENABLED, istniejące uchwyty obiektów mogą zostać unieważnione z kodem przyczyny MQR_OBJECT_CHANGED. Więcej informacji na ten temat zawiera sekcja [“Właściwości specyficzne dla kolejkowania wewnątrz grupy” na stronie 233](#) . Zgodnie z opisem w sekcji "Odpowiedzi programisty" dla tego kodu przyczyny aplikacje muszą być kodowane w celu obsługi tej sytuacji (więcej informacji na ten temat zawiera sekcja [2041 \(07F9\) \(RC2041\): MQR_OBJECT_CHANGED](#) .

Dodatkowo, ponieważ IGQ jest zaprojektowana jako długotrwałe i samowystarczalne zadanie, które rozpoczyna się podczas inicjowania i kończy pracę z zamknięciem, patrz [“Ograniczenia kolejkowania wewnątrz grupy” na stronie 225](#) , aby uzyskać dodatkowe informacje.

Wyłączenie kolejkowania wewnątrz grupy

Aby wyłączyć kolejkowanie wewnątrz grupy dla przesyłania komunikatów wychodzących, należy ustawić atrybut IGQ dla menedżera kolejek na WYŁĄCZONY. Jeśli kolejkowanie wewnątrz grupy jest wyłączone dla konkretnego menedżera kolejek, agent IGQ w tym menedżerze kolejek może przetwarzać komunikaty przychodzące, które zostały umieszczone w systemie SYSTEM.QSG.TRANSMIT.QUEUE przez menedżer kolejek, w którym włączono kolejkowanie wewnątrz grupy dla przesyłania danych wychodzących.

Ważne: Jeśli atrybut IGQ menedżera kolejek jest ustawiony na wartość ENABLED, istniejące uchwyty obiektów mogą zostać unieważnione z kodem przyczyny MQR_OBJECT_CHANGED. Więcej informacji na ten temat zawiera sekcja [“Właściwości specyficzne dla kolejkowania wewnątrz grupy” na stronie 233](#) . Zgodnie z opisem w sekcji "Odpowiedzi programisty" dla tego kodu przyczyny aplikacje muszą być kodowane w celu obsługi tej sytuacji (więcej informacji na ten temat zawiera sekcja [2041 \(07F9\) \(RC2041\): MQR_OBJECT_CHANGED](#) .

Dodatkowo, ponieważ IGQ jest zaprojektowana jako długotrwałe i samowystarczalne zadanie, które rozpoczyna się podczas inicjowania i kończy pracę z zamknięciem, patrz [“Ograniczenia kolejkowania wewnątrz grupy” na stronie 225](#) , aby uzyskać dodatkowe informacje.

Korzystanie z kolejkowania wewnątrz grupy

Gdy kolejkowanie wewnątrz grupy jest włączone, jest ono dostępne do użycia, a menedżer kolejek używa go, gdy tylko jest to możliwe. Oznacza to, że gdy aplikacja wstawi komunikat do definicji kolejki zdalnej, do pełnej kolejki zdalnej lub do kolejki klastra, menedżer kolejek określa, czy komunikat może zostać dostarczony przy użyciu kolejkowania wewnątrz grupy, a jeśli jest, umieszcza komunikat w systemie SYSTEM.QSG.TRANSMIT.QUEUE. Nie ma potrzeby zmieniania aplikacji użytkownika ani kolejek aplikacji, ponieważ w przypadku komunikatów kwalifikowanych menedżer kolejek korzysta z systemu SYSTEM.QSG.TRANSMIT.QUEUE, preferowana względem każdej innej kolejki transmisji.

Konfiguracje kolejkowania wewnątrz grupy

Oprócz typowej konfiguracji kolejkowania wewnątrz grupy, możliwe są inne konfiguracje.

Rysunek 68 na stronie 222 opisuje typową konfigurację.

Pojęcia pokrewne

[“Rozproszona kolejkowanie z kolejkowaniem wewnątrz grupy \(wiele ścieżek dostarczania\)” na stronie 227](#)
W przypadku aplikacji, które przetwarzają krótkie komunikaty, można skonfigurować kolejkowanie wewnątrz grupy tylko w celu dostarczania komunikatów między menedżerami kolejek w grupie współużytkowania kolejek.

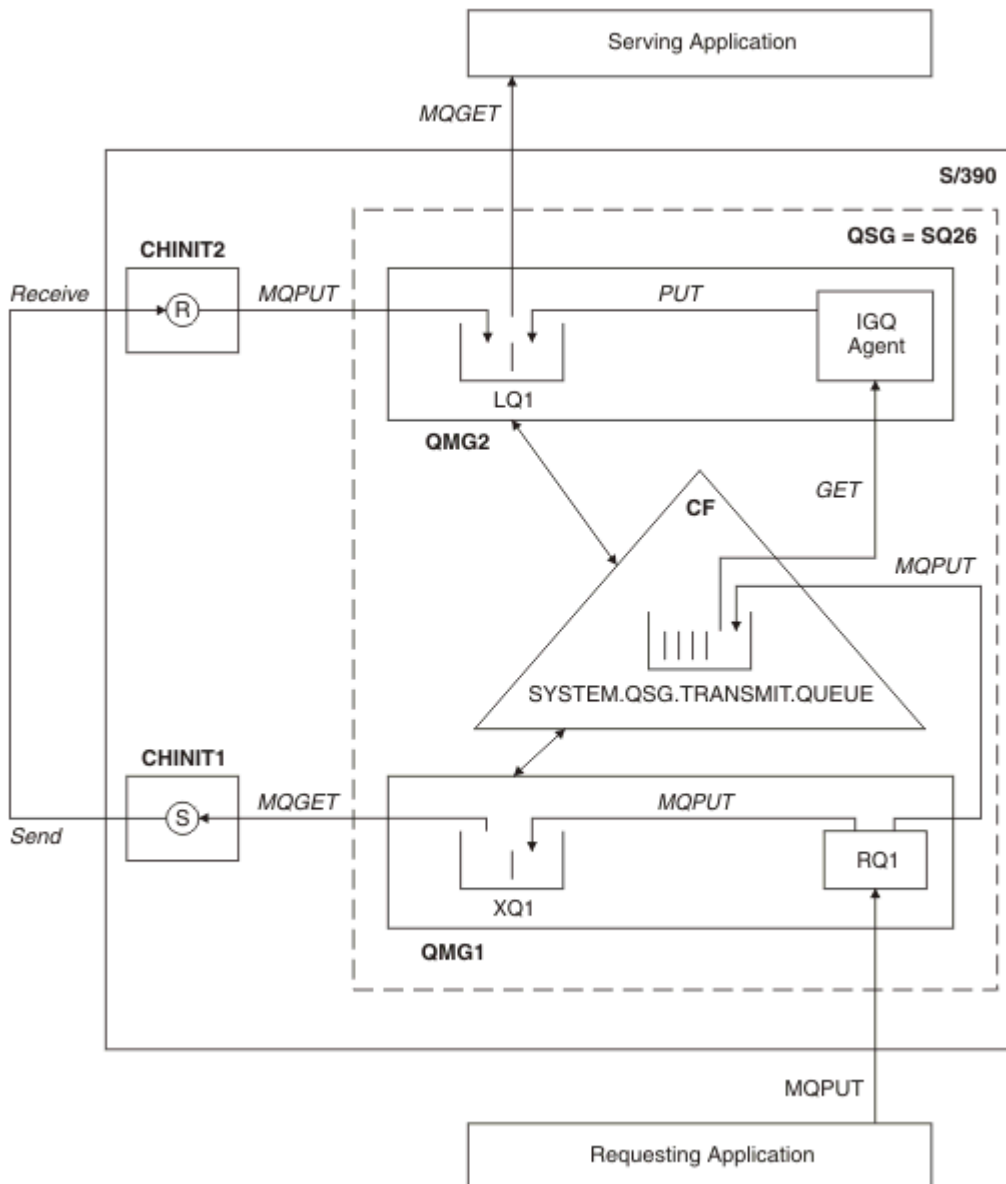
[“Łączenie w klastry z kolejkowaniem wewnątrz grupy \(wiele ścieżek dostarczania\)” na stronie 228](#)
Możliwe jest skonfigurowanie menedżerów kolejek w taki sposób, aby były one w klastrze, a także w grupie współużytkowania kolejek.

[“Łączenie w klastry, kolejkowanie wewnątrz grupy i kolejkowanie rozproszone” na stronie 231](#)
Istnieje możliwość skonfigurowania menedżera kolejek, który jest elementem klastra, a także grupy współużytkowania kolejek i jest połączony z rozproszonym menedżerem kolejek przy użyciu pary kanału nadawca/odbiorcę.

z/OS Rozproszona kolejkowanie z kolejkowaniem wewnątrz grupy (wiele ścieżek dostarczania)

W przypadku aplikacji, które przetwarzają krótkie komunikaty, można skonfigurować kolejkowanie wewnątrz grupy tylko w celu dostarczania komunikatów między menedżerami kolejek w grupie współużytkownika kolejek.

Wybór kolejkowania wewnątrz grupy w komunikacji kanału może być kontrolowany przez poziom CFSTRUCT. (3 zamiast 4 lub 5). Maksymalna długość komunikatu ustawiona w polu SYSTEM.QSQ.TRANSMIT.QUEUE.



Rysunek 70. Przykładowa konfiguracja

Otwórz/umieść przetwarzanie

1. Należy pamiętać, że gdy aplikacja żądająca otwiera kolejkę zdalną RQ1, rozstrzygnięcie nazw występuje zarówno dla niewspółużytkowanej kolejki transmisji XQ1, jak i do współużytkowanej kolejki transmisji SYSTEM.QSQ.TRANSMIT.QUEUE.
2. Gdy aplikacja żądająca wstawi komunikat do kolejki zdalnej, w zależności od tego, czy kolejkowanie wewnątrz grupy jest włączone dla przesyłania danych wychodzących w menedżerze kolejek,

jak i w charakterystyce komunikatu, komunikat jest umieszczany w kolejce transmisji XQ1 lub w kolejce transmisji SYSTEM.QSG.TRANSMIT.QUEUE. Menedżer kolejek umieszcza wszystkie duże komunikaty w kolejce transmisji XQ1, a wszystkie małe komunikaty w kolejce transmisji SYSTEM.QSG.TRANSMIT.QUEUE.

3. Jeśli kolejka transmisji XQ1 jest pełna lub nie jest dostępna, żądania umieszczenia dużych komunikatów są niepoprawne synchronicznie z odpowiednim kodem powrotu i kodem przyczyny. Jednak żądania umieszczenia małych komunikatów są nadal pomyślne i są umieszczane w kolejce transmisji SYSTEM.QSG.TRANSMIT.QUEUE.
4. W przypadku kolejki transmisji SYSTEM.QSG.TRANSMIT.QUEUE jest pełna lub nie może być wstawiona, żądania umieszczenia małych komunikatów kończą się synchronicznie, z odpowiednim kodem powrotu i kodem przyczyny. Jednak żądania umieszczenia dużych komunikatów są kontynuowane i są umieszczane w kolejce transmisji XQ1. W tym przypadku nie jest podejmowana żadna próba umieszczenia małych komunikatów w kolejce transmisji.

Przeptyw dla dużych komunikatów

1. Aplikacja żądająca umieszcza duże komunikaty w kolejce zdalnej RQ1.
2. Menedżer kolejek QMG1 umieszcza komunikaty w kolejce transmisji XQ1.
3. Nadawca MCA (S) w menedżerze kolejek QMG1 pobiera komunikaty z kolejki transmisji XQ1 i wysyła je do menedżera kolejek QMG2.
4. Odbiornik MCA (R) w menedżerze kolejek QMG2 odbiera komunikaty i umieszcza je w kolejce docelowej LQ1.
5. Aplikacja służy do pobierania, a następnie przetwarzania komunikatów z kolejki LQ1.

Przeptyw małych komunikatów

1. Aplikacja żądająca umieszcza małe komunikaty w kolejce zdalnej RQ1.
2. Menedżer kolejek QMG1 umieszcza komunikaty w kolejce transmisji SYSTEM.QSG.TRANSMIT.QUEUE.
3. IGQ w menedżerze kolejek QMG2 pobiera komunikaty i umieszcza je w kolejce docelowej LQ1.
4. Aplikacja obsługując pobiera komunikaty z kolejki LQ1.

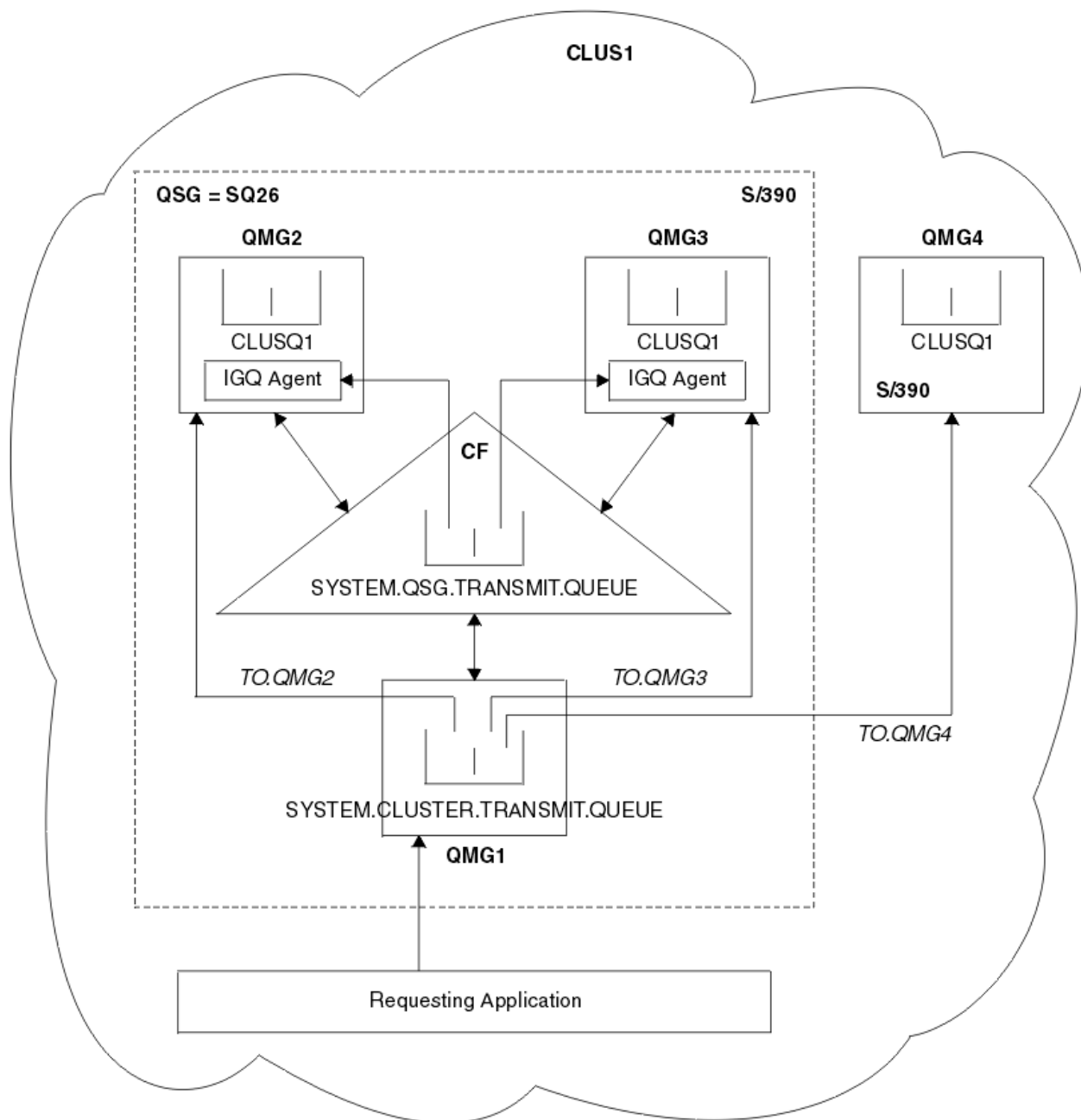
Punkty do nota

1. Aplikacja żądająca nie musi mieć informacji o mechanizmie bazowym używanym do dostarczania komunikatów.
2. Potencjalnie szybszy mechanizm dostarczania komunikatów może zostać osiągnięty w przypadku małych komunikatów.
3. Dla dostarczania komunikatów dostępnych jest wiele ścieżek (czyli normalna trasa kanału i trasa kolejkowania wewnątrz grupy).
4. Trasa kolejkowania wewnątrz grupy, która jest potencjalnie szybsza, jest wybierana w preferencjach do normalnej trasy kanału. W zależności od charakterystyki komunikatu dostarczenie komunikatu może być podzielone na dwie ścieżki. Z tego powodu komunikaty mogą być dostarczane poza kolejnością (ta dostawa jest również możliwa, jeśli komunikaty są dostarczane tylko przy użyciu zwykłej trasy kanału).
5. Gdy trasa została wybrana, a komunikaty zostały umieszczone w kolejkach transmisji, tylko wybrana trasa jest używana do dostarczania komunikatów. Wszystkie nieprzetworzone komunikaty w systemie SYSTEM.QSG.TRANSMIT.QUEUE nie jest przekierowana do kolejki transmisji XQ1.

z/OS Łączenie w klastry z kolejkowaniem wewnątrz grupy (wiele ścieżek dostarczania)

Możliwe jest skonfigurowanie menedżerów kolejek w taki sposób, aby były one w klastrze, a także w grupie współużytkownika kolejek.

Gdy komunikaty są wysyłane do kolejki klastra, a lokalne i zdalne menedżery kolejek docelowe znajdują się w tej samej grupie współużytkowania kolejek, kolejkowanie wewnątrz grupy jest używane do dostarczania małych komunikatów (przy użyciu systemu `SYSTEM.QSG.TRANSMIT.QUEUE`) oraz dostarczanie dużych komunikatów, jeśli kolejkowanie wewnątrz grupy obsługuje wielkość komunikatu. Ponadto `SYSTEM.CLUSTER.TRANSMIT.QUEUE` jest używana do dostarczania komunikatów do dowolnego menedżera kolejek znajdującego się w klastrze, ale poza grupą współużytkowania kolejek. Poniższy diagram ilustruje tę konfigurację (inicjatory kanału nie są wyświetlane).



Rysunek 71. Przykład grupowania z kolejkowaniem wewnątrz grupy

Diagram przedstawia:

- Cztery menedżery kolejek z/OS QMG1, QMG2, QMG3i QMG4 skonfigurowane w klastrze CLUS1.
- Menedżery kolejek QMG1, QMG2i QMG3 skonfigurowane w grupie współużytkowania kolejek SQ26.
- Agenty IGQ działają w menedżerach kolejek QMG2 i QMG3.
- Lokalny `SYSTEM.CLUSTER.TRANSMIT.QUEUE` zdefiniowana w QMG1.

Uwaga: Dla jasności: SYSTEM.CLUSTER.TRANSMIT.QUEUE w pozostałych menedżerach kolejek nie są wyświetlane.

- Współużytkowana SYSTEM.QSG.TRANSMIT.QUEUE zdefiniowana w systemie CF, która znajduje się w strukturze IBM MQ skonfigurowanej z atrybutem CFLEVEL (3) RECOVER (YES).
- Kanały klastra TO.QMG2 (łączenie QMG1 z QMG2), TO.QMG3 (łączenie QMG1 z QMG3) i TO.QMG4 (łączenie QMG1 z QMG4).
- Kolejka klastra CLUSQ1 jest obsługiwana w menedżerach kolejek QMG2, QMG3i QMG4.

Założmy, że aplikacja żądająca otwiera kolejkę klastra z opcją MQOO_BIND_NOT_FIXED, tak aby docelowy menedżer kolejek dla kolejki klastra został wybrany w czasie umieszczania.

Jeśli wybrany docelowy menedżer kolejek ma wartość QMG2:

- Wszystkie duże komunikaty wprowadzone przez aplikację żądają to:
 - Umieść w systemie SYSTEM.CLUSTER.TRANSMIT.QUEUE w systemie QMG1, ponieważ SYSTEM.QSG.TRANSMIT.QUEUE znajduje się w strukturze CFLEVEL (3); w związku z tym obsługuje komunikaty o wielkości do 63 kB.
 - Przesłano do kolejki klastra CLUSQ1 na serwerze QMG2 za pomocą kanału klastra TO.QMG2
- Wszystkie małe wiadomości umieszczone przez aplikację żądają
 - Umieść w systemie współużytkowanej kolejki transmisji SYSTEM.QSG.TRANSMIT.QUEUE. Ta kolejka znajduje się w strukturze skonfigurowanej z atrybutem RECOVER (YES), dlatego jest używana zarówno dla trwałych, jak i nietrwałych komunikatów.
 - Pobrane przez agenta IGQ w systemie QMG2
 - Umieść w kolejce klastra CLUSQ1 w QMG2

Jeśli wybrany docelowy menedżer kolejek ma wartość QMG4:

- Ponieważ element QMG4 nie jest elementem grupy współużytkowania kolejki SQ26, wszystkie komunikaty wprowadzone przez aplikację żądają
 - Umieść w systemie SYSTEM.CLUSTER.TRANSMIT.QUEUE w QMG1
 - Przesłano do kolejki klastra CLUSQ1 na serwerze QMG4 przy użyciu kanału klastra TO.QMG4

Punkty do nota

- Aplikacja żądająca nie musi mieć informacji o mechanizmie bazowym używanym do dostarczania komunikatów.
- Potencjalnie szybszy mechanizm dostarczania jest osiąganym na potrzeby przesyłania małych nietrwałych komunikatów między menedżerami kolejek w grupie współużytkowania kolejek (nawet wtedy, gdy te same menedżery kolejek znajdują się w klastrze).
- Dla dostarczania komunikatów dostępnych jest wiele ścieżek (to znaczy zarówno trasa klastra, jak i trasa kolejkowania wewnątrz grupy).
- Trasa kolejkowania wewnątrz grupy, która jest potencjalnie szybsza, jest wybierana w preferencjach do trasy klastra. W zależności od charakterystyki komunikatu dostarczenie komunikatu może być podzielone na dwie ścieżki. Z tego powodu komunikaty mogą być dostarczane poza kolejnością. Ważne jest, aby pamiętać, że ta dostawa jest możliwa bez względu na opcję MQOO_BIND_ * określoną przez aplikację. Kolejkowanie wewnątrz grupy powoduje dystrybuowanie komunikatów w taki sam sposób, jak w przypadku łączenia w klastry, w zależności od tego, czy określono w niej wartości MQOO_BIND_NOT_FIXED, MQOO_BIND_ON_OPEN, MQOO_BIND_ON_GROUP czy MQOO_BIND_AS_Q_DEF.
- Gdy trasa została wybrana, a komunikaty zostały umieszczone w kolejkach transmisji, tylko wybrana trasa jest używana do dostarczania komunikatów. Wszystkie nieprzetworzone komunikaty w systemie SYSTEM.QSG.TRANSMIT.QUEUE nie są przekierowane do SYSTEM.CLUSTER.TRANSMIT.QUEUE.

Łączenie w klastry, kolejkowanie wewnątrz grupy i kolejkowanie

rozproszone

Istnieje możliwość skonfigurowania menedżera kolejek, który jest elementem klastra, a także grupy współużytkowania kolejek i jest połączony z rozproszonym menedżerem kolejek przy użyciu pary kanału nadawca/odbiorcę.

Ta konfiguracja jest kombinacją rozproszonej kolejkowania z kolejkowaniem wewnątrzgrupowym i łączenia w klastry z kolejkowaniem wewnątrz grupy.

Kolejkowanie wewnątrz grupy jest opisane w sekcji [“Rozproszona kolejkowanie z kolejkowaniem wewnątrz grupy \(wiele ścieżek dostarczania\)”](#) na stronie 227.

Grupowanie z kolejkowaniem wewnątrz grupy jest opisane w sekcji [“Łączenie w klastry z kolejkowaniem wewnątrz grupy \(wiele ścieżek dostarczania\)”](#) na stronie 228.

Komunikaty kolejkowania wewnątrz grupy

W tej sekcji opisano komunikaty wprowadzone w systemie SYSTEM.QSG.TRANSMIT.QUEUE.

Struktura komunikatu

Podobnie jak wszystkie inne komunikaty, które są umieszczane w kolejkach transmisji, komunikaty umieszczane w systemie SYSTEM.QSG.TRANSMIT.QUEUE jest poprzedzona przedrostkiem kolejki transmisji (MQXQH).

Trwałość komunikatu

W produkcie IBM WebSphere MQ 5.3 i nowszych w kolejkach współużytkowanych obsługiwane są zarówno komunikaty trwałe, jak i nietrwałe.

Jeśli menedżer kolejek zakończy działanie, gdy agent IGQ przetwarza komunikaty nietrwałe lub jeśli agent IGQ zakończy działanie w sposób nieprawidłowy, podczas gdy w trakcie przetwarzania komunikatów przetwarzane komunikaty nietrwałe są przetwarzane, może zostać utracone. Aplikacje muszą wykonywać uzgodnienia dotyczące odtwarzania nietrwałych komunikatów, jeśli ich odtwarzanie jest wymagane.

Jeśli żądanie umieszczenia komunikatu nietrwałego, wysłane przez agenta IGQ, nieoczekiwanie nie powiedzie się, przetwarzany komunikat zostaje utracony.

Dostarczanie wiadomości

Agent IGQ pobiera i dostarcza wszystkie nietrwałe komunikaty poza zasięgiem punktu synchronizacji, a także wszystkie komunikaty trwałe w zasięgu punktu synchronizacji. W tym przypadku agent IGQ działa jako koordynator punktu synchronizacji. Dlatego agent IGQ przetwarza komunikaty nietrwałe, takie jak szybkie, nietrwałe komunikaty są przetwarzane w kanale komunikatów. Więcej informacji zawiera sekcja [Szybkie, nietrwałe komunikaty](#).

Przetwarzanie wsadowe komunikatów

Agent IGQ używa stałej wielkości zadania wsadowego 50 komunikatów. Wszystkie komunikaty trwałe pobrane w ramach zadania wsadowego są zatwierdzane w odstępach czasu 50 komunikatów. Agent zatwierdza partię składającą się z trwałych komunikatów, gdy nie ma więcej komunikatów dostępnych do pobrania w systemie SYSTEM.QSG.TRANSMIT.QUEUE.

Wielkość komunikatu

Maksymalna wielkość komunikatu, którą można umieścić w systemie SYSTEM.QSG.TRANSMIT.QUEUE to maksymalna obsługiwana długość komunikatu dla kolejek współużytkowanych pomniejszona o długość nagłówka kolejki transmisji (MQXQH).

Domyślna trwałość komunikatu i domyślny priorytet komunikatu

Jeśli SYSTEM.QSG.TRANSMIT.QUEUE znajduje się w ścieżce rozstrzygnięcia nazw kolejek, która została ustanowiona w czasie otwarcia, a następnie dla komunikatów umieszczonych z domyślną trwałością i domyślnym priorytetem (lub z domyślną trwałością lub priorytetem domyślnym), reguły normalne są stosowane przy wyborze kolejki, która ma domyślne wartości priorytetu i trwałości, które są używane. Więcej informacji na temat reguł wyboru kolejki można znaleźć w sekcji [Komunikaty produktu IBM MQ](#).

Pojęcia pokrewne

“Niedostarczone/nieprzetworzone komunikaty” na stronie 232

W tej sekcji opisano, co dzieje się z niedostarczonymi i nieprzetworzonymi komunikatami w systemie SYSTEM.QSG.TRANSMIT.QUEUE.

“Komunikaty raportów-Intra Group Queuing” na stronie 232

W tym temacie opisano komunikaty raportu: potwierdzenie przybycia, potwierdzenie dostarczenia, raport o utracie ważności oraz raport o wyjątkach.

Niedostarczone/nieprzetworzone komunikaty

W tej sekcji opisano, co dzieje się z niedostarczonymi i nieprzetworzonymi komunikatami w systemie SYSTEM.QSG.TRANSMIT.QUEUE.

Jeśli agent IGQ nie może dostarczyć komunikatu do kolejki docelowej, agent IGQ:

- Honoruje opcję raportu MQRO_DISCARD_MSG (jeśli pole opcji raportu MQMD dla niedostarczanego komunikatu wskazuje, że musi on być) i usuwa niedostarczone komunikaty.
- Próba umieszczenia niedostarczanego komunikatu w kolejce niedostarczanych komunikatów dla docelowego menedżera kolejek, jeśli komunikat nie został jeszcze usunięty. Agent IGQ prefikuje komunikat z nagłówkiem kolejki niewystanych wiadomości (MQDLH).

Jeśli kolejka niewystanych wiadomości nie została zdefiniowana lub jeśli niedostarczone komunikaty nie mogą zostać umieszczone w kolejce niedostarczanych komunikatów, a niedostarczone komunikaty są następujące:

- trwały, agent IGQ wycofuje bieżącą partię trwałych komunikatów, które przetwarza, a następnie wprowadza stan ponawiania. Więcej informacji na ten temat zawiera [“Właściwości specyficzne dla kolejkowania wewnątrz grupy”](#) na stronie 233.
- nietrwały, agent IGQ usuwa komunikat i kontynuuje przetwarzanie następnego komunikatu.

Jeśli menedżer kolejek w grupie współużytkowania kolejek jest przerwany, zanim powiązany z nim agent IGQ będzie miał czas na przetworzenie wszystkich komunikatów, nieprzetworzone komunikaty pozostają w systemie SYSTEM.QSG.TRANSMIT.QUEUE do momentu następnego uruchomienia menedżera kolejek. Następnie agent IGQ pobiera i dostarcza komunikaty do kolejek docelowych.

Jeśli narzędzie CF nie powiedzie się przed wszystkimi komunikatami w systemie SYSTEM.QSG.TRANSMIT.QUEUE zostały przetworzone, wszystkie nieprzetworzone nietrwałe komunikaty zostaną utracone.

Produkt IBM zaleca, aby aplikacje nie umieszczały komunikatów bezpośrednio w kolejkach transmisji. Jeśli aplikacja przekaże komunikaty bezpośrednio do systemu SYSTEM.QSG.TRANSMIT.QUEUE, agent IGQ może nie być w stanie przetworzyć tych komunikatów i pozostają w systemie SYSTEM.QSG.TRANSMIT.QUEUE. Użytkownicy muszą następnie używać własnych metod do obsługi tych nieprzetworzonych komunikatów.

Komunikaty raportów-Intra Group Queuing

W tym temacie opisano komunikaty raportu: potwierdzenie przybycia, potwierdzenie dostarczenia, raport o utracie ważności oraz raport o wyjątkach.

Potwierdzenie przybycia (COA) /potwierdzenie dostarczenia (COD) komunikatów

Komunikaty COA i COD są generowane przez menedżer kolejek, gdy używana jest kolejkowanie wewnątrz grupy.

Komunikaty raportu utraty ważności

Komunikaty raportu utraty ważności są generowane przez menedżer kolejek.

Komunikaty o wyjątkach

W zależności od opcji raportu MQRO_EXCEPTION_* określonej w polu *Opcje raportu* deskryptora komunikatu dla niedostarczanego komunikatu agent IGQ generuje wymagany raport o wyjątkach i umieszcza go w określonej kolejce odpowiedzi. Kolejowanie wewnątrz grupy może zostać użyte do dostarczenia raportu o wyjątku do docelowej kolejki odpowiedzi.

Trwałość komunikatu raportu jest taka sama, jak trwałość niedostarczanego komunikatu. Jeśli agent IGQ nie rozwiąże nazwy docelowej kolejki odpowiedzi lub jeśli nie powiedzie się umieszczenie komunikatu odpowiedzi w kolejce transmisji (w celu późniejszego przestania do docelowej kolejki odpowiedzi), próbuje on umieścić raport o wyjątku w kolejce niedostarczanej kolejki menedżera kolejek, w którym generowany jest komunikat raportu. Jeśli nie jest to możliwe, jeśli niedostarczone komunikaty są następujące:

- trwały, agent IGQ usuwa raport o wyjątku, tworzy kopię zapasową bieżącej partii komunikatów i wprowadza stan ponawiania. Więcej informacji na ten temat zawiera sekcja [“Właściwości specyficzne dla kolejkowania wewnątrz grupy”](#) na stronie 233.
- nietrwały, agent IGQ usuwa raport o wyjątkach i kontynuuje przetwarzanie następnego komunikatu w systemie SYSTEM.QSG.TRANSMIT.QUEUE.

Zabezpieczenia kolejkowania wewnątrz grupy

W tej sekcji opisano mechanizmy zabezpieczeń w kolejkach wewnątrz grup.

Atrybuty menedżera kolejek IGQAUT (uprawnienie IGQ) i IGQUSER (ID użytkownika agenta IGQ) mogą być ustawione w celu kontrolowania poziomu sprawdzania zabezpieczeń, który jest wykonywany, gdy agent IGQ otwiera kolejki docelowe.

Upewnienie kolejkowania wewnątrz grupy (IGQAUT)

Atrybut IGQAUT może być ustawiony w celu wskazania typu sprawdzeń zabezpieczeń, które mają zostać wykonane, a tym samym określenia identyfikatorów użytkowników, które mają być używane przez agenta IGQ podczas ustalania uprawnień do umieszczania komunikatów w kolejce docelowej.

Atrybut IGQAUT jest analogiczny do atrybutu PUTAUT, który jest dostępny w definicjach kanałów.

Identyfikator użytkownika kolejkowania wewnątrz grupy (IGQUSER)

Atrybut IGQUSER może być używany do wyznaczania ID użytkownika, który ma być używany przez agenta IGQ podczas tworzenia przez niego uprawnień do umieszczania komunikatów w kolejce docelowej.

Atrybut IGQUSER jest analogiczny do atrybutu MCAUSER, który jest dostępny w definicjach kanałów.

Właściwości specyficzne dla kolejkowania wewnątrz grupy

W tej sekcji opisano specyficzne właściwości kolejkowania wewnątrz grupy, w tym: unieważnianie uchwytów obiektów, samoodtworzenie i ponawianie możliwości agenta kolejkowania wewnątrz grupy oraz wewnątrzgrupowy agent kolejkowania i serializację.

Unieważnianie uchwytów obiektów (MQRC_OBJECT_CHANGED)

Jeśli znalezione atrybuty obiektu zostały zmienione po otwarciu obiektu, menedżer kolejek unieważnia uchwyt obiektu z opcją MQRC_OBJECT_CHANGED przy następnym użyciu.

Kolejkowanie wewnątrz grupy wprowadza następujące reguły unieważnienia uchwytu obiektu:

- Jeśli SYSTEM.QSG.TRANSMIT.QUEUE została włączona do ścieżki rozstrzygania nazw podczas przetwarzania otwartego, ponieważ kolejkowanie wewnątrz grupy było WŁĄCZONE w czasie otwarcia, ale w kolejkowaniu wewnątrzgrupowym znaleziono wartość DISABLED w momencie umieszczenia, a następnie menedżer kolejek unieważnia uchwyt obiektu i nie powiedzie się z żądaniem umieszczenia komendy MQRC_OBJECT_CHANGED przez żądanie umieszczenia.
- Jeśli SYSTEM.QSG.TRANSMIT.QUEUE nie została uwzględniona w ścieżce translacji nazw podczas przetwarzania otwartego, ponieważ kolejkowanie wewnątrz grupy było WYŁĄCZONE w czasie otwarcia, ale w czasie umieszczania w kolejkach wewnątrz grupy jest włączone kolejkowanie, a następnie menedżer kolejek unieważnia uchwyt obiektu i nie powiedzie się z żądaniem umieszczenia w obiekcie MQRC_OBJECT_CHANGED.
- Jeśli SYSTEM.QSG.TRANSMIT.QUEUE została włączona do ścieżki tłumaczenia nazw podczas przetwarzania otwartego, ponieważ kolejkowanie wewnątrz grupy było włączone w czasie

otwarcia, ale SYSTEM.QSG.TRANSMIT.QUEUE zmieniono czas umieszczenia, a następnie menedżer kolejek unieważnia uchwyt obiektu i nie powiedzie się z żądaniem umieszczenia w obiekcie MQRC_OBJECT_CHANGED.

Samodzielne odtwarzanie wewnątrzgrupowego agenta kolejkowania

Jeśli agent IGQ zakończy działanie nieprawidłowo, zostanie wyświetlony komunikat CSQM067E , a agent IGQ zostanie uruchomiony ponownie.

Możliwość ponawiania dla wewnątrzgrupowego agenta kolejkowania

Jeśli agent IGQ napotka problem z dostępem do SYSTEM.QSG.TRANSMIT.QUEUE (ponieważ nie jest ona zdefiniowana, na przykład lub jest zdefiniowana z niepoprawnymi atrybutami lub jest zablokowana dla pobrań lub z jakiegoś innego powodu), agent IGQ przechodzi w stan ponawiania.

Agent IGQ obserwuje krótkie i długie liczby ponowień i przedziały czasu. Wartości dla tych liczebności i przedziałów, których nie można zmienić, są następujące:

| Stała | Wartość |
|------------------------------------|----------------------|
| Licznik krótkookresowych ponowień | 10 |
| Interwał krótkookresowych ponowień | 60 sekund = 1 min |
| Licznik długookresowych ponowień | 999,999,999 |
| Interwał długookresowych ponowień | 1200 sekund = 20 min |

Wewnątrzgrupowy agent kolejkowania i szeregowanie

Próba przekształcenia przez agenta IGQ do postaci szeregowego dostępu do współużytkowanych kolejek podczas odtwarzania węzła sieci w dalszym ciągu może się nie powieść.

Jeśli w grupie współużytkowania kolejek wystąpi awaria menedżera kolejek, podczas gdy agent IGQ ma do czynienia z niezatwierdzanymi komunikatami w współużytkowanej kolejce lub kolejkach, agent IGQ kończy działanie, a odtwarzanie równorzędne kolejki współużytkowanej ma miejsce dla tego menedżera kolejek, w którym wystąpił błąd. Ponieważ odtwarzanie równorzędne kolejki współużytkowanej jest działaniem asynchronicznym, pozostawia ona możliwość uruchomienia menedżera kolejek, który uległ awarii, a także agenta IGQ dla tego menedżera kolejek, aby restartować przed ukończonym odtwarzaniem równorzędnych kolejek współużytkowanych. Co z kolei pozostawia możliwość przetwarzania wszystkich zatwierdzonych komunikatów przed i poza sekwencją z komunikatami, które nadal są odtwarzane. Aby zapewnić, że komunikaty nie są przetwarzane poza kolejnością, agent IGQ serializuje dostęp do kolejek współużytkowanych przez wywołanie wywołania MQCONN API.

Próba przekształcenia przez agenta IGQ do postaci szeregowego dostępu do współużytkowanych kolejek podczas odtwarzania węzła sieci w dalszym ciągu może się nie powieść. Wysyłany jest komunikat o błędzie, a agent IGQ jest w stanie ponawiania. Po zakończeniu odtwarzania równorzędnego menedżera kolejek, na przykład w czasie kolejnej ponowienia, agent IGQ może zostać uruchomiony.

z/OS

Zarządzanie pamięcią masową w systemie z/OS

Produkt IBM MQ for z/OS wymaga trwałych i tymczasowych struktur danych oraz używa zestawów stron i buforów pamięci do przechowywania tych danych. W tych tematach przedstawiono bardziej szczegółowe informacje na temat używania przez produkt IBM MQ tych zestawów stron i buforów.

Pojęcia pokrewne

[“Zestawy stron dla IBM MQ for z/OS” na stronie 235](#)

W tym temacie opisano, w jaki sposób produkt IBM MQ for z/OS używa zestawów stron do przechowywania komunikatów.

[“Klasy pamięci masowej dla IBM MQ for z/OS” na stronie 236](#)

Klasa pamięci masowej to koncepcja produktu IBM MQ for z/OS, która umożliwia menedżerowi kolejek odwzorowywać kolejki na zestawy stron. Klas pamięci masowej można użyć do sterowania, które zestawy danych są używane przez które kolejki.

[“Bufory i pule buforów dla IBM MQ for z/OS” na stronie 237](#)

Produkt IBM MQ for z/OS używa buforów i pul buforów do tymczasowego buforowania danych. W tym temacie opisano sposób organizacji buforów i ich wykorzystania.

Odsyłacze pokrewne

[“Gdzie znaleźć więcej informacji na temat zarządzania pamięcią masową dla produktu IBM MQ for z/OS” na stronie 239](#)

Ten temat zawiera informacje dotyczące znajdowania dalszych informacji na temat zarządzania pamięcią masową dla produktu IBM MQ for z/OS.

Zestawy stron dla IBM MQ for z/OS

W tym temacie opisano, w jaki sposób produkt IBM MQ for z/OS używa zestawów stron do przechowywania komunikatów.

Zestaw stron jest liniowym zestawem danych VSAM, który został specjalnie sformatowany do użycia przez produkt IBM MQ. Zestawy stron są używane do przechowywania większości komunikatów i definicji obiektów.

Wyjątkami są definicje globalne, które są przechowywane w repozytorium współużytkowanym w systemie Db2, a także komunikaty w kolejkach współużytkowanych. Nie są one zapisywane w zestawach stron menedżera kolejek. Więcej informacji na temat kolejek współużytkowanych zawiera sekcja [“Współużytkowane kolejki i grupy współużytkowania kolejek” na stronie 172](#), a także więcej informacji na temat definicji globalnych zawiera sekcja [Definicje prywatne i globalne](#).

Zestawy stron produktu IBM MQ mogą mieć wielkość do 64 GB. Każdy zestaw stron jest identyfikowany przez identyfikator zestawu stron (PSID), czyli liczbę całkowitą z zakresu od 00 do 99. Każdy menedżer kolejek musi mieć własne zestawy stron.

Produkt IBM MQ korzysta z zestawu stron zero (PSID=00) do przechowywania definicji obiektów i innych ważnych informacji istotnych dla menedżera kolejek. W przypadku normalnego działania produktu IBM MQ istotne jest, aby zestaw stron zero nie stał się pełny, więc nie należy używać go do przechowywania komunikatów.

Aby zwiększyć wydajność systemu, należy również oddzielić krótkotrłe komunikaty od komunikatów o długim czasie życia, umieszczając je w różnych zestawach stron.

Należy sformatować zestawy stron, a produkt IBM MQ udostępnia program narzędziowy FORMAT. Patrz sekcja [Formatowanie zestawów stron \(FORMAT\)](#). Zestawy stron muszą być również zdefiniowane w podsystemie IBM MQ.

Produkt IBM MQ for z/OS można skonfigurować w taki sposób, aby dynamicznie rozwijał zestaw stron, jeśli stanie się on pełny. Program IBM MQ kontynuuje rozszerzanie zestawu stron, jeśli jest to wymagane, aż do momentu istnienia 123 logicznych przydziałów, jeśli dostępna jest wystarczająca ilość miejsca na dysku. Obszary mogą obejmować woluminy, jeśli liniowy zestaw danych jest zdefiniowany w ten sposób, jednak produkt IBM MQ nie może rozwinąć zestawów stron powyżej 64 GB.

Nie można używać zestawów stron z jednego menedżera kolejek produktu IBM MQ w innym menedżerze kolejek produktu IBM MQ ani zmieniać nazwy menedżera kolejek. Jeśli dane mają być przesyłane z jednego menedżera kolejek do innego, należy usunąć wszystkie obiekty i komunikaty z pierwszego menedżera kolejek i przeladować je na inny.

Nie jest możliwe użycie zestawów stron większych niż 4 GB w menedżerze kolejek, w którym jest uruchomiona wersja wcześniejsza niż V6. W okresie migracji, gdy jest prawdopodobne, że konieczne może być powrót do poprzedniego wydania kodu:

- Nie zmieniaj zbioru stron 0 tak, aby był większy niż 4 GB.
- Inne zestawy stron większe niż 4 GB będą pozostawione w trybie bez połączenia podczas restartowania menedżera kolejek z poprzednią wersją.

Więcej informacji na temat migrowania istniejących zestawów stron, które mogą rozwinąć się poza 4 GB, zawiera sekcja [Definiowanie zestawu stron w celu zwiększenia ich wielkości niż 4 GB](#).

Administrator może dynamicznie dodawać zestawy stron do działającego menedżera kolejek lub usuwać zestawy stron z działającego menedżera kolejek (z wyjątkiem zerowego zestawu stron). Komenda DEFINE PSID może zostać uruchomiona po zakończeniu restartu menedżera kolejek, tylko wtedy, gdy komenda zawiera słowo kluczowe DSN.

Klasy pamięci masowej dla IBM MQ for z/OS

Klasa pamięci masowej to koncepcja produktu IBM MQ for z/OS, która umożliwia menedżerowi kolejek odwzorowywać kolejki na zestawy stron. Klas pamięci masowej można użyć do sterowania, które zestawy danych są używane przez które kolejki.

Wprowadzenie do klas pamięci masowych

Klasa pamięci masowej odwzorowuje jedną lub więcej kolejek na zestaw stron. Oznacza to, że komunikaty dla tej kolejki są zapisywane na tym zestawie stron.

Klasy pamięci masowej umożliwiają sterowanie miejscem, w którym przechowywane są niewspółużytkowane dane komunikatów w celach administracyjnych, związanych z zarządzaniem przestrzenią danych i zarządzaniem obciążeniem lub w celu odseparowania aplikacji. Klasy pamięci masowej można również użyć do zdefiniowania grupy XCF i nazwy elementu regionu IMS, jeśli używany jest most IMS (opisany w sekcji [“IBM MQ i IMS”](#) na stronie 296).

Kolejki współużytkowane nie korzystają z klas pamięci masowej w celu uzyskania odwzorowania zestawu stron, ponieważ komunikaty na nich nie są zapisywane w zestawach stron.

Sposób pracy klas pamięci

- Klasę pamięci masowej definiuje się za pomocą komendy DEFINE STGCLASS, określając identyfikator zestawu stron (PSID).
- Po zdefiniowaniu kolejki należy określić klasę pamięci w atrybucie STGCLASS.

W poniższym przykładzie kolejka lokalna QE5 jest odwzorowywana na zestaw stron 21 przy użyciu klasy pamięci masowej ARC2.

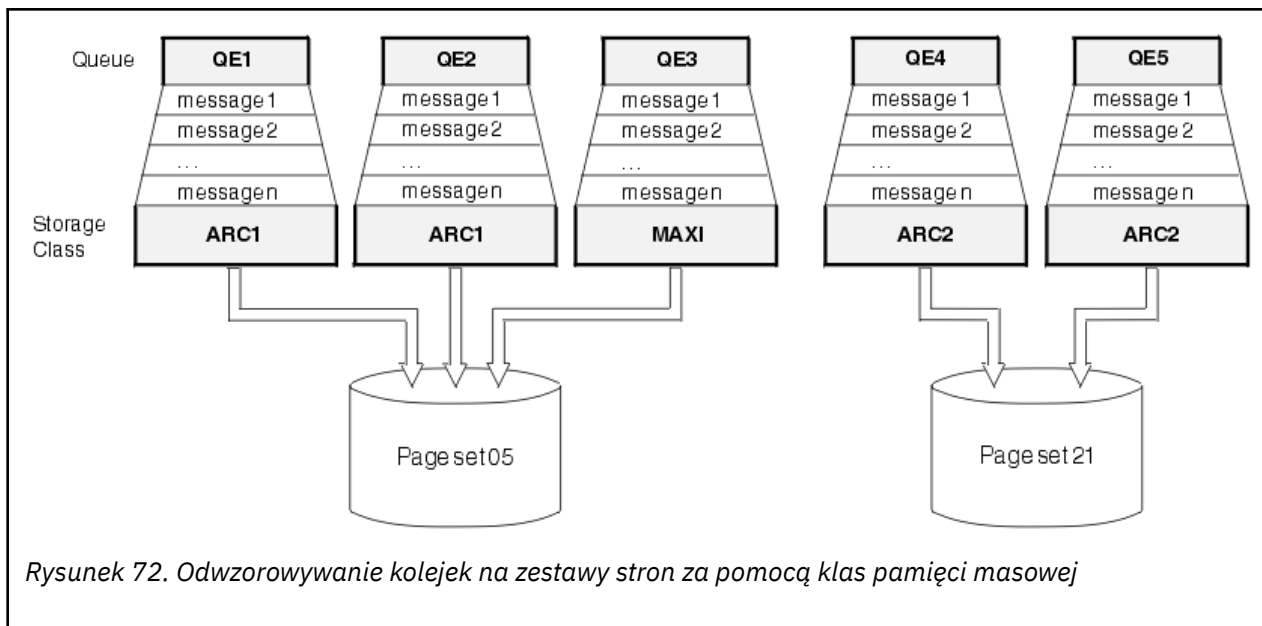
```
DEFINE STGCLASS(ARC2) PSID(21)
DEFINE QLOCAL(QE5) STGCLASS(ARC2)
```

Oznacza to, że komunikaty umieszczane w kolejce QE5 są przechowywane na stronie zestawu 21 (jeśli pozostają w kolejce wystarczająco długo, aby mogły zostać zapisane w pamięci DASD).

Więcej niż jedna kolejka może korzystać z tej samej klasy pamięci masowej, a użytkownik może zdefiniować tyle klas pamięci masowej, ile chcesz. Na przykład można rozszerzyć poprzedni przykład, aby uwzględnić więcej klas pamięci masowej i definicji kolejek, w następujący sposób:

```
DEFINE STGCLASS(ARC1) PSID(05)
DEFINE STGCLASS(ARC2) PSID(21)
DEFINE STGCLASS(MAXI) PSID(05)
DEFINE QLOCAL(QE1) STGCLASS(ARC1) ...
DEFINE QLOCAL(QE2) STGCLASS(ARC1) ...
DEFINE QLOCAL(QE3) STGCLASS(MAXI) ...
DEFINE QLOCAL(QE4) STGCLASS(ARC2) ...
DEFINE QLOCAL(QE5) STGCLASS(ARC2) ...
```

W produkcie Rysunek 72 na stronie 237 zarówno klasy pamięci masowej ARC1, jak i MAXI są powiązane z zestawem stron 05. Oznacza to, że kolejki QE1, QE2 i QE3 są odwzorowane na zestaw stron 05. Podobnie klasa pamięci masowej ARC2 wiąże kolejki QE4 i QE5 z zestawem stron 21.



Jeśli użytkownik zdefiniuje kolejkę bez określania klasy pamięci masowej, produkt IBM MQ użyje domyślnej klasy pamięci.

Jeśli komunikat jest umieszczany w kolejce, która określa nieistniejącą klasę pamięci masowej, aplikacja odbierze błąd. Należy zmienić definicję kolejki, aby nadać jej nazwę istniejącej klasy pamięci masowej, lub utworzyć klasę pamięci masowej nazwaną przez kolejkę.

Klasę pamięci masowej można zmienić tylko wtedy, gdy:

- Wszystkie kolejki, które używają tej klasy pamięci masowej, są puste i nie mają niezatwierdzonego działania.
- Wszystkie kolejki, które używają tej klasy pamięci masowej, są zamknięte.

z/OS Bufory i pule buforów dla IBM MQ for z/OS

Produkt IBM MQ for z/OS używa buforów i pul buforów do tymczasowego buforowania danych. W tym temacie opisano sposób organizacji buforów i ich wykorzystania.

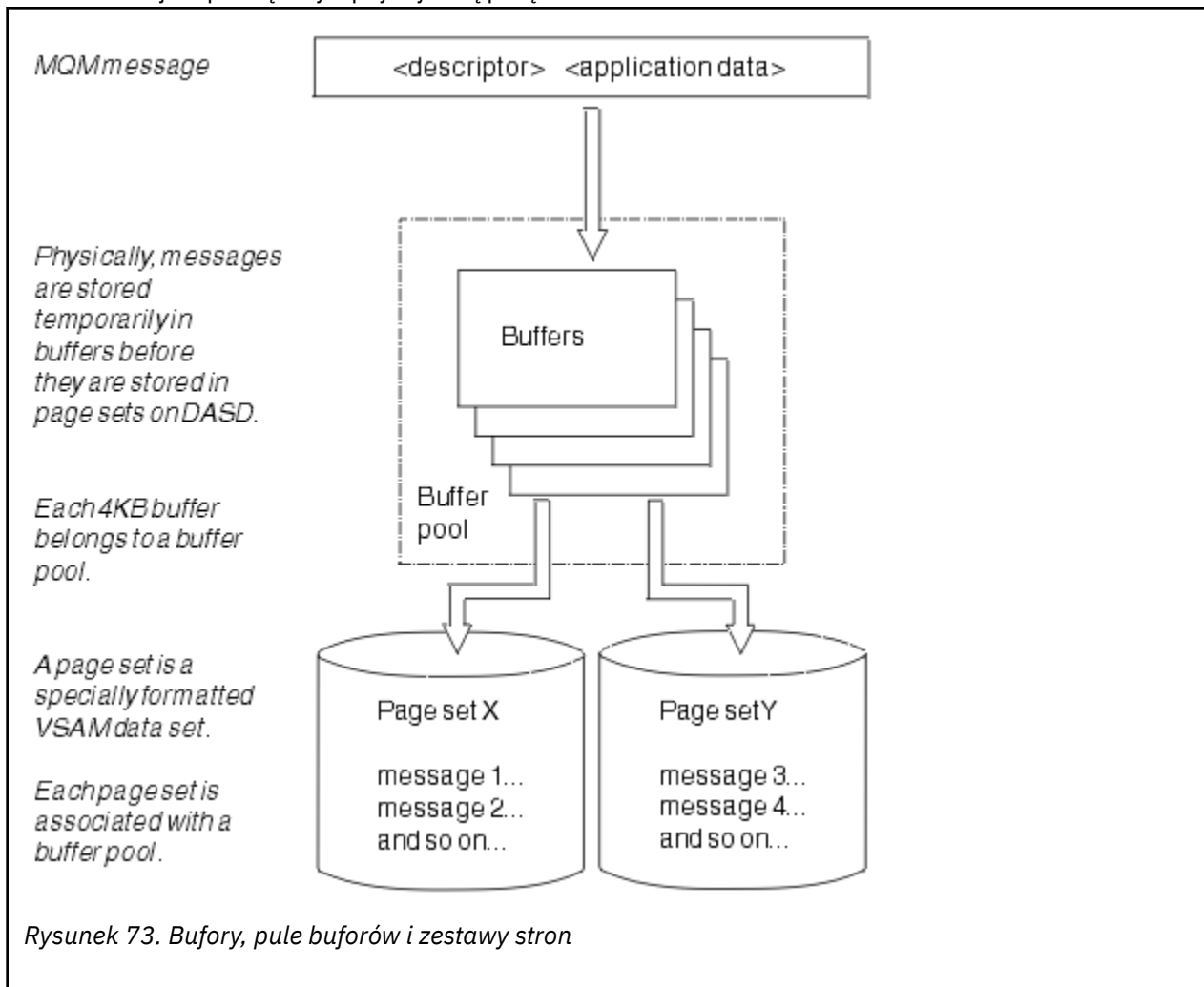
W celu zwiększenia wydajności produkt IBM MQ używa formy buforowania, w której komunikaty (i definicje obiektów) są tymczasowo przechowywane w buforach przed ich zapisami w zestawach stron w pamięci DASD. Krótkotrwali komunikaty, czyli komunikaty pobierane z kolejki zaraz po ich odebraniu, mogą być przechowywane tylko w buforach. To działanie buforowania jest kontrolowane przez menedżera buforów, który jest komponentem produktu IBM MQ.

Bufory są zorganizowane w *pule buforów*. Dla każdego menedżera kolejek można zdefiniować do 100 pul buforów (od 0 do 99).

Zaleca się użycie minimalnej liczby pul buforów zgodnych z separacją obiektów i typów komunikatów, które zostały opisane w programie Rysunek 73 na stronie 238, a także wszelkie wymagania dotyczące odseparowania danych, które mogą być spełnione przez aplikację użytkownika. Każdy bufor ma długość 4 kB. Pule buforów używają domyślnie 31-bitowej pamięci masowej, w tym trybie maksymalna liczba buforów jest określana przez ilość pamięci 31-bitowej dostępnej w przestrzeni adresowej menedżera kolejek. Nie należy używać więcej niż około 70% dla buforów. Alternatywnie, przydział pamięci puli buforów może być wykonany z 64-bitowej pamięci masowej (należy użyć atrybutu LOCATION w komendzie **DEFINE BUFFPOOL**). Korzystanie z funkcji LOCATION (ABOVE) w celu użycia 64-bitowej

pamięci masowej ma dwie korzyści. Po pierwsze, dostępnych jest znacznie więcej 64-bitowej pamięci masowej, więc pule buforów mogą być znacznie większe, a po drugie, 31-bitowa pamięć masowa jest udostępniona do użycia przez inne funkcje. Zwykle im więcej buforów jest dostępnych, tym bardziej wydajna jest buforowanie i lepsza wydajność produktu IBM MQ.

Rysunek 73 na stronie 238 przedstawia relacje między komunikatami, buforami, pulami buforów i zestawami stron. Pula buforów jest powiązana z jednym lub większą liczbę zestawów stron; każdy zestaw stron jest powiązany z pojedynczą pulą buforów.



Za pomocą komendy **ALTER BUFFPOOL** można dynamicznie wydawać komendy służące do modyfikowania wielkości puli buforów i położenia. Zestawy stron mogą być dynamicznie dodawane za pomocą komendy **DEFINE PSID** lub usuwane za pomocą komendy **DELETE PSID**.

Jeśli pula buforów jest zbyt mała, program IBM MQ wysyła komunikat **CSQP020E**. Następnie można dynamicznie dodawać więcej buforów do danej puli buforów (należy pamiętać, że może być konieczne usunięcie buforów z innych pul buforów w celu wykonania tego zadania).

Liczbę buforów w puli należy określić za pomocą komendy **DEFINE BUFFPOOL**, a można dynamicznie zmieniać wielkość pul buforów za pomocą komendy **ALTER BUFFPOOL**. Bieżącą liczbę buforów w puli można określić dynamicznie, wyświetlając zestaw stron, który korzysta z puli buforów, za pomocą komendy **DISPLAY USAGE**.

Ze względu na wydajność nie należy umieszczać komunikatów i definicji obiektów w tej samej puli buforów. Należy użyć jednej puli buforów (komendy numer zero) wyłącznie dla zestawu stron zero, gdzie przechowywane są definicje obiektów. W podobny sposób zachowaj krótkotrwałe komunikaty i długotrwałe komunikaty w różnych pulach buforów, a więc w różnych zestawach stron i w różnych kolejkach.

Komendy **DEFINE BUFFPOOL** nie można użyć po restarcie w celu utworzenia nowej puli buforów. Zamiast tego, jeśli komenda **DEFINE PSID** używa słowa kluczowego DSN, może ona jawnie zidentyfikować pulę buforów, która nie jest aktualnie zdefiniowana. Nowa pula buforów zostanie utworzona.

z/OS **Gdzie znaleźć więcej informacji na temat zarządzania pamięcią masową dla produktu IBM MQ for z/OS**

Ten temat zawiera informacje dotyczące znajdowania dalszych informacji na temat zarządzania pamięcią masową dla produktu IBM MQ for z/OS.

Więcej informacji na temat tematów w tej sekcji można znaleźć z następujących źródeł:

| <i>Tabela 20. Gdzie znaleźć więcej informacji na temat zarządzania pamięcią masową</i> | |
|--|---|
| Temat | Gdzie szukać |
| Ilość potrzebnej pamięci masowej | Planowanie wymagań dotyczących pamięci masowej i wydajności w systemie z/OS |
| Jak duże jest tworzenie zestawów stron i pule buforów | Planowanie zestawów stron i pul buforów |
| Zarządzanie zestawami stron | Zarządzanie zestawami stron |
| Komendy MQSC | Komendy MQSC |

z/OS **logowanie IBM MQ for z/OS**

Produkt IBM MQ przechowuje *dzienniki* zmian danych i istotnych zdarzeń w miarę ich występowania. Dzienniki te mogą być używane do odzyskiwania danych do poprzedniego stanu, jeśli jest to wymagane.

Zestaw danych programu startowego (BSDS) przechowuje informacje na temat zestawów danych, które zawierają dzienniki.

Dziennik nie zawiera informacji dotyczących statystyk, danych śledzenia ani oceny wydajności. Więcej informacji na temat informacji statystycznych i monitorowania gromadzonych przez program IBM MQ zawiera sekcja [Monitorowanie i statystyka](#).

Więcej informacji na temat rejestrowania można znaleźć w następujących tematach:

- [“Pliki dzienników w produkcie IBM MQ for z/OS” na stronie 240](#)
- [“Sposób ustrukturyzowania dziennika” na stronie 244](#)
- [“Jak zapisywane są dzienniki produktu IBM MQ for z/OS” na stronie 244](#)
- [“Względny adres w bajtach względnego dziennika” na stronie 247](#)
- [“Zestaw danych programu startowego” na stronie 248](#)

Zadania pokrewne

[Planowanie środowiska rejestrowania](#)

[Ustawianie dzienników za pomocą modułu parametrów systemowych](#)

z/OS [administrowanie z/OS](#)

Odsyłacze pokrewne

z/OS [Komunikaty dla IBM MQ for z/OS](#)

[Źródła, z których można wydawać komendy MQSC w systemie z/OS](#)

Pliki dziennika zawierają informacje potrzebne do odtwarzania transakcji. Aktywne pliki dziennika mogą być archiwizowane, dzięki czemu można przechowywać dane dziennika przez długi czas.

Co to jest plik dziennika

Program IBM MQ rejestruje wszystkie istotne zdarzenia w postaci *aktywnego dziennika*. W dzienniku znajdują się informacje potrzebne do odtworzenia:

- Komunikaty trwałe
- Obiekty IBM MQ , takie jak kolejki
- Menedżer kolejek produktu IBM MQ

Dziennik aktywny zawiera zbiór zestawów danych (do 310), które są używane cyklicznie.

Archiwizowanie dziennika można włączyć w taki sposób, aby po wypełnieniu aktywnego dziennika w zestawie danych archiwalnych dokonano operacji kopiowania. Użycie archiwizacji pozwala na zachowanie danych dziennika przez dłuższy czas. Jeśli archiwizacja nie zostanie użyta, dane zawiązania dzienników i wcześniejsze dane zostaną nadpisane. Aby odtworzyć zestaw stron lub odzyskać dane w strukturze systemu CF, należy użyć danych dziennika, gdy kopia zapasowa zestawu stron lub struktury została utworzona. Dziennik archiwalny może być utworzony na dysku lub na taśmie.

Archiwizowanie

Ponieważ aktywny dziennik ma stałą wielkość, program IBM MQ kopiuje zawartość każdego zestawu danych dziennika okresowo do *dziennika archiwalnego*, który zwykle jest zestawem danych w urządzeniu pamięci masowej o dostępie bezpośrednim (DASD) lub na taśmie magnetycznej. Jeśli wystąpi awaria podsystemu lub transakcji, produkt IBM MQ używa aktywnego dziennika i, jeśli to konieczne, dziennika archiwalnego do odtwarzania.

Dziennik archiwum może zawierać do 1000 sekwencyjnych zestawów danych. Każdy zestaw danych można katalogować za pomocą zintegrowanego narzędzia katalogowego produktu z/OS (ICF).

Archiwizacja jest podstawowym komponentem odtwarzania produktu IBM MQ . Jeśli jednostka odtwarzania jest długotrwałą, rekordy dziennika w tej jednostce odtwarzania mogą znajdować się w dzienniku archiwalnym. W takim przypadku odtwarzanie wymaga danych z dziennika archiwalnego. Jeśli jednak archiwizacja jest wyłączona, aktywny dziennik z nowymi zapisami rekordów dziennika powoduje nadpisanie wcześniejszych rekordów dziennika. Oznacza to, że program IBM MQ może nie być w stanie wycofać jednostki odzyskiwania, a komunikaty mogą zostać utracone. Następnie menedżer kolejek kończy działanie w sposób nieprawidłowy.

W związku z tym w środowisku produkcyjnym **nigdy nie wyłączono archiwizowania**. W takim przypadku należy uruchomić ryzyko utraty danych po awarii systemu lub transakcji. Tylko w przypadku, gdy środowisko testowe jest uruchamiane w środowisku testowym, można wyłączyć archiwizowanie. W tym celu należy użyć makra CSQ6LOGP , które jest opisane w sekcji [Korzystanie z komendy CSQ6LOGP](#).

Aby zapobiec problemom z nieplanowanymi długo działającymi jednostkami pracy, produkt IBM MQ wysyła komunikat (CSQJ160I lub CSQJ161I) Jeśli podczas aktywnego przetwarzania przenoszenia dziennika zostanie wykryta długotrwała jednostka pracy.

podwójne rejestrowanie

W podwójnym rejestrowaniu każdy rekord dziennika jest zapisywany w dwóch różnych zestawach danych aktywnego dziennika w celu zminimalizowania prawdopodobieństwa wystąpienia problemów z utratą danych podczas restartowania.

Produkt IBM MQ można skonfigurować w taki sposób, aby był uruchamiany z *pojedynczym rejestrowaniem* lub *podwójnym rejestrowaniem*. W przypadku pojedynczego rejestrowania rekordy dziennika są

zapisywane od razu do aktywnego zestawu danych dziennika. Każdy aktywny zestaw danych dziennika jest pojedynczym liniowym zestawem danych VSAM (LDS-VSAM). W przypadku podwójnego rejestrowania każdy rekord dziennika jest zapisywany w dwóch różnych zestawach danych aktywnego dziennika. Podwójne rejestrowanie minimalizuje prawdopodobieństwo wystąpienia problemów z utratą danych podczas restartu.

Wyskakowanie dziennika

Wyskakowanie dziennika powoduje, że rekordy dziennika dla niektórych jednostek pracy są zapisywane dalej w dzienniku. Zmniejsza to ilość danych dziennika, które muszą zostać odczytane przy restarcie menedżera kolejek lub wycofaniu, dla długotrwałych lub długoterminowych jednostek pracy wątpliwych.

Gdy jednostka pracy jest uznawana za długa, reprezentacja każdego rekordu dziennika jest zapisywana dalej w dół dziennika. Ta technika jest znana jako *shunting*. Gdy cała jednostka pracy została przetworzona, jednostka pracy jest w stanie *zaciemniona*. Każde działanie wycofania lub restartu odnoszące się do zaciemnionej jednostki pracy może korzystać z nieokreślonych rekordów dziennika, a nie z oryginalnej jednostki rekordów dziennika pracy.

Wykrywanie długotrwałej jednostki pracy jest funkcją procesu punktu kontrolnego. W czasie punktu kontrolnego każda aktywna jednostka pracy jest sprawdzana w celu określenia, czy musi zostać wyłuszczone. Jeśli jednostka pracy przeszła dwa wcześniejsze punkty kontrolne od momentu jej utworzenia, lub od czasu ostatniego zaciemnienia, jednostka pracy jest odpowiednia do zaciemnionej pracy. Oznacza to, że pojedyncza jednostka pracy może zostać skracana więcej niż jeden raz. Jest to jednostka pracy *multi-shunted*.

Jednostka pracy jest niepodzielna co trzy punkty kontrolne. Jednak punkt kontrolny jest wykonywany asynchronicznie w przełączniku log-switch (lub zapisu rekordu dziennika, który spowodował przekroczenie LOGLOAD).

W danym momencie odbywa się tylko jeden punkt kontrolny, więc przed zakończeniem punktu kontrolnego może istnieć wiele przełączników dziennika.

Oznacza to, że jeśli nie ma wystarczającej liczby aktywnych dzienników lub jeśli są one zbyt małe, to przed zapętnieniem wszystkich dzienników może nie zakończyć się odcieniem dużej jednostki pracy.

Komunikat `CSQR027I` powoduje, że nie jest możliwe zakończenie operacji *shunting*.

Jeśli archiwizowanie dziennika jest wyłączone, `ABEND 5C6` z przyczyną `00D1032A` występuje wtedy, gdy istnieje próba utworzenia kopii zapasowej jednostki pracy, dla której nie powiodło się odcienie. Aby uniknąć tego problemu, należy użyć opcji `OFFLOAD=YES`.

Wyskakowanie dziennika jest zawsze aktywne i działa, czy archiwizowanie dziennika jest włączone.

Uwaga: Mimo że wszystkie rekordy dziennika dla jednostki pracy są zaciemnione, cała treść każdego rekordu nie jest zaciemniona, ale tylko część, która jest niezbędna do wycofania. Oznacza to, że ilość zapisanych danych dziennika jest przechowywana do minimum i że nie można użyć rekordów zaciemnionych, jeśli wystąpi błąd zestawu stron. Długo działająca jednostka pracy jest uruchomiona dla więcej niż trzech punktów kontrolnych menedżera kolejek.

Więcej informacji na temat usuwania dziennika zawiera sekcja [Zarządzanie dziennikami](#).

Kompresja dziennika

Produkt IBM MQ for z/OS można skonfigurować w taki sposób, aby kompresował i dekompresował rekordy dziennika w miarę ich zapisywania i odczytywania z zestawu danych dziennika.

Kompresja dziennika może być używana do zmniejszenia ilości danych zapisanych w dzienniku dla trwałych komunikatów w kolejkach prywatnych. Ilość osiągniętego kompresji zależy od typu danych zawartych w komunikatach. Na przykład: Uruchomienie kodowania długości (Run Length Encoding-RLE) działa poprzez zkompilowanie powtarzających się wystąpień bajtów, które mogą wydajnie dawać dobre wyniki w przypadku danych ustrukturyzowanych lub danych zorientowanych na rekordy.



Ostrzeżenie: Komunikaty trwałe, które są umieszczane w kolejce współużytkowanej, nie podlegają kompresji dziennika.

Można użyć pól w sekcji menedżera rejestrowania rekordów SMF (System Management Facility) 115 (SMF), aby monitorować liczbę operacji kompresji danych. Więcej informacji na temat SMF zawiera sekcja [Korzystanie z narzędzia System Management Facility](#) oraz [Komunikaty rozliczeniowe i statystyki](#).

Kompresja dziennika zwiększa wykorzystanie procesora w systemie. Należy rozważyć użycie kompresji tylko wtedy, gdy przepustowość menedżera kolejek jest ograniczona przez zapis pasma we/wy do zestawów danych dziennika lub jest to ograniczone przez pamięć dyskową potrzebną do przechowywania zestawów danych dziennika. Jeśli używane są kolejki współużytkowane, ograniczenia przepustowości we/wy mogą być zwalniane przez dodanie dodatkowych menedżerów kolejek do grupy współużytkowania kolejek i rozpraszanie obciążenia przez więcej menedżerów kolejek.

Opcja kompresji dziennika może być włączona i wyłączona, jeśli jest to wymagane bez konieczności zatrzymywania i restartowania menedżera kolejek. Menedżer kolejek może odczytać wszystkie skompresowane rekordy dziennika, niezależnie od bieżącego ustawienia kompresji dziennika.

Menedżer kolejek obsługuje 3 ustawienia kompresji dziennika.

BRAK

Nie jest używana żadna kompresja danych dziennika. Jest to wartość domyślna.

RLE

Kompresja danych dziennika jest wykonywana przy użyciu kodowania run-length (RLE).

ANY

Włącz menedżer kolejek, aby wybrać algorytm kompresji, który daje największy stopień kompresji rekordów dziennika. Ta opcja powoduje kompresję RLE.

Kompresję rekordów dziennika można sterować za pomocą jednej z następujących opcji:

- Komendy SET i DISPLAY LOG w MQSC; patrz [SET LOG](#) i [DISPLAY LOG](#) .
- Funkcje rejestrowania i sprawdzania dziennika w interfejsie PCF. Patrz [Set log \(Ustaw dziennik\)](#) i [Inquire log \(Rejestruj dziennik\)](#).
- Makro CSQ6LOGP w module parametrów systemowych; patrz [Korzystanie z CSQ6LOGP](#)

Oprócz tego program narzędziowy do drukowania dziennika CSQ1LOGP obsługuje rozszerzenie wszystkich skompresowanych rekordów dziennika.

Dane dziennika

Dziennik może zawierać do 18 milionów milionów milionów ($1.8 \cdot 10^{19}$) bajtów. Każdy bajt może być adresowany przez jego przesunięcie od początku dziennika, a przesunięcie to jest określane jako *względny adres bajtu* (RBA).

Do RBA odwołuje się pole 6-bajtowe lub 8-bajtowe, które daje łączny adresowalny zakres (2^{48} bajtów) lub 2^{64} bajtów (w zależności od tego, czy używane są 6-bajtowe lub 8-bajtowe RBAs dziennika).

Jednak gdy program IBM MQ wykryje, że używany zakres jest dłuższy niż F00000000000 (jeśli używane są 6-bajtowe RBAs) lub FFFF800000000000 (jeśli używane są 8-bajtowe RBAs), komunikaty CSQI045, CSQI046, CSQI047i CSQJ032 są wysyłane, ostrzegając o zresetowaniu dziennika RBA.

Jeśli wartość RBA osiągnie wartość FFF800000000 (jeśli używane są 6-bajtowe RBAs) lub FFFFFFFC00000000 (jeśli używane są 8-bajtowe RBAs), menedżer kolejek kończy działanie z kodem przyczyny 00D10257.

Po wygenerowaniu komunikatów ostrzegawczych dotyczących używanego zakresu dziennika należy zaplanować wyłączenie menedżera kolejek, w którym może zostać przekształcony menedżer kolejek w celu użycia 8-bajtowych dzienników RBAs lub można zresetować dziennik. Procedura resetowania dziennika została opisana w sekcji [Resetowanie dziennika menedżera kolejek](#).

Jeśli menedżer kolejek używa 6-bajtowego dziennika RBAs, należy rozważyć przekształcenie menedżera kolejek w celu użycia 8-bajtowych dzienników RBA, a nie zresetowania dziennika menedżera kolejek,

postępując zgodnie z procedurą opisaną w sekcji [Implementowanie większego względnego adresu bajtu dziennika](#).

Dziennik składa się z *rekordów dziennika*, z których każdy jest zestawem danych dziennika traktowanych jako pojedyncza jednostka. Rekord dziennika jest identyfikowany przez RBA pierwszego bajtu nagłówka lub przez jego numer kolejny rekordu dziennika (LRSN). RBA lub LRSN jednoznacznie identyfikuje rekord, który rozpoczyna się od określonego punktu w dzienniku.

To, czy używany jest RBA, czy LRSN do identyfikowania punktów dziennika, zależy od tego, czy używane są grupy współużytkowania kolejek. W środowisku współużytkowania kolejek nie można używać względnego adresu bajtowego do jednoznacznej identyfikacji punktu dziennika, ponieważ wiele menedżerów kolejek może jednocześnie aktualizować tę samą kolejkę, a każdy z nich ma własny dziennik. Aby rozwiązać ten problem, numer kolejny rekordu dziennika jest wyliczany na podstawie wartości znacznika czasu i nie musi oznaczać fizycznego przemieszczenia rekordu dziennika w dzienniku.

Każdy rekord dziennika zawiera nagłówek, który nadaje jego typ, podkomponent IBM MQ, który dokonał rekordu, oraz, dla jednostki rekordów odtwarzania, jednostkę identyfikatora odtwarzania.

Istnieją cztery typy rekordów dziennika, opisane w następujących nagłówkach:

- [Rekordy dziennika jednostki odzyskiwania](#)
- [Rekordy punktu kontrolnego](#)
- [Rekordy sterujące zestawu stron](#)
- [Rekordy kopii zapasowej struktury CF](#)

Jednostka rekordów dziennika odtwarzania

Większość rekordów dziennika opisuje zmiany w kolejkach produktu IBM MQ. Wszystkie takie zmiany są wprowadzane w jednostkach odzyskiwania.

W produkcie IBM MQ stosowane są specjalne techniki rejestrowania obejmujące metody *undo/redo* i *compensating log records* w celu skrócenia czasów restartu i zwiększenia dostępności systemu.

Jednym z efektów jest to, że czas restartu jest ograniczony. Jeśli błąd wystąpi podczas restartu, tak aby menedżer kolejek był restartowany po raz drugi, to wszystkie działania odtwarzania, które zakończyły się w punkcie awarii w pierwszym restarcie, nie muszą być ponownie stosowane podczas drugiego restartu. Oznacza to, że kolejne restarty nie są coraz dłuższe i nie są już kompletne.

Rekordy punktów kontrolnych

Aby zredukować czas restartu, program IBM MQ wykonuje okresowe punkty kontrolne podczas normalnego działania. Występują one w następujący sposób:

- Gdy została zapisana predefiniowana liczba rekordów dziennika. Ten numer jest definiowany przez operand częstotliwości punktu kontrolnego o nazwie LOGLOAD makra parametru systemowego CSQ6SYSP, który jest opisany w sekcji [Korzystanie z komendy CSQ6SYSP](#).
- Na koniec pomyślny restart.
- Przy normalnym zakończeniu.
- Za każdym razem, gdy program IBM MQ przetacza się do następnego aktywnego zestawu danych dziennika w cyklu.

Podczas wykonywania punktu kontrolnego program IBM MQ wysyła komendę DISPLAY CONN (opisaną w sekcji [DISPLAY CONN](#)). Wewnętrznie tak, aby lista obecnie włączonych połączeń została zapisana w dzienniku konsoli z/OS.

Rekordy sterujące zestawu stron

Rekordy te rejestrują zestawy stron i pule buforów znane menedżerowi kolejek produktu IBM MQ w każdym punkcie kontrolnym i zapisują informacje o zakresach dzienników, które są wymagane do odtworzenia nośników na stronie ustawionej w czasie punktu kontrolnego.

Niektóre zmiany dynamiczne w zestawach stron i pulach buforów są również zapisywane jako rekordy sterujące zestawu stron, dzięki czemu zmiany mogą zostać odzyskane i automatycznie przywrócone przy następnym restarcie menedżera kolejek.

Rekordy kopii zapasowej struktury CF

Te rekordy przechowują dane odczytane z struktury listy narzędzia CF w odpowiedzi na komendę BACKUP CFSTRUCT. W mało prawdopodobnym przypadku awarii struktury narzędzia CF rekordy te są używane wraz z jednostką rekordów odtwarzania przez komendę RECOVER CFSTRUCT w celu wykonania odtwarzania nośnika w strukturze narzędzia CF do punktu awarii.

Zadania pokrewne

[Implementowanie większego adresu bajtu względnego w dzienniku](#)

Sposób ustrukturyzowania dziennika

W tym temacie opisano terminologię używaną do opisywania rekordów dziennika.

Każdy zestaw danych aktywnego dziennika musi być liniowym zestawem danych VSAM (LDS). Fizyczna jednostka wyjściowa zapisana w aktywnym zestawie danych dziennika wynosi 4 kB interwału sterującego (CI). Każdy element CI zawiera jeden rekord VSAM.

Rekordy dziennika fizycznego i logicznego

Jeden element VSAM CI jest rekordem *fizycznym*. Informacje protokołowane w określonym czasie tworzą rekord *logiczny* o długości niezależnie od miejsca dostępnego w CI. Tak więc jeden rekord fizyczny może zawierać:

- Kilka rekordów logicznych
- Jeden lub większa liczba rekordów logicznych i część innego rekordu logicznego
- Tylko część jednego rekordu logicznego

Termin *rekord dziennika* odnosi się do rekordu *logicznego*, niezależnie od liczby rekordów *fizycznych* potrzebnych do jego zapisania.

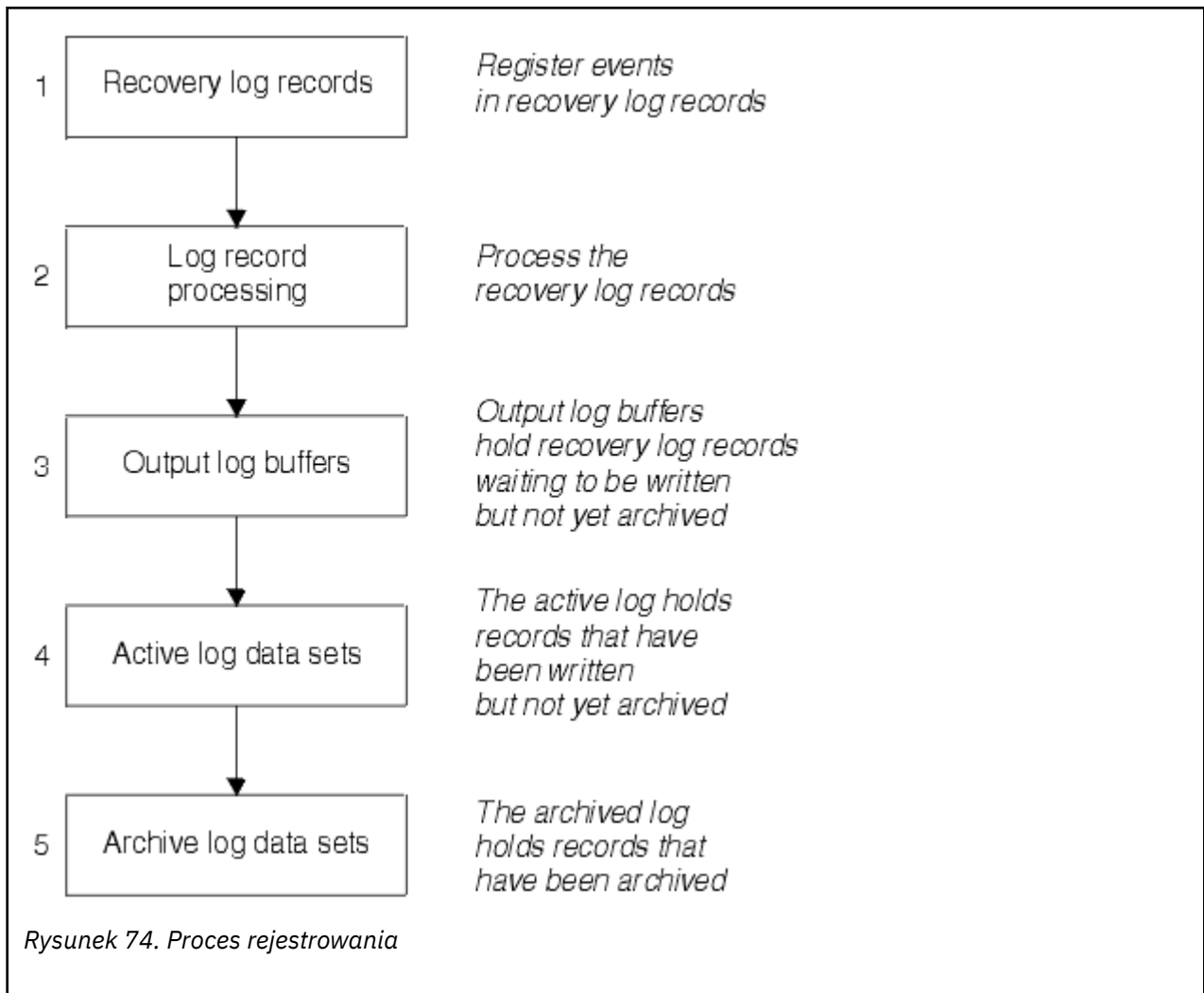
Jak zapisywane są dzienniki produktu IBM MQ for z/OS

W tym temacie opisano sposób, w jaki program IBM MQ przetwarza rekordy pliku dziennika.

Program IBM MQ zapisuje każdy rekord dziennika w zestawie danych DASD o nazwie *aktywny dziennik*. Gdy aktywny dziennik jest zapełniony, program IBM MQ kopiuje jego zawartość do zestawu danych DASD lub taśmowego o nazwie *dziennik archiwalny*. Ten proces nosi nazwę *offloading*.

Rysunek 74 na stronie 245 ilustruje proces rejestrowania. Rekordy dziennika zwykle są dostępne w następującym cyklu:

1. Program IBM MQ zapisuje zmiany w danych i znaczące zdarzenia w rekordach dziennika odtwarzania.
2. Produkt IBM MQ przetwarza rekordy dziennika odtwarzania i w razie potrzeby przerywa je w segmenty.
3. Rekordy dziennika są umieszczane sekwencyjnie w *buforach wyjściowych dziennika*, które są formatowane jako interwały sterujące VSAM (CI). Każdy rekord dziennika jest identyfikowany przez względny adres bajtowy w zakresie od zera do $2^{64}-1$.
4. Elementy CI są zapisywane w zestawie predefiniowanych zestawów danych aktywnego dziennika DASD, które są używane sekwencyjnie i poddawane recyklingowi.
5. Jeśli archiwizacja jest aktywna, ponieważ każdy aktywny zestaw danych dziennika jest zapełniony, jego zawartość jest automatycznie przesunięta do nowego zestawu danych dziennika archiwalnego.



Gdy aktywny jest dziennik aktywny

Bufory dziennika w pamięci masowej są zapisywane w aktywnym zestawie danych dziennika za każdym razem, gdy wystąpi dowolny z następujących zdarzeń:

- Bufory dziennika stają się pełne.
- Osiągnięty zostanie próg zapisu (określony w makrze CSQ6LOGP).
- Występują pewne istotne zdarzenia, takie jak punkt zatwierdzania lub gdy komenda IBM MQ BACKUP CFSTRUCT jest wydawana.

Po zainicjowaniu menedżera kolejek zestawy danych aktywnego dziennika nazwane w BSDS są dynamicznie przydzielane do wyłącznego użytku przez menedżer kolejek i pozostają przydzielane wyłącznie do programu IBM MQ do czasu zakończenia działania menedżera kolejek.

Dynamiczne dodawanie zestawów danych dziennika

Istnieje możliwość dynamicznego definiowania nowych aktywnych zestawów danych dziennika w czasie działania menedżera kolejek. Ta funkcja łagodzi problem zawieszenia menedżera kolejek, gdy archiwizacja nie jest w stanie odciążać aktywnych dzienników z powodu wystąpienia problemu przejściowego. Więcej informacji na ten temat zawiera opis komendy [DEFINE LOG](#).

Uwaga: Aby ponownie zdefiniować lub usunąć aktywne dzienniki, należy przerwać i zrestartować menedżer kolejek.

Podsystem IBM MQ i podsystem zarządzania pamięcią masową

Parametry IBM MQ umożliwiają określenie klas pamięci masowej podsystemu zarządzania pamięcią masową (Storage Management Subsystem- MVS/DFP SMS) podczas dynamicznego przydzielania zestawów danych dziennika archiwalnego produktu IBM MQ . Program IBM MQ inicjuje archiwizowanie zestawów danych dziennika, ale za pomocą programu SMS można wykonać przydzielanie zestawu danych archiwum.

Odsyłacze pokrewne

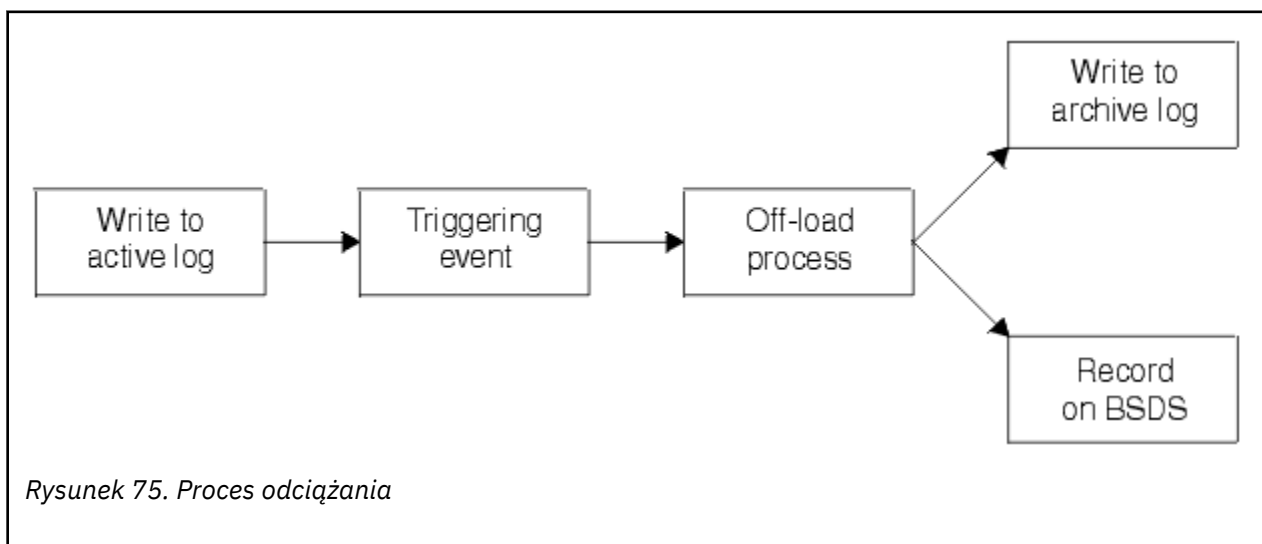
“Gdy zapisywany jest dziennik archiwum produktu IBM MQ for z/OS” na stronie 246

W tym temacie opisano proces kopiowania aktywnych dzienników do dzienników archiwalnych oraz po wystąpieniu procesu.

Gdy zapisywany jest dziennik archiwum produktu IBM MQ for z/OS

W tym temacie opisano proces kopiowania aktywnych dzienników do dzienników archiwalnych oraz po wystąpieniu procesu.

Proces kopiowania aktywnych dzienników do dzienników archiwalnych nosi nazwę *offloading*(odciążenie). Relacja przesunięcia do innych zdarzeń rejestrowania jest przedstawiana w sposób schematyczny w produkcie *Rysunek 75* na stronie 246.



Rysunek 75. Proces odciążania

Wyzwalanie procesu odciążania

Proces odciążania aktywnego dziennika do dziennika archiwalnego może być wyzwalany przez kilka zdarzeń. Na przykład:

- Wypełnianie aktywnego zestawu danych dziennika.
- Za pomocą komendy MQSC ARCHIVE LOG.
- Wystąpił błąd podczas zapisu do aktywnego zestawu danych dziennika.

Zestaw danych jest obcinany przed punktem awarii, a rekord, który nie został zapisany, staje się pierwszym rekordem nowego zestawu danych. Odciążenie jest wyzwalane dla obciętego zestawu danych, co w przypadku normalnego pełnego zestawu danych dziennika. Jeśli istnieją podwójne dzienniki aktywne, obie kopie są obcinane, dzięki czemu dwie kopie pozostają zsynchronizowane.

Komunikat CSQJ110E jest generowany, gdy ostatni dostępny dziennik aktywny jest pełny o 5%, a następnie o 5% przyrostu, podając procent pojemności dziennika, który jest w użyciu. Jeśli wszystkie aktywne dzienniki staną się pełne, program IBM MQ zatrzyma przetwarzanie, dopóki nie wystąpi odciążenie, a następnie zostanie wyświetlony następujący komunikat:

```
CSQJ111A +CSQ1 OUT OF SPACE IN ACTIVE LOG DATA SETS
```

Proces odciążania

Po zapełnieniu wszystkich aktywnych dzienników program IBM MQ uruchamia proces odciążania i zatrzymuje przetwarzanie, dopóki proces odciążający nie zostanie zakończony. Jeśli przetwarzanie odciążające nie powiedzie się, gdy aktywne dzienniki są zapełniane, program IBM MQ będzie się abkoował.

Gdy aktywny dziennik jest gotowy do odciążenia, do operatora konsoli z/OS wysyłane jest żądanie w celu podłączenia taśmy lub przygotowania jednostki DASD. Wartość opcji rejestrowania ARCWTOR (więcej informacji na ten temat zawiera sekcja [Korzystanie z komendy CSQ6ARVP](#)), określa, czy żądanie zostało odebrane. Jeśli taśma jest używana do odciążania, należy określić wartość ARCWTOR=YES. Jeśli wartością jest YES, żądanie jest poprzedzone przez WTOR (numer komunikatu CSQJ008E) informujące operatora o przygotowaniu zestawu danych dziennika archiwalnego, który ma być przydzielony.

Operator nie musi natychmiast odpowiadać na ten komunikat. Opóźnienie odpowiedzi opóźnia jednak proces odciążenia. Nie ma to wpływu na wydajność systemu IBM MQ, chyba że operator opóźni odpowiedź, tak długo, że program IBM MQ nie uruchamia aktywnych dzienników.

Operator może odpowiedzieć, anulując proces odciążania. W takim przypadku, jeśli przydział dotyczy pierwszej kopii podwójnych zestawów danych archiwalnych, proces odciążania jest opóźniony dopiero po zapełnieniu następnego aktywnego zestawu danych. Jeśli przydział jest przeznaczony dla drugiej kopii, proces archiwizacji przelączy się na tryb pojedynczego kopiowania, ale tylko dla tego zestawu danych.

Przerwa i błędy podczas odciążenia

Żądanie zatrzymania menedżera kolejek nie staje się aktywne, dopóki przetwarzanie offload nie zostanie zakończone. Jeśli działanie produktu IBM MQ zakończy się niepowodzeniem podczas przenoszenia, odciążenie rozpoczyna się ponownie po zrestartowaniu menedżera kolejek.

Komunikaty podczas przetwarzania offload

Odciążone komunikaty są wysyłane do konsoli z/OS przez program IBM MQ i proces odciążania. Komunikaty te mogą być używane do znajdowania zakresów RBA w różnych zestawach danych dziennika.

Względny adres w bajtach względnego dziennika

Ta funkcja zwiększa dostępność menedżera kolejek przez wydłużanie czasu przed zresetowaniem dziennika.

Dane odtwarzania są zapisywane w dzienniku, tak aby komunikaty trwały były dostępne po zrestartowaniu menedżera kolejek. Termin log względny Adres bajtowy (log RBA) jest używany do odwołania się do położenia danych jako przesunięcia od początku dziennika.

Przed IBM MQ 8.0, 6-bajtowy dziennik RBA może zająć do 256 terabajtów danych. Przed zapisami tej ilości rekordów dziennika należy zresetować dziennik menedżera kolejek, postępując zgodnie z procedurą opisaną w sekcji [Resetowanie dziennika menedżera kolejek](#).

Resetowanie dzienników menedżerów kolejek nie jest szybkim procesem i może wymagać rozszerzonego wyłączenia, ze względu na konieczność zresetowania zestawów stron w ramach procesu. W przypadku menedżera kolejek wysokiego użycia ta operacja może być zazwyczaj wykonywana raz w roku.

W produkcie IBM MQ 8.0 dziennik RBA może mieć długość 8 bajtów, a menedżer kolejek może teraz zająć ponad 64,000 razy więcej danych (16 eksabajtów), zanim dziennik RBA musi zostać zresetowany. Wpływu

korzystania z większego dziennika RBA polega na tym, że wielkość zapisu danych dziennika zwiększa się o kilka bajtów.

Kiedy ta funkcja jest włączona?

LTS Tę funkcję można włączyć w dowolnym momencie, chociaż najlepiej zaplanować zmianę i umożliwić jej jawne włączenie do przedsiębiorstwa, zanim bieżący dziennik RBA zbliży się do końca 6-bajtowego zakresu RBA dziennika. Informacje na temat planowania migracji do 8 bajtów dziennika RBAs można znaleźć w sekcji [Planowanie zwiększania maksymalnego zakresu dziennika adresowalnego](#).

V 9.2.5 Menedżery kolejek utworzone w systemie IBM MQ 9.2.5 lub w późniejszym czasie mają już włączoną tę funkcję.

Ważne: Zanim możliwe będzie użycie tej możliwości w grupie współużytkownika kolejek, wszystkie menedżery kolejek w grupie współużytkownika kolejek muszą mieć jeden z następujących poziomów:

- W systemie IBM MQ 9.0.n CD, IBM MQ 9.1.0 LTS lub w nowszej wersji
- W systemie IBM MQ 9.0.0 i zostały uruchomione z programem **OPMODE= (NEWFUNC,800)** lub **OPMODE= (NEWFUNC,900)**

Jeśli bieżąca wartość RBA dziennika zbliży się do końca zakresu dziennika RBA, należy rozważyć przekształcenie menedżera kolejek w taki sposób, aby używany był 8-bajtowy dziennik RBA, a nie resetowanie dziennika menedżera kolejek. Przekształcenie menedżera kolejek w celu użycia 8-bajtowych dzienników RBA wymaga krótszego wyłączenia niż zresetowanie dziennika i znacznie wydłuża czas, zanim będzie konieczne zresetowanie dziennika.

Komunikat CSQJ034I wydany podczas inicjowania menedżera kolejek wskazuje koniec zakresu dziennika RBA dla menedżera kolejek jako skonfigurowany i może być używany do określenia, czy używane są 6 bajtowe lub 8-bajtowe RBAs dziennika.

W jaki sposób ta funkcja jest włączona?

8-bajtowy dziennik RBA jest włączany przez uruchomienie menedżera kolejek w formacie BSDS w wersji 2. Podsumowując, cel ten jest osiąganym poprzez:

1. Upewniając się, że wszystkie menedżery kolejek w grupie współużytkownika kolejek spełniają wymagania dotyczące włączania 8-bajtowego dziennika RBA
2. Czyszczenie menedżera kolejek w sposób czysty
3. Uruchomienie programu narzędziowego do konwersji BSDS w celu utworzenia kopii BSDS w formacie wersji 2.
4. Restartowanie menedżera kolejek z przekształconym BSDS.

Po przekształceniu menedżera kolejek w taki sposób, aby korzystało z 8-bajtowych dzienników RBA, nie może on wrócić do użycia 6-bajtowego dziennika RBA.

Sekcja [Implementowanie większego względnego adresu bajtowego dziennika](#) zawiera szczegółowe informacje na temat włączania 8-bajtowych dzienników RBAs.

Zadania pokrewne

[Planowanie zwiększenia maksymalnego zakresu dziennika adresowalnego](#)

Odsyłacze pokrewne

[Program narzędziowy do konwersji BSDS \(CSQJUCNV\)](#)

Z/OS Zestaw danych programu startowego

Zestaw danych programu startowego jest wymagany przez produkt IBM MQ jako mechanizm do tworzenia odwołań do zestawów danych dziennika i rekordów dziennika. Informacje te są wymagane podczas normalnego przetwarzania i restartowania odtwarzania.

Co to jest zestaw danych programu startowego

Zestaw danych programu startowego (BSDS) to zestaw danych KSDS (VSAM-sequenced data set), który przechowuje informacje wymagane przez produkt IBM MQ. Zawiera on następujące elementy:

- Spis wszystkich aktywnych i zarchiwizowanych zestawów danych dziennika znanych dla produktu IBM MQ. Produkt IBM MQ używa tego spisu zasobów do:
 - Śledzenie aktywnych i zarchiwizowanych zestawów danych dziennika
 - Znajdowanie rekordów dziennika, dzięki czemu może on spełniać żądania odczytu dziennika podczas normalnego przetwarzania
 - Znajdź rekordy dziennika, aby można było obsłużyć przetwarzanie restartu

Produkt IBM MQ przechowuje informacje w magazynie za każdym razem, gdy zdefiniowany jest zestaw danych dziennika archiwalnego lub ponownie wykorzystywany jest aktywny zestaw danych dziennika. W przypadku dzienników aktywnych wyświetlane są pełne informacje o zasobach, które są dostępne do ponownego wykorzystania. W spisie magazynowym znajduje się względny adres bajtu (RBA) każdej części dziennika znajdującego się w tym zestawie danych.

- *opakowywanie* spisu wszystkich ostatnich działań IBM MQ . Jest to konieczne, jeśli konieczne jest zrestartowanie menedżera kolejek.

BSDS jest wymagany, jeśli menedżer kolejek ma błąd i musi zostać zrestartowany. IBM MQ **musi** mieć BSDS. Aby zminimalizować prawdopodobieństwo wystąpienia problemów podczas restartu, można skonfigurować produkt IBM MQ z dwoma składami BSDSs, każdy z nich rejestruje te same informacje. Korzystanie z podwójnych BSDSs jest znane jako działające w trybie *dual mode*. Jeśli to możliwe, należy umieścić kopie na oddzielnych woluminach. Zmniejsza to ryzyko utraty lub uszkodzenia woluminu, jeśli jest uszkodzony. Należy używać podwójnych BSDSs, a nie podwójnego zapisu do DASD.

Narzędzie BSDS jest skonfigurowane, gdy produkt IBM MQ jest dostosowany i można zarządzać spisem zasobów przy użyciu narzędzia do zarządzania spisem zasobów (CSQJU003). Więcej informacji na temat tego narzędzia znajduje się w sekcji Administrowanie produktem IBM MQ for z/OS. Jest on przywoływany przez instrukcję DD w procedurze uruchamiania menedżera kolejek.

Zwykle produkt IBM MQ przechowuje zduplikowane kopie BSDS. Jeśli wystąpi błąd we/wy, zwalnia ona niesprawną kopię i kontynuuje działanie z pojedynczym BSDS. Można odtworzyć operację podwójnego trybu, która jest opisana w podręczniku Administrowanie produktem IBM MQ for z/OS.

Aktywne dzienniki są najpierw rejestrowane w BSDS, gdy zainstalowany jest produkt IBM MQ . Nie można zastąpić aktywnych dzienników bez zakończenia i restartowania menedżera kolejek.

Zestawy danych dziennika archiwalnego są przydzielane dynamicznie. Po przydzieleniu jednej z nich nazwa zestawu danych jest rejestrowana w BSDS. Lista zestawów danych dziennika archiwalnego jest rozwijana w miarę dodawania archiwów i opakowuje po osiągnięciu określonej przez użytkownika liczby pozycji. Maksymalna liczba pozycji wynosi 1000 dla pojedynczego rejestrowania archiwalnego i 2000 dla rejestrowania podwójnego.

Do usuwania zestawów danych dziennika archiwalnego można użyć systemu zarządzania taśmami (IBM MQ nie ma metody zautomatyzowanej). W związku z tym informacje o zestawie danych dziennika archiwalnego mogą być w BSDS długo po usunięciu zestawu danych dziennika archiwalnego przez administratora systemu.

Z kolei maksymalna liczba zestawów danych dziennika archiwalnego mogła zostać przekroczona, a dane z BSDS zostały usunięte na długo przed osiągniętą datą utraty ważności zestawu danych.

Można użyć następującej komendy MQSC, aby określić zakres dziennika, oraz nazwę aktywnego lub archiwalnego zestawu danych przechowuj najwcześniejszy dziennik RBA, wymagany dla różnych typów odtwarzania nośnika lub menedżera kolejek:

```
DISPLAY USAGE TYPE(DATASET)
```

Jeśli moduł parametru systemowego określa, że zestawy danych dziennika archiwalnego są katalogowane przy przydzielaniu, punkty BSDS do katalogu zintegrowanego narzędzia katalogowego (ICF) są

przeznaczone dla informacji potrzebnych do późniejszego przydziału. W przeciwnym razie pozycje BSDS dla każdego woluminu rejestrują numery seryjne woluminu i informacje o jednostce, które są potrzebne do późniejszego przydziału.

Wersja BSDS


Format BSDS różni się w zależności od jego wersji. Zwiększenie wersji BSDS pozwala na korzystanie z nowych funkcji. Produkt IBM MQ obsługuje następujące wersje BSDS:

Wersja 1

Obsługiwane przez wszystkie wersje produktu IBM MQ. Wersja 1 BSDS obsługuje 6-bajtowe wartości RBA dziennika.

Wersja 2

Obsługiwane przez produkt IBM MQ 8.0 i nowsze. Wersja 2 BSDS umożliwia 8-bajtowe wartości RBA dziennika i maksymalnie 310 zestawów danych w każdej aktywnej kopii dziennika.

 **V 9.2.5** Domyślnie włączone dla menedżerów kolejek utworzonych w systemie IBM MQ 9.2.4 i nowszych.

3

Obsługiwane przez produkt IBM MQ 8.0 i nowsze. BSDS jest automatycznie przekształcany w wersję 3, z wersji 2, gdy do każdej aktywnej kopii dziennika dodawane jest więcej niż 31 zestawów danych.

Wersję BSDS można określić, uruchamiając program narzędziowy do odwzorowywania dziennika drukowania (CSQJU004). Aby przekształcić BSDS z wersji 1 do wersji 2, uruchom program narzędziowy do konwersji BSDS (CSQJUCNV).

Więcej informacji na temat 6-bajtowych i 8-bajtowych dzienników RBAs można znaleźć w sekcji [“Względny adres w bajtach względnego dziennika” na stronie 247](#).

Archiwalne zestawy danych dziennika i kopie BSDS

Za każdym razem, gdy tworzony jest nowy zestaw danych dziennika archiwalnego, tworzona jest również kopia BSDS. Jeśli dziennik archiwum znajduje się na taśmie, BSDS jest pierwszym zestawem danych na pierwszym woluminie wyjściowym. Jeśli dziennik archiwum znajduje się w pamięci DASD, BSDS jest oddzielnym zestawem danych.

Nazwy zestawów danych dziennika archiwalnego i kopii BSDS są takie same, z tą różnicą, że kwalifikator niskiego poziomu nazwy dziennika archiwalnego rozpoczyna się od A, a kopia BSDS rozpoczyna się od litery B, na przykład:

Nazwa dziennika archiwalnego

CSQ.ARCHLOG1.E00186.T2336229. A 0000001

Nazwa kopii BSDS

CSQ.ARCHLOG1.E00186.T2336229. B 0000001

Jeśli podczas kopiowania BSDS wystąpił błąd odczytu, kopia nie zostanie utworzona, zostanie wyświetlony komunikat CSQJ125E, a przesunięcie do nowego zestawu danych dziennika archiwalnego będzie kontynuowane bez kopiowania BSDS.

 **z/OS**

Definiowanie systemu w systemie z/OS

Produkt IBM MQ for z/OS używa wielu domyślnych definicji obiektów i udostępnia przykładowe JCL w celu utworzenia tych obiektów domyślnych. W tym temacie opisano te obiekty domyślne i przykładowe JCL.

Ustawianie parametrów systemowych

W programie IBM MQ for z/OS moduł parametrów systemowych steruje środowiskami rejestrowania, archiwizowania, śledzenia i połączeń, które są używane przez produkt IBM MQ w jego działaniu. Parametry systemowe są określane przez trzy makra assemblera w następujący sposób:

CSQ6SYSP

Parametry systemowe, w tym ustawianie środowiska połączeń i śledzenia.

CSQ6LOGP

Parametry rejestrowania.

CSQ6ARVP

Parametry archiwum dziennika.

Domyślne moduły parametrów są dostarczane wraz z produktem IBM MQ for z/OS. Jeśli te wartości nie zawierają wartości, które mają zostać użyte, można utworzyć własne moduły parametrów, korzystając z przykładu dostarczanego z produktem IBM MQ. Przykład: `thlqual.SCSQPROC (CSQ4ZPRM)`.

Istnieje możliwość zmiany niektórych parametrów systemowych w czasie działania menedżera kolejek. Zapoznaj się z komendami SET SYSTEM, SET LOG i SET ARCHIVE w sekcji [Komendy MQSC](#).

Więcej informacji na temat definiowania można znaleźć w następujących tematach:

- [“Definiowanie obiektów systemowych dla IBM MQ for z/OS” na stronie 251](#)
- [“Strojenie menedżera kolejek w systemie IBM MQ for z/OS” na stronie 256](#)
- [“Przykładowe definicje dostarczone wraz z produktem IBM MQ for z/OS” na stronie 257](#)

Pojęcia pokrewne

[Źródła, z których można wydawać komendy MQSC w systemie z/OS](#)

Zadania pokrewne

[Dostosuj przykładowe zestawy danych inicjowania inicjowania administrowania z/OS](#)

[Konfigurowanie klastrów](#)

[Monitorowanie IBM MQ](#)

Definiowanie obiektów systemowych dla IBM MQ for z/OS

Produkt IBM MQ for z/OS wymaga dodatkowych predefiniowanych obiektów dla aplikacji publikowania/subskrybowania, klastra i sterowania kanałami oraz innych funkcji administrowania systemem.

Obiekty systemowe wymagane przez produkt IBM MQ for z/OS można podzielić na następujące kategorie:

- [Obiekty publikowania/subskrypcji](#)
- [Domyślne obiekty systemowe](#)
- [Obiekty komend systemowych](#)
- [Obiekty administrowania systemem](#)
- [Kolejki kanałów](#)
- [Kolejki klastra](#)
- [Kolejki grupy współużytkowania kolejek](#)
- [Klasy pamięci masowej](#)
- [Definiowanie kolejki niedostarczonych obiektów systemowych](#)
- [Domyślna kolejka transmisji](#)
- [Kolejki wewnętrzne](#)
- [“Kolejka uwierzytelniania kanału” na stronie 256](#)

Obiekty publikowania/subskrybowania

Istnieje kilka obiektów systemowych, które należy zdefiniować, aby można było używać aplikacji publikowania/subskrypcji z produktem IBM MQ for z/OS. Przykładowe definicje są dostarczane wraz z produktem IBM MQ w celu ułatwienia definiowania tych obiektów. Przykłady te są opisane w sekcji [CSQ4INSG](#).

Aby użyć publikowania/subskrybowania, należy zdefiniować następujące obiekty:

- Kolejka lokalna o nazwie SYSTEM.RETAINED.PUB.QUEUE, który jest używany do przechowywania kopii każdej zachowanej publikacji w menedżerze kolejek. Każda pełna nazwa tematu może zawierać maksymalnie jedną zachowaną publikację przechowaną w tej kolejce. Jeśli aplikacje będą korzystały z zachowanych publikacji na wielu różnych tematach lub jeśli zachowane komunikaty publikacji są dużymi komunikatami, wymagania dotyczące pamięci masowej dla tej kolejki powinny być starannie zaplanowane, w tym przypisanie do własnego zestawu stron, jeśli wymagania dotyczące pamięci masowej są duże. Aby zwiększyć wydajność, należy zdefiniować tę kolejkę z typem indeksu MSGID (jak pokazano w dostarczonej przykładowej definicji kolejki).
- Kolejka lokalna o nazwie SYSTEM.DURABLE.SUBSCRIBER.QUEUE, który jest używany do przechowywania trwałej kopii trwałych subskrypcji w menedżerze kolejek. Aby zwiększyć wydajność, należy zdefiniować tę kolejkę z typem indeksu CORRELID (jak pokazano w dostarczonej przykładowej definicji kolejki).
- Kolejka lokalna o nazwie SYSTEM.DURABLE.MODEL.QUEUE, która jest używana jako model dla zarządzanych trwałych subskrypcji.
- Kolejka lokalna o nazwie SYSTEM.NDURABLE.MODEL.QUEUE, który jest używany jako model dla zarządzanych nietrwałych subskrypcji.
- Lista nazw o nazwie SYSTEM.QPUBSUB.QUEUE.NAMELIST, która zawiera listę nazw kolejek monitorowanych przez umieszczony w kolejce interfejs publikowania/subskrybowania.
- Lista nazw o nazwie SYSTEM.QPUBSUB.SUBPOINT.NAMELIST, która zawiera listę obiektów tematów używanych przez umieszczony w kolejce interfejs publikowania/subskrybowania w celu dopasowania obiektów tematu do punktów subskrypcji.
- Temat o nazwie SYSTEM.BASE.TOPIC(temat), który jest używany jako podstawowy temat rozwiązywania atrybutów.
- Temat o nazwie SYSTEM.BROKER.DEFAULT.STREAM, który jest strumieniem domyślnym używanym przez umieszczony w kolejce interfejs publikowania/subskrypcji.
- Temat o nazwie SYSTEM.BROKER.DEFAULT.SUBPOINT, który jest domyślnym punktem subskrypcji RFH2 używanym przez umieszczony w kolejce interfejs publikowania/subskrypcji.
- Temat o nazwie SYSTEM.BROKER.ADMIN.STREAM, który jest strumieniem administracyjnym używanym przez umieszczony w kolejce interfejs publikowania/subskrybowania.
- Subskrypcja o nazwie SYSTEM.DEFAULT.SUB, który jest domyślnym obiektem subskrypcji używanym do udostępniania wartości domyślnych w komendach DEFINE SUB.

Domyślne obiekty systemowe

Domyślne obiekty systemowe są używane do udostępniania atrybutów domyślnych podczas definiowania obiektu i nie określają nazwy innego obiektu, na podstawie którego ma zostać użyta definicja.

Nazwy domyślnych definicji obiektów systemowych zaczynają się od znaków "SYSTEM.DEFAULT" lub "SYSTEM.DEF. " Na przykład domyślna systemowa kolejka systemowa ma nazwę SYSTEM.DEFAULT.LOCAL.QUEUE.

Obiekty te definiują wartości domyślne systemowe dla atrybutów obiektów IBM MQ :

- Kolejki lokalne
- Kolejki modelowe
- Kolejki aliasowe
- Kolejki zdalne
- Procesy
- Listy nazw
- Kanały
- Klasy pamięci masowej
- Informacje uwierzytelniające

Kolejki współużytkowane to specjalny typ kolejki lokalnej, dlatego podczas definiowania kolejki współużytkowanej definicja ta jest oparta na SYSTEM.DEFAULT.LOCAL.QUEUE. Należy pamiętać o dostarczeniu wartości dla nazwy struktury narzędzia CF, ponieważ nie została ona określona w definicji domyślnej. Alternatywnie można zdefiniować własną domyślną definicję współużytkowanej kolejki, która będzie używana jako podstawa dla współużytkowanych kolejek, tak aby wszystkie one dziedziczyły wymagane atrybuty. Należy pamiętać, że konieczne jest zdefiniowanie kolejki współużytkowanej w jednym menedżerze kolejek tylko w grupie współużytkowania kolejki.

Obiekty komend systemowych

Nazwy obiektów komend systemowych zaczynają się od znaków SYSTEM.COMMAND. Obiekty te należy zdefiniować przed użyciem operacji IBM MQ i paneli sterujących do wydawania komend do podsystemu IBM MQ .

Istnieją dwa obiekty komend systemowych:

1. Kolejka wejściowa komend systemowych jest kolejką lokalną, w której komendy są umieszczane przed ich przetworzeniem przez procesor komend IBM MQ . Musi być ona nazywana SYSTEM.COMMAND.INPUT. Alias o nazwie SYSTEM.ADMIN.COMMAND.QUEUE , aby zachować kompatybilność z produktem IBM MQ for Multiplatforms używać ich przez produkty MQ Console i administrative REST API.
2. SYSTEM.COMMAND.REPLY.MODEL jest kolejką modelową, która definiuje kolejkę odpowiedzi komendy systemowej.

Istnieją dwa dodatkowe obiekty do użycia przez produkt IBM MQ Explorer:

- SYSTEM.MQEXPLORER.REPLY.MODEL , kolejka
- SYSTEM.ADMIN.SVRCONN

SYSTEM.REST.REPLY.QUEUE jest kolejką odpowiedzi używaną przez IBM MQ administrative REST API.

Komendy są zwykle wysyłane za pomocą nietrwałych komunikatów, tak więc zarówno obiekty komend systemu powinny mieć atrybut DEFPSIST (NO), aby aplikacje korzystające z nich (w tym dostarczone aplikacje, takie jak program narzędziowy, a także panele kontrolne) były domyślnie otrzymane nietrwałe komunikaty. W przypadku aplikacji, która używa trwałych komunikatów dla komend, należy ustawić atrybut DEFTYPE (PERMDYN) dla kolejki odpowiedzi, ponieważ komunikaty odpowiedzi do takich komend są trwałe.

Obiekty administracyjne systemu

Nazwy obiektów administracyjnych systemu zaczynają się od znaków SYSTEM.ADMIN.

Istnieje siedem obiektów administrowania systemem:

- SYSTEM.ADMIN.CHANNEL.EVENT , kolejka
- SYSTEM.ADMIN.COMMAND.EVENT , kolejka
- SYSTEM.ADMIN.CONFIG.EVENT , kolejka
- SYSTEM.ADMIN.PERFM.EVENT , kolejka
- SYSTEM.ADMIN.QMGR.EVENT , kolejka
- SYSTEM.ADMIN.TRACE.ROUTE.QUEUE , kolejka
- SYSTEM.ADMIN.ACTIVITY.QUEUE , kolejka

Kolejki kanałów

Aby korzystać z rozproszonego kolejkowania, należy zdefiniować następujące obiekty:

- Kolejka lokalna o nazwie SYSTEM.CHANNEL.SYNCQ, który jest używany do obsługi numerów kolejnych i logicznych jednostek roboczych (LUWID) kanałów. Aby zwiększyć wydajność kanałów, należy zdefiniować tę kolejkę z typem indeksu MSGID (jak pokazano w dostarczonej przykładowej definicji kolejki).
- Kolejka lokalna o nazwie SYSTEM.CHANNEL.INITQ, który jest używany dla komend kanału.

Nie można zdefiniować tych kolejek jako kolejek współużytkowanych.

Kolejki klastra

Aby używać klastrów IBM MQ, należy zdefiniować następujące obiekty:

- Kolejka lokalna o nazwie SYSTEM.CLUSTER.COMMAND.QUEUE, który jest używany do przekazywania zmian w repozytorium między menedżerami kolejek. Komunikaty zapisywane w tej kolejce zawierają aktualizacje danych repozytorium, które mają być stosowane do lokalnej kopii repozytorium lub żądania dotyczące danych repozytorium.
- Kolejka lokalna o nazwie SYSTEM.CLUSTER.REPOSITORY.QUEUE, który jest używany do przechowywania trwałej kopii repozytorium.
- Kolejka lokalna o nazwie SYSTEM.CLUSTER.TRANSMIT.QUEUE, która jest kolejką transmisji dla wszystkich miejsc docelowych w klastrze. Ze względu na wydajność należy zdefiniować tę kolejkę z typem indeksu CORRELID (jak pokazano w przykładowej definicji kolejki).

Te kolejki zwykle zawierają dużą liczbę komunikatów.

Nie można zdefiniować tych kolejek jako kolejek współużytkowanych.

Kolejki grup współużytkowania kolejek

Aby korzystać z kanałów współużytkowanych i kolejkowania wewnątrz grupy, należy zdefiniować następujące obiekty:

- Kolejka współużytkowana o nazwie SYSTEM.QSG.CHANNEL.SYNCQ, który jest używany do przechowywania informacji o synchronizacji dla kanałów współużytkowanych.
- Kolejka współużytkowana o nazwie SYSTEM.QSG.TRANSMIT.QUEUE, który jest używany jako kolejka transmisji dla kolejkowania wewnątrz grupy. Jeśli użytkownik jest uruchomiony w grupie współużytkowania kolejek, należy zdefiniować tę kolejkę nawet wtedy, gdy nie jest używana kolejkowanie wewnątrz grupy.

Klasy pamięci masowej

Zaleca się zdefiniowanie następujących sześciu klas pamięci masowej. Należy zdefiniować cztery z nich, ponieważ są one wymagane przez produkt IBM MQ. Zalecane są inne definicje klas pamięci masowej, ponieważ są one używane w przykładowych definicjach kolejek.

DOMYŚLNE (wymagane)

Ta klasa pamięci masowej jest używana dla wszystkich kolejek komunikatów, które nie są newralgiczne dla wydajności i które nie mieszczą się w żadnej innej klasie pamięci masowej. Jest to także podana domyślna klasa pamięci, jeśli podczas definiowania kolejki nie zostanie określona wartość.

NODEFINE (wymagane)

Ta klasa pamięci masowej jest używana, jeśli klasa pamięci masowej określona podczas definiowania kolejki nie jest zdefiniowana.

REMOTE (wymagane)

Ta klasa pamięci jest używana przede wszystkim dla kolejek transmisji, czyli kolejek związanych z systemem z krótkotrzymymi komunikatami o newralgicznym znaczeniu dla wydajności.

SYSLNGLV

Ta klasa pamięci jest używana w przypadku komunikatów o długim okresie życia, komunikatów o newralgicznym znaczeniu dla wydajności.

SYSTEM (wymagany)

Ta klasa pamięci masowej jest używana do wykonywania newralgicznych kolejek komunikatów związanych z systemem, na przykład SYSTEM.CHANNEL.SYNQ i SYSTEM.CLUSTER.* .

SYSVOLAT

Ta klasa pamięci jest używana dla komunikatów o krótkim czasie życia, komunikatów o newralgicznym znaczeniu dla wydajności.

Można modyfikować ich atrybuty i dodawać inne definicje klas pamięci masowej zgodnie z wymaganiami.

Definiowanie systemowej kolejki niedostarczonych komunikatów

Jeśli miejsce docelowe komunikatu jest niepoprawne, używana jest kolejka niedostarczonych komunikatów. Program IBM MQ umieszcza takie komunikaty w kolejce lokalnej nazywanych kolejką niedostarczonych komunikatów. Chociaż posiadanie kolejki niedostarczonych komunikatów nie jest obowiązkowe, należy go traktować jako istotne, szczególnie w przypadku korzystania z rozproszonego kolejkowania lub jednego z mostów IBM MQ .

Nie należy definiować kolejki niedostarczonych komunikatów jako kolejki współużytkowanej. Umieszczenie kolejki w kolejce lokalnej w jednym menedżerze kolejek może zostać umieszczone w kolejce niedostarczonych komunikatów. Jeśli kolejka niewysłanych komunikatów była kolejką współużytkowaną, procedura obsługi kolejki niedostarczanej poczty w innym systemie może przetworzyć komunikat i umieścić ją w kolejce o tej samej nazwie, ale ponieważ jest ona w innym menedżerze kolejek, może to być niepoprawna kolejka lub mieć inny profil zabezpieczeń. Jeśli kolejka nie istnieje, nie powiedzie się jej ponowne przetworzenie.

Jeśli zostanie podjęta decyzja o zdefiniowaniu kolejki niedostarczonych komunikatów, należy również poinformować menedżera kolejek o jego nazwie. W tym celu należy użyć komendy ALTER QMGR DEADQ (*queue-name*). Więcej informacji na ten temat zawiera sekcja [Wyświetlanie i modyfikowanie atrybutów menedżera kolejek](#).

Domyślna kolejka transmisji

Domyślna kolejka transmisji jest używana, gdy żadna inna odpowiednia kolejka transmisji nie jest dostępna do wysyłania komunikatów do innego menedżera kolejek. Jeśli zostanie zdefiniowana domyślna kolejka transmisji, należy również zdefiniować kanał do obsługi kolejki. Jeśli kanał nie zostanie zdefiniowany, komunikaty umieszczane w domyślnej kolejce transmisji nie są przesyłane do zdalnego menedżera kolejek i pozostają w kolejce.

Jeśli zostanie podjęta decyzja o zdefiniowaniu domyślnej kolejki transmisji, należy również poinformować menedżera kolejek o jego nazwie. W tym celu należy użyć komendy ALTER QMGR.

Kolejki wewnętrzne

• Kolejka oczekujących danych

- Kolejka zdefiniowana do użytku wewnętrznego, SYSTEM.PENDING.DATA.QUEUE, obsługuje korzystanie z trwałych subskrypcji w środowisku publikowania/subskrypcji produktu JMS .

• JMS 2.0 kolejka przemieszczania opóźnienia dostarczenia

- Jeśli funkcja opóźnienia dostarczenia udostępniana przez produkt JMS 2.0 jest używana, to wewnętrzna kolejka pomostowa SYSTEM.DDELAY.LOCAL.QUEUE, musi być zdefiniowana. Ta kolejka jest używana przez menedżera kolejek do tymczasowego przechowywania komunikatów wysłanych z niezerowym opóźnieniem dostarczenia do czasu zakończenia opóźnienia dostarczenia, a komunikat jest umieszczany w docelowym miejscu docelowym. W sekcji CSQ4INSG została udostępniona przykładowa definicja dla tej kolejki.

- Po zdefiniowaniu SYSTEM.DDELAY.LOCAL.QUEUE , należy ustawić atrybuty STGCLASS, MAXMSGL i MAXDEPTH dla przewidywanej liczby komunikatów, które zostaną wysłane z opóźnieniem dostawy. Dodatkowo podczas definiowania SYSTEM.DDELAY.LOCAL.QUEUE zapewnia, że tylko menedżer kolejek może umieszczać komunikaty w tej kolejce. Należy zwrócić uwagę na to, aby żaden identyfikator użytkownika nie miał uprawnień do umieszczania komunikatów w tej kolejce.

Kolejka uwierzytelniania kanału

Do wewnętrznego użycia uwierzytelniania kanału SYSTEM.CHLAUTH.DATA.QUEUE jest wymagana. Przykładowe definicje są dostarczane wraz z produktem IBM MQ w celu ułatwienia definiowania tych obiektów. Przykład ten jest opisany w sekcji CSQ4INSA, która definiuje również reguły domyślne.

Strojenie menedżera kolejek w systemie IBM MQ for z/OS

Istnieje kilka prostych kroków, które można wykonać, aby upewnić się, że menedżer kolejek jest dostrojony w celu uniknięcia podstawowych problemów z wydajnością.

Istnieje wiele sposobów poprawy wydajności menedżera kolejek, które są kontrolowane przez atrybuty menedżera kolejek ustawionych za pomocą komendy ALTER QMGR. Ta sekcja zawiera informacje na temat sposobu wykonania tej czynności, ustawiając maksymalną liczbę komunikatów dozwolonych w menedżerze kolejek lub wykonując operacje konserwowania w menedżerze kolejek. Program IBM MQ SupportPac MP16 - WebSphere MQ for z/OS [Planowanie mocy obliczeniowej & strojenie](#) udostępnia więcej informacji na temat wydajności i strojenia.

Punkty synchronizacji

Jedną z ról menedżera kolejek jest kontrola punktu synchronizacji w obrębie aplikacji. Aplikacja konstruuje jednostkę pracy zawierającą dowolną liczbę wywołań MQPUT lub MQGET zakończonych wywołaniem MQCMIT.

Ze względu na to, że liczba wywołań MQPUT lub MQGET w zasięgu jednej operacji MQCMIT zwiększa się znacząco, koszt wykonania zatwierdzenia zwiększa się. Aplikacje w ogóle powinny być zaprojektowane tak, aby nie było MQPUT/MQGET w pojedynczym punkcie synchronizacji.

Korzystając z atrybutu menedżera kolejek MAXUMSGS, można administracyjnie ograniczyć liczbę komunikatów w obrębie dowolnego punktu synchronizacji. Jeśli aplikacja przekroczy ten limit, otrzyma ona wartość MQRC_SYNCPOINT_LIMIT_REACHED w wywołaniu MQPUT, MQPUT1 lub MQGET, która przekracza limit. W takim przypadku aplikacja powinna wydać komendę MQCMIT lub MQBACK.

Wartością domyślną MAXUMSGS jest 10000. Tę wartość można obniżyć, jeśli chcesz wymusić ograniczenie dolnego limitu, co może również pomóc w ochronie przed zapętlaniem aplikacji. Przed zmniejszeniem wartości MAXUMSGS należy upewnić się, że zostały zrozumiane istniejące aplikacje, aby upewnić się, że nie przekraczają limitu, lub mogą tolerować kod powrotu MQRC_SYNCPOINT_LIMIT_OSIĄGNAŁ.

Przedawnione komunikaty

Komunikaty, które utraciły ważność, są odrzucane przez następne odpowiednie wywołanie MQGET. Jeśli jednak takie wywołanie nie zostanie wykonane, komunikaty, które utraciły ważność, nie są usuwane, a w przypadku niektórych kolejek, w szczególności tych, w których pobieranie komunikatów jest wykonywane przez atrybut MessageId, CorrelId lub GroupId, a kolejka jest indeksowana w celu wykonania, wiele komunikatów, które utraciły ważność, mogą być kumulowane. Menedżer kolejek może okresowo skanować dowolną kolejkę dla komunikatów, które utraciły ważność, które następnie są usuwane. Można wybrać, jak często odbywa się to skanowanie, jeśli w ogóle. Istnieją dwa sposoby wykonania tego zadania:

Żądanie jawne

Użytkownik może kontrolować, które kolejki są skanowane i kiedy. Wydać komendę REFRESH QMGR TYPE (WAŻNOŚCI), określając kolejkę lub kolejki, które mają być skanowane.

Skanowanie okresowe

Za pomocą atrybutu EXPRYINT można określić przedział czasu utraty ważności w obiekcie menedżera kolejek. Menedżer kolejek przechowuje informacje na temat komunikatów, które utraciły ważność w każdej kolejce, i wie, w jakim czasie warto przejrzeć komunikaty, które utraciły ważność, co jest warte zachodu. Za każdym razem, gdy upłynie odstęp EXPRYINT, menedżer kolejek wyszukuje kolejki kandydujące, które są warte skanowania komunikatów, które utraciły ważność, i skanuje tylko te kolejki, które uzna za warte zachodu. Nie skanuje wszystkich kolejek. Pozwala to uniknąć marnowania czasu procesora na niepotrzebne skanowanie.

Kolejki współużytkowane są skanowane tylko przez jednego menedżera kolejek w grupie współużytkowania kolejek. Na ogół pierwszy menedżer kolejek do zrestartowania lub pierwszy, który ma ustawiony zestaw EXPRYINT, wykonuje skanowanie.

Uwaga: Należy ustawić tę samą wartość EXPRYINT dla wszystkich menedżerów kolejek w grupie współużytkowania kolejek.

Przykładowe definicje dostarczone wraz z produktem IBM MQ for z/OS

Ten temat zawiera odniesienie do przykładowego kodu JCL i kodu dostarczonego razem z produktem IBM MQ for z/OS.

Poniższe definicje przykładów są dostarczane z produktem IBM MQ w bibliotece thlqual.SCSQPROC . Można ich używać do definiowania obiektów systemowych oraz do dostosowywania własnych obiektów. Niektóre z nich można dołączyć do zestawów danych wejściowych inicjowania (opisanych w sekcji [Komendy inicjalizacji](#)).

| zestaw danych wejściowych inicjowania | Przykładowa nazwa |
|--|---|
| CSQINP1 | CSQ4INP1 CSQ4INPR |
| CSQINP2 | CSQ4INSA CSQ4INYS ¹ CSQ4INSX CSQ4INSG CSQ4INSR CSQ4INSS CSQ4INSJ CSQ4INSM CSQ4INYG CSQ4INYR CSQ4INYC CSQ4INYD CSQ4INSC |
| CSQINPT | CSQ4INST CSQ4INYT |

Tabela 21. Definicje przykładów IBM MQ dla obiektów systemowych (kontynuacja)

| zestaw danych wejściowych inicjowania | Przykładowa nazwa |
|---------------------------------------|---|
| Inne | CSQ4DISP CSQ4INPX CSQ4IVPQ CSQ4IVPG CSQ4MSTR CSQ4MSRR V9.2.0 CSQ4QMIN |

Uwaga:

1. Kolejność tych przykładowych definicji jest istotna: występuje błąd, jeśli INYS, INSX i INSG są nieprawidłowo uporządkowane.

Przykłady CSQINP1

Użyj przykładowego zestawu danych CSQINP1 `thlqual.SCSQPROC(CSQ4INP1)`, gdy dla każdej klasy komunikatu używany jest zestaw stron, lub `thlqual.SCSQPROC(CSQ4INPR)`, gdy dla głównych klas komunikatów używane jest wiele zestawów stron. Zawiera on definicje pul buforów, zestaw stron do powiązań puli buforów oraz komendę ALTER SECURITY. Dołącz przykład do konkatenacji CSQINP1 uruchomionej procedury zadania menedżera kolejek.

Przykłady CSQINP2

Przykładowy obiekt systemowy CSQ4INSG

Przykładowy zestaw danych CSQINP2 o wartości `thlqual.SCSQPROC(CSQ4INSG)` zawiera definicje następujących obiektów systemowych do ogólnego zastosowania:

- Domyślne obiekty systemowe
- Obiekty komend systemowych
- Obiekty administracyjne systemu
- Inne obiekty używane przez system

Należy zdefiniować obiekty w tym przykładzie, ale trzeba to zrobić tylko raz, gdy podsystem zostanie uruchomiony po raz pierwszy. Uwzględnienie definicji w zestawie danych CSQINP2 jest najlepszym sposobem wykonania tego zadania. Są one obsługiwane przez zamknięcie i restartowanie menedżera kolejek. Nie wolno zmieniać nazw obiektów, ale w razie potrzeby można zmieniać ich atrybuty.

Po spełnieniu następujących warunków jeden komunikat jest umieszczany w systemie SYSTEM.DURABLE.SUBSCRIBER.QUEUE (nawet jeśli publikowanie subskrypcji nie jest aktywne):

- Parametr PSMODE atrybutu QMGR jest ustawiony na wartość DISABLED
- Zostanie zaprezentowany przykładowy obiekt CSQ4INST instrukcji DEFINE `SUB(' SYSTEM. DEFAULT. SUB ')`.

Aby tego uniknąć, usuń lub skomentuj instrukcję `DEFINE SUB(' SYSTEM. DEFAULT. SUB ')`.

Kolejka pomostowa opóźnienia dostarczania JMS 2.0, SYSTEM.DDELAY.LOCAL.QUEUE należy zdefiniować tylko wtedy, gdy używane jest opóźnienie dostarczania JMS 2.0. Domyślnie definicja kolejki jest przekształcana w komentarz, którą można usunąć komentarz, jeśli jest to wymagane.

Obiekt systemowy CSQ4INSA i przykład uwierzytelniania

Przykładowy zestaw danych CSQINP2 o wartości thlqual.SCSQPROC(CSQ4INSA) zawiera definicję kolejki systemu uwierzytelniania kanału. Ta kolejka zawiera rekordy uwierzytelniania kanału. Zawiera ona również domyślne reguły uwierzytelniania kanału.

Obiekty w tym przykładzie należy zdefiniować, jeśli parametr CHLAUTH ma wartość ENABLED w menedżerze kolejek, a użytkownik chce uruchomić kanały lub ma zostać ustawiony lub wyświetlony rekord CHLAUTH. Należy je zdefiniować tylko raz, gdy podsystem zostanie uruchomiony po raz pierwszy. Uwzględnienie definicji w zestawie danych CSQINP2 jest najlepszym sposobem wykonania tego zadania. Są one utrzymywane przez zamknięcie i restartowanie menedżera kolejek, nie należy zmieniać nazwy kolejki.

Przykładowy obiekt systemowy CSQ4INSS

Jeśli używane są grupy współużytkowania kolejek, można zdefiniować dodatkowe obiekty systemowe.

Przykładowy zestaw danych thlqual.SCSQPROC(CSQ4INSS) zawiera przykładowe komendy do użycia ze strukturami CF oraz zestaw definicji dla obiektów systemowych wymaganych dla kanałów współużytkowanych i kolejkowania wewnątrz grupy.

Nie można użyć tej próbki w takiej postaci, w której jest ona używana. Należy ją dostosować przed użyciem. Następnie można dołączyć ten element do konkatenacji CSQINP2 DD procedury uruchamiania menedżera kolejek lub użyć go jako danych wejściowych do funkcji COMMAND programu narzędziowego CSQUTIL w celu wydania wymaganych komend.

Podczas definiowania obiektów grupowych lub współużytkowanych należy uwzględnić je w konkatenacji CSQINP2 DD tylko dla jednego menedżera kolejek w grupie współużytkowania kolejek.

Przykładowy obiekt systemowy CSQ4INSX

Należy zdefiniować dodatkowe obiekty systemowe, jeśli używana jest rozproszona kolejkowanie i łączenie w klastry.

Przykładowy zestaw danych thlqual.SCSQPROC(CSQ4INSX) zawiera wymagane definicje kolejek. Ten element można dołączyć do konkatenacji CSQINP2 procedury uruchamiania menedżera kolejek lub można użyć go jako danych wejściowych funkcji COMMAND w programie narzędziowym CSQUTIL w celu wydania wymaganych komend DEFINE.

Istnieją dwa typy definicji obiektów:

- SYSTEM.CHANNEL.xx, wymagany dla każdego rozproszonego kolejkowania
- SYSTEM.CLUSTER.xx, wymagana do grupowania

Przykładowy obiekt systemu JMS CSQ4INSJ

Definiuje kolejki używane w domenie publikowania/subskrypcji produktu JMS .

Przykładowy obiekt systemowy CSQ4INSM

Jeśli używane są zaawansowane zabezpieczenia komunikatów, należy zdefiniować dodatkowe obiekty systemowe. Przykładowy zestaw danych thlqual.SCSQPROC(CSQ4INSM) zawiera wymagane definicje kolejek.

Przykład obiektu CSQ4INSR

Definiuje kolejki używane przez produkt WebSphere Application Server i brokery.

Przykład obiektu CSQ4INYD

Jeśli używany jest rozproszony kolejkowanie, a użytkownik musi skonfigurować własne kolejki, procesy i kanały.

Przykładowy zestaw danych thlqual.SCSQPROC(CSQ4INYD) zawiera przykładowe definicje, których można użyć w celu dostosowania rozproszonych obiektów kolejkowania. Obejmuje ona:

- Zestaw definicji dla zakończenia wysyłania
- Zbiór definicji dla odbierającego końca
- Zestaw definicji do używania klientów

Nie można użyć tej próbki w takiej postaci, w której jest ona-należy ją dostosować przed użyciem. Następnie można dołączyć ten element do konkatenacji CSQINP2 DD procedury uruchamiania menedżera kolejek lub użyć go jako danych wejściowych do funkcji COMMAND programu narzędziowego CSQUTIL w celu wydania wymaganych komend DEFINE. (Jest to zalecane, ponieważ oznacza to, że nie trzeba ponownie definiować tych obiektów przy każdym restarcie menedżera kolejek).

Przykład obiektu CSQ4INYC

W przypadku korzystania z technologii klastrowej definicje równoważne definicji kanałów i definicjom kolejek zdalnych w kolejkach rozproszonych są tworzone automatycznie, gdy jest to potrzebne. Jednak potrzebne są pewne ręczne definicje kanałów-kanał odbiorczy klastra i definicja nadawcy klastra do co najmniej jednego menedżera kolejek repozytorium klastra.

Przykładowy zestaw danych: thlqual.SCSQPROC(CSQ4INYC) zawiera następujące przykładowe definicje, których można użyć do dostosowywania obiektów klastrowych:

- Definicje menedżera kolejek
- Definicje kanału odbierającego
- Definicje kanału wysyłającego
- Definicje kolejek klastra
- Definicje dla list klastrów

Nie można użyć tej próbki w takiej postaci, w której jest ona-należy ją dostosować przed użyciem. Następnie można dołączyć ten element do konkatenacji CSQINP2 DD procedury uruchamiania menedżera kolejek lub użyć go jako danych wejściowych do funkcji COMMAND programu narzędziowego CSQUTIL w celu wydania wymaganych komend DEFINE. Jest to zalecane, ponieważ oznacza to, że nie trzeba ponownie definiować tych obiektów za każdym razem, gdy zostanie zrestartowany produkt IBM MQ.

Przykład obiektu CSQ4INYG

Przykładowy zestaw danych: thlqual.SCSQPROC(CSQ4INYG) zawiera następujące przykładowe definicje, których można użyć do dostosowywania własnych obiektów do ogólnego użytku:

- Kolejka niedostarczonych komunikatów
- Domyślna kolejka transmisji
- Obiekty adaptera CICS

Nie można użyć tej próbki w takiej postaci, w której jest ona-należy ją dostosować przed użyciem. Następnie można dołączyć ten element do konkatenacji CSQINP2 DD procedury uruchamiania menedżera kolejek lub użyć go jako danych wejściowych do funkcji COMMAND programu narzędziowego CSQUTIL w celu wydania wymaganych komend DEFINE. Jest to zalecane, ponieważ oznacza to, że nie trzeba ponownie definiować tych obiektów za każdym razem, gdy zostanie zrestartowany produkt IBM MQ.

Oprócz przykładowych definicji w tym miejscu można użyć definicji obiektów systemowych jako podstawy dla własnych definicji zasobów. Na przykład można utworzyć roboczą kopię systemu SYSTEM.DEFAULT.LOCAL.QUEUE i nazwij ją MY.DEFAULT.LOCAL.QUEUE. Następnie można w razie potrzeby zmienić dowolne parametry w tej kopii. Następnie można wydać komendę DEFINE za pomocą wybranej metody, pod warunkiem, że użytkownik ma uprawnienia do tworzenia zasobów tego typu.

Domyślna kolejka transmisji

Przed podjęciem decyzji o zdefiniowaniu domyślnej kolejki transmisji należy zapoznać się z opisem [Domyślna kolejka transmisji](#).

- Jeśli zdecydujesz, że chcesz zdefiniować domyślną kolejkę transmisji, pamiętaj, że musisz także zdefiniować kanał, który będzie mu służył.
- Jeśli zdecydujesz, że nie chcesz definiować jednego, pamiętaj o usunięciu instrukcji DEFXMLTQ z komendy ALTER QMGR w przykładzie.

Obiekty adaptera CICS

Przykład definiuje kolejkę inicjującą o nazwie CICS01.INITQ. Ta kolejka jest używana przez transakcję CKTI dostarczanej przez produkt IBM MQ. Nazwę tej kolejki można zmienić, jednak musi ona być zgodna z nazwą określoną w tabeli inicjowania systemu CICS (SIT) lub w przesłonięciu SYSIN w instrukcji INITPARM.

Przykłady obiektów CSQ4INYS/CSQ4INYR

Definicje klas pamięci masowej do użycia:

- jeden zestaw stron dla każdej klasy komunikatu
- wiele zestawów stron dla głównych klas komunikatów

Na przykład SYSTEM.COMMAND.INPUT używa parametru STGCLASS ('SYSVOLAT') i SYSTEM.CLUSTER.TRANSMIT.QUEUE używa STGCLASS ('REMOTE'). W katalogu CSQ4INYSobie te klasy pamięci masowej korzystają z tego samego zestawu stron. W CSQ4INYRte klasy pamięci masowej używają różnych zestawów stron w celu zmniejszenia wpływu zapewnienia kolejki transmisji.

Próbki CSQINPT

CSQ4INST

Przykładowy zestaw danych: thlqual.SCSQPROC(CSQ4INST) zawiera definicję domyślnej subskrypcji systemowej.

W tym przykładzie należy zdefiniować obiekt, ale należy go wykonać tylko raz, gdy mechanizm publikowania/subskrypcji zostanie uruchomiony po raz pierwszy. Uwzględnienie definicji w zestawie danych CSQINPT jest najlepszym sposobem wykonania tej czynności. Jest ona obsługiwana przez zamknięcie i restartowanie menedżera kolejek. Nie wolno zmieniać nazwy obiektu, ale w razie potrzeby można zmienić ich atrybuty.

CSQ4INYT

Przykładowy zestaw danych: thlqual.SCSQPROC(CSQ4INYT) zawiera zestaw komend, które mogą być uruchamiane po uruchomieniu mechanizmu publikowania/subskrypcji. W tym przykładzie wyświetlane są informacje o temacie i subskrypcji.

Inne

Przykład wyświetlania CSQ4DISP

Przykładowy zestaw danych: thlqual.SCSQPROC(CSQ4DISP) zawiera zestaw ogólnych komend DISPLAY, które wyświetlają wszystkie zdefiniowane zasoby w menedżerze kolejek. Obejmuje to definicje wszystkich obiektów i definicji produktu IBM MQ, takich jak klasy pamięci masowej i dane śledzenia. Te komendy mogą generować dużą ilość danych wyjściowych. Tego przykładu można użyć w zestawie danych CSQINP2 lub jako dane wejściowe dla funkcji COMMAND programu narzędziowego CSQUTIL.

Przykład CSQ4INPX

Przykładowy zestaw danych: thlqual.SCSQPROC(CSQ4INPX) zawiera zestaw komend, które mogą być wykonywane po każdym uruchomieniu inicjatora kanału. Ten przykład należy dostosować przed użyciem. Następnie można dołączyć go do zestawu danych CSQ4INPX dla inicjatora kanału.

Przykłady CSQ4IVPQ i CSQ4IVPG

Przykładowe zestawy danych: `thlqual.SCSQPROC(CSQ4IVPQ)` i `thlqual.SCSQPROC(CSQ4IVPG)` zawierają zestawy komend DEFINE, które są wymagane do uruchamiania programów weryfikacji instalacji (IVPs).

Te przykłady można uwzględnić w zestawie danych CSQINP2 . Po pomyślnym uruchomieniu programu IVPs nie ma potrzeby ich ponownego uruchamiania przy każdym restartowaniu menedżera kolejek. Oznacza to, że nie ma potrzeby trwałego przechowywania tych przykładów w konkatencji CSQINP2 .

Przykłady CSQ4MSTR i CSQ4MSRR

Poniżej przedstawiono przykładowe procedury zadań uruchomionych dla menedżera kolejek: `thlqual.SCSQPROC(CSQ4MSTR)` i `thlqual.SCSQPROC(CSQ4MSRR)`.

CSQ4MSRR korzysta z CSQ4INYR w konkatencji CSQINP2 , tak aby ważne kolejki były rozmieszczane w różnych zestawach stron.

Można usunąć komentarze, dzięki czemu można użyć karty CSQMINI dla nowo utworzonych menedżerów kolejek, jeśli jest to wymagane.

V 9.2.0 Przykład CSQ4QMIN

Przykładowy zestaw danych QMINI: `thlqual.SCSQPROC(CSQ4QMIN)`.

Szczegółowe informacje na temat zestawu danych QMINI oraz sekcji **TransportSecurity** zawiera sekcja [Zestaw danych QMINI](#) .

z/OS Odtwarzanie i restartowanie w systemie z/OS

Odsyłacze w tym temacie umożliwiają znalezienie informacji na temat funkcji produktu IBM MQ for z/OS na potrzeby restartowania i odtwarzania.

Produkt IBM MQ for z/OS ma stabilne funkcje restartowania i odtwarzania. Informacje na temat odzyskiwania menedżera kolejek po jego zatrzymaniu oraz działania po jego zrestartowaniu można znaleźć w następujących podtematach:

- [“Sposób wprowadzania zmian w danych w produkcie IBM MQ for z/OS” na stronie 262](#)
- [“Sposób zachowania spójności w produkcie IBM MQ for z/OS” na stronie 264](#)
- [“Co dzieje się podczas kończenia w programie IBM MQ for z/OS” na stronie 266](#)
- [“Co dzieje się podczas restartowania i odtwarzania w programie IBM MQ for z/OS” na stronie 268](#)
- [“Sposób rozwiązania wątpliwych jednostek odtwarzania” na stronie 270](#)
- [“Odtwarzanie kolejki współużytkowanej” na stronie 272](#)

Pojęcia pokrewne

[z/OS Działania odtwarzania produktu IBM MQ for z/OS](#)

[Źródła, z których można wydawać komendy MQSC w systemie z/OS](#)

Zadania pokrewne

[Planowanie składowania i odtwarzania](#)

[z/OS administrowaniez/OS](#)

Odsyłacze pokrewne

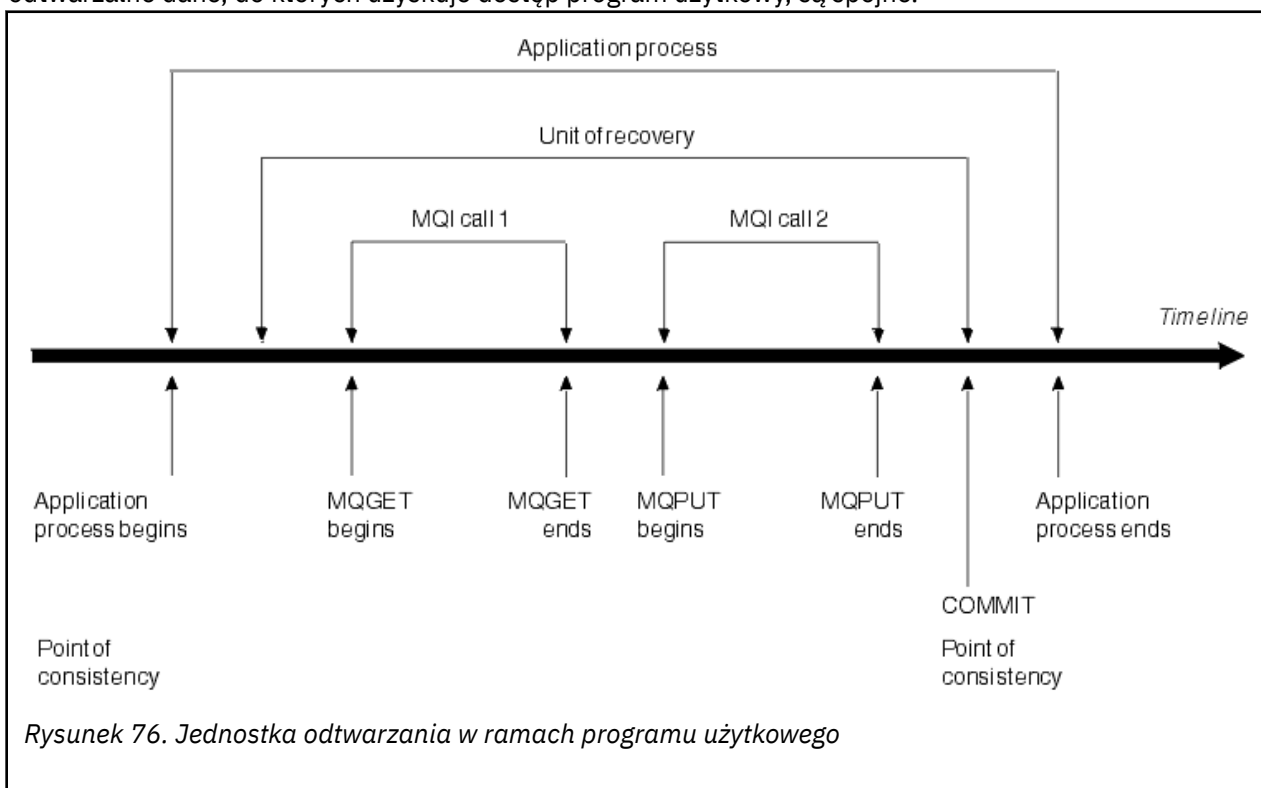
[z/OS Komunikaty dla IBM MQ for z/OS](#)

z/OS Sposób wprowadzania zmian w danych w produkcie IBM MQ for z/OS

Program IBM MQ for z/OS musi współdziałać z innymi podsystemami, aby zachować spójność wszystkich danych. Ten temat zawiera informacje na temat *jednostek odzyskiwania*, informacje o tym, czym są i w jaki sposób są one używane w *wycofanych z nich*.

Jednostki odzyskiwania

Jednostka odzyskiwania to przetwarzanie wykonywane przez pojedynczego menedżera kolejek dla aplikacji, które zmienia dane IBM MQ z jednego punktu spójności na inny. *Punkt spójności* - nazywany również *punktem synchronizacji* lub *punktem zatwierdzenia* - jest to punkt w czasie, gdy wszystkie odtwarzalne dane, do których uzyskuje dostęp program użytkowy, są spójne.



Rysunek 76. Jednostka odtwarzania w ramach programu użytkowego

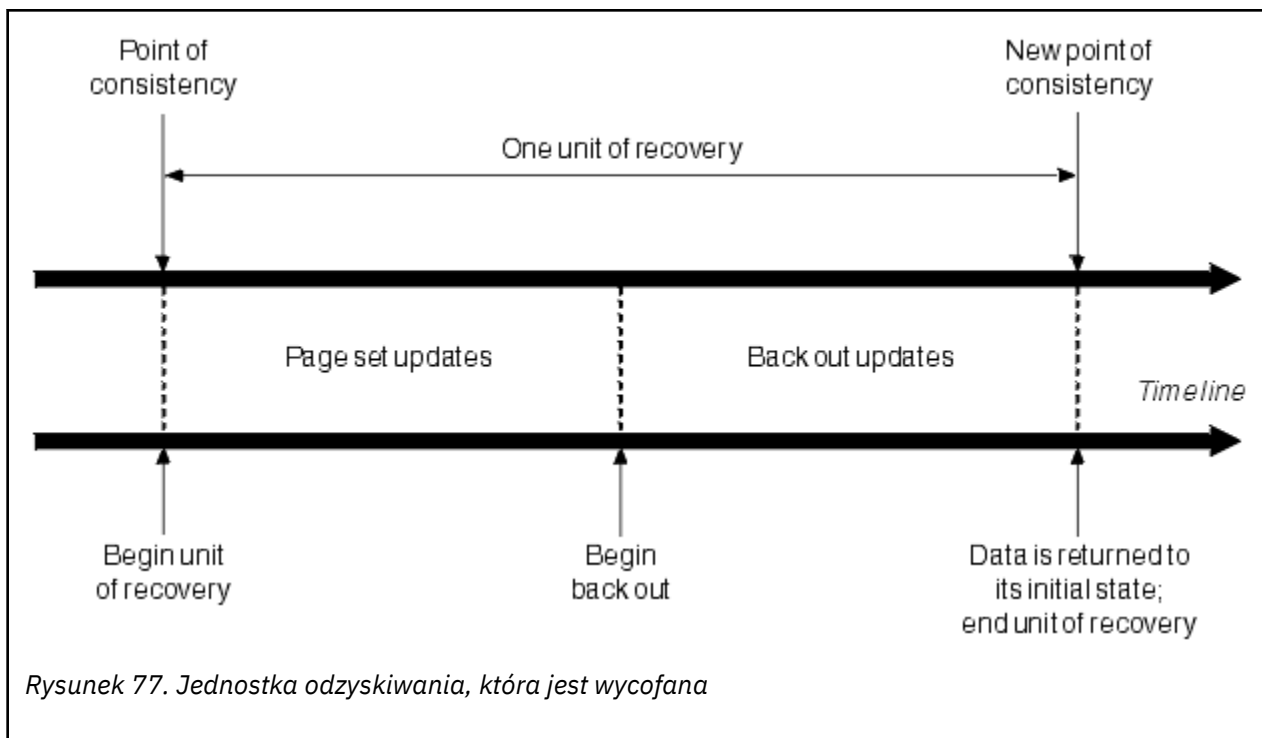
Jednostka odtwarzania rozpoczyna się od pierwszej zmiany danych po rozpoczęciu programu lub po wcześniejszym punkcie spójności; kończy się ona późniejszą końcami spójności. Rysunek 76 na stronie 263 przedstawia relacje między jednostkami odzyskiwania, punktem spójności i programem użytkowym. W tym przykładzie program użytkowy wprowadza zmiany w kolejkach za pomocą wywołań MQI 1 i 2. Program użytkowy może zawierać więcej niż jedną jednostkę odzyskiwania lub tylko jedną jednostkę. Jednak każda kompletna jednostka odtwarzania kończy się w punkcie zatwierdzenia.

Na przykład, transakcja bankowa przekazuje środki z jednego konta na inny. Po pierwsze, program odejmuje kwotę z pierwszego konta, konto A. Następnie dodaje kwotę do drugiego konta, B. Po odejmowaniu kwoty od A dwa konta są niespójne, a program IBM MQ nie może zatwierdzić. Stają się one spójne, gdy kwota zostanie dodana do konta B. Po zakończeniu obu kroków program może ogłosić punkt spójności poprzez zatwierdzenie, co powoduje, że zmiany będą widoczne dla innych aplikacji.

Normalne zakończenie programu użytkowego automatycznie powoduje punkt spójności. Niektóre żądania programów w programach CICS i IMS również powodują spójność, na przykład EXEC CICS SYNCPOINT.

Tworzenie kopii zapasowej pracy

Jeśli w jednostce odtwarzania wystąpi błąd, program IBM MQ usunie wszelkie zmiany danych, zwracając dane do stanu na początku jednostki odzyskiwania, czyli IBM MQ wycofuje pracę. Zdarzenia są wyświetlane w programie Rysunek 77 na stronie 264.



z/OS Sposób zachowania spójności w produkcie IBM MQ for z/OS

Dane w produkcie IBM MQ for z/OS muszą być spójne z danymi wsadowymi, CICS, IMS lub TSO. Wszystkie dane zmienione w jednym muszą być zgodne ze zmianą drugiej strony.

Zanim jeden system zatwierdzi zmienione dane, musi wiedzieć, że inny system może wprowadzić odpowiednią zmianę. Tak więc, systemy muszą się komunikować.

Podczas *zatwierdzania dwufazowego* (na przykład w sekcji CICS) jeden podsystem koordynuje proces. Ten podsystem jest nazywany *koordynatorem*; drugi jest *uczestnikiem*. Produkt CICS lub IMS jest zawsze koordynatorem w interakcji z produktem IBM MQ, a produkt IBM MQ zawsze jest uczestnikiem. W środowisku zadania wsadowego lub TSO produkt IBM MQ może uczestniczyć w protokołach zatwierdzania dwufazowego koordynowanych przez produkt z/OS RRS.

Podczas *zatwierdzania jednofazowego* (na przykład w ramach TSO lub zadania wsadowego), IBM MQ jest zawsze koordynatorem w interakcji i całkowicie kontroluje proces zatwierdzania.

W środowisku WebSphere Application Server semantyka obiektu sesji JMS określa, czy używana jest koordynacja zatwierdzania dwufazowego, czy jednofazowego.

Spójność z produktem CICS lub IMS

Połączenie między IBM MQ a CICS lub IMS obsługuje następujące protokoły synchronizacji:

- Zatwierdzanie dwufazowe-dla transakcji, które aktualizują zasoby należące do więcej niż jednego menedżera zasobów.
Jest to standardowy protokół synchronizacji rozproszonej. Wiąże się to z większym logowaniem i przepływami komunikatów niż zatwierdzanie jednofazowe.
- Zatwierdzanie jednofazowe-dla transakcji, które aktualizują zasoby należące do jednego menedżera zasobów (IBM MQ).

Ten protokół jest zoptymalizowany pod kątem rejestrowania i przepływów komunikatów.

- Pominięcie punktu synchronizacji-dla transakcji, które dotyczą serwera IBM MQ , ale które nie zawierają żadnych elementów w menedżerze kolejek, które wymagają punktu synchronizacji (na przykład przeglądanie kolejki).

W każdym przypadku produkt CICS lub IMS pełni rolę menedżera punktu synchronizacji.

Etapy dwufazowego zatwierdzania, którego program IBM MQ używa do komunikowania się z produktem CICS lub IMS , są następujące:

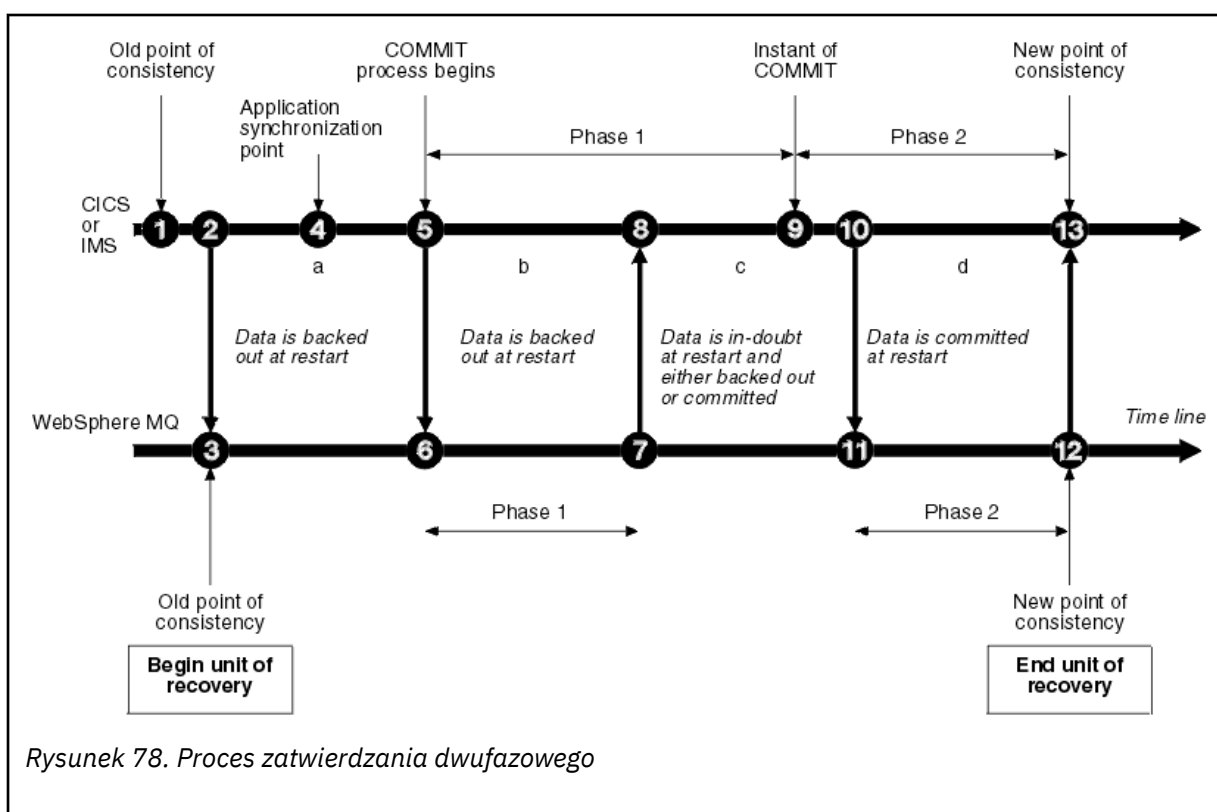
1. W fazie 1 każdy system określa niezależnie od tego, czy w swoim dzienniku zarejestował wystarczającą ilość informacji o odzyskiwaniu, i może zatwierdzić jego pracę.

Na końcu fazy komunikują się systemy. Jeśli się zgodzą, to każdy rozpoczyna następną fazę.

2. W fazie 2 zmiany wprowadzane są na stałe. Jeśli jeden z systemów zakończy działanie w fazie 2, operacja zostanie zakończona przez proces odtwarzania podczas restartu.

Ilustracja procesu zatwierdzania dwufazowego

Rysunek 78 na stronie 265 ilustruje proces zatwierdzania dwufazowego. Zdarzenia w koordynatorze CICS lub IMS są wyświetlane w górnej linii, zdarzenia w programie IBM MQ w dolnej linii.



Rysunek 78. Proces zatwierdzania dwufazowego

Liczby podane w poniższej sekcji są powiązane z numerami przedstawianych na rysunku.

1. Dane w koordynatorowi są w pewnym momencie konsekwencją.
2. Program użytkowy w koordynatorze wywołuje program IBM MQ w celu zaktualizowania kolejki przez dodanie komunikatu.
3. Spowoduje to uruchomienie jednostki odzyskiwania w produkcie IBM MQ.
4. Przetwarzanie jest kontynuowane w koordynatorze do momentu osiągnięcia punktu synchronizacji aplikacji.
5. Następnie koordynator uruchamia przetwarzanie zatwierdzania. Program CICS używa komendy SYNCPOINT lub normalnego zakończenia działania aplikacji w celu uruchomienia zatwierdzenia. Programy IMS mogą uruchamiać zatwierdzanie za pomocą wywołania CHKP, wywołania SYNC, wywołania GET UNIQUE do IOPCB lub normalnego zakończenia aplikacji. Rozpoczyna się faza 1 przetwarzania zatwierdzania.

6. Koordynator rozpoczyna przetwarzanie fazy 1, tak więc IBM MQ.
7. Program IBM MQ pomyślnie kończy fazę 1, zapisuje ten fakt w dzienniku i powiadamia koordynatora.
8. Koordynator otrzymuje powiadomienie.
9. Koordynator pomyślnie kończy przetwarzanie fazy 1. Teraz oba podsystemy wyrażają zgodę na zatwierdzenie zmian danych, ponieważ obie fazy zakończyły fazę 1 i mogą je odzyskać z dowolnego błędu. Koordynator rejestruje w swoim dzienniku natychmiastowe zatwierdzenie-nieodwołalną decyzję dwóch podsystemów w celu wprowadzenia zmian.
Koordynator rozpoczyna obecnie fazę 2 przetwarzania-faktyczne zaangażowanie.
10. Koordynator powiadamia produkt IBM MQ , aby rozpoczął fazę 2.
11. IBM MQ rejestruje początek fazy 2.
12. Faza 2 została pomyślnie zakończona, a jest to nowy punkt spójności dla produktu IBM MQ. IBM MQ następnie powiadamia koordynatora, że zakończył przetwarzanie fazy 2.
13. Koordynator kończy przetwarzanie fazy 2. Dane kontrolowane przez oba podsystemy są teraz spójne i dostępne dla innych aplikacji.

Jak zachowywana jest spójność po nieprawidłowym zakończeniu

Gdy menedżer kolejek jest restartowany po nieprawidłowym zakończeniu, musi określić, czy mają być zatwierdzane, czy wycofano jednostki odzyskiwania, które były aktywne w momencie zakończenia. W przypadku niektórych jednostek odzyskiwania program IBM MQ ma wystarczającą ilość informacji, aby podjąć decyzję. W przypadku innych nie ma ona i musi uzyskać informacje od koordynatora, gdy połączenie zostanie ponownie nawiązane.

Rysunek 78 na stronie 265 przedstawia cztery okresy w obrębie dwóch faz: a, b, c i d. Status jednostki odzyskiwania zależy od okresu, w którym nastąpiło zakończenie. Status może mieć jedną z następujących wartości:

W locie

Menedżer kolejek został zakończony przed zakończeniem fazy 1 (okres a lub b); podczas restartu program IBM MQ wycofuje aktualizacje.

Wątpliwy

Menedżer kolejek zakończył się po zakończeniu fazy 1 i przed rozpoczęciem fazy 2 (okres c); tylko koordynator wie, czy błąd wystąpił przed zatwierdzeniem lub po nim (pkt 9). Jeśli tak się stało, IBM MQ musi wycofać zmiany; jeśli to się stało, IBM MQ musi wprowadzić zmiany i zatwierdzić je. Po restarcie program IBM MQ oczekuje na informacje od koordynatora przed przetworami tej jednostki odzyskiwania.

W zatwierdzeniu

Menedżer kolejek zakończył działanie po rozpoczęciu przetwarzania własnego fazy 2 (okres d). W ten sposób zatwierdzono zmiany.

W wycofaniu

Menedżer kolejek zakończył działanie po wycofaniu jednostki odtwarzania, ale przed zakończeniem procesu (niepokazanego na rysunku) podczas restartu, program IBM MQ kontynuuje wycofanie zmian.

Co dzieje się podczas kończenia w programie IBM MQ for z/OS

Menedżer kolejek kończy działanie normalnie w odpowiedzi na komendę STOP QMGR. Jeśli menedżer kolejek zostanie zatrzymany z jakiegokolwiek innego powodu, to zakończenie jest nieprawidłowe.

Należy zauważyć, że podczas kończenia menedżera kolejek produkt IBM MQ wewnętrznie wysyła komendę

```
DISPLAY CONN(*) TYPE(CONN) ALL WHERE (APPLTYPE NE SYSTEMAL)
```

Użytkownik jest świadomy tego, które wątki mogą uniemożliwić menedżerowi kolejek zakończenie zamykania systemu.

SYSTEMAL jest zgodny z APPLTYPES albo SYSTEM albo CHINIT, więc komenda DISPLAY CONN filtrująca typy aplikacji nie jest zgodna z systemem SYSTEMAL, zwraca do protokołu zadania informacje o wątkach, które mogą uniemożliwić normalne zamknięcie systemu.

Normalne zakończenie

W normalnym zakończeniu program IBM MQ zatrzymuje wszystkie działania w uporządkowany sposób. Program IBM MQ można zatrzymać przy użyciu trybu wyciszenia, wymuszenia lub restartu. Efekty są podane w produkcie Tabela 22 na stronie 267.

| <i>Tabela 22. Zakończenie korzystania z komendy QUIESCE, FORCE i RESTART</i> | | | |
|--|------------------|-------------------|----------------------|
| Typ wątku | QUIESCE | Wymuszenie | RESTARTOWANIE |
| Aktywne wątki | Wykonaj do końca | Wycofanie | Wycofanie |
| Nowe wątki | Można uruchomić | Niedozwolone | Niedozwolone |
| Nowe połączenia | Niedozwolone | Niedozwolone | Niedozwolone |

Aplikacje wsadowe są powiadamiane, jeśli nastąpiło zakończenie, gdy aplikacja jest nadal połączona.

W przypadku produktu CICSbieżący wątek jest uruchamiany tylko na końcu jednostki odzyskiwania. W przypadku produktu CICSzatrzymanie menedżera kolejek w trybie wygaszenia zatrzymuje adapter produktu CICS , a więc jeśli aktywne zadanie zawiera więcej niż jedną jednostkę odtwarzania, to zadanie nie musi być uruchamiane do końca.

Jeśli menedżer kolejek zostanie zatrzymany w trybie wymuszonym lub w trybie restartu, żadne nowe wątki nie są przydzielane, a praca nad połączonymi wątkami jest wycofana. Za pomocą tych trybów można tworzyć wątpliwe jednostki odzyskiwania dla wątków, które są między fazami przetwarzania zatwierdzania. Są one rozstrzygane, gdy produkt IBM MQ jest ponownie połączony z kontrolowaniem podsystemu CICS, IMSlub RRS.

Po zatrzymaniu menedżera kolejek, w dowolnym trybie, kroki są następujące:

1. Połączenia zostały zakończone.
2. Program IBM MQ przestaje akceptować komendy.
3. Produkt IBM MQ zapewnia, że wszystkie zaległe aktualizacje zestawów stron zostaną zakończone.
4. Komenda DISPLAY USAGE jest wydawana wewnętrznie przez produkt IBM MQ , tak aby restartowanie RBA zostało zarejestrowane w dzienniku konsoli z/OS .
5. Punkt kontrolny zamknięcia został zabrany, a BSDS zostanie zaktualizowany.

Terminacje, które określają tryb wygaszenia, nie mają wpływu na wątpliwe jednostki odzyskiwania. Każda jednostka, która jest wątpliwa, pozostaje w wątpliwość.

Nieprawidłowe zakończenie

Nieprawidłowe zakończenie może pozostawić dane w niespójnym stanie, na przykład:

- Jednostka odtwarzania została przerwana przed osiągnięciem punktu spójności.
- Zatwierdzone dane nie zostały zapisane w zestawach stron.
- Dane niezatwierdzone zostały zapisane w zestawach stron.
- Program użytkowy został przerwany między fazą 1 i fazą 2 procesu zatwierdzania, pozostawiając w wątpliwość jednostkę odzyskiwania.

Program IBM MQ rozwiązuje wszelkie niespójności danych wynikające z nieprawidłowego zakończenia działania podczas restartowania i odtwarzania.

Co dzieje się podczas restartowania i odtwarzania w programie IBM MQ for z/OS

Program IBM MQ używa swojego dziennika odtwarzania i zestawu danych programu startowego (BSDS) w celu określenia, co należy odzyskać po restarcie. BSDS identyfikuje aktywne i archiwalne zestawy danych dziennika oraz położenie ostatniego punktu kontrolnego produktu IBM MQ w dzienniku.

Wprowadzenie do restartowania i odtwarzania

Po zainicjowaniu produktu IBM MQ proces restartowania menedżera kolejek odbywa się w następujący sposób:

- Inicjowanie dziennika
- Odbudowywanie bieżącego statusu
- Odtwarzanie dziennika w przód
- Odtwarzanie w dzienniku wstecznym
- Odbudowano indeks kolejki

Po zakończeniu odtwarzania:

- Zatwierdzone zmiany są odzwierciedlane w danych.
- Wątpliwe działanie jest odzwierciedlone w danych. Jednak dane są zablokowane i nie mogą być używane do momentu, gdy produkt IBM MQ rozpoznaje i nie będzie działać na podstawie decyzji wątpliwej.
- Przerwano zmiany w trakcie przetwarzania i w trakcie przerwania zostały usunięte z kolejek. Komunikaty są spójne i mogą być używane.
- Został zabrany nowy punkt kontrolny.
- Nowe indeksy zostały zbudowane dla poindeksowanych kolejek zawierających komunikaty trwałe (opisane w sekcji [“Odbudowanie indeksów kolejek”](#) na stronie 269).

Jeśli podwójne BSDSs są używane, program IBM MQ sprawdza spójność znaczników czasu w BSDS:

- Jeśli obie kopie BSDS są aktualne, program IBM MQ sprawdza, czy te dwa znaczniki czasu są równe. Jeśli nie są one, IBM MQ wysyła komunikat CSQJ120E i kończy działanie. Może się tak zdarzyć, gdy dwie kopie BSDS są obsługiwane na oddzielnych woluminach DASD, a jeden z woluminów został odtworzony w czasie, gdy menedżer kolejek został zatrzymany. Program IBM MQ wykrywa sytuację przy restarcie.
- Jeśli jedna kopia BSDS została odprydziona, a rejestrowanie jest kontynuowane przy użyciu pojedynczego BSDS, może pojawić się problem. Jeśli obie kopie BSDS są obsługiwane na pojedynczym woluminie, a wolumin został odtworzony, lub jeśli oba kopie BSDS zostały odtworzone osobno, produkt IBM MQ może nie wykryć odtwarzania. W takim przypadku rekordy dziennika, które nie zostały odnotowane w BSDS, będą nieznane dla systemu.

Aplikacje wsadowe nie są powiadamiane o ponownym uruchomieniu aplikacji *po* zażądaniu połączenia przez aplikację.

Informacje na temat zakresu dziennika wymaganego do odtworzenia

Podczas restartowania zakres danych dziennika, które muszą być odczytane, zależy od wielu czynników:

- W momencie nieprawidłowego zakończenia pracy w systemie istnieje zazwyczaj wiele niekompletnych jednostek pracy. Jak opisano wcześniej, ponowne uruchomienie przetwarzania spowoduje, że system będzie miał stan spójności, co może wiązać się z wycofanymi jednostkami pracy lub odzyskanymi blokadami na niepewnych jednostkach pracy. Odzyskiwanie jednostki pracy wymaga, aby wszystkie rekordy dziennika pracy dla operacji inflight, in-backout i in-doubt (w przypadku braku wątpliwości) były dostępne. Produkt IBM MQ będzie 'shunt' stare jednostki pracy, dzięki czemu można będzie wykonać odzyskiwanie jednostki pracy przy użyciu znacznie mniejszej liczby danych dziennika.
- W czasie nieprawidłowego zakończenia występuje zwykle wiele trwałych aktualizacji, które są przechowywane tylko w pamięci podręcznej puli buforów. Nie zostały one jeszcze zapisane na dysku.

Zmiany te muszą zostać odczytane z dziennika i ponownie zastosowane do danych znajdujących się w zestawach stron. Zestaw stron RBA odzyskiwania w punkcie kontrolnym opisuje najniższy log RBA wymagany do aktualizacji zestawów stron do spójnego stanu.

- Jeśli w systemie wprowadzono stare zestawy stron, na przykład została wprowadzona kopia zapasowa zestawu stron w celu odzyskania od awarii nośnika, wszystkie zmiany muszą zostać odczytane z dziennika od momentu, gdy została utworzona kopia zapasowa. Zmiany te są ponownie stosowane do danych przechowywanych w odzyskany zestawie stron. Ustawienia RBA odzyskiwania zestawu stron znajdujące się na stronie 0 zestawu stron opisują najniższy poziom RBA dziennika wymagany do odtworzenia nośnika zestawu stron.
- W przypadku korzystania z trwałych komunikatów w kolejkach współużytkowanych należy określić zakres danych dziennika w celu odzyskania CFSTRUCT, które przechowują trwałe komunikaty. Najwcześniejsze dane dziennika, które będą wymagane do wykonania odtwarzania CFSTRUCT, pochodzą z całego czasu starej komendy BACKUP CFSTRUCT.

Podczas normalnego działania można użyć komendy DISPLAY USAGE TYPE (DATASET), aby wyświetlić zakres dziennika odtwarzania powiązany z tymi czynnikami (nie jest on w stanie dostarczyć informacji z powodu ponownego wprowadzenia starych zestawów stron). Aby uniknąć problemów, które mogą przedłużać restartowanie menedżera kolejek w przypadku nieprawidłowego zakończenia, należy regularnie monitorować wartości wyjściowe wartości DISPLAY USAGE TYPE (DATASET).

Ponadto menedżer kolejek wysyła komunikaty informacyjne odnoszące się do następujących czynników:

- CSQJ160I i CSQJ161I ostrzegają o długich, działających jednostkach pracy.
- CSQR026I i CSQR027I udostępniają informacje na temat tego, czy te długie działające jednostki pracy zostały pomyślnie wyśniewane.
- CSQE040I i CSQE041E ostrzegają, że kopie zapasowe struktury są stare, a w konsekwencji operacja RECOVER CFSTRUCT zajmie dużo czasu.

Określanie, która aplikacja ma długo działającą jednostkę pracy

Istnieje możliwość określenia aplikacji z długo działającą jednostką pracy. Aby to zrobić, należy użyć komendy DISPLAY CONN.

Komenda DISPLAY CONN zwraca informacje o połączeniu dla wszystkich aplikacji połączonych z menedżerem kolejek wraz z dodatkowymi informacjami, które ułatwiają określenie, które aplikacje mają obecnie długo działającą jednostkę pracy. Informacje zwracane przez komendę DISPLAY CONN są podobne do informacji zwracanych przez komendę DISPLAY QSTATUS, ale główna różnica polega na tym, że komenda DISPLAY CONN wyświetla informacje o obiektach i informacje transakcyjne dla konkretnego połączenia, a nie szczegóły, z którymi połączenia są powiązane z konkretnym obiektem.

Dla każdej połączonej aplikacji komenda DISPLAY CONN zwraca następujące informacje:

- Podstawowe informacje, w tym identyfikator połączenia i identyfikator PID.
- Informacje transakcyjne dla tego połączenia, w tym czas i datę utworzenia transakcji (to znaczy, gdy pierwsza operacja MQGET/PUT została utworzona w punkcie synchronizacji), a transakcja najpierw zapisana się do dziennika.
- Informacje o czasie rejestrowania wskazujące, które z aplikacji nadal mają długą jednostkę pracy.
- Lista wszystkich obiektów, które są obecnie otwarte przez połączenie. Szczegóły dla każdego obiektu są zwracane jako osobny komunikat, a identyfikator połączenia jest używany jako klucz. Ponieważ istnieją różne typy obiektów, takie jak kolejki i menedżery kolejek, informacje wyświetlane z obiektem są specyficzne dla danego typu obiektu.

Odbudowanie indeksów kolejek

Aby zwiększyć szybkość operacji MQGET w kolejce, w której komunikaty nie są pobierane sekwencyjnie, można określić, że program IBM MQ ma obsługiwać indeks komunikatu lub identyfikatory korelacji lub grupy groupid dla wszystkich komunikatów znajdujących się w tej kolejce.

Po zrestartowaniu menedżera kolejek indeksy te są odbudowywane dla każdej kolejki. Dotyczy to tylko komunikatów trwałych; komunikaty nietrwałe są usuwane przy restarcie. Jeśli poindeksowane kolejki zawierają dużą liczbę trwałych komunikatów, to zwiększa to czas restartowania menedżera kolejek.

Można wybrać, aby indeksy były odbudowywane asynchronicznie do uruchamiania menedżera kolejek przy użyciu parametru QINDXBLD makra CSQ6SYSP . Jeśli wartość QINDXBLD=NOWAIT zostanie ustawiona, program IBM MQ zostanie zrestartowany bez oczekiwania na odbudowanie indeksów.

Sposób rozwiązania wątpliwych jednostek odtwarzania

Jeśli program IBM MQ utraci połączenie z innym menedżerem zasobów, zwykle podejmuje próbę odtworzenia wszystkich niespójnych obiektów przy restarcie.

Jeśli produkt IBM MQ utraci połączenie z serwerem CICS, IMS lub RRS, zwykle podejmuje próbę odtworzenia wszystkich niespójnych obiektów przy restarcie. Informacje wymagane do rozstrzygnięcia w wątpliwych jednostkach odzyskiwania muszą pochodzić z systemu koordynującego. W następujących sekcjach opisano proces rozwiązywania problemów w różnych środowiskach.

- [Jak wątpliwe jednostki odzyskiwania są rozwiązywane z poziomu produktu CICS](#)
- [Jak wątpliwe jednostki odzyskiwania są rozwiązywane z poziomu produktu IMS](#)
- [Jak wątpliwe jednostki odzyskiwania są rozstrzygane z usługi RRS](#)
- [Sposób rozwiązywania wątpliwych jednostek odtwarzania z jednostką grupy, która jest dyspozycyjność odtwarzania](#)

Sposób rozwiązania wątpliwych jednostek odtwarzania w produkcie CICS

W pewnych okolicznościach program CICS nie może uruchomić procesu IBM MQ w celu rozstrzygnięcia wątpliwych jednostek odzyskiwania. W takim przypadku program IBM MQ wysyła jeden z następujących komunikatów:

- CSQC404E
- CSQC405E
- CSQC406E
- CSQC407E

po którym następuje komunikat CSQC408I.

Szczegółowe informacje na temat tego, co oznaczają te komunikaty, zawiera podręcznik [IBM MQ for z/OS](#) , komunikaty, zakończenie, i kody przyczyn .

Rozdzielczość jednostek wątpliwych nie wpływa na zasoby produktu CICS . Produkt CICS jest odpowiedzialny za koordynację odtwarzania, a po restarcie automatycznie zatwierdza lub wycofuje każdą jednostkę w zależności od tego, czy rekord dziennika oznaczał początek zatwierdzenia. Istnienie wątpliwych obiektów nie blokuje zasobów CICS podczas ponownego połączenia z produktem IBM MQ .

Jedną z funkcji adaptera CICS jest zachowanie synchronizacji danych między CICS a IBM MQ. Jeśli menedżer kolejek kończy się podczas połączenia z serwerem CICS, to program CICS może zatwierdzić lub wycofać pracę bez tego, że produkt IBM MQ jest tego świadomy. Po zrestartowaniu menedżera kolejek ta praca jest *wątpliwa*.

Produkt IBM MQ nie może rozstrzygnąć tych wątpliwych jednostek odtwarzania (czyli zatwierdzić lub wycofać zmiany wprowadzone w zasobach IBM MQ) do momentu zrestartowania lub ponownego nawiązania połączenia z produktem CICS .

Proces rozstrzygania wątpliwych jednostek odtwarzania jest inicjowany podczas uruchamiania adaptera CICS . Proces ten rozpoczyna się, gdy adapter zażąda listy wątpliwych jednostek odzyskiwania. Następnie:

- Adapter odbiera listę niepewnych jednostek odzyskiwania dla tego identyfikatora połączenia z produktu IBM MQ i przekazuje je do produktu CICS w celu ich rozwiązania.

- Program CICS porównuje pozycje z tej listy z wpisami w swoim własnym dzienniku. Produkt CICS określa na podstawie własnej listy działania, które zostało podjęte dla każdej wątpliwej jednostki odzyskiwania.

W przypadku wszystkich jednostek rozstrzygniętych program IBM MQ aktualizuje kolejki odpowiednio do potrzeb i zwalnia odpowiednie blokady. Nierozstrzygnięte jednostki mogą pozostać po restarcie. Rozwiąż je za pomocą metod opisanych w [Administrowanie produktem IBM MQ for z/OS](#).

Sposób rozwiązania wątpliwych jednostek odtwarzania w produkcie IMS

Rozstrzygnięcie wątpliwych jednostek odtwarzania w produkcie IMS powoduje wpływ na zasoby DL/I. Produkt IMS jest odpowiedzialny za koordynację odtwarzania, a po zrestartowaniu automatycznie zatwierdza lub wycofuje niepełne działanie DL/I. Decyzja o zatwierdzeniu lub wycofaniu zmian dla regionów internetowych (ścieżka niekrótka) dotyczy obecności lub braku typów rekordów dziennika produktu IMS X'3730 'i X'3801'. Istnienie niepewnych jednostek odtwarzania nie oznacza, że rekordy DL/I są zablokowane do czasu nawiązania połączenia przez program IBM MQ .

Podczas restartu menedżera kolejek produkt IBM MQ tworzy listę wątpliwych jednostek odtwarzania. Program IMS tworzy własną listę pozostałych pozycji odzyskiwania (RREs). Elementy RREs są rejestrowane w punktach kontrolnych programu IMS do momentu rozstrzygnięcia wszystkich pozycji.

Podczas ponownego połączenia regionu IMS z serwerem IBM MQ IMS wskazuje, że IBM MQ ma być zatwierdzane lub wycofane jednostki pracy oznaczone jako wątpliwe w IBM MQ .

Gdy jednostki wątpliwe są rozstrzygane:

1. Jeśli program IBM MQ rozpoznaje, że został oznaczony jako wpis dla zatwierdzenia, a program IMS oznaczył go jako wycofany, program IBM MQ generuje komunikat CSQQ010E. Program IBM MQ wysłał ten komunikat pod kątem wszystkich niespójności tego typu między IBM MQ a IMS.
2. Jeśli w produkcie IBM MQ istnieją jakiegokolwiek pozostałe wątpliwe jednostki, adapter wysłał komunikat CSQQ008I.

W przypadku wszystkich jednostek rozstrzygniętych program IBM MQ aktualizuje w razie potrzeby kolejki i zwalnia odpowiednie blokady.

Produkt IBM MQ utrzymuje blokady w wątpliwych pracach, które nie zostały rozstrzygnięte. Może to spowodować zaleganie w systemie, jeśli wstrzymane są ważne blokady. Połączenie pozostaje aktywne, dzięki czemu możliwe jest rozstrzygnięcie RREs IMS . Odtwórz wątki wątpliwe zgodnie z metodami opisanymi w [Administrowanie produktem IBM MQ for z/OS](#).

Wszystkie wątpliwe prace powinny zostać rozstrzygnięte, chyba że wystąpią problemy z oprogramowaniem lub pracami, np. z zimnym uruchomieniem IMS . Rozstrzygnięcie wątpliwe przez region sterujący IMS ma miejsce w dwóch sytuacjach:

1. Na początku połączenia z produktem IBM MQ, podczas którego rozstrzygnięcie odbywa się synchronicznie.
2. Gdy program kończy działanie, podczas którego rozdzielczość jest wykonywana asynchronicznie.

Sposób rozwiązania wątpliwych jednostek odtwarzania z usługi RRS

Jedną z funkcji adaptera RRS jest zachowanie synchronizacji danych między programem IBM MQ a innymi menedżerami zasobów uczestniczącymi w produkcie RRS. Jeśli wystąpi awaria, gdy program IBM MQ zakończył fazę jednego z zatwierdzeń i oczekuje na decyzję RRS (koordynatora zatwierdzenia), jednostka odtwarzania wchodzi w stan wątpliwy.

Po ponownym nawiązaniu komunikacji między RRS i IBM MQ RRS automatycznie zatwierdza lub wycofuje każdą jednostkę odzyskiwania, w zależności od tego, czy rekord dziennika oznaczał początek zatwierdzenia. Produkt IBM MQ nie może rozstrzygnąć tych wątpliwych jednostek odtwarzania (czyli zatwierdzić lub wycofać zmiany wprowadzone w zasobach IBM MQ), dopóki nie zostanie ponownie nawiązane połączenie z usługą RRS.

W pewnych okolicznościach RRS nie może rozstrzygnąć wątpliwych jednostek odzyskiwania. W takim przypadku program IBM MQ wysyła jeden z następujących komunikatów do konsoli produktu z/OS :

- CSQ3011I
- CSQ3013I
- CSQ3014I
- CSQ3016I

Szczegółowe informacje na temat tego, co oznaczają te komunikaty, zawiera podręcznik [IBM MQ for z/OS](#), komunikaty, zakończenie, i kody przyczyn .

W przypadku wszystkich rozstrzygniętych jednostek odzyskiwania program IBM MQ aktualizuje kolejki odpowiednio do potrzeb i zwalnia odpowiednie blokady. Nierozstrzygnięte jednostki odzyskiwania mogą pozostać po restarcie. Rozwiąż je za pomocą metody opisanej w [Administrowanie produktem IBM MQ for z/OS](#).

Sposób rozwiązania wątpliwych jednostek odtwarzania z jednostką grupy, która jest dyspozycją odtwarzania.

Transakcje wątpliwe, które mają dyspozycję odzyskiwania jednostki GROUP, mogą zostać rozstrzygnięte przez koordynatora transakcji przez dowolnego menedżera kolejek w grupie współużytkowania kolejek (QSG), w którym włączony jest atrybut menedżera kolejek GROUPUR. Za każdym razem, gdy koordynator transakcji ponownie nawiąże połączenie, zwykle żąda listy wszystkich zaległych transakcji wątpliwych, a następnie rozwiązuje je przy użyciu informacji pochodzących z dzienników.

Gdy koordynator transakcji, który połączył się z jednostką grupy o dyspozycyjności, żąda listy wątpliwych transakcji, lista ta zawiera wszystkie wątpliwe transakcje z jednostką GROUP, która istnieje w całej grupie współużytkowania kolejki. Ta lista nie jest zależna od menedżera kolejek, w którym zostały uruchomione transakcje wątpliwe. Menedżer kolejek przetwarzający takie żądanie kompiluje listę, komunikując się ze wszystkimi innymi aktywnymi menedżerami kolejek w grupie współużytkowania kolejek za pomocą SYSTEM.QSG.UR.RESOLUTION.QUEUE. Menedżer kolejek odczytuje następnie dzienniki wszystkich nieaktywnych menedżerów kolejek, poczynawszy od ostatniego punktu kontrolnego, w celu zidentyfikowania dodatkowych transakcji wątpliwych, które zostały zgłoszone przez ich menedżera kolejek, gdyby były aktywne.

Gdy koordynator transakcji żąda rozstrzygnięcia transakcji wątpliwej, menedżer kolejek, z którym jest on połączony, określa, czy transakcja została zainicjowana sama w sobie, a jeśli tak to rozwiąże ją w taki sam sposób, jak transakcje z jednostką QMGR, która ma być usposobienie do odtwarzania. Jeśli transakcja została zainicjowana w innym aktywnym menedżerze kolejek w QSG, żądanie zakończenia tej rozdzielczości jest kierowane do tego menedżera kolejek przy użyciu SYSTEM.QSG.UR.RESOLUTION.QUEUE. W przypadku, gdy transakcja została zainicjowana w nieaktywnym menedżerze kolejek w QSG, wszelkie prace związane z kolejką współużytkowaną są rozwiązywane natychmiast, a żądanie rozwiązania wszelkich pozostałych zadań kolejki prywatnej jest umieszczane w systemie SYSTEM.QSG.UR.RESOLUTION.QUEUE. Nieaktywny menedżer kolejek przetwarza to żądanie po uruchomieniu, zanim zostanie zaakceptowanie nowej pracy. W tym scenariuszu dzienniki pierwotnego menedżera kolejek nadal odzwierciedlają to, że jednostka odtwarzania jest wątpliwa do momentu zrestartowania i przetworzenia żądania.

Odtwarzanie kolejki współużytkowanej

W tym temacie opisano odtwarzanie i odporność produktu IBM MQ na różne komponenty w środowisku grupy współużytkowania kolejek.

- [“Odtwarzanie transakcyjne” na stronie 273](#)
- [“Odtwarzanie równorzędne” na stronie 273](#)
- [“Definicje kolejek współużytkowanych” na stronie 273](#)
- [“Rejestrowanie” na stronie 274](#)

- [“Awaria narzędzia CF i struktury” na stronie 274](#)
- [“Scenariusze awarii struktury” na stronie 275](#)
- [“Odporność na awarie połączeń narzędzia CF” na stronie 276](#)
- [“Zarządzanie odpornością na awarie połączeń narzędzia CF” na stronie 277](#)
- [“Zachowanie operacyjne” na stronie 279](#)

Odtwarzanie transakcyjne

Gdy aplikacja wysyła wywołanie MQBACK lub kończy działanie w sposób nieprawidłowy (na przykład z powodu operacji EXEC CICS ROLLBACK lub IMS abend), informacje na poziomie wątku zapisane w menedźerze kolejek zapewniają, że wycofana jednostka pracy jest wycofana. Operacje MQPUT i MQGET operacje w punkcie synchronizacji w kolejkach współużytkowanych są wycofywane w taki sam sposób, jak aktualizacje kolejek niewspółużytkowanych.

Odtwarzanie równorzędne

Jeśli menedźer kolejek nie powiedzie się, rozłącza się w sposób nieprawidłowy ze struktury narzędzia CF, z którą jest obecnie połączony. Jeśli połączenie między instancją produktu z/OS i narzędziem sprzęgającym nie powiedzie się (na przykład awaria łącza fizycznego lub wyłączenie zasilania narzędzia lub partycji sprzęgającej), to jest ono również wykrywane jako nieprawidłowe zakończenie połączenia między menedźerem kolejek i strukturami narzędzia CF. Inne menedżery kolejek w tej samej grupie współużytkującej kolejkę, które pozostają połączone z tą strukturą, wykrywają nieprawidłowe rozłączenie i wszystkie próby zainicjowania *odtworzenia równorzędnego* dla menedżera kolejek, który uległ awarii, w tej strukturze. Tylko jeden z tych menedżerów kolejek inicjuje odtwarzanie równorzędne pomyślnie, ale wszystkie pozostałe menedżery kolejek współpracują przy odzyskiwaniu jednostek pracy, których właścicielem był menedźer kolejek, który uległ awarii.

Jeśli menedźer kolejek nie powiedzie się, jeśli nie ma żadnych węzłów połączonych ze strukturą, odtwarzanie jest wykonywane, gdy inny menedźer kolejek łączy się z tą strukturą lub gdy menedźer kolejek, który się nie powiódł, zostanie zrestartowany.

Odtwarzanie równorzędne, nazywane często odtwarzaniem poziomu węzła (Peer Level Recovery-PLR), jest wykonywane na podstawie struktury według struktury i możliwe jest, aby pojedynczy menedźer kolejek mógł uczestniczyć w odtwarzaniu więcej niż jednej struktury w tym samym czasie. Jednak zbiór rówieśników współpracujący przy odzyskiwaniu różnych struktur może się różnić w zależności od tego, które menedżery kolejek były połączone z różnymi strukturami w momencie awarii.

Po zrestartowaniu menedżera kolejek następuje ponowne połączenie ze strukturami, z którymi był połączony w momencie niepowodzenia, i odzyskuje wszystkie nierozstrzygnięte nierozstrzygnięte jednostki pracy, które nie zostały odzyskane przez odtwarzanie równorzędne.

Odtwarzanie równorzędne jest procesem wielofazowym. W pierwszym etapie odzyskiwać są jednostki pracy, które postępowywały poza fazą w trakcie lotu; może to wiązać się z zatwierdzaniem komunikatów dla jednostek pracy, które są w trakcie zatwierdzania i blokowania komunikatów dla jednostek pracy, które są wątpliwe. Podczas drugiej fazy sprawdzane są kolejki, w których aktywne są wątki aktywne w menedźerze kolejek, w którym wystąpił błąd, komunikaty niezatwierdzone związane z jednostkami pracy są wycofywane, a informacje o aktywnych uchwytach w kolejkach współużytkowanych w menedźerze kolejek, które się nie powiódł, są resetowane. Oznacza to, że produkt IBM MQ resetuje wszystkie indykatory, dla których menedźer kolejek, w którym wystąpił błąd, miał otwartą kolejkę współużytkowaną na wyłączność wejściową, co umożliwia innym aktywnym menedżerom kolejek otwarcie kolejki na potrzeby wprowadzania danych.

Definicje kolejek współużytkowanych

Obiekty kolejek, które reprezentują atrybuty kolejki współużytkowanej, są przechowywane we współużytkowanym repozytorium Db2 używanym przez grupę współużytkowania kolejek. Należy upewnić się, że istnieją odpowiednie procedury tworzenia i odtwarzania kopii zapasowych tabel produktu Db2 używanych do przechowywania obiektów produktu IBM MQ. Za pomocą programu narzędziowego IBM MQ CSQUTIL można również tworzyć komendy MQSC służące do powtarzania w menedżerze kolejek w celu ponownego zdefiniowania obiektów produktu IBM MQ, w tym definicji kolejek współużytkowanych i grup przechowywanych w produkcji Db2.

Rejestrowanie

Grupy współużytkowania kolejek mogą obsługiwać komunikaty trwałe, ponieważ komunikaty w kolejkach współużytkowanych mogą być protokołowane w dziennikach menedżera kolejek.

Awaria narzędzia CF i struktury

Istnieją dwa typy awarii, które mogą być zgłaszane dla struktury CF (Coupling Facility): awaria struktury i utrata połączenia. Usługa Sysplex services for data sharing (XES) informuje IBM MQ o awarii struktury CF lub awarii systemu CF ze zdarzeniem niepowodzenia struktury. Jeśli XES tworzy zdarzenie utraty połączenia, to nie musi oznaczać, że istnieje problem ze strukturą. Może się okazać, że nie ma dostępnego połączenia, aby komunikować się ze strukturą. Możliwe jest, że nie wszystkie menedżery kolejek odbierają zdarzenie utraty połączenia dla struktury; zależy to od konfiguracji połączeń z systemem CF. Zdarzenie utraty połączenia może również zostać odebrane z powodu komend operatora, na przykład VARY PATH OFFLINE lub CONFIG CHP OFFLINE.

Struktury systemu CF, które są używane przez produkt IBM MQ, można skonfigurować w taki sposób, aby korzystały z funkcji dupleksowania zarządzanego przez system. Oznacza to, że jeśli wystąpi pojedyncza awaria, przetwarzanie przełączenia awaryjnego zarządzanego przez system ukrywa awarię struktury lub utratę połączenia, a menedżer kolejek nie jest informowany o niepowodzeniu. Jeśli wystąpi awaria obu instancji dupleksowej struktury lub połączenia, menedżer kolejek odbierze odpowiednie zdarzenie i obsługuje je w taki sam sposób, jak zdarzenie niepowodzenia dla struktury simplex. Szczegóły dotyczące sposobu, w jaki menedżer kolejek obsługuje zdarzenia, są opisane w sekcji [Scenariusze](#).

W mało prawdopodobnym przypadku awarii struktury CF lub struktury wszystkie nietrwałe komunikaty zapisane w strukturach aplikacji, których dotyczy problem, są tracone. Trwałe komunikaty można odzyskać przy użyciu komendy RECOVER CFSTRUCT. Jeśli odtwarzalna struktura aplikacji nie powiodła się, wszelkie dalsze działania aplikacji w tej strukturze są zablokowane do czasu odtworzenia struktury.

Aby upewnić się, że można odzyskać strukturę systemu CF w rozsądnym czasie, należy wykonać częste tworzenie kopii zapasowych za pomocą komendy BACKUP CFSTRUCT. Można wybrać opcję wykonania kopii zapasowych wszystkich menedżerów kolejek w grupie współużytkowania kolejek lub dedykować jeden menedżer kolejek w celu wykonania wszystkich kopii zapasowych. Zautomatyzuj proces tworzenia kopii zapasowych, aby upewnić się, że są one wykonywane regularnie.

Każda kopia zapasowa jest zapisywana w aktywnym zestawie danych dziennika menedżera kolejek, w którym znajduje się kopia zapasowa. Repozytorium współużytkowanej kolejki Db2 rejestruje nazwę kopii zapasowej struktury CF, nazwę menedżera kolejek wykonujących tworzenie kopii zapasowej, zakres RBA dla tej kopii zapasowej w dzienniku menedżera kolejek oraz czas utworzenia kopii zapasowej.

Struktura administracyjna zawiera informacje na temat niekompletnych jednostek pracy w kolejkach współużytkowanych w momencie niepowodzenia struktury aplikacji, dlatego struktura administracyjna musi być dostępna podczas przetwarzania komendy RECOVER CFSTRUCT. Jeśli struktura administracyjna nie powiedzie się, wszystkie menedżery kolejek w grupie współużytkującej kolejkę muszą przed wydaniem komendy RECOVER CFSTRUCT przebudować ich pozycje struktury administracyjnej.

Menedżery kolejek odbudowują swoje pozycje struktury administracyjnej automatycznie i bez zakończenia. Jeśli menedżer kolejek nie jest uruchomiony w momencie niepowodzenia, jego pozycje struktury administracyjnej mogą zostać odbudowane przez innego menedżera kolejek w grupie współużytkowania kolejek, która działa na tym samym lub wyższym poziomie.

Aby odtworzyć strukturę aplikacji, wydaj komendę RECOVER CFSTRUCT do menedżera kolejek, który ma być odtwarzający. Można odtworzyć pojedynczą strukturę systemu CF lub jednocześnie odtworzyć kilka struktur CF. Można odtworzyć za pomocą dowolnego menedżera kolejek w grupie współużytkownika kolejek, ale nie musi to być ten, który wykonał kopię zapasową, lub taki, który był wcześniej połączony ze strukturą zakończonej niepowodzeniem.

Komenda RECOVER CFSTRUCT korzysta z kopii zapasowej, która znajduje się za pomocą informacji o repozytorium Db2 (Db2 musi być więc dostępna w menedżerze kolejek, w którym wykonywana jest operacja odtwarzania), a następnie odzyskuje to miejsce w punkcie niepowodzenia.

Komenda RECOVER CFSTRUCT umożliwia zastosowanie rekordów dziennika z każdego menedżera kolejek w grupie współużytkownika kolejek, która wykonała operację MQPUT lub MQGET między rozpoczęciem tworzenia kopii zapasowej a czasem niepowodzenia, do dowolnej współużytkowanej kolejki odwzorowanej na strukturę CF. W wyniku scalenia dzienników może być wymagane odczytywanie znacznej ilości danych dziennika, ponieważ wszystkie dane dziennika zapisywane przez uczestniczące menedżery kolejek od momentu utworzenia kopii zapasowej są odczytywane. Stanowczo zaleca się częstsze (na przykład godzinę) tworzenie kopii zapasowych, zwłaszcza jeśli w kopii zapasowej są duże komunikaty.

Scenariusze awarii struktury

Scenariusze

Jeśli dla struktury CF zostanie zgłoszone niepowodzenie, działanie wykonywane przez połączone menedżery kolejek jest zależne od następujących elementów:

- Typ awarii raportowana przez komponent XES produktu z/OS do produktu IBM MQ.
- Typ struktury (aplikacja lub administracja)
- Poziom menedżera kolejek
- CFLEVEL obiektu CFSTRUCT IBM MQ (2, 3, 4 lub 5. Nie jest to CFLEVEL mikrokodu CFCC)
- Atrybut RECAUTO obiektu CFSTRUCT IBM MQ na poziomie CFLEVEL (5)

W poniższych scenariuszach opisano, co się dzieje, gdy dla struktury administracyjnej zostanie zgłoszone niepowodzenie:

- Jeśli zdarzenie niepowodzenia struktury zostanie odebrane dla struktury administracyjnej, struktura zostanie ponownie przydzielona i odbudowana automatycznie bez zakończenia menedżera kolejek. Nowa instancja struktury jest przydzielana przez XES, gdy menedżer kolejek próbuje połączyć się z nim.

Gdy menedżer kolejek połączy się z nową instancją struktury, menedżer kolejek zapisuje pozycje dla siebie w strukturze. To przetwarzanie jest wykonywane przez menedżer kolejek i nie jest częścią procesu odbudowywania XES.

Jeśli menedżer kolejek nie był uruchomiony w momencie wystąpienia błędu lub został zakończony przed zakończeniem odtwarzania jego części struktury administracyjnej, jego pozycje struktury administracyjnej mogą zostać odbudowane przez innego menedżera kolejek w grupie współużytkownika kolejek.

Pozycje struktury administracyjnej menedżera kolejek mogą być odbudowywane tylko przez innego menedżera kolejek działającego na tym samym poziomie lub wyższym. Jeśli pozycje struktury administracyjnej menedżera kolejek nie mogą zostać odbudowane przez innego menedżera kolejek w grupie współużytkownika kolejek, zrestartuj menedżer kolejek, aby mógł zakończyć odbudowywanie jego części struktury.

Niektóre działania są zawieszane do czasu, aż zostaną odbudowane pozycje struktury administracyjnej dla wszystkich menedżerów kolejek. Zawieszane działania obejmują:

- Otwieranie i zamykanie kolejek współużytkowanych.
- Zatwierdzanie lub wycofywanie jednostek odzyskiwania.
- Aplikacje serializowane łączące się z menedżerem kolejek lub odłączające się od niego.

- Tworzenie kopii zapasowej lub odtwarzanie struktury aplikacji.

Wszystkie aplikacje serializowane, które są już połączone z menedżerem kolejek, mogą kontynuować przetwarzanie. Każda aplikacja przekształcona do postaci szeregowej, która próbuje połączyć się z parametrami MQCNO_SERIALIZE_CONN_TAG_QSG lub MQCNO_RESTRICT_CONN_TAG_QSG, otrzymuje kod powrotu MQRC_CONN_TAG_NOT_USABLE.

Gdy pozycje struktury administracyjnej dla menedżera kolejek zostały odbudowane, zawieszona działająca są wznawiane.

W poniższych scenariuszach opisano, co się dzieje, gdy dla struktury aplikacji zostanie zgłoszone niepowodzenie:

- Jeśli dla struktury aplikacji zostanie odebrane zdarzenie niepowodzenia struktury, a wartość CFLEVEL wynosi 1 lub 2, menedżer kolejek kończy działanie. Zrestartuj menedżer kolejek. Pierwszy menedżer kolejek, który próbuje połączyć się ze strukturą ponownie powoduje, że XES przydziela nową instancję struktury.
- Jeśli dla struktury aplikacji zostanie odebrane zdarzenie niepowodzenia struktury, a poziom CFLEVEL to 3, 4 lub 5, to menedżery kolejek połączone ze strukturą będą nadal działać. Aplikacje, które nie korzystają z kolejek w strukturze zakończonej niepowodzeniem, mogą kontynuować normalne przetwarzanie.

Jednak aplikacje, które próbują wykonać operacje na kolejkach w strukturze zakończonej niepowodzeniem, odbierają błąd MQRC_CF_STRUC_FAILED, dopóki struktura nie zostanie pomyślnie odbudowana. W tym momencie aplikacja może ponownie otworzyć kolejki.

Odbudowywanie struktury jest uruchamiane automatycznie dla struktur aplikacji CFLEVEL (5) zdefiniowanych za pomocą RECAUTO (YES). W przeciwnym razie, struktura zostanie odbudowana po wydaniu komendy RECOVER CFSTRUCT.

Odporność na awarie połączeń narzędzia CF

Co jest odpornością na awarie połączeń narzędzia CF?

Odporność na awarie połączeń narzędzia CF odnosi się do zdolności menedżerów kolejek w grupie współużytkowania kolejek do tolerowania utraty połączenia ze strukturą narzędzia CF bez zakończenia działania. Ta funkcja próbuje również odbudować strukturę w innym narzędziu sprzęgającym z lepszą łącznością w celu jak najszybszemu odzyskania dostępu do współużytkowanych kolejek.

Co to jest częściowa utrata łączności?

IBM MQ definiuje częściową utratę połączenia jako sytuację, w której jeden lub więcej systemów w syspleksie utraci połączenie z narzędziem CF, w którym przydzielana jest struktura systemu, ale co najmniej jeden system w syspleksie utrzymuje łączność z tym samym narzędziem sprzęgającym.

Co to jest całkowita utrata łączności?

IBM MQ definiuje łączną utratę połączenia jako sytuację, w której żadne systemy w syspleksie nie mają połączenia ze sprzęgiem sprzęgającym i przydzieloną w nim strukturą.

Dlaczego można włączyć tę funkcję?

Odporność na awarie połączenia narzędzia CF zwiększa dostępność serwera IBM MQ, co umożliwia pozostawienie niewspółużytkowanych kolejek po utracie połączenia przez menedżer kolejek z jedną lub większą liczbą struktur narzędzia CF. Dodatkowo menedżery kolejek, które tracą łączność ze strukturą narzędzia CF, automatycznie próbują odbudować strukturę w innym dostępnym narzędziu CF, poprawiając dostępność współużytkowanych kolejek w grupie współużytkowania kolejek.

Uwagi dotyczące włączania tej funkcji

Menedżer kolejek, który toleruje utratę połączenia ze strukturami narzędzia CF bez zakończenia działania, może nie być w stanie ponownie połączyć się ze strukturą narzędzia CF (Coupling Facility) przez pewien czas, jeśli nie ma innego dostępnego alternatywnego narzędzia CF. Kolejki współużytkowane zdefiniowane w strukturze, która ucierpiała z powodu utraty połączenia, pozostają niedostępne do czasu odtworzenia połączenia ze strukturą. W takiej sytuacji aplikacje, które łączą się

z elementami grupy współużytkowania kolejek w celu wykonania pracy w kolejce współużytkowanej, mogą się dowiedzieć, że dostęp do współużytkowanych kolejek, do których muszą oni uzyskać dostęp, nie jest dostępny. Aby uniknąć takiej sytuacji, zaleca się, aby menedżery kolejek były skonfigurowane w taki sposób, aby kończyli się, gdy utracono połączenie ze strukturą narzędzia CF. To rozwiązanie wymusza połączenie aplikacji z innym elementem grupy współużytkowania kolejek, który ma połączenia ze strukturami narzędzia CF, w których zdefiniowane są kolejki współużytkowane, których aplikacja wymaga.

Zarządzanie odpornością na awarie połączeń narzędzia CF

Jak włączyć tę funkcjonalność?

Aby umożliwić odporność na połączenia narzędzia CF, należy wykonać następujące kroki:

1. Upewnij się, że zestaw danych para CFRM został sformatowany w celu obsługi odbudowywania zarządzanego przez system. Pozwala to menedżerom kolejek na zainicjowanie odbudowanego systemu w celu ponownego utworzenia struktury w dostępnym narzędziu CF. Aby określić format zestawu danych CFRM, należy użyć komendy **DISPLAY XCF, COUPLE, TYPE=CFRM**. Aby można było obsługiwać odbudowywanie zarządzane przez system, zestaw danych CFRM powinien zostać sformatowany przez określenie:

```
"ITEM NAME(SMREBLD) NUMBER(1) "
```

Więcej informacji na temat formatowania zestawu danych CFRM zawiera dokumentacja [z/OS MVS Setting Up a Sysplex](#).

2. Upewnij się, że alternatywne narzędzie CF jest dostępne i znajduje się na liście preferencji CFRM dla wszystkich struktur narzędzia CF (Coupling Facility) IBM MQ. Dzięki temu menedżery kolejek mogą próbować odbudować struktury w alternatywnym dostępnym narzędziu CF w celu jak najszybszego przywrócenia dostępu do struktur.

Struktury IBM MQ muszą być zdefiniowane z ENFORCEORDER (NO) w strategii CFRM, dzięki czemu XCF jest w stanie wybrać optymalny system CF w konfiguracji, jeśli produkt IBM MQ musi ponownie zlokalizować strukturę.

Więcej informacji na temat list preferencji struktury można znaleźć w dokumentacji [z/OS MVS Setting Up a Sysplex](#).

3. Zmień wszystkie struktury narzędzia CF, które muszą tolerować utratę połączenia z CFLEVEL (5). Jest to minimalny poziom, który może tolerować utratę połączenia.
4. Należy określić wartości wymagane dla atrybutów **QMGR CFCONLOS** i **CFSTRUCT CFCONLOS** i odpowiednio je zmienić. Atrybut **QMGR CFCONLOS** określa, czy utrata połączenia ze strukturą administracyjną jest tolerowana, a atrybut **CFSTRUCT CFCONLOS** określa, czy utrata połączenia jest tolerowana przez każdą strukturę narzędzia CF. Jeśli wartości domyślne dla tych atrybutów zostaną zachowane, menedżer kolejek kończy działanie po utracie połączenia z dowolną strukturą narzędzia CF.
5. Określ wartości wymagane dla atrybutu **CFSTRUCT RECAUTO** dla każdej struktury CF (Coupling Facility) i zmień je odpowiednio. Za pomocą tego atrybutu można określić, czy struktury narzędzia CF powinny być automatycznie odtwarzane przy użyciu zarejestrowanych danych po całkowitej utracie połączenia. Jeśli wartość domyślna tego atrybutu zostanie zachowana, dla struktur aplikacji nie zostanie wykonana automatyczna operacja odtwarzania po całkowitej utracie połączenia.

Scenariusz 1-Utrata połączenia ze strukturą administracyjną

Menedżery kolejek mogą tolerować utratę połączenia ze strukturą administracyjną bez zakończenia pracy, jeśli wszystkie menedżery kolejek w grupie współużytkowania kolejek są w wersji IBM WebSphere MQ 7.1 lub nowszej. Jeśli istnieją menedżery kolejek na poziomie niższym niż IBM WebSphere MQ 7.1 w grupie współużytkowania kolejek, wszystkie menedżery kolejek w grupie współużytkującej kolejkę zakończą się kodem przyczyny [00C510AB](#), gdy utracono połączenie z strukturą administracyjną.

W przypadku utraty połączenia ze strukturą administracyjną przez dowolny menedżer kolejek, który został skonfigurowany do tolerowania utraty połączenia ze strukturą administracyjną, wszystkie elementy grupy współużytkowania kolejki rozłączają się ze strukturą administracyjną. Następnie wszystkie aktywne menedżery kolejek w grupie współużytkowania kolejek podejmą próbę ponownego nawiązania połączenia ze strukturą administracyjną, co spowoduje, że zostanie ona ponownie przydzielona do narzędzia CF z najlepszą łącznością ze wszystkimi systemami w syspleksie, a następnie odbuduj dane struktury administracyjnej.

Uwaga: Niekoniecznie musi to być narzędzie CF, które ma najlepszą łączność ze wszystkimi systemami, które mają aktywne menedżery kolejek.

Jeśli menedżer kolejek nie może ponownie połączyć się ze strukturą administracyjną, na przykład ponieważ żaden z urządzeń sprzęgających na liście preferencji CFRM dla struktury administracyjnej nie jest dostępny, niektóre operacje kolejki współużytkowanej pozostają niedostępne, dopóki menedżer kolejek nie będzie mógł pomyślnie połączyć się ze strukturą administracyjną i odbudować jej dane struktury administracyjnej. Ponowne połączenie jest wykonywane automatycznie, gdy odpowiedni mechanizm sprzęgający stanie się dostępny w systemie.

Brak połączenia ze strukturą administracyjną podczas uruchamiania menedżera kolejek w wyniku braku połączenia z narzędziem sprzęgającym lub brak dostępnego odpowiedniego narzędzia CF w celu przydzielenia struktury nie jest tolerowany. Następnie wszystkie aktywne menedżery kolejek w grupie współużytkowania kolejek podejmą próbę ponownego nawiązania połączenia ze strukturą administracyjną, co spowoduje ponowne przydzielenie jej w innym narzędziu CF, jeśli jest on dostępny, a następnie odbudowanie danych struktury administracyjnej.

Scenariusz 2-Utrata połączenia ze strukturą aplikacji

Utrata połączenia ze strukturami aplikacji w systemie **CFLEVEL (5)** lub wyższym może być tolerowana bez zakończenia działania menedżera kolejek. Menedżery kolejek połączone ze strukturami aplikacji w systemie **CFLEVEL (4)** lub niższym lub strukturami w wersji **CFLEVEL (5)**, które nie zostały skonfigurowane do tolerowania utraty połączenia, są w stanieabend z kodem przyczyny 00C510AB, gdy utracono połączenie z strukturą.

W przypadku utraty połączenia ze strukturą aplikacji, która została skonfigurowana do tolerowania utraty połączenia, wszystkie menedżery kolejek, które utraciły łączność ze strukturą, rozłączyli się. Późniejsze zachowanie menedżera kolejek zależy od tego, czy utrata połączenia jest częściowa, czy też całkowita.

Częściowa utrata połączenia ze strukturą aplikacji

Jeśli utrata połączenia jest określona jako częściowa, menedżery kolejek, które utraciły połączenie ze strukturą, próbują zainicjować przebudowę zarządzaną przez system w celu przeniesienia struktury do innego narzędzia CF z poprawą połączenia. Jeśli odbudowywanie zakończy się pomyślnie, zarówno komunikaty trwałe, jak i nietrwałe w strukturze są kopiowane do innego narzędzia CF, a dostęp do kolejek w strukturze jest odtwarzany. Menedżery kolejek, które nie utraciły połączenia, pozostają połączone ze strukturą, jednak operacje, które uzyskują dostęp do struktury, są opóźnione w trakcie procesu odbudowywania zarządzanego przez system.

Jeśli struktura aplikacji nie może zostać przebudowana do innego narzędzia CF z poprawą połączenia, albo niektóre menedżery kolejek nadal nie mają połączenia ze strukturą po jej odbudowaniu w innym narzędziu CF, kolejki zdefiniowane w strukturze pozostają niedostępne w menedżerach kolejek, które nie mają połączenia ze strukturą, dopóki połączenie nie zostanie przywrócone do narzędzia CF. Menedżery kolejek automatycznie ponownie łączą się ze strukturą, gdy staje się ona dostępna i odtwarzany jest dostęp do współużytkowanych kolejek zdefiniowanych w strukturze.

Łączna utrata połączenia ze strukturą aplikacji

Jeśli wszystkie systemy MVS w syspleksie utracą połączenie z narzędziem sprzęgającym, w którym została przydzielona struktura aplikacji, program z/OS zwalnia strukturę z narzędzia CF przy każdej próbie ponownego nawiązania połączenia z strukturą. Menedżer kolejek może podjąć próbę ponownego połączenia ze strukturą z kilku powodów, na przykład z powodu próby otwarcia kolejki współużytkowanej przez aplikację lub powiadomienia systemu, że nowe zasoby narzędzia CF

mogły zostać udostępnione. Dlatego jest prawdopodobne, że wszystkie nietrwałe komunikaty w tej strukturze zostaną utracone w wyniku całkowitej utraty połączenia ze strukturą aplikacji.

Odtwarzalne struktury aplikacji są automatycznie odzyskiwalne po całkowitej utracie połączenia, jeśli zostały zdefiniowane w produkcie **RECAUTO (YES)**. Odtwarzanie rozpoczyna się niemal natychmiast, jeśli dostępny jest alternatywny mechanizm sprzęgający, który umożliwia przydzielenie struktury w lub za każdym razem, gdy taki mechanizm sprzęgający stanie się dostępny. Jeśli struktura nie została zdefiniowana przy użyciu programu **RECAUTO (YES)**, odtwarzanie może zostać uruchomione za pomocą komendy **RECOVER CFSTRUCT**. Powoduje to odtwarzanie wszystkich trwałych komunikatów w strukturze, ale wszystkie nietrwałe komunikaty są tracone. Ponieważ proces ten obejmuje odczytywanie dziennika menedżera kolejek, może zająć trochę czasu, dlatego zaleca się regularne tworzenie kopii zapasowych w celu skrócenie czasu, aż do czasu, gdy zostanie przywrócone dostęp do współużytkowanych kolejek w strukturze.

Menedżery kolejek próbują ponownie połączyć się z nieodtworzonymi strukturami aplikacji, gdy tylko aplikacja próbuje otworzyć współużytkowaną kolejkę zdefiniowaną w strukturze, albo powiadomienie zostanie odebrane z systemu, w którym nowe zasoby narzędzia CF stały się dostępne. Jeśli dostępna jest odpowiednia funkcja sprzęgania w celu przydzielenia danej struktury, zostanie przydzielona nowa struktura, a dostęp do współużytkowanych kolejek zdefiniowanych w strukturze zostanie odtworzony. Ponieważ komunikaty trwałe nie mogą być umieszczane w kolejkach zdefiniowanych w strukturach nienaprawialnych, wszystkie komunikaty znajdujące się w kolejkach współużytkowanych są tracone.

Zachowanie operacyjne

Jeśli menedżer kolejek produktu IBM WebSphere MQ 7.1 lub nowszy, skonfigurowany do tolerowania utraty połączenia z określoną strukturą narzędzia CF, utraci połączenie, członkowie grupy współużytkującej kolejkę próbują automatycznie odzyskać dane po awarii i ponownie połączyć się ze strukturą. To działanie może obejmować ponowne przydzielanie struktury w innym narzędziu CF o lepszej łączności, jeśli jest ona dostępna. Jednakże interwencja operatora może być w dalszym ciągu wymagana do odzyskania od utraty łączności.

Zwykle wymagane działanie operatora jest następujące:

1. Rozwiąż przyczynę niepowodzenia, która skutkuje utratą połączenia.
2. Upewnij się, że narzędzie CF, w którym mogą być przydzielone struktury IBM MQ, jest dostępne we wszystkich systemach w sysplexie.

Wszystkie struktury, które zostały automatycznie ponownie przydzielone w innym narzędziu CF po utracie zdarzenia połączenia, mogą zostać przeniesione do narzędzia CF z optymalną łącznością ze wszystkimi menedżerami kolejek w grupie współużytkowania kolejek. W razie potrzeby można to zrobić, inicjując komendę odbudowywania systemu zarządzanego przez system **SETXCF START, REBUILD** zgodnie z opisem w sekcji [z/OS MVS System Commands Reference](#).

W przypadku częściowej utraty połączenia ze strukturą aplikacji menedżery kolejek, które utraciły połączenie ze strukturą, podejmą próbę zainicjowania odbudowywania zarządzanego przez system. Ten proces przydziela tylko strukturę w innym narzędziu CF, jeśli narzędzie CF ma łączność ze wszystkimi aktywnymi menedżerami kolejek, które są aktualnie połączone ze strukturą. Z tego powodu możliwe jest, że w sytuacji, gdy większość menedżerów kolejek w grupie współużytkowania kolejek utraciła połączenie ze strukturą aplikacji, nie są one w stanie odbudować struktury na inny mechanizm sprzęgający z powodu menedżerów kolejek, które nadal są połączone z pierwotną strukturą. W takiej sytuacji menedżery kolejek, które są nadal połączone z pierwotną strukturą, można zamknąć, aby umożliwić odbudowanie struktury, lub wydanie komendy **RESET CFSTRUCT ACTION(FAIL)** w celu niepowodzenia struktury. Odtwarzanie może zostać zainicjowane w odpowiednich strukturach, wydając komendę **RECOVER CFSTRUCT**.

Uwaga: W przypadku niepowodzenia i odtwarzania struktury wszystkie nietrwałe komunikaty w strukturze są tracone.

W tym temacie opisano znaczenie zabezpieczeń produktu IBM MQ oraz implikacje braku odpowiednich ustawień zabezpieczeń w systemie.

Dlaczego należy chronić zasoby produktu IBM MQ

Produkt IBM MQ obsługuje przesyłanie informacji, które są potencjalnie cenne. Zastosowanie zabezpieczeń zapewnia, że zasoby IBM MQ są własnością i są chronione przed nieautoryzowanym dostępem. Taki dostęp może prowadzić do utraty lub ujawnienia informacji.

Użytkownik powinien upewnić się, że żaden z następujących zasobów nie zostanie uzyskany lub zmieniony przez nieautoryzowanego użytkownika lub proces:

- Połączenia z IBM MQ
- Obiekty IBM MQ, takie jak kolejki, procesy i listy nazw
- Łącza transmisji produktu IBM MQ, czyli kanały produktu IBM MQ
- IBM MQ komendy sterujące systemem
- Komunikaty produktu IBM MQ
- Informacje kontekstowe powiązane z komunikatami

Aby zapewnić niezbędne bezpieczeństwo, produkt IBM MQ używa narzędzia autoryzacji systemu z/OS (SAF) do kierowania żądań autoryzacji do zewnętrznego menedżera bezpieczeństwa (External Security Manager-ESM), na przykład Security Server (wcześniej znanego jako RACF). Produkt IBM MQ nie weryfikuje zabezpieczeń własnych. W przypadku używania rozproszonej kolejki lub używanych klientów może być wymagane dodatkowe środki bezpieczeństwa, dla których produkt IBM MQ udostępnia rekordy uwierzytelniania kanału, wyjścia kanału, atrybut kanału MCAUSER oraz protokół TLS.

Decyzja o zezwoleniu na dostęp do obiektu jest podejmowana przez ESM i IBM MQ po tej decyzji. Jeśli ESM nie może podjąć decyzji, program IBM MQ uniemożliwi dostęp do obiektu.

Co się stanie, jeśli zasoby produktu IBM MQ nie są chronione

Jeśli zabezpieczenia nie są dostępne, najbardziej prawdopodobnym działaniem jest to, że użytkownicy produktu *wszystkie* mogą uzyskiwać dostęp do zasobu *co* i zmieniać jego zasoby. Dotyczy to nie tylko użytkowników lokalnych, ale także tych w systemach zdalnych korzystających z rozproszonej kolejki lub klientów, w przypadku których kontrola zabezpieczeń logowania może być mniej rygorystyczne niż zwykle w przypadku produktu z/OS.

Aby włączyć sprawdzanie zabezpieczeń, wykonaj następujące czynności:

- Zainstaluj i aktywuj ESM (na przykład Security Server).
- Zdefiniuj klasę MQADMIN, jeśli używasz ESM innego niż Security Server.
- Aktywuj klasę MQADMIN.

Należy rozważyć, czy użycie nazw zasobów o różnej wielkości liter będzie korzystne dla przedsiębiorstwa. Jeśli w profilach ESM używane są nazwy zasobów o różnej wielkości liter, należy zdefiniować i aktywować klasę MXADMIN.

z/OS szyfrowanie zestawu danych

Funkcja szyfrowania zestawu danych (Data Set Encryption-DSE) umożliwia szyfrowanie zestawów danych produktu z/OS, dzięki czemu dane, które zawierają, mogą być wyświetlane lub modyfikowane tylko przez identyfikatory użytkowników, którym nadano konkretne uprawnienie. Pozwala to na szyfrowanie danych w systemie plików i zapobiega przypadkowemu ujawnieniu poufnych informacji użytkownikom, którzy mają uzasadnione potrzeby biznesowe i uprawnienia do samodzielnego zarządzania zestawami danych.

V 9.2.0 **CD** W wersjach wcześniejszych niż IBM MQ for z/OS 9.1.4 produkt IBM MQ for z/OS nie obsługuje użycia funkcji DSE z aktywnymi dziennikami, zestawami stron i zestawami danych

współużytkowanego komunikatu (SMDS), które udostępniają podstawowe mechanizmy trwałości dla komunikatów produktu IBM MQ.

Zamiast tego produkt *Advanced Message Security* udostępnia kompleksowe rozwiązanie szyfrujące dla przesyłania komunikatów w produkcie IBM MQ, które obejmuje całą sieć produktu IBM MQ, szyfrowanie danych w locie, w stanie spoczynku, a nawet wewnątrz procesów środowiska wykonawczego IBM MQ.

Inne VSAM i sekwencyjne zestawy danych używane w podsystemie IBM MQ mogą być szyfrowane za pomocą DSE. Na przykład:

- zestaw danych programu startowego
- Pliki sekwencyjne odczytane z komend konfiguracji systemu (MQSC) podczas uruchamiania przy użyciu DDNAME CSQINPX
- Dzienniki archiwalne produktu IBM MQ, często używane do długoterminowego archiwizowania danych dziennika produktu IBM MQ do celów kontroli.

Można szyfrować za pomocą DSE przydzielając klasę danych, która jest zdefiniowana z etykietą klucza zestawu danych. Więcej informacji na ten temat zawiera sekcja [Planowanie pamięci archiwalnej dziennika](#).

V 9.2.0 W produkcie IBM MQ for z/OS 9.1.4 produkt IBM MQ for z/OS obsługuje użycie interfejsu DSE wraz z aktywnymi dziennikami i zestawami stron oprócz obsługi udostępnianej we wcześniejszych wersjach.

V 9.2.0 Produkt IBM MQ for z/OS nie obsługuje funkcji DSE dla współużytkowanych zestawów danych komunikatów (SMDS).

V 9.2.0 Patrz sekcja [poufność danych w produkcie IBM MQ for z/OS przy użyciu szyfrowania zestawu danych](#). :NONE.

Pojęcia pokrewne

[Pojęcia dotyczące bezpieczeństwa](#)

[Rekordy uwierzytelniania kanału](#)

[Uprawnienia do pracy z obiektami IBM MQ w systemie z/OS](#)

[Protokoły zabezpieczeń szyfrujących: TLS](#)

[Źródła, z których można wydawać komendy MQSC w systemie z/OS](#)

Zadania pokrewne

[Konfigurowanie zabezpieczeń w systemie z/OS](#)

[Porównywanie zabezpieczeń na poziomie łącza i zabezpieczeń na poziomie aplikacji](#)

Odsyłacze pokrewne

[Komunikaty dla IBM MQ for z/OS](#)

z/OS Elementy sterujące i opcje zabezpieczeń w produkcie IBM MQ for z/OS

Można określić, czy zabezpieczenia mają być włączone dla całego podsystemu IBM MQ, oraz czy mają być wykonywane sprawdzenia zabezpieczeń na poziomie menedżera kolejek lub grupy współużytkowania kolejek. Można również sterować liczbą sprawdzanych identyfikatorów użytkowników pod kątem zabezpieczeń zasobów API.

Bezpieczeństwo podsystemu

Bezpieczeństwo podsystemu jest elementem sterującym, który określa, czy dla całego menedżera kolejek wykonywane jest sprawdzanie zabezpieczeń. Jeśli sprawdzanie zabezpieczeń nie jest wymagane (na przykład w systemie testowym) lub jeśli poziom zabezpieczeń jest zadowalający dla wszystkich zasobów, które mogą łączyć się z produktem IBM MQ (w tym klientów i kanałów), można wyłączyć sprawdzanie

zabezpieczeń dla menedżera kolejek lub grupy współużytkowania kolejek, tak aby nie było możliwe dalsze sprawdzanie zabezpieczeń.

Jest to jedyne sprawdzenie, które może całkowicie wyłączyć zabezpieczenia i określić, czy wykonywane są jakiegokolwiek inne sprawdzenia zabezpieczeń. Oznacza to, że jeśli sprawdzanie menedżera kolejek lub grupy współużytkowania kolejek zostanie wyłączone, żadne inne sprawdzanie IBM MQ nie zostanie wykonane. Jeśli zostanie pozostawiony włączony, program IBM MQ sprawdzi wymagania dotyczące zabezpieczeń dla innych zasobów produktu IBM MQ.

Można również włączać lub wyłączać zabezpieczenia dla określonych zestawów zasobów, takich jak komendy.

Sprawdzanie poziomu menedżera kolejek lub grupy współużytkowania kolejki

Zabezpieczenia można implementować na poziomie menedżera kolejek lub na poziomie grupy współużytkowania kolejek. Jeśli zabezpieczenia są implementowane na poziomie grupy współużytkowania kolejek, wszystkie menedżery kolejek w grupie współużytkuje te same profile. Oznacza to, że istnieje mniej profili do definiowania i konserwacji, co ułatwia zarządzanie bezpieczeństwem. Dzięki temu można łatwo dodać nowego menedżera kolejek do grupy współużytkowania kolejek, ponieważ dziedziczy on istniejące profile zabezpieczeń.

Można również zaimplementować kombinację obu tych elementów, jeśli wymaga tego instalacja, na przykład podczas migracji, lub jeśli w grupie współużytkowania kolejek jest jeden menedżer kolejek, który wymaga różnych poziomów zabezpieczeń dla innych menedżerów kolejek w grupie.

Zabezpieczenia na poziomie grupy współużytkowania kolejki

Sprawdzanie zabezpieczeń na poziomie grupy współużytkowania kolejki jest wykonywane dla całej grupy współużytkowania kolejek. Pozwala to uprościć administrowanie zabezpieczeniami, ponieważ wymaga zdefiniowania mniejszej liczby profili zabezpieczeń. Autoryzacja identyfikatora użytkownika w celu użycia określonego zasobu jest obsługiwana na poziomie grupy współużytkowania kolejki i jest niezależna od tego, który menedżer kolejek używany przez użytkownika w celu uzyskania dostępu do zasobu.

Na przykład aplikacja serwera działa pod kontrolą serwera o identyfikatorze SERVER i chce mieć dostęp do kolejki o nazwie SERVER.REQUEST, a użytkownik chce uruchomić instancję serwera na każdym obrazie produktu z/OS w syspleksie. Zamiast zezwalać SERVER na otwarcie serwera SERVER.REQUEST indywidualnie dla każdego menedżera kolejek (zabezpieczenia na poziomie menedżera kolejek), można zezwolić na dostęp tylko na poziomie grupy współużytkowania kolejki.

Profile zabezpieczeń na poziomie grupy współużytkowania kolejek można używać do ochrony wszystkich typów zasobów, niezależnie od tego, czy są one lokalne, czy współużytkowane.

zabezpieczenia na poziomie menedżera kolejek

Profile zabezpieczeń na poziomie menedżera kolejek można używać do ochrony wszystkich typów zasobów, niezależnie od tego, czy są one lokalne, czy współużytkowane.

Połączenie obu poziomów

Istnieje możliwość użycia kombinacji zarówno menedżera kolejek, jak i zabezpieczeń na poziomie grupy współużytkowania kolejek.

Istnieje możliwość nadpisania ustawień zabezpieczeń na poziomie grupy współużytkowania kolejek dla konkretnego menedżera kolejek, który jest członkiem tej grupy. Oznacza to, że można wykonać inny poziom sprawdzania zabezpieczeń dla pojedynczego menedżera kolejek z tymi, które są wykonywane na innych menedżerach kolejek w grupie.

Więcej informacji na ten temat zawiera sekcja [Profile do sterowania grupą współużytkowania kolejek lub zabezpieczenia na poziomie menedżera kolejek](#).

Kontrolowanie liczby sprawdzanych identyfikatorów użytkowników

RESLEVEL jest profilem serwera Security Server, który steruje liczbą identyfikatorów użytkowników sprawdzanych pod kątem zabezpieczeń zasobów IBM MQ. Zwykle, gdy użytkownik próbuje uzyskać dostęp do zasobu IBM MQ, serwer zabezpieczeń sprawdza odpowiednie ID użytkownika lub identyfikatory w celu sprawdzenia, czy dostęp do tego zasobu jest dozwolony. Definiując profil RESLEVEL, można określić, czy sprawdzane są dwa identyfikatory użytkowników: zero, jeden lub, tam gdzie ma to zastosowanie, dwa identyfikatory użytkowników.

Te elementy sterujące są wykonywane w oparciu o połączenie przez połączenie, a ostatnie w przypadku życia połączenia.

Dla każdego menedżera kolejek istnieje tylko jeden profil RESLEVEL. Sterowanie jest implementowane przez dostęp, który użytkownik ma do tego profilu.

Mieszane litery lub wielkie litery IBM MQ RACF

Teraz można użyć obsługi profilu RACF z mieszanym przypadkiem, co pozwala na użycie nazw mieszanych zasobów elementów pracy i definiowanie profili IBM MQ RACF w celu ich ochrony.

Można wybrać jedną z następujących opcji:

- Kontynuuj używanie tylko wielkich liter IBM MQ RACF klas, jak w poprzednich wersjach, lub
- Użyj nowych klas IBM MQ RACF z mieszaną wielkością liter.

Bez użycia profilu przypadku mieszanego RACF nadal można używać nazw mieszanych zasobów elementów pracy w produkcie IBM MQ for z/OS. Jednak te nazwy zasobów mogą być chronione wyłącznie za pomocą ogólnych profili produktu RACF w klasach IBM MQ wielkich liter. W przypadku korzystania z obsługi profilu IBM MQ RACF z mieszanym przypadkiem można zapewnić bardziej szczegółowy poziom ochrony, definiując profile IBM MQ RACF w klasach IBM MQ mieszanych elementów pracy.

Zasoby, które można chronić w programie IBM MQ for z/OS

Po uruchomieniu menedżera kolejek lub po poleceniu przez komendę operatora program IBM MQ for z/OS określa zasoby, które mają być chronione.

Istnieje możliwość kontrolowania, które sprawdzenia zabezpieczeń są wykonywane dla każdego pojedynczego menedżera kolejek. Na przykład można zaimplementować pewną liczbę sprawdzeń zabezpieczeń w menedżerze kolejek produkcyjnych, ale żaden z nich nie znajduje się w menedżerze kolejek testowych.

Bezpieczeństwo połączenia

Sprawdzanie zabezpieczeń połączenia jest przeprowadzane, gdy program użytkowy próbuje połączyć się z menedżerem kolejek. Jest to realizowane przez wydanie żądania MQCONN lub MQCONNX albo gdy inicjator kanału lub adapter CICS lub IMS wysyła żądanie połączenia.

Jeśli używane są zabezpieczenia na poziomie menedżera kolejek, można wyłączyć sprawdzanie zabezpieczeń połączeń dla konkretnego menedżera kolejek. Jeśli jednak to zrobisz, każdy użytkownik może połączyć się z tym menedżerem kolejek.

W przypadku adaptera CICS do sprawdzania zabezpieczeń połączenia używany jest tylko identyfikator użytkownika przestrzeni adresowej CICS, a nie indywidualny identyfikator użytkownika terminalu CICS. W przypadku adaptera IMS, gdy element sterujący IMS lub regiony zależne łączą się z IBM MQ, sprawdzane jest ID użytkownika przestrzeni adresowej IMS. Dla inicjatora kanału sprawdzany jest identyfikator użytkownika używany przez przestrzeń adresową inicjatora kanału.

Sprawdzanie zabezpieczeń połączenia można włączać lub wyłączać na poziomie menedżera kolejek lub grupy współużytkownika kolejek.

Bezpieczeństwo komend

Sprawdzanie zabezpieczeń komend jest wykonywane, gdy użytkownik wysyła komendę MQSC z dowolnego źródła opisanego w sekcji [Komendy wydawane](#). Można wykonać osobne sprawdzenie zasobu określonego przez komendę zgodnie z opisem w sekcji ["Zabezpieczenia zasobów komend"](#) na stronie 284.

Jeśli sprawdzanie komend zostanie wyłączone, emitenci komend nie są sprawdzani, aby sprawdzić, czy mają one uprawnienia do wydawania komendy.

Jeśli komendy MQSC są wprowadzane z konsoli, konsola musi mieć atrybut uprawnienia do konsoli systemowej z/OS SYS. Komendy, które są wydawane z zestawów danych CSQINP1 lub CSQINP2 lub wewnątrz przez menedżera kolejek, są zwolnione ze wszystkich kontroli zabezpieczeń, podczas gdy dla CSQINPX używany jest ID użytkownika przestrzeni adresowej inicjatora kanału. Użytkownik musi kontrolować, kto może aktualizować te zestawy danych przy użyciu normalnego zabezpieczenia zestawu danych.

Sprawdzanie zabezpieczeń komend można włączać lub wyłączać na poziomie menedżera kolejek lub grupy współużytkowania kolejek.

Zabezpieczenia zasobów komend

Niektóre komendy MQSC, na przykład definiujące kolejkę lokalną, obejmują manipulowanie zasobami produktu IBM MQ. Gdy zabezpieczenia zasobów komend są aktywne, za każdym razem, gdy komenda z użyciem zasobu jest wydawana, program IBM MQ sprawdza, czy użytkownik może zmienić definicję tego zasobu.

Zabezpieczenia zasobów komend mogą być pomocne w egzekwowaniu standardów nazewnictwa. Na przykład administrator płac może mieć możliwość usunięcia i zdefiniowania tylko kolejek o nazwach rozpoczynających się od "PAYROLL". Jeśli zabezpieczenia zasobów komend są nieaktywne, w zasobie, który jest manipulowany przez komendę, nie są wykonywane żadne sprawdzenia zabezpieczeń. Zabezpieczenia zasobów komendy nie należy mylić z zabezpieczeniami komend; te dwa są niezależne.

Wyłączenie sprawdzania zabezpieczeń zasobów komendy nie ma wpływu na sprawdzanie zasobów, które jest wykonywane specjalnie dla innych typów przetwarzania, które nie dotyczą komend.

Sprawdzanie zabezpieczeń zasobów komend można włączać lub wyłączać na poziomie menedżera kolejek lub grupy współużytkowania kolejek.

Uwagi dotyczące zabezpieczeń kanału

Bezpieczeństwo kanału

Jeśli używane są kanały, dostępne opcje zabezpieczeń zależą od używanego przez użytkownika protokołu komunikacyjnego. Jeśli używany jest protokół TCP, z protokołem komunikacyjnym nie są udostępniane żadne funkcje zabezpieczeń, chociaż można używać protokołu TLS. Jeśli jest używany protokół APPC, informacje o ID użytkownika można przepływać z wysyłającego agenta MCA przez sieć do docelowego agenta MCA w celu weryfikacji.

W przypadku obu protokołów można określić identyfikatory użytkowników, które mają być sprawdzane w celu zapewnienia bezpieczeństwa, oraz liczbę tych identyfikatorów. Dostępne opcje zależą od używanego przez użytkownika protokołu, określonego podczas definiowania kanału oraz ustawień RESLEVEL dla inicjatora kanału.

Więcej informacji na temat typów zabezpieczeń kanałów zawiera sekcja [Rekordy uwierzytelniania kanału i Przegląd wyjścia zabezpieczeń](#).

Odsyłacze pokrewne

["Interfejs API-zabezpieczenia zasobów w produkcie IBM MQ for z/OS" na stronie 285](#)

Zasoby są sprawdzane, gdy aplikacja otwiera obiekt przy użyciu wywołania MQOPEN lub MQPUT1 . Dostęp wymagany do otwarcia obiektu zależy od tego, jakie opcje otwarcia są określone podczas otwierania kolejki.

z/OS Interfejs API-zabezpieczenia zasobów w produkcji IBM MQ for z/OS

Zasoby są sprawdzane, gdy aplikacja otwiera obiekt przy użyciu wywołania MQOPEN lub MQPUT1 . Dostęp wymagany do otwarcia obiektu zależy od tego, jakie opcje otwarcia są określone podczas otwierania kolejki.

Zabezpieczenia zasobów interfejsu API są podzielone na następujące operacje sprawdzania:

- [kolejka](#)
- [Proces](#)
- [Lista nazw](#)
- [Alternatywny użytkownik](#)
- [Kontekst](#)

Podczas otwierania obiektu menedżera kolejek lub podczas uzyskiwania dostępu do obiektów klasy pamięci nie są wykonywane żadne sprawdzenia zabezpieczeń.

Kolejka

Kontrola zabezpieczeń kolejki, która umożliwia otwarcie kolejki oraz opcje, z którymi są one dozwolone, aby można było je otworzyć. Na przykład użytkownik może być uprawniony do otwarcia kolejki o nazwie PAYROLL.INCREASE.SALARY , aby przeglądać komunikaty w kolejce (za pomocą opcji MQOO_BROWSE), ale nie usuwać komunikatów z kolejki (przy użyciu jednej z opcji MQOO_INPUT_*). Jeśli sprawdzanie kolejek jest wyłączone, każdy użytkownik może otworzyć dowolną kolejkę z dowolną poprawną opcją otwarcia (czyli dowolną poprawną opcją MQOO_* w wywołaniu komendy MQOPEN lub MQPUT1).

Sprawdzanie zabezpieczeń kolejki można włączać lub wyłączać na poziomie menedżera kolejek lub grupy współużytkowania kolejek.

Proces

Sprawdzanie zabezpieczeń procesu jest wykonywane po otwarciu obiektu definicji procesu przez użytkownika. Jeśli sprawdzanie jest wyłączone, każdy użytkownik może otworzyć dowolny proces.

Sprawdzanie zabezpieczeń procesu można włączać lub wyłączać na poziomie menedżera kolejek lub grupy współużytkowania kolejek.

Lista nazw

Sprawdzanie zabezpieczeń listy nazw jest przeprowadzane, gdy użytkownik otwiera listę nazw. W przypadku wyłączenia sprawdzania listy nazw każdy użytkownik może otworzyć dowolną listę nazw.

Sprawdzanie zabezpieczeń listy nazw można włączać lub wyłączać na poziomie menedżera kolejek lub grupy współużytkowania kolejek.

Alternatywny użytkownik

Alternatywne zabezpieczenia użytkownika kontrolują, czy jeden z identyfikatorów użytkowników może używać uprawnień innego ID użytkownika do otwierania obiektu IBM MQ .

Na przykład:

- Program serwera działający pod ID użytkownika PAYSERV pobiera komunikat żądania z kolejki umieszczonej w kolejce przez użytkownika o identyfikatorze USER1.

- Gdy program serwera pobiera komunikat z żądaniem, przetwarza żądanie i umieszcza odpowiedź z powrotem w kolejce odpowiedzi określonej za pomocą komunikatu żądania.
- Zamiast używać własnego ID użytkownika (PAYSERV) do autoryzowania otwarcia kolejki odpowiedzi, serwer może określić inny ID użytkownika, w tym przypadku USER1. W tym przykładzie alternatywne zabezpieczenia użytkownika będą kontrolować, czy użytkownik o identyfikatorze PAYSERV może określić ID użytkownika USER1 jako alternatywny identyfikator użytkownika podczas otwierania kolejki odpowiedzi.

Alternatywny identyfikator użytkownika jest określony w polu *AlternateUserId* deskryptora obiektu (MQOD).

Alternatywnych identyfikatorów użytkowników można używać na dowolnym obiekcie IBM MQ, na przykład w procesach lub na listach nazw. Nie ma to wpływu na identyfikator użytkownika używany przez innych menedżerów zasobów, na przykład dla zabezpieczeń CICS lub dla zabezpieczeń zestawu danych z/OS.

Jeśli alternatywne zabezpieczenia użytkownika nie są aktywne, każdy inny użytkownik może użyć innego ID użytkownika jako alternatywnego ID użytkownika.

Na poziomie menedżera kolejek lub grupy współużytkowania kolejek można włączać lub wyłączać funkcję sprawdzania bezpieczeństwa dla użytkowników alternatywnych.

Kontekst

Kontekst to informacja, która ma zastosowanie do konkretnego komunikatu i jest zawarta w deskrytorze komunikatu (MQMD), który jest częścią komunikatu. Informacje o kontekście są wyświetlane w dwóch sekcjach:

Sekcja Tożsamość

Użytkownik aplikacji, która po raz pierwszy umiała komunikat w kolejce. Składa się on z następujących pól:

- *UserIdentifier*
- *AccountingToken*
- *ApplIdentityData*

Sekcja Pochodzenie

Aplikacja, która umieszczała komunikat w kolejce, w której jest obecnie przechowywany. Składa się on z następujących pól:

- *PutApplType*
- *PutApplName*
- *PutDate*
- *PutTime*
- *ApplOriginData*

Aplikacje mogą określać dane kontekstu, gdy zostanie wykonane wywołanie MQPUT lub MQPUT1. Aplikacja może wygenerować dane, dane mogą być przekazywane z innego komunikatu lub menedżer kolejek może generować dane domyślnie. Na przykład, programy serwerowe mogą używać danych kontekstowych do sprawdzania tożsamości requestera, czyli czy ten komunikat pochodzi z poprawnej aplikacji? Zwykle pole *UserIdentifier* jest używane do określenia identyfikatora użytkownika alternatywnego.

Zabezpieczenia kontekstu są używane do określania, czy użytkownik może określić dowolne opcje kontekstu w dowolnym wywołaniu komendy MQOPEN lub MQPUT. Więcej informacji na temat opcji kontekstu zawiera sekcja Opcje MQOPEN związane z kontekstem komunikatów. Opisy pól deskryptora komunikatu odnoszące się do kontekstu znajdują się w sekcji MQMD-Message descriptorMQMD-deskryptor komunikatu.

Jeśli sprawdzanie zabezpieczeń kontekstu zostanie wyłączone, każdy użytkownik będzie mógł użyć dowolnej opcji kontekstu, która pozwala na to bezpieczeństwo kolejki.

Sprawdzanie zabezpieczeń kontekstu można włączać lub wyłączać w każdej kolejce, menedżerze kolejek lub na poziomie grupy współużytkowania kolejek.

z/OS Dostępność w systemie z/OS

Produkt IBM MQ for z/OS ma wiele funkcji w celu zapewnienia wysokiej dostępności. W tej sekcji opisano niektóre zagadnienia związane z dostępnością.

Jeśli menedżer kolejek lub inicjator kanału nie powiedzie się, kilka funkcji produktu IBM MQ może zwiększyć dostępność systemu. Więcej informacji na temat tych funkcji można znaleźć w następujących sekcjach:

- [Uwagi dotyczące systemu Sysplex](#)
- [Kolejki współużytkowane](#)
- [Współużytkowane kanały](#)
- [IBM MQ Dostępność sieci](#)
- [Korzystanie z programu z/OS Automatic Restart Manager \(ARM\)](#)
- [Korzystanie z narzędzia z/OS Extended Recovery Facility \(XRF\)](#)
- [Korzystanie z atrybutu z/OS GROUPUR do odtwarzania w grupie współużytkowania kolejek](#)
- [Gdzie znaleźć więcej informacji o dostępności](#)

Uwagi dotyczące systemu Sysplex

W systemie *sysplex* jest to liczba obrazów systemu operacyjnego z/OS współpracujących w jednym obrazie systemu i komunikacja za pomocą narzędzia CF. Produkt IBM MQ może korzystać z narzędzi środowiska *sysplex* w celu zwiększenia dostępności.

Usunięcie powinowactwa między menedżerem kolejek a konkretnym obrazem produktu z/OS umożliwia zrestartowanie menedżera kolejek na innym obrazie produktu z/OS w przypadku wystąpienia błędu obrazu. Mechanizm restartu może być ręczny, używać ARM lub korzystać z automatyzacji systemu, jeśli użytkownik zapewnia następujące działania:

- Wszystkie zestawy stron, dzienniki, zestawy danych programu startowego, biblioteki kodu i zestawy danych konfiguracji menedżera kolejek są zdefiniowane w woluminach współużytkowanych.
- Definicja podsystemu ma zasięg *sysplex* i unikalna nazwa w obrębie *sysplexu*.
- Poziom *wczesnego kodu* zainstalowanego na każdym obrazie z/OS w czasie IPL jest na tym samym poziomie.
- Wirtualne adresy IP TCP (VIPA) są dostępne na każdym stosie TCP w *syspleksie* oraz skonfigurowane obiekty nasłuchiwanie TCP programu IBM MQ i połączenia przychodzące, aby używać VIPAs zamiast domyślnych nazw hostów.

Więcej informacji na temat używania protokołu TCP w *syspleksie* zawiera sekcja *TCP/IP w systemie sysplex*, SG24-5235, publikacja IBM Redbooks .

Dodatkowo można skonfigurować wiele menedżerów kolejek działających na różnych obrazach systemu operacyjnego w *syspleksie*, aby działać jako grupa współużytkowania kolejek, która może korzystać z kolejek współużytkowanych i współużytkowanych kanałów dla wyższej dostępności i równoważenia obciążenia.

Kolejki współużytkowane

W środowisku grupy współużytkowania kolejek aplikacja może połączyć się z dowolnym z menedżerów kolejek w grupie współużytkowania kolejek. Ponieważ wszystkie menedżery kolejek w grupie współużytkowania kolejek mogą uzyskiwać dostęp do tego samego zestawu współużytkowanych kolejek, aplikacja nie jest uzależniona od dostępności konkretnego menedżera kolejek. Menedżer

kolejek w grupie współużytkowania kolejek może serwisować każdą kolejkę. Zapewnia to większą dostępność, jeśli menedżer kolejek zostanie zatrzymany, ponieważ wszystkie pozostałe menedżery kolejek w grupie współużytkowania kolejek mogą kontynuować przetwarzanie kolejki. Więcej informacji na temat wysokiej dostępności współużytkowanych kolejek zawiera sekcja [“Zalety korzystania z kolejek współużytkowanych”](#) na stronie 193.

Aby jeszcze bardziej zwiększyć dostępność komunikatów w grupie współużytkowania kolejek, program IBM MQ wykrywa, czy inny menedżer kolejek w grupie nie łączy się z obiektem sprzęgającym w sposób nieprawidłowy, a następnie wykonuje jednostki pracy dla tego menedżera kolejek, które nadal oczekują, o ile to możliwe. Jest on znany jako *odtworzenie równorzędne* i jest opisany w sekcji [“Odtwarzanie równorzędne”](#) na stronie 273.

Odtwarzanie równorzędne nie może odzyskać jednostek pracy, które były wątpliwe w momencie wystąpienia awarii. Za pomocą menedżera automatycznego restartu (ARM) można zrestartować wszystkie systemy związane z awarią (na przykład CICS, Db2 i IBM MQ), a także upewnić się, że wszystkie systemy są restartowane na tym samym nowym procesorze. Oznacza to, że mogą one resynchronizować, a także przyspieszać szybkie odzyskiwanie wątpliwych jednostek pracy. Jest to opisane w sekcji [“Korzystanie z programu z/OS Automatic Restart Manager \(ARM\)”](#) na stronie 289.

Kanały współużytkowane

W środowisku grupy współużytkowania kolejek produkt IBM MQ udostępnia funkcje, które zapewniają wysoką dostępność sieci. Inicjator kanału umożliwia korzystanie z produktów sieciowych, które równoważą żądania sieciowe w zestawie uprawnionych serwerów i ukrywać awarie serwera z sieci (na przykład w przypadku zasobów ogólnych VTAM). Produkt IBM MQ używa ogólnego portu dla żądań przychodzących, dzięki czemu żądania przyłączenia mogą być kierowane do dowolnego dostępnego inicjatora kanału w grupie współużytkowania kolejek. Jest to opisane w sekcji [“Kanały współużytkowane”](#) na stronie 216.

Współużytkowane kanały wychodzące przyjmują komunikaty, które są wysyłane ze współużytkowanej kolejki transmisji. Informacje na temat statusu kanału współużytkowanego są przechowywane w jednym miejscu dla całego poziomu grupy współużytkowania kolejki. Oznacza to, że kanał może zostać zrestartowany automatycznie na innym inicjatorze kanału w grupie współużytkowania kolejki, jeśli inicjator kanału, menedżer kolejek lub podsystem komunikacyjny nie powiedzie się. Jest to nazywane *odtworzeniem kanałów równorzędnych* i jest opisane w sekcji [Współużytkowane kanały wychodzące](#).

Dostępność sieci IBM MQ

Komunikaty produktu IBM MQ są przenoszone z menedżera kolejek do menedżera kolejek w sieci produktu IBM MQ przy użyciu kanałów. Konfigurację można zmienić na wielu poziomach, aby zwiększyć dostępność sieci menedżera kolejek oraz zdolność kanału IBM MQ do wykrywania problemów z siecią i do ponownego nawiązania połączenia.

Protokół TCP *Keepalive* jest dostępny dla kanałów TCP/IP. Protokół TCP wysyła pakiety okresowo między sesjami w celu wykrycia awarii sieci. Atrybut kanału KAINTE określa częstotliwość tych pakietów dla kanału.

AdoptMCA zezwala na zakończenie i zastąpienie kanału przez kanał, który jest blokowany podczas przetwarzania odbioru w wyniku wyłączenia sieci i jest zastępowany przez nowe żądanie połączenia. Element *AdoptMCA* jest sterowany za pomocą właściwości *ADOPTMCA* menedżera kolejek za pomocą programu narzędziowego MQSC lub właściwości *AdoptNewMCAType* z interfejsem *programmable Command Formats*.

Opcja *ReceiveTimeout* uniemożliwia trwałe zablokowanie kanału w wywołaniu sieciowym. Parametry inicjatora kanału RCVTIME i RCVTMIN określają parametry limitu czasu odbierania dla kanałów, jako funkcję interwału pulsu. Więcej informacji na ten temat zawiera sekcja [Parametr menedżera kolejek](#).

Korzystanie z programu z/OS Automatic Restart Manager (ARM)

Programu IBM MQ for z/OS można używać w połączeniu z menedżerem automatycznego uruchamiania produktu z/OS (ARM). Jeśli menedżer kolejek lub inicjator kanału nie powiedzie się, menedżer ARM zrestartuje go na tym samym obrazie produktu z/OS. Jeśli z/OS nie powiedzie się, cała grupa powiązanych podsystemów i aplikacji również nie powiedzie się. ARM może automatycznie restartować wszystkie systemy, które nie powiodły się, w predefiniowanym porządku, na innym obrazie z/OS w sysplexie. Jest to tzw. restart międzysystemowy.

ARM umożliwia szybkie odtwarzanie wrażliwych transakcji w środowisku kolejki współużytkowanej. Zapewnia ona również wyższą dostępność, jeśli nie są używane grupy współużytkowania kolejek.

Za pomocą menedżera ARM można zrestartować menedżer kolejek na innym obrazie produktu z/OS w środowisku sysplex w przypadku niepowodzenia programu z/OS.

Aby włączyć automatyczne restartowanie, należy wykonać następujące czynności:

1. Skonfiguruj zestaw danych sprzęgu ARM.
2. Zdefiniuj działania automatycznego restartowania, które mają być wykonywane przez program z/OS w strategii ARM.
3. Uruchom strategię ARM.

Aby automatycznie zrestartować menedżery kolejek w różnych obrazach produktu z/OS, każdy menedżer kolejek w każdym obrazie produktu z/OS, w którym ten menedżer kolejek może zostać zrestartowany, musi być zdefiniowany z unikalną 4-znakową nazwą podsystemu w systemie sysplex.

Użycie ARM z IBM MQ jest opisane w sekcji [Używanie ARM w sieci IBM MQ](#).

Korzystanie z narzędzia z/OS Extended Recovery Facility (XRF)

Produktu IBM MQ można używać w środowisku Extended Recovery Facility (XRF). Wszystkie zestawy danych będące własnością IBM MQ (kod wykonywalny, BSDSs, dzienniki i zestawy stron) muszą znajdować się w pamięci DASD współużytkowanej między aktywnymi i alternatywnymi procesorami XRF.

Jeśli do odtwarzania jest używany produkt XRF, należy zatrzymać menedżer kolejek na aktywnym procesorze i uruchomić go na alternatywnym procesorze. W przypadku systemu CICS można to zrobić za pomocą tabeli listy komend (CLT) udostępnionej przez program CICS lub ręcznie. W przypadku systemu IMS jest to operacja ręczna i należy ją wykonać po zakończeniu przez system koordynujący IMS przełącznika procesora.

Programy narzędziowe produktu IBM MQ muszą zostać zakończone lub zakończony, zanim menedżer kolejek zostanie przełączony na procesor alternatywny. Podczas planowania planów odtwarzania XRF należy starannie rozważyć wpływ tego potencjalnego przerwania.

Należy zadbać o to, aby menedżer kolejek uruchamiał się na alternatywnym procesorze przed zakończeniem działania menedżera kolejek na aktywnym procesorze. Przedwczesny start może spowodować poważne problemy z integralnością danych, katalogu i dziennika. Użycie funkcji GRS (global resource serialization) pomaga uniknąć problemów z integralnością, uniemożliwiając jednoczesne korzystanie z produktu IBM MQ w dwóch systemach. Użytkownik musi uwzględnić BSDS jako zasób chroniony, a ponadto należy uwzględnić aktywne i alternatywne procesory XRF w pierścieniu GRS.

Używanie atrybutu GROUPUR produktu z/OS do odtwarzania w grupie współużytkowania kolejek

Grupy współużytkowania kolejek (QSG) umożliwiają dodatkowe obiekty transakcyjne, które zostały opisane w tym temacie. Atrybut GROUPUR umożliwia aplikacjom klienckim XA dowolne odzyskiwanie transakcji wrażliwej, które może być wymagane, wykonywane na dowolnym elemencie QSG.

Jeśli aplikacja kliencka XA łączy się z grupą współużytkowania kolejek (QSG) za pośrednictwem systemu Sysplex, nie może zagwarantować, do którego konkretnego menedżera kolejek łączy się ta aplikacja. Użycie atrybutu GROUPUR przez menedżery kolejek w grupie współużytkowania kolejek może umożliwić dowolne odtwarzanie transakcji wątpliwych, które może być konieczne do wystąpienia dowolnego członka QSG. Nawet jeśli menedżer kolejek, do którego aplikacja była początkowo połączona, nie jest dostępny, można wykonać odtwarzanie transakcji.

Ta funkcja zwalnia aplikację kliencką XA z dowolnej zależności od konkretnych członków QSG, a tym samym rozszerza dostępność menedżera kolejek. Grupa współużytkowania kolejek jest wyświetlana w aplikacji transakcyjnej jako pojedynczy obiekt udostępniający wszystkie funkcje produktu IBM MQ i bez pojedynczego punktu menedżera kolejek.

Ta funkcja nie jest widoczna dla aplikacji transakcyjnej.

Gdzie znaleźć więcej informacji o dostępności

Więcej informacji na temat tych tematów można znaleźć w następujących źródłach:

| <i>Tabela 23. Gdzie znaleźć więcej informacji o dostępności</i> | |
|---|---|
| Temat | Gdzie szukać |
| Grupy współużytkowania kolejek | “Współużytkowane kolejki i grupy współużytkowania kolejek” na stronie 172 |
| Parametry systemu | Konfigurowanie parametrów systemowych |
| Korzystanie z menedżera automatycznego restartu Programy narzędziowe | Używanie ARM w sieci IBM MQ |
| Komendy MQSC | Komendy MQSC |

z/OS

Monitorowanie i statystyka w systemie IBM MQ for z/OS

Produkt IBM MQ for z/OS zawiera zestaw narzędzi do monitorowania menedżera kolejek i zbierania statystyk.

Produkt IBM MQ udostępnia narzędzia do monitorowania systemu i gromadzenia statystyk. Więcej informacji na temat tych narzędzi można znaleźć w następujących sekcjach:

- [“Monitorowanie bezpośrednio” na stronie 290](#)
- [“IBM MQ ślad” na stronie 290](#)
- [“Zdarzenia” na stronie 291](#)

Monitorowanie bezpośrednio

Produkt IBM MQ zawiera następujące komendy służące do monitorowania statusu obiektów produktu IBM MQ :

- Komenda DISPLAY CHSTATUS wyświetla status określonego kanału.
- Komenda DISPLAY QSTATUS wyświetla status określonej kolejki.
- Komenda DISPLAY CONN wyświetla status określonego połączenia.

Więcej informacji na temat tych komend zawiera sekcja [Komendy MQSC](#).

IBM MQ ślad

Produkt IBM MQ udostępnia narzędzie śledzenia, którego można użyć do zebrania następujących informacji w czasie działania menedżera kolejek:

Statystyki wydajności

Śledzenie statystyk zbiera następujące informacje, aby pomóc w monitorowaniu wydajności i dostrajania systemu:

- Liczniki różnych żądań MQI (statystyki menedżera komunikatów)
- Liczniki różnych żądań obiektów (statystyki menedżera danych)
- Informacje na temat wykorzystania produktu Db2 (statystyki menedżera produktu Db2)
- Informacje na temat wykorzystania narzędzia CF (Coupling Facility Manager)
- Informacje na temat wykorzystania SMDS (statystyki współużytkowanego zestawu danych komunikatów)
- Informacje o wykorzystaniu puli buforów (statystyki menedżera buforów)
- Informacje na temat rejestrowania (statystyki menedżera rejestrowania)
- Informacje na temat wykorzystania pamięci masowej (statystyki menedżera pamięci masowej)
- Informacje o żądaniach blokad (statystyka menedżera blokad)

Dane księgowe

- Śledzenie rozliczeniowe zbiera informacje na temat czasu procesora zużytego na przetwarzanie wywołań MQI oraz liczby żądań MQPUT i MQGET wykonanych przez konkretnego użytkownika.
- Produkt IBM MQ może również zbierać informacje o każdym zadaniu przy użyciu programu IBM MQ. Dane te są gromadzone jako rekordy rozliczeniowe na poziomie wątku. W przypadku każdego wątku program IBM MQ zbiera również informacje o każdej kolejce używanej przez ten wątek.

Dane wygenerowane przez dane śledzenia są wysyłane do narzędzia System Management Facility (SMF) lub do uogólnionego narzędzia śledzenia (GTF).

Zdarzenia

Zdarzenia produktu IBM MQ udostępniają informacje o błędach, ostrzeżeniach i innych istotnych wystąpieniach w menedżerze kolejek. Dzięki włączeniu tych zdarzeń do własnej aplikacji zarządzania systemem można monitorować działania w wielu menedżerach kolejek, w przypadku wielu aplikacji produktu IBM MQ. W szczególności można monitorować wszystkie menedżery kolejek w systemie z poziomu jednego menedżera kolejek.

Zdarzenia mogą być zgłaszane za pośrednictwem mechanizmu raportowania napisanego przez użytkownika do aplikacji administracyjnej, która obsługuje prezentacje zdarzeń do operatora. Zdarzenia umożliwiają również aplikacjom działającym jako agenty dla innych sieci administracyjnych, na przykład NetView, monitorowanie raportów i tworzenie odpowiednich alertów.

Zadania pokrewne

[Używanie usługi śledzenia produktu IBM MQ](#)

[Korzystanie z zdarzeń produktu IBM MQ](#)

z/OS

Dyspozycja jednostki odtwarzania w systemie z/OS

Niektóre aplikacje transakcyjne mogą używać grupy, a nie QMGR, jednostki odtwarzania po nawiązaniu połączenia z menedżerem kolejek w grupie współużytkowania kolejek (QSG), określając nazwę QSG podczas nawiązywania połączenia, a nie nazwy menedżera kolejek. Dzięki temu odtwarzanie transakcji może być bardziej elastyczne i stabilne przez usunięcie wymagania ponownego połączenia z tym samym menedżerem kolejek w QSG.

Transakcje uruchomione przez aplikacje, które nawiązują połączenie przy użyciu nazwy grupy współużytkowania kolejki, mają również dyspozycję odzyskiwania jednostki GROUP.

Gdy aplikacja transakcyjna łączy się z dyspozycją odzyskiwania jednostki GROUP, jest ona logicznie połączona z grupą współużytkowania kolejek i nie ma powinowactwa do żadnego konkretnego menedżera kolejek. Wszystkie dwufazowe transakcje zatwierdzania, które zostały rozpoczęte, które zakończyły phase-1 procesu zatwierdzania, czyli są wątpliwe, mogą być docieklive i rozstrzygane, po połączeniu z dowolnym menedżerem kolejek w QSG. W scenariuszu odtwarzania oznacza to, że koordynator transakcji nie musi ponownie łączyć się z tym samym menedżerem kolejek, który może być w tym czasie niedostępny.

Aplikacje, które łączą się z jednostką QMGR z dyspozycją odtwarzania, mają bezpośrednie powinowactwo do menedżera kolejek, z którym są połączone. W scenariuszu odtwarzania koordynator transakcji musi ponownie połączyć się z tym samym menedżerem kolejek w celu rozstrzygnięcia wszystkich wątpliwych transakcji, niezależnie od tego, czy menedżer kolejek należy do grupy współużytkowania kolejek.

Gdy aplikacje określają nazwę grupy współużytkowania kolejki i w związku z tym nawiąże połączenie z menedżerem kolejek w QSG z dyspozycją odzyskiwania jednostki GROUP, to grupa współużytkowania kolejki jest logicznie odrębnym menedżerem zasobów. Oznacza to, że transakcje wątpliwe są widoczne tylko dla aplikacji, jeśli ponownie łączy się z tą samą jednostką dyspozycji odtwarzania. Wątpliwe transakcje z jednostką QMGR nie są widoczne dla aplikacji, które łączą się z jednostką grupy, która ma dyspozycję odtwarzania i vice versa.

Pojęcia pokrewne

“Włączanie jednostek odzyskiwania grupy” na stronie 292

Grupa współużytkowania kolejek może skonfigurować i włączyć obsługę jednostek odzyskiwania grupy.

“Obsługa aplikacji” na stronie 293

Ta strona służy do określania, które aplikacje mogą łączyć się z jednostką grupy, która ma dyspozycję odtwarzania.

Włączanie jednostek odzyskiwania grupy

Grupa współużytkowania kolejek może skonfigurować i włączyć obsługę jednostek odzyskiwania grupy.

Aby użyć jednostek grupy do odtwarzania w menedżerze kolejek w QSG, należy włączyć atrybut menedżera kolejek GROUPUR. Więcej informacji na temat tego pojęcia można znaleźć w sekcji “Dyspozycja jednostki odtwarzania w systemie z/OS” na stronie 291 przed przeczytaniem reszty tego tematu.

Gdy atrybut menedżera kolejek GROUPUR jest włączony, menedżer kolejek akceptuje nowe połączenia z jednostką grupy, która ma dyspozycję odtwarzania. Jeśli ten atrybut zostanie wyłączony, nowe połączenia z tą dyspozycją nie zostaną zaakceptowane, mimo że aplikacje już połączone nie będą mieć wpływu na ich odłączenie.

Gdy aplikacja łączy się z jednostką grupy, która ma dyspozycję odtwarzania, a także sprawdzenie, jakie transakcje są wątpliwe lub próby rozstrzygnięcia transakcji, która została uruchomiona w innym miejscu w grupie współużytkowania kolejek (QSG), menedżer kolejek, do którego jest teraz połączony, musi mieć możliwość komunikowania się z innymi elementami grupy współużytkowania kolejek, aby mógł przetworzyć żądanie. W tym celu używana jest kolejka współużytkowana o nazwie SYSTEM.QSG.UR.RESOLUTION.QUEUE. Ta kolejka musi znajdować się w odtwarzalnej strukturze aplikacji o nazwie CSQSYSAPPL. Struktura musi być odtwarzalna, ponieważ komunikaty trwałe są przechowywane w tej kolejce podczas przetwarzania żądań rozwiązania.

Zanim możliwe będzie włączenie jednostek grupy odzyskiwania, należy upewnić się, że zdefiniowano strukturę narzędzia CF i kolejkę współużytkowaną. Definicji można użyć w przykładzie CSQ4INSS . Gdy kolejka jest definiowana lub wykrywana podczas uruchamiania, każdy menedżer kolejek w grupie współużytkowania kolejek otwiera kolejkę w taki sposób, aby mogła odbierać przychodzące żądania. Aby usunąć lub przenieść kolejkę, ponieważ została ona niepoprawnie zdefiniowana, można zażądać, aby menedżery kolejek zamykowały na nim otwarte uchwyty, aktualizując obiekt kolejki w taki sposób, aby zahamował żądania MQGET. Po wprowadzeniu niezbędnych poprawek, zezwalanie aplikacjom na pobieranie komunikatów z kolejki po raz kolejny powoduje, że każdy menedżer kolejek będzie go ponownie otwierał. Użyj komendy DISPLAY QSTATUS, aby określić, które uchwyty są otwarte w kolejce.

Po zakończeniu tej konfiguracji można następnie włączyć jednostki odzyskiwania grupy w każdym menedżerze kolejek, z którym aplikacje transakcyjne mają mieć możliwość nawiązywania połączenia z jednostką grupy, która ma dyspozycję odtwarzania. Nie muszą to być wszystkie menedżery kolejek w grupie współużytkowania kolejek, ale jeśli zostanie wybrana opcja włączenia tej funkcji tylko w podzbiorze grupy współużytkowania kolejek, należy upewnić się, że aplikacje próbują połączyć się tylko z menedżerami kolejek, na których włączono tę funkcję. Więcej informacji na ten temat zawiera sekcja [“Obsługa aplikacji”](#) na stronie 293.

Podczas próby włączenia atrybutu menedżera kolejek GROUPUR wykonywane jest kilka operacji sprawdzania konfiguracji. Menedżer kolejek sprawdza, czy:

- Należy do grupy współużytkowania kolejek.
- Kolejka współużytkowana o nazwie SYSTEM.QSG.UR.RESOLUTION.QUEUE została zdefiniowana, zgodnie z definicją w CSQ4INSS.
- SYSTEM.QSG.UR.RESOLUTION.QUEUE znajduje się na odtwarzalnej strukturze CF o nazwie CSQSYSAPPL.

Jeśli którekolwiek z powyższych kontroli nie powiedzie się, atrybut GROUPUR pozostanie wyłączone, a zwrócony zostanie kod komunikatu.

Te sprawdzenia konfiguracji są również wykonywane podczas uruchamiania menedżera kolejek, jeśli włączony jest atrybut menedżera kolejek. Jeśli którekolwiek z kontroli nie powiedzie się podczas uruchamiania jednostek odzyskiwania grupy, zostanie wyłączone, a menedżer kolejek wysyła komunikat identyfikujący, które sprawdzenie nie powiodło się. Po wykonaniu niezbędnych czynności naprawczych należy ponownie włączyć atrybut menedżera kolejek.

Obsługa aplikacji

Ta strona służy do określania, które aplikacje mogą łączyć się z jednostką grupy, która ma dyspozycję odtwarzania.

Obsługa jednostki GROUP w dyspozycji dyspozycyjności odzyskiwania jest ograniczona do pewnych typów aplikacji transakcyjnych, dla których IBM MQ for z/OS jest menedżerem zasobów, ale nie jest koordynatorem transakcji. Obecnie obsługiwane aplikacje transakcyjne to:

- Rozszerzone aplikacje klienckie transakcyjne produktu IBM MQ
- Aplikacje IBM MQ classes for JMS działające na serwerze aplikacji, takie jak WebSphere Application Server.
- Aplikacje produktu CICS działające w produkcie CICS Transaction Server 4.2 lub nowszym, gdy definicja zasobu MQCONN produktu CICS jest skonfigurowana z elementem RESYNCMEMBER (GROUPRESYNC).

Pojęcia pokrewne

[“Rozszerzone aplikacje klienckie transakcyjne produktu IBM MQ”](#) na stronie 293

Ta strona służy do określania, w jaki sposób rozszerzone aplikacje klienckie transakcyjne produktu IBM MQ mogą używać jednostki grupy dyspozycyjnej odtwarzania.

[“Aplikacje CICS”](#) na stronie 294

Ta strona służy do określania, w jaki sposób produkt CICS może używać jednostki grupy dyspozycyjności odzyskiwania.

Rozszerzone aplikacje klienckie transakcyjne produktu IBM MQ

Ta strona służy do określania, w jaki sposób rozszerzone aplikacje klienckie transakcyjne produktu IBM MQ mogą używać jednostki grupy dyspozycyjnej odtwarzania.

Przykładem rozszerzonej aplikacji klienckiej transakcyjnej produktu IBM MQ jest aplikacja, która używa produktu JMS i działa w produkcie WebSphere Application Server, łącząc się z serwerem IBM MQ za pośrednictwem protokołu TCP/IP, a nie z powiązaniem lokalnym. Te aplikacje klienckie łączą się z serwerem IBM MQ for z/OS za pośrednictwem połączeń sieciowych, takich jak TCP/IP. W przypadku tych aplikacji jest to wartość określona dla parametru QMNAME łańcucha xa_info przekazanego w wywołaniu xa_open, który określa, czy używana jest jednostka QMGR lub jednostka grupy, w której ma być oddane

odtworzenie. Więcej informacji na temat komendy `xa_open` zawiera sekcja [Format łańcucha `xa_open`](#) i [Dodatkowe przetwarzanie błędów dla `xa_open`](#). W przypadku aplikacji JMS jest to realizowane przez określenie nazwy grupy współużytkowania kolejek (QSG) w obiekcie `ConnectionFactory` zamiast nazwy konkretnego menedżera kolejek.

Aby aplikacje klienckie XA korzystały z jednostki grupy, należy skonfigurować protokół TCP/IP w taki sposób, aby umożliwić kierowanie aplikacji klienckich do menedżerów kolejek w grupie współużytkowania kolejek z włączonym atrybutem `GROUPUR`, a nie w określonym menedżerze kolejek. Jedną z dynamicznych technologii wirtualnych adresów IP, które można wykorzystać w tym celu jest dystrybutor `SysPlex` produktu `z/OS`. Więcej informacji na ten temat zawiera sekcja [z/OS Communications Server i z/OS Podstawowe umiejętności: Dynamiczne adresowanie wirtualne](#). Jeśli chcesz włączyć jednostki odzyskiwania grupy w podzbiorze menedżerów kolejek w grupie współużytkowania kolejek, upewnij się, że aplikacje klienckie nie mogą być kierowane do tych, na których nie jest włączony. Dotyczy to wszystkich menedżerów kolejek w wersji wcześniejszej niż `IBM WebSphere MQ 7.0.1`.

Aby użyć funkcji rozdysponowania jednostki grupy w aplikacji klienckiej produktu JMS, należy użyć bibliotek klienta `IBM WebSphere MQ 7.0.1 JMS`. Wcześniejsze wersje bibliotek klienta produktu JMS używają jednostki dyspozycyjności odtwarzania menedżera kolejek (`QMGR`).

Aplikacje klienckie nie muszą łączyć się z grupą współużytkowania kolejek przy użyciu kanałów współużytkowanych.

Aplikacje CICS

Ta strona służy do określania, w jaki sposób produkt CICS może używać jednostki grupy dyspozycyjności odzyskiwania.

CICS 4.2 i później udostępnia opcję resynchronizacji grupy, `RESYNCMEMBER (GROUPRESYNC)` w definicji zasobu `MQCONN`. Produkt CICS skonfigurowany przy użyciu tej opcji może łączyć się z dowolnym odpowiednim menedżerem kolejek w grupie współużytkowania kolejek, która jest uruchomiona na tej samej partycji LPAR, co ten region produktu CICS. W celu obsługi opcji `CICS GROUPRESYNC` menedżer kolejek musi być uruchomiony w produkcie `MQ V7.1` lub nowszym i musi być włączony dla obsługi `GROUPUR`.

Transakcje działające w obszarze CICS połączonym z produktem `MQ` za pomocą komendy `GROUPRESYNC` tworzą jednostki pracy z dyspozycją odzyskiwania jednostki `GROUP`.

Można użyć komendy `RESYNCMEMBER (GROUPRESYNC)`, aby umożliwić szybsze odtwarzanie po awarii menedżera kolejek, ponieważ umożliwia to regionowi CICS natychmiastowe nawiązanie połączenia z alternatywnym kwalifikowanym menedżerem kolejek działającym na tej samej partycji LPAR, co w razie potrzeby rozwiąże wszystkie transakcje wątpliwe, nie czekając na restart menedżera kolejek.

`RESYNCMEMBER (GROUPRESYNC)` umożliwia również bardziej elastyczne opcje restartu dla produktu CICS. Region CICS ze swoim połączeniem `MQ` skonfigurowanym do korzystania z współużytkowanych kolejek `GROUPRESYNC` i `MQ` może zostać zrestartowany na dowolnej partycji LPAR, na której jest uruchomiony menedżer kolejek jako członek tej samej grupy współużytkowania kolejek.

IBM MQ i inne produkty z/OS

W tym temacie opisano, w jaki sposób produkt `IBM MQ` może pracować z innymi produktami `z/OS`.

Pojęcia pokrewne

[“IBM MQ i CICS” na stronie 295](#)

Wszystkie wersje produktu CICS obsługiwane przez produkt `IBM MQ 9.0.0i` i nowsze wersje korzystają z dostarczonej wersji adaptera i mostu w wersji CICS.

[“IBM MQ for z/OS i WebSphere Application Server” na stronie 301](#)

Ten temat zawiera informacje na temat używania produktu `IBM MQ for z/OS` przez produkt `WebSphere Application Server`.

Odsyłacze pokrewne

[“IBM MQ i IMS” na stronie 296](#)

W tym temacie opisano sposób działania produktu IBM MQ z produktem IMS. Adapter IMS umożliwia połączenie menedżera kolejek z produktem IMS i umożliwia aplikacjom produktu IMS korzystanie z interfejsu MQI.

[“Adaptory IBM MQ i z/OS Batch, TSO i RRS” na stronie 300](#)

W tym temacie opisano sposób działania produktu IBM MQ przy użyciu adapterów z/OS Batch, TSO i RRS.

IBM MQ i CICS

Wszystkie wersje produktu CICS obsługiwane przez produkt IBM MQ 9.0.0i i nowsze wersje korzystają z dostarczonej wersji adaptera i mostu w wersji CICS .

Więcej informacji na temat konfigurowania adaptera IBM MQ CICS oraz komponentów IBM MQ CICS bridge zawiera sekcja [Konfigurowanie połączeń z produktem MQ](#) w dokumentacji produktu CICS .

Zadania pokrewne

[Używanie produktu IBM MQ z produktem CICS](#)

Dołączanie grupy CICS

Program CICS group attach umożliwia regionowi CICS nawiązanie połączenia z dowolnym aktywnym elementem grupy współużytkowania kolejek produktu IBM MQ na tej samej partycji LPAR, a nie z określeniem pojedynczego menedżera kolejek. Program CICS nadal łączy się z jednym menedżerem kolejek w danym momencie.

Do obsługi dołączania grupy CICS wymagane są co najmniej dwa menedżery kolejek na partycji LPAR. Korzystanie z funkcji dołączania grup zapewnia wyższą dostępność, ponieważ nie jest potrzebny aktywny menedżer kolejek. Program CICS łączy się z dowolnym menedżerem kolejek w grupie współużytkowania kolejek na partycji LPAR.

Więcej informacji na ten temat można znaleźć w dokumentacji produktu CICS na zasobie MQCONN.

CICS próby nawiązania połączenia z nazwą MQNAME przekazywały się tak, jakby były menedżerem kolejek:

- Jeśli menedżer kolejek istnieje i jest aktywny, połączenie będzie działać.
- Jeśli nawiązanie połączenia nie powiedzie się, program CICS wysyła zapytanie o status menedżerów kolejek w grupie, aby upewnić się, które są aktywne na tej samej partycji LPAR.
- Jeśli jest aktywnych wiele menedżerów kolejek, program CICS sprawdza status RESYNCMEMBER (YES) i jednostki pracy, aby określić, czy program CICS musi nawiązać połączenie, czy powinien nawiązać połączenie z określonym elementem, czy też powinien czekać, czy nie jest aktywny.
- Jeśli nie ma potrzeby łączenia się z określonym elementem, produkt CICS wybiera menedżera kolejek (przy użyciu algorytmu randomizowanego).
- CICS próbuje połączyć się z wybranym menedżerem kolejek.
- Jeśli próba zakończy się niepowodzeniem, wówczas w zależności od kodu powrotu program CICS wybierze następny element, a następnie ponownie przechodzi przez pętlę wyboru.
- Jeśli żaden menedżer kolejek nie jest aktywny, program CICS wysyła wiele połączeń z listą menedżerów kolejek i czeka na ECBLIST do momentu, gdy pierwszy menedżer kolejek stanie się dostępny.

Pojęcia pokrewne

[“Jednostki odzyskiwania grupy \(GROUPUR\) dla CICS” na stronie 296](#)

IBM MQ GROUPUR dla CICS zapewnia odtwarzanie równorzędne dla wątpliwych jednostek pracy w grupie współużytkowania kolejek (QSG). Jeden menedżer kolejek produktu IBM MQ może rozstrzygać wątpliwe jednostki pracy w imieniu innego menedżera kolejek w grupie współużytkowania kolejek. Oznacza to, że jeśli program CICS ponownie łączy się za pośrednictwem grupy przyłączonej do innego menedżera kolejek w QSG, może rozstrzygać transakcje wątpliwe z poprzedniego połączenia IBM MQ .

Informacje pokrewne

[Obsługa grup współużytkowania kolejek produktu IBM MQ](#)

Jednostki odzyskiwania grupy (GROUPUR) dla CICS

IBM MQ GROUPUR dla CICS zapewnia odtwarzanie równorzędne dla wątpliwych jednostek pracy w grupie współużytkowania kolejek (QSG). Jeden menedżer kolejek produktu IBM MQ może rozstrzygać wątpliwe jednostki pracy w imieniu innego menedżera kolejek w grupie współużytkowania kolejek. Oznacza to, że jeśli program CICS ponownie łączy się za pośrednictwem grupy przyłączonej do innego menedżera kolejek w QSG, może rozstrzygać transakcje wątpliwe z poprzedniego połączenia IBM MQ.

Jeśli region produktu CICS współpracuje z menedżerem kolejek, a menedżer kolejek zostanie zakończony nieprawidłowo, odzyskane są wszystkie transakcje wątpliwe. Eliminuje to konieczność oczekiwania przez region produktu CICS na oczekiwanie na menedżer kolejek, z którym pracował, aby zrestartować, a następnie rozwiązać wszystkie wątpliwe jednostki pracy. Oznacza to, że na partycji LPAR potrzebne są co najmniej dwa menedżery kolejek, dzięki czemu program CICS może połączyć się z innym menedżerem kolejek w przypadku nieprawidłowego zakończenia pierwszego menedżera kolejek.

Nowe ustawienie RESYNCMEMBER (GROUPRESYNC) w definicji MQCONN produktu CICS :

- Korzysta z funkcji przyłączenia grupy IBM MQ i odtwarzania równorzędnego.
- Wymaga, aby menedżer kolejek z atrybutem GROUPUR był włączony.
- Nadal obsługuje istniejące ustawienia produktu CICS MQCONN RESYNCMEMBER (YES i NO):
 - Korzysta z istniejącej funkcji przyłączenia grupy CICS i nie ma odtwarzania równorzędnego.
 - Zmiana ustawień RESYNCMEMBER spowoduje, że następnym razem program CICS połączy się z serwerem IBM MQ.

Pojęcia pokrewne

[“Włączanie jednostek odzyskiwania grupy” na stronie 292](#)

[Grupa współużytkowania kolejek może skonfigurować i włączyć obsługę jednostek odzyskiwania grupy.](#)

IBM MQ i IMS

W tym temacie opisano sposób działania produktu IBM MQ z produktem IMS. Adapter IMS umożliwia połączenie menedżera kolejek z produktem IMS i umożliwia aplikacjom produktu IMS korzystanie z interfejsu MQI.

Dodatkowy most IBM MQ - IMS umożliwia aplikacjom uruchamianie aplikacji IMS, która nie korzysta z interfejsu MQI. Oznacza to, że można korzystać z wcześniejszych aplikacji z produktem IBM MQ, bez konieczności ich przepisania.

Więcej informacji na temat tych komponentów można znaleźć w następujących podtematach:

Pojęcia pokrewne

[Aplikacje pomostowe IMS i IMS w systemie IBM MQ for z/OS](#)

Zadania pokrewne

[Konfigurowanie adaptera IMS](#)

[Konfigurowanie mostu IMS](#)

[Działanie adaptera IMS](#)

Odsyłacze pokrewne

[MQIIH-nagłówek informacji IMS](#)

Adapter IMS

Adapter IMS jest interfejsem między programami aplikacji IMS a podsystemem IBM MQ.

Adaptory IBM MQ umożliwiają różnym środowiskom aplikacji wysyłanie i odbieranie komunikatów za pośrednictwem sieci kolejowania komunikatów. Adapter IMS jest interfejsem między programami aplikacji IMS a podsystemem IBM MQ. Umożliwia on programom aplikacji IMS korzystanie z interfejsu MQI.

Adapter IMS odbiera i interpretuje żądania dostępu do produktu IBM MQ przy użyciu zewnętrznego narzędzia dołączania podsystemu (External Subsystem Attach Facility-ESAF) udostępnionego przez produkt IMS. Ten obiekt został opisany w publikacji *IMS Customization Guide*. Zwykle program IMS łączy się z produktem IBM MQ automatycznie bez interwencji operatora.

Adapter IMS umożliwia dostęp do zasobów produktu IBM MQ dla programów działających w następujących trybach lub w następujących stanach:

- Tryb zadania (TCB)
- Stan problemu
- Tryb niepamięciowy
- Tryb rejestru bez dostępu

Adapter udostępnia wątek połączenia z bloku kontrolnego zadania aplikacji (TCB) do produktu IBM MQ.

Adapter obsługuje protokół zatwierdzania dwufazowego w celu wprowadzenia zmian w zasobach należących do produktu IBM MQ z IMS pełnionym jako koordynator punktu synchronizacji. Konwersacje, w których IMS nie jest koordynatorem punktu synchronizacji, na przykład konwersacje zabezpieczone przez APPC (SYNCLVL = SYNCPT), nie są obsługiwane przez adapter IMS.

Adapter udostępnia także transakcję monitora wyzwalacza (CSQQTRMN). Jest to opisane w sekcji [“Monitor wyzwalacza IMS” na stronie 297](#).

Produkt IBM MQ można używać z programem IMS Extended Recovery Facility (XRF), aby pomóc w odzyskaniu pomocy z poziomu błędu IMS. Więcej informacji na temat XRF zawiera podręcznik *Podręcznik administrowania produktowi IMS : System*.

Korzystanie z adaptera

Aplikacje i adapter IMS działają w tej samej przestrzeni adresowej. Menedżer kolejek jest oddzielony, w swojej własnej przestrzeni adresowej.

Każdy program, który wysyła wywołania MQI do odpowiedniego modułu interfejsu językowego produktu IMS, musi być dowiązywanie-edytowany, a jeśli nie używa dynamicznych wywołań MQI, dostarczony przez produkt IBM MQ kod pośredniczący interfejsu API, CSQQSTUB. Gdy aplikacja wysyła wywołanie MQI, kod pośredniczący przekazuje sterowanie do adaptera za pomocą zewnętrznego interfejsu podsystemu IMS, który zarządza przetwarzaniem żądania przez menedżer kolejek komunikatów.

Administrowanie systemem i jego działanie z produktem IMS

Autoryzowany operator terminalu IMS może wydawać komendy IMS do sterowania i monitorowania połączenia z programem IBM MQ. Jednak operator terminalu IMS nie ma żadnej kontroli nad przestrzenią adresową IBM MQ. Na przykład operator nie może zamknąć programu IBM MQ z przestrzeni adresowej IMS.

Ograniczenia

Następujące wywołania funkcji API produktu IBM MQ nie są obsługiwane w ramach aplikacji przy użyciu adaptera IMS:

- MQCB
- MQCB_FUNCTION
- Komenda MQCTL

Monitor wyzwalacza IMS

The IMS trigger monitor (**CSQQTRMN**) is an IBM MQ-supplied IMS application that starts an IMS transaction when an IBM MQ event occurs, for example, when a message is put onto a specific queue.

Jak to działa?

Gdy komunikat jest umieszczany w kolejce komunikatów aplikacji, wyzwalacz jest generowany, jeśli spełnione są warunki wyzwalacza. Następnie menedżer kolejek zapisuje komunikat (zawierający niektóre dane zdefiniowane przez użytkownika), znany jako *komunikat wyzwalacza*, do kolejki inicjuj określonej dla tej kolejki komunikatów. W środowisku IMS można uruchomić instancję CSQQTRMN w celu monitorowania kolejki inicjuj. i pobierania z niej komunikatów wyzwalacza po ich przybyciu. Zwykle CSQQTRMN planuje inną transakcję IMS przez instrukcję INSERT (ISRT) do kolejki komunikatów produktu IMS . Uruchomiona aplikacja IMS odczytuje komunikat z kolejki komunikatów aplikacji, a następnie przetwarza ją. CSQQTRMN musi być uruchamiane jako niewiadomość BMP.

Każda kopia usług CSQQTRMN jest kolejką inicjacji. Po uruchomieniu monitor wyzwalacza jest uruchamiany do momentu zakończenia działania IBM MQ lub IMS .

Makro APPLCTN dla CSQQTRMN musi określać SCHDTYP=PARALLEL.

Ponieważ monitor wyzwalacza jest BMP zorientowanym na zadanie, transakcje IMS uruchamiane przez monitor wyzwalacza zawierają następujące elementy:

- Odstępy w polu LTERM dla IOPCB
- Nazwa PSB wyzwalacza BMP monitora w polu ID użytkownika IOPCB

Jeśli docelowa transakcja IMS jest chroniona przez serwer Security Server (wcześniej znany jako RACF), może być konieczne zdefiniowanie CSQQTRMN jako identyfikatora użytkownika na serwerze Security Server.

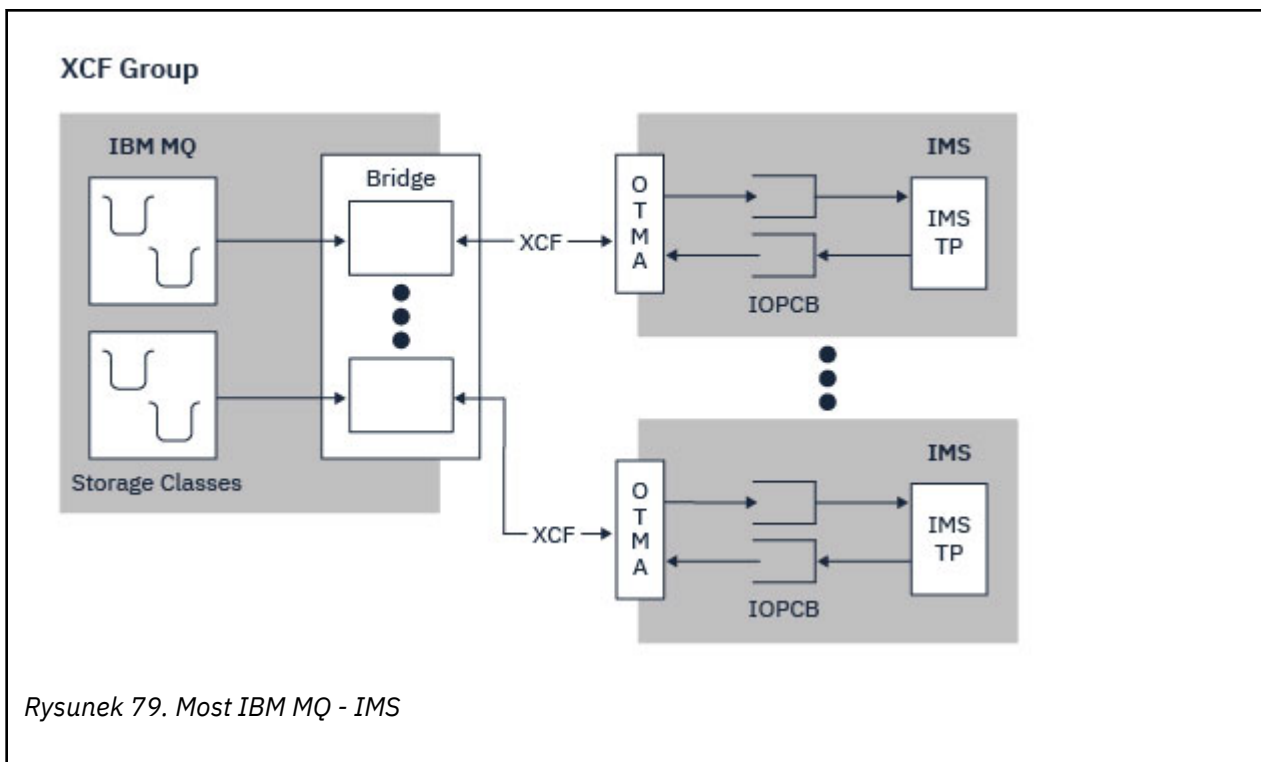
Most IBM MQ - IMS

Most IBM MQ - IMS jest komponentem produktu IBM MQ for z/OS , który umożliwia bezpośredni dostęp z aplikacji produktu IBM MQ do aplikacji w systemie IMS .

Most IBM MQ - IMS umożliwia *niejawne wsparcie MQI*. Oznacza to, że można ponownie zaprojektować wcześniejsze aplikacje, które były kontrolowane przez terminale połączone z programem 3270, aby były kontrolowane przez komunikaty produktu IBM MQ , bez konieczności ponownego zapisywania, ponownego kompilowania lub ponownego połączenia. Most jest klientem typu IMS *Open Transaction Manager Access* (OTMA).

W aplikacjach pomostowych nie ma żadnych wywołań IBM MQ w aplikacji IMS . Aplikacja pobiera swoje dane wejściowe za pomocą komendy GET UNIQUE (GU) do IOPCB i wysyła dane wyjściowe przy użyciu ISRT do IOPCB. Aplikacje produktu IBM MQ używają nagłówka IMS (struktura MQIIH) w danych komunikatu, aby upewnić się, że aplikacje mogą być wykonywane w taki sam sposób, jak w przypadku, gdy są one sterowane przez nieprogramowalne terminale. Jeśli używana jest aplikacja IMS , która przetwarza komunikaty w wielu segmentach, należy zauważyć, że wszystkie segmenty powinny być zawarte w jednym komunikacie IBM MQ .

Most IMS jest ilustrowany w programie Rysunek 79 na stronie 299.



Rysunek 79. Most IBM MQ - IMS

Menedżer kolejek może łączyć się z jednym lub większą liczbą systemów IMS, a więcej niż jeden menedżer kolejek może łączyć się z jednym systemem IMS. Jedynym ograniczeniem jest to, że wszystkie muszą należeć do tej samej grupy XCF i muszą znajdować się w tym samym sysplexie.

Informacje na temat konfigurowania mostu IMS i dodawania dodatkowego połączenia z produktem IMS do tego samego menedżera kolejek zawiera sekcja [Konfigurowanie mostu IMS](#).

Co to jest OTMA?

IMS OTMA to protokół bezpołączeniowego klienta/serwera oparty na transakcji, który działa w systemie IMS 5.1 lub nowszym. Działa on jako interfejs dla serwerów komunikacyjnych opartych na hostach, które uzyskują dostęp do aplikacji IMS TM za pomocą narzędzia *Cross Systems coupling facility (XCF) z/OS*.

OTMA umożliwia klientom łączenie się z produktem IMS w celu zapewnienia wysokiej wydajności interakcji między klientami a IMS w przypadku dużej sieci lub dużej liczby sesji. OTMA jest implementowane w środowisku sysplex w systemie z/OS. Oznacza to, że domena OTMA jest ograniczona do domeny XCF.

Monitorowanie zasobów OTMA

Obsługa komunikatów protokołu X'3C' OTMA, dostępna w wersji IMS v10 lub nowszej, została dodana do mostu IBM MQ - IMS w produkcie IBM MQ for z/OS v7.1. Te komunikaty są wysyłane do klientów OTMA za pomocą programu IMS w celu zgłoszenia jego statusu poprawności. Jeśli partner IMS nie może przetworzyć wysłanego wolumenu żądań transakcji, powiadomi on IBM MQ o wystąpieniu ostrzeżenia powodziowego. W odpowiedzi IBM MQ zwolni wskaźnik, przy którym żądania są wysyłane przez most. Jeśli produkt IMS nadal nie może przetworzyć żądań transakcji, a pełny warunek powodziowy powoduje, że wszystkie elementy TPIPEs zostały zawieszony dla partnera IMS, zostaną zawieszony. Po otrzymaniu powiadomienia od partnera IMS, że warunek powodziowy lub warunek powodziowy został zwolniony IBM MQ, w razie potrzeby wznowi wszystkie zawieszony TPIPEs, a następnie stopniowo zwiększą szybkość wysyłania żądań transakcji do momentu osiągnięcia maksymalnej szybkości. Komunikaty konsoli są wysyłane przez produkt IBM MQ w odpowiedzi na zmianę statusu partnerów programu IMS.

Jeśli używane są partnerzy programu IMS v10, należy sprawdzić, czy poprawka PTF UK45082 została zastosowana.

Wprowadzanie transakcji produktu IMS z produktu IBM MQ

Aby wysłać transakcję IMS, która korzysta z mostu, aplikacje umieszczają komunikaty w kolejce IBM MQ jak zwykle. Komunikaty zawierają dane transakcji IMS; mogą mieć nagłówek IMS (struktura MQIIH) lub umożliwiają tworzenie mostu IBM MQ - IMS w celu utworzenia założeń dotyczących danych w komunikacie.

Następnie produkt IBM MQ umieszcza komunikat w kolejce produktu IMS (jest on umieszczany w kolejce w programie IBM MQ po pierwsze, aby umożliwić korzystanie z punktów synchronizacji w celu zapewnienia integralności danych). Klasa pamięci masowej kolejki IBM MQ określa, czy kolejka jest *kolejką OTMA* (czyli kolejką używaną do przesyłania komunikatów do mostu IBM MQ - IMS), a konkretnym partnerem IMS, do którego wysyłane są dane komunikatu.

Zdalne menedżery kolejek mogą również uruchamiać transakcje IMS, zapisując je do tych kolejek OTMA w systemie IBM MQ for z/OS.

Dane zwracane z systemu IMS są zapisywane bezpośrednio w kolejce odpowiedzi IBM MQ określonej w strukturze deskryptora komunikatu (MQMD). (Może to być kolejka transmisji do menedżera kolejek określonego w polu *ReplyToQMgr* w strukturze MQMD).

Pojęcia pokrewne

[Aplikacje pomostowe IMS i IMS w systemie IBM MQ for z/OS](#)

Zadania pokrewne

[Dostosowywanie mostu IMS](#)

Odsyłacze pokrewne

["IBM MQ i IMS" na stronie 296](#)

W tym temacie opisano sposób działania produktu IBM MQ z produktem IMS. Adapter IMS umożliwia połączenie menedżera kolejek z produktem IMS i umożliwia aplikacjom produktu IMS korzystanie z interfejsu MQI.

z/OS

Adaptory IBM MQ i z/OS Batch, TSO i RRS

W tym temacie opisano sposób działania produktu IBM MQ przy użyciu adapterów z/OS Batch, TSO i RRS.

Wprowadzenie do adapterów wsadowych

Adaptory Batch/TSO to interfejs między programami aplikacji IBM MQ i z/OS, które działają pod kontrolą JES, TSO lub z/OS UNIX System Services. Te adaptory umożliwiają programom aplikacji z/OS korzystanie z interfejsu MQI.

Adaptory zapewniają dostęp do zasobów produktu IBM MQ dla programów działających w następujących trybach lub w następujących stanach:

- Tryb zadania (TCB)
- Stan problemu lub nadzorcy
- Tryb niepamięciowy
- Tryb rejestru bez dostępu

Połączenia między aplikacjami i programem IBM MQ są dostępne na poziomie zadania. Adaptory udostępniają wątek połączenia z bloku kontrolnego zadania aplikacji (TCB) do produktu IBM MQ.

Adapter Batch/TSO obsługuje protokół zatwierdzania jednofazowego w przypadku zmian wprowadzonych w zasobach należących do produktu IBM MQ. Nie obsługuje protokołów zatwierdzania wielofazowego. Adapter RRS umożliwia aplikacjom produktu IBM MQ udział w protokołach zatwierdzania dwufazowego z innymi produktami z obsługą RRS, koordynowanymi przez produkt z/OS Resource Recovery Services (RRS).

Adaptory korzystają z usługi z/OS STIMERM w celu zaplanowania zdarzenia asynchronicznego co sekundę. To zdarzenie uruchamia blok żądania przerwania (IRB), który nie wymaga oczekiwania przez zadanie aplikacji zadania wsadowego. Ten IRB sprawdza, czy nie zostało opublikowane zakończenie IBM

MQ przez EBC. Jeśli wypowiedzenie EBC zostało opublikowane, IRB umieszcza wszystkie obiekty ECB aplikacji, które oczekują na zdarzenie w IBM MQ (na przykład sygnał lub oczekiwanie).

Adapter Batch/TSO

Adapter IBM MQ Batch/TSO udostępnia obsługę IBM MQ dla aplikacji z/OS Batch i TSO. Wszystkie programy użytkowe, które działają w ramach zadania wsadowego z/OS Batch lub TSO, muszą mieć z nimi edytowane łącze CSQBSTUB kodu pośredniczącego API. Kod pośredniczący udostępnia aplikację z dostępem do wszystkich wywołań MQI. Zatwierdzanie jednofazowe i wycofanie dla aplikacji jest używane przez wywołanie wywołań MQI **MQCMIT** i **MQBACK**.

Adapter RRS

Usługa *Resource Recovery Services* (RRS) jest podkomponentem produktu z/OS, który udostępnia ogólnosystemową usługę koordynującą zatwierdzanie dwufazowe w produktach z/OS. Adapter IBM MQ Batch/TSO RRS (adapter RRS) udostępnia obsługę produktu IBM MQ dla aplikacji z/OS Batch i TSO, które chcą korzystać z tych usług. Adapter RRS umożliwia produktowi IBM MQ stały się pełnoporatowym uczestnikiem koordynacji RRS. Aplikacje mogą uczestniczyć w przetwarzaniu zatwierdzania dwufazowego z innymi produktami, które obsługują usługi RRS (na przykład Db2).

Adapter RRS udostępnia dwa kody pośredniczące. Należy połączyć programy aplikacji, które mają używać usługi RRS z jednym z tych kodów pośredniczących.

CSQBRSTB

Ten kod pośredniczący umożliwia użycie zatwierdzania dwufazowego i wycofywania dla aplikacji przy użyciu usług odtwarzania zasobów wywoływalnych zasobów RRS zamiast wywołań MQI **MQCMIT** i **MQBACK**.

Konieczne jest również połączenie-edycja modułu ATRSCSS z biblioteki SYS1.CSSLIB z aplikacją. Jeśli używane są wywołania MQI **MQCMIT** i **MQBACK**, kod powrotu MQRC_ENVIRONMENT_ERROR zostanie zwrócony.

CSQBRSI

Ten kod pośredniczący umożliwia korzystanie z wywołań MQI **MQCMIT** i **MQBACK**; Produkt IBM MQ implementuje te wywołania jako wywołania RRS produktu **SRRCMIT** i **SRRBACK**.

Więcej informacji na temat budowania aplikacji, które korzystają z adaptera RRS, zawiera sekcja [Adapter wsadowy RRS](#).

Gdzie znaleźć więcej informacji o adapterach z/OS Batch, TSO i RRS

Więcej informacji na temat tematów w tej sekcji można znaleźć w następujących źródłach:

| Temat | Gdzie szukać |
|--|--|
| Konfigurowanie adapterów wsadowych | Czynność 19: Konfigurowanie adapterów Batch, TSO i RRS |
| Usługi odtwarzania zasobów wywoływalnych RRS | <i>MVS Programming: Callable Services for High Level Languages</i> |

z/OS

IBM MQ for z/OS i WebSphere Application Server

Ten temat zawiera informacje na temat używania produktu IBM MQ for z/OS przez produkt WebSphere Application Server.

Aplikacje napisane w produkcie Java , które działają pod kontrolą produktu WebSphere Application Server , mogą używać specyfikacji Java Messaging Service (JMS) do przesyłania komunikatów. Przesyłanie komunikatów typu punkt z punktem w tym środowisku może być udostępniane przez menedżer kolejek produktu IBM MQ for z/OS .

Zaletą korzystania z menedżera kolejek produktu IBM MQ for z/OS w celu udostępnienia mechanizmu przesyłania komunikatów jest to, że łączenie aplikacji produktu JMS może w pełni uczestniczyć w funkcjach sieci produktu IBM MQ . Na przykład mogą one korzystać z mostu IMS lub wymieniać komunikaty z menedżerami kolejek działającymi na innych platformach.

Połączenie między produktem WebSphere Application Server i menedżerem kolejek

Dla obiektu fabryki połączeń kolejki można wybrać opcję *Transport klienta* lub *Transport powiązań* . Jeśli zostanie wybrany transport powiązań, zarówno produkt WebSphere Application Server , jak i menedżer kolejek muszą istnieć na tym samym obrazie produktu z/OS .

Należy pamiętać, że w przypadku *transportu powiązań* wymagane są biblioteki rodzime.

Oba typy połączeń obsługują aplikacje transakcyjne: transport klienta przy użyciu protokołów XA; transport powiązań przy użyciu kodu pośredniczącego WebSphere Application Server , CSQBWSTB, który korzysta z usług RRS.

Więcej informacji na temat konfigurowania fabryk połączeń kolejki można znaleźć w sekcji *IBM MQ Using Java* .

Korzystanie z funkcji IBM MQ z aplikacji JMS

Domyślnie komunikaty produktu JMS znajdujące się w kolejkach produktu IBM MQ używają nagłówka MQRFH2 do przechowywania niektórych informacji nagłówka komunikatu produktu JMS . Wiele wcześniejszych aplikacji produktu IBM MQ nie może przetwarzać komunikatów z tymi nagłówkami i wymagać ich własnych, charakterystycznych nagłówków, na przykład MQCIH for CICS Bridge, lub MQWIH dla aplikacji IBM MQ Workflow. Więcej szczegółowych informacji na temat tych szczególnych uwag zawiera sekcja [Odzworowanie komunikatów produktu JMS na komunikaty produktu IBM MQ](#).

Managed File Transfer

Produkt Managed File Transfer przesyła pliki między systemami w sposób zarządzany i podlegający kontroli, niezależnie od wielkości pliku lub używanych systemów operacyjnych.

Produkt Managed File Transfer może być używany do tworzenia dostosowanych, skalowalnych i zautomatyzowanych rozwiązań, które umożliwiają zarządzanie transferami plików, zaufanie i zarządzanie nimi. Managed File Transfer eliminuje kosztowne nadmiarowości, obniża koszty konserwacji i maksymalizuje istniejące inwestycje informatyczne.

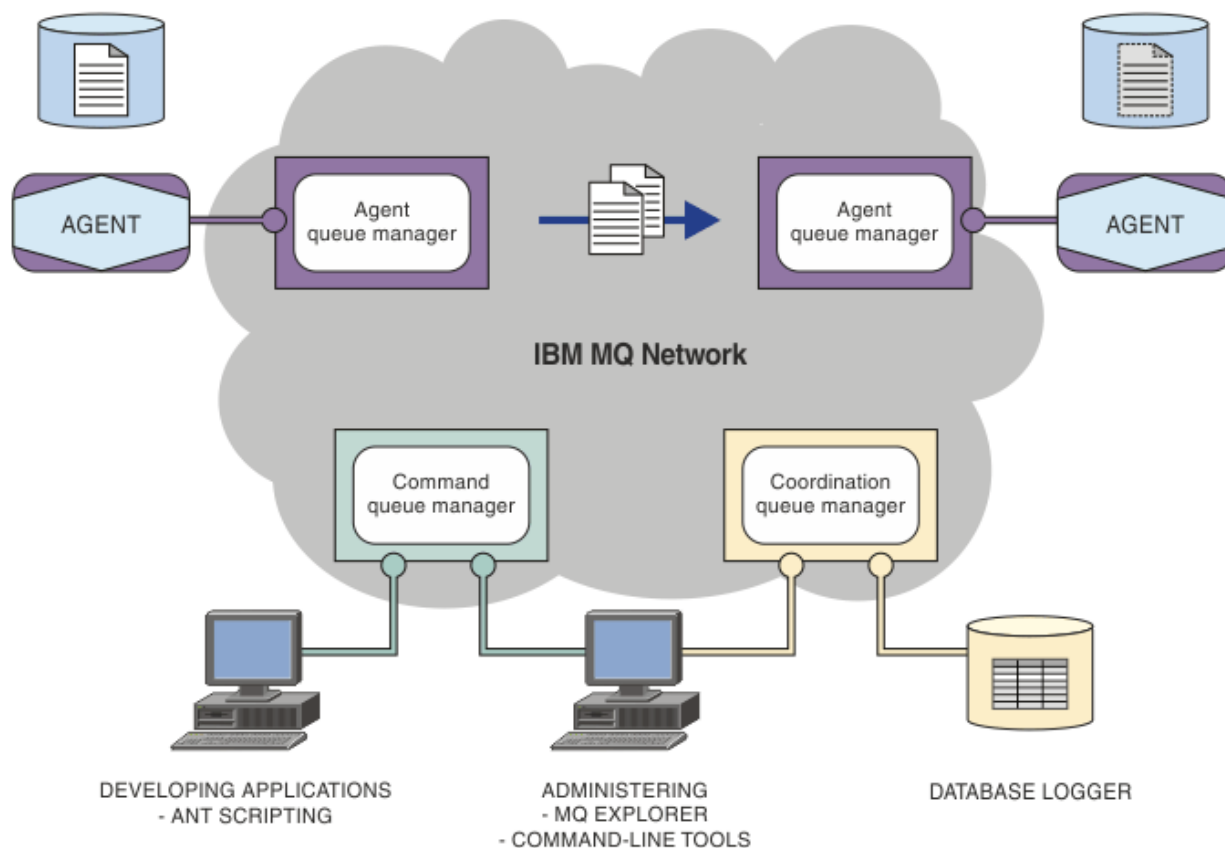

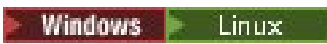


Diagram przedstawia prostą topologię produktu Managed File Transfer. Istnieją dwa agenty, każdy z nich łączy się z własnym menedżerem kolejek agenta w sieci IBM MQ. Plik jest przesyłany z agenta po jednej stronie diagramu, za pośrednictwem sieci IBM MQ, do agenta po drugiej stronie diagramu. Również w sieci IBM MQ są to menedżery kolejek koordynacji i menedżer kolejek komend. Aplikacje i narzędzia łączą się z tymi menedżerami kolejek w celu skonfigurowania, administrowania, obsługi i rejestrowania działania Managed File Transfer w sieci IBM MQ.

Produkt Managed File Transfer może być instalowany w postaci czterech różnych opcji, w zależności od systemu operacyjnego i konfiguracji ogólnej. Są to następujące opcje: Managed File Transfer Agent, Managed File Transfer Logger, Managed File Transfer Service lub Managed File Transfer Tools. Więcej informacji na ten temat zawiera sekcja [Opcje produktu Managed File Transfer](#).

Produktu Managed File Transfer można używać do wykonywania następujących zadań:

- Utwórz zarządzane przesyłanie plików
 - Windows Linux Create new file transfers from IBM MQ Explorer on Linux or Windows platforms.
 - Utwórz nowe transfery plików z wiersza komend na wszystkich obsługiwanych platformach.
 - Zintegruj funkcję przesyłania plików w narzędziu Apache Ant.
 - Napisz aplikacje, które sterują Managed File Transfer, umieszczając komunikaty w kolejkach komend agenta.
 - Zaplanuj przesyłanie plików, które odbędzie się w późniejszym czasie. Możliwe jest również wyzwolenie zaplanowanych operacji przesyłania plików w oparciu o zakres zdarzeń systemu plików, na przykład tworzony nowy plik.
 - Stale monitoruj zasób, na przykład katalog, a gdy zawartość tego zasobu spełnia jakiś predefiniowany warunek, uruchom zadanie. To zadanie może być transferem plików, skryptem produktu Ant lub zadaniem JCL.

- Przesyłanie plików do i z kolejek produktu IBM MQ .
- Przesyłanie plików do i z serwerów FTP, FTPS lub SFTP.
- Przesyłanie plików do i z węzłów Connect:Direct .
- Prześlij pliki tekstowe i binarne. Pliki tekstowe są automatycznie przekształcane między stronami kodowymi a konwencjami końca wiersza w systemach źródłowych i docelowych.
- Transfery mogą być zabezpieczone, korzystając ze standardów branżowych dla połączeń opartych na protokole SSL (Secure Socket Layer).
- Wyświetlanie transferów w toku i protokołowanie informacji o wszystkich transferach w sieci
 -  View the status of transfers in progress from IBM MQ Explorer on Linux or Windows platforms.
 -  Check the status of completed transfers by using the IBM MQ Explorer on Linux or Windows platforms.
 - Funkcja programu rejestrującego bazy danych Managed File Transfer służy do zapisywania komunikatów dziennika w bazie danych Db2 lub Oracle .

Produkt Managed File Transfer jest zbudowany w oparciu o produkt IBM MQ, który zapewnia gwarantowane, jednorazowe dostarczanie komunikatów między aplikacjami. Użytkownik może skorzystać z różnych funkcji produktu IBM MQ. Na przykład można użyć kompresji kanału, aby skompresować dane wysyłane między agentami za pośrednictwem kanałów produktu IBM MQ, a następnie użyć kanałów SSL w celu zabezpieczenia danych, które są wysyłane między agentami. Pliki są przesyłane niezawodnie i mogą tolerować awarię infrastruktury, na której odbywa się przesyłanie plików. Jeśli nastąpi przerwa w sieci, przesyłanie plików jest restartowane od miejsca, w którym zostało ono wyłączone podczas odtwarzania połączenia.

Konsolidując przesyłanie plików z istniejącą siecią produktu IBM MQ, można uniknąć wydatków na zasoby wymagane do obsługi dwóch oddzielnych infrastruktur. Jeśli nie jesteś już klientem IBM MQ, tworząc sieć IBM MQ do obsługi Managed File Transfer, budujesz kręgosłup dla przyszłej implementacji architektury SOA. Jeśli użytkownik jest już klientem IBM MQ, produkt Managed File Transfer może skorzystać z istniejącej infrastruktury produktu IBM MQ, w tym IBM MQ Internet Pass-Thru i IBM Integration Bus.

Istnieje możliwość skorzystania z rozwiązań wysokiej dostępności produktu IBM MQ w celu zwiększenia odporności konfiguracji produktu Managed File Transfer. Jeśli agenty używają replikowanych menedżerów kolejek danych (RDQMs), należy skonfigurować je tak, aby korzystały z opcji sieciowej adresu IP. Oznacza to, że agenty używają tego samego adresu IP do komunikowania się z jedną z trzech instancji RDQM, która jest obecnie uruchomiona i automatycznie ponownie nawiąże połączenie przy przełączeniu awaryjnym (patrz sekcja [Wysoka dostępność RDQM](#) oraz [Tworzenie i usuwanie zmiennego adresu IP](#)). Jeśli używane jest rozwiązanie menedżera kolejek z wieloma instancjami, aplikacje używają innego adresu IP do komunikowania się z każdą instancją, która jest obsługiwana przez ponowne połączenie klienta przy przełączeniu awaryjnym (patrz sekcja [Menedżery kolejek z wieloma instancjami](#) oraz [ponowne połączenie kanału i klienta](#)).

Produkt Managed File Transfer integruje się z wieloma innymi produktami IBM :

IBM Integration Bus

Pliki procesów, które zostały przesłane przez produkt Managed File Transfer jako część przepływu IBM Integration Bus. Więcej informacji na ten temat zawiera sekcja [Praca z produktem MFT w produkcie IBM Integration Bus](#).

IBM Sterling Connect:Direct

Przesyłanie plików do i z istniejącej sieci Connect:Direct za pomocą mostu Managed File Transfer Connect:Direct. Więcej informacji na ten temat zawiera sekcja [Most Connect:Direct](#).

IBM Tivoli Composite Application Manager

Produkt IBM Tivoli Composite Application Manager udostępnia agenta, którego można używać do monitorowania informacji publikowanych w menedżerze kolejek koordynacji.

Pojęcia pokrewne

Opcje produktu Managed File Transfer

[“MFT przegląd topologii” na stronie 305](#)

Przegląd informacji o tym, w jaki sposób agenty Managed File Transfer są połączone z menedżerem kolejek koordynacji w sieci IBM MQ .

[“W jaki sposób produkt MFT działa z produktem IBM MQ?” na stronie 305](#)

Produkt Managed File Transfer współdziela na wiele sposobów z produktem IBM MQ.

W jaki sposób produkt MFT działa z produktem IBM MQ?

Produkt Managed File Transfer współdziela na wiele sposobów z produktem IBM MQ.

- Managed File Transfer przesyła pliki między procesami agenta, dzieląc każdy plik na jeden lub większą liczbę komunikatów i przysyłając komunikaty za pośrednictwem sieci IBM MQ .
- Procesy agenta przenoszą dane plików za pomocą nietrwałych komunikatów w celu zminimalizowania wpływu na dzienniki produktu IBM MQ . Komunikując się między innymi, procesy agenta regulują przepływ komunikatów zawierających dane plikowe. Zapobiega to tworzeniu komunikatów zawierających dane plików w kolejkach transmisji produktu IBM MQ i zapewnia, że jeśli jakiegokolwiek komunikaty nietrwałe nie zostaną dostarczone, dane zbioru zostaną wysłane ponownie.
- Agenty Managed File Transfer używają wielu kolejek produktu IBM MQ . Więcej informacji na ten temat zawiera sekcja [Kolejki systemowe produktu MFT i temat systemu](#).
- Chociaż niektóre z tych kolejek są ściśle używane do użytku wewnętrznego, agent może akceptować żądania w postaci specjalnie sformatowanych komunikatów komend wysyłanych do określonej kolejki, z której odczyty są odczytywane przez agenta. Zarówno komendy wiersza komend, jak i wtyczka IBM MQ Explorer wysyłają do agenta komunikaty produktu IBM MQ w celu poinstruowania agenta w celu wykonania poszukiwanych czynności. W ten sposób można pisać aplikacje produktu IBM MQ , które wchodzi w interakcje z agentem. Więcej informacji na ten temat zawiera sekcja [Kontrolowanie systemu MFT przez umieszczanie komunikatów w kolejce komend agenta](#).
- Agenty produktu Managed File Transfer wysyłają informacje o ich stanie oraz o postępach i wynikach operacji przesyłania do menedżera kolejek produktu MQ , który został wyznaczony jako menedżer kolejek koordynacji. Ta informacja jest publikowana przez menedżer kolejek koordynacji i może być subskrybowana przez aplikacje, które mają monitorować postęp przesyłania lub przechowywać rekordy z operacji przesyłania, które wystąpiły. Zarówno komendy wiersza komend, jak i wtyczka IBM MQ Explorer mogą korzystać z opublikowanych informacji. Użytkownik może napisać aplikacje produktu IBM MQ , które korzystają z tych informacji. Więcej informacji na temat tematu, do którego są publikowane informacje, zawiera sekcja [SYSTEM.FTE](#).
- Kluczowe komponenty produktu Managed File Transfer umożliwiają menedżerom kolejek produktu IBM MQ zapisywanie i przekazywanie komunikatów. Oznacza to, że jeśli cierpisz awaryjnie, niezmiennione części infrastruktury mogą nadal przysyłać pliki. Dotyczy to menedżera kolejek koordynacji, w którym kombinacja składnic i subskrypcji trwałych i trwałych umożliwia menedżerowi kolejek koordynacji tolerowanie niedostępności bez utraty kluczowych informacji na temat dokonanych transferów plików.

MFT przegląd topologii

Przegląd informacji o tym, w jaki sposób agenty Managed File Transfer są połączone z menedżerem kolejek koordynacji w sieci IBM MQ .

Agenty Managed File Transfer wysyłają i odbierają przesłane pliki. Każdy agent ma własny zestaw kolejek w powiązaniu z menedżerem kolejek, a agent jest przyłączony do swojego menedżera kolejek w powiązaniach lub w trybie klienta. Agent może również używać menedżera kolejek koordynacji jako jego menedżera kolejek.

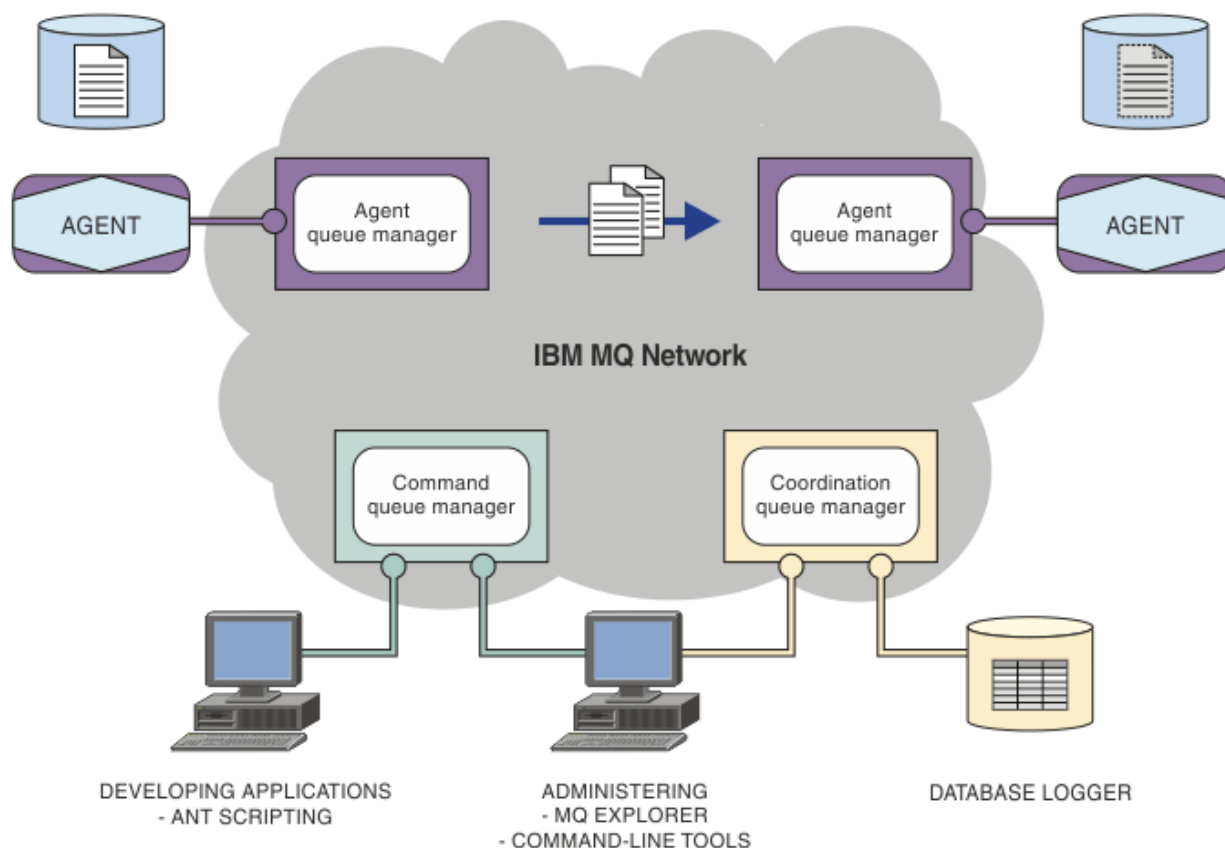
Menedżer kolejek koordynacji przekazuje informacje o kontroli i przesyłaniu plików. Menedżer kolejek koordynacji reprezentuje pojedynczy punkt zbierania informacji o agencie, statusie przesyłania i kontroli przesyłania. Menedżer kolejek koordynacji nie musi być dostępny, aby można było wykonać przelew. Jeśli menedżer kolejek koordynacji chwilowo staje się niedostępny, transfery są kontynuowane w miarę normalnego działania. Komunikaty kontroli i statusu są zapisywane w menedżerach kolejek agenta do

momentu, gdy menedżer kolejek koordynacji stanie się dostępny, a następnie może być przetwarzany jako normalny.

Agenty rejestrują się w menedżerze kolejek koordynacji i publikują swoje szczegóły w tym menedżerze kolejek. Informacje o agencie są używane przez wtyczkę Managed File Transfer w celu umożliwienia uruchamiania operacji przesyłania z serwera IBM MQ Explorer. Informacje o agencie zebrane w menedżerze kolejek koordynacji są również używane przez komendy do wyświetlania informacji o agencie i statusu agenta.

Informacje o statusie przesyłania i kontroli przesyłania są publikowane w menedżerze kolejek koordynacji. Informacje o statusie przesyłania i kontroli przesyłania są używane przez wtyczkę Managed File Transfer do monitorowania postępu operacji przesyłania danych z serwera IBM MQ Explorer. Informacje kontroli przesyłania zapisane w menedżerze kolejek koordynacji mogą być przechowywane w celu zapewnienia kontroli.

Menedżer kolejek komend jest używany do łączenia się z siecią produktu IBM MQ i jest menedżerem kolejek podłączonym do komend podczas wydawania komend produktu Managed File Transfer .



Pojęcia pokrewne

[“Managed File Transfer” na stronie 302](#)

Produkt Managed File Transfer przesyła pliki między systemami w sposób zarządzany i podlegający kontroli, niezależnie od wielkości pliku lub używanych systemów operacyjnych.

[“W jaki sposób produkt MFT działa z produktem IBM MQ?” na stronie 305](#)

Produkt Managed File Transfer współdziela na wiele sposobów z produktem IBM MQ.

Odsyłacze pokrewne

[Managed File Transfer scenariusz](#)

Przegląd produktu MFTREST API

Produkt REST API obsługuje niektóre komendy produktu Managed File Transfer , w tym listę transferów oraz szczegółowe informacje na temat agentów przesyłania plików.

Produkt IBM MQ 9.1.0 REST API zawiera opcje służące do wyświetlania listy wszystkich bieżących operacji przesyłania Managed File Transfer oraz do wysyłania zapytań o status agentów Managed File Transfer . Więcej informacji na ten temat zawiera sekcja [Pierwsze kroki z REST API MFT](#).

IBM MQ Internet Pass-Thru

Produkt IBM MQ Internet Pass-Thru (MQIPT) jest opcjonalnym komponentem produktu IBM MQ , który może być używany do implementowania rozwiązań przesyłania komunikatów między zdalnymi serwisami w sieci Internet.

V 9.2.0 W produkcie IBM MQ 9.2.0 MQIPT jest opcjonalnym komponentem produktu IBM MQ . Aby uzyskać pliki instalacyjne produktu MQIPT dla produktu IBM MQ 9.2.x, należy przejść do strony [IBM Fix Central for IBM MQ](#). Przed IBM MQ 9.2.0 produkt MQIPT był dostępny jako pakiet wsparcia.

Program IBM MQ 9.2.0 nie musi być uruchomiony, aby można było używać produktu MQIPT w produkcie IBM MQ 9.2.0 lub nowszym. Produkt MQIPT może być używany do łączenia dowolnej obsługiwanej wersji produktu IBM MQi nie ma potrzeby instalowania żadnych innych komponentów produktu IBM MQ w tej samej wersji, co produkt MQIPT.

Jeśli zakupiono uprawnienie IBM MQ , można zainstalować tyle kopii, ile jest wymagane w produkcie MQIPT. Instalacje produktu MQIPT nie są zaliczane do zakupionego uprawnienia IBM MQ . Więcej informacji na temat licencjonowania produktu IBM MQ można znaleźć w sekcji [Informacje licencyjne produktu IBM MQ](#).

Uwaga: Ta dokumentacja dotyczy produktu MQIPT w produkcie IBM MQ 9.2.0 i nowszych wersjach. Dokumentację pakietu wsparcia dla produktu MQIPT (wersja 2.1) w produkcie IBM Documentation zawiera sekcja [MQIPT \(SupportPac MS81\)](#) w dokumentacji produktu IBM MQ 9.0 .

Uwaga: Jeśli używany jest produkt MQIPT 2.1 lub wcześniejszy, zaleca się aktualizację do wersji MQIPT dla produktu IBM MQ 9.2.x, ponieważ data zakończenia wsparcia dla pakietu wsparcia dla produktu MQIPT wynosi 30th września 2020.

Produkt IBM MQ Internet Pass-Thru działa jako usługa autonomiczna, która może odbierać i przekazywać przepływy komunikatów produktu IBM MQ , między dwoma menedżerami kolejek produktu IBM MQ lub między klientem IBM MQ a menedżerem kolejek produktu IBM MQ .

Produkt MQIPT włącza to połączenie, gdy klient i serwer nie znajdują się w tej samej sieci fizycznej.

Jedną lub więcej instancji produktu MQIPT można umieścić w ścieżce komunikacyjnej między dwoma menedżerami kolejek produktu IBM MQ lub między klientem IBM MQ a menedżerem kolejek produktu IBM MQ . Instancje serwera MQIPT pozwalają na wymianę komunikatów między dwoma systemami IBM MQ bez konieczności bezpośredniego połączenia TCP/IP między tymi dwoma systemami. Jest to przydatne, jeśli konfiguracja zapory firewall uniemożliwia bezpośrednie połączenie TCP/IP między tymi dwoma systemami.

Produkt MQIPT nasłuchuje na jednym lub większej liczby portów TCP/IP dla połączeń przychodzących, które mogą przenosić zwykłe komunikaty produktu IBM MQ , IBM MQ komunikaty tunelowane wewnątrz protokołu HTTP lub komunikaty zaszyfrowane przy użyciu protokołu TLS (Transport Layer Security) lub protokołu SSL (Secure Sockets Layer). Produkt MQIPT może obsługiwać wiele współbieżnych połączeń.

Kanał IBM MQ , który tworzy początkowe żądanie połączenia TCP/IP, jest określany jako *program wywołujący*, kanał, z którym próbuje się połączyć jako *responder*, oraz menedżer kolejek, który ostatecznie próbuje nawiązać połączenie jako *docelowy menedżer kolejek*.

Produkt MQIPT przechowuje dane w pamięci, ponieważ przekazuje je ze źródła do miejsca docelowego. Żadne dane nie są zapisywane na dysku (z wyjątkiem pamięci stronicowanej na dysk przez system operacyjny). Jedynym razem, gdy produkt MQIPT uzyskuje dostęp do dysku, jest odczytywanie jego pliku konfiguracyjnego oraz zapisywanie rekordów dziennika i śledzenia połączeń.

Pełny zakres typów kanałów produktu IBM MQ może być dokonany za pośrednictwem jednej lub wielu instancji produktu MQIPT. Obecność produktu MQIPT w ścieżce komunikacyjnej nie ma wpływu na parametry funkcjonalne podłączonych komponentów produktu IBM MQ, ale może to mieć wpływ na wydajność przesyłania komunikatów.

Produkt MQIPT może być używany w połączeniu z produktem IBM MQ i produktem IBM Integration Bus, zgodnie z opisem w sekcji [“Możliwe konfiguracje produktu MQIPT”](#) na stronie 311.

Informacje na temat instalowania produktu MQIPT zawiera sekcja [Instalowanie produktu MQIPT](#).

Zadania pokrewne

[Konfigurowanie produktu IBM MQ Internet Pass-Thru](#)

[Administrowanie i konfigurowanie produktu IBM MQ Internet Pass-Thru](#)

Odsyłacze pokrewne

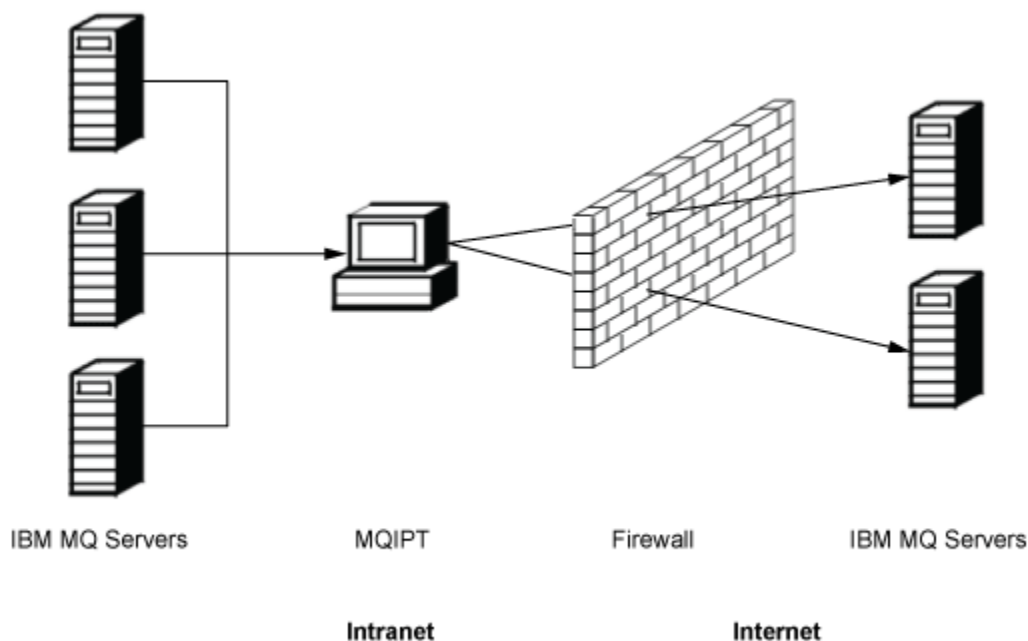
[Skorowidz konfiguracji produktu IBM MQ Internet Pass-Thru](#)

Użycie produktu MQIPT

Istnieje wiele potencjalnych zastosowań dla produktu IBM MQ Internet Pass-Thru (MQIPT).

MQIPT może być używany jako koncentrator kanałów

Za pomocą programu MQIPT w ten sposób kanały do lub z wielu oddzielnych hostów mogą być wyświetlane na firewallu tak, jakby były one wszystkie lub pochodzące z hosta MQIPT. Ułatwia to definiowanie reguł filtrowania firewalla i zarządzanie nimi.

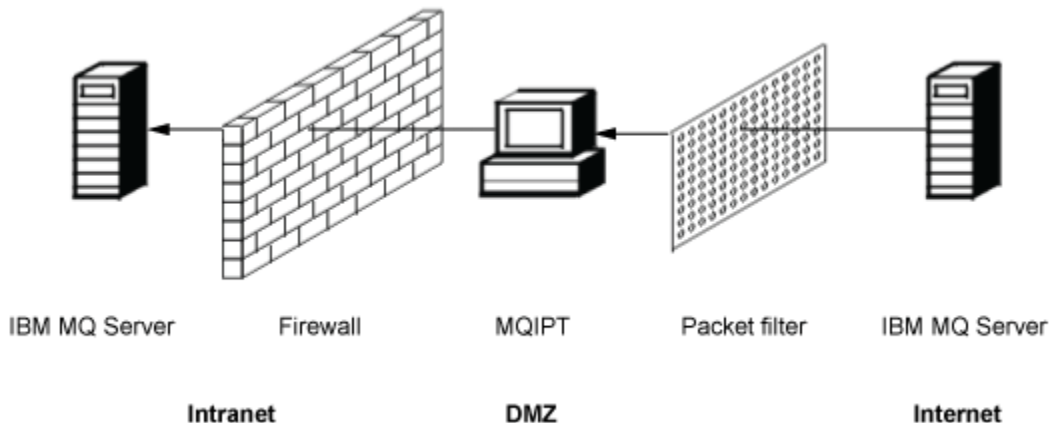


Rysunek 80. Przykład MQIPT jako koncentratora kanału

Produkt MQIPT można umieścić w strefie DMZ, aby zapewnić pojedynczy punkt dostępu.

Jeśli produkt MQIPT jest umieszczony w firewallu DMZ (konfiguracja firewalla do zabezpieczania lokalnych sieci obszarów), na komputerze z znanym i zaufanym adresem IP (Internet Protocol), program MQIPT może nasłuchiwać przychodzących połączeń kanału IBM MQ, które może następnie przesyłać do zaufanego intranetu; wewnętrzny firewall musi zezwalać temu zaufanemu komputerowi na nawiązywanie

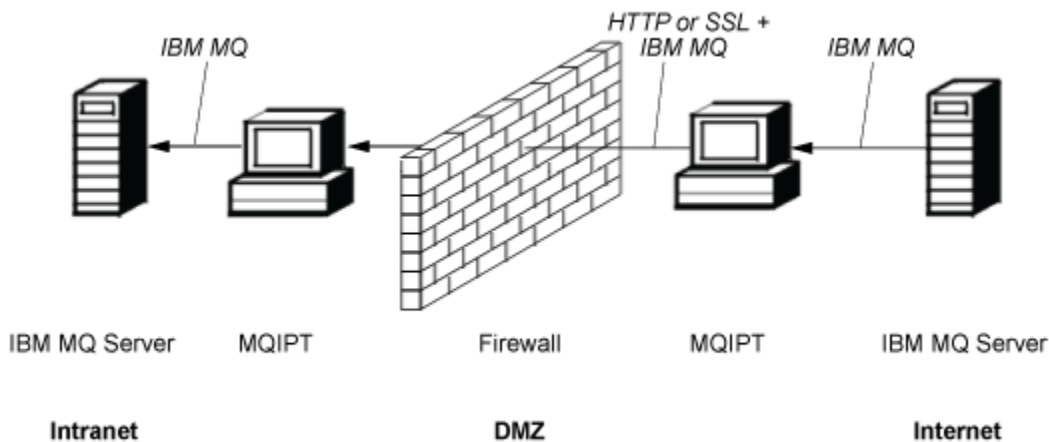
połączeń przychodzących. W tej konfiguracji produkt MQIPT zapobiega dostępowi do zewnętrznych żądań dostępu od odbierania prawdziwych adresów IP komputerów w zaufanym intranecie. W ten sposób produkt MQIPT udostępnia pojedynczy punkt dostępu.



Rysunek 81. Przykład produktu MQIPT w firewallu DMZ

Program MQIPT może komunikować się za pomocą tunelowania HTTP

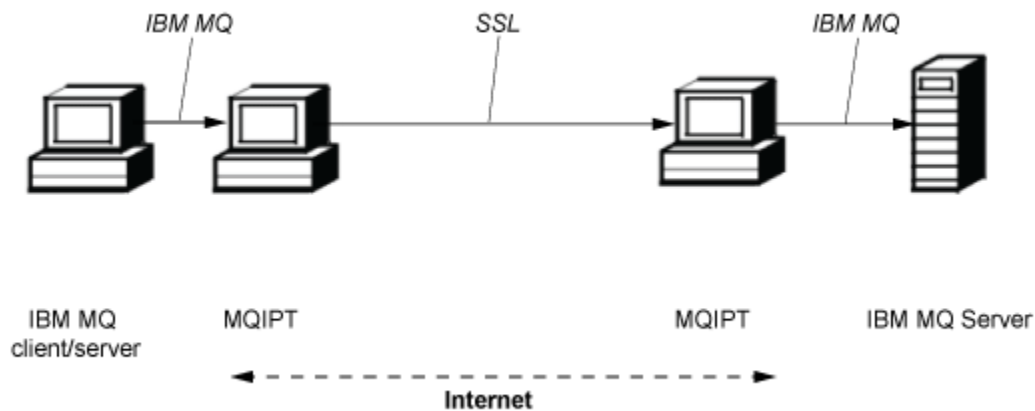
Jeśli w wierszu wdrożono dwie instancje produktu MQIPT , mogą one komunikować się za pomocą protokołu HTTP. Funkcja tunelowania HTTP umożliwia przesyłanie żądań za pośrednictwem firewalli, korzystając z istniejących proxy HTTP. Pierwsza MQIPT wstawia protokół IBM MQ do protokołu HTTP, a drugi wyodrębnia protokół IBM MQ z opakowania HTTP, a następnie przekazuje go do docelowego menedżera kolejek.



Rysunek 82. Przykład tunelowania MQIPT i HTTP

Program MQIPT może szyfrować komunikaty

Jeśli serwer MQIPT został skonfigurowany tak, jak w poprzednim przykładzie, żądania mogą być szyfrowane przed transmisją przez firewalla. Pierwsza MQIPT szyfruje dane, a druga deszyfruje za pomocą protokołu SSL/TLS przed wystaniem go do docelowego menedżera kolejek.



Rysunek 83. Przykład MQIPT i SSL/TLS

Jak działa MQIPT

W najprostszej konfiguracji produkt MQIPT działa jako serwer przekazujący protokołu IBM MQ . Następuje on na porcie TCP/IP i akceptuje żądania połączeń z kanałów produktu IBM MQ .

Jeśli zostanie odebrane poprawnie utworzone żądanie, produkt MQIPT nawiązuje dalsze połączenie TCP/IP między sobą i docelowym menedżerem kolejek produktu IBM MQ . Następnie przekazuje on wszystkie pakiety protokołu, które otrzymuje od połączenia przychodzącego do docelowego menedżera kolejek, i zwraca pakiety protokołu z docelowego menedżera kolejek z powrotem do oryginalnego połączenia przychodzącego.

Nie ma żadnych zmian w protokole IBM MQ (klient/serwer lub menedżer kolejek do menedżera kolejek), ponieważ żaden z nich nie jest bezpośrednio świadomy obecności pośrednika. Nowe wersje kodu klienta lub serwera IBM MQ nie są wymagane.

Aby można było używać produktu MQIPT, kanał programu wywołującego musi być skonfigurowany tak, aby używała nazwy hosta i portu serwera MQIPT , a nie nazwy hosta i portu docelowego menedżera kolejek. Wartość ta jest definiowana za pomocą właściwości **CONNNAME** kanału IBM MQ . Program MQIPT odczytuje dane przychodzące i po prostu przekazuje je do docelowego menedżera kolejek. Inne pola konfiguracji, takie jak ID użytkownika i hasło w kanale klient/serwer, są podobnie przekazywane do docelowego menedżera kolejek.

Wiele menedżerów kolejek

Produkt MQIPT może być używany w celu umożliwienia dostępu do więcej niż jednego docelowego menedżera kolejek. Aby to działało, musi istnieć mechanizm poinformowania MQIPT , z którym menedżer kolejek ma nawiązać połączenie, dlatego program MQIPT używa przychodzącego numeru portu TCP/IP w celu określenia menedżera kolejek, z którym ma zostać nawiązane połączenie.

W związku z tym można skonfigurować program MQIPT do nasłuchiwania na wielu portach TCP/IP. Każdy port nasłuchiwania jest odwzorowany na docelowy menedżer kolejek za pomocą *tras* serwera MQIPT . Można zdefiniować do 100 takich tras, które kojarzą nasłuchiwanie portu TCP/IP z nazwą hosta i portem docelowego menedżera kolejek. Oznacza to, że nazwa hosta (adres IP) docelowego menedżera kolejek nigdy nie jest widoczna dla kanału inicjującego. Każda trasa może obsługiwać wiele połączeń między portem nasłuchiwania i miejscem docelowym, a każde połączenie działa niezależnie.

MQIPT plik konfiguracyjny

Produkt MQIPT korzysta z pliku konfiguracyjnego o nazwie `mqipt.conf`. Ten plik zawiera definicje wszystkich tras i powiązanych z nimi właściwości. Więcej informacji na temat produktu `mqipt.conf` zawiera sekcja [Administrowanie i konfigurowanie produktu IBM MQ Internet Pass-Thru](#) .

Po uruchomieniu programu MQIPT uruchamiana jest każda trasa, która jest wymieniona w pliku konfiguracyjnym. Komunikaty są zapisywane w konsoli systemowej, w której wyświetlane są statusy poszczególnych tras. Jeśli dla trasy zostanie wyświetlony komunikat MQCPI078, trasa ta jest gotowa do akceptowania żądań połączenia.

Możliwe konfiguracje produktu MQIPT

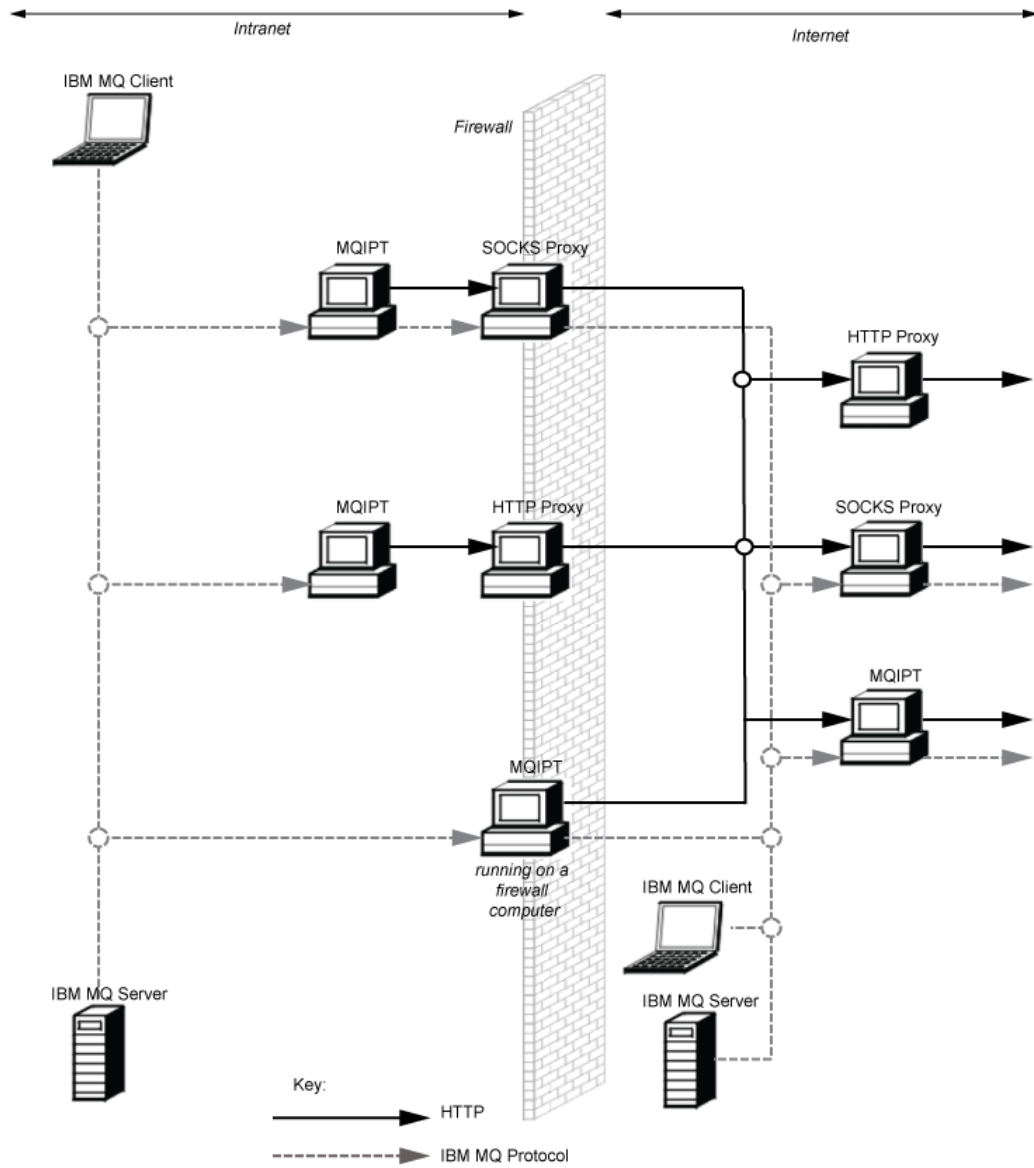
Produkt MQIPT może być używany w połączeniu z produktem IBM MQ i produktem IBM Integration Bus.

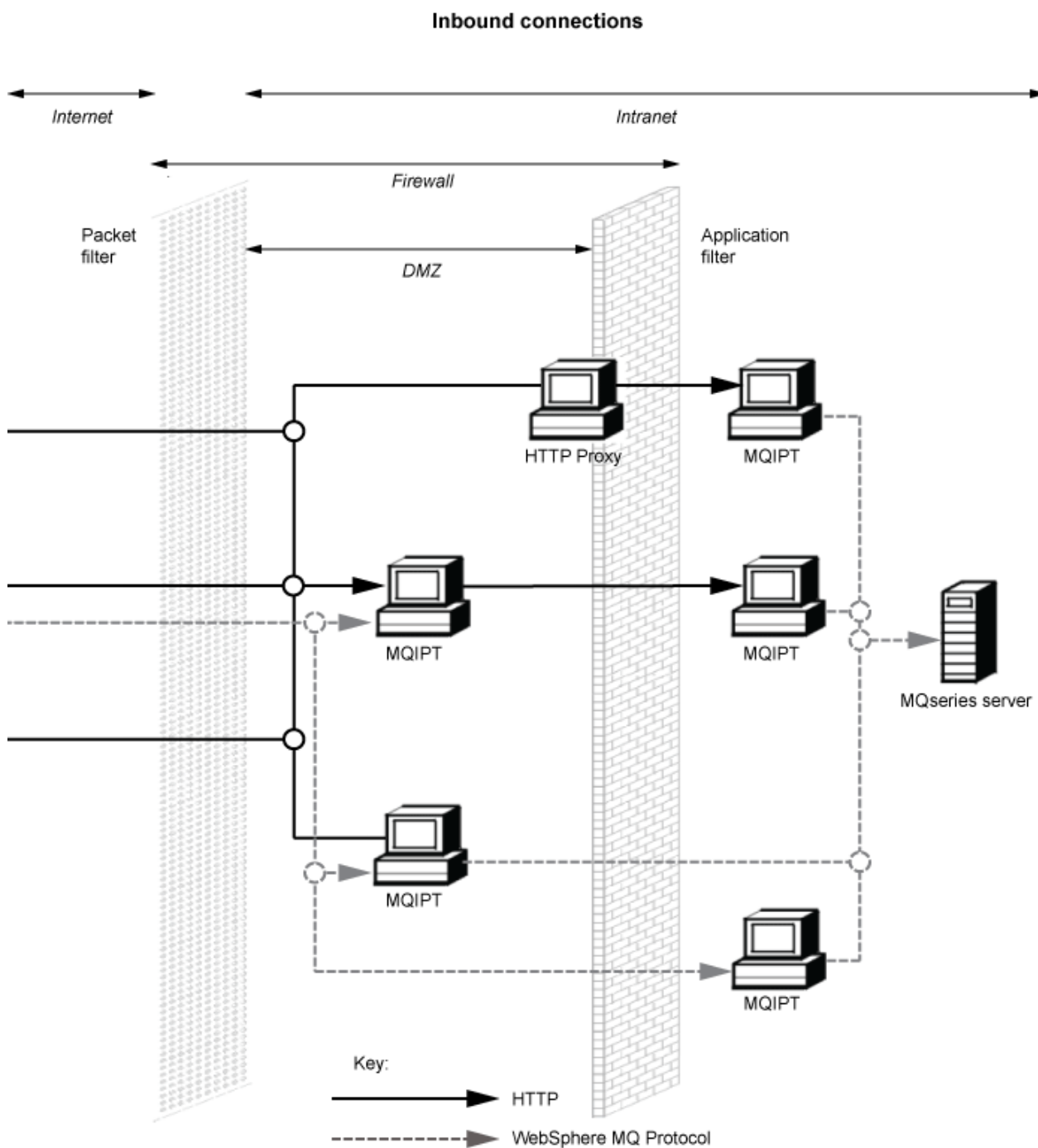
Na poniższym rysunku wieloczęściowym przedstawiono wiele możliwych konfiguracji produktu MQIPT w topologii produktu IBM MQ. Ilustruje on różne sposoby wysyłania komunikatów przez produkt MQIPT. Przedstawia on klienty i serwery w intranecie, w firewallu i w Internecie poza firewallem, przekazując komunikaty do serwera MQIPT, serwera proxy HTTP lub serwera proxy SOCKS, który przekazuje je do wiadomości.

Komunikaty są odbierane przez serwer proxy produktu MQIPT lub proxy HTTP w strefie DMZ przed przesłaniem komunikatu przez firewall przychodzący do serwera.

Należy zauważyć, że serwer proxy HTTP, serwer proxy SOCKS oraz komputery MQIPT po stronie intranetowej firewalla reprezentują możliwość połączenia wielu komputerów w sieci w sieci Internet. Na przykład komputer z systemem MQIPT może komunikować się za pośrednictwem jednego lub większej liczby komputerów proxy SOCKS lub HTTP albo innych komputerów MQIPT przed osiągnięciem jego celu.

Outbound connections





Kompatybilne konfiguracje

Zgodne scenariusze połączeń, w których klient lub menedżer kolejek produktu IBM MQ komunikuje się z produktem MQIPT. Ta sama lub druga trasa produktu MQIPT jest używana do komunikowania się z docelowym menedżerem kolejek.

Zgodne konfiguracje z pojedynczą trasą MQIPT

Aby komunikować się z produktem IBM MQ, można użyć pojedynczej trasy MQIPT .

Kolumny w programie [Tabela 25 na stronie 314](#) zawierają następujące informacje:

1. Protokół używany między IBM MQ a trasą MQIPT . Połączenie może zostać utworzone przez klienta lub menedżera kolejek produktu IBM MQ i może być używane zarówno w formacie IBM MQ Formats, jak i protokołach (FAP) lub SSL/TLS.
2. Tryb, w którym działa trasa MQIPT . Format komunikacji w sieci Internet między MQIPT a IBM MQ jest określany przez konfigurację trasy MQIPT . Należy zauważyć, że w przypadku, gdy tabela zawiera protokół SSL, można również użyć protokołu TLS.
3. Protokół używany między trasą MQIPT a docelowym menedżerem kolejek.

| 1. IBM MQ Protokół źródła | 2. Tryb trasy MQIPT | 3. IBM MQ Protokół docelowy |
|----------------------------------|----------------------------|------------------------------------|
| FAP | FAP-proxy (domyślnie) | FAP |
| | FAP-serwer i klient SSL | Protokół SSL/TLS |
| Protokół SSL/TLS | SSL-proxy | Protokół SSL/TLS |
| | SSL-serwer i klient FAP | FAP |
| | SSL-serwer i klient SSL | Protokół SSL/TLS |

Zgodne konfiguracje z więcej niż jedną trasą MQIPT

Użytkownik może zdecydować się na użycie więcej niż jednej trasy, na jednej lub kilku instancjach produktu MQIPT, w celu komunikowania się z produktem IBM MQ.

Kolumny w programie [Tabela 26 na stronie 315](#) zawierają następujące informacje:

1. Protokół używany między IBM MQ a pierwszą trasą MQIPT . Połączenie może zostać utworzone przez klienta lub menedżera kolejek produktu IBM MQ i może być używane zarówno w formacie IBM MQ Formats, jak i protokołach (FAP) lub SSL/TLS.
2. Tryb, w którym działa pierwsza trasa MQIPT . Format komunikacji w sieci Internet między MQIPT a IBM MQ jest określany przez konfigurację trasy MQIPT . Należy zauważyć, że w przypadku, gdy tabela zawiera protokół SSL, można również użyć protokołu TLS.
3. Tryb, w którym działa druga trasa MQIPT .
4. Protokół używany między drugą trasą MQIPT a docelowym menedżerem kolejek.

Tabela 26. Poprawne konfiguracje z wieloma instancjami produktu MQIPT

| 1. IBM MQ Protokół źródła | 2. Tryb pierwszej trasy MQIPT | 3. Tryb drugiej trasy MQIPT | 4. IBM MQ Protokół docelowy |
|-------------------------------|-------------------------------|-----------------------------|-----------------------------|
| FAP (domyślny obszar roboczy) | FAP-proxy (domyślnie) | FAP-proxy (domyślnie) | FAP |
| | FAP-serwer i klient SSL | SSL-proxy | Protokół SSL/TLS |
| | | SSL-serwer i klient FAP | FAP |
| | | SSL-serwer i klient SSL | Protokół SSL/TLS |
| | Klient HTTP | HTTP-serwer i klient SSL | Protokół SSL/TLS |
| | HTTPS-klient | HTTPS-serwer i klient SSL | Protokół SSL/TLS |
| | Klient HTTP | Serwer HTTP | FAP |
| HTTPS-klient | HTTPS-serwer | FAP | |
| Protokół SSL/TLS | SSL-proxy | SSL-proxy | Protokół SSL/TLS |
| | | SSL-serwer i klient FAP | FAP |
| | | SSL-serwer i klient SSL | Protokół SSL/TLS |
| | Klient HTTP | Serwer HTTP | FAP |
| | HTTPS-klient | HTTPS-serwer | Protokół SSL/TLS |
| | Klient HTTP | HTTP-serwer i klient SSL | FAP |
| | HTTPS-klient | HTTPS-serwer i klient SSL | Protokół SSL/TLS |

Obsługiwane konfiguracje kanałów

Obsługiwane są wszystkie typy kanałów produktu IBM MQ, ale konfiguracja jest ograniczona do połączeń TCP/IP. W przypadku klienta lub menedżera kolejek produktu IBM MQ program MQIPT jest wyświetlany, jeśli jest to docelowy menedżer kolejek. Jeśli konfiguracja kanału wymaga hosta docelowego i numeru portu, podana jest nazwa hosta MQIPT i numer portu nastuchiwania.

Kanały klient/serwer

Program MQIPT nastuchuje przychodzących żądań połączeń klientów, a następnie przekazuje je za pomocą tunelowania HTTP, protokołu SSL/TLS lub jako standardowych pakietów protokołu IBM MQ. Jeśli program MQIPT używa tunelowania HTTP lub protokołu SSL/TLS, przekazuje je w połączeniu z drugim MQIPT. Jeśli nie jest używany tunelowanie HTTP, przekazuje je w połączeniu z tym, co widzi jako docelowy menedżer kolejek (mimo że może to być dalej MQIPT). Gdy docelowy menedżer kolejek zaakceptuje połączenie klienta, pakiety są przekazywane między klientem a serwerem.

Kanały nadawca/odbiorcze klastra

Jeśli program MQIPT odbiera przychodzące żądanie z kanału nadawczego klastra, zakłada, że menedżer kolejek został włączony (SOCKS), a prawdziwy adres docelowy zostanie uzyskany podczas procesu ręcznego wykonywania SOCKS. Przekazuje on żądanie do następnego menedżera kolejek produktu MQIPT lub docelowego w taki sam sposób, jak w przypadku kanałów połączenia klienckiego. Obejmuje to również automatycznie zdefiniowane kanały nadawcze klastra.

Nadawca/odbiorca

Jeśli program MQIPT odbiera przychodzące żądanie z kanału nadawczego, przekazuje go do następnego MQIPT lub docelowego menedżera kolejek w taki sam sposób, jak w przypadku kanałów połączenia klienckiego. Docelowy menedżer kolejek sprawdza poprawność żądania przychodzącego i w razie potrzeby uruchamia kanał odbiorczy. Wszystkie połączenia między nadawcą a kanałem odbiorczym (włącznie z przepływami bezpieczeństwa) są przekazywane.

Żądający/serwer

Ta kombinacja jest obsługiwana w taki sam sposób, jak w poprzednich konfiguracjach. Sprawdzenie poprawności żądania połączenia jest wykonywane przez kanał serwera w docelowym menedżerze kolejek.

Żądający/nadawca

Konfiguracja "zwrotna" może być używana, jeśli dwa menedżery kolejek nie mogą nawiązać bezpośrednich połączeń między sobą, ale obie mogą łączyć się z programem MQIPT i akceptować połączenia z nimi.

Serwer/requester i serwer/odbiornik

Są one obsługiwane przez produkt MQIPT w taki sam sposób, w jaki obsługuje konfigurację serwera Sender/Receiver .

Warunki zakończenia kanału i niepowodzenia

Gdy program MQIPT wykryje zamknięcie kanału (normalny lub nieprawidłowy) kanału IBM MQ , propaguje zamknięcie kanału. Jeśli trasa zostanie zamknięta za pomocą programu MQIPT, wszystkie kanały przechodnie przez tę trasę są zamykane.

Produkt MQIPT udostępnia opcjonalną funkcję limitu czasu bezczynności. Jeśli program MQIPT wykryje, że kanał był bezczynny przez okres przekraczający limit czasu, wykonuje natychmiastowe zamknięcie tych dwóch połączeń w wątpliwość.

Systemy IBM MQ na obu końcach kanału obserwują te nieprawidłowe warunki zamknięcia, albo jako awarie sieci, albo jako zakończenie kanału przez ich partnera. Kanał jest następnie w stanie restartować i odzyskiwać (jeśli błąd występuje podczas okresu wątpliwego protokołu), tak jak w przypadku, gdy nie jest używany produkt MQIPT .

Bezpieczeństwo komunikatów

Zarządzanie kolejkami rozproszonymi IBM MQ zapewnia, że komunikaty są dostarczane poprawnie. Jest to nadal przypadek, gdy produkt MQIPT występuje między dwoma końcami kanału. Produkt MQIPT nie przechowuje żadnych danych komunikatu ani nie uczestniczy w procedurze punktu synchronizacji, która zapewnia poprawne dostarczanie komunikatów.

W przypadku korzystania z szybkich i nietrwałych komunikatów produktu IBM MQ , jeśli trasa MQIPT nie powiedzie się lub zostanie zrestartowana, gdy komunikat IBM MQ jest w trakcie tranzytu, komunikat może zostać utracony. Przed zrestartowaniem trasy należy upewnić się, że wszystkie kanały produktu IBM MQ korzystające z trasy MQIPT są nieaktywne.

Więcej informacji na temat bezpieczeństwa komunikatów w produkcie IBM MQ zawiera sekcja [Bezpieczeństwo komunikatów](#).

Menedżery kolejek o wielu instancjach i wysoka dostępność

Produkt MQIPT może być używany z menedżerami kolejek z wieloma instancjami w środowiskach wysokiej dostępności.

MQIPT nie ma trwałego stanu i dlatego nie ma korzyści z powodu awarii serwera MQIPT do innego systemu. Zamiast tego istnieje wiele instancji produktu MQIPT z identycznymi plikami konfiguracyjnymi mqipt.conf działającymi w różnych systemach. Monitoruj każdą instancję produktu MQIPT pod kątem dostępności, a następnie zrestartuj ją (w tym samym systemie), jeśli to konieczne. Udostępnia on zestaw identycznych instancji produktu MQIPT , które mogą być używane do kierowania połączeń. Następnie należy upewnić się, że produkt IBM MQ może kierować połączenia do produktu MQIPT i że produkt MQIPT może przekazywać te połączenia do docelowego menedżera kolejek.

Kanały wychodzące IBM MQ mogą być kierowane do dostępnej instancji produktu MQIPT na różne sposoby, na przykład:

- Użyj systemu równoważenia obciążenia lub routera o wysokiej dostępności, takiego jak IBM Network Dispatcher (Program rozsyłający sieci) z produktu WebSphere Edge Components.

- Podaj wiele nazw połączeń w definicji kanału produktu IBM MQ , używając listy rozdzielanej przecinkami. Następnie program IBM MQ próbuje połączyć się z każdym adresem MQIPT z kolei do momentu znalezienia dostępnej instancji produktu MQIPT .

Należy również kierować połączenia z produktu MQIPT do docelowego menedżera kolejek. Jeśli konfiguracja wysokiej dostępności zapewnia, że adres IP kończy się niepowodzeniem z docelowym menedżerem kolejek, nie jest wymagana specjalna konfiguracja produktu MQIPT : należy określić docelowy adres IP we właściwości trasy **Destination** i zezwolić na przeniesienie adresu IP z menedżerem kolejek na operację przełączenia awaryjnego.

Jeśli jednak adres IP menedżera kolejek ulegnie zmianie po przełączeniu awaryjnym, należy zaaranżować MQIPT w celu przekazania połączenia do właściwego miejsca docelowego. Można to zrobić na jeden z kilku sposobów:

- Napisz wyjście routingu, które sprawdza, który adres IP i numer portu są dostępne, a następnie nadpisz miejsce docelowe trasy dla każdego połączenia. Niektóre przykładowe wyjścia routingu są dostarczane z produktem MQIPT; mogą być one przystosowane do tego celu.
- Aby przekierować połączenie, należy użyć systemu równoważenia obciążenia o wysokiej dostępności.
- Zdefiniuj wiele tras MQIPT , po jednym dla każdego adresu IP i portu, na którym może być uruchomiony menedżer kolejek. Następnie należy kierować połączenia produktu IBM MQ do różnych tras produktu MQIPT , na przykład przez umieszczenie wszystkich adresów IP trasy i numerów portów w postaci listy rozdzielanej przecinkami w nazwie połączenia kanału wychodzącego.

Ważne jest również dostrojenie wszystkich komponentów typu end-to-end w ścieżce sieciowej:

1. Próby nawiązania połączenia z niedostępnymi systemami muszą się nie powieść szybko, aby próby ponownego połączenia mogły zostać przeniesione do pierwszego dostępnego miejsca docelowego.

W przypadku tras SSL produktu MQIPT dostrój właściwość trasy **SSLClientConnectTimeout** , aby upewnić się, że połączenie z połączeniem nie powiodło się dla niedostępnych miejsc docelowych. Szczegółowe informacje na temat dostrajania parametrów produktu IBM MQ zawiera dokumentacja produktu IBM MQ . Szczegółowe informacje na temat strojenia TCP/IP dla systemu operacyjnego można znaleźć w dokumentacji systemu operacyjnego. We wszystkich przypadkach nieudane próby nawiązania połączenia powinny szybko zwrócić awarię sieci (na przykład pakiet resetowania TCP) lub powinny być bez zbędnej zwłoki czas.

2. Aktywne połączenia z systemem, który uległ awarii, muszą zostać natychmiast odcięte, tak aby możliwe było nawiąwanie nowych połączeń.

Należy również rozważyć wpływ przełączenia awaryjnego w czasie, gdy połączenia aktywnie korzystają z produktu MQIPT. Prawdopodobnie połączenia sieciowe zostaną odcięte podczas przełączania awaryjnego. W przypadku aplikacji klienckich można użyć funkcji automatycznego ponownego połączenia klienta IBM MQ w celu ponownego ustanowienia zerowanych połączeń. W przypadku kanałów komunikatów można określić krótki odstęp czasu między ponownymi próbami, tak aby kanał ponownie łączył się ponownie. Zapoznaj się z dokumentacją produktu IBM MQ , aby uzyskać więcej informacji na temat automatycznego ponownego nawiązywania połączenia z klientem oraz konfiguracji ponowień kanału komunikatów.

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: